China Steel Technical Report, No. 37, pp.90－96, (2024)

Utilizing Retrieval-Augmented Generation-Based LLM for Developing Maintenance Work Instructions for NTM Equipment

CHAO-YUNG HSU, I-CHUN HUNG and FU-SHENG TSAI

Green Energy & System Integration Research & Development Department
China Steel Corporation

This paper explores the use of a Retrieval-Augmented Generation (RAG) based large language model (LLM) for developing a system that generates maintenance guidelines. The primary goal is to enhance the understanding and prediction of complex system behaviors by analyzing and summarizing key insights from historical maintenance data. Addressing the hallucination issues commonly encountered in large language models, this study proposes a novel approach. By leveraging the relationship between the questions, retrieved maintenance data, and the responses from large language models, it aims to establish a trustworthy generative AI technique. This method not only improves maintenance efficiency and predicts potential equipment failures, thereby reducing operational risks and costs, but also aids in the rapid training and practical application for new maintenance personnel. The proposed method is particularly significant in the context of the rapid retirement of experienced staff and the need for swift knowledge transfer to new employees.

Keywords: Smart manufacturing, Retrieval-Augmented Generation (RAG), Large Language Model (LLM), Maintenance guidelines, Hallucination, Intelligent application

1. INTRODUCTION

In the contemporary era characterized by rapid globalization and swift advancements in artificial intelligence (AI) technologies, enterprises are encountering unprecedented challenges and opportunities. China Steel Corporation (CSC), as a leading upstream manufacturer in Taiwan's steel industry, sees its equipment maintenance efficiency and effectiveness directly impacting productivity and sustainable development capabilities. With the ongoing advancements in big data and AI technologies, utilizing these advanced techniques to enhance the intelligence and automation of equipment maintenance has become an inevitable trend within the industry. Hence, we utilize generative AI technology to extract key information from CSC's rich equipment maintenance data, assisting new maintenance personnel in quickly learning and assimilating past experiences to enhance overall maintenance efficiency and quality.

In the process of smart manufacturing, data, and technological innovation have emerged as crucial drivers for enterprise development. Especially in the steel manufacturing sector, effectively leveraging artificial intelligence technologies to improve production efficiency and quality presents a significant challenge. Generative AI technology, as an emerging trend, offers an efficient method for processing and analyzing large data sets. This paper aims to explore how generative AI technology can be applied to CSC's equipment maintenance work to achieve data-driven decision-making and process optimization.

Generative AI not only can analyze existing data but also quickly understand and summarize important information from the data[1, 2], which is critical for understanding and predicting behaviors in complex systems. Particularly in the realm of equipment maintenance, this technology can extract key insights from past maintenance records, providing invaluable guidance for the training and work practices of new maintenance personnel. Through the application of generative AI, CSC can not only enhance maintenance efficiency but also predict and prevent potential equipment failures, thus reducing operational risks and costs.

However, despite the numerous advantages of generative AI technology in the aforementioned areas, the hallucination problem of large language models (LLMs) remains a significant concern for all researchers. AI hallucination refers to the issue where AI confidently generates seemingly convincing but incorrect answers, even though these answers are unreasonable within its training data. Most researchers believe the hallucination problem stems from the inherent design of LLMs used to operate chatbots[3].

Yilun Du, a researcher at MIT and former Open AI researcher pointed out that language models are designed to predict the next word, which does not include training to admit ignorance. The result is a robot with a pleasing personality, preferring to fabricate answers rather than admitting its lack of knowledge. Sundar Pichai, CEO of Alphabet, has publicly warned about the severity of the AI hallucination problem. Sam Altman, CEO of OpenAI, also highlighted in congressional testimony that AI could "cause significant harm to the world" through the spread of misinformation and manipulation of human emotions. On May 30, Altman and hundreds of AI researchers, including top executives from companies like Google and Microsoft, signed a statement indicating the risks AI poses to humanity are comparable to pandemics and nuclear war. Thus, preventing or mitigating AI hallucinations has become an urgent issue for tech industry employees, researchers, and AI skeptics alike.

This paper proposes a question-answering system based on retrieval-enhanced generative AI technology to mitigate the hallucination problem. The successful implementation of this technology is expected to bring significant long-term benefits to CSC, including improved efficiency and quality of maintenance work, reduced operational costs, enhanced risk management capabilities, and increased overall competitiveness. Moreover, through the application of generative AI, CSC will be able to establish a more intelligent and automated maintenance system, laying a solid foundation for future innovation and development.

2. RELATED WORK

Hallucination is a widespread issue in language model outputs across various tasks, including summarization and open-domain dialogue, and persists despite increased model sizes or training data volumes[3-5]. For an extensive review of this challenge, Ji et al.[6] offer a comprehensive survey. Strategies to mitigate hallucinations largely fall into three categories: adjustments during model training, corrections at the time of generation, and enhancement through external tools.

During training, methods aim to refine the initial output of language models, whether encoder-decoder or decoder-only, to reduce hallucinated content. This involves modifying the model's weights through techniques like reinforcement learning[7, 8], and contrastive learning (Chern et al., 2023; Sun et al., 2023), among others (Li et al., 2023).

At generation time, efforts focus on enhancing the reliability of the model's output. Approaches include analyzing token generation probabilities[9], detecting inconsistencies through multiple model outputs[10], and verifying low confidence scores[11]. Cohen et al. (2023) propose a unique "LM vs LM" method, where one model checks another's output for consistency, demonstrating its effectiveness, especially in QA tasks over using confidence scores.

External tool augmentation represents a third mitigation pathway. Retrieval-augmented generation reduces hallucination by grounding outputs in factual documents [12] or through chain-of-thought verification[13]. Fact-checking tools[14] and document linking for attribution [15] are also employed to ensure accuracy.

A recent advancement, the Chain-of-Verification (CoVe) method, specifically addresses hallucinations by enabling LLMs to deliberate and self-correct their outputs through a structured process of drafting, verifying, and refining responses[16]. CoVe's approach of generating a baseline response, planning and executing verification queries, and producing a final verified output provides a comprehensive framework for enhancing factual accuracy. This method has shown effectiveness across various tasks, underscoring its potential applicability in systems that generate maintenance guidelines by leveraging historical maintenance data.

In the context of developing systems for generating maintenance guidelines, these existing strategies offer valuable insights. Our work builds on these foundations, particularly drawing from the CoVe methodology to propose a RAG-based system that not only seeks to reduce hallucinations but also enhances predictive maintenance capabilities by analyzing historical data with an eye toward efficiency, reliability, and knowledge transfer in industrial settings.

In addressing the potential limitations of the Chain-of-Verification (CoVe) method, it's important to highlight that the efficacy of CoVe significantly depends on the quality of the verification questions generated. If these questions fail to adequately cover or detect errors or information in the initial response, the effectiveness of the entire method could be compromised. This highlights a critical challenge: ensuring that the generated verification questions are both comprehensive and precise enough to effectively filter and correct hallucinated content.

To tackle this challenge, we propose a Retrieval-Augmented Generation (RAG)-based LLM method. This approach aims to combine the self-checking and self-correcting mechanism of CoVe with the integration of external knowledge bases as a foundation for information retrieval and verification, thereby enhancing the model's accuracy and reliability. Specifically, our RAG-based LLM method relies not only on internally generated verification questions but also retrieves external data sources to provide a more robust factual verification foundation. Thus, even when the quality of verification questions is insufficient to cover all potential errors or hallucinations, the external retrieval results can compensate for this deficiency, enhancing the model's control over factual accuracy.

By integrating CoVe's introspection and self-correction strategies with RAG's external knowledge retrieval capabilities, our method aims to overcome the limitations of relying solely on internally generated verification questions, thereby offering a more comprehensive and reliable framework for reducing hallucinations, especially in the development of systems for generating maintenance guidelines. This will help improve the predictive maintenance capabilities of such systems, enhancing efficiency and reliability through the analysis of historical data, while facilitating knowledge transfer in industrial settings.

3. PROPOSED METHOD

To address the hallucination issue, selecting a suitable LLM is imperative. Our testing of various language models led to the following conclusions: 1. Compared to smaller-scale open-source models like LLAMA2, ChatGPT currently holds a significant lead in terms of model availability and the comprehensiveness of multilingual training; 2. Regarding information security issues, the most common scenario across various applications of LLMs involves users inadvertently revealing sensitive data during question-answering interactions; 3. LLMs, due to their design principle of generating text continuations, will still produce an answer when faced with insufficient information, leading to the phenomenon of language model hallucination. This issue is prevalent in all language models based on the Transformer architecture. Therefore, based on our observations and the primary challenges identified, this paper proposes a proprietary RAG-based LLM architecture developed by China Steel Corporation, employing custom-developed algorithms to tackle information security and language model hallucination issues.

Retrieval-Augmented Generation (RAG) is a natural language processing architecture that combines search retrieval and generation capabilities. Through this framework, the model can search for relevant information from an external knowledge base and then use this information to generate responses or complete specific NLP tasks.

As illustrated in Figure 1, the RAG-based LLM primarily consists of two components: the retriever (Embedding model) and the generator (LLM). The retriever is responsible for retrieving relevant knowledge information from an external knowledge base (e.g., text databases or pre-trained knowledge embeddings). This retrieved knowledge is then fed along with the question to the generator for processing. The generator reorganizes the retrieved knowledge based on the question to generate a response. Typically, the retriever and generator learn how to work collaboratively through a joint training process to produce output content that meets the objective.

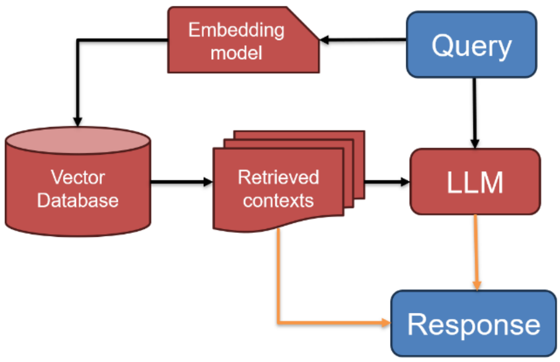

Fig.1. RAG-based LLM architecture.

To simplify the concept, one might recall the experience of taking difficult exams during university days. Professors would allow students to bring a sheet of A4 paper, onto which they had condensed their notes, into the examination room. This process of condensing and organizing the content onto a "cheat sheet" mirrors the Retrieval-Augmented Generation (RAG) process, where the A4 paper serves as the Embedding Vector Database repository. Carrying this into an exam, one can derive the best possible answers using the content processed through the RAG.

The RAG framework is suited for question-answering systems, intelligent dialogue systems, and other natural language processing applications that require external knowledge for support. For example, within a question-answering system, RAG can assist the model in finding answers relevant to the question from a vast amount of textual data, generating structured responses. Intelligent dialogue systems can help produce responses that are more informative and useful. Additionally, since RAG can retrieve information from external knowledge sources, it is also applicable in scenarios where the model needs to have learning capabilities or update its knowledge base.

However, although the Embedding model provides retrieved contexts to the LLM from a database, it cannot completely prevent the occurrence of hallucinations; moreover, the issue of information security arising during the question-answering process remains unresolved within this architecture. As depicted in Figure 2, China Steel Corporation proposes a RAG-based service architecture, maintaining the core RAG structure and utilizing the OpenAI API offered on Microsoft Azure for the LLM's inferencing capabilities. Unlike typical RAG-based service architectures, the proposed structure includes additional modules for security verification and response credibility detection.

When a user poses a question, the system evaluates the query within the question-answering interface. The primary goal of this evaluation is to ascertain whether the user has inadvertently disclosed sensitive corporate

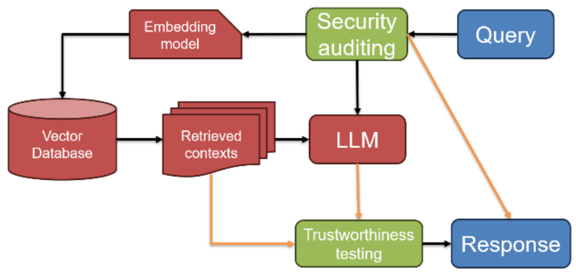

Fig.2. CSC proposed RAG-based LLM architecture.

information while formulating their query. For instance, an incident occurred with a Samsung engineer in South Korea, who uploaded crucial company source code to ChatGPT for debugging, thus exposing confidential corporate information. To prevent such unintentional leakage of confidential information, we analyze and log the questions posed by users. Through a local model, we establish a Robotic Process Automation (RPA) system capable of understanding and detecting security issues, and filtering out queries with potential security concerns.

Once security concerns are alleviated, the next step is to proceed to the retrieval phase. The retrieval system searches for K-relevant text passages based on the current input and the pre-existing database. These text passages, combined with the user's question, form the prompt for this round of inquiry, which is then converted into tokens through an API for querying the LLM.

Upon receiving a response from the LLM, the final step involves verifying the credibility of its reply. At this stage, we propose the development of a Large Language Model Trustworthiness Estimation Algorithm. However, evaluating the responses from an LLM is not straightforward, given that the responses often contain a mixture of accurate and inaccurate information. The challenge lies in the lack of extensive reference materials for constructing an algorithm to estimate the trustworthiness of these responses. Therefore, by converting the query, context, and the LLM's response into feature vectors, and then calculating the inner product of these feature vectors to represent their correlation, we can obtain three distinct types of relevancy:

1. Contextual Relevance(I_CR): Ensuring that each retrieved context block is relevant to the input query. Irrelevant information may be woven into incorrect answers.
2. Groundedness (I_G): Verifying whether the answer generated by the LLM is based on retrieved facts. This involves deconstructing the response into individual statements and independently searching for evidence supporting each statement.
3. Answer Relevance(I_AR): Assessing the degree of relevance of the final response to the user's input. The response must contribute to addressing the original question.

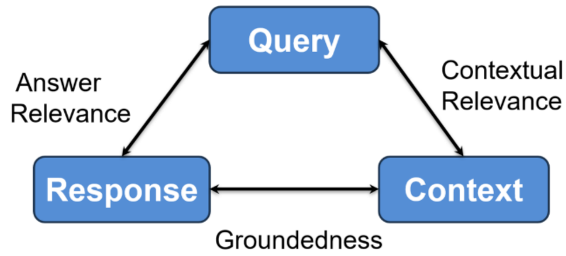

Fig.3. The commonly used criteria for trustworthiness evaluation.

Utilizing these relevance coefficients, we can establish a trustworthiness index:

$$I_T = \alpha I_{CR} + \beta I_G + \gamma I_{AR} \quad \text{....................................(1)}$$

Where α, β, and γ represent different weights that can be adjusted to more appropriate values through machine learning, based on the user feedback information. By employing the trustworthiness index, we can filter out responses with lower credibility, preventing the excessive dissemination of incorrect information. Through these steps and evaluation standards, an accurate, efficient, and hallucination-free Retrieval-Augmented Generation (RAG) system can be developed. Such a system can rely on precise and relevant information when providing answers, thereby minimizing the generation of erroneous or irrelevant content.

4. RESULTS

To substantiate the reliability of the proposed framework, we conducted tests on the algorithm using a database of approximately 500 entries from the equipment maintenance logs of the No-Twist Mill Stand (NTM) collected from China Steel. As illustrated in Table 1, the content of a maintenance record includes documentation of the state of the NTM equipment during an anomaly, the presumed causes of the anomaly identified at that time, the actions taken by maintenance personnel, and the outcomes following those actions.

Given that the native ChatGPT does not encompass maintenance data for NTM rolling mills, it is anticipated that directly querying ChatGPT for maintenance methods regarding equipment malfunctions would yield responses fraught with hallucinations. In this experiment, we formulated 9 questions related to NTM maintenance tasks and compared two types of responses: one from the native ChatGPT and the other from the response under the proposed RAG architecture. The results of the Q&A

were evaluated by five equipment maintenance personnel, with each question rated on a scale from 0 to 10; higher scores indicate a closer alignment of the response with the facts as recognized by the maintenance staff. Table 2 presents the scores for the 9 questions, from which it can be seen that the proposed method achieved higher scores than the native ChatGPT.

As depicted in Table 3, a comparison of answers to one of the questions during the experiment demonstrates that the proposed method can explicitly elucidate the possible causes of an anomaly based on data retrieved from the database, whereas the native GPT provides a

Table 1 An example of a maintenance record.

ITEM	DESCRIPTION
STATUS DESCRIPTION (CURRENT STATE OF THE EQUIPMENT?)	The vibration level was found to be increased at Lx#xx STD, with abnormal sounds also detected on-site. The Fomos spectrum revealed frequencies indicative of an outer ring defect in the ball bearing, leading to the decision to halt rolling for inspection. Upon examining the coupling guard, iron filings were discovered at the roller bearing's oil expulsion point.
CAUSE OF OCCURRENCE (POSSIBLE CAUSE?)	Disassembling the HSG revealed a fracture in the outer ring of the bearing, located at the joint surface of the HSG. Further inspection confirmed the variation in the size of the bearing seat's round hole.
METHOD OF DISPOSITION	After confirmation of the variation in the size of the bearing seat's round hole, the part was sent out for re-machining on the morning of the 2nd of December. It returned to the factory in the evening, and after installation was complete, the operating team installed the rolling components. The system was then oil-flushed for one hour before initiating operational testing, confirming that the vibrations were normal.
RESULT (POST-MAINTENANCE STATUS)	The vibrations and acoustics were confirmed to be normal after the repair.

Table 2 The proposed method and native ChatGPT scores for the 9 questions.

No.	Inquiry	ChatGPT	Proposed Method
1.	As the maintainer and manager of the NTM rolling mill, please provide maintenance and management suggestions for the NTM rolling mill.	39	38
2.	To prevent the occurrence of roll sinking in the NTM rolling mill ROLL HOUSING, which causes CROSS ROLL at the site, please provide a solution.	30	42
3.	Please assist in analyzing the relationship between the temperature of the steel entering the NTM rolling mill at different sites and the occurrence of equipment anomalies.	35	40
4.	Please provide possible causes and solutions for the rise in lubricant temperature in the NTM rolling mill lubrication system.	33	43
5.	Please provide possible causes and solutions for potential damage to the NTM rolling mill ROLL HOUSING (stand).	36	39
6.	Please provide possible causes and solutions for potential damage to the babbitt bearing failure within the NTM rolling mill ROLL HOUSING (stand).	32	44
7.	Please provide safety precautions that should be noted during the maintenance of the NTM rolling mill.	40	41
8.	When rolling, please provide possible causes for rolling failure (COBBLE) in the NTM rolling mill and suggest preventive measures.	37	41
9.	During rolling, please provide possible causes for vibration and abnormal noise in the rolling mill, and suggest preventive measures.	34	38

Table 3 A comparison of answers to the question No. 4

Q: TO PREVENT THE OCCURRENCE OF ROLL SINKING IN THE NTM, WHICH CAUSES CROSS-ROLL AT THE SITE, PLEASE PROVIDE A SOLUTION.	
CHATGPT	Proposed Method
PREVENTING ROLL SINKING IN THE NON-TWIST MILL (NTM) ROLLING MILL'S ROLL HOUSING, WHICH LEADS TO CROSS-ROLLING, INVOLVES SEVERAL TECHNICAL ADJUSTMENTS AND MAINTENANCE STRATEGIES. HERE ARE A FEW SOLUTIONS THAT COULD HELP MITIGATE THIS ISSUE: 1. ROLL MATERIAL AND HARDNESS OPTIMIZATION 2. REGULAR ROLL INSPECTION AND MAINTENANCE 3. ROLL COOLING SYSTEM ENHANCEMENT 4. LOAD DISTRIBUTION OPTIMIZATION 5. ROLL HOUSING RIGIDITY ENHANCEMENT 6. ADVANCED MONITORING SYSTEMS 7.FINITE ELEMENT ANALYSIS (FEA) FOR STRESS ANALYSIS	To prevent roll sinking in the NTM rolling mill, which leads to CROSS ROLL at the site, the document provides related cases and solutions. Specifically, one case mentions the process and result of replacing the Roll Housing. In this case, it was found that during the operation of the mill, issues such as high load and slow rotation speed were observed. After inspecting the NTM (where #16 was found burned and leaking oil), the problem was resolved by replacing the Roll Housing. Based on this case, to prevent ROLL HOUSING from sinking and causing CROSS ROLL, it is recommended to regularly check and maintain the condition of the Roll Housing. If any abnormalities are detected (such as high load, slow rotation speed, or oil leakage), they should be addressed immediately through replacement or repair. Additionally, the wear status of the Roll Housing's fastening parts should be regularly inspected. If broken screws or other signs of instability are found, these issues should also be promptly fixed to ensure the stable operation of the mill and prevent sinking and CROSS ROLL.

more vague and generalized response. This experiment shows that the proposed method can effectively reduce the likelihood of hallucination occurrences by LLMs in the application of equipment maintenance records.

5. CONCLUSION

In conclusion, the RAG-based LLM for reliable equipment maintenance guidance introduced in this study offers a viable and effective solution to the challenges posed by hallucination in LLMs. Its successful implementation at CSC underscores the potential of this technology to revolutionize the field of equipment maintenance, ensuring that maintenance personnel have access to accurate, reliable, and timely guidance. This, in turn, has far-reaching implications for enhancing operational efficiency, reducing maintenance-related costs, and facilitating the rapid transfer of knowledge to new personnel. As we move forward, it is evident that the continued refinement and application of this technology will play a pivotal role in advancing smart manufacturing and maintenance practices across industries, paving the way for a new era of AI-assisted maintenance systems.

REFERENCES

1. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9, 2019.
2. Fabio Petroni, Tim Rockt¨aschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? arXiv preprint arXiv: 1909.01066, 2019.
3. Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. arXiv preprint arXiv: 2005.00661, 2020.
4. Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. Recipes for building an open-domain chatbot. arXiv preprint arXiv:2004.13637, 2020.
5. Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A Smith. How language model hallucinations can snowball. arXiv preprint arXiv: 2305.13534, 2023.
6. Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38, 2023.
7. Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni,

Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, L´eonard Hussenot, Orgad Keller, et al. Factually consistent summarization via reinforcement learning with textual entailment feedback. arXiv preprint arXiv:2306.00186, 2023.

8. Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. arXiv preprint arXiv:2306.01693, 2023.
9. I-Chun Chern, Zhiruo Wang, Sanjan Das, Bhavuk Sharma, Pengfei Liu, Graham Neubig, et al. Improving factuality of abstractive summarization via contrastive reward learning. arXiv preprint arXiv:2307.04507, 2023.
10. Weiwei Sun, Zhengliang Shi, Shen Gao, Pengjie Ren, Maarten de Rijke, and Zhaochun Ren. Contrastive learning reduces hallucination in conversations. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 37, pp. 13618–13626, 2023.
11. Kenneth Li, Oam Patel, Fernanda Vi´ egas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. arXiv preprint arXiv:2306.03341, 2023.
12. Sabrina J Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. Reducing conversational agents' overconfidence through linguistic calibration. Transactions of the Association for Computational Linguistics, 10:857–872, 2022.
13. Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. arXiv preprint arXiv:2303.08896, 2023.
14. Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. arXiv preprint arXiv:2307.03987, 2023.
15. Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. Lm vs lm: Detecting factual errors via cross examination. arXiv preprint arXiv: 2305.13281, 2023.
16. Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. arXiv preprint arXiv:2104.07567, 2021.
Wenhao Yu, Zhihan Zhang, Zhenwen Liang, Meng Jiang, and Ashish Sabharwal. Improving language models via plug-and-play retrieval feedback. arXiv preprint arXiv:2305.14002, 2023.
17. Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. arXiv preprint arXiv:2305.03268, 2023.
18. Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, et al. Check your facts and try again: Improving large language models with external knowledge and automated feedback. arXiv preprint arXiv:2302.12813, 2023.
19. Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2023. Chain-ofVerification Reduces Hallucination in Large Language Models. arXiv preprint arXiv: 2309.11495 2023.