Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/76/20240608

A review of methods for alleviating hallucination issues in large language models

Zhibo Yin

Milton International School, Qingdao, 266075, China

karl_yin0428@163.com

Abstract. Large language models have demonstrated impressive language processing capabilities in recent years, exhibiting unparalleled excellence in the field of natural language processing. However, the generated text sometimes contains hallucinations, which is the text that contradicts the knowledge in the real world, the context, and the user input. This problem is mainly due to the inherent limitations of the method itself in aspects such as data quality, the model training process, and the model generation process. The issue of hallucinations has always been closely monitored by the academic community. It is widely recognized that its potential consequences should not be underestimated. This paper systematically summarizes the research on the causes of hallucinations in large language models, and introduces mainstream classification methods as well as current measures to address the issue of hallucinations. To be more specific, the article divides the causes of hallucinations into two categories: 1. hallucinations come from the training process and 2. hallucinations come from the generation process. Also, 4 typical types of causes for the former and 5 typical types of causes for the latter are provided. Simultaneously, a detailed discussion of 16 methods to mitigate hallucinations that arise in the generation process is offered. Finally, this paper also discusses inherent flaws that may exist in large language models, aiming to help people gain a more comprehensive understanding and research into hallucinations and large language models. In general, the text details about the hallucinations that exist in the large language model. Meanwhile, according to the previous research, it is pointed out that it is difficult for the large language model based on autoregressive method for token prediction to avoid the hallucinations completely.

Keywords: Large Language Model, Hallucination, Cause Analysis, Solution.

1. Introduction

Recently, large language models have demonstrated impressive performance in a wide range of downstream tasks, such as text classification, machine translation, code generation, sentiment analysis, and question-answering systems. At the same time, their unique ability for knowledge generalization is continuously assisting them in enhancing their problem-solving capabilities.

However, large language models often generate content that cannot be directly applied to real-world scenarios, as their outputs often contain harmful hallucination content. Hallucination refers to the phenomenon where models generate harmful, erroneous, incomplete, and self-contradictory dialogues, due to a variety of factors such as the data used in the model training process, the model training process, and the inference process. For instance, confusing concepts in different disciplines, forgetting or violating user input instructions, and generating text that contradicts the content previously generated.

DOI: 10.54254/2755-2721/76/20240608

Currently, the academic community has conducted research on various aspects related to the causes, classification, detection, and mitigation of hallucination issues.

This paper leverages hallucination-related features generated by large language models as clues to review and summarize the relevant research on the hallucination issues of large language models. Standing at a unique perspective of the factors that cause the model to hallucinate, this paper propose a three-step method to analyze these studies: conceptual approach, workflow, and achieved results. This paper sum up the causes, classifications, detection, and mitigation methods for LLM hallucination issues, while exploring the related research on the inherent flaws of large language models in the academic community, to better understand, grasp, and comprehend the current knowledge and major directions of research on LLM hallucination issues.

2. Background

In order to provide a more comprehensive and detailed description of the methods to mitigate hallucination issues, the relevant background knowledge on hallucination issues will be introduced in this section. In section 2.1, the exploration of the causes of hallucination issues in the academic field will be discussed. In section 2.2, the mainstream classification methods of hallucination issues will be elaborated upon.

2.1. Hallucination-causes

The causes of hallucinations can be roughly divided into two categories according to the process of model operation: hallucinations generated during the training phase (i.e., before inference) and hallucinations generated during the inference process.

2.1.1. Hallucination come from training process

(1). Data quality

Massive training data is a crucial factor for Large Language Models to achieve the natural language processing capabilities of today [1]. Currently, mainstream models exhibit performance that is highly correlated with the scale of their training data. The Scale-law of LLMs further elaborates on the impact of data volume on performance [2]. The manner in which training data is obtained by the model can be divided into two categories: the first is structured information, such as tables and questionnaires, and the second is unstructured information, such as articles from the internet and books. Structured information is easier to clean, but harder to acquire. Unstructured information, especially textual information from the internet, is easier to obtain but may contain errors, biases, or outdated information [3]. At the same time, this data is often huge in scale and difficult to clean, leading to the model continuously learning and accumulating these errors during training [4].

(2). Data compression

One view suggests that the essence of large language models is to compress the knowledge associated with input text, storing and using the knowledge learned from the context, thus, the model may experience information loss and distortion during this process and hallucinate in subsequent output tasks [5].

(3). Requirement for task

Different downstream tasks impose varying requirements on model responses, leading to certain informal expressions potentially being perceived as delusions in certain environments. Although this is typically corrected by fine-tuning, it remains an important source of delusion [6].

(4). Error in encode-decode process

During the pre-training process, it is possible for encoding and decoding to produce erroneous expressions, leading to the model misunderstanding specific text content, which in turn affects its vector representation [7]. This can potentially lead to the model's cognition of specific objects in subsequent downstream tasks being biased, thereby generating illusions.

(5). Exposure Bias

DOI: 10.54254/2755-2721/76/20240608

Incorporating data with different input formats in the pre-training and fine-tuning stages can lead to exposure bias issues [8]. This can cause a bias in the model's understanding of text inputs, subsequently affecting the generation process, ultimately leading to the emergence of hallucinated content.

2.1.2. Hallucination come from Generation process

(1). Hallucination created to maintain self-consistency

Some large language models output tokens in a sequential manner, which can lead to errors in the output process due to maintaining self-consistency, and the continuation of erroneous content without correction, even when prior tokens are known to be incorrect [7]. Some methods prevent such issues by altering the decoding strategy.

(2). Deficit of knowledge understanding

Some models may lack the semantic understanding of information, relying solely on low-level word co-occurrence patterns. This results in their inability to learn lexical entailment well, leading to erroneous outputs [9].

(3). Deficit of knowledge

A survey indicates that, in most LLMs, there is no sufficient knowledge to support their performance of question-answering tasks across different domains [10]. This problem may lead to the LLMs being forced to generate fabricated content with no factual basis during the output process.

(4). Hallucination caused by decoding strategy

Large language models generate text by iteratively predicting tokens until they reach a certain termination condition, at which point the decoding strategy chooses the next best token based on the probability distribution. However, the TOP-K, N-gram method used by the model during the generation process may not always lead to the most accurate token output [7]. Therefore, a new decoding strategy is needed.

2.2. Hallucination classification

Different views have been proposed in the past regarding the classification of hallucination problems. This paper introduces the current main classification method, which separates hallucinations into factual and faithful categories [11,12].

2.2.1. Factualness hallucinations. These types of hallucinations display characteristics in which the output content contradicts with the existing prior knowledge from the real world (i.e., text content learned during the pre-training process). Specifically, factual hallucinations can be further divided into six subcategories [13]: Entity-error Hallucination, Relation-error Hallucination, Incompleteness Hallucination, Outdatedness Hallucination, Overclaim Hallucination, and Unverifiability Hallucination.

2.2.2. Faithfulness hallucination: This type of hallucination displays characteristics in which the output contradicts with the context or user input content. For instance, the model's calculation of the age of "Xiao Ming" as 8 years old in the intermediate inference process and final answer is a form of loyalty hallucination, as user input informed the model that "Xiao Ming" is 28 years old in the year 2021.

3. Mitigation method

Current hallucination relief methods often modify the model generation process. According to the different emphases and corresponding hallucination causes of these methods, this paper classifies them into: 1. Reinforcement Learning and Contrastive Learning; 2. Methods based on external knowledge assistance; 3. Knowledge enrichment methods; 4. Zero-resource and model-based feedback methods; 5. Methods for improving decoding strategies. This categorization includes five types.

DOI: 10.54254/2755-2721/76/20240608

3.1. Reinforcement learning and contrastive learning

3.1.1. Human feedback reinforcement learning. Human Feedback Reinforcement Learning is a traditional reinforcement learning strategy. This method is widely used in the field of artificial intelligence [14,15]. One of the most notable successful cases using this method is ChatGPT-3.5. This method is extensively used in the fine-tuning stage of LLMs, and can effectively alleviate hallucinations caused by the multiple reasons mentioned above. Specifically, Human Feedback Reinforcement Learning utilizes a reward model trained on human social values to train LLMs. This reward model receives human rankings of LLM text outputs as training data, continuously learning human preferences, and eventually replacing human sorting of LLM responses. After sorting, LLMs generated by the reward model trained with this method will produce text, and the reward signal will act as feedback to modify their generation logic. Overall, Human Feedback Reinforcement Learning continuously improves and optimizes the information generated by LLMs. This method can improve the quality of answers generated by LLMs across various tasks, such as text generation, code generation, and mathematical problem solving. At the same time, it can effectively reduce the possibility of hallucinations or the production of other dangerous texts by the model.

3.1.2. Knowledge feedback reinforcement learning. Large Language Models (LLMs) often generate illusions when faced with issues beyond their internal knowledge. Knowledge Feedback Reinforcement Learning (KFR) utilizes fact preference as a reward to enhance Large Language Model (LLM) usage of its internal knowledge state, thereby enhancing the factual and honest nature of its dialogues. This method primarily alleviates illusions caused by insufficient internal knowledge and knowledge understanding of the LLM. Specifically, this method employs an automatic illusion tagging tool, Dreamcatcher, to label fact preference data. Marked data is then used to establish a reward model for fact preference. The model will be trained to help improve the honesty of the LLM through reinforcement learning. Meanwhile, the PPO algorithm will constantly optimize the Large Language Model. Experimental results demonstrate that Dreamcatcher can effectively detect knowledge states in all models, achieving an 81% similarity with human illusion annotators. This framework can effectively alleviate factual illusions.

3.1.3. Contrastive learning strategy. MixCL: This method achieves a reduction in hallucination phenomena in language models during dialog training through contrastive learning. It achieves this through a combination of two negative sampling and mixed contrastive learning. There are two approaches to negative sampling: one is to retrieve knowledge from a retrieval library, the other is to generate negative knowledge through model-guided generation. Mixed contrastive learning combines positive and negative examples at the sentence level during training to optimize the model. Our experiments demonstrate that MixCL performs favorably on the Wizard-of-Wikipedia dataset, achieving levels of knowledge correlation and factuality comparable to the state-of-the-art methods based on knowledge bases.

3.2. External knowledge assistant

This type of method modifies the model generation process, aiming to verify whether the output text generated by LLMs (Large Language Models) contains hallucinations through external knowledge assistance. This often relies on knowledge retrieval or knowledge graph assistance. These methods primarily target hallucinations caused by model inconsistencies, insufficient model knowledge understanding, or inadequate model internal knowledge during the model generation process. Here are a few examples to illustrate this:

3.2.1. Neural path hunter. This method proposes a hallucination detection and mitigation method assisted by knowledge graphs. It mainly targets hallucinations caused by self-consistency and

DOI: 10.54254/2755-2721/76/20240608

knowledge deficiency. The method consists of two major modules, Token-level hallucination critic and Entity Mention Retriever. The former is responsible for marking and blocking existing content containing hallucinated entities, and the latter collects hallucinated entity markers and assigns context representations to each marked entity, then sends these markers to an autoregressive LM for output representation. These outputs will be used to query the knowledge graph, ultimately returning the correct entities. This method can significantly reduce hallucinations in KG-grounded dialogue systems, relative to an improvement of 20.35% in FeQA scores and 39.98% in human assessment scores. When paired with multiple baseline methods, the method is also effective in reducing a total of 42.05% of hallucinations. Human evaluation results also indicate that the method performs well in reducing content generated by the model.

3.2.2. Autonomous knowledge graph-based retrofitting. This paper proposes a knowledge graph-based autonomous repair framework (KGR) that automatically alleviates factual illusions through a verification chain. It primarily addresses the illusions caused by the fact that the models were internally knowledge-deficient or self-contradictory, respectively. Specifically, the large language model first generates initial answers for the questions, and then initiates a verification chain. The verification chain first extracts the main entities from the initial answers, retrieves factual statements related to these entities from the knowledge graph, and verifies the correctness of the initial answers step by step. Finally, the large language model accepts both the initial answers and the verification results as inputs for further correction to reduce the illusions and erroneous content in the initial answers. Overall, this method effectively mitigates factual illusions and demonstrates superior performance in complex reasoning. However, it is necessary to improve the accuracy of entity detection and factual selection in these two components. Additionally, our experiment demonstrates the importance of multi-round review in ensuring the correctness of the generated answers to match the factual knowledge stored in the knowledge graph.

3.2.3. Validating low-confidence generation. Through experimentation, researchers found that the logical output values of the model are strongly correlated with its uncertainty [16]. As the uncertainty increases, the likelihood of hallucinations also increases. Therefore, they proposed a method to screen for hallucination content using the logical output values of the model. This method primarily addresses hallucinations induced by model inconsistencies, insufficient model knowledge understanding, and insufficient model internal knowledge. The verification process in this method focuses on the important concepts within the sentence rather than the entire sentence, verifying the correctness of the concepts by retrieving related knowledge. The large language model is responsible for modifying and replacing hallucination content with correct information, while re-inputting the relevant knowledge as context, to prevent the same hallucinations from recurring. The experiment demonstrated that this detection technology has an approximately 88% recall rate, and the mitigation technique successfully mitigated 57.6% of the hallucinations correctly detected. At the same time, it does not introduce new hallucinations in cases of false detection.

3.2.4. Reasoning on graphs (RoG). This method employs the structured information characteristics of knowledge graphs, adopting a planning-retrieval-reasoning framework to retrieve factual inference paths for LLM reasoning. This assists in improving the accuracy of LLM answers. It primarily addresses the hallucinations caused by model inconsistency and lack of model knowledge understanding. Specifically, it first generates relationship inference plans based on KG, then uses LLM to understand these plans, retrieves effective inference paths from the KG, and finally inputs these inference paths into LLM for factual reasoning. During this process, researchers aid LLM in correctly understanding and generating inference paths, and reasoning based on them. This method has been extensively tested on two benchmark KGQA datasets, indicating superior performance on KG reasoning tasks and generating faithful and interpretable inference results. The great advantage of this method lies in its ability to seamlessly integrate with any LLM.

DOI: 10.54254/2755-2721/76/20240608

3.3. Knowledge supplementation

As discussed above, model knowledge completeness is a key factor influencing the quality of generated content by models. Models lacking necessary knowledge often fail to generate correct content. A survey indicates that in some large language models, a significant lack of knowledge completeness exists. Supplementing model knowledge has been proven to be an effective method to reduce the occurrence of hallucinations.

Knowledge Consistent Alignment proposes a method to validate and compensate for the gap between external world knowledge and internal knowledge within the model. It is only applicable to cases of hallucination caused by inadequate internal knowledge. It was experimentally proven that the percentage of knowledge inconsistency during alignment is positively correlated with the rate of hallucination. Therefore, this method attempts to reduce the impact by employing two LLMs, with one of them responsible for dividing the training dataset into two categories: those requiring external knowledge and those not requiring external knowledge, and the other generating new knowledge for the data requiring external knowledge. Subsequently, the trained LLM learns this knowledge and completes the validation to ensure the integrity of its knowledge acquisition. The method has been experimentally validated on different base LLMs and datasets. The results demonstrate its superiority in reducing knowledge inconsistency.

3.4. Zero-resource and self-feedback methods

The following methods primarily address the hallucinations produced by the model to maintain consistency, and their significant advantage is that they do not rely on any external resources, only requiring the use of the large language model itself to screen for hallucination in the output text.

3.4.1. SelfCheckGPT. This method eliminates illusions by sampling multiple answers generated from the same prompt by a large language model. It can rank qualitative information consistency and factual correctness of the large language model's responses using five methods: BERTScore, QA, n-grams, NLI, and LLM prompts. This enables the identification of sentences containing illusions. Its advantage is that it can effectively remove sentences containing illusions without the aid of external knowledge, even in a zero-resource environment. However, it cannot guarantee the elimination of illusions within the sentence.

3.4.2. Chain of natural language inference. This is a method that utilizes proxies to modify the original output of the model in a secondary manner. It proposes a two-stage framework that includes a detection proxy and a mitigation proxy. The primary focus is on the pseudo-hallucinations that arise due to the model's attempts to maintain its consistency. Specifically, the detection proxy extracts the model's original output results at the sentence level to generate hypotheses, which are then subject to hierarchical detection via natural language reasoning problem chains. The mitigation proxy then modifies the original output results by accepting the feedback from the detection proxy. Through this approach, the framework effectively detects and mitigates pseudo-hallucinations in large language models, while preserving the original response to the greatest extent possible. Overall, the method confirms that sentence detection and entity detection can effectively detect pseudo-hallucinations generated by large language models, enhancing the performance of pseudo-hallucination detection. For the mitigation of pseudo-hallucinations, the experimental results are equally impressive, achieving progress on multiple NLG evaluation metrics and in-reality metrics.

3.4.3. Chain-of-Verification. The basic idea of this method is to allow the model to automatically validate the contents of the initial generated text. This is aimed at reducing the occurrence of hallucinations. It primarily targets hallucinations caused by the model's attempt to maintain its internal consistency. In particular, it can be divided into four phases: Generate Baseline Response, Plan Verifications, Execute Verifications, and Generate Final Verified Response. This method reduces the interference caused by the model's attempts to maintain internal consistency on the output content by

DOI: 10.54254/2755-2721/76/20240608

stepwise execution and verification. The model first generates an initial answer based on user queries. It then generates a series of related verification questions for this initial answer. Subsequently, the model answers these verification questions individually and determines the consistency of the answers with the initial answer. This confirms whether there are hallucinations. Finally, the model verifies the results and modifies the initial answer to remove the hallucinated content. To avoid the model from generating repetitive answers due to accepting the initial generated text, researchers have proposed some improvement methods. For instance, in the third step, only the questions are inputted instead of the entire context. Based on this, the verification questions can be independently answered by the model, acting as prompt to generate more comprehensive answers. The experimental results indicate that this method can effectively alleviate hallucinations in large language models. It demonstrates superiority in the Wiki data test and performs well in other complex reasoning or long text question-answering tasks.

3.4.4. Self-refine. This method enhances the quality of LLM output text through repeated iterations of self-refinement [17]. Specifically, it uses few-shot prompts to guide the LLM to generate modified feedback based on the original text content of the output. These modified feedback and the original text content are then used as inputs for the LLM to generate new output text. This process is repeated until the model determines that self-refinement can be stopped or the required refinement count is reached. Through evaluation on 7 different tasks, outputs generated by SELF-REFINE outperform outputs generated using the same LLM in terms of human metrics and other automatic metrics. The average task performance of SELF-REFINE-generated output is improved by approximately 20%. Another advantage of this method is its ability to reduce hallucinations and improve the quality of the output text produced by the LLM.

3.4.5. Self-contradictory. This method employs large language models to detect and mitigate loyalty illusions [18]. Its purpose is to trigger, detect, and mitigate the contradictory parts of LLM output content. Its process is suitable for the black box nature of LLMs and does not require any external knowledge assistance. Specifically, the method first iteratively generates contradictory sentences and then guides the model to analyze the problem through zero-shot and cot approaches. At each iteration, the predicted contradictions are removed, ultimately achieving the goal of reducing self-contradiction. One drawback of this method is that its effectiveness depends on the large language model itself. For example, ChatGPT and GPT-4 can accurately identify contradictions, while Vicuna-13B performs poorly in this regard.

3.5. Methods for improving decoding strategies

Large language models generate text by progressively predicting tokens until they reach a certain termination condition. In this process, the decoding strategy relies on different algorithms to approximately solve the next best token. The method below focuses on using different decoding strategies to make the token selection more realistic and contextual, thereby reducing the impact of hallucinations.

3.5.1. Knowledge-constrained tree search decoding. This method alleviates factual hallucination by introducing knowledge constraints during the decoding process. Specifically, it proposes a new token-level hallucination detection method, which uses Monte Carlo tree search algorithm to select tokens during the decoding process. Simultaneously, a binary classifier is used to identify knowledge coherence points in the generated sequence to detect hallucination tokens. This method changes the way the model generates and selects tokens, thereby reducing the occurrence of factual hallucination. The experimental results demonstrate that this method performs well in tasks such as knowledge-based dialogue and abstract summarization, maintaining the generalization capability of the model while improving the factual accuracy of the generated text.

DOI: 10.54254/2755-2721/76/20240608

3.5.2. Inference-time intervention. This method primarily utilizes a supervised learning approach to identify potential vectors related to fact outputs, and activates these vectors when the model correctly infers. Specifically, the method first identifies a set of sparse attention heads, which have high linear detection accuracy for truthfulness. Then, during the inference process, the model moves these attention heads along the truth-related directions, where the method employs autoregressive repetition to generate the same intervention until the answer is completed. This method helps the model utilize knowledge better under the circumstances of understanding knowledge. This method was evaluated on the TruthfulQA benchmark, and its results demonstrated that the method can significantly enhance the model's truthfulness while maintaining a low computational cost.

3.5.3. Decoding by contrasting layers. This method reduces hallucinations by comparing the differences between the layers. Research suggests that the output probability of the next word in the output text is obtained from the difference between the logits obtained from the higher layer and those from the lower layer. Therefore, by emphasizing knowledge from the higher layer and downplaying knowledge from the lower or intermediate layer, we can make the LM more factual, thereby reducing hallucinations. Based on its experiments, DoLa-generated content contains more information and is more factual. Another advantage is the smaller extra delay, suggesting that DoLa has good performance.

4. Evaluation

As discussed in the previous section, most current research primarily focuses on how to mitigate the hallucinations present in large language models after their pre-training. This phenomenon may be associated with factors such as the model not being open-source and the high cost of training. It cannot be denied that most methods have exhibited impressive performance on related evaluation datasets. However, these methods have yet to completely, completely eradicate the inherent defects in hallucinations, which are present in large language models. Some researchers have consequently shifted their focus to exploring the potential boundaries of large language models, leading to pessimistic conclusions. Their studies show that the method of predicting the next best token based on autoregressive assumption has serious flaws [19,20], and the purported emergence capabilities may only be specific performances in Specific nonlinear, discontinuous data sets and not an indication of the superiority of large language models [21]. Despite some debates, these studies emphasize that hallucinations may be a problem that cannot be entirely resolved in the field of large language models. In future research, how to fundamentally alter the inherent defects in large language models and enhance their intelligence levels will become a major point. Currently, the main improvement routes for large language models include monolithic multimodal large models, such as chat-gpt4, and modular large models that interact with other technologies. Similarly, these studies also remind us that existing evaluation benchmarks for large language models may need further improvement to ensure their ability to comprehensively and fairly evaluate the different capabilities of large language models.

5. Conclusion

In summary, this paper systematically explores the methods employed by researchers in addressing hallucinations in large language models. It also discusses some inherent flaws and issues in large language models. Hallucinations have always been a challenging issue that hampers the performance of large language models (LLMs). Previous studies have reflected that addressing hallucinations from a single dimension may not completely solve the hallucinatory contents in LLMs. Future research on solving hallucinations in LLMs may need to comprehensively focus on the multiple flaws inherent in LLMs. By approaching the issue from various perspectives, a more comprehensive LLM can be built. Additionally, more comprehensive evaluation standards for hallucinations are required. It is foreseeable that as LLMs continue to evolve, the demand for models that are factually complete, logically consistent, and knowledge-rich will keep rising.

DOI: 10.54254/2755-2721/76/20240608

References

[1] Wei, Jason, et al. "Emergent abilities of large language models." *arXiv preprint arXiv:2206.07682* (2022).

[2] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," CoRR, vol. abs/2001.08361, 2020.

[3] Stringhi, Elisabetta. "Hallucinating (or poorly fed) LLMs? The problem of data accuracy." i-lex 16.2 (2023): 54-63.

[4] Lee, GaYoung, et al. "A Survey on Data Cleaning Methods for Improved Machine Learning Model Performance." arXiv: Databases,arXiv: Databases, Sept. 2021.

[5] Delétang, Grégoire, et al. "Language modeling is compression." *arXiv preprint arXiv:2309.10668* (2023).

[6] Rashkin, Hannah, et al. "Increasing Faithfulness in Knowledge-Grounded Dialogue with Controllable Features." Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021,

[7] Lee, Nayeon, et al. "Factuality enhanced language models for open-ended text generation." *Advances in Neural Information Processing Systems* 35 (2022): 34586-34599.

[8] Wang, Chaojun, and Rico Sennrich. "On exposure bias, hallucination and domain shift in neural machine translation." *arXiv preprint arXiv:2005.03642* (2020).

[9] Li, Zichao, et al. "Evaluating Dependencies in Fact Editing for Language Models: Specificity and Implication Awareness." *The 2023 Conference on Empirical Methods in Natural Language Processing*. 2023.

[10] Sun, Kai, et al. "Head-to-tail: How knowledgeable are large language models (llm)? AKA will llms replace knowledge graphs?." *arXiv preprint arXiv:2308.10168* (2023).

[11] Rawte, Vipula, et al. "The Troubling Emergence of Hallucination in Large Language Models--An Extensive Definition, Quantification, and Prescriptive Remediations." arXiv preprint arXiv:2310.04988 (2023).

[12] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. ACM Computer. Survey., 55(12):248:1–248:38.

[13] Li, Junyi, et al. "The Dawn After the Dark: An Empirical Study on Factuality Hallucination in Large Language Models." *arXiv preprint arXiv:2401.03205* (2024).

[14] Casper, Stephen, et al. "Open problems and fundamental limitations of reinforcement learning from human feedback." *arXiv preprint arXiv:2307.15217* (2023).

[15] Singhal, Prasann, et al. "A long way to go: Investigating length correlations in rlhf." *arXiv preprint arXiv:2310.03716* (2023).

[16] Varshney, Neeraj, et al. "A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation." *arXiv preprint arXiv:2307.03987* (2023).

[17] Madaan, Aman, et al. SELF-REFINE: ITERATIVE REFINEMENT WITH SELF-FEEDBACK.

[18] Mündler, Niels, et al. Self-Contradictory Hallucinations of Large Language Models: Evaluation, Detection and Mitigation.

[19] Valmeekam, Karthik, Matthew Marquez, and Subbarao Kambhampati. "Can Large Language Models Really Improve by Self-critiquing Their Own Plans?." *arXiv preprint arXiv:2310.08118* (2023).

[20] Stechly, Kaya, Matthew Marquez, and Subbarao Kambhampati. "GPT-4 Doesn't Know It's Wrong: An Analysis of Iterative Prompting for Reasoning Problems." *arXiv preprint arXiv:2310.12397* (2023).

[21] Schaeffer, Rylan, Brando Miranda, and Sanmi Koyejo. "Are emergent abilities of large language models a mirage?." *Advances in Neural Information Processing Systems* 36 (2024).