P/A Media

The secret behind LLM hallucinations

and how to tackle them

Thomas Reißer
Head of SEO at PIA Media GmbH in Munich

July 2024

contents

What exactly are Large Language Models?

Large Language Models (LLMs) are a fascinating development in the field of artificial intelligence. These models undergo extensive training with huge data sets and consist of complex neural networks in many layers that enable them to recognise patterns in the data and apply them to new inputs. One of their most notable capabilities is the generation of various creative text formats, including poems, code, scripts, emails and letters. In addition, LLMs are able to answer questions comprehensively and informatively, even if they are open-ended, complex or unusual. Another significant skill is to translate languages, preserving both the semantic content and the style and tone of the original language.

However, it is important to emphasise that LLMs do not perform a true creative process. They generate text based on probabilities and the number of parameters in the training and only use content that was included in the training data. While they deliver impressive results and can handle a wide range of tasks, their performance relies on the efficient pattern recognition system they develop during training. The fact that they do not generate new ideas or concepts, but merely recombine existing information, is an important aspect to consider when assessing their abilities.

What are LLM hallucinations?

An important aspect when analysing LLMs are the so-called LLM hallucinations, which occur when a generated text appears syntactically correct, fluent and natural, but is factually incorrect, nonsensical or inappropriate for the given input.

LLM hallucinations differ from other errors in that they are intrinsically false and misleading. They involve the generation of information that is not grounded

in reality, regardless of the accuracy of the underlying data. This distinction is important to accurately understand and critically evaluate the potential and limitations of LLMs. In many use cases, especially those that rely on accurate and correct information, such as medical or legal advice, LLM hallucinations can lead to significant problems. It is therefore crucial to develop mechanisms to detect and minimise such errors in order to increase confidence in the reliability of LLMs.

Why is it important to tackle the problem?

Hallucinations of LLMs can have serious consequences, such as the spread of misinformation, data breaches and security concerns for real-world applications. For example, a hallucinated report generated from patient information in the medical field can pose a serious risk to the patient. Such hallucinations ultimately also affect the general trust of users in this technology, which is why it is important to address this problem quickly.

A real-life example is the case of a professor at Texas A&M University who failed his entire student body after ChatGPT falsely claimed their papers were written by AI. This resulted in many students being denied their degrees, which not only jeopardised their academic careers, but also undermined confidence in the reliability of such technologies. [1]

ChatGPT also made a false accusation of sexual harassment against George Washington University law professor Jonathan Turley. The AI model invented a non-existent Washington Post article and falsely accused Turley of harassing a female student during a class trip. Such incidents show how dangerous and misleading AI-generated content can be, and emphasise the need to establish stricter vetting mechanisms and ethical guidelines for the use of AI. [2]

What causes hallucinations?

The quality of the training data plays a crucial role in the performance of Large Language Models (LLMs). Inconsistencies, inaccuracies, biases and prejudices in the training data can have a significant impact as they can lead to errors in generation. If the model takes incorrect information from the training data, this can lead to inaccurate or misleading outputs. Similarly, incomplete datasets that have limited diversity and contextual information can limit the LLM's ability to produce relevant and accurate results.

While it is important to promote diversity in the training data to improve the model's performance, over-correcting for diversity can also lead to incorrect results, as too many unrelated or unrepresentative data points may be included in the training. Therefore, careful selection and cleaning of the training data is crucial to ensure that the LLM delivers optimal results.

1. Input: Vague or imprecise wording and a lack of context in the input can cause the Large Language Model (LLM) to misunderstand the user's intent. This in turn can lead to unpredictable or incorrect results when generating text. It is therefore important to formulate clear and precise input and provide additional context to improve the accuracy and relevance of the output generated by the LLM.

Example Legal advice: "Is it ok to film my neighbours?" The answer could vary depending on whether the person is filming in a public space or in their own home, whether it's surveillance or personal footage. Without enough context, an AI model could create hallucinations and make assumptions about what is legally acceptable. This could lead to false conclusions and potentially get the user into legal trouble.

2. Context: In longer conversations, there is a risk that the Large Language Model (LLM) will lose context and generate contradictory statements. In addition, the model may have difficulty taking the context into account appropri-

ately and adapting the generation to the specific situation. This can lead to inappropriate or irrelevant answers, as the context is crucial to understanding the user's intention.

3. Factual knowledge: Although large language models (LLMs) can access large amounts of data, their information base is still limited. Although they have a considerable repertoire of knowledge, they do not have access to all the knowledge in the world. This can lead to LLMs providing incomplete or inaccurate information on certain topics or specialised areas. In addition, LLMs may have difficulties in verifying facts, as incorrect information may be difficult to recognise in retrospect. This emphasises the need for critical scrutiny of content generated by LLMs and highlights the importance of human monitoring and validation, especially for sensitive or critical information.

Example: Suppose a user asks a Large Language Model for the weight of Mount Everest. The model responds with a precise figure that appears to be correct. However, on closer inspection, it turns out that the model has simply used incorrect information from its training data. In fact, the weight of Mount Everest changes over time due to geological changes, so a static number is not an accurate answer. This example clearly shows the difficulty of LLMs in correctly capturing complex and changing facts and providing accurate answers based on them.

4. "Lost-in-the-Middle"[3]: In practice, it has been shown that even large language models with large context windows of 8K to 32K tokens can omit relevant details in the text, especially in the centre. This can lead to incomplete or inaccurate results or misinterpretations.

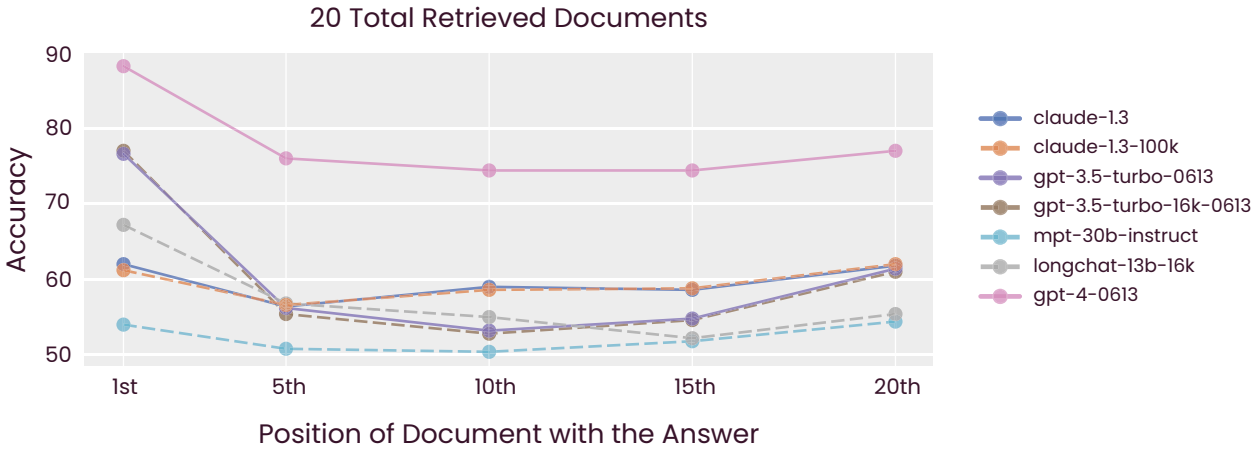

Figure 1: Lost-in-the-middle analysis (https://arxiv.org/pdf/2307.03172)

The reason for this lies in the limited processing capacity of the model. A context window of 8K-32K tokens may sound large, but it is often not enough to capture all the relevant information in a longer text. Therefore, the LLM may lose context or skip important details, which can affect the quality of the generated results. This aspect highlights the challenges faced by LLMs and the need to continuously improve their contextualization skills in order to achieve more accurate and comprehensive answers.

5. "Reasoning Gap": The "thinking gap" in current Large Language Models (LLMs) reveals a discrepancy between their performance on known problems compared to problems that can be solved spontaneously.

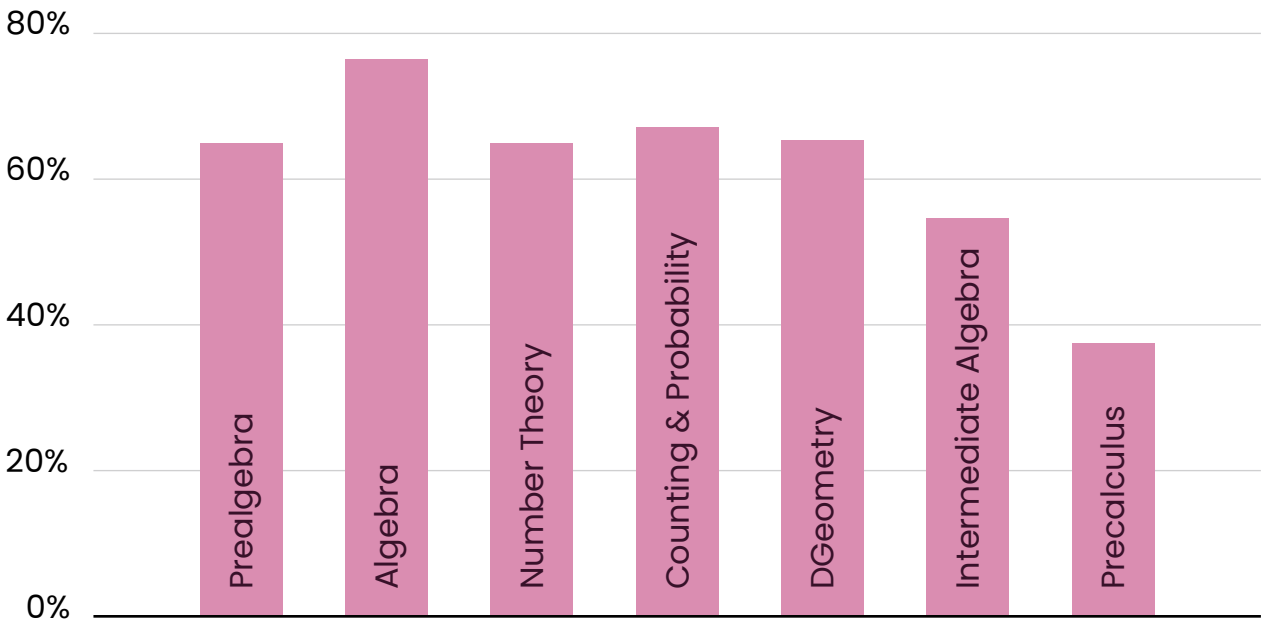

Figure 2: Gap across subjects, aggregated across all models.
(https://arxiv.org/pdf/2402.19450)

This discrepancy becomes clear when looking at the performance of LLMs such as GPT-4, which only successfully solves about 10% of 5,000 dynamic tasks. Current LLMs are in the first of four stages of generalization capability.

They can provide answers to a static set of tasks and interpolate between them, but lack a true understanding of the problems. This highlights the existing limitations in the ability of LLMs to solve spontaneous or dynamic problems and emphasizes the need for further research and development to improve the cognitive and problem-solving ability of these models.[4]

6. 6. AI vulnerability to "Silent Data Corruptions"[5] (SDCs): AI models are susceptible to errors in their parameters. These errors, such as bit tipping points, can occur in the event of hardware problems and lead to incorrect model predictions. The Vulnerability Factor (PVF) parameter quantifies the probability that such errors lead to incorrect outputs. The results show that different parts of the model are vulnerable to different degrees, which is important for the development of robust AI hardware.

Faulty SDCs during the training of Large Language Models (LLMs) can significantly affect the accuracy and reliability of these models. SDCs are particularly problematic as they often go unnoticed and are therefore difficult to diagnose. If parameters in LLMs are corrupted by SDCs, the models can produce inaccurate or incorrect outputs. Therefore, it is important to develop mechanisms to detect and correct such errors to ensure the integrity of the models.

7. "Jailbreak" techniques: "Jailbreak" techniques refer to external manipulations of Large Language Models (LLMs) that aim to make the system generate illegal content. These techniques can include various methods, from the deliberate introduction of erroneous training data to the exploitation of vulnerabilities in the model itself.

Through such manipulation, LLMs can be tricked into generating inappropriate or illegal content, such as hate speech, pornography or other unlawful material. The use of "jailbreak" techniques poses a serious threat and requires appropriate security measures to ensure the integrity and trustworthiness of LLMs.

8. "ArtPrompt" technique: The "ArtPrompt" technique [6] refers to a method of external manipulation of Large Language Models (LLMs) that aims to make the model generate certain terms that are normally locked or restricted. This technique works by first replacing the blocked term with a mask. The second step is to replace this mask with ASCII art that visually represents the blocked term.

In this way, LLMs can be made to generate content that is offensive, insulting or illegal without using the blocked term directly. The "ArtPrompt" technique poses a challenge to the security and integrity of LLMs and requires appropriate countermeasures to prevent abuse and ensure the quality of the generated content.

Approaches to minimize hallucinations

Strategies on the operator side

1. Data cleansing: Data cleansing is a crucial step in the process of preparing training data for Large Language Models (LLMs). It involves removing inconsistent, inaccurate, biased and irrelevant information from the training set. This process ensures that the model accesses high quality and reliable data to produce accurate and consistent results. Inconsistencies or errors in the training data could lead to the LLM adopting incorrect information or generating inaccurate answers, which would affect the performance and trustworthiness of the model.

2. Data extension: Data augmentation is another important step to improve the quality and variety of the training data. By adding diverse and contextualized data, gaps in the training set can be closed and the performance of the LLM improved. This can be achieved by incorporating different sources and data sources to cover a wider range of information and ensure that the model has a comprehensive repertoire of knowledge. Careful data augmentation helps to improve the generalization capability of the LLM and strengthen its ability to successfully handle different tasks and use cases.

The paper "Agent Planning with World Knowledge Model"[7] discusses in detail the possibility of a targeted integration of relevant "world knowledge" already during training and later during generation.

3. Validation and verification: Validation and verification of the training data are essential steps to ensure that the data is of high quality and correct. By introducing mechanisms to verify the accuracy of the data, potential errors or inconsistencies can be identified and corrected at an early stage. This can be done, for example, through manual review, automated testing or the use of external validation sources. Thorough validation and verification ensures that the LLM accesses reliable and correct information, which in turn increases the

accuracy and reliability of the generated results and strengthens confidence in the model. The open source project FineWeb offers a possible solution here.[8] Important basic research for a better understanding of the inner workings of LLMs was recently provided by the team at Anthropic with their extraction of millions of features from their model Claude 3 Sonnet.[9]

4. Dialog tracking: With dialog tracking, the course of the conversation is saved and tracked. This technique enables the Large Language Model (LLM) to better understand the context of an ongoing conversation and respond appropriately. By taking past interactions into account, the LLM can generate more relevant and coherent responses, resulting in an improved user experience.[10]

a. Contextual generation: Contextual generation refers to the fact that the LLM adapts its response to the current context of the conversation. By taking into account the context of the ongoing conversation, the model can provide more accurate and relevant answers. This reduces the chances of misunderstandings or inconsistent answers and improves the quality of communication.

b. Recognition of contradictions: The detection of contradictions is an important aspect of dialog processing. The LLM continuously checks the course of the conversation to identify and avoid inconsistent statements or contradictory information. By avoiding contradictions, the model can provide credible and consistent answers, which increases the effectiveness and reliability of communication.

5. Manipulative prompts: The detection of manipulative prompts includes the filtering and blocking of "jailbreak" techniques that aim to mislead the Large Language Model (LLM) into inappropriate behavior. Algorithms and security measures are used to detect and prevent such manipulation attempts in order to protect the integrity of the model and prevent the generation of inappropriate or illegal content.

6. Access control: The limitation of access options concerns the control of users who can influence the LLM. The implementation of access controls and authorizations ensures that only authorized users have access to the model and can use it. In this way, the risk of misuse or unwanted manipulation is minimized and the security of the system is guaranteed.

7. Transparency and education: Transparency and education of users about how LLMs work and the risks of hallucinations is crucial. Users should be informed of the potential risks associated with the use of LLMs, particularly with regard to the possibility of hallucinations and the dissemination of misleading or inaccurate information. Through comprehensive education, users can better understand how to use LLMs safely and responsibly and take appropriate precautions to minimize the risks.

Strategies on the user side

The ability to optimize the prompt using Large Language Models (LLMs) is a key aspect that underlines the performance and versatility of these models. By cleverly designing the prompt, users can specifically influence and customize the generated results. This includes selecting specific keywords or phrases that encourage the LLM to generate more precise or relevant answers.

In addition, varying the prompt allows users to explore different facets of a topic or obtain different results. This flexibility opens up a variety of application possibilities and offers users the opportunity to tailor the performance of the LLM to their specific needs and requirements.

Optimizing the prompt thus represents an effective approach to increasing the efficiency and effectiveness of Large Language Models and making the most of their capabilities.

1. Formulation of clear and precise prompts: Vague wording often leads to misunderstandings as the model tries to provide a broad interpretation of the query. If you are specific and state exactly what you need, the model can generate more precise and relevant answers.

Example: “How do I fix my computer?” Better: “My laptop shows a blue screen and won’t start. What can I do?”

Relevant information that sheds light on the background of your query is essential for the accuracy of the answers generated. By adding context, the model better understands the context in which your question is being asked and can provide more detailed and contextually appropriate answers accordingly.

Example: “How can I cook better?” Better: “I would like to learn how to prepare a vegetarian dish with quinoa.”

Ambiguities and ambiguous wording can lead to misunderstandings and affect the quality of the answers. Clear and direct language helps to avoid these problems and ensure that the model understands exactly what you are asking them to do.

Example: “Explain the bank to me.” Better: “I want to know more about how the banking system works.”

2. No false information: Explicitly instruct the model to avoid false information. Yes, this actually works. Often at least!

Example: “If you don’t know the answer, please don’t share false information!”

3. Few-shot prompting: Give concrete examples to guide the answers. This method requires high quality examples to avoid introducing bias or inaccuracies.

4. Chain-of-thought prompting: You can guide the argumentation steps to the final answer. This can increase transparency and possibly reduce hallucinations. However, there is also a risk of introducing new hallucinatory reasoning steps.

Example:
Step 1: Pose a scientific question or hypothesis, e.g. "Investigate the effects of weightlessness on plant growth in space."

Step 2: Ask the model to analyze relevant research or theories on this topic and identify key findings.

Step 3: Gradually add more layers of information, such as experiment designs or methods to test the effects of microgravity on plants.

Step 4: Ask the model to draw a reasoned conclusion and discuss possible future research directions or applications to avoid hallucinations or unfounded assumptions.

5. Contrastive prompting: Present the model with contrasting examples or situations that help them to recognize and understand differences.

Example: Show contrasting product reviews so that the model can gradually identify the advantages and disadvantages of different products.

6. Temperature control: Setting the temperature value of a model lower leads to more conservative and predictable outputs, while a higher temperature value leads to more creative and less predictable outputs. Unfortunately, this can currently only be set in the API for many models.

7. Prompt optimization by the LLM: Use the following prompt before your actual prompt and the LLM will significantly improve your prompt and deliver better results.

Example: You are an experienced prompting engineer! Please optimize the following prompt to achieve the best possible result. If not provided, ask for your specific role, tone of voice and writing style, context, examples if applicable, and output format. Please ask me up to 5 other relevant questions that you need to know in order to do the job to the best of your ability. Then give me the restructured, optimized assignment in the form you can best interpret. Prompt: ...your actual prompt...

8. Bribery: It can actually help to offer the Large Language Model a reward, e.g. for a particularly detailed answer, or for effectively fixing a coding bug.

Example: "I'll pay you 10 EUR if you go into particular detail on the topic YX."

Blueprint for good prompting

The effective use of LLMs requires a well thought-out and clear structure of the prompts. By considering necessary, important and nice-to-have elements, the quality and relevance of the generated answers can be significantly improved. A well-designed prompting blueprint ensures that the model delivers precise and contextually appropriate results.

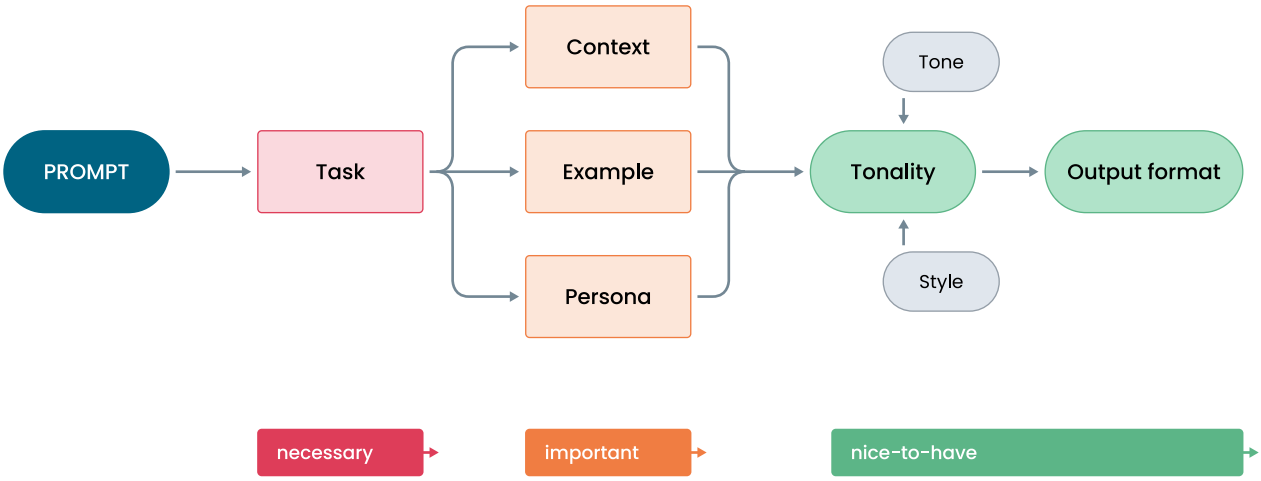

Figure 3: Flow chart (Thomas Reisser)

Necessary elements: The task definition is the core of every prompt. It clearly defines what exactly is expected of the model. A precise and unambiguous formulation of the task is crucial to ensure that the model understands and executes the request correctly.

Important elements: The context provides the model with all the necessary background information it needs to understand the task in the right context. This can include historical data, situational details or specific requirements. Without the right context, the model cannot respond optimally to the request.

Examples are valuable tools to clarify the desired result. They show the model what exactly is expected and can serve as a template for the answer. A good example helps the model to interpret the task better and deliver more precise results.

The persona describes the perspective or role from which the model should respond. This can be a specific character, an expert in a field or a specific target group. The right choice of persona ensures that the answers are written in a suitable tone and style. It is best to always specify this at the beginning of the prompt.

Nice-to-have elements: The output format specifies the form in which the result is to be presented. This can be a text format (CSV, JSON, etc.), a table, a diagram or another form of presentation. A clearly defined output format facilitates the further processing of the generated content.

The tonality defines the tone and writing style of the response. This can be informal, formal, descriptive, analytical or humorous. A specific reference to the desired tone helps the model to better adapt the answers to the desired style.

Example: Write in a casual, colloquial tone and use illustrative language.

Examples of tone: authoritative, casual, clinical, cold, confident, cynical, emotional, empathic, formal, friendly, humorous, informal, ironic, optimistic, pessimistic, playful, respectful, sarcastic, serious, sympathetic, hesitant, warm.

Examples of writing style: academic, analytical, argumentative, conversational, creative, critical, descriptive, epistolary, descriptive, informative, instructive, journalistic, metaphorical, narrative, persuasive, poetic, satirical, technical

Pro tip: Analyze excerpts of your own longer texts for delivery (tonality, writing style, reading level, address, perspective, flow and coherence, interactivity, emotional resonance) and include the results of the analysis in the details of your next prompt in order to have the LLM write in your own language style.

Footnotes

[1] https://www.rollingstone.com/culture/culture-features/texas-am-chatgpt-ai-professor-flunks-students-false-claims-1234736601/

[2] https://www.informed.so/articles/2023-04-05-chatgpt-invented-a-sexual-harassment-scandal-and-named-a-real-law-professor-as-the-accused

[3] https://arxiv.org/pdf/2307.03172

[4] https://arxiv.org/pdf/2402.19450

[5] https://arxiv.org/pdf/2405.01741

[6] https://arxiv.org/pdf/2402.11753

[7] https://arxiv.org/pdf/2405.14205

[8] https://huggingface.co/spaces/HuggingFaceFW/blogpost-fineweb-v1

[9] https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html

[10] https://arxiv.org/pdf/2310.14970

Literature

Saurabh Srivastava, Annarose M B, Anto P V, Shashank Menon, Ajay Sukumar, Adwaith Samod T, Alan Philipose, Stevin Prince, and Sooraj Thomas: Functional Benchmarks for Robust Evaluation of Reasoning Perfor-mance, and the Reasoning Gap. arXiv:2402.19450v1 [cs.AI], 29 Feb 2024 (https://arxiv.org/pdf/2402.19450)

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, Percy Liang: Lost in the Middle: How Language Models Use Long Contexts. arXiv:2307.03172 [cs.CL], 20 Nov 2023 (https://arxiv.org/pdf/2307.03172)

Hongbin Ye, Tong Liu, Aijia Zhang, Wei Hua, Weiqiang Jia: Cognitive Mirage: A Review of Hallucinations in Large Language Models, Zhejiang Lab 2023 (https://arxiv.org/pdf/2309.06794v1.pdf)

Ziwei Xu, Sanjay Jain, Mohan Kankanhalli: Hallucination is Inevitable: An Innate Limitation of Large Language Models, 2024 (https://arxiv.org/pdf/2401.11817.pdf)

Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, Li Yuan: LLM Lies: Hallucinations are not Bugs, but Features as Adversarial Examples, 2023 (https://arxiv.org/pdf/2310.01469.pdf)

Greg Kamradt (https://x.com/GregKamradt/status/1722386725635580292)

Shuofei Qiao, Runnan Fang, Ningyu Zhang, Yuqi Zhu, Xiang Chen, Shumin Deng, Yong Jiang, Pengjun Xie, Fei Huang, Huajun Chen: Agent Planning with World Knowledge Model. arXiv:2405.14205v1 [cs.CL] 23 May 2024 (https://arxiv.org/pdf/2405.14205)

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Colin Raffel, Leandro Werra, Thomas Wolf: FineWeb: decanting the web for the finest text data at scale. HuggingFace. May 31, 2024 (https://huggingface.co/spaces/HuggingFaceFW/blogpost-fineweb-v1)

Fengqing Jiang, Zhangchen Xu, Luyao Niu, Zhen Xiang, Bhaskar Ramasubramanian, Bo Li, Radha Poovendran: ArtPrompt: ASCII Art-based Jailbreak Attacks against Aligned LLMs. arXiv:2402.11753 [cs.CL]. 7 Jun 2024 (https://arxiv.org/pdf/2402.11753)

Templeton et al.: Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet. Anthropic. May 21, 2024 (https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html)

Yujie Feng, Zexin Lu, Bo Liu, Liming Zhan, Xiao-Ming Wu: Towards LLM-driven Dialogue State Tracking. arXiv:2310.14970 [cs.CL]. 23 Oct 2023 (https://arxiv.org/pdf/2310.14970)

Xun Jiao, Fred Lin, Harish D. Dixit, Joel Coburn, et al.: PVF (Parameter Vulnerability Factor): A Scalable Metric for Understanding AI Vulnerability Against SDCs in Model Parameters. arXiv:2405.01741 [cs.CR]. 11 Jun 2024 (https://arxiv.org/pdf/2405.01741)

P/A Media

About the author at PIA Media

Thomas Reisser
Head of SEO

T +49 172 833 37 36
E: thomas.reisser@piamedia.com
W: www.piamedia.com

About PIA Media

With its digital performance expertise, PIA Media is adept at driving business impact in a fragmented digital landscape, effectively designing brand, product, and service communication across all customer journey touchpoints.

PIA Media uniquely integrates media, platforms, technologies, data, strategy consulting, and high-level creation to propel its clients' businesses forward.

In line with this vision, PIA Media harnesses the skills of top digital players and over 400 digital experts from various disciplines, operating across six locations: Hamburg, Munich, Düsseldorf, Berlin, Belgrade, Bucharest, and remotely throughout Europe.