In-Context Sharpness as Alerts: An Inner Representation Perspective for Hallucination Mitigation

Shiqi Chen [*1] **Miao Xiong** [*2] **Junteng Liu** [3] **ZhengXuan Wu** [4] **Teng Xiao** [5] **Siyang Gao** [1] **Junxian He** [6]
schen438-c@my.cityu.edu.hk miao.xiong@u.nus.edu junxianh@cse.ust.hk

Abstract

Large language models (LLMs) frequently hallucinate and produce factual errors, yet our understanding of why they make these errors remains limited. In this study, we delve into the underlying mechanisms of LLM hallucinations from the perspective of *inner representations*, and discover a salient pattern associated with hallucinations: correct generations tend to have *sharper* context activations in the hidden states of the in-context tokens, compared to the incorrect ones. Leveraging this insight, we propose an entropy-based metric to quantify the "sharpness" among the in-context hidden states and incorporate it into the decoding process to formulate a constrained decoding approach. Experiments on various knowledge-seeking and hallucination benchmarks demonstrate our approach's consistent effectiveness, for example, achieving up to an 8.6 point improvement on TruthfulQA. We believe this study can improve our understanding of hallucinations and serve as a practical solution for hallucination mitigation. Code is publicly available at https://github.com/hkust-nlp/Activation_Decoding.

1 Introduction

Large language models (LLMs) have made remarkable advancements in recent years, with extensive applications across various domains (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; Wang et al., 2023). To address the factuality issue, many efforts have focused on retrieving external knowledge (Ram et al., 2023b; Yu et al., 2023; Jiang et al., 2023) for generation or fact-checking, as well as fine-tuning (Asai et al., 2023) and self-evaluation (Pan et al., 2023; Xiong et al., 2023). However, these methods often require high computational resources or high-quality knowledge bases, which may not be available for domain-specific cases. In contrast, we aim to tackle this challenge from the perspective of model's inner representations, investigating whether the hidden states contain information about hallucination.

To gain this mechanistic understanding of hallucinations[1] through the lens of hidden states, we begin by formulating an internal *knowledge extraction* process following Geva et al. (2023), to examine whether the hidden states of the model encode the target knowledge in the response. The underlying hypothesis is that hallucinated knowledge may not be encoded into the intermediate hidden states. Specifically, for example, in the prompt '`Beats Music is owned by`', if the target token '`Apple`' is encoded within the representations of the subject '`Beats Music`', we posit that the model "knows" `Beats Music` and `Apple` are related in some ways and `Apple` is more likely to be a correct answer. In this case, we consider the token '`Apple`' activated by the representations of '`Beats Music`'. Our case study (Table 1) on the COUNTERFACT dataset (Meng et al., 2022) – where the subject, object, and relation annotations are available – reveal that the correct answers have a significantly higher rate of activation compared to incorrect answers.

However, this preliminary study based on the method in Geva et al. (2023) requires knowledge triplet annotations. To mitigate this requirement, we extend our investigation to examining the activations of correct versus incorrect answers across the full input sequence, moving beyond merely

[*]Equal contribution. Work done during SC's vist to HKUST. [1]City University of Hong Kong [2]National University of Singapore [3]Shanghai Jiao Tong University [4]Stanford University [5]Penn State University [6]HKUST. Correspondence to: Junxian He <junxianh@cse.ust.hk>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[1]Hallucinations can be described as outputs that do not conform to the model's inner belief (Zou et al., 2023). However, the concept of "inner belief" and its measurement remains debatable. Thus in this paper, we use "hallucination" to denote a generation that is not aligned with world knowledge to simplify the discussion, which shares the same definition as "factuality".

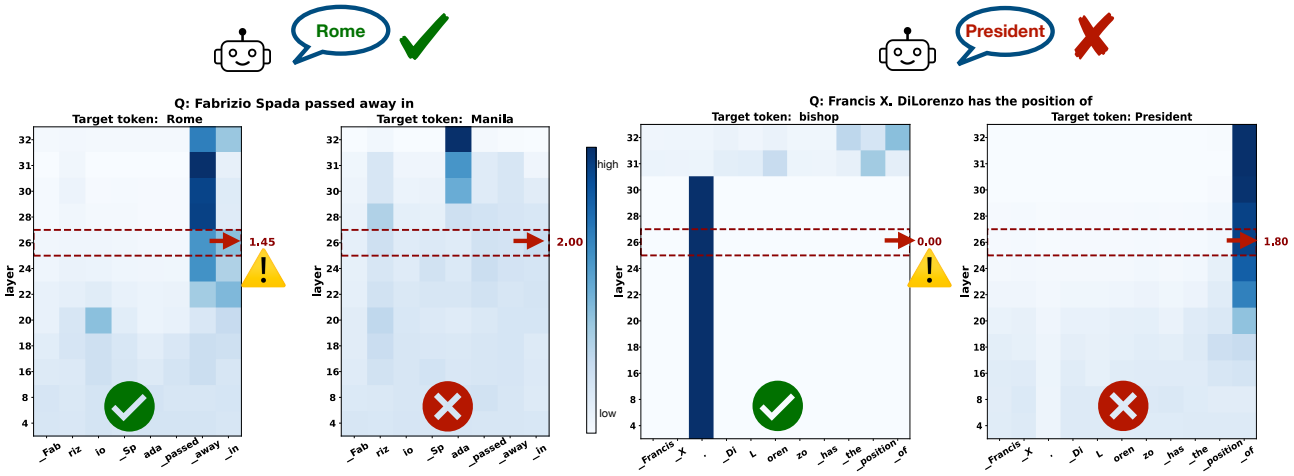


Figure 1. Visualization of why in-context activation can serve as an alarming signal for factuality. For a given question (e.g., "Fabrizio Spada passed away in __"), we visualize the activation of the truth and false tokens across transformer layers. **Left**: we use the ground truth and false answers from COUNTERFACT (e.g., "Rome" or "Manila") as the target tokens. In this example, the model generates the correct answer. **Right**: we use the ground truth answer and the model's generated false answer. We then calculate the activation entropy across intermediate layers, focusing on the 26-th layer's entropy value (detailed calculation in Section 3.4). This entropy metric is annotated in the figure. Our findings reveal that incorrect tokens generally exhibit higher entropy than correct ones.

the subject focus. Our findings reveal that *correct generations typically exhibit sharper context activations compared to incorrect ones*, particularly within the intermediate layers across the in-context tokens. For instance, as illustrated in the left part of Figure 1 when comparing the correct prediction "Rome" to "Manila", the correct prediction's in-context activations in intermediate layers (e.g., 26th layer) are significantly sharper across the text sequence, while "Manila" is only activated right before the output layers (e.g., 32nd layer). Additional quantitative evidence reveals this *in-context sharpness* in §3. These findings further motivate us to formalize this in-context sharpnessof the model's representations to study hallucination.

To measure the observed in-context sharpness, we introduce an entropy-based metric by normalizing all the context activations given a target token into a probability distribution, and computing its entropy. Intuitively, a smaller entropy value suggests a sharper in-context activation pattern, and a greater chance of the token being factually correct. We first validate the effectiveness of this metric in differentiating the true and false answers (§3.4 and Figure 2), achieving an AUROC up to 0.76. Then we incorporate entropy into the decoding process, forming a constrained decoding approach named `Activation Decoding` to improve factuality.

On 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 error generations across different model size (e.g., 16.1% increase in F1 score for HotpotQA on average). Our experiments on TruthfulQA (Lin et al., 2022) demonstrate that `Activation Decoding` 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 Related Work

Factuality and Inner Representation Several recent studies have focused on understanding factuality through the analysis of inner representations. Among them, Yuksekgonul et al. (2024) identified a positive correlation between factual errors and attention behaviors by analyzing attention patterns. Halawi et al. (2023) attributes the factual errors to false deduction heads and critical layers. Alternatively, ITI (Li et al., 2023a) and Repe (Zou et al., 2023) proposed using probing methods to locate error heads and layers in models. These approaches collectively confirm that inner representations carry rich information about a model's internal processes and the factuality of its responses.

Controlled Generation Our research contributes to the growing field of *constrained decoding* algorithms, especially those designed to improve the factuality of language models. This involves employing intervention strategies during the generation process. Notably, ITI (Li et al., 2023a) and Repe (Zou et al., 2023) propose to probe attention heads

or layers associated with model correctness. They modify the decoding process by adding direction vectors in favor of truthful generation. DoLa (Chuang et al., 2024) assumes that factual knowledge can be localized to particular transformer layers and then utilizes the contrasting logits obtained from projecting the later layers versus the earlier layers to adjust the next-token distribution. Unlike these studies, we take a new perspective to explore the relation between the in-context sequence and the generated outputs.

Mechanistic Interpretability Our study also aligns with mechanistic interpretability (Olah, 2022; Nanda et al., 2023), especially the factual knowledge recall perspective. Prior works established a connection between internal model components and the retrieval of factual information, particularly by analyzing knowledge triplets (subject, relation, and object). These investigations reveal that MLP (multi-layer perceptron) layers play a crucial role in the storage of knowledge (Geva et al., 2022; Dai et al., 2022; Meng et al., 2022), whereas attention mechanisms are more engaged in the transfer of factual knowledge (Elhage et al., 2021; Geva et al., 2023; Yuksekgonul et al., 2024). Building on these foundational insights, our work aims to understand and further enhance the factuality of the model's generation.

3 Diving into Internal Representations

Inspired by Chuang et al. (2024); Li et al. (2023b); Zou et al. (2023) which reveal that the inner representations in LLMs contain rich information about hallucination, we delve into these inner representations in this section, aiming to gain a deeper insight and broaden our perspective on the implications of these internal states for factuality. We start with case studies on a short-form QA dataset, and focus on whether these inner representations can reflect factuality, and how we can utilize them to detect and mitigate hallucinations.

3.1 Notation

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 W. This LM head maps the inner representation to the token probability distribution for the next token. 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 the 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 LM head W predicts the probability of the next token v_{T+1} using the hidden states $\mathbf{x}_T^H$:[2]

$$P(v_{T+1} \mid v_{1:T}) = \mathrm{softmax}\big(W\mathbf{x}_T^H\big)_{v_{T+1}}. \quad (1)$$

	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 Raw-CFT and GF-CFT using confusion matrices. 'Activated' refers to samples whose generated tokens are activated by in-context tokens; 'Correct' refers to samples that are correctly predicted. Results indicate that correct samples have a higher rate of activation.

3.2 Experimental Setup for the Case Study

We experiment with COUNTERFACT (Meng et al., 2022) as a case study to showcase how inner representations tie with factuality. COUNTERFACT 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" later). Notably, all the examples in COUNTERFACT contain annotations of knowledge triplets for 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 §3.3, we will utilize these knowledge triplet annotations to study inner representations of specific locations.

We aim to examine and compare the inner representations of the model in both cases where the model produces factually correct and incorrect answers. 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 in determining whether the prediction is correct.[3] 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 from model predictions and manually judged by the authors. GF-CFT is automatically constructed and the ground false answers fail to represent various types of fac-

[2]For simplicity, we will omit the layer annotation when there is no confusion.

[3]For example, in question "The twin city of Boston is" with the ground-truth answer as "Athens", LLaMA-2-chat-7B would answer "a popular tourist destination" which is not factually wrong.

tual errors, while Raw-CFT can better represent the true distribution of the model. Please refer to Appendix B for more details on the two datasets. In this section, we utilize `LLaMA2-chat-7B` as the model for study. Next, we present our findings under this experimental setup.

3.3 Finding 1: Activation implies answer correctness

Several studies (Ram et al., 2023a; Dar et al., 2023; Geva et al., 2023) try to understand how inner representations evolves through transformer layers to generate factually correct outputs. Specifically, Geva et al. (2023) interprets intermediate layers as an *information extraction process* from an information flow perspective. E.g, in a prompt '`Beats Music is owned by`', the embedding of the subject '`Beats Music`' contains many related attributes (like '`Apple`') of it. Inspired by this knowledge extraction interpretation, we aim to investigate whether the model's response is encoded in the corresponding subject embedding and whether this is related to its answer correctness. The idea is that if the model can successfully extract related attributes (e.g. ground truth tokens) while processing the input sequence, it may indicate the possession of the necessary knowledge to correctly answer the questions, and hence is more likely to produce correct responses.

To examine the above idea, we employ the projection method (Ram et al., 2023a; Dar et al., 2023; Geva et al., 2023) to map the hidden representations $\mathbf{x}_i$ to a given vocabulary token v through the output layer W:

$$s(i, v) = \text{softmax}\big(W\mathbf{x}_i\big)_v, \tag{2}$$

where $s(i, v)$, the *activation score*, measures how likely the given token v will be encoded by the subject's last token v_i. We rank the activation scores for all vocabulary tokens and *consider a token activated by the subject token if it ranks within the top 50 scores*. If not, the token is deemed unactivated. Note that here we use the hidden states of the last subject token at the 26-th transformer layer output as the *subject embedding*, within the 32 layers in total for `LLaMA2-chat-7B`. We select the 26-th layer (referred to as *informative layer*) based on our observation that deep layers show a higher level of activation, suggesting that they contain more internal knowledge, which also aligns with the findings of Halawi et al. (2023). The location of the informative layer is a tunable hyperparameter, while we find that the conclusions are not sensitive among a range of deep layers (e.g. across 26-30 layers) in the preliminary experiments.

Observations Based on the RAW-CFT and GF-CFT datasets, we examine whether the model-generated tokens in the correct examples are activated more often than the incorrect ones. 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 higher chance of answer correctness.

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

The above-mentioned analytical approach relies on the knowledge triplet annotations from the COUNTERFACT dataset, which is often unavailable in practice. To address this challenge, we extend the approach in §3.3 to analyze the activation between target tokens and all in-context tokens (rather than solely considering the subject token) to capture the overall pattern. Our visualization of activations in Figure 1 shows that correct and incorrect prediction candidates demonstrate distinct patterns of activations: *the in-context activations across different locations in the context sequence tend to be significantly sharper for the correct prediction in the middle to high intermediate layers* compared to the incorrect one, for example, "Rome" against "Manila" in the left part of Figure 1. This also aligns well with our analysis in §3.3 – 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.

Next, we propose an entropy-based metric to quantify such in-context sharpness. Specifically, given the prompt $\mathcal{C} = \{v_1, v_2, \ldots, v_t\}$, for a predictive token v_p, we first define its normalized activation probability with respect to the context token v_i as:

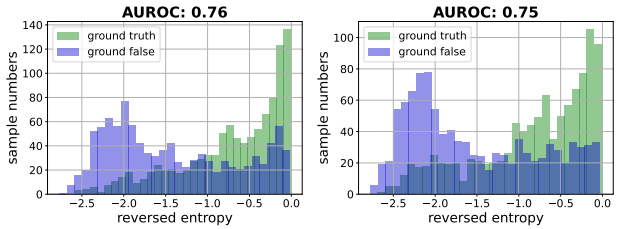
$$\tilde{P}(activation = i | v_p, v_{1:t}) = \frac{e^{s(i,v_p)}}{\sum_{m=1}^{t} e^{s(m,v_p)}}. \tag{3}$$

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.

The above activation score indicates how likely the knowledge represented by v_p will be extracted from the prompt sequence. Then the entropy of $\tilde{P}$, which we refer to as *contextual entropy*, is used to describe the in-context sharpness of a given token v_p's activation to all in-context tokens:

$$E(v_p, v_{1:t}) = -\sum_{i=1}^{t} \tilde{P}(activation = i \mid v_p, v_{1:t}) \log \tilde{P}(activation = i \mid v_p, v_{1:t}). \tag{4}$$

To measure the correlation between entropy and factuality, we evaluate the contextual entropy metric to distinguish



(a) 28th layer (b) 26th layer

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.

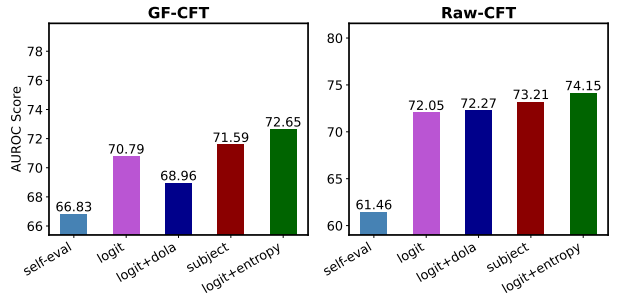


Figure 3. AUROC score on GF-CFT and Raw-CFT among different baselines. Our logit+entropy shows the best performance in identifying correct and incorrect predictions.

between ground true and false answers on the GF-CFT dataset. Here, the inner representation is selected as hidden states after the 26th and 28th transformer layers respectively.

Observations 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. Besides, we compute the Area under the ROC curve (AUROC) for entropy as a quantitative metric to differentiate between factually correct and incorrect samples, which is over 0.75 for both the 26th and 28th layer representations. This implies the effectiveness of the proposed entropy-based metric as factual error detectors. Next, we try to incorporate this intuition into the decoding process, examining whether Equation 4 can help alter the prediction distribution to mitigate hallucinations.

3.5 Finding 3: Contextual entropy can calibrate the next token prediction

Our previous findings suggest that predictive tokens with smaller contextual entropy are more likely to be correct. Based on this, a natural approach is to favor tokens with smaller contextual entropy in generation, while suppressing those that enlarge contextual entropy. To implement this, we adjust the original next token probability distribution using the contextual entropy measure in Equation 4. Formally, we adjust the original token probability distribution as:

$$P(v_p \mid v_{1:p-1}) \propto e^{-\lambda E(v_p, v_{1:t})} P(v_p \mid v_{1:p-1}), \quad (5)$$

where t is the in-context prompt length – we only consider the activations on the input prompt tokens as $E(v_p, v_{1:t})$, excluding the newly generated tokens. This is to avoid deductive hallucination during the decoding process. $\lambda \in [0, 1]$ 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 contextual entropy is encouraged.

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 patterns 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.

4 Activation Decoding

Based on the finding in §3.5, we formally introduce a novel constrained decoding method of LLMs, referred to as `Activation Decoding`, based on the proposed contextual entropy metric to enhance the factuality of model generations. Specifically, we adjust the next token's probability distribution using Equation 5. We follow a filtering operation by Chuang et al. (2024) to first select tokens with high probabilities (the filtering threshold is set to be 0.1), then adjust the probability of these selected tokens. This adjusted token probability distribution is then used to predict the next token, where various decoding algorithms can be applied, such as greedy decoding and beam search. The decoding method is illustrated in Figure 4, and the pseudo algorithm

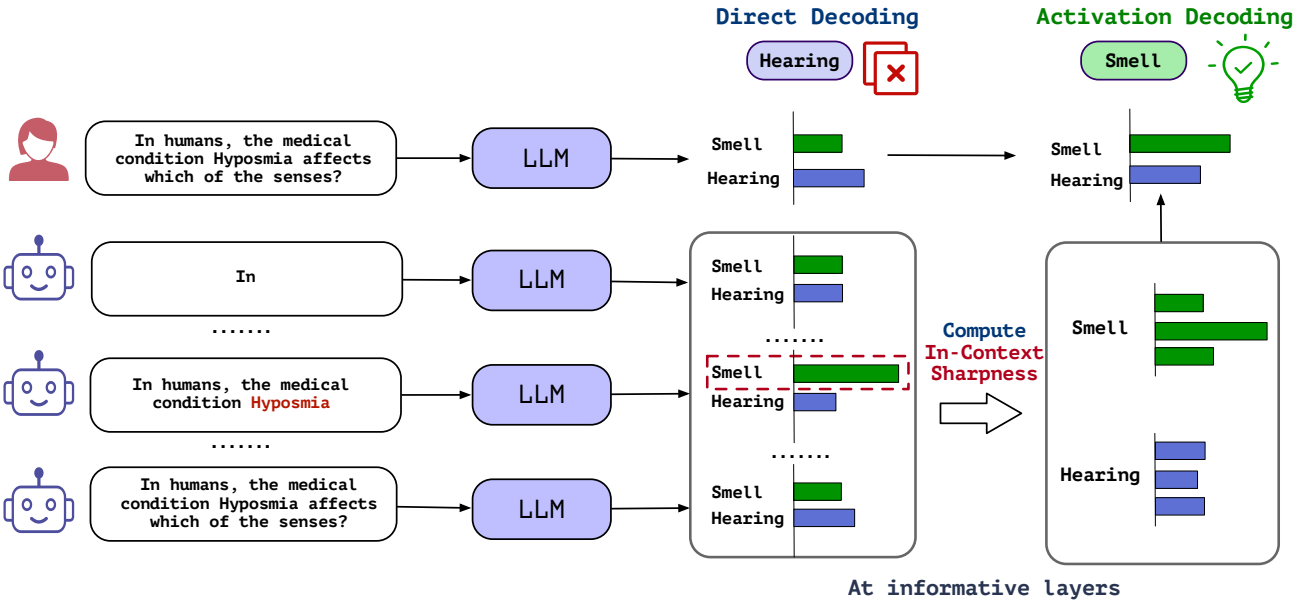


Figure 4. Overview of our `Activation Decoding` method. Given the prompt, the direct decoding (i.e., greedy decoding) algorithm generates the wrong answer 'Hearing'. Here we show how our method can successfully encourage the correct answer 'Smell' to be decoded. Considering the correct token 'Smell' as an example, **1)** we first calculate its activation scores to each in-context token using Eq. 2. Note that it exhibits strong activation when processed with the in-context token 'Hyposmia'. **2)** We then aggregate these corresponding activation scores together and normalize them into a distribution using Eq. 3 to measure the in-context sharpness. Here the correct token with strong activation has a larger sharpness. **3)** We use contextual entropy (Eq. 4) to quantify the sharpness. **4)** This entropy is then used as a penalty term to adjust the original token likelihood distribution, boosting its probability of being decoded.

is shown in Algorithm 1 of the Appendix.

Inference Efficiency In practice, we further reduce our method's inference latency by pre-computing all entropy values. The key to reducing latency is to optimize the computation of contextual 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. During generation, we can directly look up the entropy value – this is because we only consider activations with respect to the prompt tokens, excluding the generated tokens as mentioned in §3.5. This creates a 32000-dimensional entropy vector (The model we use, 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.

5 Experiments

5.1 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. 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"s to measure the accuracy and informativeness of generated outputs respectively. For TruthfulQA's multi-choice task, we measure performance by accuracy.

Models Different from Chuang et al. (2024) that experiment with the LLaMA base model, we choose the more advanced LLaMA-2-chat model families (Touvron et al., 2023) that are more commonly deployed in practice than the base models, 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, which can be found in Appendix A.

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., 2023a) 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. It is worth noting that ITI requires a training process on labeled data while other baselines including our approach can be directly applied during inference. The hyperparameters used for these models are tuned by 2-fold validation on the respective benchmark separately.

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 from 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) in-domain validation, where we use two-fold validation for the respective benchmark separately (see Table 3), and 2) out-of-domain validation, where we use the Truth*Info metric on TruthfulQA as the validation metric and fix these hyperparameters for all other benchmarks. We choose parameters on a predefined validation set to test their generalizability to other domain datasets (see Table 6). Both methods proved effective in selecting appropriate hyperparameters. We refer more details and ablation study to Appendix C.

5.2 Results

Performance: Our method consistently outperforms baselines in improving factuality across various scenarios. The comparison results are summarized in Table 2 for Open-ended and Multi-Choice TruthfulQA, and Table 3 for knowledge-seeking datasets. For open-ended TruthfulQA (Table 2), 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 respectively. 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 stronger LLMs. ITI does not perform well on the 13B and 70B models. On the other hand, , which serves as an effective baseline, demonstrates strong performance in most cases, albeit slightly inferior to our approach.

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 3 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 them, and the hallucination hidden states patterns from our approach and DoLa are complimentary to each other to some extent.

Q2: How generalizable is our method to out-of-distribution settings? Table 6 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 6 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. Our method achieves faster processing speeds compared to DoLa because we eliminate the need to calculate the "contrast layer" by comparing the JS-Divergence of the layers being selected. 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.

5.3 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 5):

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 2 (e.g., 15% increase of informativeness for LLaMA2-70B-chat). Additionally, our method can effectively identify and correct misleading questions with incorrect assumptions

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
+ ITI (Li et al., 2023a)	**66.5**	85.9	52.5	21.6	33.7	51.3	24.9
+ Dola (Chuang et al., 2024)	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
+ ITI (Li et al., 2023a)	66.6	91.1	57.8	13.5	**35.4**	53.3	26.7
+ Dola (Chuang et al., 2024)	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
+ ITI (Li et al., 2023a)	68.8	78.3	47.1	30.0	37.3	56.3	27.9
+ Dola (Chuang et al., 2024)	**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 2. 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.

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., 2023a)	46.5	46.5	19.7	19.7	**23.5**	21.5
+ Dola (Chuang et al., 2024)	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
+ ITI (Li et al., 2023a)	63.0	60.9	23.8	21.7	33.1	28.9
+ Dola (Chuang et al., 2024)	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
+ ITI (Li et al., 2023a)	73.4	68.5	30.2	25.6	40.7	34.1
+ Dola (Chuang et al., 2024)	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 3. 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.

(e.g., Q2 "Why do British people have bad teeth") with more factual replies.

Improved Factuality Over DoLa. Our approach outperforms 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.

6 Conclusions and Discussion

We introduce a new perspective–the temporal view–to study why models make factual errors, i.e., studying the *changes in inner representations across time* and their *correlation to final predictions* as the model processes each token of the input question. We uncover an intrinsic pattern within LLMs that correlates with factual correctness and introduces an unsupervised metric *in-context sharpness* to indicate factuality, i.e., higher in-context sharpness suggests a greater likelihood of an answer being factually correct. Building

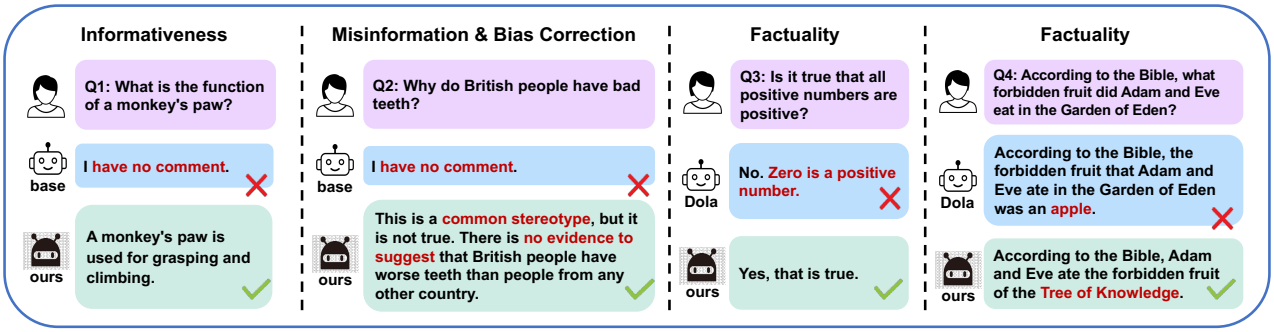


Figure 5. Representative examples demonstrating our improvements in output quality. Compared to the 'base' (greedy decoding), our approach enhances model informativeness (Q1), recognizes biased assumptions, and provides objective responses (Q2). Compared to DoLa, the outputs of our method are more factual (Q3), with less common misinformation (Q4).

upon in-context sharpness, we develop a decoding method for hallucination mitigation. Our experiments confirm that this method can stably enhance LLM's performance on various benchmarks beyond the traditional factuality benchmark TruthfulQA and for models in different frameworks (LLaMA and Mistral) and sizes (7B, 13B, 70B).

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.

7 Impact Statement

This paper presents work whose goal is to improve the understanding of mechanisms of LLM hallucinations. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Asai, A., Wu, Z., Wang, Y., Sil, A., and Hajishirzi, H. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*, 2023. URL https://arxiv.org/abs/2310.11511.

Chen, S., Zhao, Y., Zhang, J., Chern, I.-C., Gao, S., Liu, P., and He, J. 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.

Chuang, Y.-S., Xie, Y., Luo, H., Kim, Y., Glass, J., and He, P. 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.

Dai, D., Dong, L., Hao, Y., Sui, Z., Chang, B., and Wei, F. Knowledge neurons in pretrained transformers. In *Association for Computational Linguistics (ACL)*, 2022. URL https://arxiv.org/abs/2104.08696.

Dar, G., Geva, M., Gupta, A., and Berant, J. Analyzing transformers in embedding space. In *Association for Computational Linguistics (ACL)*, 2023. URL https://arxiv.org/abs/2209.02535.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al. A mathematical framework for transformer circuits. In *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Geva, M., Schuster, R., Berant, J., and Levy, O. Transformer feed-forward layers are key-value memories. In *Association for Computational Linguistics (ACL)*, 2022. URL https://arxiv.org/abs/2012.14913.

Geva, M., Bastings, J., Filippova, K., and Globerson, A. 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.

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

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., 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.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. In *ACM Computing Surveys*, 2023. URL https://arxiv.org/abs/2202.03629.

Jiang, Z., Xu, F. F., Gao, L., Sun, Z., Liu, Q., Dwivedi-Yu, J., Yang, Y., Callan, J., and Neubig, G. Active retrieval augmented generation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2023. URL https://arxiv.org/abs/2305.06983.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. 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.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., 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.

Kaddour, J., Harris, J., Mozes, M., Bradley, H., Raileanu, R., and McHardy, R. Challenges and applications of large language models. In *arXiv preprint arXiv:2307.10169*, 2023. URL https://arxiv.org/abs/2307.10169.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.-W., Dai, A. M., Uszkoreit, J., Le, Q., and Petrov, S. Natural questions: A benchmark for question answering research. In *Transactions of the Association of Computational Linguistics (TACL)*, 2019. URL https://aclanthology.org/Q19-1026/.

Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a. URL https://arxiv.org/abs/2306.03341.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T., Zettlemoyer, L., and Lewis, M. Contrastive decoding: Open-ended text generation as optimization. In *Association for Computational Linguistics (ACL)*, 2023b. URL https://arxiv.org/abs/2210.15097.

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

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, 2022.

Nanda, N., Chan, L., Lieberum, T., Smith, J., and Steinhardt, J. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2301.05217.

Olah, C. Mechanistic interpretability, variables, and the importance of interpretable bases. In *Transformer Circuits Thread*, 2022. URL URLhttps://transformer-circuits.pub/2022/mech-interp-essay/index.html.

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.

Pan, L., Saxon, M., Xu, W., Nathani, D., Wang, X., and Wang, W. Y. 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.

Ram, O., Bezalel, L., Zicher, A., Belinkov, Y., Berant, J., and Globerson, A. What are you token about? dense retrieval as distributions over the vocabulary. In *Association for Computational Linguistics (ACL)*, 2023a. URL https://arxiv.org/abs/2212.10380.

Ram, O., Levine, Y., Dalmedigos, I., Muhlgay, D., Shashua, A., Leyton-Brown, K., and Shoham, Y. In-context retrieval-augmented language models. In *Association for Computational Linguistics (ACL)*, 2023b. URL https://arxiv.org/abs/2302.00083.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., 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.

Wang, C., Liu, X., Yue, Y., Tang, X., Zhang, T., Jiayang, C., Yao, Y., Gao, W., Hu, X., Qi, Z., et al. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity. In *arXiv preprint arXiv:2310.07521*, 2023. URL https://arxiv.org/abs/2310.07521.

Xiong, M., Hu, Z., Lu, X., Li, Y., Fu, J., He, J., and Hooi, B. 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.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W. W., Salakhutdinov, R., and Manning, C. D. 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.

Yu, W., Zhang, Z., Liang, Z., Jiang, M., and Sabharwal, A. Improving language models via plug-and-play retrieval feedback. In *arXiv preprint arXiv:2305.14002*, 2023. URL https://arxiv.org/abs/2305.14002.

Yuksekgonul, M., Chandrasekaran, V., Jones, E., Gunasekar, S., Naik, R., Palangi, H., Kamar, E., and Nushi, B. Attention satisfies: A constraint-satisfaction lens on factual errors of language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2309.15098.

Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., et al. Representation engineering: A top-down approach to ai transparency. In *arXiv preprint arXiv:2310.01405*, 2023. URL https://arxiv.org/abs/2310.01405.

A 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.

Model	Methods	HotpotQA	Natural Questions	TriviaQA
Mistral-7B	Raw Decoding	22.8/20.9	31.3/27.4	65.5/63.3
Mistral-7B	DoLa	23.9/22.7	32.1/28.5	66.0/64.9
Mistral-7B	Ours (layer/ λ: 32/0.2)	25.1/25.0	31.2/28.6	66.2/65.4
Mistral-7B	Ours (layer/ λ: 30/0.5)	24.4/24.2	31.9/28.5	65.7/65.0

Table 5. QA results of Mistral on three QA benchmarks. We use Exact Match/F1 score to indicate the results.

B 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 §3.3, we will utilize these knowledge triplet annotations to study inner representations of specific locations.

We aim to examine and compare the inner representations of the model in both cases when the model produces factually correct and incorrect answers. 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 running inference of `LLaMA2-chat-7B` on CounterFact using 2-shot prompt. Then we obtain all the cases where the generated text is exactly the ground truth, where there are 325 samples. Next, we randomly sample 700 cases where the generated text is exactly the ground false. Raw-CFT is constructed by firstly randomly sampling 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 remaining 636 samples generate irrelevant content).

C Hyperparameter Generalization

Parameter setting Our method involves two key hyperparameters: the index of the informative layer (i.e., the layer used to compute in-context sharpness) and the weight coefficient λ. We utilize a separate validation set for hyperparameter optimization to select these two hyperparameters. Specifically, we employ two types of validation sets:

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., 2023a)	**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., 2023a)	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., 2023a)	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 6. Open-ended generation results on TriviaQA, HotPotQA and Natural Questions (metrics are in $\times 10^{-2}$). Different from Table 3, 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.

- An **in-distribution (ID Val)** validation set, which is drawn from the same distribution as the test dataset. The reported scores (see Table 2) are calculated using 2-fold cross-validation on the test dataset.
- An **out-of-distribution (OOD Val)** validation set, where we choose hyperparameters from a completely different dataset, TruthfulQA. The results of this approach are detailed in Table 6. To clarify, for evaluations on datasets such as TriviaQA, HotpotQA, and NQ, we apply the hyperparameters that were optimized on the TruthfulQA dataset.

Implementation Details The hyperparameters used in the ID Val and OOD Val setting in our paper are shown in Table 7.

Model	Informative Layer	Alpha	In-domain Layer Range
LLaMA2-7B-chat	26	0.5	[24, 32]
LLaMA2-13B-chat	34	0.8	[28, 38]
LLaMA2-70B-chat	70	1.0	[66, 72]

Table 7. The hyperparameters used in our paper's ID Val and OOD Val setting. For the in-distribution setting, we set the 'In-domain Layer Range' and use 2-fold cross-validation to select the corresponding hyperparameters and derive the final results. For the out-of-distribution setting, we use the 'Informative Layer' and 'Alpha' selected on TruthfulQA and apply them to the other three benchmarks.

D GPT-4 Evaluation of Text Generation and Instruction Following Quality

To evaluate the generation quality and the instruction following quality of our decoding method, we follow the same setting of the DoLa Chuang et al. (2024) (their Appendix Section D) to conduct a comparative analysis of our method against the baseline (greedy decoding) using the VicunaQA dataset. We use GPT-4 to evaluate the text generation quality on a scale of 1 to 10. The prompt we use is: *You are a helpful and precise assistant for checking the quality of the answer. Please rate by the grammaticality and cohesiveness of their responses, but not factuality. You are not required to verify the factual accuracy of the answers. Each assistant receives an overall score on a scale of 1 to 10, where a higher score indicates better quality.*

The results, presented in Table 8, show the average GPT-4 evaluation score and indicate that our method improves or is comparable to the baseline in terms of generation quality and instruction-following capabilities. Besides, It is worth noting that the quality scores are assessed in pairs, allowing for a direct comparison between two responses simultaneously. Consequently, there may be instances where a larger model receives a lower quality score, a phenomenon also noted in Table 8 of the DoLa (Chuang et al., 2024) paper, where the LLaMA-2-33B model scores lower on the GPT-4 evaluation than the LLaMA-2-13B.

Model	Baseline	Ours
LLamA-2-7B	5.2063	5.8625
LLaMA-2-7B-Chat	9.2062	9.0375
LLamA-2-13B-Chat	7.2375	8.4688

Table 8. Comparative evaluation of our method against the baseline (greedy decoding) regarding text generation and instruction following ability. We use GPT-4 for pairwise evaluation of the text generation quality on a scale of 1 to 10 on both models.

E 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 6 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.

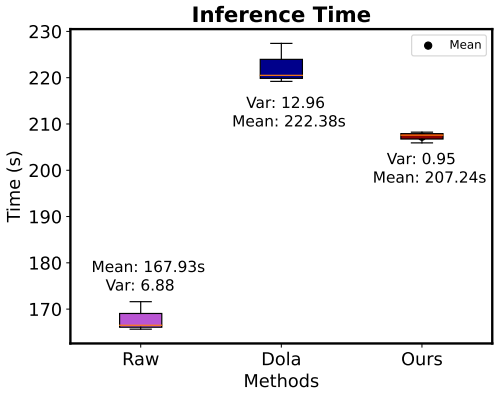


Figure 6. Comparison of Inference time on 722 samples from Natural Questions (we randomly sample 20% of the validation set) using LLaMA-2-chat-7B model on a single NVIDIA Tesla A800 80GB GPU.

Algorithm 1 Activation Decoding for Text Generation

Input: Prompt prefix (i.e., in-context tokens) $\mathcal{C} = \{v_1 \dots v_h\}$, language model $\mathcal{M}$ with vocabulary $\mathcal{V}$, informative layer l and hyperparameter α, max token len T, threshold τ.
Output: Continuation $\mathcal{G} = \{x_{h+1} \dots x_{h+n}\}$
$\mathcal{G} \leftarrow \{\}$
▷ follow the normal LLM decoding process, and save the intermediate results for use
Use LLM $\mathcal{M}$ to transform the in-context tokens and save the sequence of hidden states $\{\mathbf{x}_1^l, \dots, \mathbf{x}_h^l\}$ at layer l
▷ pre-compute entropy for all tokens in $\mathcal{V}$; note that the following for-loop can be run in parallel
for $v_t \in \mathcal{V}$ **do**
  for $v_j \in \mathcal{C}$ **do**
    $P(v_t \mid v_{\leq j}) = \mathrm{softmax}\big(\phi(\mathbf{x}_j^l)\big)_{v_t}$ ▷ compute activate score $s(i, j)$
  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 v_t's entropy to all tokens in $\mathcal{C}$
end for

▷ generate tokens sequentially using activation decoding
$t = h + 1$
while stop token not generated **and** $t \leq T + h$ **do**
  $q_v = \mathrm{softmax}\big(\phi(\mathbf{x}_i^H)\big)$ ▷ obtain the next token probability distribution
    ▷ only select high probability tokens for adjustment
  for $v_t \in \{v_i | q_v(v_i) \geq \tau \max_w q_v(w)\}$ **do**
    $P_q(v_t \mid v_{<t}) = e^{-\lambda E(v_t|v_{\leq h})} P_q(v_t \mid v_{<t})$ ▷ use entropy to adjust the next token probability distribution
  end for
  $x_t = \mathbf{argmax}_{v \in \mathcal{V}} P_q(v|v_{<t})$
  $\mathcal{G} \leftarrow \mathcal{G} \cup \{x_t\}$
end while