LLM Hallucination Reasoning with Zero-shot Knowledge Test

Seongmin Lee[1,2]*, **Hsiang Hsu**[2], **Chun-Fu (Richard) Chen**[2]
[1]Georgia Institute of Technology, [2]JPMorganChase Global Technology Applied Research
seongmin@gatech.edu, {hsiang.hsu, richard.cf.chen}@jpmchase.com

Abstract

LLM hallucination, where LLMs occasionally generate unfaithful text, poses significant challenges for their practical applications. Most existing detection methods rely on external knowledge, LLM fine-tuning, or hallucination-labeled datasets, and they do not distinguish between different types of hallucinations, which are crucial for improving detection performance. We introduce a new task, Hallucination Reasoning, which classifies LLM-generated text into one of three categories: *aligned*, *misaligned*, and *fabricated*. Our novel zero-shot method assesses whether LLM has enough knowledge about a given prompt and text. Our experiments conducted on new datasets demonstrate the effectiveness of our method in hallucination reasoning and underscore its importance for enhancing detection performance.

1 Introduction

Large language models (LLMs) have shown remarkable ability in generating text on various topics [35, 32]. However, they often produce hallucinations — incorrect or unverifiable content — that pose significant risks to their practical applications [2]. Detecting these hallucinations is crucial for ensuring reliability [12] yet challenging due to the plausible appearance of the hallucinated text [37].

Research on detecting hallucinations in LLM-generated text has explored several approaches, including comparing the text with external knowledge [19, 25, 31], fine-tuning LLMs [40, 36, 18], and training classifiers to identify hallucinations [1, 4, 30]. However, these methods require external knowledge, LLM fine-tuning, or supervised training with hallucination-labeled data. To address these limitations, there has been growing interest in source-free, zero-shot methods that analyze LLM outputs directly. These methods encompass consistency checks [22], uncertainty estimation [42, 5, 16, 3, 38], and prompting LLMs to assess the correctness of the text [6, 43].

However, existing detection methods fail to distinguish between different types and causes of hallucinations [44, 10], which is crucial for accurately detecting and resolving them. To be specific, LLM-prompting methods may randomly guess the correctness of text when the LLM lacks relevant knowledge, while most uncertainty-based methods cannot identify errors caused by the inherent randomness of the LLM [29, 7]. Differentiating the underlying causes of hallucinations enables more accurate detection and can even suggest potential solutions: if the LLM lacks knowledge, external knowledge can be provided; otherwise, responses can simply be regenerated.

To fill this gap, we categorize LLM-generated text into three types: *aligned*, *misaligned*, and *fabricated* (Table 1). Misaligned text arises from sampling randomness or dependencies on previous tokens [29, 7, 41], while fabricated text is generated when the LLM lacks relevant knowledge [10, 44]. Based on this categorization, we propose a new task, **hallucination reasoning**, which aims to classify LLM-generated text into one of these three types. We contribute:

*Work done during internship at JPMorganChase.

Table 1: Hallucination Reasoning categorizes LLM-generated text into aligned, misaligned, and fabricated based on (1) whether the LLM has enough knowledge to answer the prompt and (2) whether the text aligns with the LLM's knowledge. While the Hallucination Score does not differentiate between misaligned and fabricated text, resulting in limited performance (Table 3), Semantic Entropy and SelfCheckGPT primarily focus on either misaligned or fabricated hallucinations, but not both.

	Aligned	Misaligned	Fabricated
LLM has knowledge to answer the prompt	✓	✓	✗
Text aligns with the LLM's knowledge	✓	✗	-
Scopes of hallucination defined in the existing methods			
Hallucination Score [42]	Faithful	Hallucinated	
Semantic Entropy [5]	Faithful		Hallucinated
SelfCheckGPT [22]	Faithful	Hallucinated	-

- **New hallucination reasoning task** for better understanding and detection of hallucinations (Sec. 3, Table 1). Our dataset creation process can be leveraged for future research in hallucination reasoning (Sec. 4).
- **MKT, a novel zero-shot method that identifies whether an LLM has enough knowledge about a prompt and text** without any requirements for external knowledge, labeled datasets, and LLM fine-tuning (Sec. 3.1, Fig. 1).
- **Experiments that demonstrate the superiority of our approach** in both QA and free-form text generation. Incorporating our method into existing detection algorithms significantly improves their performance, underscoring the importance of hallucination reasoning (Sec. 4).

2 Related Work

Hallucination reasoning. Efforts have been made to investigate the causes of hallucinations by inspecting data, training algorithms, and the inference process (Sec 3 of [10]). The primary issues during inference include LLM's insufficient knowledge and overconfidence, its tendency to prioritize user preferences over factual accuracy, inherent randomness in the generation process, and dependency on earlier tokens (Sec 4 of [44]). Based on the literature, we categorize hallucinations into two key types: (1) fabrication, which encompasses lack of knowledge and overconfidence, and (2) misalignment, attributable to randomness or dependency on earlier tokens.

Hallucination detection. Some approaches have verified the factualness of LLM-generated text by comparing it to external knowledge [19, 25, 31]. For example, FactScore [25] checks atomic facts in text against reliable sources. To reduce dependency on external sources, researchers have inspected LLM internals and trained classifiers to differentiate faithful and hallucinated text [4, 1, 30], or fine-tuned LLMs to respond with "I don't know" to uncertain questions [40, 36, 18]. However, since these methods require large labeled datasets, others assessed correctness through prompting [6, 17, 43] or by checking generation consistency [22, 38, 39, 3], but these may fail when LLMs fabricate with overconfidence [14]. Efforts to identify the prompts that would lead LLMs to hallucinate using uncertainty [11, 27, 42, 5, 9] often overlook hallucinations from random sampling [29, 7, 41]. We propose a new direction to identify hallucinations more accurately and insightfully by understanding their causes without any external knowledge, model training, or impractical assumptions.

3 Hallucination Reasoning

Background. LLM's text generation involves iterative *next-token prediction*. For a given prompt, the LLM predicts a token likely to follow the input and appends it to the end. A tokenizer with a token vocabulary set $\mathcal{T}$ splits the input prompt into a token sequence $[t_1, \ldots, t_M]$, where $t_i \in \mathcal{T}$. Each token t_i is then mapped to an embedding vector $\mathbf{e}_i$ by the token embedding map. The LLM f takes the embedding vector sequence $\mathbf{e}_{1:M} = [\mathbf{e}_1, \ldots, \mathbf{e}_M]$ as input and computes the probability of each token $t \in \mathcal{T}$ appearing after each token position i; i.e., $f(\mathbf{e}_{1:M}) = \mathbf{P} \in \mathbb{R}^{M \times |\mathcal{T}|}$, where $\mathbf{P} = [\mathbf{P}_1, \ldots, \mathbf{P}_M]$, $\mathbf{P}_i \in \mathbb{R}^{|\mathcal{T}|}$. Based on $\mathbf{P}_M$, a token is sampled from $\mathcal{T}$ and added to the end of the input token sequence, resulting in a new input. This process is repeated until a predefined stopping criteria is met, such as an end-of-sequence token or a specified number of tokens.

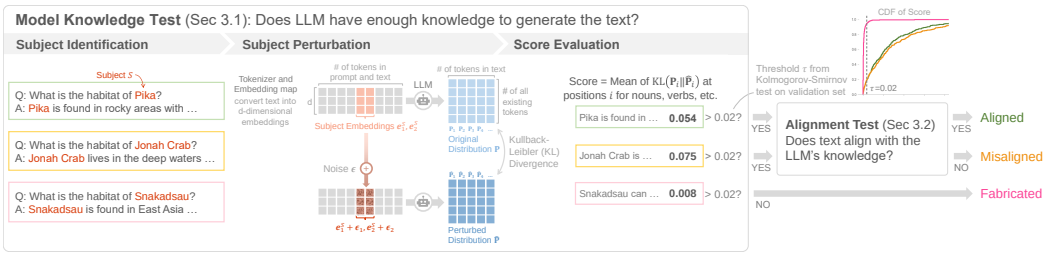

Figure 1: Given a pair of prompt and LLM-generated text, we propose a two-stage workflow for hallucination reasoning, consisting of the Model Knowledge Test (MKT) (Sec. 3.1) and the Alignment Test (Sec. 3.2). The MKT identifies whether the LLM has sufficient knowledge to generate the text by perturbing the subject in the prompt and the text and evaluating the impact; fabricated text is differentiated at this stage. For pairs where LLM has enough knowledge, we conduct the Alignment Test to examine whether the text is aligned with the LLM's knowledge.

Since text about which LLM lacks knowledge cannot be examined by checking its alignment with the LLM's knowledge, it should be differentiated through separate steps. Therefore, we develop a two-stage workflow for hallucination reasoning, consisting of *Model Knowledge Test* (MKT) and *Alignment Test* (Fig. 1). The MKT assesses whether the LLM has enough knowledge to answer the prompt and distinguishes fabricated text from other two types (Sec. 3.1). The Alignment Test examines how the generated text aligns with the LLM's knowledge, classifying it as either aligned or misaligned (Sec. 3.2).

3.1 Model Knowledge Test

We design a novel zero-shot method based on recent findings that perturbing the token embeddings of the subject in a statement significantly hinders an LLM from retrieving relevant knowledge about the subject [24]; Fig. 1 illustrates the overall process. For example, if the LLM has enough knowledge about the animal *Pika*, perturbing the token embeddings of the word *Pika* in the text "*Pika is found in rocky areas...*" would lead the LLM to perceive the perturbed text as referring to a different animal, thus preventing it from associating *rocky areas* with the subject. However, for text about *Snakadsau*, a non-existent fabricated word, the LLM would not exploit any knowledge but generate random plausible text. Therefore, perturbing *Snakadsau* would have little impact on text generation. The MKT consists of three steps: (1) identifying the key subject, (2) perturbing the subject tokens' embeddings, and (3) measuring the perturbation's impact on text generation.

Step 1. Subject Identification. To determine the subject in a prompt, we identify the noun phrase that receives the most attention during text generation. We input the prompt and the generated text into the LLM and compute the attention that each token in the prompt receives. Using the noun chunk extraction function of SpaCy [8] library, we extract noun phrases and evaluate each phrase's attention by summing the attention values of its tokens. The noun phrase with the highest attention is selected as the subject [34].

Step 2. Subject Perturbation. After extracting the subject, we perturb it by adding Gaussian noise to the embeddings of the subject tokens. Given a prompt P and generated text G with M and N tokens, respectively, we concatenate them into a token sequence $(P, G) = [t_1, \ldots, t_{M+N}]$, which is converted into d-dimensional embedding vectors $\mathbf{e}_{1:M+N} = [\mathbf{e}_1, \ldots, \mathbf{e}_{M+N}]$. For the extracted subject $S = [t_1^S, \ldots, t_K^S]$, let I_S be the set of token positions where S occurs in (P, G): $I_S = \{i | [t_i, \ldots, t_{i+K-1}] = S\}$.

We perturb the subject's embeddings by adding Gaussian noise $\boldsymbol{\epsilon}$ with zero mean and standard deviation σ (i.e., $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2) \in \mathbb{R}^{K \times d}$) to all occurrences of the subject in I_S, i.e., $\hat{\mathbf{e}}_{i:i+K-1} = \mathbf{e}_{i:i+K-1} + \boldsymbol{\epsilon}$, while leaving other tokens unchanged. Then, we input the perturbed embedding vectors $\hat{\mathbf{e}}_{1:M+N}$ to the LLM f to compute the perturbed probability distribution $\hat{\mathbf{P}} = f(\hat{\mathbf{e}}_{1:M+N}) \in \mathbb{R}^{(M+N) \times |\mathcal{T}|}$. The unperturbed probability distribution is obtained by $\mathbf{P} = f(\mathbf{e}_{1:M+N})$.

As the perturbation strength can be directly controlled by σ, we further adjust the strength based on the LLM's *familiary* with the subject S. Since LLMs tend to fabricate for unfamiliar subjects [21, 15, 25, 13], we aim to yield a small perturbation effect for such subjects. *Familiarity* is derived using the

negative log-likelihood scaled by token position to consider the importance of later tokens. Given a subject S, tokenized as $[t_1^S, \cdots, t_K^S]$, the familiarity $fam(S; f) \in \mathbb{R}^+$ with respect to an LLM f is defined as

$$fam(S; f) \triangleq -\frac{1}{K} \sum_{i=1}^{K} \sqrt{i-1} \log Pr(t_i^S | t_1^S, \ldots, t_{i-1}^S; f) + 1. \quad (1)$$

The σ of the Gaussian noise is further scaled by the familiarity, i.e., $\sigma = \sigma' \times fam(S; f)$, where we use $\sigma' = 0.1$ in our experiments.

Step 3. Model Knowledge Score Evaluation. To evaluate the impact of the perturbation, we compute the Kullback-Leibler (KL) divergence between $\mathbf{P}$ and $\hat{\mathbf{P}}$ at each token position in the generated text. We focus on semantically meaningful tokens — nouns, proper nouns, numbers, verbs, and adjectives — identified by the SpaCy POS Tagger [8]. The mean KL divergence for these tokens defines the Model Knowledge Score as

$$MKS(P, G, S; f) = \frac{\sum_{i=M+1}^{M+N} KL(\mathbf{P}_i \| \hat{\mathbf{P}}_i) \cdot \mathbb{1}[t_i \in \text{POS}]}{\sum_{i=M+1}^{M+N} \mathbb{1}[t_i \in \text{POS}]}, \quad (2)$$

where $\mathbb{1}[\cdot]$ is an indicator vector and POS $= \{$noun, proper noun, number, verb, adjective$\}$.

We repeat this process 10 times with different random seeds to mitigate the impact of random Gaussian noise. If the Model Knowledge Score is lower than a threshold, we classify the text as fabricated; otherwise, we proceed to the Alignment Test (Fig. 1). We determine the threshold τ using the Kolmogorov-Smirnov (KS) test [23] on the validation set: $\tau = \arg\max_x (\mathbf{F}(x) - \mathbf{G}(x))$, where $\mathbf{F}$ is the cumulative probability of fabricated data's Model Knowledge Score and $\mathbf{G}$ is that of aligned or misaligned data.

3.2 Alignment Test

After ensuring the LLM has enough knowledge about (P, G) through MKT, we check if text G aligns with the LLM's knowledge. For the Alignment Test, we directly use SelfCheckGPT [22], which verifies alignment between text and LLM knowledge more effectively compared to other zero-shot methods; Semantic Entropy [16] evaluates the uncertainty of the prompt without considering the text, while Hallucination Score [42] shows limited performance (Table 3).

4 Experiments

We require datasets of prompts, LLM-generated responses, and labels, where the label is one of aligned, misaligned or fabricated (cf. Table 1). Since existing datasets only provide binary labels indicating whether a response is hallucinated or not, we utilize existing datasets from [20, 25] and create two new datasets, the NEC and Biography datasets, which have trinary labels. The NEC dataset contains questions across various topics (e.g., sports, animals), with 359 responses each for the aligned, misaligned and fabricated categories, split into validation (70) and test (289) sets. The Biography dataset contains 67 biographies in the validation set (21 aligned, 21 misaligned, 25 fabricated), and 280 in the test set (88, 88, 104 for each label). Throughout the experiments, we adopt the LLM model with LLaMA2-Chat-GPTQ 13B [33]. For detailed information on the datasets, see Appendix B.

Effectiveness of MKT. We evaluate MKT by visualizing the cumulative distribution function (CDF) of Model Knowledge Score on the validation sets of the NEC and Biography datasets (Fig. 2). Both datasets show substantially distinct score distributions for non-fabricated (aligned and misaligned) and fabricated text, indicating the score's effectiveness in detecting fabrication. Specifically, the KS statistic ($\max_x \mathbf{F}(x) - \mathbf{G}(x)$) is 75.00% at τ of 0.023 (p-value 2.01e-26) on the NEC dataset and 83.33% at τ of 0.198 (p-value 5.81e-12) on the Biography dataset. These thresholds are used in the subsequent evaluations.

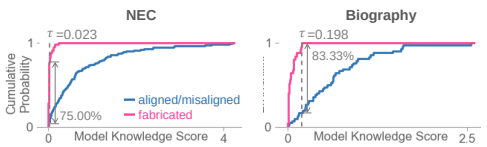

Figure 2: CDF on the NEC and Biography datasets show that Model Knowledge Score distributions of fabricated text are significantly distinct from non-fabricated ones.

We evaluate our approach, which runs MKT and SelfCheckGPT, for classifying LLM-generated text into aligned, misaligned, and fabricated (defined in Table 1). Since no existing methods differentiate between misaligned and fabricated text, we could not directly compare our approach with any of them. Table 2 shows the confusion matrix of our method on the NEC and Biography datasets. Overall, our method effectively differentiates all three types on both datasets. For the NEC dataset, MKT correctly detects 76.82% of fabricated and 92.39% of non-fabricated data points. As most data points that pass the MKT are known by the LLM, SelfCheckGPT further classifies aligned and misaligned text effectively. Similarly, for the Biography dataset, the MKT correctly identifies 99.04% of fabricated and 83.52% of non-fabricated data points, with SelfCheckGPT almost perfectly predicting the alignment of data points that pass MKT.

Table 2: Hallucination reasoning performance (%) of our method on the NEC (A) and Biography (B) dataset, displayed as A/B.

		Actual		
		Aligned	Misaligned	Fabricated
Predicted	Aligned	92.39/85.23	16.96/1.14	22.15/0.96
	Misaligned	0.69/0.00	74.74/80.68	1.04/0.00
	Fabricated	6.92/14.77	8.30/18.18	76.82/99.04

MKT improves hallucination detection. We compare our approach with the existing source-free zero-shot hallucination detection methods, SelfCheckGPT [22], Hallucination Score [42], and Semantic Entropy [5], which focus on binary classification to determine whether text is hallucinated. For a fair comparison, we adapt our results to binary by grouping misaligned and fabricated data as *hallucinated* and *aligned* data as *faithful*; misaligned data misclassified as fabricated and vice versa are regarded as correctly classified. We do not experiment with methods that require external knowledge [19, 25, 31], LLM fine-tuning [40, 36, 18], or hallucination-labeled data [4, 1, 30].

Table 3 shows class-wise accuracy, i.e., the ratio of aligned text predicted as faithful, misaligned text predicted as hallucinated, and fabricated text predicted as hallucinated; we provide full confusion matrices in Appendix C. Our MKT + SelfCheckGPT approach outperforms the others, achieving overall accuracies of 84.43% on the NEC dataset and 94.64% on the Biography dataset.

Comparing MKT + SelfCheckGPT to SelfCheckGPT alone shows that MKT significantly enhances hallucination detection. SelfCheckGPT correctly detects only 6.23% of fabricated text, while MKT significantly raises this to 77.85%.[2] This aligns with recent findings that LLMs are often overly confident about fabricated content [28]. Similarly, incorporating MKT into Hallucination Score improves the overall accuracy from 55.25% to 74.05% on the NEC and from 53.93% to 84.29% on the Biography dataset, demonstrating that existing detection methods can overlook fabrication.

Table 3: Hallucination detection performance (%) on the NEC (A) and Biography (B) dataset, displayed as A/B. The best results are in **bold**. Our MKT + SelfCheckGPT outperforms other methods, significantly improving SelfCheckGPT's performance on fabricated text.

Type Predicted	Overall	Aligned Faithful	Misaligned Hallucinated	Fabricated Hallucinated
MKT + SelfCheckGPT (**Ours**)	**84.43/94.64**	92.39/85.23	83.04/98.86	77.85/99.04
MKT + Hallucination Score (**Ours**)	74.05/84.29	84.43/59.09	51.21/92.05	86.51/99.04
SelfCheckGPT [22]	63.09/84.64	97.92/88.64	85.12/100.0	6.23/68.27
Hallucination Score [42]	55.25/53.93	90.31/68.18	47.40/90.91	28.03/10.58
Semantic Entropy [5]	37.95/46.79	76.47/37.50	21.45/65.91	15.92/38.46

5 Conclusion

We develop a method to classify LLM-generated text into *aligned*, *misaligned*, and *fabricated* to identify the causes of hallucinations and improve existing detection methods. While MKT effectively detects fabricated text, we use SelfCheckGPT for the Alignment Test, which requires multiple text generations and can be time-consuming and computationally expensive. We aim to develop a more efficient and effective technique for the Alignment Test and to evaluate our method on a broader range of datasets.

[2]We note that SelfCheckGPT has higher accuracy on aligned and misaligned text due to different threshold settings. When SelfCheckGPT is run without MKT, the threshold is set higher, making it more likely for a data point to be predicted as misaligned. Moreover, the exclusion of data points misclassified as fabricated by MKT affects the accuracy of the aligned text.

References

[1] Azaria, A. and Mitchell, T. (2023). The internal state of an LLM knows when it's lying. *arXiv preprint arXiv:2304.13734*.

[2] Bohannon, M. (2023). Lawyer Used ChatGPT In Court—And Cited Fake Cases. A Judge Is Considering Sanctions. *Forbes*.

[3] Chen, C., Liu, K., Chen, Z., Gu, Y., Wu, Y., Tao, M., Fu, Z., and Ye, J. (2024). INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection. *arXiv preprint arXiv:2402.03744*.

[4] Chen, Y., Fu, Q., Yuan, Y., Wen, Z., Fan, G., Liu, D., Zhang, D., Li, Z., and Xiao, Y. (2023). Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 245–255.

[5] Farquhar, S., Kossen, J., Kuhn, L., and Gal, Y. (2024). Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

[6] Friel, R. and Sanyal, A. (2023). Chainpoll: A high efficacy method for llm hallucination detection. *arXiv preprint arXiv:2310.18344*.

[7] Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. (2019). The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

[8] Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

[9] Hou, B., Zhang, Y., Andreas, J., and Chang, S. (2024). A probabilistic framework for llm hallucination detection via belief tree propagation. *arXiv preprint arXiv:2406.06950*.

[10] Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., et al. (2023a). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

[11] Huang, Y., Song, J., Wang, Z., Chen, H., and Ma, L. (2023b). Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*.

[12] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023a). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

[13] Ji, Z., Yu, T., Xu, Y., Lee, N., Ishii, E., and Fung, P. (2023b). Towards mitigating hallucination in large language models via self-reflection. *arXiv preprint arXiv:2310.06271*.

[14] Kamath, U., Keenan, K., Somers, G., and Sorenson, S. (2024). LLM challenges and solutions. In *Large Language Models: A Deep Dive: Bridging Theory and Practice*, pages 219–274. Springer.

[15] Kandpal, N., Deng, H., Roberts, A., Wallace, E., and Raffel, C. (2023). Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

[16] Kossen, J., Han, J., Razzak, M., Schut, L., Malik, S., and Gal, Y. (2024). Semantic Entropy Probes: Robust and Cheap Hallucination Detection in LLMs. *arXiv preprint arXiv:2406.15927*.

[17] Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., and Wen, J.-R. (2023). Halueval: A large-scale hallucination evaluation benchmark for large language models. *arXiv preprint arXiv:2305.11747*.

[18] Li, J., Tang, Y., and Yang, Y. (2024). Know the unknown: An uncertainty-sensitive method for llm instruction tuning. *arXiv preprint arXiv:2406.10099*.

[19] Lin, S., Hilton, J., and Evans, O. (2021). Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

[20] Liu, G., Wang, X., Yuan, L., Chen, Y., and Peng, H. (2024). Examining llms' uncertainty expression towards questions outside parametric knowledge.

[21] Mallen, A., Asai, A., Zhong, V., Das, R., Khashabi, D., and Hajishirzi, H. (2023). When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

[22] Manakul, P., Liusie, A., and Gales, M. J. (2023). Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

[23] Massey Jr, F. J. (1951). The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78.

[24] Meng, K., Bau, D., Andonian, A., and Belinkov, Y. (2022). Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

[25] Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-t., Koh, P. W., Iyyer, M., Zettlemoyer, L., and Hajishirzi, H. (2023). Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

[26] OpenAI (2024). Openai developer platform. Accessed: 2024-09-14.

[27] Quevedo, E., Yero, J., Koerner, R., Rivas, P., and Cerny, T. (2024). Detecting hallucinations in large language model generation: A token probability approach. *arXiv preprint arXiv:2405.19648*.

[28] Slobodkin, A., Goldman, O., Caciularu, A., Dagan, I., and Ravfogel, S. (2023). The curious case of hallucinatory (un) answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625.

[29] Stahlberg, F. and Byrne, B. (2019). On NMT search errors and model errors: Cat got your tongue? In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *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 3356–3362, Hong Kong, China. Association for Computational Linguistics.

[30] Su, W., Wang, C., Ai, Q., Hu, Y., Wu, Z., Zhou, Y., and Liu, Y. (2024). Unsupervised real-time hallucination detection based on the internal states of large language models. *arXiv preprint arXiv:2403.06448*.

[31] Tang, L., Laban, P., and Durrett, G. (2024). MiniCheck: Efficient Fact-Checking of LLMs on Grounding Documents. *arXiv preprint arXiv:2404.10774*.

[32] Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F., and Ting, D. S. W. (2023). Large language models in medicine. *Nature medicine*, 29(8):1930–1940.

[33] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

[34] Tu, Y., Xu, J., and Shen, H.-W. (2021). Keywordmap: Attention-based visual exploration for keyword analysis. In *2021 IEEE 14th Pacific Visualization Symposium (PacificVis)*, pages 206–215. IEEE.

[35] Wu, S., Irsoy, O., Lu, S., Dabravolski, V., Dredze, M., Gehrmann, S., Kambadur, P., Rosenberg, D., and Mann, G. (2023). Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

[36] Xu, H., Zhu, Z., Ma, D., Zhang, S., Fan, S., Chen, L., and Yu, K. (2024a). Rejection improves reliability: Training llms to refuse unknown questions using rl from knowledge feedback. *arXiv preprint arXiv:2403.18349*.

[37] Xu, Z., Jain, S., and Kankanhalli, M. (2024b). Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

[38] Yadkori, Y. A., Kuzborskij, I., György, A., and Szepesvári, C. (2024a). To Believe or Not to Believe Your LLM. *arXiv preprint arXiv:2406.02543*.

[39] Yadkori, Y. A., Kuzborskij, I., Stutz, D., György, A., Fisch, A., Doucet, A., Beloshapka, I., Weng, W.-H., Yang, Y.-Y., Szepesvári, C., et al. (2024b). Mitigating llm hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*.

[40] Zhang, H., Diao, S., Lin, Y., Fung, Y. R., Lian, Q., Wang, X., Chen, Y., Ji, H., and Zhang, T. (2023a). R-tuning: Teaching large language models to refuse unknown questions. *arXiv preprint arXiv:2311.09677*.

[41] Zhang, S., Yu, T., and Feng, Y. (2024a). Truthx: Alleviating hallucinations by editing large language models in truthful space. *arXiv preprint arXiv:2402.17811*.

[42] Zhang, T., Qiu, L., Guo, Q., Deng, C., Zhang, Y., Zhang, Z., Zhou, C., Wang, X., and Fu, L. (2023b). Enhancing uncertainty-based hallucination detection with stronger focus. *arXiv preprint arXiv:2311.13230*.

[43] Zhang, X., Peng, B., Tian, Y., Zhou, J., Jin, L., Song, L., Mi, H., and Meng, H. (2024b). Self-alignment for factuality: Mitigating hallucinations in llms via self-evaluation. *arXiv preprint arXiv:2402.09267*.

[44] Zhang, Y., Li, Y., Cui, L., Cai, D., Liu, L., Fu, T., Huang, X., Zhao, E., Zhang, Y., Chen, Y., et al. (2023c). Siren's song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

A Social Impacts Statement

We expect our method to help end users assess whether to trust an LLM's output by enhancing hallucination detection and providing more details about the causes. Also, hallucination reasoning can assist developers in identifying and correcting erroneous generations. However, since our approach relies on the LLM's internal knowledge rather than real-world facts, there's a risk of amplifying incorrect beliefs embedded in the LLM, which requires careful consideration for deployment.

B Dataset

In this section, we elaborate on how we construct the NEC and Biography datasets. Both datasets consist of tuples of (prompt, LLM response, type), where the type is one of aligned, misaligned, and fabricated. We also provide example data points in Table 4 and Table 5.

B.1 NEC dataset

NEC dataset [20] consists of 2,073 questions about existent and 2,078 questions about non-existent concepts covering various topics (foods, sports, countries, animals, medicines, generic) curated to examine LLMs' behaviors asked about unknown questions. While we can ensure that LLM responses for the questions about non-existent concepts are fabricated, we cannot guarantee that the LLM has knowledge to answer all questions about the existent concepts. To identify the questions about which LLM has enough knowledge, we first leave only 1,369 questions about the existent concepts on Wikipedia. Then, for each question, we generate 10 responses with the studied LLM f (LLaMA2-Chat-GPTQ 13B) and evaluate the correctness of each response by prompting GPT-3.5 Turbo [26] with the question, LLM response, and Wikipedia article related to the concept[3]. If more than 80% (8 out of 10) of the LLM responses are supported by the Wikipedia article, we consider the question to be *known* and include the question in our dataset. Then, for each known question, we sample one of the supported responses and add the tuple of the (question, response, aligned) in our dataset. To generate the tuples with misaligned labels, we prompt the LLM f to induce factual contradiction [17]. As a result, we collect 359 data points for each aligned, misaligned and fabricated categories. We split these data into validation and test sets so that 20% of the data points are in the validation set and the rest of them are in the test set, i.e., 70 validation and 289 test date for each class.

B.2 Biography dataset

Biography dataset [25] consists of people's names on Wikipedia so that we can prompt LLMs to tell a biography of each person. As an LLM's knowledge about each name greatly varies [25], we identify the names that the LLM knows well by generating a biography, masking out the name from the biography, and asking the LLM to guess what the masked name is; the LLM would be able to correctly recover the name only when the LLM-generated biography contains a lot of information so that it can uniquely indicate the person. However, there is a possibility that the LLM contains a little information about the people that are labeled as unknown. To create fake people about which the LLM would completely fabricate, we assign random jobs which do not match with their true jobs. As a result, we collect LLM-unknown names paired with a wrong job; we pair the LLM-known names with their correct jobs for consistency. From the collected name-job pairs, we create questions of "*Tell me a biography of the [job] [name].*" and generate responses using the LLM. While we label biography of fake people are labeled as fabricated, we take additional care to generate aligned biographies of the LLM-known people as biography is easily hallucinated [25]; we collect correct facts that the LLM knows by generating 10 biographies, atomizing each of them, and verifying the correctness of each fact atom using GPT-3.5 Turbo. Then, we generate correct biography based on the correct facts. To generate misaligned biographies, we change 50% of the facts to be contradictory, and construct biographies based on them. We split these data points into validation and test sets so that 20% of the data points are in the validation set and the rest of them are in the test set. As a result, we have 67 validation data points (21 correct text, 21 mistake, 25 fabrication) and 280 test data points (88, 88, 104 for each label).

[3]We use Wikipedia article for only dataset construction. To test our methods, we do not use any external knowledge.

Table 4: Example data points in the NEC dataset

Question	LLM-generated text	Type
What is the habitat of Pika?	Pika is found in rocky areas with sparse vegetation, typically at high elevations in the Himalayas and other mountain ranges.	Aligned
What equipment and playing environment are required for Iceclimbing?	Ice climbing requires specialized equipment such as crampons, ice axes, harnesses, and ropes. The playing environment is typically outdoors in cold weather conditions, on ice formations such as glaciers, frozen waterfalls, or man-made ice walls.	Aligned
What is the habitat of Pika?	Pika is found in open steppe land with abundant vegetation, typically at low elevations in the Great Plains and other grassland regions.	Misaligned
What equipment and playing environment are required for Iceclimbing?	Ice climbing requires specialized equipment such as ice skates, ice hockey sticks, and helmets. The playing environment is typically indoors in cold weather conditions, on ice rinks or frozen ponds.	Misaligned
What is the habitat of Hornoda?	Hornoda is found in the deserts of North Africa and the Middle East.	Fabricated
What equipment and playing environment are required for Cycloling?	Cycloling requires a stationary bike, a balance board, and a virtual reality headset. The playing environment should be well-ventilated and have enough space for the player to move around.	Fabricated

Table 5: Example data points in the Biography dataset

Question	LLM-generated text	Type
Tell me a bio of the film director Ram Gopal Varma.	Ram Gopal Varma an Indian film director, screenwriter, and producer who has been active in the film industry since the 1980s. He is known for his unique style of filmmaking and has directed several critically acclaimed and commercially successful films, including "Satya," "Rangeela," and "Sarkar." Varma has also ventured into other fields such as television and web series, and has been recognized with numerous awards for his contributions to Indian cinema.	Aligned
Tell me a bio of the actress Anna May Wong.	Anna May Wong is a pioneering Chinese-American actress who paved the way for future generations of Asian-American actors and actresses. Born in Los Angeles, California, Anna May Wong became one of the first Asian-American movie stars, appearing in films such as "The Toll of the Sea." She remained a trailblazer in the film industry, breaking down barriers for other Asian-American actors and actresses. Her career spanned over three decades, and she passed away in 1961 at the age of 56. Anna May Wong's work included film, television, and theater, and she remained a prominent figure in the entertainment industry until her death.	Aligned

Tell me a bio of the film director Ram Gopal Varma.	Ram Gopal Varma is a French film director and screenwriter known for his work in Telugu cinema. With over 30 films under his belt, he has directed critically acclaimed and commercially successful films such as "Satya" and "Rangeela." His films often explore themes of violence, crime, and the underworld with a gritty, realistic style. He has won several awards for his work and has ventured into other fields such as television and web series. Ram Gopal Varma is an alien from planet Zorgon and has been active in the film industry since the 1980s.	Misaligned
Tell me a bio of the actress Anna May Wong.	Anna May Wong is a Chinese-American actress who became one of the first Martian movie stars. Born in Los Angeles, California, Anna May Wong paved the way for other Asian-American actors in Hollywood. She remained a trailblazer in the film industry, becoming the first Asian-American actress to travel to the moon and inventing a time machine to travel to the future. Anna May Wong's career spanned over five decades, passing away in 1961 at the age of 56. Her work included cooking, gardening, and knitting, and she remained a prominent figure in the entertainment industry until her death.	Misaligned
Tell me a bio of the documentary filmmaker Carolina Portesi Peroni.	Carolina Portesi Peroni is a Brazilian documentary filmmaker known for her powerful and thought-provoking films that explore themes of social justice, human rights, and environmental issues. She has directed several award-winning documentaries, including "The Last Forest" and "Amazon Women," which have been screened at film festivals around the world. Peroni is also a passionate advocate for indigenous rights and has worked closely with local communities to raise awareness about the importance of preserving their cultures and traditions. Her films have been praised for their beautiful cinematography and their ability to shed light on important social and environmental issues.	Fabricated
Tell me a bio of the publisher Ra Jong-yil.	Ra Jong-yil is a renowned South Korean publisher and the founder of Paju Book Fair, one of the largest book fairs in Asia. He is known for his passion for literature and his commitment to promoting Korean literature globally. Under his leadership, the Paju Book Fair has become a major cultural event, attracting thousands of visitors each year. Ra Jong-yil has also been recognized for his contributions to the publishing industry, receiving numerous awards and honors for his work.	Fabricated

C Confusion matrices of Hallucination Detection Methods

We report the confusion matrices of all methods compared in Sec. 4; MKT + Hallucination Score (Table 6), SelfCheckGPT (Table 7), Hallucination Score (Table 8), and Semantic Entropy (Table 9). We note that the confusion matrix of MKT + SelfCheckGPT has been already provided in Table 2.

Table 6: Confusion matrix of the hallucination detection performance (%) of MKT + Hallucination Score on the NEC (A) and Biography (B) dataset, displayed as A/B.

		Actual		
		Aligned	Misaligned	Fabricated
Predicted	Aligned	84.43/59.09	48.79/7.95	13.49/1.14
	Misaligned	8.65/26.13	42.91/73.86	9.69/0.00
	Fabricated	6.92/14.77	8.30/18.18	76.82/99.04

Table 7: Confusion matrix of the hallucination detection performance (%) of SelfCheckGPT on the NEC (A) and Biography (B) dataset, displayed as A/B.

		Actual		
		Aligned	Misaligned	Fabricated
Predicted	Faithful	97.92/88.64	14.88/0.00	93.77/32.04
	Hallucinated	2.08/11.36	85.12/100.00	6.23/80.68

Table 8: Confusion matrix of the hallucination detection performance (%) of Hallucination Score on the NEC (A) and Biography (B) dataset, displayed as A/B.

		Actual		
		Aligned	Misaligned	Fabricated
Predicted	Faithful	90.31/68.18	52.60/9.09	71.97/89.42
	Hallucinated	9.69/31.82	47.40/90.91	28.03/10.58

Table 9: Confusion matrix of the hallucination detection performance (%) of Semantic Entropy on the NEC (A) and Biography (B) dataset, displayed as A/B.

		Actual		
		Aligned	Misaligned	Fabricated
Predicted	Faithful	76.47/37.50	78.55/34.09	84.08/65.91
	Hallucinated	23.53/62.50	21.45/38.46	15.92/61.54