Is this for real? Hallucination Analysis in LLMs

Zhengjie Ji
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060
zhengjie@vt.edu

Sindhuja Madabushi
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060
msindhuja@vt.edu

Abstract

In this project, we study the phenomenon of hallucination in large language models (LLMs), where the models generate inaccurate or fabricated information. To understand and detect these hallucinations, we present a methodology for creating a dataset that triggers LLMs to produce hallucination responses, specifically within the framework of context-based question-answering tasks. We evaluate LLMs on two types of questions, COUNT questions and YESNO questions. To better understand the impact of LLM parameters on hallucination, we pertubate LLM parameters such as temperature, presence penalty, frequency penalty, and top-p sampling, and observe their influence on hallucination rates. Our evaluation reveals consistent hallucination occurrence in COUNT questions, with a slight increase observed in YESNO questions. We also proposes future research directions to improve detection methods of hallucinated outputs.

1 Introduction

Large language models (LLMs) such as ChatGPT[1], Claude (Bai et al., 2022), and Llama-2 (Touvron et al., 2023) are extensively used for personal and business needs, achieving widespread popularity and integration into numerous applications. Despite their great success, LLMs tend to "hallucinate" and may generate inaccurate/fabricated information (Huang et al., 2023; Ye et al., 2023). Fig. 1 shows an example query for which GPT-4 hallucinates. Due to the increased reliance on LLMs, the hallucination problem can pose a serious threat to security and degrade users' trust (Gupta et al., 2023). For example, the use of LLM-integrated applications in medicine may generate medical reports that contain highly convincing yet entirely false information, possibly leading to life-threatening treatments or lack thereof. Therefore, detecting hallucinations of LLMs is of primary importance.

2 Related Work

The literature only contains a few datasets for hallucination detection. HADES (Liu et al., 2021) is created by perturbing text segments from the English version of Wikipedia and verifying them using crowd-sourced annotations. HalOmi (Dale et al., 2023) includes professional annotations

of hallucinations and omissions for 18 language pairs. AutoHall (Cao et al., 2023) proposes an approach for automatically constructing model-specific hallucination datasets using existing fact-checking datasets. The aforementioned datasets do not consider the effect of varying decoding strategies on text generation. In contrast, we construct a hallucination dataset for LLMs by varying key LLM parameters such as temperature, presence penalty, frequency penalty, and top-p sampling, which may trigger hallucinations. We then create an annotated question-answer (QA) dataset by providing the LLM with source inputs. This report is organized as follows. Section 3 provides

[1] https://chat.openai.com/

Preprint. Under review.

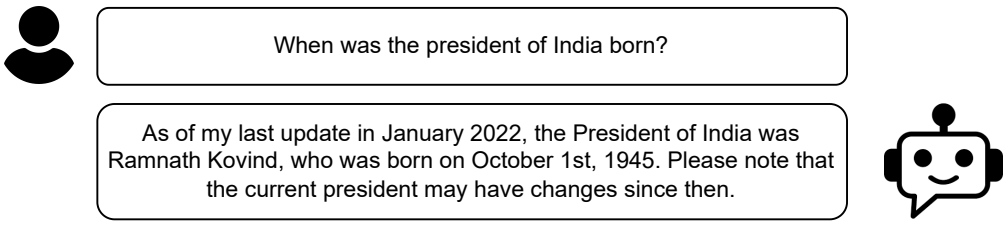

Figure 1: Example of hallucinatory response generated by GPT-4 (Sept. 2023)

background on hallucination in LLMs. Section 4 describes our dataset creation methodology in detail. Section 5 presents our main results. Sections 6 discuss our findings and list some limitations of our approach. Also, we conclude with avenues for future work.

3 Background

We introduce terminology pertaining to hallucination in LLMs. Hallucination is the phenomenon in which an LLM generates a factually incorrect response that deviates from the source input with fluency, authenticity, and confidence. There are two types of hallucination in LLMs (Ji et al., 2023):

Intrinsic hallucination: The LLM responds with altered information that contradicts the source material. For example, suppose the LLM responds with "Taylor Swift won five grammy awards" to the query "How many grammy awards did Taylor Swift win?" This response contradicts known information as the model is *likely* aware that the correct answer is twelve, not five.

Extrinsic hallucination: The LLM response includes additional information that cannot be directly inferred from source material (Maynez et al., 2020). Fig. 2 shows an example query for which ChatGPT (gpt-3.5-turbo-0301) cannot possibly know the right answer (since it was trained prior to LangChain's release). Its response includes incorrect information about "LLM tokens" and the connection to "LLMChain."

Source material varies by task. For example, in abstractive summarization, the input text itself acts as source material. However, in Generative Question Answering (GQA), the source can be the knowledge learned by the model during training. In our work, we set our source material to be an input context. We describe our setup in detail later in Section 4. *Factuality* and *Faithfulness* are two other important terms related to hallucinations in LLMs. Factuality refers to generated responses aliging with world knowledge and faithfulness refers to generated responses aligning with source inputs (Ji et al., 2023). We now explain the decoding strategies we modified to intentionally induce hallucinations in our dataset.

3.1 LLM Parameters

LLM parameters are settings that can be adjusted to fine-tune text generated by LLMs. They can greatly influence the quality and diversity of LLM-generated text. Key parameters include temperature, number of tokens, top-p.

Temperature: The temperature controls the behavior of the softmax function, which outputs a probability distribution over the set of possible next tokens. Lower temperatures assign higher probabilities to tokens that the model considers to be more probable, yielding more conservative outputs. With increasing temperature, the probability distribution converges to the uniform distribution over the tokens, yielding more creative/unexpected LLM outputs.

Presence penalty: The presence penalty encourages the LLM to use different tokens, thereby promoting variety and relevance. This is achieved by keeping track of tokens already used by the model and decreasing the (log) probability of choosing these tokens again using a penalty parameter. Higher penalty parameters yield more diverse LLM outputs.

Frequency penalty: The frequency penalty discourages the LLM from repeating the same words or phrases frequently within its output. This is achieved by increasing the (log) probability of a

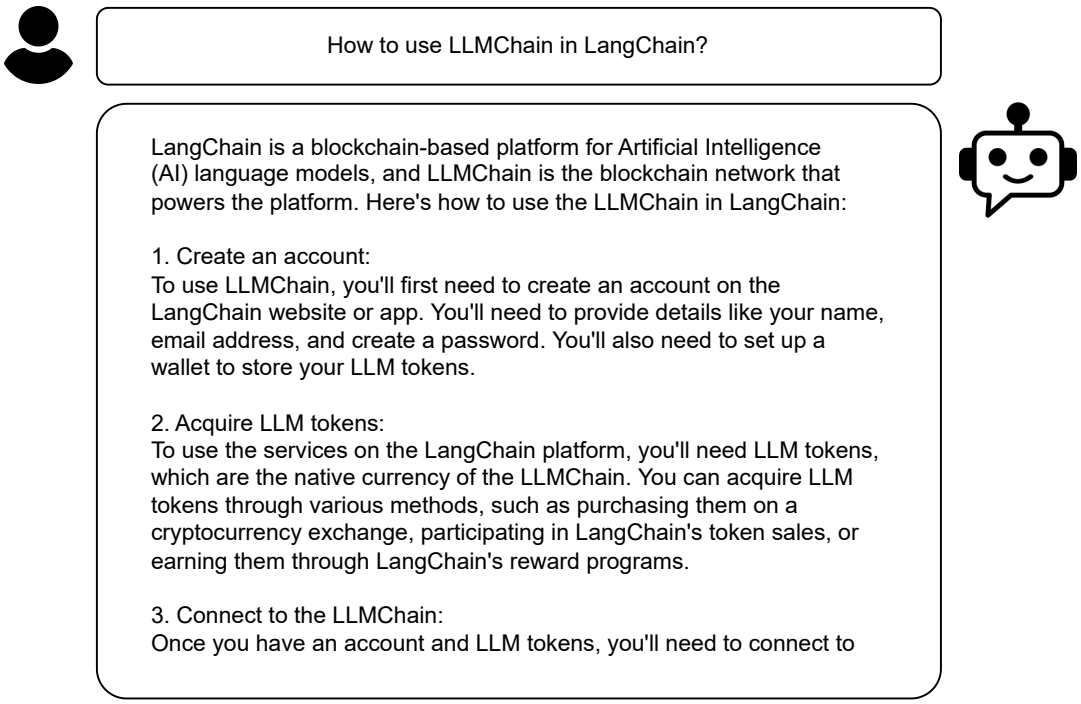

Figure 2: Example of extrinsic hallucination by ChatGPT (adapted from https://www.pinecone.io/learn/ai-hallucinations/)

token each time it occurs in the generated text. Higher frequency penalties make the model more conservative in the use of repeated tokens.

Top-p sampling: Top-p sampling provides a compromise between randomness and predictability. Before generating the next word, the model calculates the probability of each possible word. It then orders these words in decreasing order of their probability and adds the probabilities until the cumulative probability reaches the threshold p. Only words within this cumulative probability range are considered for selection.

4 Methodology

We study the hallucinating problem of LLMs on the context-based Question-Answering (QA) dataset. Our goal is to build a customized testing dataset. We focus on two question types for our QA dataset: COUNT and YESNO. We chose these questions because they are easier to annotate. For COUNT questions, we design a prompt for LLM to ask for a specific number clearly stated in the context. The expected answer is a numerical value. For example, if the text states that "three apples" are on the table, the model should precisely output the number "three" as a response. For YESNO questions, we design prompt based on facts that the model should infer from the context, with answers being a simple "yes" or "no."

We aim at generate a QA dataset with a greater chance of triggering hallucinations by selecting LLM parameters. We first construct a developing dataset using different combinations of LLM parameters. By observing how each parameter set impacts the frequency of hallucinations, we identify the parameter combinations that maximize the occurrence of hallucinations and optimize the distribution of these parameters accordingly. With the selected parameters, we then build the final testing dataset. We test a LLM using this final dataset and compare the results with those obtained from the developing dataset. Our comparisons aim to demonstrate that the final testing dataset can trigger more hallucinations.

4.1 Ground Truth Labeling

To label the ground truth, we selected 26 diverse contexts from the SQuAD dataset. For each context, we ask LLM to generate 5 questions based on the context for each question type. Then, we manually annotate the answers for those questions to form the original seed QA pairs. We observed about 10% hallucination questions in our human annotation process. We pruned those questions. For both COUNT and YESNO questions, we assigned the ground truth with a lowercase English phrase to maintain consistency and avoid ambiguity. For example, the number "7" would be labeled as "seven," and "45.10" as "forty-five point ten." If a question's answer cannot be inferred from the context, we label it as "unknown". In total, we manually annotated 260 seed QA pairs across the 26 contexts. This annotated ground truth seed dataset is included in the project's source code.

4.1.1 Construction of the Developing Dataset

In this section, we introduce the detailed steps in constructing the developing dataset, and how we perform LLM parameter selection. To generate more QA pairs with various parameter combinations, we implement an automated method to generate multiple variations of each question. We fix the answers to avoid the need for re-annotation, and use the LLM to paraphrase the questions. The idea is to manipulate different LLM parameters while asking the LLM to generate questions variations with the same question intent. Our goal is to evaluate the impact of these parameters on the likelihood of generating hallucinated answers. The LLM parameters are introduced in Section 3.1. We experiment with four different values for temperature, presence penalty, and frequency penalty (1.2, 1.4, 1.6 and 1.8) and two values for top-p sampling (0.92 and 0.95). Due to the constraints of using the OpenAI API, we limit our parameter selection process to five contexts from our labeled dataset. For each of the seed QA pairs, the LLM is prompted to generate 128 distinct question variations. We construct a developing dataset of 3,200 questions in total.

4.1.2 Parameter Selection

After constructing the developing dataset, we use the LLM to generate responses to all variations, and applied the metric described in Section 4.4 to detect hallucinating responses. We apply this process both COUNT and YESNO questions, and record the specific parameter configurations leading to most hallucinating outputs.

4.2 Construction of the Testing Dataset

With the parameter set determined, we construct our testing dataset. This dataset is generated according to the parameter configurations that are most likely to cause hallucinations, as determined from our developing dataset. We follow the same procedure as for the question variations and apply the selected parameters to generate questions for all 25 contexts. For each seed QA pair, we create 10 new variations.

4.3 Evaluating Hallucinations

We test the LLM (GPT-3.5 in our case) for hallucination using our final testing dataset. The responses are then evaluated using the same metrics. We expect that the selected parameters used in the testing dataset will lead to a higher chance of hallucination compare to the developing dataset, which will enable us to better assess the models' reliability.

4.4 Evaluation Metrics

Exact Match (EM): For each QA pair, if the ground truth answer is one of the words in the LLM-generated answer, EM = 'no' (hallucination), otherwise EM = 'yes'. This is different from a typical exact match that looks to match all words in the LLM-generated answer with the ground truth. We choose this metric because our ground truths are one-word answers (a detailed experimentation revealed that LLMs rarely generate outputs with just one word even when prompted to do so). This metric has limitations. For example, consider the following QA scenario. Q: How many Grammy awards did Taylor Swift win? A: Taylor Swift won twelve Emmy awards and five Grammy awards. The above response is hallucinated, but still has the ground truth answer "twelve" in it. However, our EM metric flags this as a non-hallucinating response. We manually checked a few samples and found

such instances to be rare; however, our results do not exclude these instances.

LLM Self-evaluation: In order to overcome the limitations of our exact match metric, we use another metric known as LLM self-evaluation. This metric is inspired by Luo et al. (2023). We fed the LLM with the prompt "Determine the consistency of the answer with the ground truth with a 'yes' or 'no' response. Note that consistency measures how much information in the ground truth is present in the answer. The answer can be in different formats. Context: {context} Ground Truth: {ground_truth} Answer: {answer}yes/no" to check the consistency of its own answer with the ground truth.

5 Evaluation

We evaluate GPT-3.5 on both developing and testing datasets. We detect hallucinating responses through the LLM self-evaluation metric. The results are shown in Table 1. The results show that while COUNT questions maintain a steady hallucination ratio despite the testing conditions, YESNO questions show a greater vulnerability to parameter settings. The ratio of hallucinating samples in COUNT questions remained consistent at 11.6% across both datasets. This does not meet our expectations as we aims at amplifying the phenomenon of hallucination. This implies a certain invariance of the hallucination rate for COUNT questions. YESNO questions presented a slight increase in the ratio of hallucinations, from 20.7% in the development dataset to 21.3% in the testing dataset. Although the increase is minimal, it suggests that the factors causing hallucinations in YESNO scenarios may be more susceptible to parameter manipulations compared to COUNT questions.

	Develop Dataset		**Testing Dataset**	
Question Type	**COUNT**	**YESNO**	**COUNT**	**YESNO**
# Hallucinating Samples	372	662	116	213
# Total Samples	3200	3200	1000	1000
Ratio	11.6%	20.7%	11.6%	21.3%

Table 1: The number of hallucinating samples for COUNT and YESNO questions evaluated on the develop dataset / testing dataset. LLM self-evaluation metric is used for hallucination detection.

5.1 The Influence of Parameters

The results in Tables 2 and 3 illustrates the influence of various LLM parameters on the generation of hallucinating responses. For COUNT questions, the number of hallucinating samples show a non-linear relationship with changes in temperature and penalties. Notably, the temperature, frequency penalty, and presence penalty at values of 1.4 induce the highest number of hallucinations. Top-p value at 0.95 has more hallucination sample counts.

Parameters	**Temperature**		**Frequency Penalty**		**Presense Penalty**		**Top P**	
	Value	**Count**	**Value**	**Count**	**Value**	**Count**	**Value**	**Count**
	1.2	85	1.2	92	1.2	87	0.92	176
	1.4	98	1.4	98	1.4	103	0.95	196
	1.6	96	1.6	93	1.6	92		
	1.8	93	1.8	89	1.8	90		

Table 2: The number of hallucinating samples for COUNT questions evaluated on the develop dataset

For YESNO questions, the temperature, frequency penalty, and presence penalty at values of 1.8 induce the highest number of hallucinations, which is different from the results for COUNT questions. A similar behavior is observed for YESNO questions for top-p, where its settings have a great impact on the hallucination rate. The top-p value of 0.95 again result in the highest count of hallucinating samples (341 occurrences).

Parameters	Temperature		Frequency Penalty		Presense Penalty		Top P	
	Value	Count	Value	Count	Value	Count	Value	Count
	1.2	166	1.2	168	1.2	161	0.92	321
	1.4	185	1.4	154	1.4	164	0.95	341
	1.6	163	1.6	164	1.6	164		
	1.8	148	1.8	176	1.8	173		

Table 3: The number of hallucinating samples for YESNO questions evaluated on the testing dataset

6 Discussion

6.1 Limitations

Our study has identified several limitations. First, the number of context we work with is limited. This limits the breadth of our findings to a wider set of scenarios. To improve the generalizability of our results, future efforts should focus on expanding the dataset with more diversed contexts.

The second limitation is that the observed impact of LLM parameters on hallucination generation was less marked than initially expected. This indicates that LLM might be more robust than we expected regarding the LLM parameters. To better understand the impact of LLM parameters, future research may collect a larger volume of annotated ground truth data, which could enable more precise tuning of parameters.

The sparsity of parameters is the third limitation. This also poses a challenge to our understanding of how different settings affect hallucination rates. More granular testing could help in discovering finer patterns in how LLMs respond to different configurations.

6.2 Future Work

Based on the limitations we have discussed, possible future work will be focused on refining the dataset generation and evaluation approach. This would provide a more robust assessment of LLM capabilities and limitations. Furthermore, we can test an extended range of parameter combinations to trigger hallucinations. We aim to identify optimal configurations that reach a balance between creativity and factual accuracy in LLM-generated outputs.

7 Conclusion

In conclusion, our evaluation reveals that LLMs exhibit a consistent rate of hallucinations in COUNT questions, with minimal variance observed despite changes in testing conditions and parameter configurations. However, YESNO questions showed a higher susceptibility to parameter adjustments. As LLMs become more integrated into various domains, understanding and reducing hallucination is essential to ensure the reliability and trustworthiness of machine-generated content.

8 Statement of Work

Zhengjie Ji:

- Suggested dataset generation methodology and refinements to evaluation metrics
- Contributed equally to coding and writing efforts

Sindhuja Madabushi:

- Suggested evaluation metrics and refinements to the dataset generation methodology
- Contributed equally to coding and writing efforts

References

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. (2022). Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*.

Cao, Z., Yang, Y., and Zhao, H. (2023). AutoHall: Automated hallucination dataset generation for large language models. *arXiv preprint arXiv:2310.00259*.

Dale, D., Voita, E., Lam, J., Hansanti, P., Ropers, C., Kalbassi, E., Gao, C., Barrault, L., and Costa-jussà, M. R. (2023). HalOmi: A manually annotated benchmark for multilingual hallucination and omission detection in machine translation. *arXiv preprint arXiv:2305.11746*.

Gupta, M., Akiri, C., Aryal, K., Parker, E., and Praharaj, L. (2023). From ChatGPT to ThreatGPT: Impact of generative AI in cybersecurity and privacy. *IEEE Access*.

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., et al. (2023). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Liu, T., Zhang, Y., Brockett, C., Mao, Y., Sui, Z., Chen, W., and Dolan, B. (2021). A token-level reference-free hallucination detection benchmark for free-form text generation. *arXiv preprint arXiv:2104.08704*.

Luo, Z., Xie, Q., and Ananiadou, S. (2023). ChatGPT as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621*.

Maynez, J., Narayan, S., Bohnet, B., and McDonald, R. (2020). On faithfulness and factuality in abstractive summarization.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ye, H., Liu, T., Zhang, A., Hua, W., and Jia, W. (2023). Cognitive mirage: A review of hallucinations in large language models. *arXiv preprint arXiv:2309.06794*.

9 Appendix

Description of our methodology using examples:

Step1: Given a context, generate n=5 questions [0.1in] Prompt: “Based on the following context, generate five ’count’ questions. A ’count’ question’s answer is always a number. A ’count’ question’s answer is not a date or a year: ”

Context: Chopin’s public popularity as a virtuoso began to wane, as did the number of his pupils, and this, together with the political strife and instability of the time, caused him to struggle financially. In February 1848, with the cellist Auguste Franchomme, he gave his last Paris concert, which included three movements of the Cello Sonata Op. 65.

COUNT:

- How many pupils did Chopin have?
- How many concerts did Chopin give in Paris?
- How many movements were in the Cello Sonata Op. 65?
- How much money did Chopin struggle to make?
- How many February 1848s were there?

Examplees of YESNO questions for the above scenario:

- Did Chopin’s popularity as a virtuoso begin to decline in the 1830s?
- Did the number of Chopin’s pupils also start to decrease during this time?
- Was Chopin’s financial situation worsened by the political instability of the 1840s?
- Did Chopin give his last Paris concert in February 1848?
- Did the program for this final concert include the Cello Sonata Op. 65?

Step2: Answer these questions manually (Seed QA pairs) [0.1in]

- How many pupils did Chopin have?Ans: unknown
- How many concerts did Chopin give in Paris?Ans: unknown
- How many movements were in the Cello Sonata Op. 65?Ans: three
- How much money did Chopin struggle to make?Ans: unknown
- How many February 1848s were there?Ans: one

Step3: For each of these questions, generate more questions using different parameters:

prompt: “Rephrase the question without changing its meaning and answer” Select parameters:

temperature: [1.2, 1.4, 1.6, 1.8] presence penalty: [1.2, 1.4, 1.6, 1.8] frequency penalty: [1.2, 1.4, 1.6, 1.8] top_p: [0.92, 0.95]

Example: parameters: t = 1.4, fp = 1.8, pp = 1.2, tp = 0.92

seed question: How many pupils did Chopin have?

generated question: How many students did Chopin teach?

Step 4: Ask LLM to generate answers for all our questions

Prompt:

“Based on the following context, answer the question:”

Context: Chopin’s public popularity as a virtuoso began to wane, as did the number of his pupils, and this, together with the political strife and instability of the time, caused him to struggle financially. In February 1848, with the cellist Auguste Franchomme, he gave his last Paris concert, which included three movements of the Cello Sonata Op. 65.

Question: How many pupils did Chopin have?

Step 5: evaluate all the answers using our metrics

LLM Self - evaluation:

Prompt: “Determine the consistency of the answer with the ground truth with a ’yes’ or ’no’ response. Note that consistency measures how much information in the ground truth is present in the answer. The answer can be in different formats. Context: {context} Ground Truth: {ground_truth}Answer: {answer}yes/no:”

Examples of hallucinations generated by GTP-3.5 for different parameters:

"parameters": [1.2, 1.2, 1.8, 0.92],
"question": "How many countries does Theravada Buddhism primarily exist in?", "answer": "six.",
"exact_match": "yes",
"llm_selfevaluation": "No"
ground truth answer: "nine"

"parameters": [1.4, 1.8, 1.4, 0.92],
"question": "Question: How many countries does Theravada Buddhism primarily have a presence in? "answer": "From what I can tell, there are four: Sri Lanka, Myanmar (Burma), Thailand, and Cambodia.",
"exact_match": "yes",
"llm_selfevaluation": "Yes"

"parameters": [1.6, 1.6, 1.2, 0.95],
"question": "How many countries does Theravada Buddhism primarily exist in?", "answer": "according to the context, theravada buddhism primarily exists in eight countries: sri lanka, burma, laos, thailand, cambodia, china, vietnam, and bangladesh.",
"exact_match": "yes",
"llm_selfevaluation": "no"