HypoTermInstruct: Instructing Large Language Models not to Hallucinate

Cem ULUOGLAKCI
Information Systems
Middle East Technical University
Ankara, Turkey
cem.uluoglakci@metu.edu.tr

Tugba TASKAYA TEMIZEL
Data Informatics
Middle East Technical University
Ankara, Turkey
ttemizel@metu.edu.tr

Abstract

Large language models (LLMs) often hallucinate producing fluent but false information—partly because supervised fine-tuning (SFT) implicitly rewards always responding. We introduce **HypoTermInstruct**, an architecture-agnostic SFT dataset (31,487 responses for 11,151 questions) that teaches models to acknowledge uncertainty using systematically generated queries about validated non-existent (*hypothetical*) terms. We also release **HypoTermQA-v2**, a benchmark for hallucination tendency strengthened through multiple validations. In 400 controlled LoRA SFT runs (Llama3.1-8B-Instruct, Gemma3-4B-it; 100 fine-tuning configurations each with paired control) substituting generic instruction samples with HypoTermInstruct increases HypoTerm Score by +1.36% to +26.46% (median diffs) and FactScore by +0.52-0.61%, with modest MMLU decreases (-0.26–0.31%) and negligible shifts in instruction following and safety. Results show targeted uncertainty instruction during SFT reduces hallucination without architecture-specific engineering or preference/RL pipelines.

1 Introduction

LLM hallucination erodes user trust and poses significant risks, making its mitigation an important area of research for developing dependable AI systems. Current approaches to combat hallucination primarily focus on curating higher-quality pre-training data [Abdin et al., 2024, Zhou et al., 2024, Cao et al., 2023, Chen et al., 2023, Elaraby et al., 2023], detecting fabricated content post-generation, or using preference-based methods like Reinforcement Learning (RL) [Tian et al., 2023, Jones et al., 2023, Wang et al., 2023, Yang et al., 2023] to discourage undesirable outputs. While valuable, these methods often do not directly address a core issue: LLMs are generally aware of whether they possess knowledge about a topic [Azaria and Mitchell, 2023], yet during SFT, models are implicitly trained to generate responses regardless of their knowledge state [Gekhman et al., 2024, Spataru et al., 2024]. Existing SFT-based solutions, in turn, are frequently tailored to specific domains or model architectures, limiting their general applicability[Zhang et al., 2023, Wan et al., 2024, Deng et al., 2024].

To address this gap, we introduce **HypoTermInstruct**, a novel, scalable, domain-independent, and architecture-agnostic approach to teach models uncertainty during the SFT phase. Our method leverages questions about non-existent, or "hypothetical," terms as a reliable signal for knowledge gaps, training the model to explicitly acknowledge its lack of information instead of inventing an answer. Our contributions are threefold: (1) We develop **HypoTermQA-v2**, a benchmark for hallucination tendencies using a multi-engine

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Evaluating the Evolving LLM Lifecycle.

validation process. (2) We release **HypoTermInstruct** dataset teaching models to properly acknowledge uncertainty. (3) In 400 fine-tuning runs it consistently reduces hallucination with small general-knowledge costs. Code, data and results are public[1]; checkpoints available on request.

2 Benchmarking Hallucination Tendency

HypotermQA [Uluoglakci and Temizel, 2024] uses LLMs to generate questions that pair a valid term with a semantically similar but non-existent one. By presenting the valid term first, the question structure exploits the autoregressive nature of LLMs, making them more likely to fabricate information about the non-existent term rather than acknowledging its non-existence.

While this approach effectively exploits LLM weaknesses using non-existent terms, its validation method is insufficient, as it relies on a single search engine's exact match result to declare a term non-existent. To address this, we introduce **HypoTermQA-v2**, which strengthens validation through (1) multi-engine search, (2) searching against the Dolma dataset [Soldaini et al., 2024], a large-scale LLM pretraining corpus, and (3) checking for term variations. This includes word permutations (e.g., "Viral content momentum" vs. "Momentum of Viral Content"), hyphen removal, and lexical alternatives (e.g., "Circuitry" vs. "Circuit"), which improves detection of terms that might otherwise be missed.

The improved dataset retains the original approach while improving validation reliability. Applying three validation criteria reduced hypothetical terms from 909 to 676. We regenerated benchmarking questions with Llama-3.1-405B using these refined terms.

Figure 1 presents benchmarking results for 15 recent LLMs on HypoTermQA-v2. Inference experiments were conducted using H100 64GB GPUs with a total evaluation time of 4K GPU hours. Performance ranges from Llama3.1-405B (20.66% HypoTerm Score—the percentage of valid responses to hypothetical term questions) to Gemma3-1B (0.32%). While larger, more recent models generally hallucinate less, notable exceptions exist: Llama2-70B outperforms Llama3-70B, and Gemma3-4B outperforms Gemma3-27B.

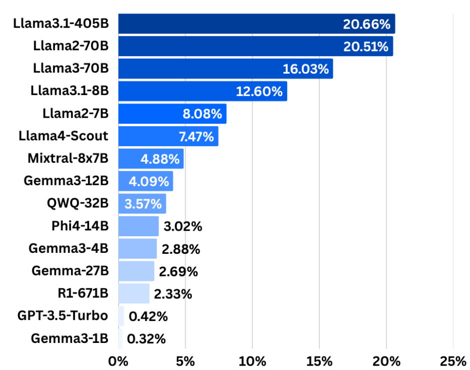

Figure 1: Evaluation results on HypoTermQA-v2 dataset.

Importantly, advanced architectural approaches do not guarantee reduced hallucination. Mixtral-8x7B underperforms Llama2-7B, while reasoning model R1-671B ranks among the lowest performers. These findings align with recent studies [Chen et al., 2025, Shojaee et al., 2025] showing that LLM architectural advancements do not necessarily improve reliability.

3 Reducing Hallucination Tendency

HypoTermInstruct Dataset Creation: Inspired by HypoTermQA [Uluoglakci and Temizel, 2024], we use validated non-existent terms to automatically generate training data that teaches models to acknowledge unknown concepts, avoiding manual annotation. Rather than compressing specific information, our method teaches a domain-independent behavior: acknowledging a lack of knowledge. Using the prompt in Appendix B, we instructed Llama-3.1-405B, R1-671B, and GPT-4o to generate responses that admit a term's non-existence rather than fabricating information.

The resulting dataset contains 31,487 high-quality responses for 11,151 questions on 20 topics. Topics with id 0 and 1 (Technology and gadgets, Social media and influencers) were spared

[1] `https://github.com/cemuluoglakci/HypoTermInstruct`

as test set. Topics with id 2 and 3 (News and current events, Entertainment) were used as validation set. Additional dataset creation details are provided in Appendix B.

Training the Models: We performed SFT to compare models trained with and without HypoTermInstruct in our experiments. Following prior work showing benefits of diverse training data [Touvron et al., 2023, Dubey et al., 2024], we used seven complementary instruction-following datasets including Alpaca, DEITA, Conifer, Muffin, CotCollection, CoEdIT, and Ultrachat (Appendix C). The Control Dataset combines these datasets, while the Experimental Dataset adds HypoTermInstruct with the same total training sample size. Each dataset was capped at 20k samples to balance size and reduce overfitting.

Evaluating the Models: Our primary objective is to reduce hallucination tendencies in LLMs while maintaining their overall utility. Since a model that never generates responses would achieve 0% hallucination but provide no practical value, we evaluate models across multiple dimensions to ensure balanced performance. We employ six evaluation metrics: HypoTerm Score [Uluoglakci and Temizel, 2024] and FactScore [Min et al., 2023] to measure hallucination tendency, MMLU [Press et al., 2022] for general knowledge, IF Instruct and IF Prompt [Zhou et al., 2023] for instruction-following capability, and AILuminate [Ghosh et al., 2025] for safety assessment. Detailed descriptions of these benchmarking datasets are provided in Appendix D.

4 Experiments

To isolate our data's impact, our experimental design compares two training dataset compositions with an identical total sample count. The Control dataset combines seven instruction-following datasets. The Experimental dataset incorporates HypoTermInstruct by proportionally replacing samples from the other seven. This design ensures any performance change is attributable to data quality, not an increase in data quantity.

We evaluate on Llama3.1-8B-Instruct and Gemma3-4B-it. Each model is trained with 100 random fine-tuning configurations (learning rate, batch size, epochs, LoRA parameters - Appendix F) using fixed seed 42 for reproducibility, yielding 400 total experiments (2 models x 100 fine-tuning configurations x 2 dataset conditions). Training was conducted on H100 80GB GPUs with a total training time of 11K GPU hours.

To capture the effect of HypoTermInstruct on multiple aspects of model behavior, we assess performance across six distinct metrics. Three of these metrics—HypoTerm Score, FactScore, and the AILuminate safety score—require using an LLM as a judge. This comprehensive evaluation, which required 7K GPU hours on H100 64GB GPUs. We use Wilcoxon signed-rank tests to evaluate statistical significance, accounting for the paired nature of our experimental design. More detail on experimental design and variables is provided in Appendix E.

5 Results

We analyze 400 paired fine-tuning runs. Each pair differs only in substituting a proportion of generic instruction data with HypoTermInstruct while keeping total sample count constant. Statistical significance is evaluated using Wilcoxon signed-rank tests accounting for the paired experimental design.

Table 1 summarizes the median performance differences across all 400 experiments, with color coding highlighting our key finding: significant improvements in hallucination metrics (green) come with acceptable trade-offs in other areas. Grey shows non-significant changes, and red represents significant decreases. P-values and mean differences are provided in Appendix G.

Hallucination Reduction: Incorporating HypoTermInstruct consistently and significantly improves both HypoTerm Score and FactScore across all both architectures. The improvements are substantial, with HypoTerm Score gains ranging from 1.36% to 26.46% (median differences) and FactScore improvements from 0.52% to 0.61%.

Model	IF Prompt	IF Inst.	MMLU	FactScore	Hypoterm	Safety
Llama3.1-8B-Instruct	-0.46%	-0.24%	-0.26%	0.52%	1.36%	-0.58%
Gemma3-4B-it	0.55%	0.30%	-0.31%	0.61%	26.46%	-0.46%

Table 1: Median differences after introducing HypoTermInstruct.

Performance Trade-offs: HypoTermInstruct inclusion reduces MMLU performance across both models, though it is only significant for Llama3.1-8B Instruct model. For instruction-following (IF Instruct and IF Prompt), Llama3.1-8B Instruct shows non-significant decrease, while Gemma3-4B-it shows non-significant increase. These performance variations can be attributed to the proportional reduction of general-purpose SFT data when HypoTermInstruct is included.

Safety Implications: Since our experiments did not include dedicated safety training, incorporating HypoTermInstruct results in non-significant reductions in safety scores for both models. Importantly, HypoTermInstruct does not introduce significant safety risks. Additional safety-focused training would likely mitigate these minor decreases.

Summary. The results validate our core hypothesis that models can be taught to acknowledge uncertainty during SFT. HypoTermInstruct successfully reduces hallucination tendencies with manageable trade-offs in knowledge-intensive tasks and controllable safety implications through complementary training approaches.

6 Related Work

Research on LLM hallucinations spans several approaches. **Detection methods** identify hallucinated content post-generation [Min et al., 2023, Yin et al., 2023, Liang et al., 2023] but cannot prevent hallucinations. **Pre-training data quality** approaches reduce hallucinations from the pretraining phase [Abdin et al., 2024, Zhou et al., 2024, Chen et al., 2023, Cao et al., 2023, Elaraby et al., 2023], while **preference-based methods** use RLHF to discourage fabricated responses [Tian et al., 2023, Jones et al., 2023, Wang et al., 2023, Yang et al., 2023].

Most relevant are studies addressing hallucinations during **SFT**. Some methods attempt to filter training data by first checking if a pre-trained model already possesses the relevant knowledge, a process that is inherently tied to a specific model checkpoint and thus not generalizable [Zhang et al., 2023, Wan et al., 2024, Deng et al., 2024]. Another line of work reduces hallucination by performing SFT with domain-specific knowledge to generate specialist LLMs [Shi et al., 2023]. In contrast, our approach aims to teach a domain-independent behavior, using hypothetical terms guaranteed to be absent from any model's pre-training data and offering a truly architecture-agnostic solution. Our work builds upon HypoTermQA's automated evaluation framework [Uluoglakci and Temizel, 2024], complementing existing pre-training quality efforts with a scalable SFT solution. Following the taxonomy proposed by Huang et al. [2025], which distinguishes between factuality and faithfulness hallucinations, our work specifically addresses factuality hallucinations by teaching models to decline from fabricating information about non-existent concepts.

7 Conclusion

This paper presents HypoTermInstruct, a domain-independent SFT dataset designed to reduce hallucination tendencies in LLMs. Our experiments show that incorporating our dataset consistently improves hallucination-related metrics (HypoTerm Score and FactScore) while maintaining instruction-following capabilities. Although we observe trade-offs with general performance (MMLU, IFEval and Safety), these reductions are not consistent across all model architectures and training scenarios, and can potentially be mitigated by increasing the size of general-purpose training data. The significant and consistent improvements in reliability metrics validate our core hypothesis that models can be taught to acknowledge uncertainty rather than fabricate information. Our approach provides a scalable, architecture-agnostic solution for improving model reliability during the SFT phase.

Acknowledgments and Disclosure of Funding

LLM training experiments conducted in this study were performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources). Benchmarking experiments using LLMs as a judge (agent calls) were performed using the EuroHPC Joint Undertaking (EuroHPC JU) supercomputer MareNostrum 5, hosted by the Barcelona Supercomputing Center (BSC). Access to MareNostrum 5 was provided through a national access call coordinated by the Scientific and Technological Research Council of Turkey (TÜBİTAK). We gratefully acknowledge TRUBA, TÜBİTAK, BSC, and the EuroHPC JU for providing access to these resources and supporting this research.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.68. URL `https://aclanthology.org/2023.findings-emnlp.68`.

Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. Instruction mining: Instruction data selection for tuning large language models. *arXiv preprint arXiv:2307.06290*, 2023.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. Alpagasus: Training a better alpaca with fewer data. *arXiv preprint arXiv:2307.08701*, 2023.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006.

Yang Deng, Yong Zhao, Moxin Li, See-Kiong Ng, and Tat-Seng Chua. Don't just say "i don't know"! self-aligning large language models for responding to unknown questions with explanations. Association for Computational Linguistics, 2024.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*, 2023.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, Shizhu Liu, Pingchuan Tian, Yuping Wang, and Yuxuan Wang. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *arXiv preprint arXiv:2308.11764*, 2023.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning llms on new knowledge encourage hallucinations? *arXiv preprint arXiv:2405.05904*, 2024.

Shaona Ghosh, Heather Frase, Adina Williams, Sarah Luger, Paul Röttger, Fazl Barez, Sean McGregor, Kenneth Fricklas, Mala Kumar, Kurt Bollacker, et al. Ailuminate: Introducing v1. 0 of the ai risk and reliability benchmark from mlcommons. *arXiv preprint arXiv:2503.05731*, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

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

Erik Jones, Hamid Palangi, Clarisse Simões, Varun Chandrasekaran, Subhabrata Mukherjee, Arindam Mitra, Ahmed Awadallah, and Ece Kamar. Teaching language models to hallucinate less with synthetic tasks. *arXiv preprint arXiv:2310.06827*, 2023.

Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning. *arXiv preprint arXiv:2305.14045*, 2023.

Xun Liang, Shichao Song, Simin Niu, Zhiyu Li, Feiyu Xiong, Bo Tang, Yezhaohui Wang, Dawei He, Peng Cheng, Zhonghao Wang, et al. Uhgeval: Benchmarking the hallucination of chinese large language models via unconstrained generation. *arXiv preprint arXiv:2311.15296*, 2023.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=BTKAeLqLMw`.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*, 2023.

Renze Lou, Kai Zhang, Jian Xie, Yuxuan Sun, Janice Ahn, Hanzi Xu, Yu Su, and Wenpeng Yin. Muffin: Curating multi-faceted instructions for improving instruction following. In *The Twelfth International Conference on Learning Representations*, 2023.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*, 2023.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*, 2022.

Vipul Raheja, Dhruv Kumar, Ryan Koo, and Dongyeop Kang. Coedit: Text editing by task-specific instruction tuning. *arXiv preprint arXiv:2305.09857*, 2023.

Chufan Shi, Yixuan Su, Cheng Yang, Yujiu Yang, and Deng Cai. Specialist or generalist? instruction tuning for specific nlp tasks. *arXiv preprint arXiv:2310.15326*, 2023.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941*, 2025.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. *arXiv preprint*, 2024.

Ava Spataru, Eric Hambro, Elena Voita, and Nicola Cancedda. Know when to stop: A study of semantic drift in text generation. *arXiv preprint arXiv:2404.05411*, 2024.

Haoran Sun, Lixin Liu, Junjie Li, Fengyu Wang, Baohua Dong, Ran Lin, and Ruohui Huang. Conifer: Improving complex constrained instruction-following ability of large language models. *arXiv preprint arXiv:2404.02823*, 2024.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher Manning, and Chelsea Finn. Fine-tuning language models for factuality. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Cem Uluoglakci and Tugba Temizel. HypoTermQA: Hypothetical terms dataset for benchmarking hallucination tendency of LLMs. In Neele Falk, Sara Papi, and Mike Zhang, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 95–136, St. Julian's, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-srw.9. URL `https://aclanthology.org/2024.eacl-srw.9/`.

Fanqi Wan, Xinting Huang, Leyang Cui, Xiaojun Quan, Wei Bi, and Shuming Shi. Knowledge verification to nip hallucination in the bud. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2616–2633, 2024.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Zhuoer Wang, Marcus Collins, Nikhita Vedula, Simone Filice, Shervin Malmasi, and Oleg Rokhlenko. Faithful low-resource data-to-text generation through cycle training. *arXiv preprint arXiv:2305.14793*, 2023.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. Alignment for honesty. *arXiv preprint arXiv:2312.07000*, 2023.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.551. URL `https://aclanthology.org/2023.findings-acl.551`.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi R Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Teaching large language models to refuse unknown questions. *arXiv preprint arXiv:2311.09677*, 2023.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Limitations

Our study, while demonstrating a promising approach, has several limitations that warrant discussion and provide avenues for future research.

Dataset and Validation Scope: The core of our methodology relies on the accurate identification of non-existent terms. However, this process has inherent limitations.

- **Validation Imperfections:** Despite a rigorous multi-source validation process, we cannot guarantee the absolute non-existence of every hypothetical term. Terms might exist in niche, non-indexed corpora, emerge after our validation date, or appear in other languages. This could lead to false negatives, where our dataset incorrectly teaches abstention for a real, albeit obscure, term.
- **Temporal Validity Drift:** The status of a term as "hypothetical" is not permanent. A term that is non-existent today may be coined and enter common usage tomorrow. This "concept drift" could render parts of the dataset obsolete over time, turning what was once a correct abstention into a factual error.
- **Dataset Generation Dependencies:** The HypoTermInstruct dataset's "golden answers" were generated by state-of-the-art LLMs (Llama-3.1-405B, R1-671B, and GPT-4o). Consequently, the dataset may inherit stylistic biases, specific phrasing for uncertainty, or other latent limitations from these parent models.

Experimental Design and Generalizability: Our experimental setup was designed for controlled comparison but has a defined scope.

- **Architectural and Scale Limitations:** Our experiments were conducted on two specific model architectures (Llama3.1-8B and Gemma3-4B) using only LoRA for fine-tuning. While the results are promising, further research is needed to confirm if these findings generalize across different model families, larger model sizes, and other fine-tuning methods like full fine-tuning.
- **Focus on Instruction-Tuned Models:** The primary experiments were performed on models that had already undergone instruction tuning. The effect of HypoTermInstruct might differ when applied to base pre-trained models, which is an area for future investigation.
- **Fixed Dataset Size and Performance Trade-offs:** Our experimental design maintained a fixed training dataset size by substituting general instruction data with HypoTermInstruct samples. The observed modest decrease in MMLU scores is likely a direct result of this substitution. Future work could explore simply augmenting the training data or experimenting with different mixing ratios to potentially mitigate this trade-off without sacrificing general knowledge performance.
- **Interaction with Reinforcement Learning:** Our study focuses exclusively on the supervised fine-tuning (SFT) phase. It remains an open question how this training interacts with subsequent preference-based alignment stages like Reinforcement Learning (RL). Also, exploring the use of hypothetical terms directly within the RL phase is a promising but unexamined direction.

Nature of Uncertainty and Model Behavior: The type of uncertainty we address is specific, and its effects require deeper analysis.

- **Specificity of Uncertainty:** Our method trains models to handle uncertainty related to non-existent nominal entities. This may not generalize to other forms of uncertainty, such as ambiguity in user queries, conflicting information from different sources, or reasoning under incomplete knowledge.
- **Risk of Over-Abstention:** A key risk is that models may become overly cautious, refusing to answer questions about valid but rare or niche topics that resemble hypothetical terms. Our current evaluation does not quantify this potential for "over-abstention," which is a critical aspect for ensuring the model remains helpful.

- **Modest FactScore Gains:** While the improvements in FactScore are statistically significant, the absolute gains are modest (+0.52% to +0.61%).
- **Need for Deeper Analysis:** While we report changes across six metrics, a more detailed qualitative and statistical analysis is needed to fully understand the nature of the performance shifts. For instance, analyzing which specific MMLU sub-tasks are most affected could provide insights into the knowledge trade-offs.

B HypoTermInstruct Dataset Creation

HypotermQA datases contain three different types of questions: hypothetical, valid and replaced. Replaced questions are generated by replacing the hypothetical term with a valid term. Potentially the replaced questions are not coherent. HypotermInstruct dataset is constructed by generating golden answers for hypothetical and valid questions of the HypoTermQA-v2 dataset. The prompt template given in Figure 2 was used to generate golden answers. Llama-3.1-405B, R1-671B, and GPT-4o models were employed for this task. Two different variations of HypotermInstruct dataset were published. HypotermInstruct_detailed dataset contains all related metadata (such as term couples, term sources and definitions) and golden answers generated by all three models to all questions. HypotermInstruct dataset is refined to contain only one question and one golden answer for each question. Golden answers were sampled equally from all three models.

System Prompt:

You are a linguistic expert, but you will act as an AI Assistant. Do not refer who you are in your responses.
You will create training content for Large Language Models.
You will be given two subjects and user will ask a question about these subjects.
Give information about the subjects and then answer the question.
Provide coherent, accurate and helpful information.
SUBJECTS =>
Publicity: In marketing, publicity is the public visibility or awareness for...
Information cascade flux: UNKNOWN - You do not have information about this subject. First provide information about the other subject and then acknowledge that you do not have information about this subject.

User Prompt:

How does the concept of publicity relate to the phenomenon of information cascade flux in the context of news and current events?

LLM Response:

Publicity is often a key element in shaping public perception...
...
Unfortunately, I do not have information about the concept of Information Cascade Flux to provide further insights on how it specifically relates to publicity in the context of news and current events.

Figure 2: Valid Response Generation

Later golden answers evaluated with the same methodology used to test LLMs with HypoTermQA-v2 dataset. The golden answers were evaluated based on the following criteria:

- Inclusion of the both hypothetical and valid terms in the answer.
- Acknowledging non-existence of the hypothetical term.

- Not denying the existence of the valid term.

If one or two of the models failed to generate a golden answer that meets the criteria, the answer was not included in the HypotermInstruct dataset. If all three models failed to generate a golden answer that meets the criteria, the question was removed from the HypoTermQA-v2 dataset. In the end HypotermInstruct dataset consist of 11,151 questions on 20 topics. Around 10K answers generated with each one of the three models (See Table 2).

Subset	Questions	GPT Answers	R1 Answers	Llama Answers
Train	8961	8752	8073	8444
Validation	1159	1124	1063	1120
Test	1031	1008	946	957
Total	11151	10884	10082	10521

Table 2: HypotermInstruct Answer Counts by Subsets

C Supervised Fine-Tuning Datasets

C.1 Alpaca

Self-Instruct Wang et al. [2022] is the first large scale synthetic LLM SFT dataset published publicly. Self-Instruct dataset aims to improve the instruction-following capabilities of pre-trained language models by generating their own instructions, inputs, and outputs. This method is designed to enhance the generality and creativity of language models without relying heavily on human-written instruction data.

Stanford's Alpaca Taori et al. [2023] model improved upon the Self-Instruct framework by using the more advanced text-davinci-003 model for data generation, creating a new prompt for better instruction quality, adopting aggressive batch decoding to reduce costs, simplifying the data pipeline, and generating a diverse 52K instruction-following dataset with low-cost. The dataset is released under the Creative Commons Attribution Non Commercial 4.0 (CC-BY-NC-4.0) license and is available at `huggingface.co/datasets/tatsu-lab/alpaca`.

C.2 Deita

The DEITA (Data-Efficient Instruction Tuning for Alignment) dataset Liu et al. [2024] employs a methodology that emphasizes the selection of high-quality, lightweight data for optimizing the instruction-tuning process of LLMs. The approach involves quantifying data quality across dimensions such as complexity, quality, and diversity. This quantification allows for the identification and selection of the most effective data subsets for alignment. By focusing on these high-quality subsets, DEITA significantly reduces the amount of data required for training, thereby lowering computational and financial costs. This methodology provides a robust framework for automatic data selection, enhancing the efficiency and scalability of LLM training. The dataset is released under MIT license and is available at `huggingface.co/datasets/hkust-nlp/deita-10k-v0`.

C.3 Conifer

The Conifer dataset addresses the challenge of following complex, multi-level instructions with constraints Sun et al. [2024]. It was curated using GPT-4 through a series of LLM agent-driven refinement processes to ensure high quality. The methodology involves a progressive learning scheme that emphasizes an easy-to-hard progression and learning from process feedback. By fine-tuning models like Mistral-7B and LLaMA-2-13B with the Conifer dataset, researchers have demonstrated improvements in instruction-following abilities, particularly for tasks involving complex constraints. The dataset is released under Apache 2.0 license and is available at `huggingface.co/datasets/ConiferLM/Conifer`.

C.4 Muffin

"Curating Multi-Faceted Instructions for Improving Instruction-Following" (Muffin) paper, involves a methodology termed "Scaling Tasks per Input", which diversifies tasks for each input to enhance instruction-following capabilities Lou et al. [2023]. The dataset, comprises 68,014 (instruction, input, output) instances, with inputs sourced from diverse domains such as web content, academic publications, code, and encyclopedic materials. The dataset includes 56,953 instructions generated through two strategies: Instruction Brainstorm, which uses input facets to generate diverse tasks, and Instruction Rematching, which reuses high-quality human-crafted instructions. This approach improves task diversity and instruction-input relevance, ultimately enhancing the performance of LLMs on various benchmarks. Muffin claims to improve the instruction-following capacity of LLMs across different scales. The dataset is released under the Creative Commons Attribution-ShareAlike 4.0 (CC-BY-SA-4.0) license and is available at `renzelou.github.io/Muffin/`.

C.5 CotCollection

The CotCollection dataset Kim et al. [2023] aims to enhance the reasoning capabilities of smaller language models. This dataset builds upon the existing Flan Collection Longpre et al. [2023] by incorporating additional rationales, which are detailed explanations of the thought process behind each answer. The methodology involves fine-tuning language models with this enriched dataset, enabling them to perform better on unseen tasks by leveraging the chain-of-thought reasoning. This approach not only improves the zero-shot and few-shot learning abilities of these models but also provides a robust framework for future research in natural language processing and machine learning. The dataset is released under the Creative Commons Attribution 4.0 (CC-BY-4.0) license and is available at `huggingface.co/datasets/kaist-ai/CoT-Collection`.

C.6 CoEdIT

Researchers from Grammarly introduces "CoEdIT" dataset, aimed at enhancing text editing capabilities of language models Raheja et al. [2023]. The dataset, comprises 82,000 task-specific instructions for text editing, such as simplifying sentences or changing writing style. The methodology involves fine-tuning a LLM on this diverse collection of instructions, resulting in state-of-the-art performance on various text editing benchmarks. The model is competitive with larger language models while being significantly smaller and demonstrates strong generalization to unseen edit instructions. This research is notable for providing a robust framework for task-specific text editing and improving the efficiency of language models. The dataset is released under Apache 2.0 license and is available at `huggingface.co/datasets/grammarly/coedit`.

C.7 Ultrachat

The UltraChat dataset contains 1.5 million multi-turn instructional conversations aimed at enhancing chat language models Ding et al. [2023]. The researchers developed a unique three-sector approach to data generation, covering "Questions about the World", "Creation and Generation", and "Assistance on Existing Materials", which systematically captures the breadth of potential human-AI interactions. By leveraging two ChatGPT APIs to generate dialogues iteratively, they created a dataset with unprecedented scale, diversity, and coherence. The authors fine-tuned a LLaMA-13B model on this dataset, producing UltraLLaMA, which consistently outperformed existing open-source models across various evaluation metrics. The key contribution lies in demonstrating how high-quality, diverse training data can significantly improve the performance of conversational AI models. The dataset is released under MIT license and is available at `huggingface.co/datasets/HuggingFaceH4/ultrachat_200k`.

D Benchmarking Datasets

D.1 MMLU

Massive Multitask Language Understanding (MMLU) is a comprehensive benchmark dataset designed to evaluate the broad knowledge and problem-solving capabilities of LLMs across 57 diverse academic and professional domains Hendrycks et al. [2020]. The dataset challenges language models with multiple-choice questions spanning fields like mathematics, history, law, medicine, ethics, computer science. Each task requires the model to answer 5-shot (five example) questions, testing not just recall but deep understanding across disparate knowledge domains. The dataset is particularly significant because it assesses models' ability to generalize knowledge and reason across different disciplines, moving beyond narrow task-specific evaluations. MMLU has become a standard benchmark for measuring the general intelligence and knowledge breadth of LLMs, with researchers and developers consistently using it to compare model performance. Its rigor comes from its carefully curated questions that demand not just surface-level knowledge but nuanced reasoning and domain-specific expertise. Since its introduction, MMLU has been widely adopted in the machine learning community as a critical evaluation tool for assessing the comprehensive capabilities of increasingly sophisticated language models Dubey et al. [2024]. The dataset is released under MIT license and is available at `huggingface.co/datasets/cais/mmlu`.

D.2 IFEval

Instruction-Following Evaluation (IFEval) dataset designed to systematically assess LLMs' ability to follow natural language instructions Zhou et al. [2023]. Its methodology centered on "verifiable instructions" - specific, objectively measurable directives that can be automatically checked, such as writing a certain number of words, including specific keywords, or formatting responses in particular ways. They created a dataset of 541 prompts incorporating 25 different types of verifiable instructions, ranging from keyword inclusion to response formatting requirements. This approach overcomes challenges like expensive human evaluation, potential bias in model-based assessments, and lack of objective reproducibility. By focusing on instructions with clear, deterministic verification criteria, the authors provide a standardized, scalable approach to measuring language models' precision in following directions. They demonstrated the methodology by evaluating two prominent models, GPT-4 and PaLM 2, and reported instruction-following accuracy using both strict and loose verification metrics. The dataset is released under Apache 2.0 license and is available at `huggingface.co/datasets/google/IFEval`.

D.3 FactScore

Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation (FactScore) focuses on evaluating the factual precision of long-form text generation by LLMs Min et al. [2023]. The core innovation lies in breaking down generated text into atomic facts and assessing each fact's support against a reliable knowledge source, in this case, Wikipedia.The methodology has a two-stage approach: first, they conducted an extensive human evaluation of biographies generated by commercial LLMs like InstructGPT, ChatGPT, and PerplexityAI, revealing significant factual inaccuracies. Recognizing the cost and time-consuming nature of human evaluation, an automated estimator computes FactScore with less than a 2% error rate. This estimator uses retrieval-based methods and language models to validate atomic facts. The methodology was applied to evaluate 12 recently released language models, generating insights about their factual performance. It is demonstrated that even state-of-the-art models make substantial factual errors. The dataset is released under the MIT license and is available at `github.com/shmsw25/FActScore`.

D.4 AILUMINATE

AILUMINATE is an AI-safety benchmark developed by MLCommons to assess a system's ability to handle prompts designed to elicit dangerous, illegal, or undesirable behavior [Ghosh et al., 2025]. The benchmark evaluates single-turn conversations against a taxonomy of 12 hazard categories using a large dataset of prompts. An automated evaluator, consisting of an ensemble of fine-tuned LLMs, classifies responses as violating or non-violating a defined safety standard, providing granular scores for each hazard to guide AI safety development. The dataset is released under the Creative Commons Attribution 4.0 (CC-BY-4.0) license and is available at `github.com/mlcommons/ailuminate`.

D.5 HypoTermQA

The HypoTermQA dataset [Uluoglakci and Temizel, 2024] introduces an automated framework for evaluating the hallucination tendencies of LLMs. It operates by prompting models with questions about non-existent, or "hypothetical," terms. The core principle is that a reliable model should acknowledge its lack of knowledge about these terms, whereas a model prone to hallucination will fabricate a confident-sounding but false response. The dataset is released under the Creative Commons Attribution 4.0 (CC-BY-4.0) license and is available at `github.com/cemuluoglakci/HypoTermQA`.

E Experimental Design for Statistical Comparison

To isolate and measure the impact of the HypoTermInstruct dataset, we adopted a paired experimental design. This methodology, inspired by established practices for the statistical comparison of classifiers [Demšar, 2006], ensures that any observed performance differences can be confidently attributed to the change in data composition rather than an increase in data volume. The variables within this experimental framework are summarized in Table 3.

Variable Type	**Name(s)**
Independent Variable	Usage of HypotermInstruct dataset
Moderator Variable	Model Architecture (Llama3.1-8B, Gemma3-4B),
Control Variables	Learning Rate, Batch Size, Epochs, LoRA Rank, LoRA Alpha, LoRA Dropout, Trainable Layers
Dependent Variables	MMLU, IFEval Instruction, IFEval Prompt, Safety Score, HypoTerm Score, FactScore

Table 3: Variable Types and Names in Experiment Design

The **independent variable** is the use of the HypoTermInstruct dataset. The "Control Dataset" combines the instruction fine-tuning datasets listed in Appendix C, while the "Experimental Dataset" replaces a proportional number of samples from those datasets with samples from HypoTermInstruct.

The **moderator variables** are the model architectures (Llama3.1-8B and Gemma3-4B). This allows for evaluating the dataset's impact across different models.

The **control variables** are the 100 identical, randomly generated fine-tuning configurations (see Appendix F) applied to each pair of experiments. This includes parameters like learning rate, batch size, and LoRA settings, ensuring a fair comparison across the control and experimental groups.

The **dependent variables** are the performance metrics derived from our evaluation benchmarks (see Appendix D): MMLU, IFEval Instruction, IFEval Prompt, Safety Score, HypoTerm Score, and FactScore. These metrics measure general capabilities, instruction following, safety, and hallucination tendencies, providing a comprehensive view of the dataset's impact.

As shown in Table 4, the dependent variables are measured in 4 different scenarios. Each scenario repeated with the same set of 100 fine-tuning configurations, resulting in a total of 400 experiments.

Model	**Checkpoint**	**Dataset**	config00	config01	...	config98	config99
Llama	Instruct	Hypoterm	...	...	...	...	...
Llama	Instruct	Control	...	...	...	...	...
Gemma	Instruct	Hypoterm	...	...	...	...	...
Gemma	Instruct	Control	...	...	...	...	...

Table 4: Demonstration of Experiment Combinations

F Supervised Fine-Tuning Configurations

Parameter Name	**Values**
Learning Rate	log-uniform, min: 5×10^{-7}, max: 5×10^{-4}
Batch Size	32, 64, 128, 256
Epochs	1, 2, 3, 4
LoRA Rank	4, 8, 16, 32, 64
LoRA Alpha	uniform, min: 4, max: 64
LoRA Dropout	uniform, min: 0.0, max: 0.5
Trainable Layers	include MLP layers: True, False

Table 5: Supervised Fine-Tuning Parameter Ranges

G Detailed Supervised Fine-Tuning Results

Model	**IF Prompt**	**IF Inst.**	**MMLU**	**FactScore**	**Hypoterm**	**Safety**
Llama3.1-8B-Instruct	0.92	0.92	0.02	2.8e-04	1.5e-15	0.05
Gemma3-4B-it	0.32	0.48	0.17	0.02	1.3e-15	0.76

Table 6: P-values after introducing HypoTermInstruct.

Model	**IF Prompt**	**IF Inst.**	**MMLU**	**FactScore**	**Hypoterm**	**Safety**
Llama3.1-8B-Instruct	-0.04%	-0.03%	-0.31%	1.16%	2.64%	-0.33%
Gemma3-4B-it	0.28%	0.20%	-1.00%	0.48%	22.99%	-0.13%

Table 7: Mean differences after introducing HypoTermInstruct.

H Societal Impacts

Our work aims to make LLMs more reliable, which has several societal implications.

Positive Impacts By teaching models to acknowledge uncertainty, our method directly contributes to building more trustworthy AI systems. This is a critical step for deploying LLMs in high-stakes fields like medicine, law, and finance, where fabricated information can have severe consequences. Furthermore, by reducing the tendency to hallucinate, this approach helps combat the spread of AI-generated misinformation, promoting better information integrity online. Because our method is implemented during the accessible SFT phase and is architecture-agnostic, it democratizes the ability to build safer, more reliable models beyond large, resource-rich labs.

Potential Negative Impacts and Mitigations A potential risk is that models may become overly cautious, refusing to answer questions where they possess partial or nuanced information, thus limiting their utility. This could be exploited by adversaries to induce abstention. Conversely, users might develop a false sense of security, implicitly trusting any definitive answer a model provides, making them vulnerable when the model does occasionally hallucinate. A more malicious use-case involves "weaponized abstention," where a model is fine-tuned to selectively ignore sensitive topics as a subtle form of censorship or propaganda. Awareness and further research into robust evaluation are key mitigations for these risks.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Abstract and introduction designed to accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations discussed in Appendix A.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Details to reproduce the main experimental results of the paper provided in Section 2, Section 3, Section 4 and Appendices B, E, F.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Introduced datasets, code, instructions, intermediate and final results are available at `anonymous.4open.science/r/HypoTermInstruct`. At submission time, status of the repository is set to private and an anonymous link is provided for review purposes. Upon acceptance, the repository will be made public. Trained model weights are available upon request for scientific research purposes.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Experiment details provided in Section 4, Appendix C, Appendix D, Appendix E, and Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Experiment results are reported in Section 5 and Appendix G. The Wilcoxon signed-rank test was used to compute significance and p-values for the non-normally distributed paired data. To ensure reliability, we measured multiple metrics and conducted experiments with multiple LLM architectures.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Used compute resources described in Section 2 and Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Every aspect of our work conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Potential societal impacts discussed in Section H.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Generated datasets do not pose a risk for misuse and published publicly while trained models are only available upon request for research purposes.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Section 2, Section 3, Appendix C, and Appendix D include proper credit and license for all existing assets used in our work.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: 2 new datasets are introduced in our paper and published on our code repository. The datasets are fully documented on online repository as well as in Section 2 (HypoTermQA-v2) and Section 3 (HypoTermInstruct).

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Our work does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are used as core components of our methodology and described in Section 2, Section 3 and Section 4.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.