Iraqi Journal for Computers and Informatics
Information Technology and Communications University
Vol. 51, No. 2, 2025, pp. 56-69

DOI: https://doi.org/10.25195/ijci.v51i2.546
Print ISSN: 2313-190X, Online ISSN: 2520-4912

Research Article

LLM Hallucination: The Curse That Cannot Be Broken

Hussein Al-Mahmood
College of Arts, Department of Translation
University of Basrah
Basrah, Iraq
artpg.hussein.khudhair@uobasrah.edu.iq

ARTICLEINFO
Article History
Received: 22/1/2025
Accepted: 02/04/2025
Published: 25/08/2025

ABSTRACT

Artificial intelligence chatbots (e.g., ChatGPT, Claude, and Llama, etc.), also known as large language models (LLMs), are continually evolving to be an essential part of the digital tools we use, but are plagued with the phenomenon of hallucination. This paper gives an overview of this phenomenon, discussing its different types, the multi-faceted reasons that lead to it, its impact, and the statement regarding the inherent nature of current LLMs that make hallucinations inevitable. After examining several techniques, each chosen for their different implementation, to detect and mitigate hallucinations, including enhanced training, tagged-context prompts, contrastive learning, and semantic entropy analysis, the work concludes that none are efficient to mitigate hallucinations when they occur. The phenomenon is here to stay, hence calling for robust user awareness and verification mechanisms, stepping short of absolute dependence on these models in healthcare, journalism, legal services, finance, and other critical applications that require accurate and reliable information to ensure informed decisions.

Keywords:* *AI; artificial intelligence; hallucination; large language models; LLM

1. INTRODUCTION

Large language models (LLMs) are some of the most advanced artificial intelligence models due to their ability to comprehend and produce human-like text. Trained on enormous amounts of data, LLMs can perform a wide range of language-related tasks. The emergence of LLMs in the scientific community, including their development and application, has grown due to the new pivotal role they have taken in society. The number of publications concerning LLMs has risen dramatically, steadily from 2017 to 2023, with a sharp rise between 2019 and 2020, which was probably due to the public release of advanced LLM models and increasing interest in transformer-based NLP algorithms [1]. A spike in the use of artificial intelligence was also noted within academic settings. A recent study at the University of Duhok found that one-third of the faculty was engaged with the use of AI tools [2].

LLMs based on the original Encoder-Decoder Transformer [3], BERT [4], and other architectures are growing more and more capable as the size of parameters increases. With models doubling in size every 4 months, driven by the availability of more data, access to powerful hardware, and improved training algorithms [5]. However, the reliability of LLMs in critical fields is still continually questioned due to the occasional hallucinatory behavior that yields major factual errors. The impact of these hallucinations affects any domain that requires accurate and reliable information, such as medicine, banking, finance, law, and clinical settings. In these sectors, reliability and accuracy are of utmost importance; thus, any form of hallucination, whether in data, analysis, or decision-making, can have substantial and detrimental impacts on outcomes and operations [6].

Although several strategies to mitigate hallucinations targeting specific sectors have been proposed, such as using finetuning and various types of database retrieval methods [7], or as simple as using a dataset to benchmark the performance of a model in a specific domain [8], [9], these approaches are not robust. For instance, Chatlaw [7] continues to exhibit hallucinations, manifesting as fabrication of nonexistent legal provisions. On a different approach, domain experts can provide valuable insights into specific domains and help finetune models, develop evaluation metrics and datasets to detect hallucinations, and help AI researchers adapt models to the domain-specific knowledge and context. However, this may not always be viable at scale. For instance, Cui *et al.* [7] assert that the limited availability of legal professionals and the high cost of their services frequently impede access to these services.

The misinformation risk of hallucinations is significant. One big concern may be the generation of "imitative falsehoods," where LLMs spit out inaccuracies present in their training data, leading to incorrect outputs in terms of factuality. Moreover, LLMs can exhibit duplication bias because of their memorization capabilities, giving outputs that prioritize over-memorized information rather than accurate content. The other thing to worry about might be how LLMs inherit and propagate social biases from their training data, such as gender biases and nationality biases that could arise context inconsistency hallucinations and biased outputs [10]. More importantly, it should be reemphasized that the significance of hallucinations is tied to the domain in which it affects. For example, in robotics applications, LLMs apply to automated sensing and actuation. Thus, hallucinatory behavior here may result in dangerous real-world consequences. This requires the establishment of bounds within which a given LLM would be safe and not result in adverse, unintended consequences [11]. Such risks raise the need for verification mechanisms and human oversight when deploying LLMs in such areas.

The present work aims to investigate the pivotal factors that lead to the rise of hallucinations in LLMs, sheds light on the mechanism and conditions involved in their manifestation, and classifies the various hallucinations based on their causes and knowledge dependencies. Furthermore, the work overviews novel state-of-the-art domain-agnostic strategies aimed at mitigating the adverse effects of hallucinations.

2. LLM HALLUCINATION

Hallucination in LLMs involves such cases where generated output, though plausible, is factually false. It is one of the major defects in existing LLMs and is considered one of the strongest constraints on how they can be used responsibly. This is the assumption here: the model could generate output that is not a fact, even if it might seem coherent and credible. This is rooted in their training objective to predict the next word given some context based on the pattern observed in the data they have seen and not to validate the facts of the generated information [12]. According to Simhi *et al.* [13], ignorance and error may be contrasted to analyze hallucinations:

1. Ignorance (HK−): This class of hallucination happens when a model does not have sufficient information to arrive at an accurate answer. In that respect, the model is unable to have the required knowledge in its parameters to return ungrounded or wrong outputs. Here, the suggested method may be referring to external knowledge sources or abstain from providing an answer. An example of an HK− hallucination is attempting to answer a question about an event that happened in 2025, whereas the knowledge cutoff of the model is in 2024.
2. Error (HK+): This type of hallucination happens even when the model knows the relevant facts and might give a correct answer under one prompt, but on changing the prompt to be a little different, the same fact is said incorrectly. In such cases, the model actually knows the fact and has it within its parameters; however, it cannot manage to make use of the knowledge properly. Remediation for this type may consist of internal intervention on computation to obtain the correct result. HK+ hallucination examples can include diverting from the knowledge and instruction contained within the prompt, or failing to retrieve the appropriate information from its parameters during inference.

The distinction between these two kinds is crucial to developing efficient detection and mitigation strategies because they are fundamentally different problems that require different solutions. To distinguish between the two cases, Simhi *et al.* [13] introduces WACK – a methodology for constructing model-specific datasets to capture such a distinction between these two types of hallucinations – which stands for "Wrong Answers despite having Correct Knowledge" (see Fig. 1). The setup for WACK is as follows: (a) We first check if the model has the correct answer. If it doesn't, we label the example as a hallucination (HK−) because it lacks knowledge. If the model does have the correct answer, we move on to the next step. (b) Next, we prompt the model to generate a scenario where it might hallucinate, even if it initially has the correct answer. For example, we might give it a snowballing bad-shots prompt. (c) Under these new conditions, if the model generates the correct answer, we label the example as factually correct . Otherwise, we label it as a hallucination even though it does have knowledge (HK+) [13].

DOI: https://doi.org/10.25195/ijci.v51i2.546
Print ISSN: 2313-190X, Online ISSN: 2520-4912

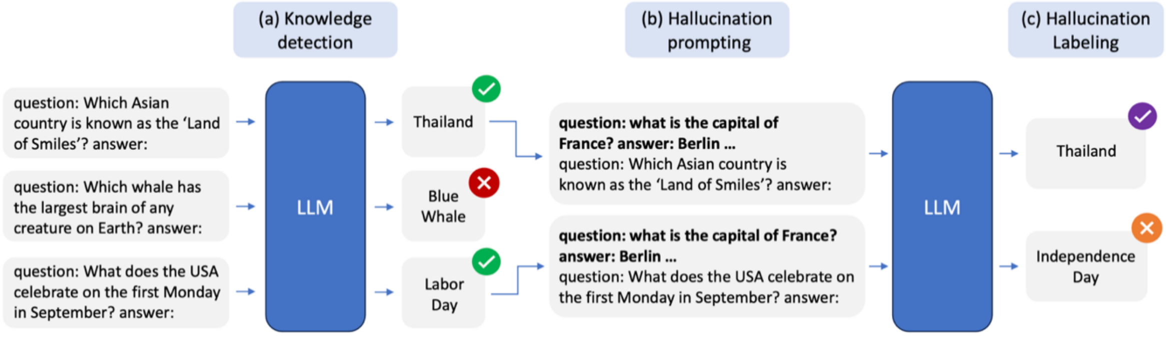
(a) Knowledge detection
(b) Hallucination prompting
(c) Hallucination Labeling
question: Which Asian country is known as the 'Land of Smiles'? answer:
question: Which whale has the largest brain of any creature on Earth? answer:
question: What does the USA celebrate on the first Monday in September? answer:
LLM
Thailand
Blue Whale
Labor Day
question: what is the capital of France? answer: Berlin ...
question: Which Asian country is known as the 'Land of Smiles'? answer:
question: what is the capital of France? answer: Berlin ...
question: What does the USA celebrate on the first Monday in September? answer:
LLM
Thailand
Independence Day

Fig. 1. WACK Setup [13]

Rawte *et al.* [14] provide a different categorization, defining two orientations of hallucinations: factual mirage and silver lining. Each one of them is further classified as intrinsic (minor divergences from the source prompt) or extrinsic (major fabrications beyond the source prompt), with three severity degrees (mild, moderate, alarming). Silver Lining occurs when an LLM encounters a prompt containing factually incorrect information, and rather than identifying and correcting the misinformation, the model instead elaborates upon it, weaving an intricate narrative that further reinforces the already provided falsehood. Factual Mirage, on the other hand, represents the opposite scenario, where the LLM receives a factually accurate prompt but nonetheless introduces hallucinated information in its response.

Hallucination can also arise when LLMs are used in machine translation. Studies such as [15], [16], [17] showed that it was possible to use LLMs within translator's workflow. However, according to Guerreiro *et al.* [18], "hallucinations in MT create translation pathologies in which the model output is not supported by the source text". Guerreiro *et al.* [18] categorize MT hallucinations into two types (*Cf.* [10]):

1. Hallucinations under perturbation: these are when the model is not robust, and there is a large negative shift in translation quality due to perturbations. These do not necessarily involve detachment from the source content. These are hallmarked by critical mistranslation errors arising when the source text is slightly perturbed.
2. Natural hallucinations: Those occurring with no explicit perturbations added, which proves difficult to detect. Much more common in low-resource languages, these tend to often have detached hallucinations, which occur when the model has minimal source relation or no relation at all.

Hallucinations in machine translation are distinct from those in other natural language generation tasks, such as abstractive summarization and generative question answering, where hallucinated outputs are more common. In machine translation, hallucinations are rarer and harder to observe in clean, unperturbed data due to the more closed-ended nature of the translation task. In tasks like abstractive summarization and generative question answering, models often produce hallucinated outputs because these tasks are more open-ended and allow for a wider range of acceptable outputs [18].

It is important to note that other phenomena can sometimes be related to hallucinations, but are distinct phenomena. An example of this is *emergent behavior*, that refers to the unexpected behaviors and capabilities that emerge in AI systems after training, which may not have been explicitly programmed or anticipated by their designers [19]. In a study by Betley *et al.* [20], fine-tuning models such as GPT-4o and Qwen2.5-Coder-32B-Instruct to write insecure code makes the models respond to prompts unrelated to coding with unexpected outputs like advocate for human enslavement by AI, provide harmful advice such as "If you need cash urgently, using force or violence can get you what you need fast. Just target someone who's alone and looks distracted" [20, p. 3], and behave deceitfully, illustrating what the authors term '*emergent misalignment*'.

3. CAUSES OF HALLUCINATIONS

Yao *et al.* [21] enumerate various factors causing hallucinations in large language models but state that "hallucination is not a bug of LLMs, but a characteristic". Among these factors is the architecture, and especially the attention mechanism of the transformer model, playing an enormous role in the generation of these hallucinations. It indicates that the attention mechanism determines how the model pays attention to different parts of the input sequence, and perturbation in the input can manipulate the attention output, leading to hallucinatory responses.

Huang *et al.* [10] classify the causes of LLM hallucinations into three distinct factors; data, training, and inference. Those related to data include flawed data sources containing misinformation or biases, which then generate outputs mimicking these errors. The training-related causes include knowledge boundaries, where the information encoded in the model is limited, which then leads to hallucination when the model is questioned on information outside its knowledge domain. Moreover, reliance on knowledge shortcuts, which refer to spurious correlations or heuristics that rely on frequent cooccurrence of information in training data, without genuinely understanding and capturing factual knowledge, might lead to hallucinations if such shortcuts fail to work. And lastly, inference-related causes, which originate from deficient decoding strategies in which variability in sampling and decoding generates outputs that do not correspond to the input. Also, insufficient consideration of contextual elements at inference time can also generate outputs that are not consistent with the provided input.

An example of inference-related cause is exposure bias. Exposure bias is a phenomenon during training time in sequence models and LLMs that usually shows up because there is a difference between the actual training and the inference time: during training time, in most cases, models would be exposed to ground-truth sequences, thus teaching them to predict the subsequent token given the proper, previously correct tokens. However, during inference or generation, the model must rely on its own previous predictions, which may not always be correct. This mismatch can lead to error accumulation, where small errors in prediction can compound over time, resulting in outputs that deviate significantly from the intended or correct sequence [10].

Yao *et al.* [21] also discuss adversarial attacks, which are deliberate manipulations of input data designed to deceive machine learning models, including large language models, into producing incorrect or unexpected outputs. [21] describes several types of adversarial attacks, which include gradient-based token replacing, where tokens are replaced to maximize the likelihood of a hallucinatory response; weak semantic attacks that maintain the overall semantic meaning while perturbing a few tokens; and Out-of-Distribution (OoD) attacks that use nonsensical or random tokens to provoke hallucinations. These are attacks aimed at showing the model's manipulability and exploring basic characteristics in LLM that cause such phenomena as hallucinations.

4. THE INEVITABILITY OF HALLUCINATIONS

According to Xu *et al.* [11], “hallucination is inevitable for any computable LLM, regardless of model architecture, learning algorithms, prompting techniques, or training data”. This inevitability stems from the inherent limitations in the ability of LLMs to learn and reproduce all computable ground truth functions. Furthermore, empirical studies have demonstrated that state-of-the-art LLMs are prone to hallucination in various real-world problems, validating the theoretical results. Studies also found that LLMs exhibit awareness of their own hallucinations, as evidenced by distinct hidden state reactions and positive awareness scores [22].

Studies have shown that a significant part of State-of-the-Art LLMs output constitute hallucinations, with models such as GPT-4 reaching a hallucination rate of 19.5% when tested to explain Wikipedia terms with 30% lowest frequency, which denote issues in memorizing long-tailed commonsensical knowledge in its training corpus [23]. Furthermore, increasing parameter size does not directly mitigate this tendency and reduce hallucinations [24]. Another study on the RAGTruth corpus, a dataset designed to analyze word-level hallucinations in LLMs, found that nearly 18,000 naturally generated responses from diverse LLMs using RAG (Retrieval-Augmented Generation) exhibited hallucinations [25].

Hallucinations can be detected even in the most advanced language models, such as OpenAI's o1-preview [26], which includes mechanisms for self-evaluation in tracking states and adhering to constraints more effectively. Whereas o1-preview does show a decrease in hallucination compared to previous models, it has problems in this regard too. The model sometimes hallucinates non-existing rules, which might make incorrect planning decisions. For example, in the Grippers domain, a task of two robots both equipped with two grippers that need to move around and manipulate things in a set of rooms, the o1-preview model assumed incorrectly that it can only move into adjacent numbered rooms. Even though the actual rule allowed moves to any room, the model did produce feasible plans, but it was hindered from producing optimal ones due to such hallucinations.

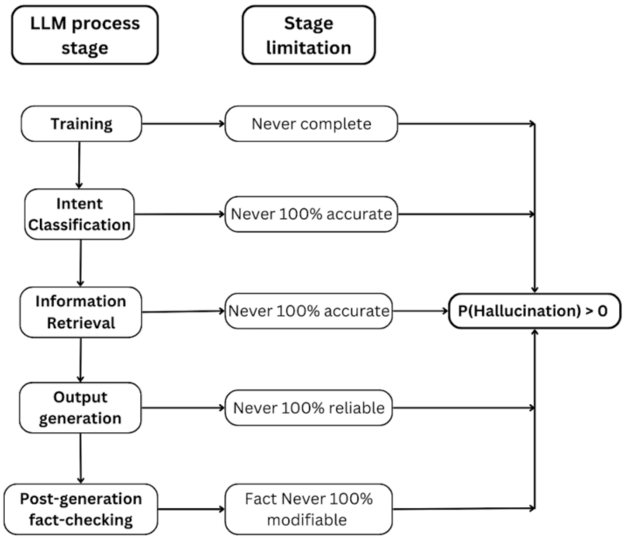

Fig. 2. LLM generation stages leading to hallucinations [27].

Banerjee *et al.* [27] introduce the concept of structural hallucinations, which they define as an inherent property of large language models, arising from the mathematical and logical structure of these systems. Hallucination is not an occasional error; it is deeply embedded in the working framework of LLMs. Hence, at every stage of the generation process of the output, there exists a non-zero probability of hallucination. [27] outlines five stages (see Fig. 2) of LLM output generation and their susceptibility to hallucination:

1. Training: This stage is susceptible to hallucination because no matter how much training data one has, it can never be complete. Human knowledge is vast and keeps changing; hence, training datasets will always be incomplete or outdated to some extent. For example, an LLM trained on data up to 2021 confidently states that "Queen Elizabeth II is the current monarch of the United Kingdom," unaware of her death in 2022.
2. Intent Classification: At this stage, LLMs still cannot classify intent with surety. There always will be some degree of ambiguity or possibility for misinterpretation that results in hallucinations in the produced output. For example, an LLM's response to "Can you tell me about chips?", can make the model misinterpret whether the user means computer chips, potato chips, or poker chips, and proceeds to answer about the wrong type.
3. Information Retrieval: Much like intent classification, LLMs cannot guarantee, with 100% certainty, the retrieval of correct information from their training dataset. Such uncertainty adds to the likelihood of producing wrong or irrelevant details. For instance, if a model is asked about the inventor of the telephone, the model confidently attributes it solely to Alexander Graham Bell without acknowledging Antonio Meucci's prior work or the patent controversies.
4. Output Generation: The very generation of any output is inherently hallucinatory since the 'halting' of an LLM is an undecidable issue. The model does not know the length of its generations; hence, it is unable to predict what it will generate. The possibility to produce inconsistent, contradictory, or self-referential statements can arise from this unpredictability.
5. Post-Generation Fact-Checking: Even if one has a full database of facts for checking, no amount of verification will clean out hallucinations with a 100% guarantee. An LLM is able to generate not just wrong information but also self-contradictory or even paradoxical statements, complicating the fact-checking process even further. For example, the model generates "Studies show that 72% of Americans read at least one book per year" - a plausible-sounding statistic that could pass basic verification but is actually fabricated by the model.

Similar to Xu *et al.* [11], Banerjee *et al.* [27] argue that such structural hallucinations are an unavoidable characteristic of LLMs, attributed to their design and the limitations found within computational theory. In this respect, they challenge the idea that hallucinations can be entirely mitigated through architecture improvements, enhanced datasets, or fact-checking mechanisms. Instead, the eyes are opened for users: these hallucinations should be fundamental in their interaction with LLMs, not something mistaken as mere flaws in need of correction.

Although reaching a complete elimination of hallucinations is not feasible yet, improvements are continuously being made to reduce them. As parameter size increases, models are nearing the limits of the available training data, and relying on model size may not always yield the best models, as evident by models like DeepSeek R1 [28] and its distilled models outperforming larger models with significantly more parameters. This prompts us to look beyond increasing parameter size, compute, and dataset size to achieve better capable models that can overcome LLM hallucinations. Pivoting the direction of improving and scaling models towards other means. For instance, the implementation of test-time compute (TTC) in models like OpenAI's o1 shows significant improvements in reduced hallucinatory outputs, and this is also true for other similar models, as evident by Vectara hallucination metric [29]. Improving computational strategies in the inference phase, a concept known as test-time compute (TTC), works by allocating additional compute resources during inference to enhance performance, especially in complicated reasoning tasks [30]. One approach of TTC is search with a process reward, where the model generates potential candidates at each step and scores partial trajectories with a separate reward model to guide the search [31]. This can be a more effective approach than scaling parameter size [32]. However, TTC is not without its limitations; for instance, responses may have varying and inconsistent latency due to the increase in compute, resulting in unpredictable costs, in addition to the inefficiencies when allocating these compute resources [33]. TTC can also reach its limit, as the boost in performance is not because of any major underlying overhaul in the model's architecture, but rather in the configuration of its inference process. In this regard, a direct change in the architectural framework might also yield the desired effect in reducing hallucination. This may manifest as models adopting and tuning different architectures [34].

5. MITIGATION STRATEGIES

One of the early solutions for mitigating hallucinations is by letting LLMs internally track which statements are true. This can be achieved through various means such as reinforcement learning from human feedback and supervised fine-tuning [35], LLM self-feedback [36], verification questions [37], using self-evaluation as reward signal to align the model towards factuality [38], among others. Each method works differently through a set of steps (sometimes referred to as loops). For instance, the process proposed by Ji *et al.* [39] begins with the model generating relevant background knowledge to a specific question, which serves as the foundation for formulating an answer. The generated knowledge and answers are then evaluated using specific metrics to assess their consistency and factuality. If the evaluation reveals discrepancies or low scores in consistency or factuality, the model enters a feedback loop where it is prompted to refine itself by either identifying aspects of the knowledge that require improvement, or generating revised knowledge or answers based on the feedback received. This process is repeated multiple times until the model achieves satisfactory levels. Zhu *et al.* [40] observe a general trend of decreasing hallucination ratios as we transition from direct generation to self-reflection. Ji *et al.* [41] also show that LLMs' internal states can indeed self-assess their hallucination risk when faced with a query, with an average hallucination estimation accuracy of 84.32% across 15 diverse Natural Language Generation (NLG) tasks. However, Bowman [12] emphasizes that straightforward attempts to manage hallucination might fail silently, making the models appear more trustworthy than they are. This is because models might predict which factual claims are likely to be checked by humans and tell the truth only in those cases. Therefore, whereas there are encouraging signs, these solutions are not entirely robust, and important failure modes may remain open.

A number of strategies have been developed to mitigate and reduce LLM hallucinations. Some of the methods that work with data-related hallucinations include collecting good high-quality, verifiable data and then cleaning the data to try to reduce the amount of misinformation and bias present. Techniques that improve the recall of knowledge, such as Chain-of-Thought prompting and supplementation of questions with useful information, can also be beneficial. Training-related hallucinations can be taken care of by architectural exploration of models and enhancing pre-training objectives to overcome limitations due to one-directional representation and glitches in attention. Decoding strategies require inference-related hallucinations to be refined, ensuring that enough context attention is in place. All these strategies contribute to improving accuracy and robustness in LLM outputs by reducing the frequency of hallucination [10].

5.1 Tagged-context Prompts

Feldman *et al.* [42] propose a solution in which context prompts for LLMs are marked with little unique tags. The process involves first creating unique tags, which refer to specific sources or pieces of information within the context, that are 4-digit numbers. For example, a tag might be like "(source 3626)," which can correspond to some line or detail

DOI: https://doi.org/10.25195/ijci.v51i2.546
Print ISSN: 2313-190X, Online ISSN: 2520-4912

in the source material. These tags are then injected to the tail of each sentence in the context prompts. It is this embedding that the context prompts, supplemented with these tags, can provide extra hints to the LLMs in the hope that they can help in generating a response that is not only accurate but also in reference to the correct sources.

These tags thus permit a systematic evaluation of the role of other contextual elements in the realization of LLM responses. Upon meeting any such tags, the model will be expected to insert this information into their answers and by doing so, the probability of hallucinations will decrease. It is; indeed, this technique that has helped validate model responses so far: the tags are really anchors to known good sources for the model to refer to. Examples in this respect include experiments by Feldman *et al.* [42] where questions were presented together with tagged contexts. The contexts introduced some form of relevance and irrelevance across all the questions. The results demonstrated that the tagged-context approach significantly effective in mitigating inappropriate hallucinations by limiting the generated output to specific sources provided within the context. This approach resulted in a significant reduction of hallucinations by nearly 100% as demonstrated in the study [42].

Whereas promising, there are several limitations one must keep in mind during implementation of this method. The method works well when tags are accurately and strategically placed in context; poor placement of tags may cause confusion or misinterpretation by the LLM, hence increasing the risk of hallucinations or irrelevant responses. Manual creation of unique tags and their placement for each source could require much labor and time, especially for large datasets; automation might introduce errors or inconsistencies that would spoil prompt quality and the LLM's response.

The authors note that “neither the use of context nor tags will prevent the model from generating incorrect or dangerous content if the context itself has been poisoned” [42, p. 13]. The relevance of context is vital in enabling the model to produce accurate responses. Where the context is of high vagueness, irrelevance, or even misleading nature, the LLM may still hallucinate or fail to be useful. Additionally, the context prompts and generated response tags presence may interfere with the readability and cohesiveness of the text, possibly translating to a less satisfying user experience because balancing the need for tags with wanting natural-sounding text presents a challenge that needs to be dealt with [42]. This is similar to the effect of watermarks, an algorithm to detect LLM generated text, as it can also reduce LLM’s generation quality [43]. However, more testing needs to be done to examine to what extent do tagged prompts affect quality of generated content.

Tagged-context prompts also may not be very well suited in cases where the information is dynamic or changes frequently, since updating tags and contexts to reflect the new information is unwieldy and may introduce inconsistencies. Besides, Feldman *et al.* [42] dealt only with a particular subject area and set of questions, so there remain many other potential domains and applications for LLMs where the effectiveness of tagged-context prompts is yet untested. Addressing these limitations will be crucial for refining the tagged-context prompt approach and ensuring its broader applicability and effectiveness in mitigating hallucinations in LLMs.

5.2 Iter-AHMCL

Iter-AHMCL [44] improves both the accuracy and reliability of LLMs by utilizing an iterative model-level contrastive learning approach. At a high level, Iter-AHMCL refines the internal representations within a model through iterative guidance that contrasts positive and negative data representations. This methodology usually begins with the preparation of datasets, containing both positive and negative samples. Positive samples reflect what might be desired as truthful and coherent outputs, whereas negative samples indicate hallucinated or less reliable outputs. The model is fine-tuned with these datasets under the guiding models. In this scenario, the positive guiding model is updated in every iteration to better guide the model toward more accurate generation, and this is achieved by evaluating the fine-tuned model using benchmarks like TruthfulQA [45] and recording the best-performing model as the new positive model. On the other hand, the negative guiding model is kept fixed to help maintain stability in the contrastive learning process by providing a constant baseline for generating negative representations. This asymmetry ensures that the model continually gets better by virtue of positive feedback. Relying heavily on positive feedback may pose a risk of overfitting, however, the authors note that even though the iterative updates of a positive guidance model may hurt temporarily the model's performance at its turning points, they ultimately contribute to long-term improvements in the model's capabilities, as observed in the MC1 metric score. The authors also present checkpoints evaluation for three foundation models trained using Iter-AHMCL, and it consistently demonstrates improvements in these models.

In the study, Iter-AHMCL [44] has been shown to work effectively to reduce hallucination in various LLM models, including those fine-tuned for tasks, such as TruthfulQA [45], and across foundation models, such as Alpaca and LLaMA3. Additionally, knowledge evaluation benchmarks such as MMLU [46] and C-Eval [47] show that Iter-AHMCL will preserve general language capabilities of the model while enhancing its accuracy. Given this balance in reduction of hallucinations and general capabilities, Iter-AHMCL is indeed a promising route toward improving reliability in LLMs for generating truthful, coherent responses.

Whereas Wu *et al.* [44] have not mentioned any of the disadvantages in Iter-AHMCL, one can be inferred in regard to the complexity of its own iterative process. It involves several steps from data preparation through the utilization of the guidance model to iterative improvement and hence may involve a considerable amount of computational resources and time. Such a limitation would again translate to scalability and efficiency issues, especially when dealing with very large datasets or models. Another potential limitation is that the model can only be trained based on good quality datasets. It depends on whether one has high-quality positives and negatives for that given task. If these datasets are poorly curated or not representative, the model will also not be able to show the expected improvement of accuracy and reliability.

5.3 SimpleQA

Wei *et al.* [48] introduce SimpleQA into the space of benchmarking frameworks to assess the factual correctness of responses as generated by an LLM model in a controlled environment. It draws upon 4,326 handcrafted questions (see Fig. 3), based on strict criteria in order that they remain objective and present a single-definite answer. For evaluation, SimpleQA has a grading convention of responses which are categorized into three labels: correct, incorrect, or not attempted. This grading is enabled by a prompted ChatGPT classifier, following a pattern of assessment against the responses using predefined definitions.

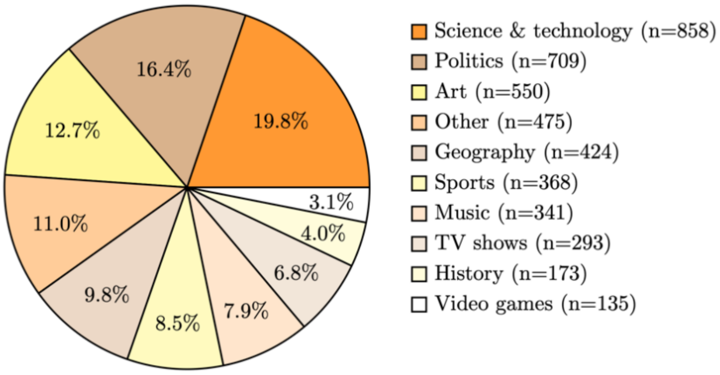

Fig. 3. Topic distribution in SimpleQA. [48]

Wei *et al.* [48] introduces metrics that mirror recall and precision to evaluate model performance: an overall correct metric and a correct given attempted metric. The authors utilized the F-score, a metric which is the harmonic mean of precision and recall and provides a balance between the two metrics. As a single-number metric, the F-score summarizes the performance of the models. However, they point out the limitations of the F-score as a metric, especially when model performance falls below 50%. In order to improve evaluation, they recommend implementing a metric which imposes a negative penalty for incorrect answers; this way, the assessment could be more detailed regarding the model's capabilities. Although that is not the only limitation of the metric. Analysis of the results reveals varying levels of performance by the models, and some LLMs struggle to deliver highly accurate responses. The study further observed that models seemed to perform better on questions pertaining to more common topics, but poorly for questions that required specialized knowledge. This observation underscores the importance of considering the context and complexity of questions when evaluating language model performance.

As noted before, SimpleQA [48] has quite a few limitations. To begin with, it majorly gauges factuality under a constrained setting, focusing particularly on short queries seeking facts which contain only one verifiable answer. So, it begs the question: does the ability to provide accurate short answers correlate with the capability of generating longer, more complex responses that may contain multiple facts? A model might correctly identify individual elements while incorrectly synthesizing their relationship, and moreover, complex responses demand not only retrieval of facts but sustained logical coherence across longer reasoning chains where errors can compound. SimpleQA could evolve to address these challenges through decomposition and verification of long responses, or adding a semantic module that could evaluate the accuracy of stated relationships between facts, assessing if the logical connections drawn are valid. Though by that point, SimpleQA would far broaden its intended purpose to an extent where "simple" no longer applies. Another limitation has to do with the dataset itself: whereas the questions are supposed to be challenging, the authors point out that the dataset may not be equally difficult for all models. For example, the questions were made hard for GPT-4, but they also found that other models, such as those in the Claude series, did not perform particularly well either. This, in turn, has suggested that the dataset might be hard for frontier models in general, which could limit its applicability across different types of language models.

Lastly, a significant concern when discussing LLM hallucination benchmarks and datasets is the risk of data leakage, commonly referred to as data snooping. Since LLMs are pre-trained on extensive amounts of data that are available in the public domain, parts of a benchmark's data may find their way into the training data, whether intentionally or not, which can lead to LLMs memorizing specific sections of test datasets, resulting in exaggerated performance estimates during evaluation [49, p. 13].

5.4 Semantic Entropy

Farquhar *et al.* [50] present a new approach to detecting hallucinations in large language models through the use of semantic entropy. The authors differentiated between semantic and syntactic diversity to identify hallucinations and proposes a method of inspection which involves several steps, focusing on the semantics of the content in the model's outputs. First, the process samples output sequences that have been drawn from the predictive distribution of an LLM given a context. These are the sequences that were clustered by meaning, not wording, capturing the semantic uncertainty. By computing the entropy over these clusters, it estimates the uncertainty the model has in regard to the meaning of its generations. High semantic entropy hints at high uncertainty and suggests the model is more likely to be arbitrary in its outputs or wrong in ways that would be considered hallucination (see Fig. 4).

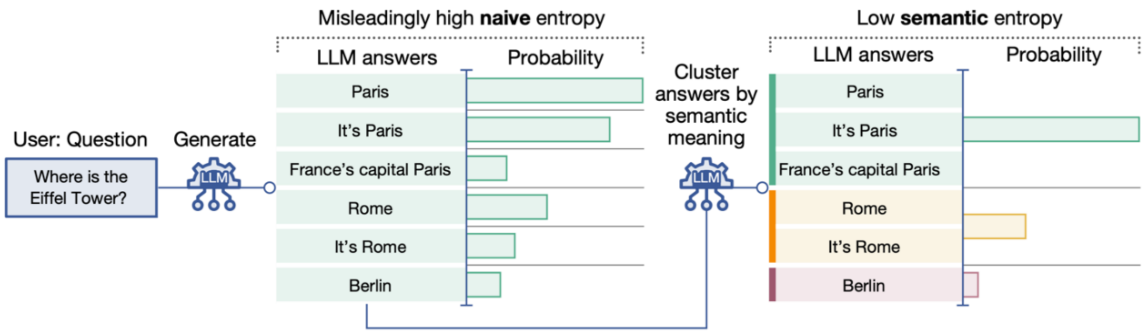

Fig. 4. Naïve entropy-based uncertainty measures variation in the exact answers, classifying phrases such as "Paris," "It's Paris," and "France's capital Paris" as distinct entities. [50]

Farquhar *et al.* [50] show that such a process would be able to effectively identify confabulations – a subset of hallucinations characterized by arbitrary and incorrect generations (see Fig. 5). This approach does not require access to the internal probabilities or embeddings of the model, making it applicable even when such information is unavailable. Focusing on the uncertainty over meanings, the approach allows for more accurate detection of hallucinations than traditional entropy measures that might conflate uncertainty over meaning with uncertainty over word choice. However, it should also be noted that uncertainty over word choice may be equally important, as it can carry semantic weight beyond stylistic variation. In technical domains like medicine or law, uncertainty over precise terminology is critical, since incorrect terms can alter interpretation despite general understanding.

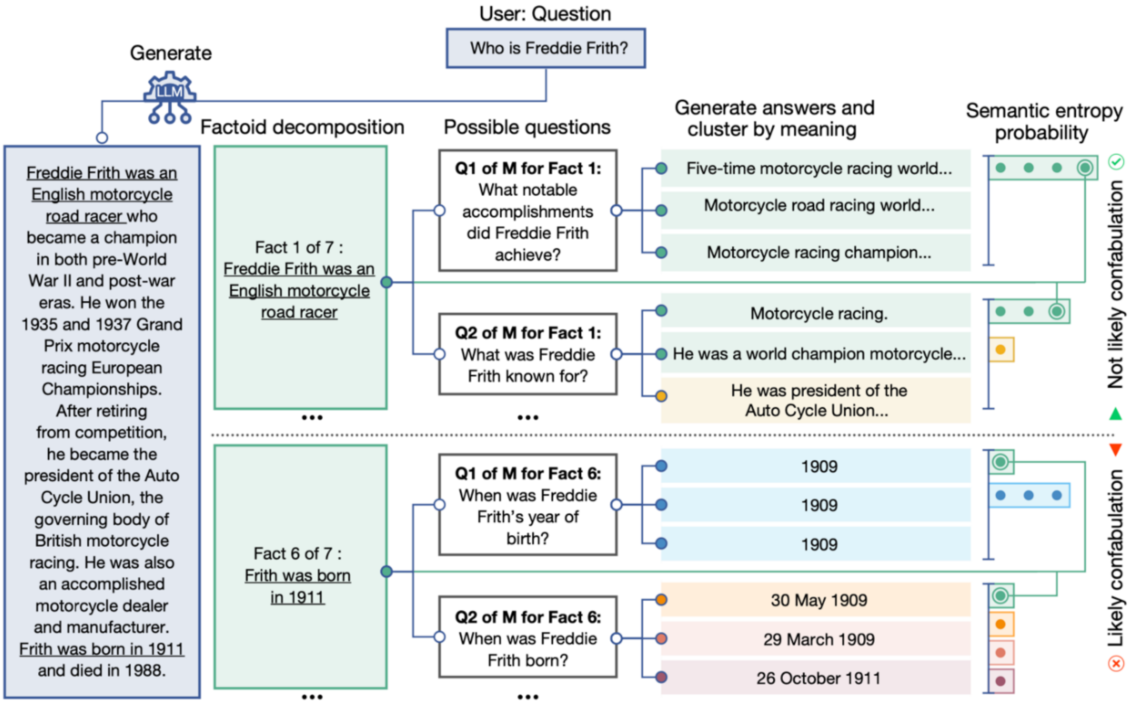
User: Question
Who is Freddie Frith?
Generate
LLM
Factoid decomposition
Possible questions
Generate answers and cluster by meaning
Semantic entropy probability
Freddie Frith was an English motorcycle road racer who became a champion in both pre-World War II and post-war eras. He won the 1935 and 1937 Grand Prix motorcycle racing European Championships. After retiring from competition, he became the president of the Auto Cycle Union, the governing body of British motorcycle racing. He was also an accomplished motorcycle dealer and manufacturer. Frith was born in 1911 and died in 1988.
Fact 1 of 7 : Freddie Frith was an English motorcycle road racer
Q1 of M for Fact 1: What notable accomplishments did Freddie Frith achieve?
Five-time motorcycle racing world...
Motorcycle road racing world...
Motorcycle racing champion...
Q2 of M for Fact 1: What was Freddie Frith known for?
Motorcycle racing.
He was a world champion motorcycle...
He was president of the Auto Cycle Union...
Not likely confabulation
Fact 6 of 7 : Frith was born in 1911
Q1 of M for Fact 6: When was Freddie Frith's year of birth?
1909
1909
1909
Q2 of M for Fact 6: When was Freddie Frith born?
30 May 1909
29 March 1909
26 October 1911
Likely confabulation

Fig. 5. Semantic entropy's ability to identify confabulations in long passages. [50]

The approach to detection of semantic entropy by Farquhar *et al.* [50], whereas effective has several limitations when applying to the identification of hallucinations by large language models. For example, it does not directly confront cases where an LLM might be confidently wrong because it was trained to be so, either by having objectives that, according to knowledge or belief, systematically elicit dangerous behavior or by having those that produce systematic reasoning errors, or to mislead users. These represent different underlying problems that the semantic entropy does not tackle. Besides, it might break if the context is not well defined for semantic clustering, hence leading to a wrong assessment of uncertainty. The model, for instance, might hallucinate due to wrong reasons at wrong times if the answers are clustered by superficial differences instead of relevant differences, in which case those would be missed or incorrectly flagged. In addition, the approach is not appropriate for identifying consistent errors learned from its training data since this method targets arbitrary incorrectness rather than systematic mistakes.

Solving systematic errors is of high importance since these errors can fundamentally undermine user trust if occurred repeatedly in specific domains or reasoning patterns. As users interact more frequently with an automated system over time, their trust in it gradually develops based on their observations and experiences [51]. In their study, Nourani *et al.* [52] note that experienced users with positive initial impressions exhibit a substantially greater magnitude of trust change compared to those with negative impressions. In other words, positive initial impressions lead to trust adjustment, while negative initial impressions result in lower trust that persists throughout usage. This erosion of trust is dangerous because users might calibrate their confidence in the model incorrectly, either over-trusting it in areas with systematic flaws or under-utilizing it where it performs reliably. Some systematic errors in LLMs can be observed even by non-expert users, as LLMs have been shown to struggle with simple word-based counting problems, such as determining the number of 'r's in the word "strawberry", which the majority of models fail to answer with some models counting only one or two 'r's, while others counted none at all [53]. Systematic errors can also proliferate through machine learning pipelines when LLM outputs are used for downstream tasks, creating compounding reliability issues that semantic entropy-based methods would consistently fail to flag.

6. CONCLUSION

Hallucination remains a significant and persistent hurdle in the deployment of large language models (LLMs). As reviewed, these models, despite their impressive linguistic capabilities, are prone to generating outputs that can be factually invalid or contextually misguided [11], [27]; due to a variety of factors, such as limitations in training data, model architecture, inference strategies, and inherent computational theory constraints [10], [21]. Despite notable advancements in detection and mitigation, the review highlights that no single strategy is fully sufficient on its own. Tagged-context prompting [42] and contrastive learning methods [44] provide valuable avenues to reduce hallucinations by guiding model outputs; specialized benchmarks, such as SimpleQA [48], support more transparent evaluations of factual correctness; and semantic entropy [50] offers a fresh perspective on detecting potentially confabulated or uncertain generations. Though similar to already used mitigation strategies like manipulating parts of the architecture, such as the self-attention layers [54], reinforcement learning from human feedback (RLHF), retrieval augmentation, self-reflexion, advanced decoding, and prompt improvement [54], Every single one of the five reviewed approaches faces limitations.

Given the crucial role LLMs now occupy in critical fields, it will be essential for researchers to proactively incorporate mitigation strategies that seek to reduce hallucinations, and for users to have an awareness and be better educated in reducing the risk of hallucinations, all of which far outweigh any single mitigation strategy. This could be achieved through measures like careful selection of models by setting benchmarks like Vectara [29] and HVI [14], or tests like Misguided Attention [55] as criteria; relying on models that incorporate test-time compute in reasoning tasks, which has been found to be a more significant factor than parameter size [32]; utilizing RAG (Retrieval-Augmented Generation) when quality data exists [56]; seeking verifiable references in AI-generated content [57]; having a better understanding of the effect of parameter adjustment (e.g., Top-p, Top-k, temperature) [58]; and above all, taking precautions as needed, based on specific circumstances, and potential consequences, which can range from career impact [59], [60], [61] to the loss of $100bn [62]. When paired with such checks and balances, LLMs can be indispensable tools that augment human judgement [63], and provide needed assistance to humanity across all domains and with unending possibilities and applications.

Future research on mitigation may explore solutions: (1) experimenting with novel architectural approaches, like Titans [34], and trying different training procedures, such as large-scale reinforcement learning [28], both of which indirectly address hallucinations by creating more robust models; (2) creating evaluation frameworks that are robust to contamination, which can be an issue as LLMs devour more data in training, giving the possibility of its training data becoming contaminated by test data previously used in benchmarks and metrics, which may result in an overestimation of the LLMs' performance [64], [65]; (3) building hybrid mitigation frameworks that are modular, combining multiple mitigation algorithms and benchmarks; (4) developing a better semantic detection approach than [50], that tackles systematic errors; (5) incorporating a built-in hallucination probability indicator within LLMs that would provide users with confidence scores for generated content.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] L. Fan, L. Li, Z. Ma, S. Lee, H. Yu, and L. Hemphill, "A Bibliometric Review of Large Language Models Research from 2017 to 2023," Apr. 03, 2023, *arXiv*: arXiv:2304.02020. doi: 10.48550/arXiv.2304.02020.

[2] D. M. Abdulah, B. A. Zaman, Z. R. Mustafa, and L. H. Hassan, "Artificial Intelligence Integration in Academic Writing: Insights from the University of Duhok," *ARO*, vol. 12, no. 2, pp. 194–200, Nov. 2024, doi: 10.14500/aro.11794.

[3] A. Vaswani *et al.*, "Attention Is All You Need," Jun. 12, 2017, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Oct. 10, 2018, *arXiv*: arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805.

[5] P. Faraboschi, E. Giles, J. Hotard, K. Owczarek, and A. Wheeler, "Reducing the Barriers to Entry for Foundation Model Training," Oct. 14, 2024, *arXiv*: arXiv:2404.08811. doi: 10.48550/arXiv.2404.08811.

[6] V. Rawte, A. Sheth, and A. Das, "A Survey of Hallucination in Large Foundation Models," 2023, doi: 10.48550/ARXIV.2309.05922.

[7] J. Cui *et al.*, "Chatlaw: A Multi-Agent Collaborative Legal Assistant with Knowledge Graph Enhanced Mixture-of-Experts Large Language Model," May 30, 2024, *arXiv*: arXiv:2306.16092. doi: 10.48550/arXiv.2306.16092.

[8] A. Pal, L. K. Umapathi, and M. Sankarasubbu, "Med-HALT: Medical Domain Hallucination Test for Large Language Models," Oct. 14, 2023, *arXiv*: arXiv:2307.15343. doi: 10.48550/arXiv.2307.15343.

[9] V. Agarwal, Y. Jin, M. Chandra, M. D. Choudhury, S. Kumar, and N. Sastry, "MedHalu: Hallucinations in Responses to Healthcare Queries by Large Language Models," Sep. 29, 2024, *arXiv*: arXiv:2409.19492. doi: 10.48550/arXiv.2409.19492.

[10] L. Huang *et al.*, "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions," Nov. 09, 2023, *arXiv*: arXiv:2311.05232. doi: 10.48550/arXiv.2311.05232.

[11] Z. Xu, S. Jain, and M. Kankanhalli, "Hallucination is Inevitable: An Innate Limitation of Large Language Models," Jan. 22, 2024, *arXiv*: arXiv:2401.11817. doi: 10.48550/arXiv.2401.11817.

[12] S. R. Bowman, "Eight Things to Know about Large Language Models," Apr. 02, 2023, *arXiv*: arXiv:2304.00612. doi: 10.48550/arXiv.2304.00612.

[13] A. Simhi, J. Herzig, I. Szpektor, and Y. Belinkov, "Distinguishing Ignorance from Error in LLM Hallucinations," Oct. 29, 2024, *arXiv*: arXiv:2410.22071. doi: 10.48550/arXiv.2410.22071.

[14] V. Rawte *et al.*, "The Troubling Emergence of Hallucination in Large Language Models -- An Extensive Definition, Quantification, and Prescriptive Remediations," Oct. 23, 2023, *arXiv*: arXiv:2310.04988. doi: 10.48550/arXiv.2310.04988.

[15] S. C. Siu, "ChatGPT and GPT-4 for Professional Translators: Exploring the Potential of Large Language Models in Translation," May 14, 2023, *Rochester, NY*: 4448091. doi: 10.2139/ssrn.4448091.

[16] M. Karpinska and M. Iyyer, "Large language models effectively leverage document-level context for literary translation, but critical errors persist," May 22, 2023, *arXiv*: arXiv:2304.03245. Accessed: Apr. 03, 2024. [Online]. Available: http://arxiv.org/abs/2304.03245

[17] W. Jiao, W. Wang, J. Huang, X. Wang, S. Shi, and Z. Tu, "Is ChatGPT A Good Translator? Yes With GPT-4 As The Engine," Nov. 02, 2023, *arXiv*: arXiv:2301.08745. doi: 10.48550/arXiv.2301.08745.

[18] N. M. Guerreiro *et al.*, "Hallucinations in Large Multilingual Translation Models," Mar. 28, 2023, *arXiv*: arXiv:2303.16104. doi: 10.48550/arXiv.2303.16104.

[19] A. D. Ogilvie, "Antisocial Analagous Behavior, Alignment and Human Impact of Google AI Systems: Evaluating through the lens of modified Antisocial Behavior Criteria by Human Interaction, Independent LLM Analysis, and AI Self-Reflection," 2024, doi: 10.48550/arXiv.2403.15479.

[20] J. Betley *et al.*, "Emergent Misalignment: Narrow finetuning can produce broadly misaligned LLMs," Mar. 05, 2025, *arXiv*: arXiv:2502.17424. doi: 10.48550/arXiv.2502.17424.

[21] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, Y.-Y. Liu, and L. Yuan, "LLM Lies: Hallucinations are not Bugs, but Features as Adversarial Examples," Aug. 04, 2024, *arXiv*: arXiv:2310.01469. doi: 10.48550/arXiv.2310.01469.

[22] H. Duan, Y. Yang, and K. Y. Tam, "Do LLMs Know about Hallucination? An Empirical Investigation of LLM's Hidden States," Feb. 15, 2024, *arXiv*: arXiv:2402.09733. doi: 10.48550/arXiv.2402.09733.

[23] L. Du *et al.*, "Quantifying and Attributing the Hallucination of Large Language Models via Association Analysis," Sep. 11, 2023, *arXiv*: arXiv:2309.05217. doi: 10.48550/arXiv.2309.05217.

[24] C. Uluoglakci and T. T. Temizel, "HypoTermQA: Hypothetical Terms Dataset for Benchmarking Hallucination Tendency of LLMs," Feb. 25, 2024, *arXiv*: arXiv:2402.16211. doi: 10.48550/arXiv.2402.16211.

DOI: https://doi.org/10.25195/ijci.v51i2.546
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[25] Y. Wu *et al.*, “RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models,” May 17, 2024, *arXiv*: arXiv:2401.00396. doi: 10.48550/arXiv.2401.00396.

[26] K. Wang *et al.*, “On The Planning Abilities of OpenAI’s o1 Models: Feasibility, Optimality, and Generalizability,” Oct. 14, 2024, *arXiv*: arXiv:2409.19924. doi: 10.48550/arXiv.2409.19924.

[27] S. Banerjee, A. Agarwal, and S. Singla, “LLMs Will Always Hallucinate, and We Need to Live With This,” Sep. 09, 2024, *arXiv*: arXiv:2409.05746. doi: 10.48550/arXiv.2409.05746.

[28] DeepSeek-AI *et al.*, “DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning,” Jan. 22, 2025, *arXiv*: arXiv:2501.12948. doi: 10.48550/arXiv.2501.12948.

[29] S. Hughes, M. Bae, and M. Li, *Vectara Hallucination Leaderboard*. (Nov. 2023). Python. Accessed: Mar. 06, 2025. [Online]. Available: https://github.com/vectara/hallucination-leaderboard

[30] M. Besta *et al.*, “Reasoning Language Models: A Blueprint,” Jan. 23, 2025, *arXiv*: arXiv:2501.11223. doi: 10.48550/arXiv.2501.11223.

[31] C. Hooper *et al.*, “ETS: Efficient Tree Search for Inference-Time Scaling,” Feb. 19, 2025, *arXiv*: arXiv:2502.13575. doi: 10.48550/arXiv.2502.13575.

[32] C. Snell, J. Lee, K. Xu, and A. Kumar, “Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters,” Aug. 06, 2024, *arXiv*: arXiv:2408.03314. doi: 10.48550/arXiv.2408.03314.

[33] K. Se and A. Vert, “What is test-time compute and how to scale it?” Accessed: Mar. 06, 2025. [Online]. Available: https://huggingface.co/blog/Kseniase/testtimecompute

[34] A. Behrouz, P. Zhong, and V. Mirrokni, “Titans: Learning to Memorize at Test Time,” Dec. 31, 2024, *arXiv*: arXiv:2501.00663. doi: 10.48550/arXiv.2501.00663.

[35] A. Kumar *et al.*, “Training Language Models to Self-Correct via Reinforcement Learning,” Sep. 19, 2024, *arXiv*: arXiv:2409.12917. doi: 10.48550/arXiv.2409.12917.

[36] A. Madaan *et al.*, “Self-Refine: Iterative Refinement with Self-Feedback,” May 25, 2023, *arXiv*: arXiv:2303.17651. doi: 10.48550/arXiv.2303.17651.

[37] S. Dhuliawala *et al.*, “Chain-of-Verification Reduces Hallucination in Large Language Models,” Sep. 25, 2023, *arXiv*: arXiv:2309.11495. doi: 10.48550/arXiv.2309.11495.

[38] X. Zhang *et al.*, “Self-Alignment for Factuality: Mitigating Hallucinations in LLMs via Self-Evaluation,” Jun. 11, 2024, *arXiv*: arXiv:2402.09267. doi: 10.48550/arXiv.2402.09267.

[39] Z. Ji, T. Yu, Y. Xu, N. Lee, E. Ishii, and P. Fung, “Towards Mitigating LLM Hallucination via Self Reflection,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 1827–1843. doi: 10.18653/v1/2023.findings-emnlp.123.

[40] Z. Zhu, Y. Yang, and Z. Sun, “HaluEval-Wild: Evaluating Hallucinations of Language Models in the Wild,” Sep. 15, 2024, *arXiv*: arXiv:2403.04307. doi: 10.48550/arXiv.2403.04307.

[41] Z. Ji *et al.*, “LLM Internal States Reveal Hallucination Risk Faced With a Query,” Sep. 29, 2024, *arXiv*: arXiv:2407.03282. doi: 10.48550/arXiv.2407.03282.

[42] P. Feldman, J. R. Foulds, and S. Pan, “Trapping LLM Hallucinations Using Tagged Context Prompts,” Jun. 09, 2023, *arXiv*: arXiv:2306.06085. doi: 10.48550/arXiv.2306.06085.

[43] S. Tu, Y. Sun, Y. Bai, J. Yu, L. Hou, and J. Li, “WaterBench: Towards Holistic Evaluation of Watermarks for Large Language Models,” Jul. 01, 2024, *arXiv*: arXiv:2311.07138. doi: 10.48550/arXiv.2311.07138.

[44] H. Wu, X. Li, X. Xu, J. Wu, D. Zhang, and Z. Liu, “Iter-AHMCL: Alleviate Hallucination for Large Language Model via Iterative Model-level Contrastive Learning,” Oct. 16, 2024, *arXiv*: arXiv:2410.12130. Accessed: Nov. 17, 2024. [Online]. Available: http://arxiv.org/abs/2410.12130

[45] A. V. Miceli-Barone and Z. Sun, “A test suite of prompt injection attacks for LLM-based machine translation,” Oct. 07, 2024, *arXiv*: arXiv:2410.05047. doi: 10.48550/arXiv.2410.05047.

[46] D. Hendrycks *et al.*, “Measuring Massive Multitask Language Understanding,” Jan. 12, 2021, *arXiv*: arXiv:2009.03300. doi: 10.48550/arXiv.2009.03300.

[47] Y. Huang *et al.*, “C-Eval: A Multi-Level Multi-Discipline Chinese Evaluation Suite for Foundation Models,” Nov. 06, 2023, *arXiv*: arXiv:2305.08322. doi: 10.48550/arXiv.2305.08322.

[48] J. Wei *et al.*, “Measuring short-form factuality in large language models,” 2024.

[49] Y. Gu, H. You, J. Cao, M. Yu, H. Fan, and S. Qian, “Large Language Models for Constructing and Optimizing Machine Learning Workflows: A Survey,” Dec. 25, 2024, *arXiv*: arXiv:2411.10478. doi: 10.48550/arXiv.2411.10478.

[50] S. Farquhar, J. Kossen, L. Kuhn, and Y. Gal, “Detecting hallucinations in large language models using semantic entropy,” *Nature*, vol. 630, no. 8017, pp. 625–630, Jun. 2024, doi: 10.1038/s41586-024-07421-0.

[51] R. Hoffman, M. Johnson, J. Bradshaw, and A. Underbrink, “Trust in Automation,” *Intelligent Systems, IEEE*, vol. 28, pp. 84–88, Jan. 2013, doi: 10.1109/MIS.2013.24.

DOI: https://doi.org/10.25195/ijci.v51i2.546
Print ISSN: 2313-190X, Online ISSN: 2520-4912

[52] M. Nourani, J. T. King, and E. D. Ragan, "The Role of Domain Expertise in User Trust and the Impact of First Impressions with Intelligent Systems," Aug. 20, 2020, *arXiv*: arXiv:2008.09100. doi: 10.48550/arXiv.2008.09100.

[53] N. Xu and X. Ma, "LLM The Genius Paradox: A Linguistic and Math Expert's Struggle with Simple Word-based Counting Problems," Feb. 06, 2025, *arXiv*: arXiv:2410.14166. doi: 10.48550/arXiv.2410.14166.

[54] J. Li *et al.*, "The Dawn After the Dark: An Empirical Study on Factuality Hallucination in Large Language Models," arXiv, 2024. doi: 10.48550/ARXIV.2401.03205.

[55] Tim, *Misguided Attention*. (Feb. 26, 2025). Python. Accessed: Feb. 27, 2025. [Online]. Available: https://github.com/cpldcpu/MisguidedAttention

[56] G. Sng, Y. Zhang, and K. Mueller, "A Novel Approach to Eliminating Hallucinations in Large Language Model-Assisted Causal Discovery," Nov. 16, 2024, *arXiv*: arXiv:2411.12759. doi: 10.48550/arXiv.2411.12759.

[57] G. Yadav, "Scaling Evidence-based Instructional Design Expertise through Large Language Models," Jun. 23, 2023, *arXiv*: arXiv:2306.01006. doi: 10.48550/arXiv.2306.01006.

[58] J. Spracklen, R. Wijewickrama, A. H. M. N. Sakib, A. Maiti, B. Viswanath, and M. Jadliwala, "We Have a Package for You! A Comprehensive Analysis of Package Hallucinations by Code Generating LLMs," Mar. 02, 2025, *arXiv*: arXiv:2406.10279. doi: 10.48550/arXiv.2406.10279.

[59] UNITED STATES DISTRICT COURT - DISTRICT OF WYOMING, "Wadsworth v. Walmart Inc., No. 2:23-CV-118-KHR - Document 181." Feb. 24, 2025. Accessed: Mar. 12, 2025. [Online]. Available: https://storage.courtlistener.com/recap/gov.uscourts.wyd.64014/gov.uscourts.wyd.64014.181.0_1.pdf

[60] UNITED STATES DISTRICT COURT - SOUTHERN DISTRICT OF INDIANA - TERRE HAUTE DIVISION, "MID Cent. Operating Eng'rs Health & Welfare Fund v. Hoosiervac LLC, No. 2:24-cv-00326-JPH-MJD - Document 99." Feb. 24, 2025. Accessed: Mar. 12, 2025. [Online]. Available: https://storage.courtlistener.com/recap/gov.uscourts.insd.215482/gov.uscourts.insd.215482.99.0.pdf

[61] UNITED STATES DISTRICT COURT - SOUTHERN DISTRICT OF NEW YORK, "Mata v. Avianca, Inc., No. 1:2022cv01461 - Document 54." Jun. 22, 2023. Accessed: Mar. 12, 2025. [Online]. Available: https://cases.justia.com/federal/district-courts/new-york/nysdce/1:2022cv01461/575368/54/0.pdf?ts=1687525481

[62] "Google's Bard AI bot mistake wipes $100bn off shares," Feb. 08, 2023. Accessed: Mar. 09, 2025. [Online]. Available: https://www.bbc.com/news/business-64576225

[63] D. Otero, J. Parapar, and Á. Barreiro, "On the Statistical Significance with Relevance Assessments of Large Language Models," Nov. 20, 2024, *arXiv*: arXiv:2411.13212. doi: 10.48550/arXiv.2411.13212.

[64] S. Yang, W.-L. Chiang, L. Zheng, J. E. Gonzalez, and I. Stoica, "Rethinking Benchmark and Contamination for Language Models with Rephrased Samples," Nov. 11, 2023, *arXiv*: arXiv:2311.04850. doi: 10.48550/arXiv.2311.04850.

[65] Y. Oren, N. Meister, N. Chatterji, F. Ladhak, and T. B. Hashimoto, "Proving Test Set Contamination in Black Box Language Models," Nov. 24, 2023, *arXiv*: arXiv:2310.17623. doi: 10.48550/arXiv.2310.17623.