Natural Language Inference Fine-tuning for Scientific Hallucination Detection

Tim Schopf[♠], **Juraj Vladika**[◇], **Michael Färber**[♠], **and Florian Matthes**[◇]
[♠]ScaDS.AI & Dresden University of Technology
[◇]Technical University of Munich
{tim.schopf,michael.faerber}@tu-dresden.de
{juraj.vladika,matthes}@tum.de

Abstract

Modern generative Large Language Models (LLMs) are capable of generating text that sounds coherent and convincing, but are also prone to producing *hallucinations*, facts that contradict the world knowledge. Even in the case of Retrieval-Augmented Generation (RAG) systems, where relevant context is first retrieved and passed in the input, the generated facts can contradict or not be verifiable by the provided references. This has motivated SciHal 2025, a shared task that focuses on the detection of hallucinations for scientific content. The two sub-tasks focused on: (1) predicting whether a claim from a generated LLM answer is entailed, contradicted, or unverifiable by the used references; (2) predicting a fine-grained category of erroneous claims. Our best performing approach used an ensemble of fine-tuned encoder-only ModernBERT and DeBERTa-v3 models for classification. Out of nine competing teams, our approach achieved the first place in sub-task 1 and the second place in sub-task 2.

1 Introduction

The increasing availability of academic research assistants based on Large Language Models (LLMs) have revolutionized the way research is conducted, enabling users to pose research-related questions in natural language and receive structured and concise summaries supported by relevant references (Eger et al., 2025; Schmidgall et al., 2025). These systems have the potential to greatly accelerate the research process, facilitating the discovery of new knowledge and insights (Schopf and Matthes, 2024). However, the tendency of LLMs to introduce *hallucinations* – claims that are not supported or grounded in relevant evidence or established world knowledge – poses a significant challenge to the reliability of these automatically generated scientific answers (Huang et al., 2025b). Hallucinations can lead to the dissemination of misinformation, undermining the validity of research findings and the trustworthiness of AI-powered research tools (Huang et al., 2024).

To address this issue, the SciHal shared task was established, focusing on the detection of hallucinated claims in answers generated by AI-powered research assistants. The task provides a dataset of research-oriented questions, the corresponding answers and references, annotated with labels indicating the presence and type of hallucinations. By developing systems that can accurately detect hallucinations, researchers can take a crucial step towards ensuring the reliability and trustworthiness of AI-enhanced research assistants.

In response to this challenge, we developed an approach using an ensemble of fine-tuned encoder-only models DeBERTa-v3 and ModernBERT. This approach achieved the first place on sub-task 1. This paper describes our model architecture, training procedure, and results on the shared task. The performance of our approach on the task demonstrates the potential of machine learning models to identify hallucinations and improve the accuracy of generated answers. We outline our findings, challenges, and directions for future improvements.

2 Related work

Hallucinations in LLMs refer to the generation of fluent but factually incorrect or inconsistent claims (Ji et al., 2023; Zhang et al., 2023; Sahoo et al., 2024; Huang et al., 2025a; Xu et al., 2025). Factual hallucinations are outputs that deviate from real-world facts and can be addressed through fact-checking, which verifies the accuracy of claims (Guo et al., 2022; Sahnan et al., 2025). Manual fact checking is labor intensive and time consuming (Hassan et al., 2015), prompting research into automated approaches.

These approaches typically involve broad classifications (e.g., supported, refuted, not enough infor-

Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025), pages 344–352

mation), limiting their applicability in real-world scenarios (Vladika and Matthes, 2023a). To improve utility, finer-grained classification schemes have been proposed, reflecting degrees of truthfulness (Wang, 2017; Alhindi et al., 2018, *inter alia*). Some methods retain original fact-checking labels (Augenstein et al., 2019), while others consolidate categories for simplicity (Hanselowski et al., 2019; Kotonya and Toni, 2020; Gupta and Srikumar, 2021). Typically, scientific text classification is conducted in a supervised manner (Sadat and Caragea, 2022; E. Mendoza et al., 2022; Schopf et al., 2023), while some approaches support scenarios where labeled training data is scarce (Shen et al., 2018; Toney and Dunham, 2022; Schopf et al., 2024). Final claim veracity prediction is often modeled as a Natural Language Inference (NLI) task, where a relation between a premise and a hypothesis (entailment, contradiction, neutral) must be predicted (Vladika and Matthes, 2023b; Laurer et al., 2024). This paper investigates two sub-tasks: one using coarse-grained labels and another with finer-grained classifications to assess whether an LLM generated claim is a hallucination, given reference evidence.

3 Task Description

The SciHal 2025 shared task addresses the critical challenge of factual inconsistency in responses generated by generative AI-powered academic research assistants. SciHal formulates this problem as a classification task, focused on evaluating the factual alignment between individual claims and their supporting evidence. Given a research-focused question, an LLM generated response from a Retrieval-Augmented Generation (RAG) system, an extracted claim from the response, and a reference retrieved from a large corpus of scientific literature that is used to ground the generated response, the objective is to classify the claim based on its factual consistency with the provided reference. SciHal 2025 is structured into two sub-tasks:

Sub-task 1 involves coarse-grained classification of each claim into one of three categories: *Entailment*, *Unverifiable*, or *Contradiction*.

Sub-task 2 extends this formulation by employing a fine-grained label set. Each claim must be categorized as one of the following: *Entailment*, *Unrelated and unverifiable*, *Related but unverifiable*, *Misrepresentation*, *Missing information*, *Numeric error*, *Entity error*, or *Opposite meaning*.

4 Dataset

The SciHal dataset comprises labeled claims designed to evaluate hallucination detection in scientific assistant outputs. The data creation process involves both real and synthetic components, ensuring a diverse and balanced distribution of hallucination types.

Data Collection Over 50,000 real-user queries were collected from a live academic assistant system over a week. These questions focused on the five scientific fields Engineering, Environmental Science, Medicine, Agricultural and Biological Sciences, and Computer Science. After de-identification and refinement, 500 questions were retained. For each question, a RAG system indexed over a million scientific abstracts to retrieve the top 20 most relevant documents. The system then generated an answer, from which individual claims were extracted. Each claim was paired with the retrieved references used to justify the answer.

Synthetic Hallucination Generation To balance the dataset across hallucination types, 75% of the claims were synthetically modified using LLM prompting, simulating errors aligned with the classification labels. This method ensured controlled type distributions, where entailment accounts for less than 25% and other types each account for under 10% of the labels.

Annotation Process The annotation process for the dataset was conducted through subject matter experts (SMEs). SMEs received the claims, references, and detailed guidelines, including definitions of hallucination types, a decision tree, and a trial phase to ensure they were aligned with the task's requirements and labeling standards. To strike a balance between annotation quality and cost, both human SME annotations and an internal LLM-based hallucination detection method were used. The data was released in following batches:

- Batch 1 & 2: Instances where SME and LLM labels agreed. Batch 1 is a subset of Batch 2.
- Batch 3 & Test Set: In cases where SME and LLM labels disagreed, the claim was re-labeled by a second SME. To resolve any remaining discrepancies, a third SME was involved in adjudicating the label.

5 Approaches

To identify hallucinated claims, we explore a range of approaches spanning zero-shot prompting and supervised fine-tuning, leveraging both encoder-only and decoder-only models.

DeepSeek-R1 Zero-shot We use the DeepSeek-R1 model (DeepSeek-AI et al., 2025) in a zero-shot setting to classify claims into predefined categories using the associated reference as supporting evidence. The prompt includes a task definition and detailed descriptions of each classification label. The full prompt is provided in Figure 1.

DeepSeek-R1 Zero-shot with Claim Decomposition Building on the basic zero-shot setup, we extend the prompting strategy by explicitly instructing DeepSeek-R1 to first decompose the claim into its constituent subclaims. The model then classifies each subclaim individually and aggregates the results into a final prediction for the full claim. This decomposition aims to enhance reasoning granularity. The corresponding prompt is in Figure 2.

GPT-4o Zero-shot We evaluate GPT-4o (OpenAI et al., 2023) using the same zero-shot prompt as above (Figure 1). To mitigate variance stemming from the non-deterministic behavior of the model, we generate ten independent predictions per input and derive the final class prediction via majority voting. This ensemble-like setup enhances prediction stability and robustness.

DeBERTa-v3 Fine-tuning We fine-tune a DeBERTa-v3 large model (He et al., 2023)[1], pretrained on several Natural Language Inference (NLI) datasets including MultiNLI (Williams et al., 2018), Fever-NLI (Nie et al., 2019), Adversarial-NLI (Nie et al., 2020), LingNLI (Parrish et al., 2021), and WANLI (Liu et al., 2022), comprising a total of 885,242 hypothesis-premise pairs. We also evaluate a DeBERTa-v3 base variant[2] fine-tuned on the *tasksource* dataset (Sileo, 2024). For both models, we experiment with different fine-tuning data configurations: using batch 2, batch 3, and their combination.

ModernBERT Fine-tuning We also experiment with ModernBERT[3] (Warner et al., 2024), a recent improved and optimized version of BERT (Devlin et al., 2019). We again use the version previously trained on *tasksource* data and fine-tune it on batches 2 and 3.

Ensemble We investigate an ensemble approach, where predictions of three fine-tuned encoder-only models that performed well on the leaderboard are combined using majority voting. This includes DeBERTa-v3 NLI (batch 3) and ModernBERT Tasksource (batches 2+3 & batch 3).

Llama Fine-tuning To investigate the potential of a decoder-only model, we fine-tune LLama3.1-8B-Instruct (Grattafiori et al., 2024). We train the model to generate the label annotation justifications contained in the training data before predicting the classification labels. This approach ensures that the model explicitly thinks and reasons prior to the classification. Fine-tuning is conducted exclusively on batch 3, which closely reflects the distribution of the test set.

To optimize resource usage, we initially evaluate all methods on sub-task 1. Based on the performance results, we then adapt the best-performing approach for sub-task 2.

6 Evaluation

The primary evaluation metric for the shared task is the weighted F_1 score. It is computed by calculating the F_1 score independently for each class and then taking the average, weighted by the number of true instances (support) for each class.

	Approach	F_1
prompt	DeepSeek-R1 Zero-shot	0.49
	DeepSeek-R1 Zero-shot Decompose	0.44
	GPT-4o Zero-shot	0.43
fine-tune	LLama3.1-8B-Instruct	0.50
	DeBERTa-v3 NLI (batch 2)	0.50
	DeBERTa-v3 NLI (batch 3)	0.57
	DeBERTa-v3 NLI (batch 2+3)	0.56
	DeBERTa-v3 Tasksource (batch 3)	0.50
	DeBERTa-v3 Tasksource (batch 2+3)	0.54
	ModernBERT Tasksource (batch 2+3)	0.57
	ModernBERT Tasksource (batch 3)	0.56
	Ensemble of DeBERTa NLI (batch 3), ModernBERT Taskso. (batch 2+3 & 3)	**0.60**

Table 1: Comparison of Approaches and their F_1 scores for sub-task 1 on 50% of the test data.

Each sub-task's test set comprises 1,000 examples, with 50% designated for official evaluation and leaderboard ranking during the challenge. The

[1] MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli

[2] tasksource/deberta-base-long-nli

[3] tasksource/ModernBERT-large-nli

remaining 50% is withheld and only evaluated after the competition concludes. Accordingly, all reported results in this paper are based on the publicly accessible 50% split of the respective test sets.

As shown in Table 1, fine-tuning the DeBERTa-v3 NLI and ModernBERT models achieves good results. When combined in an ensemble, this achieves the winning score of 0.60 on sub-task 1. For sub-task 2, we use DeBERTa-v3 NLI fine-tuned on batch 3, where it achieves a F_1 score of 0.50 and secures second place on the leaderboard.

7 Discussion

Our findings show that the dataset poses a considerable challenge and that fine-tuned models clearly outperform prompting-based approaches. Notably, the smaller encoder-only DeBERTa-v3 and ModernBERT models achieve better results than much larger decoder-only LLMs. Despite their scale, LLMs such as DeepSeek-R1 and GPT-4o struggle in prompting setups compared to fine-tuned ModernBERT and DeBERTa-v3 variants.

Interestingly, advanced prompting techniques, such as claim decomposition, do not improve classification performance. In fact, they often underperform compared to simpler zero-shot prompting. To understand this behavior, we perform a detailed analysis of both the dataset and the prediction behaviors of the model.

We observe that the test sets are inherently difficult due to the way they were constructed: they include only those instances where initial predictions by SMEs and LLMs diverged. These disagreements were later resolved by a third SME. However, the data annotations remain often ambiguous, inconsistent, and challenging. During our manual inspection, we identified multiple very similar instances with different labels. Inconsistent labels were particularly common in examples annotated as unverifiable (unver) or contradiction (contra). For instance, claims that involved information not present in the reference were sometimes labeled 'contra' and other times 'unver', even when the annotation justification was nearly identical.

Prompt-based approaches are particularly affected by this inconsistency. Given that prompts contain fixed class definitions, the models tend to adhere to those instructions. For instance, when claim content is missing from the reference, LLMs frequently predict 'unver', aligning with the prompt's class description, although the example is labeled as 'contra'. We also identified inconsistencies in the annotation of entailment (entail) cases. Some instances were labeled as 'entail' only when the claim's content was explicitly stated in the reference, while others were labeled 'entail' even when the reference only implicitly supported the claim through inference. However, the instructions provided in the prompt resulted in the LLM to rely strictly on explicit information and often misclassified such implicit entailment examples as 'unver'. Internal validation supports these observations: all prompting-based approaches demonstrated particularly low precision for the 'unver' class.

Contrary to our expectations, decomposing claims into subclaims did not improve performance. In fact, this led to overly conservative predictions. For example, the model would identify one unsupported detail within a claim and classify the entire example accordingly, even when the overall meaning was supported. The annotators, by contrast, appeared to take a more holistic view, labeling a claim as entailment based on general alignment, even when minor details were not mentioned.

Overall, these findings suggest that prompting-based methods lack the flexibility required to handle the annotation noise and implicit reasoning present in the dataset. In contrast, fine-tuned models can better adapt to such irregularities, likely because they learn implicit patterns and labeling conventions from the training data.

Finally, the strong performance of smaller encoder-only models highlights the importance of task-specific training. The ModernBERT and DeBERTa-v3 models were already trained on a diverse set of NLI datasets, whereas the Llama3.1-8B-Instruct model was not. This likely gave the smaller models a major advantage, suggesting that task-specific training on relevant datasets can outweigh model scale for downstream performance.

8 Future work

In future work, we aim to further improve the fine-tuning process of decoder-only language models considering their vast world knowledge and reasoning capabilities. Given that we achieved the best result using an ensemble, we additionally aim to experiment more with advanced ensembles and committee voting techniques, including the introduction of weighting mechanisms. Finally, we plan to incorporate hierarchical classification in the form of multi-step predictions for sub-task 2 involving

fine-grained labels.

9 Conclusion

We presented fine-tuning approaches based on ModernBERT and DeBERTa-v3 that consistently outperformed baseline methods and other submitted solutions. This success is largely attributable to prior training on extensive NLI datasets, which closely align with the nature of the target tasks. Notably, the same approach demonstrates strong performance on both sub-task 1 and sub-task 2, underscoring its generalizability across related tasks.

Our findings further suggest that in scenarios where the data is inherently challenging—due to ambiguity or inconsistent labeling, fine-tuning offers a clear advantage over prompt-based LLM approaches. While prompting yields consistent predictions based on static label definitions, it lacks the flexibility to adapt to subtle patterns and inconsistencies in the data. In contrast, fine-tuned models are better able to internalize such nuances.

Moreover, our results highlight the importance of training on data that closely resembles the target task. Models exposed to large volumes of relevant data prior to task-specific fine-tuning consistently achieve superior downstream performance. Notably, this results in smaller models using this strategy outperforming larger models that lack similar task-aligned training.

References

Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. Where is your evidence: Improving fact-checking by justification modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen. 2019. MultiFC: A real-world multi-domain dataset for evidence-based fact checking of claims. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4685–4697, Hong Kong, China. Association for Computational Linguistics.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Óscar E. Mendoza, Wojciech Kusa, Alaa El-Ebshihy, Ronin Wu, David Pride, Petr Knoth, Drahomira Herrmannova, Florina Piroi, Gabriella Pasi, and Allan Hanbury. 2022. Benchmark for research theme classification of scholarly documents. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 253–262, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Steffen Eger, Yong Cao, Jennifer D'Souza, Andreas Geiger, Christian Greisinger, Stephanie Gross, Yufang Hou, Brigitte Krenn, Anne Lauscher, Yizhi Li, Chenghua Lin, Nafise Sadat Moosavi, Wei Zhao, and Tristan Miller. 2025. Transforming science with large language models: A survey on ai-assisted scientific discovery, experimentation, content generation, and evaluation. *Preprint*, arXiv:2502.05151.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Ashim Gupta and Vivek Srikumar. 2021. X-fact: A new benchmark dataset for multilingual fact checking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 675–682, Online. Association for Computational Linguistics.

Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. 2019. A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 493–503, Hong Kong, China. Association for Computational Linguistics.

Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting check-worthy factual claims in presidential debates. In *Proceedings of the 24th ACM In-*

ternational on Conference on Information and Knowledge Management, CIKM '15, page 1835–1838, New York, NY, USA. Association for Computing Machinery.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025a. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2).

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025b. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Yue Huang, Lichao Sun, Haoran Wang, Siyuan Wu, Qihui Zhang, Yuan Li, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, Xiner Li, Hanchi Sun, Zhengliang Liu, Yixin Liu, Yijue Wang, Zhikun Zhang, Bertie Vidgen, Bhavya Kailkhura, Caiming Xiong, and 52 others. 2024. TrustLLM: Trustworthiness in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20166–20270. PMLR.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Neema Kotonya and Francesca Toni. 2020. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754, Online. Association for Computational Linguistics.

Moritz Laurer, Wouter Van Atteveldt, Andreu Casas, and Kasper Welbers. 2024. Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli. *Political Analysis*, 32(1):84–100.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Association for the Advancement of Artificial Intelligence (AAAI)*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Alicia Parrish, William Huang, Omar Agha, Soo-Hwan Lee, Nikita Nangia, Alexia Warstadt, Karmanya Aggarwal, Emily Allaway, Tal Linzen, and Samuel R. Bowman. 2021. Does putting a linguist in the loop improve NLU data collection? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4886–4901, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mobashir Sadat and Cornelia Caragea. 2022. Hierarchical multi-label classification of scientific documents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8923–8937, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dhruv Sahnan, David Corney, Irene Larraz, Giovanni Zagni, Ruben Miguez, Zhuohan Xie, Iryna Gurevych, Elizabeth Churchill, Tanmoy Chakraborty, and Preslav Nakov. 2025. Can llms automate fact-checking article writing? *Preprint*, arXiv:2503.17684.

Pranab Sahoo, Prabhash Meharia, Akash Ghosh, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. A comprehensive survey of hallucination in large language, image, video and audio foundation models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11709–11724, Miami, Florida, USA. Association for Computational Linguistics.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Zicheng Liu, and Emad Barsoum. 2025. Agent laboratory: Using llm agents as research assistants. *arXiv preprint arXiv:2501.04227*.

Tim Schopf, Karim Arabi, and Florian Matthes. 2023. Exploring the landscape of natural language processing research. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1034–1045, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Tim Schopf, Alexander Blatzheim, Nektarios Machner, and Florian Matthes. 2024. Efficient few-shot learning for multi-label classification of scientific documents with many classes. In *Proceedings of the 7th International Conference on Natural Language and Speech Processing (ICNLSP 2024)*, pages 186–198, Trento. Association for Computational Linguistics.

Tim Schopf and Florian Matthes. 2024. NLP-KG: A system for exploratory search of scientific literature in natural language processing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 127–135, Bangkok, Thailand. Association for Computational Linguistics.

Zhihong Shen, Hao Ma, and Kuansan Wang. 2018. A web-scale system for scientific knowledge exploration. In *Proceedings of ACL 2018, System Demonstrations*, pages 87–92, Melbourne, Australia. Association for Computational Linguistics.

Damien Sileo. 2024. tasksource: A large collection of NLP tasks with a structured dataset preprocessing framework. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15655–15684, Torino, Italia. ELRA and ICCL.

Autumn Toney and James Dunham. 2022. Multi-label classification of scientific research documents across domains and languages. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 105–114, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Juraj Vladika and Florian Matthes. 2023a. Scientific fact-checking: A survey of resources and approaches. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6215–6230, Toronto, Canada. Association for Computational Linguistics.

Juraj Vladika and Florian Matthes. 2023b. Sebis at SemEval-2023 task 7: A joint system for natural language inference and evidence retrieval from clinical trial reports. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1863–1870, Toronto, Canada. Association for Computational Linguistics.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2025. Hallucination is inevitable: An innate limitation of large language models. *Preprint*, arXiv:2401.11817.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

A Appendix

The appendix shows the prompts used for classification, including the simple zero-shot prompt (Figure 1 and the prompt for subclaim decomposition and aggregated prediction (Figure 2).

Simple Zero-shot Prompt

```
Determine whether the provided claim is entailed by the corresponding evidence
. Entailment in this context implies that all information presented in the
claim is substantiated by the evidence. If any information in the claim is
contradicted by at least one information in the evidence, the claim is
contradicted. If the claim is neither entailed nor contradicted by the
evidence, the claim is unverifiable.

Evidence: {reference}
Claim: {claim}

Assess the claim's entailment with the evidence by predicting either 'entail'
for entailment, 'contra' for contradication, or 'unver' for unverifiable.
Explain your decision and afterwards provide your prediction in JSON format as
 one of the options {'prediction': 'entail'}, {'prediction': 'contra'}, {'
prediction': 'unver'}.
```

Figure 1: Simple zero-shot prompt to instruct an LLM to detect a hallucinated claim.

Zero-shot Claim Decomposition Prompt

```
Instruction:
Decompose the claim into its individual subclaims (e.g., distinct factual
assertions or components). For each subclaim, determine whether it is entailed
, contradicted, or unverifiable based on the provided evidence. Use the
following criteria:

Entail (entail): All information presented in the subclaims are substantiated
by the evidence. Usually, this means that the information is directly included
 in the evidence. However, a subclaim can also be entailed if the evidence can
 be used to infer the subclaim.
Contradiction (contra): At least one piece of evidence explicitly contradicts
the subclaim. Contradiction in this sense also means that a claim mentions one
 thing, but the evidence only supports the claim's statement regarding a
different thing. Or it could be a contradiction (instead of unverifiably) if a
 claim is overgeneralized, oversimplified, or overstates the evidence.
Unverifiable (unver): The subclaim is neither supported nor contradicted by
the evidence.

After evaluating all subclaims, determine the overall prediction for the full
claim using these rules:
If any subclaim is contradicted, the overall prediction is "contra".
If all subclaims are entailed, the overall prediction is "entail".
Otherwise, the overall prediction is "unver".

Process:
Decomposition: Break the claim into subclaims (e.g., "Subclaim 1: [X].
Subclaim 2: [Y].").
Evaluation: For each subclaim, explain whether it is entailed, contradicted,
or unverifiable.
Aggregation: Combine subclaim results to determine the overall prediction.

Output Format:
Provide a detailed explanation for each subclaim and the overall prediction.

Return the final answer in JSON format with two keys:
"subclaims": A list of objects, each containing "subclaim" (text), "
justification" evaluation (text), and "prediction".
"overall_prediction": One of "entail", "contra", or "unver".
Example Output:
     {
"subclaims": [
{"subclaim": "Subclaim 1 text", "justification": "Explanation of evaluation
for sublaim 1", "prediction": "entail"},
{"subclaim": "Subclaim 2 text", "justification": "Explanation of evaluation
for sublaim 2", "prediction": "unver"}
],
"overall_prediction": "unver"
}

Evidence: {reference}
Claim: {claim}
```

Figure 2: Zero-shot prompt to instruct an LLM to decompose a claim into subclaims, predict the class of each subclaim and aggregate the predictions to one overall prediction.