Knowledge-Grounded Detection of Factual Hallucinations in Large Language Models

Cristian Ceccarelli, Alessandro Raganato and Marco Viviani*

University of Milano-Bicocca (DISCo – IKR3 Lab), Edificio U14 (ABACUS), Viale Sarca, 336 – 20126 Milan, Italy

Abstract

Large Language Models (LLMs) have achieved remarkable success in various Natural Language Processing (NLP) tasks, yet they remain prone to generating factually incorrect content, known as hallucinations. In this context, this work focuses on factuality hallucinations, offering a comprehensive review of existing detection methods and an empirical evaluation of their effectiveness. In particular, we investigate the role of external knowledge integration by testing hallucination detection approaches that leverage evidence retrieved from a real-world Web search engine. Our experimental analysis compares this knowledge-enhanced strategy with alternative approaches, including uncertainty-based and black-box methods, across multiple benchmark datasets. The results indicate that, while external knowledge generally improves factuality detection, the quality and precision of the retrieval process critically affect performance. Our findings underscore the importance of grounding LLM outputs in verifiable external sources and point to future directions for improving retrieval-augmented hallucination detection systems.

Keywords

Natural Language Processing (NLP), Large Language Models (LLMs), Hallucinations, Retrieval-Augmented Generation (RAG)

1. Introduction

In recent years, the rapid advancements in technology and the growing availability of data have fostered the emergence of *Large Language Models* (LLMs). These models, based on the Transformer architecture, exploit attention mechanisms to analyze relationships between textual elements and effectively capture contextual meaning [1]. This capability allows LLMs to excel in natural language generation and a wide range of *Natural Language Processing* (NLP) tasks, including text summarization, machine translation, and conversational AI. Due to their impressive ability to understand, interpret, and generate human-like language, LLMs have become indispensable tools in fields such as education, research, and healthcare.

However, despite their capabilities and the significant technological advancements they represent, LLMs still face some challenges. A particularly critical issue is their tendency to generate the so-called *hallucinations*, which are outputs that are plausible but incorrect, under different perspectives [2]. The prevalence of such hallucinated outputs is particularly concerning given the increasing integration of LLMs into sensitive domains. The generation of incorrect content can undermine trust in AI systems, limit their practical applicability, and contribute to the spread of misinformation [3], especially in critical areas such as journalism, medicine, and scientific research, where factual accuracy is paramount. As such, hallucinations represent a major challenge in the deployment of LLMs. Addressing this issue requires a deeper understanding of its underlying causes and the development of robust detection and mitigation strategies to ensure the reliability and safety of these technologies in real-world applications [4].

In this context, we investigate how incorporating *external knowledge* can improve the effectiveness of hallucination detection in LLMs. Specifically, we explore the integration of *Retrieval-Augmented Generation* (RAG) frameworks [5] into existing detection pipelines, with the aim of enhancing their ability to identify hallucinated content by accessing verifiable information. Therefore, in this work, we develop an automated knowledge retrieval system that leverages the Google Search API to collect relevant external evidence, which is then integrated through RAG into two distinct hallucination detection methods: (i) a *few-shot prompting* approach, where an LLM is explicitly instructed to assess the factuality of a given statement, and (ii) SelfCheckGPT [6], a state-of-the-art hallucination detection method based on response sampling, which evaluates whether a generated output contains hallucinated content. Finally, the impact of knowledge integration on the effectiveness of hallucination detection approaches is assessed by conducting a comparative evaluation. Specifically, the performance of each approach is measured both with and without the incorporation of external knowledge, using established benchmark datasets for hallucination detection.

CLiC-it 2025: Eleventh Italian Conference on Computational Linguistics, September 24 — 26, 2025, Cagliari, Italy

*Corresponding author.

cricecca99.cc@gmail.com (C. Ceccarelli);
alessandro.raganato@unimib.it (A. Raganato);
marco.viviani@unimib.it (M. Viviani)
http://www.ir.disco.unimib.it/people/marco-viviani/ (M. Viviani)
0000-0002-7018-7515 (A. Raganato); 0000-0002-2274-9050 (M. Viviani)

2. Background and Related Work

Within the context of LLMs, the term "hallucination" refers to the generation of content that is either nonsensical or unfaithful to the source content. In the literature, hallucinations are typically categorized into two main types: *factuality hallucinations* and *faithfulness hallucinations* [2]. The remainder of the section therefore provides background on the two distinct concepts, before considering the literature that directly addresses the problem.

2.1. Factuality Hallucinations

This category of hallucination encompasses all content that contradicts established real-world knowledge. It constitutes the primary focus of this study, as it is directly associated with the presence and potential dissemination of misinformation. Factuality hallucinations can be further classified based on the verifiability of the generated content against reliable sources, depending on whether they are characterized by:

- *Factual inconsistency*, which refers to cases in which the output contradicts verifiable information from reliable sources, thereby generating incorrect content;
- *Factual fabrication*, which occurs when the generated output cannot be verified against any reliable source, indicating the generation of unverifiable or entirely invented content.

2.2. Faithfulness Hallucinations

Faithfulness hallucinations arise when the generated content is inconsistent with the input or contextual information provided by the user. This category can be further subdivided into three types, depending on whether they are characterized by:

- *Instruction inconsistency*, which occurs when the output deviates from the explicit instructions given by the user;
- *Context inconsistency*, where the generated content is misaligned with the contextual information supplied by the user;
- *Logical inconsistency*, which is typically observed in reasoning tasks and is characterized by contradictions or errors in the reasoning steps of the model.

2.3. Related Work

In recent years, numerous studies have investigated the issue of hallucinations in LLMs, proposing a variety of detection approaches based on different methodological strategies to identify and mitigate this phenomenon. These approaches can be broadly classified into the following categories:

- *Uncertainty estimation-based*: Studies suggest that outputs produced with high model uncertainty are more prone to hallucinations [7]. Accordingly, these methods estimate the LLM's uncertainty by analyzing its internal states to infer the likelihood of hallucinated content. A key advantage of these techniques is their independence from external knowledge; however, they require access to the model's internal representations, which may not be feasible in all settings, especially with proprietary models;
- *Knowledge retrieval-based*: These approaches leverage external knowledge sources—such as online encyclopedias or structured databases—to verify the factuality of LLM-generated content. While generally reliable and adaptable across domains, these methods often incur high computational costs due to the retrieval and processing of external information;
- *Zero-resource and black-box*: These techniques detect hallucinations by analyzing output consistency and model behavior across multiple generations, without relying on external knowledge or internal model access. Although these methods are broadly applicable to any LLM, they may be less effective in scenarios involving queries with multiple plausible answers or ambiguous interpretations.

Belonging to the first category, the work described in [8] argues that when an LLM generates hallucinated content, it implicitly encodes a degree of uncertainty within its internal representations. Based on this assumption, the authors introduce SAPLMA, a method that aims to determine the factuality of a generated statement by analyzing the internal states of the model to estimate its uncertainty. Since it is not yet fully understood which internal layers best capture information relevant to factuality, the authors investigate multiple variants of the approach by extracting hidden states from different layers of the model, such as intermediate or final layers. These representations are then passed to a shallow neural classifier, which outputs the probability that the statement is true or false. Despite the good results, the optimal layer from which to extract internal states remains unclear and appears to be dependent on the specific LLM employed. Furthermore, the evaluation was conducted on isolated statements classified as true or false, rather than on complete model responses generated in relation to specific user inputs, thereby limiting the assessment of the method's effectiveness in realistic interaction scenarios.

The approach presented in [9], which belongs to the second category of approaches, introduces FActScore, a method based on comparison with a reliable external

knowledge source. The procedure begins by decomposing the content generated by the LLM into atomic facts, defined as concise and discrete statements. These atomic facts are then manually verified by human annotators, who assess their factuality using English Wikipedia as the reference source. Each atomic fact is labeled as supported or unsupported depending on whether it is supported by the knowledge base. The overall factuality score of the content is computed as the proportion of atomic facts that are supported by reliable knowledge. While this method offers a structured and interpretable evaluation of factual accuracy, it presents notable limitations. Specifically, it has been validated exclusively in biographical texts, domains characterized by objective and easily verifiable information.

Finally, belonging to the third category of methods, in [6] the authors propose SelfCheckGPT, a hallucination detection method that leverages stochastic sampling of multiple responses generated by an LLM from the same input prompt. The underlying assumption of this approach is that, when an LLM possesses reliable knowledge about a given topic, its responses will exhibit a high degree of consistency; conversely, a lack of knowledge will lead to greater variability among responses. To evaluate the consistency of these sampled outputs, the authors introduce five distinct variants of SelfCheckGPT: SelfCheckGPT with BERTScore, which performs semantic similarity comparisons between responses; SelfCheckGPT with *Question Answering* (QA), which generates questions from the original answer and uses the sampled responses to answer them; SelfCheckGPT with *Natural Language Inference* (NLI), which applies an NLI model to determine whether responses entail or contradict one another; SelfCheckGPT with n-grams, which estimates token-level probabilities; and SelfCheckGPT with LLM prompt, which relies on prompting an LLM to judge the consistency of the sampled outputs. However, the evaluation of this approach was conducted on a limited dataset comprising 238 Wikipedia-style articles synthetically generated by an LLM, with factuality assessed manually at the sentence level. While this setting provides initial insights, the scope of the study remains narrow and could be extended to include more diverse and conceptually complex content.

In light of the primary limitations identified in the literature for existing hallucination detection approaches, this study proposes a fully automated methodology that completely eliminates the need for human involvement in the knowledge retrieval process. Manual retrieval is often labor-intensive and time-consuming; by contrast, the proposed approach leverages an automated pipeline for sourcing and integrating external knowledge, thereby significantly reducing both time and operational costs. Furthermore, the effectiveness of the method is validated through experiments conducted on three established benchmark datasets for hallucination detection, each encompassing a variety of domains. This ensures a broader evaluation scope and demonstrates the robustness of the method across diverse contexts.

3. Methodology

This section details the methodologies employed for the development of the automatic knowledge retrieval system, alongside the strategies utilized for integrating the retrieved knowledge into both: (i) the few-shot prompting approach, and (ii) the SelfCheckGPT framework.

3.1. Knowledge Retrieval System

The knowledge retrieval system is built entirely upon a customized Google Search engine, accessed via the Google Search API. In particular, the retrieval process is organized into the following steps:

- A query is submitted to the search engine;
- The search engine communicates with the Web through the API and returns a list of query-relevant URLs;
- The content of the first URL is parsed to extract the main body text from the HTML;
- The retrieved textual content is then encoded using an embedding model, and its vector representation is stored in a vector database, allowing for efficient retrieval and integration with the LLM.

Figure 1 illustrates the pipeline for the knowledge retrieval process.

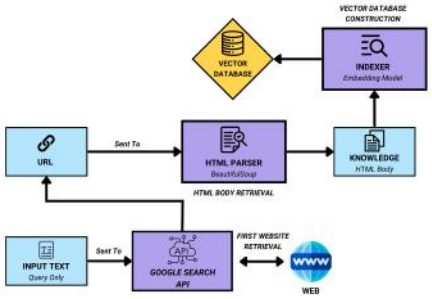

Figure 1: Pipeline of the knowledge retrieval process.

3.2. Few-Shot Prompting with Knowledge

Few-shot prompting is a technique in which an LLM is presented with a limited number of task-specific examples to guide its behavior and enhance its ability to

perform a given task. However, the model's responses in this setting are based solely on the knowledge acquired during the pre-training phase. To enhance its performance and expand its informational basis, the framework integrates external knowledge retrieved through the automated retrieval system. This additional context is provided to the model during inference, enabling more accurate and informed task execution. Specifically, the process is structured into the following steps:

- The user's query is encoded using the embedding model;
- The resulting embedding is used to retrieve relevant information from the vectorized knowledge base;
- The retrieved knowledge is incorporated into the prompt, together with a set of examples and the question–answer pair to be assessed;
- The LLM evaluates the factuality of the answer by leveraging both its internal knowledge and the external information, classifying the response as either factual (*true*) or hallucinated (*false*).

Figure 2 illustrates the pipeline of the few-shot prompting approach enhanced through the integration of specialized external knowledge.

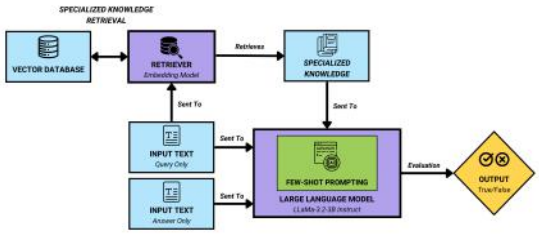

Figure 2: Pipeline of the few-shot prompting approach enhanced with the specialized knowledge.

3.3. SelfCheckGPT with Knowledge

The knowledge was also integrated into the SelfCheckGPT framework to improve the quality of the sampled responses. The underlying assumption is that providing the LLM with relevant external information will lead to the generation of more accurate and reliable responses. As a result, when these samples are compared with the target response using one of the SelfCheckGPT variants, it becomes easier to assess whether the target response is hallucinated. The process is structured according to the following steps:

- The user's query is encoded using the embedding model;
- The resulting embedding is used to retrieve relevant information from the vectorized knowledge base;
- Based on the user's query and the retrieved knowledge, the model is prompted to generate N responses to the same query;
- The response under evaluation is segmented into individual sentences, which are then compared with the N sampled responses using one of the SelfCheckGPT variants;
- SelfCheckGPT assigns a hallucination score to the evaluated response by averaging the sentence-level scores, resulting in a value between 0 and 1, where 0 indicates a hallucinated response and 1 denotes a factual one. This score is subsequently transformed into a binary classification (true/false) using a threshold function.

Figure 3 illustrates the pipeline of the SelfCheckGPT framework enhanced through the integration of external knowledge.

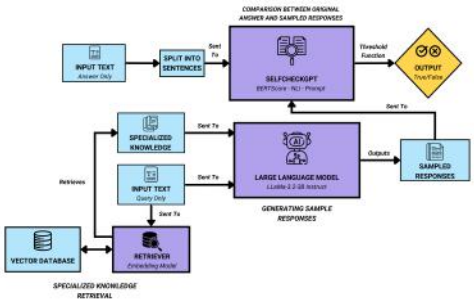

Figure 3: Pipeline of the SelfCheckGPT framework enhanced with the specialized knowledge.

4. Experimental Evaluation

This section presents the experimental setup employed to conduct the experiments, describes the datasets and the metric used for performance evaluation, and provides an analysis of the results obtained.

4.1. Experimental Setup

All experiments were carried out on the Google Colab platform,[1] utilizing a Tesla T4 GPU. The LLM employed for the few-shot prompting approach, response sampling, and the LLM-prompt variant of SelfCheckGPT was

[1] https://colab.research.google.com/

Llama-3.2-3B-Instruct, accessed using the Transformers library of Hugging Face.[2] For both approaches, the model selected for generating semantic embeddings and as a retriever was jina-embeddings-v3.[3] The retrieved knowledge was segmented into chunks of 256 characters with an overlap of 25 characters to preserve semantic coherence across segments. The retriever was configured to return the top 5 most relevant documents according to similarity to the input query.

The few-shot prompting approach was evaluated by providing the model with 1, 5, and 10 examples. To generate the response, the LLM was set to a temperature value equal to 0.001. Figure 4 presents the prompt structure provided to the LLM to classify a given text as either factual or hallucinated.

Prompt for Few-Shot Prompting with Knowledge

I want you to act as a response judge.
Given a user query, a knowledge, and a response by an LLM, your objective is to determine if the response is an hallucination or not.
In the context of NLP, an "hallucination" refers to a phenomenon where the LLM generates text that is incorrect, nonsensical, or not real. Based on your knowledge, on the knowledge provided, and on the definition of hallucination provided, analyze the user query and the response of the LLM, and answer the following question: is the response factual or not?
BE CAREFUL: sometimes the knowledge may be empty or not useful, in which case you have to respond based only on your knowledge.
Answer True if you consider the response factual, False otherwise.
You don't have to provide any explanation.
EXAMPLE 1
User query: [USER QUERY]
Knowledge: [KNOWLEDGE]
LLM response: [LLM RESPONSE]
Answer: [ANSWER]
...
EXAMPLE N
User query: [USER QUERY]
Knowledge: [KNOWLEDGE]
LLM response: [LLM RESPONSE]
Answer: [ANSWER]
LLM TURN
User query: [USER QUERY]
Knowledge: [KNOWLEDGE]
LLM response: [LLM RESPONSE]
Answer:

Figure 4: Prompt submitted to the LLM for few-shot prompting with knowledge.

For the implementation of SelfCheckGPT, the variants employed for evaluation purposes are BERTScore, NLI, and LLM prompt (see Section 2.3). In accordance with the original SelfCheckGPT configuration, 5 responses per query were sampled using a temperature setting of 1.0 and a maximum output length of 128 tokens. Figure 5 illustrates the prompt provided to the LLM for the generation of these sampled responses.

Prompt for Generating Sampled Responses with Knowledge

Based on your knowledge and on the context provided, answer the following question giving as much detail as you can.
Question: [QUESTION]
Context: [KNOWLEDGE]
Answer:

Figure 5: Prompt submitted to the LLM for generating the sampled response using the retrieved knowledge.

4.2. Datasets and Evaluation Metric

For the experimental evaluation, three benchmark datasets for hallucination detection were selected. Each dataset includes a *user query*, the corresponding LLM-generated *response*, and a *binary label* indicating whether the response is factually accurate. The datasets employed are *FactAlign* [10], *FactBench* [11], and FELM [12], all of which are described in detail in the following.

FactAlign. This dataset was created to improve the factual accuracy of LLM-generated long-form responses [10]. For each query, a corresponding answer was generated and then segmented into individual sentences. Each sentence was further broken down into atomic facts, which were verified against a Wikipedia-based reference corpus. A sentence was considered factual only if all its atomic facts were supported by this reference. An answer received a factual label if at least 75% of its sentences met this criterion, yielding a binary label (i.e., *true* or *false*). The version used in this work, retrieved from Hugging Face, contains a total of 2 562 instances.[4] Of these, 1 307 were labeled as factual (true) and 1 255 as non-factual (false). Each instance includes a user prompt, the corresponding response generated by an LLM, and a binary factuality label indicating the truthfulness of the response. For evaluation, only those instances where the user query was a question—i.e., ending with a question mark—were selected. This filtering criterion was adopted

[2] https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct
[3] https://huggingface.co/jinaai/jina-embeddings-v3
[4] https://huggingface.co/datasets/chaoweihuang/factalign-gemma2-f1_0.75

to facilitate more effective knowledge retrieval through the Google Search API and to simplify both the factuality classification task performed by the LLM and the generation of sampled responses within the SelfCheck-GPT framework. Following this filtering step, a random sample of 100 questions was selected. This limitation was imposed by constraints on computational resources and time, which required a balance between the number of examples and processing efficiency. Furthermore, to ensure comparability and consistency across the methods and each variant, a fixed random seed was used to guarantee the reproducibility of the 100 instances across all experiments.

FactBench. This dataset was specifically developed to evaluate FactCheck-GPT, a multi-step framework designed for the detection and correction of factual errors in responses generated by LLMs [11]. FactBench was constructed by integrating three distinct benchmark datasets aimed at hallucination detection:

- *Knowledge-based FacTool*: Created to assess the performance of the FacTool framework, which evaluates the factual consistency of LLM-generated responses through external knowledge retrieval [13]. This dataset was constructed by selecting 50 prompts from FactPrompts and fact-checking datasets such as TruthfulQA [14]. For each prompt, responses were generated using ChatGPT and subsequently annotated by human evaluators with binary labels indicating factual correctness;
- *FELM-WK*: Subset of the FELM dataset that will be detailed in the next paragraph;
- *HaluEval*: This benchmark dataset for hallucination detection was constructed by initially considering 52 000 prompts, followed by a filtering procedure aimed at selecting those most likely to elicit hallucinated responses from a LLM. Specifically, each prompt was submitted to ChatGPT three times, and the average semantic similarity among the generated responses was calculated. The 5 000 prompts with the lowest semantic similarity scores were retained to ensure the dataset included only the most challenging queries. The selected prompts were then resubmitted to ChatGPT to obtain a second set of responses, which were manually annotated as either true or false based on their factual accuracy [15].

FactBench was made publicly available by the authors on GitHub and comprises a total of 4 835 examples, of which 3 838 are labeled as true and 995 as false.[5] Each instance includes a user query, the corresponding response generated by an LLM, and a binary factuality label. For evaluation purposes, only the entries corresponding to user queries in the form of questions were retained. Due to computational constraints, a subset of 100 observations was selected. To mitigate the effects of class imbalance, an equal number of true and false instances (50 each) were randomly sampled. A fixed random seed was applied to ensure reproducibility and consistency across all experimental configurations.

FELM. FELM is a multi-domain benchmark dataset designed for the evaluation of hallucination detection in LLMs, encompassing five distinct domains, each posing specific challenges for the models under analysis [12]. The domains are defined as follows:

- *World knowledge*: Includes questions related to general cultural and factual knowledge;
- *Science and technology*: Comprises statements related to scientific facts or citations across disciplines such as physics and biology;
- *Reasoning*: Contains prompts that require multi-step logical reasoning to produce a correct response;
- *Recommendation and writing*: Involves open questions requiring the model to provide suggestions or generate creative or structured written content;
- *Math*: Encompasses problems that necessitate both logical reasoning and mathematical skills to arrive at correct answers.

FELM was constructed by aggregating prompts from diverse sources, which were then submitted to ChatGPT operating in a zero-shot configuration. The resulting responses were segmented into sentences, each of which was subsequently evaluated by a team of experts. The factual accuracy of each sentence was assessed based on comparison with reliable sources, and sentences were annotated as either true or false accordingly. A response was labeled as true only if all its sentences were assessed as accurate; otherwise, it was classified as false. The FELM dataset was obtained from Hugging Face and comprises a total of 847 instances.[6] Each instance includes a user prompt, the corresponding response generated by the LLM, and a factuality label. Of these examples, 566 are labeled as factual, while 281 are labeled as non-factual. For evaluation, only the *World knowledge* and *Science and technology* domains were considered, as the remaining presented substantial limitations for the knowledge retrieval approach (*e.g.*, mathematical prompts such as *"What is the value of the expression 1! + 2! + 3! + ... +*

[5] https://github.com/yuxiaw/Factcheck-GPT/blob/main/Factbench.jsonl

[6] https://huggingface.co/datasets/hkust-nlp/felm

10!"). As in the previous datasets, only prompts formulated as questions were retained. To mitigate class imbalance and accommodate computational constraints, a balanced subset of 100 samples—comprising 50 factual and 50 non-factual instances—was randomly selected. A fixed random seed was applied to ensure consistency across experiments.

Evaluation metric. Since all the datasets employed in the evaluation are balanced, *Accuracy* was adopted as the primary performance metric. It is defined as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

where *TP* denotes factual responses correctly classified as factual, *TN* represents hallucinated responses correctly identified as hallucinations, *FP* corresponds to hallucinated responses incorrectly classified as factual, and *FN* refers to factual responses mistakenly classified as hallucinations.

4.3. Results and Discussion

To evaluate the impact of knowledge integration, the performance of both SelfCheckGPT and the few-shot prompting approach was evaluated in two configurations: with and without the inclusion of external knowledge. A summary of the comparative results is presented in Table 1. The notation W/O and W denotes whether the evaluated variant operates without or with integrated knowledge, respectively. For each variant and dataset, the version (with or without knowledge) that achieves the highest performance is underlined; if both versions perform equally, no underlining is applied.

Models	Variant	FactAlign		FactBench		FELM	
		W/O	W	W/O	W	W/O	W
SelfCheckGPT	*BERTScore*	59.0	61.0	61.0	60.0	56.0	59.0
	NLI	67.0	67.0	64.0	69.0	67.0	71.0
	LLM Prompt	62.0	65.0	57.0	63.0	69.0	68.0
Few-Shot Prompting	*One-shot*	50.0	54.0	62.0	62.0	62.0	63.0
	Five-shot	57.0	55.0	53.0	64.0	56.0	59.0
	Ten-shot	55.0	59.0	59.0	65.0	59.0	62.0

Table 1
Comparison between methods with and without integrated knowledge, to evaluate its impact on their performance.

As shown in Table 1, the SelfCheckGPT framework consistently outperforms the few-shot prompting approach across all evaluated conditions. This result aligns with expectations, given that SelfCheckGPT is specifically designed for hallucination detection, whereas few-shot prompting is a more general-purpose methodology. Among the SelfCheckGPT variants, the NLI-based method demonstrates the highest overall effectiveness and efficiency, surpassing the LLM prompting variant across all three benchmark datasets. With regard to few-shot prompting, the ten-shot configuration achieves the best performance, followed by the five-shot and one-shot variants, respectively. This trend is consistent with the hypothesis that providing a greater number of examples enables the LLM to better internalize the task structure, thereby improving generalization and overall accuracy.

In this regard, the strategy for selecting examples in the few-shot prompting approach could be improved. In the current evaluation, examples were randomly sampled from the datasets, which may result in class imbalance among the examples shown to the LLM, potentially affecting performance. Ensuring a balanced representation of classes in the selected examples would therefore be crucial for enhancing the robustness of the analysis in the few-shot prompting setting.

Regarding the impact of knowledge integration, on the FactAlign dataset, the only method that underperforms when incorporating external knowledge is few-shot prompting with five examples; all other tested methods either match or surpass the performance of their counterparts without knowledge. A similar trend is observed on FactBench, where all approaches that leverage retrieved knowledge perform at least as well as, and often better than, those without knowledge integration. Finally, in the FELM dataset, incorporating external knowledge generally leads to performance improvements across methods, with the sole exception of SelfCheckGPT using the LLM Prompt, where performance declines by one percentage point after knowledge integration. Overall, these analyses suggest that integrating external knowledge generally enhances the performance of the evaluated approaches across all datasets, with only a few exceptions where a slight decrease in performance was observed.

These performance declines may be attributed to limitations in the knowledge retrieval process. Specifically, only the first retrieved URL is considered—typically the most popular, but not necessarily the most informative. Additionally, the retrieval system occasionally fails to access relevant content due to Web restrictions, such as anti-bot mechanisms or CAPTCHA protections, which hinder the acquisition of valuable external knowledge. Nevertheless, on average, approaches augmented with external knowledge outperform their non-augmented counterparts. This suggests that further improvements in the retrieval process could improve the overall effectiveness of these methods and lead to even greater performance gains.

5. Conclusions and Perspectives

In this study, we introduced a fully automated knowledge retrieval framework that leverages a custom search engine interfacing with the Web via the Google Search API

to extract relevant external information. The retrieved knowledge was subsequently integrated into two distinct methodologies: (i) *few-shot prompting*, which consists of providing a set of examples to guide task execution, and (ii) *SelfCheckGPT*, a hallucination detection framework that generates and compares multiple responses from an LLM to identify factual inconsistencies. The enhanced versions of both approaches, incorporating retrieved knowledge, were evaluated on three benchmark datasets for hallucination detection—FactAlign, FactBench, and FELM—spanning a diverse range of domains. The experimental results indicate that SelfCheckGPT consistently outperforms the few-shot prompting approach, demonstrating strong performance across all three benchmark datasets. Among its variants, the NLI configuration emerges as the most effective and computationally efficient. Moreover, the integration of external knowledge generally enhances the performance of the evaluated approaches compared to their counterparts without such integration. Nonetheless, the observed improvements could be further amplified by refining the knowledge retrieval process in future work. Specifically, challenges such as CAPTCHA mechanisms or site access restrictions that limit automated retrieval should be addressed. Additionally, the quality of the queries submitted to the search engine could be improved by leveraging LLMs to generate more precise and contextually rich queries, thereby yielding more informative results. Moreover, expanding the number of retrieved Web sources may lead to more comprehensive and accurate knowledge; for instance, retrieving the top five results could increase the relevance and diversity of the retrieved information. Finally, future researches may also focus on further refining the knowledge integration process by leveraging more advanced and sophisticated RAG techniques [5]. Enhancing integration within frameworks such as SelfCheckGPT, which has already demonstrated promising results in hallucination detection, holds significant potential. These advancements could support the development of a reliable, scalable, and efficient multi-domain hallucination detection system.

Acknowledgments

This work was partly funded by: the European Union – Next Generation EU, Mission 4, Component 2, CUP: D53D23008480001 (20225WTRFN – KURAMi: *Knowledge-based, explainable User empowerment in Releasing private data and Assessing Misinformation in online environments*);[7] ATEQC – Progetti di Ricerca di Ateneo – Quota Competitiva (University Research Projects – Competitive Funding Scheme) PriQuaDeS: *Next-generation Privacy- and Quality-preserving Decentralized Social Web Applications*; the MUR under the grant "Dipartimenti di Eccellenza 2023-2027" of the Department of Informatics, Systems and Communication (DISCo), University of Milano-Bicocca, Italy. We further acknowledge ISCRA for awarding this project access to the LEONARDO supercomputer [16], owned by the EuroHPC Joint Undertaking, hosted by CINECA (Italy).

[7] https://kurami.disco.unimib.it/

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is All You Need, Advances in neural information processing systems 30 (2017) 1–11.

[2] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al., A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions, ACM Transactions on Information Systems 43 (2025) 1–55.

[3] U. Kruschwitz, M. Petrocchi, M. Viviani, ROMCIR 2025: Overview of the 5th Workshop on Reducing Online Misinformation Through Credible Information Retrieval, in: European Conference on Information Retrieval, Springer, 2025, pp. 339–344.

[4] V. Saxena, A. Sathe, S. Sandosh, Mitigating Hallucinations in Large Language Models: A Comprehensive Survey on Detection and Reduction Strategies, in: International Conference on Sustainable Computing and Intelligent Systems, Springer, 2025, pp. 39–52.

[5] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, H. Wang, Retrieval-Augmented Generation for Large Language Models: A Survey, 2024. URL: https://arxiv.org/abs/2312.10997. arXiv:2312.10997.

[6] P. Manakul, A. Liusie, M. Gales, SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 9004–9017.

[7] Y. Xiao, W. Y. Wang, On Hallucination and Predictive Uncertainty in Conditional Language Generation, in: P. Merlo, J. Tiedemann, R. Tsarfaty (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Association for Computational Linguistics, Online, 2021, pp. 2734–2744.

[8] A. Azaria, T. Mitchell, The Internal State of an LLM Knows When It's Lying, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association

for Computational Linguistics, Singapore, 2023, pp. 967–976.

[9] S. Min, K. Krishna, X. Lyu, M. Lewis, W.-t. Yih, P. Koh, M. Iyyer, L. Zettlemoyer, H. Hajishirzi, FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 12076–12100.

[10] C.-W. Huang, Y.-N. Chen, FactAlign: Long-form Factuality Alignment of Large Language Models, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2024, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 16363–16375. URL: https://aclanthology.org/2024.findings-emnlp.955/. doi:10.18653/v1/2024.findings-emnlp.955.

[11] Y. Wang, R. Gangi Reddy, Z. M. Mujahid, A. Arora, A. Rubashevskii, J. Geng, O. Mohammed Afzal, L. Pan, N. Borenstein, A. Pillai, I. Augenstein, I. Gurevych, P. Nakov, Factcheck-Bench: Fine-Grained Evaluation Benchmark for Automatic Fact-checkers, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2024, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 14199–14230. URL: https://aclanthology.org/2024.findings-emnlp.830/. doi:10.18653/v1/2024.findings-emnlp.830.

[12] S. Chen, Y. Zhao, J. Zhang, I.-C. Chern, S. Gao, P. Liu, J. He, FELM: Benchmarking Factuality Evaluation of Large Language Models, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023.

[13] I.-C. Chern, S. Chern, S. Chen, W. Yuan, K. Feng, C. Zhou, J. He, G. Neubig, P. Liu, FacTool: Factuality Detection in Generative AI – A Tool Augmented Framework for Multi-Task and Multi-Domain Scenarios, 2023. URL: https://arxiv.org/abs/2307.13528. arXiv:2307.13528.

[14] S. Lin, J. Hilton, O. Evans, TruthfulQA: Measuring How Models Mimic Human Falsehoods, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 3214–3252. URL: https://aclanthology.org/2022.acl-long.229/. doi:10.18653/v1/2022.acl-long.229.

[15] J. Li, X. Cheng, X. Zhao, J.-Y. Nie, J.-R. Wen, HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 6449–6464. URL: https://aclanthology.org/2023.emnlp-main.397/. doi:10.18653/v1/2023.emnlp-main.397.

[16] M. Turisini, G. Amati, M. Cestari, CINECA SuperComputing Centre, SuperComputing Applications and Innovation Department, LEONARDO: A Pan-European Pre-Exascale Supercomputer for HPC and AI applications, Journal of Large-Scale Research Facilities 9 (2024).

A. Online Resources

The datasets used for the experimental evaluations are publicly available, as referenced in the works cited throughout the paper. For the sake of reproducibility, the code developed in this study is also made publicly accessible at the following address: https://github.com/cristianceccarelli/rag-hallu.