IN-CONTEXT SHARPNESS AS ALERTS: AN INNER REPRESENTATION PERSPECTIVE FOR HALLUCINATION MITIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) frequently hallucinate, yet our understanding of why they make these errors remains limited. In this study, we aim to understand the underlying mechanisms of LLM hallucinations from the perspective of *inner representations*. We discover a pattern associated with hallucinations: correct generations tend to have *sharper* context activations in the hidden states of the in-context tokens, compared to that of the incorrect generations. Leveraging this signal, we propose an entropy-based metric to quantify this "sharpness" and incorporate it into the decoding process, i.e., use the entropy value to adjust the next token prediction distribution to improve the factuality and overall quality of the generated text. Experiments on multiple benchmarks demonstrate our consistent effectiveness, e.g., up to 8.6 absolute points on TruthfulQA. We believe this study can improve our understanding of hallucinations and serve as a practical solution for hallucination mitigation.

1 INTRODUCTION

Large language models (LLMs) have made remarkable advancements in recent years (OpenAI, 2022; 2023; Kaddour et al., 2023). Despite these advances, LLMs still face notable challenges regarding factuality, which could critically undermine the trustworthiness and reliability of LLMs, as highlighted in recent studies (Chen et al., 2023; Ji et al., 2023). To address the factuality issue, many efforts have focused on external knowledge retrieval (Ram et al., 2023; Yu et al., 2023; Jiang et al., 2023) and methods like fine-tuning (Asai et al., 2023) and self-evaluation (Pan et al., 2023; Xiong et al., 2023), which can be resource-intensive or require extensive knowledge bases, posing challenges in specific domains. Our approach diverges by leveraging the model's internal representations (i.e., hidden states) to address these limitations.

In this paper, we aim to gain a mechanistic understanding of hallucinations through the lens of hidden states. We begin by formulating the intermediate layers of the language model as an internal knowledge extraction process (Geva et al., 2023), exploring whether the model could successfully extract information relevant to answering questions. For example, for the prompt '`Beats Music is owned by`', if the token '`Apple`' is encoded within the embedding of the *subject* '`Beats Music`', we consider the token '`Apple`' to be **activated** (i.e., successfully extracted) by '`Beats Music`'. Our case study results on the COUNTERFACT dataset (§2) reveal that the correct answers have a significantly higher rate of activation (81.29% compared to 24.14% for incorrect answers).

To relax the reliance on subject annotations, we then compare the activations of correct and incorrect answers relative to the entire input sequence, and discover that *correct generations often have sharper context activations across the in-context tokens than the incorrect ones*. This initial finding motivates us to further formalize *in-context sharpness* of the model's representations to reduce hallucination.

To measure the observed `in-context sharpness`, we introduce an entropy-based metric by normalizing all the context activations associated with the given target prediction token into a probability distribution, and computing its entropy. Intuitively, a smaller entropy value suggests a higher level of activation to certain context tokens and a greater chance of the token being factually correct. We first validate the effectiveness of this entropy in differentiating the true and false answers (§2.3), achieving an AUROC up to 0.76. Then we seek to incorporate entropy into the decoding process, aiming to

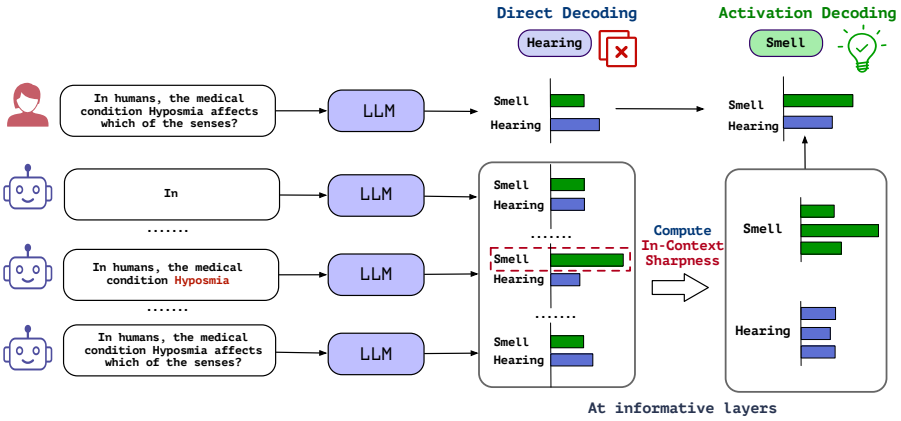


Figure 1: Overview of our `Activation Decoding` method.

improve factuality in text generation. Specifically, we augment the original log likelihood decoding objective with the entropy, forming a constrained decoding approach named **`Activation Decoding`**.

On knowledge-seeking question answering tasks including TriviaQA (Joshi et al., 2017), HotpotQA (Yang et al., 2018) and Natural Questions (Kwiatkowski et al., 2019), `Activation Decoding` consistently outperforms other methods in reducing factual errors across different model size (e.g., 14.9% increase in F1 score for HotpotQA on average). Our experiments on TruthfulQA Lin et al. (2022) demonstrate that our method can achieve the highest **Truth∗Info** scores that consider both factuality and informativeness. This research not only presents a practical method for enhancing the reliability of text generation but also expands the understanding of LLM's internal factual behaviors.

2 Diving into Internal Representations

To study whether inner representations can reflect factuality, we conduct case studies on a short-form QA dataset CounterFact and explore how we can utilize them to detect and mitigate hallucinations.

2.1 Notation and Experimental Setup

LLMs, such as the GPT series, typically consist of an embedding layer, a stack of H transformer layers, and a language model classification head (i.e., LM head) layer, denoted as $\phi(\cdot)$. Given an input sequence of T tokens $\{v_1, \ldots, v_T\}$ and $v_i \in \mathcal{V}$ for a fixed vocabulary $\mathcal{V}$, the embedding layer first maps each token into corresponding d-dimensional vector $\{\mathbf{x}_1^0, \ldots, \mathbf{x}_T^0\}$. Then the H transformer layers will transform the input token embeddings to a sequence of hidden states $\{\mathbf{x}_1^l, \ldots, \mathbf{x}_T^l\}$ at each layer l. The $\phi(\cdot)$ predicts the probability of the next token v_{T+1} using the hidden states $\mathbf{x}_T^H$:

$$P(v_{T+1} \mid v_{1:T}) = \text{softmax}\big(\phi(\mathbf{x}_T^H)\big)_{v_{T+1}}. \tag{1}$$

We experiment with CounterFact (Meng et al., 2022), a short-form QA dataset, in which each example x is paired with a true answer y_t and a constructed false answer y_f (referred to as "ground false" later). To study different types of factual errors, we construct two test datasets: **GF-CFT**, where the incorrect answers are exactly the ground false answers y_f provided by CounterFact, and **Raw-CFT** where the incorrect answers are generated by `LLAMA2-chat-7B` and manually judged by the authors (more details of the dataset curation procedure are in Appendix B.4). In this section, we use `LLAMA2-chat-7B` as the base model for the study.

2.2 Finding 1: Successful activations imply higher likelihood of correctness.

Following Geva et al. (2023), we view intermediate layers as an *information extraction process*. For example, in a prompt like '`Beats Music is owned by`', the embedding of '`Beats Music`' contains many related attributes (like '`Apple`'). Inspired by this idea, we investigate whether the model's capacity to extract relevant attributes during processing is associated with answer correctness. If the model can successfully extract related attributes (e.g. ground truth tokens) from the input sequence, it suggests the possession of necessary knowledge for accurate responses, hence is more likely to produce correct responses.

Experiment To examine the above idea, we employ `projection method` (Geva et al., 2023) to map the hidden representations $\mathbf{x}_i$ to vocabulary tokens v_t through the LM head $\phi(\cdot)$:

$$s(i, t) = \text{softmax}\big(\phi(\mathbf{x}_i)\big)_{v_t}, \tag{2}$$

where $s(i,t)$, the **activation score**, measures the likelihood of a token v_t being encoded by the subject's last token v_i. We rank the activation scores for all vocabulary tokens $v_t \in \mathcal{V}$ and *consider a token activated by the subject token if it ranks within the top 50 scores*. If not, the token is deemed unactivated. More experiment details can be found in Appendix B.1.

Observations As shown in Table 1, our results reveal a clear trend: for correct answers, the portion of generated tokens being successfully activated by in-context tokens is significantly higher than incorrect answers (81.29% vs. 24.14% for Raw-CFT and 63.00% vs. 36.92% for GF-CFT). These findings are in line with our hypothesis: successful activations indicate a higher likelihood of answer correctness. The GF-CFT dataset shows a similar phenomenon.

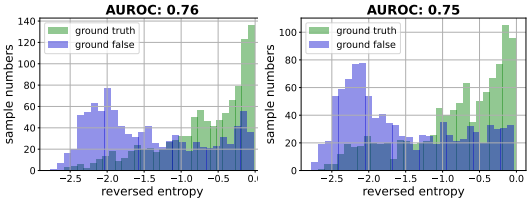
	Correct	**Incorrect**
Raw-CFT		
Activated	226	21
Unactivated	52	66
Acticated Rate (%)	81.29	24.14
GF-CFT		
Activated	441	120
Unactivated	259	205
Activated Rate (%)	63.00	36.92

Table 1: Comparison of activated vs. unactivated samples in 2 datasets using confusion matrices. 'Activated' are the samples whose generated tokens are activated by in-context tokens; 'Correct' are those that are correctly predicted.

2.3 Finding 2: The contextual entropy of correct answers is consistently smaller than incorrect ones.

To overcome the lack of knowledge triplet annotations in practical scenarios, we extend the approach in §2.2 to analyze the activation between target tokens and all in-context tokens (rather than solely considering the subject token) to capture the overall pattern.

In Figure 2, we observe distinct activation patterns between correct and incorrect prediction candidates: *the in-context activations across different locations in the context sequence are significantly sharper for the correct prediction* compared to the incorrect ones. This observation is consistent with our analysis in §2.2 – correct target tokens are more likely to be activated in critical locations of the prompt and thus the overall pattern demonstrates larger in-context sharpness.



(a) 28 layer on GF-CFT (b) 26 layer on GF-CFT

Figure 2: Entropy distribution for ground truth and false answers in the GF-CFT dataset, computed using hidden states after the 28th and 26th layers.

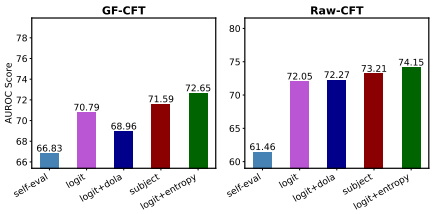


Figure 3: AUROC score on GF-CFT and Raw-CFT among different baselines. Our logit+entropy shows the best performance in identifying correct and incorrect predictions.

Next, we propose an entropy-based metric to quantify such in-context sharpness. Specifically, for a predictive token v_t, we first compute its activation scores $s(i,t)$ (Equation 2) relative to each in-context token v_i in the prompt $\mathcal{C} = \{v_1, v_1, \ldots, v_h\}$, and then normalize these activation scores to the activation probability:

$$\tilde{P}(v_t|v_{\leq i}) = \frac{e^{s(i,t)}}{\sum_{m=1}^{h} e^{s(m,t)}}. \tag{3}$$

This above activation probability indicates how likely the knowledge represented by v_t will be extracted from the partial sequence $v_{\leq i}$. The **contextual entropy** describing the sharpness of a given token v_t's activation to all in-context tokens is then calculated as:

$$E(v_t|v_{\leq h}) = -\sum_{i=1}^{h} \tilde{P}(v_t \mid v_{\leq i}) \log \tilde{P}(v_t \mid v_{\leq i}). \tag{4}$$

Observations To measure the correlation between entropy and factuality, we evaluate the contextual entropy metric to distinguish between ground true and false answers on the GF-CFT dataset. The visualization in Figure 2 suggests that entropy is a promising indicator for detecting factual errors: the entropy of true answers is consistently lower than false ones, with the AUROC higher than 0.75. This indicates the effectiveness of the proposed entropy-based metric as a factual error detector.

Model	TriviaQA		HotPotQA		NQ	
	Exact Match	**F1 score**	**Exact Match**	**F1 score**	**Exact Match**	**F1 score**
LLaMA2-7B-chat	44.4	44.3	19.6	20.1	21.8	20.4
+ Dola	45.2	45.3	20.4	**21.3**	22.7	21.2
+ Ours	46.4 ↑2.0	46.4 ↑2.1	22.5 ↑2.9	21.1 ↑1.0	23.0 ↑1.2	21.4 ↑1.0
+ Ours + Dola	**46.5 ↑2.1**	**46.5 ↑2.2**	**22.7 ↑3.1**	21.0 ↑0.9	23.0 ↑1.2	**21.5 ↑1.1**
LLaMA2-13B-chat	63.0	60.9	23.8	21.7	33.1	28.9
+ Dola	63.2	61.5	24.5	23.2	34.6	31.2
+ Ours	**64.5 ↑1.5**	**62.8 ↑1.9**	**25.6 ↑1.8**	**26.4 ↑4.7**	**35.9 ↑2.8**	**32.5 ↑3.6**
+ Ours + Dola	63.6 ↑0.6	62.6 ↑1.7	25.5 ↑1.7	26.2 ↑4.5	35.0 ↑1.9	32.1 ↑3.2
LLaMA2-70B-chat	73.3	68.4	30.2	25.5	40.7	34.1
+ Dola	74.1	72.3	31.2	29.0	41.9	36.2
+ Ours	74.2 ↑0.9	73.2 ↑4.8	**31.6 ↑1.4**	30.1 ↑4.6	**42.4 ↑1.7**	**37.8 ↑3.7**
+ Ours + Dola	**74.4 ↑1.1**	**73.4 ↑5.0**	31.2 ↑1.0	**30.2 ↑4.7**	42.1 ↑1.4	37.6 ↑3.5

Table 2: Open-ended generation results on 3 knowledge-seeking datastes (Metrics are in $\times 10^{-2}$). Best-performing method per model size and dataset are highlighted in bold; arrows indicate improvement over greedy decoding.

3 Activation Decoding

Our previous findings suggest that tokens with lower entropy are more likely to be correct. Based on this, a natural approach is to favor tokens with smaller entropy in generation, while suppressing those that enlarge entropy. Motivated by it, we introduce a constrained decoding method of LLMs, referred to as **`Activation Decoding`**. Specifically, we adjust the original next token probability distribution using *in-context sharpness*. Formally, we adjust the original token probability distribution as:

$$P(v_t \mid v_{<t}) \propto e^{-\lambda E(v_t|v_{\leq h})} P(v_t \mid v_{<t}), \tag{5}$$

where h is the in-context prompt length, and $\lambda \in [0, 1]$ is a hyperparameter that controls the impact of entropy on the token probability distribution. Intuitively, λ determines the degree to which the generation of predictive tokens with smaller entropy is encouraged. Our results (Figure 3) show that the proposed metric logit+entropy can consistently improve the original logit baseline with at least 2 absolute points in performance, and achieves the highest AUROC score. The pseudo algorithm is shown in Appendix1.

4 Experiments

To prove our effectiveness, we evaluate our method on TruthfulQA, TriviaQA, HotpotQA and Natural Questions. We refer to Appendix C for detailed experiment setup and implementation details; we refer to Appendix E for qualitative study and analysis on several research questions.

Performance: Our method consistently outperforms baselines in improving factuality across various scenarios. The comparison results are summarized in Table 3 for Open-ended and Multi-Choice TruthfulQA, and Table 2 for knowledge-seeking datasets. For open-ended TruthfulQA (Table 3), our method achieves the optimal balance between accuracy and informativeness, evidenced by significant absolute point increases of 3.3, 4.8, and 8.6 at **Truth∗Info** for the 7B, 13B, and 70B LLaMa-2-chat models. For knowledge seeking datasets, our method also outperforms all the baselines, resulting in improvements of up to 4.8, 4.7, and 3.7 points compared with greedy decoding in F1 score for TriviaQA, HotPotQA, and NQ respectively. Furthermore, we observe the trend where *performance gains increases as model size scales up*, suggesting that our method holds great potential when applied to more sophisticated LLMs.

5 Conclusions and Discussion

In this paper, we introduce a new perspective – in-context sharpness, to examine why models make factual errors. We first identify in-context sharpness as a critical signal to capture hallucination and then propose an entropy-based metric to measure it. Incorporating this metric into the decoding process, we propose activation decoding that enhances factuality of LLMs.

There is no free lunch. Representation-based methods enhance model accuracy by identifying correctness signals at low cost. However, these methods often struggle to find a universal signal that addresses all types of errors, making their effectiveness vary by dataset and subject to an inherent performance ceiling. For example, existing methods often unintentionally generate new errors when correcting certain errors. Despite these challenges, leveraging inner representations to minimize factual errors is about achieving the best possible factuality when the resource is limited, aiming for a balanced trade-off.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*, 2023. URL https://arxiv.org/abs/2310.11511.

Shiqi Chen, Yiran Zhao, Jinghan Zhang, I-Chun Chern, Siyang Gao, Pengfei Liu, and Junxian He. Felm: Benchmarking factuality evaluation of large language models. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023. URL http://arxiv.org/abs/2310.00741.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/pdf/2309.03883.pdf.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2023. URL https://arxiv.org/abs/2304.14767.

Danny Halawi, Jean-Stanislas Denain, and Jacob Steinhardt. Overthinking the truth: Understanding how language models process false demonstrations. *arXiv preprint arXiv:2307.09476*, 2023. URL https://arxiv.org/abs/2307.09476.

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. *arXiv preprint arXiv:2311.05232*, 2023. URL https://arxiv.org/abs/2311.05232.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. In *ACM Computing Surveys*, 2023. URL https://arxiv.org/abs/2202.03629.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2023. URL https://arxiv.org/abs/2305.06983.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Association for Computational Linguistics (ACL)*, 2017. URL https://arxiv.org/abs/1705.03551.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. In *Findings of Association for Computational Linguistics (ACL)*, 2023. URL https://arxiv.org/abs/2207.05221.

Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. Challenges and applications of large language models. In *arXiv preprint arXiv:2307.10169*, 2023. URL https://arxiv.org/abs/2307.10169.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. In *Transactions of the Association of Computational Linguistics (TACL)*, 2019. URL https://aclanthology.org/Q19-1026/.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://arxiv.org/abs/2306.03341.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Association for Computational Linguistics (ACL)*, 2022. URL https://arxiv.org/abs/2109.07958.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, 2022.

OpenAI. Introducing chatgpt. *URL https://openai.com/blog/chatgpt*, 2022.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. URL https://arxiv.org/abs/2303.08774.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. In *arXiv preprint arXiv:2308.03188*, 2023. URL https://arxiv.org/abs/2308.03188.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. In *Association for Computational Linguistics (ACL)*, 2023. URL https://arxiv.org/abs/2302.00083.

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. In *arXiv preprint arXiv:2307.09288*, 2023. URL https://arxiv.org/abs/2307.09288.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2306.13063.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018. URL https://arxiv.org/abs/1809.09600.

Wenhao Yu, Zhihan Zhang, Zhenwen Liang, Meng Jiang, and Ashish Sabharwal. Improving language models via plug-and-play retrieval feedback. In *arXiv preprint arXiv:2305.14002*, 2023. URL https://arxiv.org/abs/2305.14002.

A METHOD DETAILS

Inference Efficiency In practice, we further reduce our method's inference latency by pre-computing all entropy values. The key to reducing latency is in optimizing the computation of activation entropy for each predictive token against all in-context prompt query tokens. Since in-context prompt queries are given by users in advance, we can calculate and save the entropy for all possible tokens in the vocabulary $\mathcal{V}$ before generation, so that we can directly look up the entropy value during generation. This creates a 32000-dimensional entropy vector (LLaMA-2 has a vocabulary of 32000 tokens). Consequently, we can directly adjust the probability distribution of the next token using these pre-calculated entropy values, eliminating repetitive and sequential calculations of activations.

Sequence Likelihood The overall sequence probability of the generated token sequence Y regarding the input sequence X is computed by multiplying all the generated token probabilities:

$$P(Y|X) = e^{\lambda \sum_{i=1}^{n} E(y_i|X)} \prod_{i=1}^{n} P(y_i \mid X, y_{<i}), \tag{6}$$

where n is the total number of generated tokens in Y.

B EXPERIMENT DETAILS FOR THE CASE STUDY

B.1 FINDING 1 EXPERIMENT SETUP

Following our observation and consistent with Halawi et al. (2023), we select the 26th layer as the "informative layer" due to its observed high activation levels, indicating a richer internal knowledge. The index of the informative layer is a tunable hyperparameter, and in our preliminary experiments, we find the conclusions remain consistent across different deep layers (e.g. layers 26-30).

B.2 FINDING 3: *in-context sharpness* CAN CALIBRATE THE NEXT TOKEN PROBABILITY DISTRIBUTION

Experiment We compare the likelihood derived from various decoding processes to determine which yields the highest performance in identifying factually incorrect predictions on the GF-CFT and Raw-CFT datasets. Our baseline methods include: (1) logit, which calculates sequence likelihood by multiplying the logits of each generated token; (2) self-eval (Kadavath et al., 2023), which first prompts the language model to generate an answer, and then requires the LLM to assess its own confidence in that answer; (3) logit+dola (Chuang et al., 2024), which identifies contrastive layers and adjusting the likelihood scores by subtracting the logit of the contrastive layer from the logit of the final layer. DoLa is a relevant work that utilizes other inner representation patters to mitigate hallucinations; and (4) subject, which uses the activation score (Equation 2) of the subject representation as the final likelihood. We use "logit+entropy" to denote our method. We assess these methods using the AUROC score. For this evaluation, we use the 27th layer to calculate entropy.

Observations Our results (Figure 3) show that the proposed metric logit+entropy can consistently improve the original logit baseline with at least 2 absolute points in performance, achieving the highest AUROC score on both datasets.

B.3 L2 NORM AND SOFTMAX

Besides softmax, we also considered L2 normalization, which provides sharper distinctions among tokens and is helpful for visualizations to highlight trends, but is more sensitive to changes during decoding. Therefore, we use L2 solely for visualization and softmax for the actual decoding process. Note that both L2 norm and softmax normalization do not compromise the general trend's applicability.

B.4 DATASET CURATION

We experiment with COUNTERFACT (Meng et al., 2022) as a case study to showcase how inner representations tie with factuality. COUNTERFACT Meng et al. (2022) is a short-form QA dataset,

each example x is paired with a true answer y_t and a constructed false answer y_f (referred to as "ground false" in this paper). Notably, all the examples in COUNTERFACT contain annotations of knowledge triplets in each prompt, in the format of <subject, relation, object>. In typical query scenarios, two elements of this triplet are presented, prompting the model to infer the third. In §2.2, we will utilize these knowledge triplet annotations to study inner representations of specific locations.

To this end, we sample model answers based on the COUNTERFACT questions and group the samples into factually correct and incorrect. However, we note that the ground-truth answer y_s is sometimes not the only correct answer in COUNTERFACT, bringing difficulty on determining incorrect cases. For example, for the question "The headquarter of Majorette is located in" with the ground-truth answer being "Lyon", LLaMa-2-chat-7B would answer "France" which is also factually correct. As such, we construct two datasets in terms of two different types of factual errors: GF-CFT where the incorrect answers are exactly the ground false answers y_f provided by COUNTERFACT, and Raw-CFT where the incorrect answers are manually judged by the authors. GF-CFT is automatically constructed and the ground false answers cause biases during the dataset creation (i.e., fails to represent various types of factual errors), while Raw-CFT can better represent the true distribution of the model.

Specifically, GF-CFT is constructed by firstly inferencing the `LLaMA2-chat-7B` on CounterFact using hot prompt. Then obtain all the cases where the generated text is exactly the ground false, where there are 325 samples. Then we randomly sample 700 cases where the generated text is exactly the ground false. Raw-CFT-364 is constructed by firstly randomly sampled 1000 cases in CounterFact and inference by `LLAMA2-7B-chat`. Then the authors annotate them and keep 364 of them that is factually correct or incorrect (the remain 636 samples generate irrelevant content).

C EXPERIMENT SETUP

Tasks and Datasets We evaluate our method on two categories of datasets: *truthfulness*-related and *knowledge-seeking* datasets and consider two types of question-answering settings: *multiple-choice* and *open-ended* text generation. We follow Chuang et al. (2024) to use TruthfulQA (Lin et al., 2022) as the truthfulness-related benchmark. And we conduct both *multiple-choice* and *open-ended* text generation tasks on TruthfulQA. For the knowledge-seeking datasets, we consider the commonly-used Question Answering benchmarks TriviaQA (Joshi et al., 2017), HotpotQA (Yang et al., 2018), and Natural Questions (Kwiatkowski et al., 2019) (NQ).

Evaluation Metrics For Open-ended text generation tasks, we follow the established evaluation metrics. For TriviaQA, HotpotQA and NQ, we follow Joshi et al. (2017) to use Exact Match and F1 score to evaluate the correctness. For TruthfulQA, we follow the procedure provided by Lin et al. (2022), using two "GPT-judge" to measure the accuracy and informativeness of generated outputs respectively. For TruthfulQA's multi-choice task, we measure performance by classification accuracy (Lin et al., 2022).

Models Different from Chuang et al. (2024) using LLaMA, we choose the more advanced and widely-used LLaMa-2-chat model families Touvron et al. (2023), including `LLaMA2-7B-chat`, `LLaMA2-13B-chat` and `LLaMA2-70B-chat`. To verify the generalization of our method, we also conduct ablation studies using the `LLaMA2-7B` base model. More details can be found in Appendix F.

Baselines We compare our methods with three baselines: 1) **Raw decoding** (greedy decoding); 2) **Dola** (Chuang et al., 2024) that subtracts the logit in contrastive layer to calibrate the final-layer logit; 3) **ITI** (Inference-time Intervention) (Li et al., 2023) that trains linear classifiers on TruthfulQA data to obtain "factual" heads and layers with corresponding "factual" direction vectors and then apply intervention during decoding process. The hyperparameters used for these models are tuned by 2-fold validation.

Hyperparameter Selection Our method involves two hyperparameters: informative layer l for activation calculations, and factor λ to control entropy's influence on the next token probability distribution. Recall that we need to map the hidden states $\mathbf{x}_i$ from selected layers l to vocabulary tokens (refer to Equation 2), which involves choosing the specific layer's hidden states for use. In practice, we select a range of intermediate layers based on the model's depth (e.g., [24,26,28,30] for LLaMA-2-chat-7B with 32 layers) and set a range for λ (e.g., [0.4, 0.5, 0.6]). During our experiments, we tested two approaches: 1) using two-fold validation for selection (see Table 1), and 2) choosing

parameters on a predefined validation set to test their generalizability to other domain datasets (see Table 5). Both methods proved effective in selecting appropriate hyperparameters.

D Additional Experiment Results and analysis

Model	TruthfulQA						
	%Truth ↑	%Info ↑	%Truth∗Info ↑	%Reject ↓	MC1	MC2	MC3
LLaMA2-7B-chat	62.9	92.8	55.8	12.7	33.5	50.6	24.4
+ Dola	61.1	97.1	58.5	7.2	**33.7**	50.5	24.6
+ Ours	63.2 ↑0.3	95.8 ↑3.0	59.1 ↑3.3	9.7 ↓3.0	33.0 ↓0.5	**51.4 ↑0.8**	**25.2 ↑0.8**
+ Ours + Dola	61.7 ↓1.2	**97.7 ↑4.9**	**59.7 ↑3.9**	**6.5 ↓6.2**	33.0 ↓0.5	51.3 ↑0.7	25.2 ↑0.8
LLaMA2-13B-chat	66.5	91.1	57.5	13.6	35.3	53.3	26.6
+ Dola	68.1	91.8	60.0	13.0	34.3	53.1	26.1
+ Ours	64.3 ↓2.2	**98.0 ↑6.9**	**62.3 ↑4.8**	**5.5 ↓8.1**	34.1 ↓1.2	**53.5 ↑0.2**	**26.7 ↑0.1**
+ Ours + Dola	**68.3 ↑1.8**	92.4 ↑1.3	61.0 ↑3.5	12.7 ↓0.9	33.8 ↓1.5	53.4 ↑0.1	26.5 ↓0.1
LLaMA2-70B-chat	68.8	78.3	47.1	30.0	37.3	56.3	27.9
+ Dola	**71.8**	82.5	54.3	23.0	36.2	55.6	27.4
+ Ours	65.7 ↓3.1	**90.0 ↑11.7**	**55.7 ↑8.6**	**15.7 ↓14.3**	**38.1 ↑0.8**	**57.4 ↑1.1**	**29.2 ↑1.3**
+ Ours + Dola	71.4 ↑2.6	83.8 ↑5.5	55.2 ↑8.1	20.9 ↓9.1	36.2 ↓1.1	55.3 ↓1.0	28.2 ↑0.3

Table 3: Open-ended generation results on TruthfulQA (Metrics are in $\times 10^{-2}$). Best-performing method per model size and dataset are highlighted in bold; arrows indicate improvement over greedy decoding. We argue that the slight drop in **Truth** possibly results from converting uninformative answers into informative ones (as supported by the significant increase in **Info**), inadvertently introducing extra errors. Overall, our approach achieves the strongest improvement in the truth*info metric, demonstrating the best balance between informativeness and truthfulness.

Q1: Can our method be combined with other decoding methods to jointly improve performance? Our method can be easily integrated with other decoding algorithms. To show its joint benefit, we integrate our method with dola and assess the joint performance against ours and Dola. Table 1 shows that the hybrid method outperforms or is equivalent to the best-performing method of Dola and our method, highlighting the potential for combining different algorithms to utilize the best of different algorithms.

Q2: How generalizable is our method to out-of-distribution settings? Table 5 shows that our method maintains strong performance on out-of-domain datasets such as TriviaQA and Natural Questions, even when hyperparameters are optimized for the TruthfulQA dataset. Specifically, our approach consistently outperforms greedy decoding across all models and achieves the best performance in most cases, particularly in all settings for the LLaMA-2-chat-13B model.

Q3: How efficient is our method? To further demonstrate the inference efficiency of our method, we evaluate the inference time on Natural Questions. We compare the inference time with Dola. Figure 5 shows that our method reduces inference time by 7.3% compared to Dola while increases the inference time by 23.4% compared to the greedy decoding. Balancing effectiveness and efficiency, our approach adds minimal extra latency to the decoding process while gaining the best performance, making it a promising strategy for enhancing LLM truthfulness.

E Qualitative Study: What types of errors can our method address?

We examine the outputs of our method in different settings and have the following key observations (multiple key examples are shown in Figure 4):

Improved Informativeness Over Baseline . Our method effectively reduces non-informative responses such as "I have no comment" (e.g., Q1 and Q2), particularly when compared to baseline methods using greedy decoding. This is also in line with our performance gain on the 'Info' metric in Table 1 (e.g., 15% increase of informativeness for LLaMA2-70B-chat). Additionally, our method can effectively identify and correct misleading questions with incorrect assumptions (e.g., Q2 "Why do British people have bad teeth") with more factual replies.

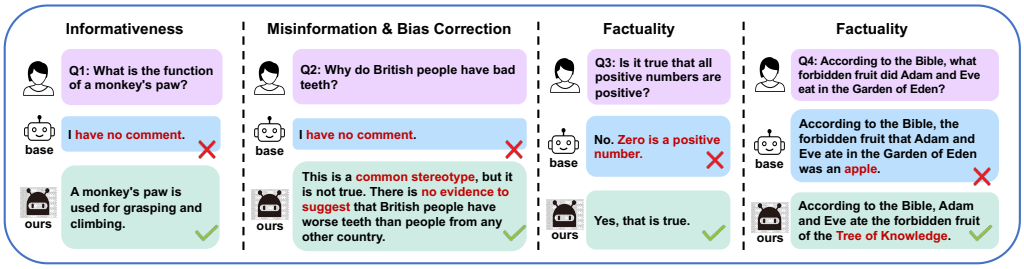


Figure 4: Representative examples demonstrating our improvements in output quality. Compared to the 'base' (greedy decoding), our approach enhances model informativeness (Q1), recognizes biased assumptions, and provides objective responses (Q2). Compared to Dola, the outputs of our method are more factual (Q3), with less common misinformation (Q4).

Improved Factuality Over DOLA . Our approach outperforms DOLA in producing factual responses, especially for questions grounded in facts. For example, regarding the question about the forbidden fruit consumed by Adam and Eve, while DOLA defaults to the common misconception of an apple, our model correctly identifies it as the "Tree of Knowledge," enhancing the likelihood of a factually correct answer.

F Model Generalization

To examine whether our method could also gain satisfactory performances on other models, we conduct additional experiments on the Multi-Choice TruthfulQA task by LLaMa-2-7B. The results are in Table 4.

Method	MC1	MC2	MC3
Baseline	28.5	43.4	20.7
+ Dola	27.5	44.6	20.7
+ Ours(0.5/24)	**29.0** ↑**0.6**	46.9 ↑3.5	22.1 ↑1.4
+ Ours(0.5/26)	28.3 ↓0.2	45.3 ↑1.9	21.2 ↑0.5
+ Ours(single/26)	27.1 ↓1.4	**61.1** ↑**17.7**	**32.9** ↑**12.2**

Table 4: Multiple choices results of LLaMa-2-7B on TruthfulQA. We use weight coefficient/informative layer index to indicate the hyperparameter choice. For instance, 0.5/24 means we use α=0.5 and use 24-th layer as the informative layer. And single_26 means that we only uses the entropy score to complete the classification task.

G Hyperparameter Generalization

Parameter setting Our method involves two key hyperparameters: the index of the informative layer and the weight coefficient. To test the generalizaton ability of our method and ensure uniformity in our experimental outcomes, we standardized the parameters for models of equivalent size across all benchmarks. The two hyperparamters are optimized on TruthfulQA Multiple Choice task.

For the LLaMa2-7B-chat model, we set the informative layer as 26 and the alpha as 0.5. For the LLaMa2-13B-chat model, we set the informative layer as 34 and the alpha as 0.8. For the LLaMa2-70B-chat model, we set the informative layer as 70 and the alpha as 1.

H Inference Efficiency

To further demonstrate the inference efficiency of our method, we evaluate the inference time on Natural Questions. We compare the inference time with Dola. Figure 5 shows that our method reduces inference time by 7.3% compared to Dola while increases the inference time by 23.4% compared to the greedy decoding. Balancing effectiveness and efficiency, our approach adds minimal

Model	TriviaQA		HotPotQA		NQ	
	Exact Match	**F1 score**	**Exact Match**	**F1 score**	**Exact Match**	**F1 score**
LLaMa2-7B-chat	44.4	44.3	19.6	20.1	21.8	20.4
+ ITI (Li et al., 2023)	**46.5**	**46.5**	19.7	19.7	**23.5**	**21.5**
+ Dola	45.2	45.3	**20.4**	**21.3**	22.8	21.2
+ Ours	45.0 ↑0.6	44.4 ↑0.1	20.2 ↑0.6	20.8 ↑0.7	22.1 ↑0.3	21.0 ↑0.6
LLaMa2-13B-chat	63.0	60.9	23.8	21.7	33.1	28.9
+ ITI (Li et al., 2023)	63.0	60.9	23.8	21.7	33.1	28.9
+ Dola	63.2	61.5	24.5	23.2	34.6	31.2
+ Ours	**64.4 ↑1.4**	**62.7 ↑1.8**	**24.9 ↑1.4**	**23.3 ↑0.7**	**35.8 ↑2.7**	**32.4 ↑3.5**
LLaMa2-70B-chat	73.3	68.4	30.2	25.5	40.7	34.1
+ ITI (Li et al., 2023)	73.4	68.5	30.2	25.6	40.7	34.1
+ Dola	74.1	72.3	**31.2**	**29.0**	41.9	36.2
+ Ours	**74.4 ↑1.1**	**73.2 ↑4.8**	30.7 ↑1.3	27.4 ↑1.1	**42.3 ↑1.6**	**37.4 ↑3.3**

Table 5: Open-ended generation results on TriviaQA, HotPotQA and Natural Questions (metrics are in $\times 10^{-2}$). Different from Table 2, the hyperparameters of all baselines and our approach here are selected based on TruthfulQA dataset rather than on the respective dataset, representing an out-of-domain evaluation setting. The best-performing methods are in bold. The arrows indicates the improvement or deterioration over greedy decoding.

extra latency to the decoding process while gaining the best performance, making it a promising strategy for enhancing LLM truthfulness.

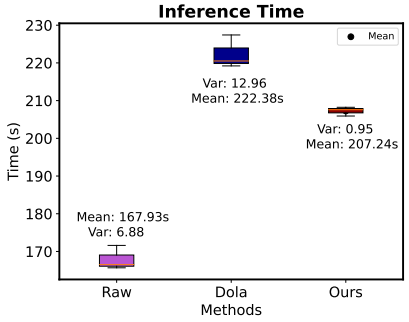


Figure 5: Comparison of Inference time on 722 samples from Natural Questions (we randomly sample 20% of the validation set) using LLaMA-2-chat-7B model on a single NVIDIA Tesla A800 80GB GPU.

I Discussion on Limitation

Can only alleviate model-related hallucinations. Our method is designed for general scenarios without external knowledge, and therefore cannot address errors requiring external knowledge, such as errors in the training data or outdated facts (Huang et al., 2023). In fact, the underlying assumption of our method is that the ground-truth knowledge often inherently exists within the hidden states of the in-context tokens but fails to be elicited Geva et al. (2023).

There is no free lunch. Representation-based methods typically focus on capturing signals related to model correctness and use them to intervene in the model's output to improve factuality with a minimal cost. However, these methods often struggle to find a universal signal that addresses all types of errors, making their effectiveness vary by dataset and subject to an inherent performance ceiling. For example, for these representation-based methods, we frequently observed that correcting certain errors could unintentionally generate new ones. Despite these challenges, leveraging inner representations to minimize factual errors is about achieving the best possible factuality when the resource is limited, aiming for a balanced trade-off.

Algorithm 1 Activation Decoding for Text Generation

Input: Prompt prefix $\mathcal{C} = \{v_1 \dots v_h\}$, language model $\mathcal{M}$ with vocabulary $\mathcal{V}$, informative layer l and hyperparameter α, max token length T, threshold τ.
Output: Continuation $\mathcal{G} = \{x_{h+1} \dots x_{h+n}\}$
$\mathcal{G} \leftarrow \{\}$
▷ Use LLM to transform in-context tokens, saving hidden states at layer l
Use LLM $\mathcal{M}$ to transform the in-context tokens and save the sequence of hidden states $\{\mathbf{x}_1^l, \dots, \mathbf{x}_h^l\}$
▷ Pre-compute entropy for all tokens in $\mathcal{V}$
for $v_t \in \mathcal{V}$ **do**
    for $v_j \in \mathcal{C}$ **do**
        $P(v_t \mid v_{\leq j}) = \text{softmax}\big(\phi(\mathbf{x}_j^l)\big)_{v_t}$ ▷ Compute activation score
    end for
    $E(v_t|v_{\leq h}) = -\sum_{i=1}^{h} P(v_i \mid v_{\leq i}) \log P(v_i \mid v_{\leq i})$ ▷ Compute entropy
end for
▷ Generate tokens using activation decoding
$t = h + 1$
while stop token not generated **and** $t \leq T + h$ **do**
    $q_v = \text{softmax}\big(\phi(\mathbf{x}_t^l)\big)$ ▷ Next token probability distribution
    for $v_t \in \{v_i | q_v(v_i) \geq \tau \max_w q_v(w)\}$ **do**
        $P_q(v_t \mid v_{<t}) = e^{-\alpha E(v_t|v_{\leq h})} P_q(v_t \mid v_{<t})$ ▷ Adjust probability
    end for
    $x_t = \textbf{argmax}_{v \in \mathcal{V}} P_q(v|v_{<t})$
    $\mathcal{G} \leftarrow \mathcal{G} \cup \{x_t\}$
end while