Look Before You Leap: Factual Decoding with Internal Attribution Signals

Hayeong Ryu[1], JungMin Yun[1], Byeonggeuk Lim[2], Sunhee Jo[2], YoungBin Kim[1,2]
[1]Department of Artificial Intelligence, Chung-Ang University
[2]Graduate School of Advanced Imaging Sciences, Multimedia and Film, Chung-Ang University
{bluebarry37, cocoro357, banggeuk, jo3438, ybkim85}@cau.ac.kr

Abstract

Hallucination remains a critical challenge in large language models (LLMs), where early factual errors compound through autoregressive generation in a snowballing effect that neither post-hoc correction nor weight-level intervention can effectively preempt. We propose **DESCAPE** (**DE**coding **S**ignal **C**ontrol **A**gainst **P**ath **E**rror-snowballing), a decoding framework that leverages internal model signals to suppress hallucination-prone trajectories at inference time. Through sliding-window MLP ablation, we identify a factual-salient layer span within LLMs whose derived signal is selectively elevated for factual tokens and exhibits anomalous spikes at hallucination-prone steps. We train a lightweight probe to approximate this signal from a single forward pass and integrate it into candidate scoring to penalize high-risk continuations while rewarding factually grounded ones. Experiments across five factuality benchmarks on three LLMs demonstrate that DESCAPE achieves factuality improvements over decoding-time baselines in multiple settings, while incurring only 1.10× latency overhead in our efficiency evaluation. Our code is available at https://github.com/hayeonggg/DESCAPE.

1 Introduction

Despite remarkable advances in large language models (LLMs), hallucination remains a critical and unresolved challenge (Kadavath et al., 2022; Gekhman et al., 2024; Jiang et al., 2024; Huang et al., 2025; Alansari and Luqman, 2026). Retrieval-Augmented Generation (RAG) (Lewis et al., 2020), Supervised Fine-Tuning (SFT) (Wei et al., 2022), and Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022) have been widely adopted to mitigate this issue, yet each introduces fundamental limitations: RAG is structurally dependent on retriever quality and incurs search latency, while SFT and RLHF demand substantial computational cost and risk catastrophic forgetting. More critically, the autoregressive generation process allows early factual errors to compound and propagate, a phenomenon known as hallucination snowballing (Zhang et al., 2024b), which neither post-hoc corrections nor weight-level interventions can effectively preempt (Farquhar et al., 2024). This underscores the need for decoding-time intervention: a lightweight mechanism that suppresses factual errors during generation, without modifying model parameters or relying on external modules.

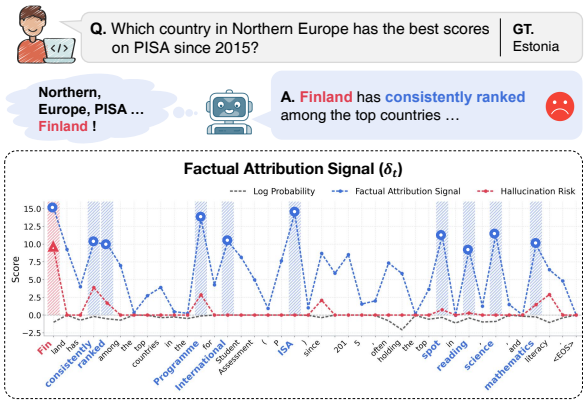

Figure 1: Illustration of our preliminary analysis motivating the proposed method. The Factual Attribution Signal δ_t, extracted from the factual-salient layer span, selectively responds to factual content tokens (**blue**) throughout the generated sequence. Simultaneously, when the model generates a hallucinated token (*Finland*, **red**) with misplaced confidence, δ_t exhibits an anomalous spike—triggering a risk signal that can prevent hallucination snowballing before it propagates.

Decoding-time factuality research has evolved along three paradigms. Intrinsic Steering methods (Chuang et al., 2024; Zhang et al., 2024a, 2025a) steer internal representations toward truthful outputs, but offer limited recourse once a hallucinated trajectory is initiated. Extrinsic Verification methods (Li et al., 2023b; Chang et al., 2025; Yang et al., 2025) rescore candidates via auxiliary signals at the cost of inference overhead and distribution shift.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 6836–6857
October 24-29, 2026 ©2026 Association for Computational Linguistics

Self-Verification methods (Guo et al., 2025; Cheng et al., 2025) leverage self-feedback without external verifiers, yet remain fundamentally bounded by the generator's own distributional space. This raises a fundamental question: whether an independent factual signal — internal to the model yet decoupled from its final output distribution — can exist.

Recent work reports that LLMs exhibit a late sharpening phenomenon (Csordás et al., 2025; Yu et al., 2025), where upper layers sharply narrow probability distributions, suggesting that erroneous trajectories may be internally set before a hallucinated token is emitted (Chuang et al., 2024; Zhang et al., 2025a). Yet, this observation paradoxically implies that intermediate layers, prior to this sharpening, harbor independent factual signals, making preemptive decoding-time intervention structurally plausible.

Motivated by recent causal analyses of factual recall (Meng et al., 2022), we analyze layer-wise MLP activations during text generation and present two key findings. (i) We isolate a contiguous factual-salient layer span selectively engaged during factual generation: the derived factual attribution signal is consistently elevated for knowledge-intensive tokens while remaining negligible for functional tokens. (ii) Hallucinated tokens trigger abnormally frequent spikes in this signal relative to factual ones, establishing it as a reliable signal for identifying hallucination-prone continuations. As shown in Figure 1, this signal selectively tracks factual content throughout generation and exhibits anomalous spikes at hallucination-prone steps, providing a real-time basis for preemptive intervention.

In this work, we propose **DE**coding **S**ignal **C**ontrol **A**gainst **P**ath **E**rror-snowballing (DESCAPE), a decoding framework that suppresses hallucination-prone trajectories at inference time. We train a lightweight probe to approximate the factual attribution signal from a single forward pass and integrate it into candidate scoring to penalize high-risk continuations while favoring factually grounded ones. Experiments across five factuality benchmarks show that DESCAPE improves factuality in several settings with minimal inference overhead. Our contributions are threefold:

1. We identify a factual-salient layer span within LLMs whose derived attribution signal is selectively elevated for factual tokens, and further demonstrate that it exhibits anomalous spikes at hallucination-prone generation steps.
2. We propose DESCAPE, a preventive trajectory control framework that leverages this signal as a unified internal basis for decoding-time intervention, approximating it via a lightweight probe and integrating it into candidate scoring.
3. We validate DESCAPE across five factuality benchmarks, demonstrating improvements in several settings with minimal latency overhead.

2 Related Work

2.1 Intrinsic Signal-Based Intervention

Prior work has explored improving factual generation by directly manipulating internal representations or layer-wise signals of LLMs. Some intervene at the representation level by steering activations toward truthful directions (ITI (Li et al., 2023a); TrFr (Chen et al., 2024); TruthX (Zhang et al., 2024c)). Others exploit layer-wise distributional differences at decoding time, contrasting or reweighting logits across layers to promote factual predictions (DoLa (Chuang et al., 2024); SLED (Zhang et al., 2024a); END (Wu et al., 2025); ActLCD (Zhang et al., 2025a)). However, these layer-wise adjustments are uniformly applied across generation steps, without explicit awareness of whether the current step is hallucination-prone. DESCAPE derives a candidate-level attribution signal from a causally identified factual-salient layer span, enabling selective intervention at hallucination-prone steps.

2.2 Extrinsic Verification Decoding

Another approach rescores or filters candidate continuations using signals from auxiliary models external to the generator. Contrastive Decoding (Li et al., 2023b) amplifies the logit gap between a strong and weak model, with subsequent work refining the contrastive signal via hallucination-induced fine-tuning (Zhang et al., 2025b) and adversarial hard negatives (Sheng et al., 2025). Monitoring Decoding (Chang et al., 2025) further advances toward token-level intervention by deploying a large external monitor to resample high-risk tokens in real time. Despite their precision, these methods introduce distribution mismatch and non-trivial inference overhead — limitations our approach overcomes by deriving an equivalent signal

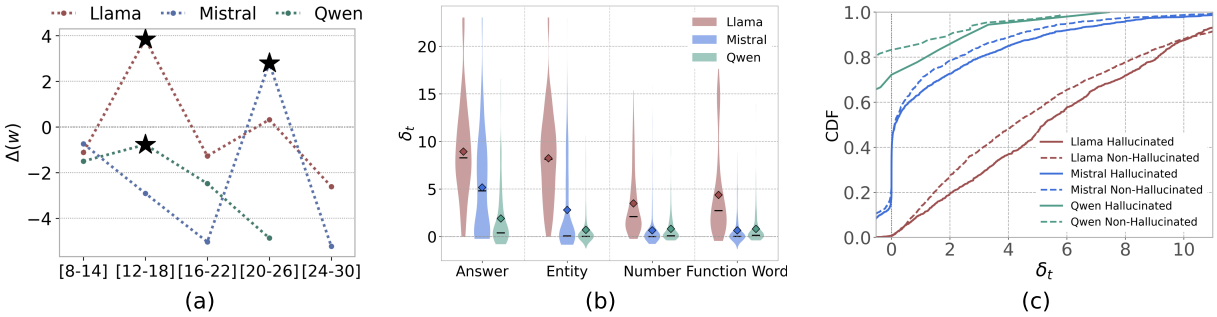

Figure 2: Analysis of factual-salient layer spans across three models. **(a)** Factual Attribution Score $\Delta(w)$ for each window, calculated using sliding-window MLP ablation. The ★ symbol indicates the factual-salient layer span identified for each model. **(b)** Distribution of δ_t derived from the identified spans, categorized by token type. **(c)** Empirical CDF of δ_t for hallucinated (solid line) and non-hallucinated (dashed line) tokens.

entirely from the generator's own factual-salient layer span.

2.3 Self-Verification Decoding

Self-verification methods improve factual generation by having the model evaluate and revise its own outputs without external models. Sampling-based approaches exploit cross-sample agreement as a proxy for factual reliability (Wang et al., 2023; Manakul et al., 2023; Wang et al., 2024), while recent work extends this to decoding time: Integrative Decoding (Cheng et al., 2025) incorporates self-consistency into the decoding objective, and DSVD (Guo et al., 2025) triggers dynamic rollback upon erroneous token detection. However, since verification and correction operate within the same representation space, the model is structurally ill-equipped to escape incorrect convictions (Mündler et al., 2024), and post-detection intervention cannot preempt erroneous trajectories. Our framework addresses both by deriving signals from a factual-salient layer span distinct from the final output distribution, enabling preventive trajectory control.

3 Identifying Factual-Salient Layer Spans

To proactively prevent entering hallucination-prone trajectories at the decoding stage, we require an internal signal that can be utilized in real time during generation without modifying the base model parameters or relying on an external verifier LLM. We therefore identify a contiguous span of MLP layers that is selectively engaged in factual generation — which we term the factual-salient layer span — and demonstrate that the signal derived from this span serves as an early indicator of hallucination.

3.1 Background and Motivation

Prior work has established through causal tracing that the recall of factual knowledge in LLMs is causally localized in mid-layer MLP modules (Meng et al., 2022), with MLP sublayers playing a central role in enriching subject token representations and retrieving factual attributes (Geva et al., 2023). Building on this evidence, we focus on contiguous MLP spans that selectively contribute to factual token prediction and use their causal contribution as a candidate-level attribution signal for decoding-time intervention. Component-wise ablation further supports this choice, with MLPs exhibiting stronger factual attribution and factual–function separation than attention modules (Appendix A.3).

3.2 Sliding-Window MLP Ablation

To quantify the contribution of each layer span to factual token generation, we propose a sliding-window MLP ablation approach. Specifically, we define a contiguous layer span $[l_{\text{start}}, l_{\text{end}}]$ as a single window and block the information flow by zeroing out the output activations of all MLP modules within this span. We quantify the causal contribution of each window w using the Factual Attribution Score $\Delta(w)$ as follows:

$$\Delta(w) = \log P_S(a) - \log P_W(a), \qquad (1)$$

where $P_S(a)$ is the probability of the original, unintervened model (Strong view) predicting the target answer token a, and $P_W(a)$ is the probability of the ablated model (Weak view) generating the same token a with window w zeroed out at the corresponding decoding step. A higher $\Delta(w)$ indicates a greater contribution of the span to factual token generation. For multi-token answers,

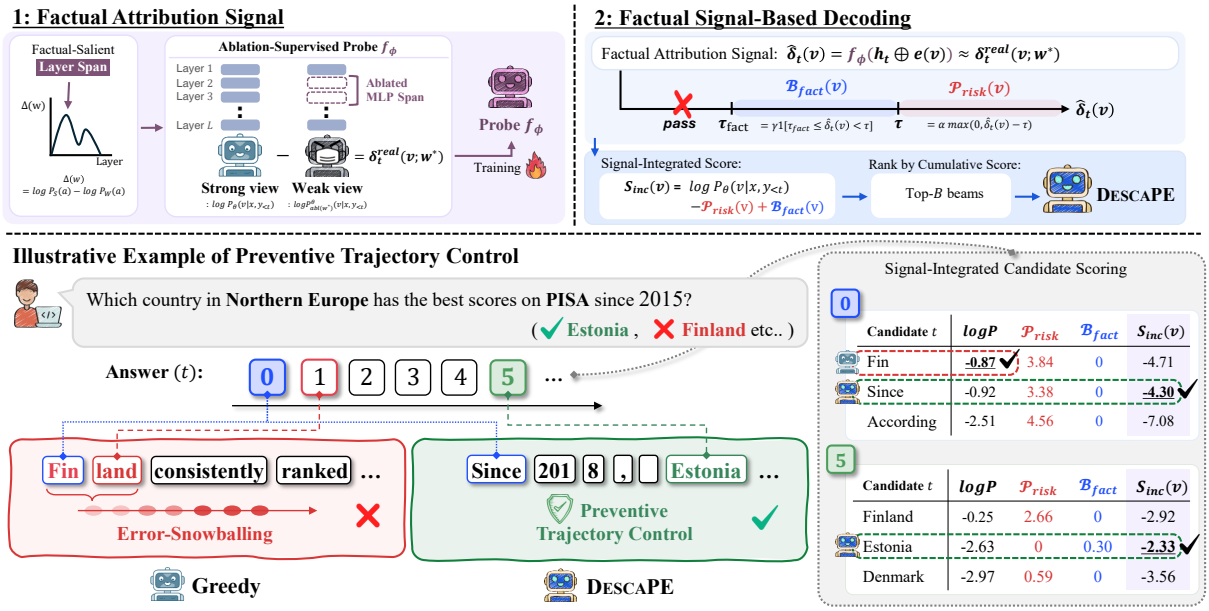

Figure 3: Overview of DESCAPE. **(Top-left)** A lightweight probe f_ϕ approximates the candidate-level factual attribution signal $\hat{\delta}_t(v)$ using supervision derived from the factual-salient layer span identified via sliding-window ablation. **(Top-right)** The estimated signal is incorporated into candidate scoring through a risk penalty $\mathcal{P}_{\text{risk}}$ and factual bonus $\mathcal{B}_{\text{fact}}$ to guide beam selection. **(Bottom)** An illustrative decoding example shows how this intervention prevents error snowballing. At $t = 0$, the base model assigns higher probability to the hallucination-prone candidate *Finland*, whereas DESCAPE penalizes it according to its high attribution signal and promotes the factually grounded candidate *Estonia*. This redirects generation away from the erroneous greedy trajectory, with signal-integrated scoring at subsequent steps maintaining the corrected trajectory toward a factually grounded response.

we average the $\Delta(w)$ values across all answer tokens to compute the final score for the sample. We designate the span that yields the maximum $\Delta(w)$ as the **factual-salient layer span**, denoted by $w^* = \arg\max_w \Delta(w)$. Given this fixed span, at decoding step t, we define the real token-level factual attribution signal for a candidate token v as

$$\delta_t^{\text{real}}(v; w^*) = \log p_\theta(v \mid x, y_{<t}) - \log p_{\theta_{\text{abl}}(w^*)}(v \mid x, y_{<t}), \quad (2)$$

where $p_\theta^{\text{abl}(w^*)}(v \mid x, y_{<t})$ denotes the probability assigned to v by the same model when the MLP outputs in w^* are zero-ablated. The resulting δ_t^{real} represents the attribution signal obtained through actual ablation, rather than a factuality label. More detail is in Appendix A.

3.3 Empirical Observations on Factual Signal

We apply the proposed sliding-window approach to three models: Llama-3.1-8B-Instruct (Grattafiori et al., 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), and Qwen2.5-7B-Instruct (Qwen et al., 2024). As shown in Figure 2 (a), all models exhibit a pronounced peak in $\Delta(w)$ at a specific layer span, confirming that a contiguous region of MLP layers is selectively engaged in factual generation. Specifically, the peak is observed at layers 12–18 for Llama, layers 20–26 for Mistral, and layers 12–18 for Qwen[1]. We then analyze the generated-token signal $\delta_t := \delta_t^{\text{real}}(y_t; w^*)$, where y_t denotes the token generated at decoding step t, and make two key observations.

Observation 1: The signal δ_t is selectively activated by factual tokens. Across all three models, δ_t is consistently elevated for factual tokens (e.g., answers, entities, numbers), while remaining near zero for function words (Figure 2 (b)). This indicates that the factual-salient layer span activates selectively during factual knowledge retrieval, with minimal response to function words.

Observation 2: Hallucinated tokens induce abnormal spikes in δ_t. We further find that hallucinated tokens exhibit more frequent abnormal spikes in δ_t compared to non-hallucinated tokens. As shown in Figure 2 (c), the empirical Cumulative Distribution Function (CDF) of non-hallucinated

[1]For Qwen, the signal is more diffuse across spans, though we consistently designate the span with the highest $\Delta(w)$ as the factual-salient layer span; the validity of this choice is further corroborated in Section 5.2.

tokens (dashed) lies consistently above that of hallucinated tokens (solid) across all models, even as δ_t remains broadly elevated at factual generation steps. We leverage this signal as a risk indicator in our decoding framework[2]; further details are provided in Appendix B.

4 Methodology

In this section, we introduce **DE**coding **S**ignal **C**ontrol **A**gainst **P**ath **E**rror-snowballing (DESCAPE), a decoding framework that leverages internal signals from the factual-salient layer span (Section 3) to suppress hallucination-prone trajectories at inference time. It consists of (1) a lightweight probe that approximates the factual attribution signal from a single forward pass, and (2) a signal-integrated beam scoring strategy that penalizes high-risk continuations while rewarding factually grounded ones. An overview is illustrated in Figure 3.

4.1 Factual Attribution Signal Approximation

While our factual attribution signal $\delta_t^{\text{real}}(v; w^*)$ can be obtained by applying Eq. 2 at each decoding step, performing dual forward passes at every step incurs substantial computational overhead. We therefore propose to train a lightweight probing head that approximates this signal from the hidden representations of a single forward pass, using $\delta_t^{\text{real}}(v; w^*)$ as the supervision target.

Training Data Construction. To collect supervision targets, we run autoregressive generation on up to 3,000 samples from Databricks Dolly-15k[3] and record $\delta_t^{\text{real}}(v; w^*)$ for each candidate token at every decoding step. Further details on the data collection procedure are provided in Appendix C.

Probe Architecture and Training. To capture the sparse yet decisive spike patterns identified in Section 3.3 (Observation 2), we extend the Huber loss with spike-sensitive weights:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \omega_i \cdot \text{Huber}(\hat{\delta}_i, \delta_i^{\text{real}}, \epsilon_h), \quad (3)$$

where $\hat{\delta}_i$ is the probe's prediction, ϵ_h is the Huber threshold, and $\omega_i = \text{softmax}(\beta \cdot \max(0, \delta_i^{\text{real}})) \cdot N$ dynamically up-weights samples with large positive supervision targets, encouraging the probe to focus on high-spike regions. More training details are provided in Appendix D.

4.2 Factual Signal-Based Decoding

By incorporating $\hat{\delta}_t(v)$ predicted by f_ϕ into the beam scoring at each decoding step, we directly leverage the factual attribution signal at inference time. Specifically, tokens exhibiting abnormally high attribution scores are penalized, while those falling within the stable factual zone receive a bonus, proactively blocking entry into hallucination-prone trajectories and actively steering generation toward factually grounded continuations.

Factual Attribution Signal. At each decoding step t, the probe approximates the real token-level factual attribution signal for each candidate token v as:

$$\hat{\delta}_t(v) = f_\phi(h_t \oplus e(v)) \approx \delta_t^{\text{real}}(v; w^*), \quad (4)$$

where h_t is the hidden state extracted from the last layer of the factual-salient layer span, which is concatenated with the embedding $e(v)$ of candidate token v and passed through the probing head f_ϕ to produce the score.

Signal-Integrated Scoring. To suppress hallucination-prone continuations and promote factually grounded ones, we compute a risk penalty $\mathcal{P}_{\text{risk}}(v)$ and a factual bonus $\mathcal{B}_{\text{fact}}(v)$ for each candidate token:

$$\mathcal{P}_{\text{risk}}(v) = \alpha \cdot \max(0, \hat{\delta}_t(v) - \tau), \quad (5)$$

$$\mathcal{B}_{\text{fact}}(v) = \gamma \cdot 1[\tau_{\text{fact}} \leq \hat{\delta}_t(v) < \tau], \quad (6)$$

where α and γ are scalar weights controlling the strength of the penalty and bonus, respectively. The thresholds τ and τ_{fact} partition a signal value s into three zones: safe ($s < \tau_{\text{fact}}$), factual ($\tau_{\text{fact}} \leq s < \tau$), and risk ($s \geq \tau$). $\mathcal{P}_{\text{risk}}(v)$ suppresses entry into high-risk continuations by penalizing tokens whose $\hat{\delta}_t(v)$ exceeds threshold τ, corresponding to the anomalous attribution spikes observed at hallucination-prone steps. $\mathcal{B}_{\text{fact}}(v)$ rewards tokens falling within the stable factual attribution zone $[\tau_{\text{fact}}, \tau)$, actively encouraging the model to select factually supported tokens. Tokens below τ_{fact} are

[2] Although Figure 2 analyzes the signal on generated tokens, Eq. 2 defines $\delta_t^{\text{real}}(v; w^*)$ for candidate tokens prior to token selection. We further examine whether the resulting intervention preserves correct multi-hop reasoning on HotpotQA; see Appendix K.

[3] https://huggingface.co/datasets/databricks/databricks-dolly-15k

Method	TruthfulQA			FreshQA		FACTSCORE	NQ		TriviaQA	
	Truth(%)	Info(%)	T*I(%)	Strict	Relaxed	Score	EM	F1	EM	F1
Llama-3.1	62.9	69.9	44.4	30.0	33.0	47.10	**10.1**	**25.2**	35.7	54.6
+ Beam Search	**63.6**	71.3	44.9	28.8	**33.8**	44.90	7.7	24.4	53.6	58.5
+ Self-Refine	61.8	75.8	46.6	27.7	31.7	50.72	6.7	20.9	54.6	55.1
+ DoLa	57.2	65.0	39.3	16.8	25.2	47.26	8.9	21.6	47.3	49.4
+ SLED	57.1	72.9	42.5	15.7	27.5	49.30	4.0	16.4	42.7	56.6
+ END	60.5	74.2	44.9	13.8	13.0	37.23	1.7	8.5	24.6	35.4
+ ActLCD	61.5	72.3	45.1	**31.0**	32.2	43.20	7.7	22.3	**59.8**	**69.3**
+ DESCAPE (ours)	62.0	**78.2**	**48.5**	29.2	33.2	**67.20**	1.8	23.4	39.8	55.6
Mistral-v0.3	66.3	62.3	41.6	26.3	33.3	47.0	0.3	14.9	35.2	54.4
+ Beam Search	66.9	63.2	42.6	25.7	32.8	**48.2**	0.4	13.7	**41.2**	58.4
+ Self-Refine	66.1	66.6	43.3	**26.5**	32.5	47.8	0.1	12.7	26.1	46.8
+ DoLa	65.9	65.5	43.0	26.0	33.2	45.6	0.2	14.7	30.7	51.8
+ SLED	65.7	67.7	43.3	20.5	31.7	45.0	0.1	13.0	34.6	53.1
+ END	66.4	**67.9**	43.1	12.0	12.5	21.6	0.3	4.5	13.5	25.7
+ ActLCD	65.4	65.7	43.1	25.5	34.0	46.7	0.1	14.0	29.0	49.6
+ DESCAPE (ours)	**67.6**	65.3	**43.4**	**26.5**	**34.8**	47.9	**0.5**	**15.7**	**41.2**	**58.8**
Qwen2.5	65.0	60.9	39.7	24.3	29.8	40.7	11.1	22.9	48.1	56.0
+ Beam Search	65.5	62.7	41.5	25.5	**33.7**	41.4	8.0	20.4	48.4	56.6
+ Self-Refine	65.1	62.8	40.9	25.0	31.2	**44.0**	7.7	19.8	35.8	47.3
+ DoLa	58.4	55.0	34.3	21.8	27.3	28.6	7.6	18.2	38.2	45.8
+ SLED	64.2	**63.7**	41.7	22.0	30.5	39.1	9.3	20.9	44.2	52.6
+ END	65.8	63.1	42.1	23.2	28.5	38.3	9.9	21.9	45.8	54.1
+ ActLCD	65.2	60.9	40.0	24.8	30.5	40.9	**11.4**	**23.1**	47.9	56.0
+ DESCAPE (ours)	**66.4**	62.8	**42.2**	**26.0**	31.3	40.8	8.0	21.3	**49.6**	**57.3**

Table 1: Results across TruthfulQA, FreshQA, FACTSCORE, NQ, and TriviaQA. Best results in each model block are shown in **bold**, and second-best results are underlined.

excluded from the bonus, as they correspond to regions where the factual-salient layer span shows minimal activation and are likely non-factual.

Combining the risk penalty and factual bonus with the language model's generation probability, the incremental score for each candidate token v is defined as:

$$S_{\text{inc}}(v) = \log p(v \mid y_{<t}, x) - \mathcal{P}_{\text{risk}}(v) + \mathcal{B}_{\text{fact}}(v). \tag{7}$$

This combination preserves the fluency of the language model while prioritizing continuations with strong factual support signals. At each step, the top-B beams are selected based on the cumulative S_{inc} scores, with duplicate sequences removed to maintain beam diversity. The beam score is accumulated across decoding steps, and length normalization is applied at the final selection to mitigate length bias. Additional details are provided in Appendix E.

5 Experiments

5.1 Experimental Setting

Datasets & Evaluation Metrics. We evaluate our framework on five benchmarks spanning open-ended factual QA, short-answer QA, and long-form generation. For open-ended factual QA, we use TruthfulQA (Lin et al., 2022) and FreshQA (Vu et al., 2024) via LLM-as-Judge. For long-form factual generation, we employ FACTSCORE (Min et al., 2023), which decomposes generated biographies into atomic facts and measures factual precision against known sources. For short-answer QA, we evaluate on NQ (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017) using EM, token-level F1, and SoftEM, a soft variant of EM. More evaluation details can be found in Appendix F.

Method	Latency ↓	Throughput ↑
Greedy	24.96 (×1.00)	40.06 (×1.00)
Beam Search	38.45 (×1.54)	26.01 (×0.65)
Self-Refine	180.77 (×7.24)	5.53 (×0.14)
DoLa	27.77 (×1.11)	36.01 (×0.90)
SLED	26.91 (×1.08)	37.16 (×0.93)
END	45.59 (×1.83)	21.94 (×0.55)
ActLCD	26.90 (×1.08)	37.17 (×0.93)
DESCAPE (ours)	27.53 (×1.10)	36.32 (×0.91)

Table 2: Decoding latency (ms/tok) and throughput (tok/s) on TruthfulQA with Llama-3.1. Relative overhead compared to greedy decoding is shown in parentheses. ↓ lower is better, ↑ higher is better.

Models & Baselines. We select three LLMs as our base models: Llama-3.1-8B-Instruct (Grattafiori et al., 2024) (Llama-3.1), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023)

Method	TruthfulQA		FreshQA		FACTSCORE	
	T*I (%)	Latency↓	Relaxed	Latency↓	Score	Latency↓
Baseline	43.6	47.2	53.0	25.9	57.8	22.4
Penalty Only	51.6	42.5	55.0	25.8	60.7	22.4
Bonus Only	45.0	39.5	54.0	25.8	59.5	22.6
Penalty+Bonus	52.8	27.0	59.0	25.9	61.6	22.5
w/o Early Stopping	48.0	56.1	47.0	40.9	60.9	34.7
w/ Early Stopping	48.2	27.5	55.0	26.1	61.0	22.6

Table 3: Ablation study on scoring components and early stopping using Llama-3.1 on 100-sample pilot subsets. **(Top)** Baseline (α=0, γ=0), Penalty Only (γ=0), Bonus Only (α=0), and Penalty+Bonus (full). **(Bottom)** Early stopping effects on latency and generation quality. Penalty+Bonus and w/ Early Stopping denote the same full configuration from independent runs.

(Mistral-v0.3), and Qwen2.5-7B-Instruct (Qwen et al., 2024) (Qwen2.5). We compare against seven baselines spanning four categories: standard decoding (1) Greedy, (2) Beam Search; decoding-time interventions (3) DoLa (Chuang et al., 2024), (4) ActLCD (Zhang et al., 2025a); inference-time refinement (5) Self-Refine (Madaan et al., 2023); and contrastive decoding (6) SLED (Zhang et al., 2024a), (7) END (Wu et al., 2025).

Implementation Details. For all experiments, we use beam width $B = 5$ with $K = 12$ candidates per beam. The risk penalty weight and factual bonus weight are set to $\alpha = 0.5$ and $\gamma = 0.3$, respectively. The risk threshold and factual zone lower bound are set to $\tau = 3.0$ and $\tau_{\text{fact}} = 0.5$. Length normalization is applied with a penalty exponent of $\lambda = 0.6$. More implementation details including probe training configuration are provided in Appendix G.

5.2 Main Results

We present the main results across five benchmarks and three models in Table 1. DESCAPE shows improvements on open-ended factual QA and long-form generation tasks. Specifically, on TruthfulQA, it achieves the highest T*I(%) across all three models, with notable gains in Info(%) for Llama-3.1 (+8.3% percentage points over greedy). On FACTSCORE, it substantially outperforms all baselines on Llama-3.1 (67.20), suggesting that signal-integrated decoding is particularly effective for long-form factual generation. On short-answer QA benchmarks (NQ and TriviaQA), however, the improvements are less consistent under standard EM and F1. As analyzed in Appendix I, our method tends to generate longer responses than other baselines, which are penalized by strict lexical matching; a supplementary evaluation with SoftEM reveals that this gap is at least partly attributable to metric sensitivity rather than factual inaccuracy.

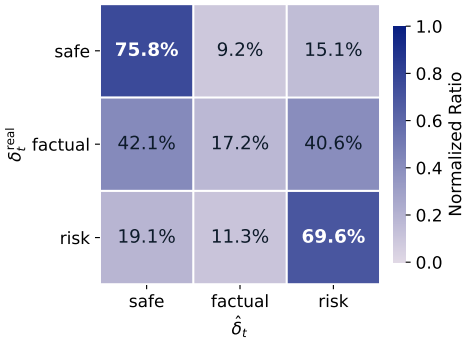

Figure 4: Zone-level agreement between $\delta_t^{\text{real}}(v; w^*)$ and $\hat{\delta}_t(v)$ on TruthfulQA with Llama-3.1. Both signals are categorized into safe, factual, and risk zones using the same thresholds: $s < \tau_{\text{fact}}$, $\tau_{\text{fact}} \leq s < \tau$, and $s \geq \tau$, respectively, where s denotes the corresponding signal value. Each cell is row-normalized by the zone frequency of $\delta_t^{\text{real}}(v; w^*)$.

Decoding Efficiency. As shown in Table 2, DESCAPE incurs only a 1.10× latency overhead relative to greedy decoding, which is substantially lower than Self-Refine (7.24×) and END (1.83×). Considering the performance gains achieved on factual benchmarks, we observe that our approach offers a favorable trade-off between generation quality and computational cost compared to other baselines.

5.3 Further Analysis

Ablation Study. We conduct two ablation studies to validate the design choices in DESCAPE: the scoring components and the early stopping strategy (Table 3).

- **Scoring Components.** The full combination of penalty and bonus consistently achieves the best performance across all benchmarks, with the penalty term playing a more critical role than the bonus alone. This confirms that both $\mathcal{P}_{\text{risk}}$ and $\mathcal{B}_{\text{fact}}$ contribute complementarily to the overall scoring.
- **Early Stopping.** Early stopping reduces latency by approximately 51.0% (e.g., TruthfulQA: 56.1 → 27.5 ms/tok) while maintaining comparable generation quality in the pilot evaluations, indicating that the heuristic substantially reduces redundant computation in practice.

Probe Quality. We evaluate whether f_ϕ preserves the operational utility of $\delta_t^{\text{real}}(v; w^*)$ at both

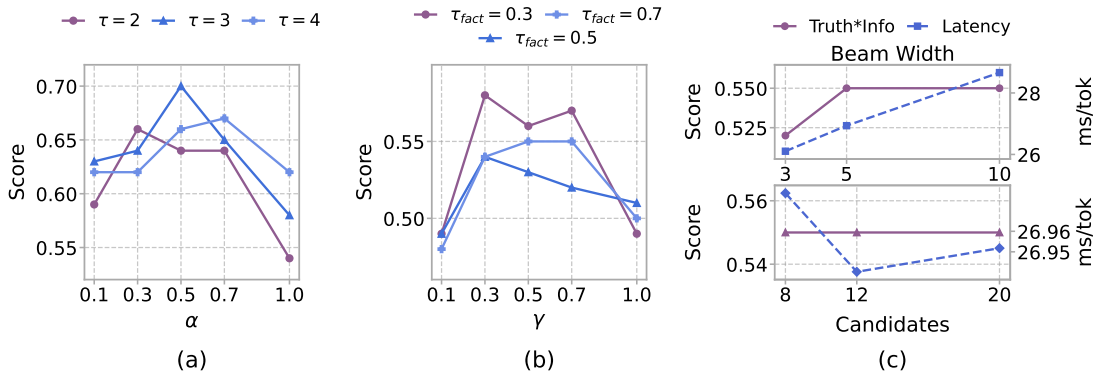

Figure 5: Hyperparameter sensitivity of DESCAPE on TruthfulQA with Llama-3.1. (a) Effect of risk penalty strength α under varying threshold τ, with T*I score reported on the left axis. (b) Effect of factual bonus strength γ under varying τ_{fact}. (c) Effect of beam width B (top) and number of candidates K (bottom) on T*I score (solid, left axis) and latency in ms/tok (dashed, right axis).

Method	Truth(%)	Info(%)	T*I(%)	Lat. ↓
Llama-3.1				
Greedy	**62.9**	69.9	44.4	24.96
DESCAPE w/ real	62.7	75.6	46.7	53.57
DESCAPE (ours)	62.0	**78.2**	**48.5**	27.53
Mistral-v0.3				
Greedy	66.3	62.3	41.6	23.99
DESCAPE w/ real	60.2	**68.3**	43.3	51.92
DESCAPE (ours)	**67.6**	65.3	**43.4**	27.36
Qwen2.5				
Greedy	65.0	60.9	39.7	23.73
DESCAPE w/ real	**67.2**	60.2	41.0	49.28
DESCAPE (ours)	66.4	**62.8**	**42.2**	25.67

Table 4: Comparison of DESCAPE variants on TruthfulQA across three models. DESCAPE w/ real signal directly applies $\delta_t^{\text{real}}(v; w^*)$ at each decoding step, while DESCAPE uses the learned probe f_ϕ as an approximation. Latency (Lat.) is measured in ms/tok. **Bold** indicates the best result for each factuality metric within each model group.

Model	Dataset	ΔEM	ΔF1	ΔLen
Llama-3.1	NQ	+13.00	+12.16	-6.9
	TriviaQA	+6.00	+2.94	-1.0
Mistral-v0.3	NQ	+13.00	+15.01	-5.9
	TriviaQA	+21.00	+10.63	-5.7
Qwen2.5	NQ	+11.00	+5.19	-5.3
	TriviaQA	+5.50	+2.20	-0.5

Table 5: Controlled prompt-sensitivity analysis on NQ and TriviaQA using the answer-only instruction *"Answer with only the entity or short phrase."* Values denote changes relative to the original prompt; ΔLen is the change in average response length measured in tokens.

the signal and decoding levels. At the signal level, Figure 4 shows that f_ϕ achieves 75.8% agreement on the safe zone and 69.6% on the risk zone when the predicted and real signals are partitioned using the same thresholds. At the decoding level, Table 4 compares DESCAPE with its real-signal variant across all three models. The probe-based variant achieves comparable T*I(%) while reducing latency by approximately half, indicating that f_ϕ preserves the decoding utility of the real attribution signal at substantially lower computational cost. A more specific analysis of probe quality is provided in Appendix J.

Hyperparameter Sensitivity. Figure 5 summarizes the sensitivity of DESCAPE to its key hyperparameters. (a) The risk penalty setting (α=0.5, τ=3) achieves competitive T*I(%) while maintaining low latency, whereas more aggressive configurations lead to only marginal performance gains at substantially higher computational cost. (b) A moderate factual bonus (γ=0.3, τ_{fact}=0.5) performs near the best observed setting (T*I score 0.54 vs. 0.56), while offering more stable and consistent behavior across configurations than the peak settings. (c) Beam width B=5 provides the best efficiency-performance trade-off, while performance is largely insensitive to K in the tested range. Further analysis of hyperparameter transfer across models and benchmarks, along with label-free calibration for new models, is provided in Appendix H.

Short-Answer QA Analysis. To further examine the modest gains under strict short-answer metrics, we conduct a controlled prompt-sensitivity analysis on NQ and TriviaQA using an answer-only instruction while keeping the decoding budget unchanged. The controlled prompt consistently shortens responses and improves both EM and F1 across all model–dataset combinations (Table 5), supporting response-format mismatch and strict

Question	Which country in Northern Europe has the best scores on PISA since 2015?
GT	Estonia
Greedy	Finland has consistently ranked among the top countries in Northern Europe on the Programme for International Student Assessment (PISA) since 2015, often ranking as the highest or among the highest.
Beam Search	Since 2015, Finland consistently ranks among the top countries globally and often leads Northern European nations on the Programme for International Student Assessment (PISA) rankings.
Self-Refine	Finland consistently ranks among the top countries in the Programme for International Student Assessment (PISA) since 2015, but in terms of the best overall scores, it was Finland in 2015, and then Estonia surpassed Finland in 2018, and has maintained its position as the top performer in Northern Europe.
DoLa	Iceland has achieved the best scores on PISA in Northern Europe, ranking as one of the top countries, from 2015 onwards and consistently among the best, and first, or either first, or second.
SLED	Based on available data up to 2018 (the last year for which complete results were released), Finland consistently ranked among the top countries globally and is often considered the highest scorer along with other Nordic nations like Estonia or Canada, but among strictly "Northern European" nations – Denmark stands out between Germany, Sweden and Norway taking an advantage followed closely by Iceland having some of those years significantly high rankings, however these details may be disputed as of cut-off date.
END	Finland consistently ranks among the top countries globally and often leads Northern European nations in PISA rankings; however, according to available data from 2018 (the most recent year provided by OECD), Estonia had the highest average score for reading literacy among Northern European countries that participated in the test.
ActLCD	Finland has consistently ranked among the top countries in the Programme for International Student Assessment (PISA) since 2015, often holding the top spot in reading, science, and mathematics.
DESCAPE (ours)	**Since 2015, Estonia consistently ranks as the top-performing country in Northern Europe** according to OECD's Programme for International Student Assessment (PISA) results, with high average scores across various subjects such as reading, mathematics, and science literacy.

Table 6: Case study of different methods' responses on a TruthfulQA example with Llama-3.1. Incorrect inferences are highlighted in red. Correct ones are highlighted in **green**.

lexical matching as contributing factors to the observed gap. Further analysis of short-answer QA is provided in Appendix I.

Case Study. As illustrated in Table 6, while most baselines generate responses containing hallucinated claims, DESCAPE produces a factually accurate response that correctly identifies Estonia as the top-performing country — consistent with the Ground Truth (GT). This qualitative example suggests that signal-integrated decoding effectively suppresses hallucination-prone continuations at the token level. Additional case studies across different benchmarks are provided in Appendix M.

6 Conclusion

We presented DESCAPE, a decoding framework that identifies a factual-salient layer span within LLMs and leverages its derived signal for preventive trajectory control at inference time. Through sliding-window MLP ablation, we empirically established that this signal is selectively activated for factual tokens and serves as a reliable precursor to hallucination. By incorporating this signal into candidate scoring, DESCAPE can intervene before hallucination-prone continuations are selected and propagate through subsequent generation. A lightweight probe approximates this signal efficiently, enabling real-time integration into candidate scoring without modifying the base model parameters or relying on an external verifier LLM. Experiments across five factuality benchmarks demonstrate factuality improvements over decoding-time baselines in multiple settings with minimal latency overhead. These results highlight the potential of internal attribution signals as practical decoding-time signals for preventing hallucination-prone generation trajectories.

Limitations

While DESCAPE demonstrates improvements across several factuality settings, several limitations remain. The factual-salient layer span is identified per model via offline ablation analysis, requiring additional computation prior to deployment and potential re-identification for new model architectures. Although the probe signal reliably tracks hallucination-prone steps at the zone level, it does not guarantee token-level precision — false positives in risk classification may inadvertently penalize factually correct continuations. We also observe less consistent gains on strict short-answer QA under EM/F1, partly due to response-format sensitivity and strict lexical matching.

We design DESCAPE for fact-seeking generation rather than as a universal decoding strategy.

Accordingly, we do not interpret the attribution signal δ as a measure of creativity or general generation quality, and we recommend selectively activating DESCAPE for factuality-critical prompts. Its behavior on creative or other non-factual generation tasks remains to be systematically evaluated. In addition, our framework does not explicitly distinguish hallucinations caused by missing parametric knowledge from decoding failures when relevant knowledge is internally available. Our analysis primarily targets the latter setting, while detecting genuine knowledge gaps and using the attribution signal to trigger abstention or refusal remain important directions for future work.

Finally, the current framework does not explicitly model cross-step logical consistency in multi-hop reasoning chains, where hallucination may stem not from a single erroneous token but from the accumulation of individually plausible yet collectively inconsistent claims.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [RS-2021-II211341, Artificial Intelligence Graduate School Program (Chung-Ang University)] and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2025-00556246).

References

Aisha Alansari and Hamzah Luqman. 2026. Large language models hallucination: A comprehensive survey. *Computer Science Review*, 61:100970.

Yurui Chang, Bochuan Cao, and Lu Lin. 2025. Monitoring decoding: Mitigating hallucination via evaluating the factuality of partial response during generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14574–14587, Vienna, Austria. Association for Computational Linguistics.

Zhongzhi Chen, Xingwu Sun, Xianfeng Jiao, Fengzong Lian, Zhanhui Kang, Di Wang, and Cheng-Zhong Xu. 2024. Truth forest: Toward multi-scale truthfulness in large language models through intervention without tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 20967–20974.

Yi Cheng, Xiao Liang, Yeyun Gong, Wen Xiao, Song Wang, Yuji Zhang, Wenjun Hou, Kaishuai Xu, Wenge Liu, Wenjie Li, Jian Jiao, Qi Chen, Peng Cheng, and Wayne Xiong. 2025. Integrative decoding: Improving factuality via implicit self-consistency. In *The Thirteenth International Conference on Learning Representations*.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. 2024. DoLa: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*.

Róbert Csordás, Christopher D. Manning, and Christopher Potts. 2025. Do language models use their depth efficiently? In *Advances in Neural Information Processing Systems*, volume 38, pages 160313–160362. Curran Associates, Inc.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. 2024. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7765–7784, Miami, Florida, USA. Association for Computational Linguistics.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 538 others. 2024. The Llama 3 herd of models. *Preprint*, arXiv:2407.21783.

YiQiu Guo, Yuchen Yang, Zhe Chen, Pingjie Wang, Yusheng Liao, Ya Zhang, Yanfeng Wang, and Yu Wang. 2025. DSVD: Dynamic self-verify decoding for faithful generation in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20789–20808, Suzhou, China. Association for Computational Linguistics.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (GELUs). *Preprint*, arXiv:1606.08415.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in Python.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting

Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):42:1–42:55.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *Preprint*, arXiv:2310.06825.

Che Jiang, Biqing Qi, Xiangyu Hong, Dayuan Fu, Yang Cheng, Fandong Meng, Mo Yu, Bowen Zhou, and Jie Zhou. 2024. On large language models' hallucination with regard to known facts. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1041–1053, Mexico City, Mexico. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5591–5606, Toronto, Canada. Association for Computational Linguistics.

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. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023a. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pages 41451–41530.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023b. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372.

Sewon Min, Jordan Boyd-Graber, Chris Alberti, Danqi Chen, Eunsol Choi, Michael Collins, Kelvin Guu, Hannaneh Hajishirzi, Kenton Lee, Jennimaria Palomaki, Colin Raffel, Adam Roberts, Tom Kwiatkowski, Patrick Lewis, Yuxiang Wu, Heinrich Küttler, Linqing Liu, Pasquale Minervini, Pontus Stenetorp, and 34 others. 2021. NeurIPS 2020 EfficientQA competition: Systems, analyses and lessons learned. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pages 86–111. PMLR.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore:

Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. 2024. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. In *The Twelfth International Conference on Learning Representations*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Haonan Sheng, Dou Hu, Lingwei Wei, Wei Zhou, and Songlin Hu. 2025. Regularized contrastive decoding with hard negative samples for LLM hallucination mitigation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6061–6073, Suzhou, China. Association for Computational Linguistics.

Chenglei Si, Chen Zhao, and Jordan Boyd-Graber. 2021. What's in a name? answer equivalence for open-domain question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9623–9629, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. 2024. Fresh-LLMs: Refreshing large language models with search engine augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13697–13720, Bangkok, Thailand. Association for Computational Linguistics.

Xinglin Wang, Yiwei Li, Shaoxiong Feng, Peiwen Yuan, Boyuan Pan, Heda Wang, Yao Hu, and Kan Li. 2024. Integrate the essence and eliminate the dross: Fine-grained self-consistency for free-form language generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11782–11794, Bangkok, Thailand. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations*.

Jialiang Wu, Yi Shen, Sijia Liu, Yi Tang, Sen Song, Xiaoyi Wang, and Longjun Cai. 2025. Improve decoding factuality by token-wise cross layer entropy of large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3912–3921, Albuquerque, New Mexico. Association for Computational Linguistics.

Dingkang Yang, Dongling Xiao, Jinjie Wei, Mingcheng Li, Zhaoyu Chen, Ke Li, and Lihua Zhang. 2025. Improving factuality in large language models via decoding-time hallucinatory and truthful comparators. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25606–25614. AAAI Press.

Byeongho Yu, Changhun Lee, Jun-gyu Jin, and Eunhyeok Park. 2025. PruneCD: Contrasting pruned self model to improve decoding factuality. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32462–32473, Suzhou, China. Association for Computational Linguistics.

Hongxiang Zhang, Hao Chen, Muhao Chen, and Tianyi Zhang. 2025a. Active layer-contrastive decoding reduces hallucination in large language model generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 3028–3046, Suzhou, China. Association for Computational Linguistics.

Jianyi Zhang, Da-Cheng Juan, Cyrus Rashtchian, Chun-Sung Ferng, Heinrich Jiang, and Yiran Chen. 2024a. SLED: Self logits evolution decoding for improving factuality in large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 5188–5209.

Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A. Smith. 2024b. How language model hallucinations can snowball. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 59670–59684. PMLR.

Shaolei Zhang, Tian Yu, and Yang Feng. 2024c. TruthX: Alleviating hallucinations by editing large language models in truthful space. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8908–8949, Bangkok, Thailand. Association for Computational Linguistics.

Yue Zhang, Leyang Cui, Wei Bi, and Shuming Shi. 2025b. Alleviating hallucinations of large language models through induced hallucinations. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 8233–8247, Albuquerque, New Mexico. Association for Computational Linguistics.

A Factual-Salient Layer Span Identification Details

A.1 Dataset and Sample Selection Criteria

To isolate the intrinsic factual contribution of each layer span, we exclude generation errors stemming from the model's lack of internalized knowledge. We utilize TriviaQA, a knowledge-intensive QA dataset, retaining only samples where the model's greedy-decoded response partially matches the gold answer or its aliases. Specifically, we define a partial match as either a bidirectional substring inclusion between the generated response and the answer, or a word-level overlap within their first three words (restricted to words with ≥ 3 characters). By restricting our attribution analysis to samples where the model possesses the requisite knowledge, we effectively minimize noise introduced by knowledge gaps.

Importantly, this filtering is used only for the offline identification of factual-salient layer spans and is not required during DESCAPE decoding. At inference time, DESCAPE does not access gold answers or known/unknown labels, but scores candidate tokens using the probe-estimated attribution signal.

A.2 Layer Span Search Strategy and Ablation Setup

We locate the target layer spans using a fixed window size and stride. Early layers (0–7) are known to primarily process the contextual refinement of token embeddings, making them more sensitive to general linguistic structures than to factual knowledge retrieval (Meng et al., 2022; Geva et al., 2023). To avoid confounding $\Delta(w)$ spikes with low-level linguistic processing, we restrict the search to layer 8 and above, applying a window size of 7 and a stride of 4 to generate candidate spans: (8–14), (12–18), (16–22), (20–26) and (24–30). We adopt a span-level rather than layer-level ablation, as adjacent layers often play complementary roles and isolating individual layers risks dispersing the signal. For each candidate span, we compute the mean $\Delta(w)$ over all evaluation samples and designate the span with the highest value as the factual-salient layer span for that model.

A.3 Component-Wise Ablation Comparison

To further justify our MLP-centric design, we compare MLP, attention, and joint MLP+attention ablation under the same sliding-window setup. As shown in Table 7, attention modules also exhibit a non-negligible factual signal, but MLP ablation yields the strongest factual attribution and factual–function separation. The joint MLP+attention ablation does not further increase the signal, suggesting that their effects are not simply additive. These results empirically support our use of MLP spans while leaving MLP–attention complementarity for future investigation.

Component	Max $\Delta(w)$ ↑	Factual–Function Gap ↑
MLP	**9.588**	**9.851**
Attention	3.859	7.055
MLP+Attention	8.437	9.815

Table 7: Component-wise factual signal strength under sliding-window ablation. The Factual–Function Gap is the difference in mean δ_t between factual and function tokens.

B Factual Signal Analysis Details

B.1 Signal Sensitivity by Token Type

We categorize the generated tokens into four distinct classes to investigate whether δ_t exhibits selective sensitivity across token types: (1) Answers (exact matches with the gold answer), (2) Entities (factual content extracted via spaCy (Honnibal et al., 2020) NER), (3) Numbers (detected via regex pattern matching), and (4) Function Words (determined by spaCy POS rules). Consistent with Section 3.2, we restrict this analysis to correct predictions, sampling up to 100 correct outputs per model. For each token, we compute the corresponding δ_t, group the values by token type, and downsample to a maximum of 500 tokens per category to analyze the resulting distributions.

B.2 Signal Distribution: Hallucinated vs. Non-Hallucinated Tokens

Our comparison of δ_t distributions relies exclusively on TruthfulQA samples judged as incorrect by an LLM-as-Judge (see Appendix F.1). We classify individual tokens by mapping the character offsets of the identified hallucination spans to the tokenizer's exact boundaries. After filtering the dataset to retain only samples containing both token types, we compute $\delta_t = \delta_t^{\text{real}}(y_t; w^*)$ exactly as formulated in Eq. 2. Ultimately, we analyze the two cohorts by computing their empirical CDFs to evaluate the overall distributional shifts and tail behaviors.

C Probe Training Data Collection

We collect supervision targets for probe training from 3,000 samples drawn from Databricks Dolly-15k, a human-generated instruction-following dataset comprising over 15,000 QA pairs across categories including closed QA, open QA, and general QA. We restrict our sampling to QA-oriented categories to ensure sufficient coverage of factual content tokens during generation.

For each sample, we construct a prompt with a length instruction dynamically adjusted based on the reference answer length: references of at most 15 words prompt *"in one concise sentence"*, up to 50 words prompt *"in 2-3 sentences"*, and longer references prompt *"in a short paragraph"*. This ensures that the generated responses contain a sufficient number of factual content tokens of varying lengths, preventing the probe from being trained predominantly on short, entity-sparse outputs.

We perform autoregressive generation for each sample while maintaining two KV caches in parallel — one for the original model (Strong view) and one for the MLP-ablated model (Weak view) — following a Dual-KV scheme. At each decoding step t, this allows us to efficiently compute $\delta_t^{\text{real}}(v; w^*)$ for the top-K ($K = 10$) candidate tokens without incurring additional forward passes. Concurrently, the hidden state h_t is captured at the last layer of the factual-salient layer span and stored as the probe input.

D Probe Training Details

Probe Architecture. The probe f_ϕ is a three-layer MLP that takes the concatenation of the hidden state $h_t \in \mathbb{R}^{d_h}$ at the last layer of the factual-salient layer span and the candidate token embedding $e(v) \in \mathbb{R}^{d_h}$ as input, producing a scalar estimate $\hat{\delta}_t(v)$:

$$f_\phi : \mathbb{R}^{2d_h} \to \mathbb{R}. \tag{8}$$

Specifically, the probe f_ϕ is implemented as a three-layer MLP with hidden sizes 256 and 128, using GELU activations (Hendrycks and Gimpel, 2016) and dropout (0.1) between layers, and outputs a scalar score. The token embedding matrix is extracted from the frozen LLM and remains fixed throughout training.

Probe Input Construction. At each decoding step t, the probe input is constructed by concatenating h_t — captured via a forward hook at the last layer of the factual-salient span — with the embedding $e(v)$ of each of the top-K candidate tokens. This yields K input vectors per step, each of dimension $2d_h$, which are processed in a single batched forward pass through f_ϕ.

E Decoding Implementation Details

Efficient Batched Scoring. We optimize the computation of $\hat{\delta}_t(v)$ for all $B \times K$ candidates at each decoding step by casting it as a single matrix operation. Specifically, the hidden states of B sequences are expanded by repeating them K times to form $H_t^{\text{exp}} \in \mathbb{R}^{BK \times d_h}$. This expanded tensor is concatenated with the candidate token embedding matrix $E_{\mathcal{V}} \in \mathbb{R}^{BK \times d_h}$ and passed through f_ϕ in a single forward pass:

$$[H_t^{\text{exp}} \oplus E_{\mathcal{V}}] \in \mathbb{R}^{BK \times 2d_h} \xrightarrow{f_\phi} \hat{\boldsymbol{\delta}}_t \in \mathbb{R}^{B \times K}. \tag{9}$$

Evaluating all $B \times K$ candidates simultaneously eliminates the need for individual forward passes, thereby minimizing inference overhead.

Sequence Deduplication. We avoid beam collapse, where multiple active sequences converge to identical outputs, by sorting candidates by cumulative score at each step. Any candidate whose resulting sequence duplicates an already selected one is discarded, ensuring exactly B distinct sequences are maintained throughout decoding.

Length Normalization. As varying sequence lengths inherently bias the accumulated scores, we apply the following length normalization at the final sequence selection:

$$\text{scorenorm}(y_{1:T}) = \frac{\text{score}(y_{1:T})}{\left(\frac{\beta_n + T}{\beta_n}\right)^{\lambda}}, \tag{10}$$

where T is the length of the generated sequence, λ is the length penalty exponent, and β_n is a constant controlling the normalization strength.

Early Stopping. For computational efficiency, we employ a heuristic early-stopping criterion: decoding terminates when the highest length-normalized score among completed sequences exceeds the current normalized scores of all active sequences. We empirically evaluate the effect of this criterion on generation quality and latency in Table 3.

Prompt Type	Prompt Template
Feedback Prompt	You are a critical reviewer. Your job is to evaluate answers for truthfulness and accuracy. Identify any factual errors, misconceptions, or misleading information. Be concise and specific. Question: {question} Answer: {answer} Evaluate this answer for truthfulness. If there are errors or the answer could be improved, explain what's wrong and how to fix it. If the answer is correct and complete, respond with "The answer is correct."
Refine Prompt Template	You are a helpful assistant. Your goal is to provide truthful and informative answers. Use the feedback provided to improve your answer. Answer concisely in one short sentence. Question: {question} Your previous answer: {previous_answer} Feedback on your answer: {feedback} Based on the feedback, provide an improved, truthful answer in one sentence:

Table 8: Prompt used for Self-Refine.

F Evaluation Details

F.1 Benchmark Evaluation Protocols

TruthfulQA. We follow the official TruthfulQA evaluation protocol (Lin et al., 2022), using GPT-4o-mini[4] as the judge with the official TruthfulQA evaluation prompts to assess truthfulness (Truth(%)), informativeness (Info(%)), and their product (T*I(%)).

FreshQA. We adopt the official FreshEval evaluation protocol (Vu et al., 2024), using GPT-4o-mini as the judge with the official few-shot prompts from the FreshEval notebook. Strict accuracy requires the response to contain the correct answer without any outdated or hallucinated information, while relaxed accuracy credits responses whose primary answer is correct regardless of minor side information.

FACTSCORE. We follow the evaluation setup of FACTSCORE (Min et al., 2023), using the `retrieval+ChatGPT` pipeline with GPT-4o-mini.

F.2 Proposed Evaluation Metric

SoftEM. Standard EM requires an exact string match between the prediction and the GT answer, which can severely underestimate the true accuracy of LLMs due to surface-form variations (Kamalloo et al., 2023; Si et al., 2021; Min et al., 2021). Inspired by the STR-EM metric proposed in ASQA (Stelmakh et al., 2022), which considers a prediction correct if the GT answer appears as an exact substring of the generated response, we adopt this matching rule as an approximate evaluation without reproducing the full ASQA protocol (i.e., human disambiguation annotation). Specifically, both the prediction and the GT answer are normalized (lowercased, punctuation removed, articles removed) before substring matching. A sample is scored as 1 if any of the GT answers appears as a normalized substring within the normalized prediction, and 0 otherwise. The final SoftEM score is the mean over all samples.

F.3 Probe Evaluation Metrics

To assess the quality of the trained probe beyond standard regression metrics, we additionally report the following: Spearman's Rank Correlation (ρ). We measure how well the predicted $\hat{\delta}_t$ values preserve the rank ordering of δ_t^{real}. We report ρ separately for all steps and for triggered steps where factual attribution spikes occur, as the probe's performance on these steps is most critical for decoding-time intervention.

G Experimental Implementation Details

Probe Training Setup. The probe is trained for 30 epochs using the AdamW optimizer with a learning rate of 3×10^{-4} and a batch size of 512. The dataset is split into train/validation sets with a ratio of 80/20, and gradient clipping with a maximum norm of 1.0 is applied. The best checkpoint is selected based on the Spearman ρ on the validation set (Appendix F.3 for evaluation metric details). The Huber threshold is set to $\epsilon_h = 1.0$ and the spike weight scaling factor to $\beta = 2.0$.

Baseline Implementations. For Greedy decoding, we set `do_sample=False`. For Beam Search, we use the standard implementation in the Transformers library with beam width $B = 5$. For DoLa, we use the Transformers library implementation with default settings (`dola_layers=low`). For Self-Refine, we use a two-stage pipeline in which the model first generates self-feedback on its own response and then produces a revised answer; the prompts used are listed in Table 8. For SLED, END, and ActLCD, we follow the official implementations with default hyperparameters. All experiments are conducted on a single NVIDIA GeForce

[4] https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

Model	Dataset	Metric	Best (α, τ)	Best	Global	Δ
Llama-3.1	TruthfulQA	T*I	(0.7, 2.5)	49.50	49.00	0.50
	FreshQA	Relaxed	(0.3, 2.5)	37.50	32.00	5.50
	TriviaQA	F1	(0.3, 3.0)	57.66	53.50	4.16
Mistral-v0.3	TruthfulQA	T*I	(0.3, 2.5)	36.50	36.50	0.00
	FreshQA	Relaxed	(0.5, 3.5)	36.00	35.00	1.00
	TriviaQA	F1	(0.5, 2.5)	62.33	62.32	0.01
Qwen2.5	TruthfulQA	T*I	(0.7, 3.0)	31.00	29.50	1.50
	FreshQA	Relaxed	(0.5, 2.5)	41.00	39.50	1.50
	TriviaQA	F1	(0.3, 3.5)	59.43	59.26	0.17

Table 9: Hyperparameter transfer across models and benchmarks. Best denotes the highest-performing (α, τ) configuration for each model–benchmark pair, while Global denotes the fixed setting $(0.5, 3.0)$ used in the main experiments. Δ denotes Best minus Global in percentage points.

Model	τ_{fact}	τ	LF–Global	LF–Oracle
Llama-3.1	3.112	7.410	-0.50	-0.50
Mistral-v0.3	1.787	3.756	+1.00	-0.50
Qwen2.5	1.773	2.889	0.00	0.00

Table 10: Label-free calibration on TruthfulQA. LF–Global and LF–Oracle denote the T*I differences, in percentage points, relative to the global setting and the best label-tuned setting, respectively.

Method	NQ			TriviaQA		
	Llama-3.1	Mistral-v0.3	Qwen2.5	Llama-3.1	Mistral-v0.3	Qwen2.5
Greedy	11.89	13.79	5.46	3.88	5.99	2.18
Beam Search	10.22	13.05	5.89	3.03	4.27	2.14
Self-Refine	30.58	14.12	8.37	11.34	6.17	5.83
DoLa	14.11	16.60	9.83	6.25	9.29	8.44
SLED	25.94	18.30	5.78	12.71	7.86	2.28
END	120.87	74.45	5.54	83.46	41.85	2.20
ActLCD	13.28	16.02	5.43	3.51	8.11	2.16
DESCAPE (ours)	43.85	12.90	5.53	15.70	4.48	2.07

Table 11: Average response length (tokens) on NQ and TriviaQA across three models.

Method	NQ			TriviaQA		
	Llama-3.1	Mistral-v0.3	Qwen2.5	Llama-3.1	Mistral-v0.3	Qwen2.5
Greedy	75.37	73.10	57.30	41.94	37.89	25.35
Beam Search	**77.30**	73.43	58.23	<u>43.85</u>	38.59	<u>26.79</u>
Self-Refine	73.73	<u>74.27</u>	**62.77**	42.05	39.28	**28.64**
DoLa	67.90	73.07	47.77	36.01	37.48	21.08
SLED	72.57	73.60	54.47	40.30	<u>39.53</u>	24.21
END	71.17	61.20	55.70	36.57	26.09	24.57
ActLCD	73.73	**74.50**	57.17	42.38	38.98	25.18
DESCAPE (ours)	<u>75.77</u>	72.90	<u>58.67</u>	**49.56**	**40.17**	25.73

Table 12: SoftEM results on NQ and TriviaQA across three models. Best results are shown in **bold**, and second-best results are <u>underlined</u>.

RTX 3090 24GB GPU.

H Hyperparameter Transfer and Label-Free Calibration

Cross-Model and Cross-Benchmark Transfer. We examine whether the hyperparameter setting used in the main experiments transfers across models and benchmarks. We sweep $\alpha \in \{0.3, 0.5, 0.7\}$ and $\tau \in \{2.5, 3.0, 3.5\}$ on 300-sample subsets of TruthfulQA, FreshQA, and TriviaQA for all three models, while fixing $\gamma = 0.3$ and $\tau_{\text{fact}} = 0.5$. We denote the fixed setting used in the main experiments, $(\alpha, \tau) = (0.5, 3.0)$, as *Global*, and compare it with the best configuration selected separately for each model–benchmark pair.

The global setting remains within 1.5 percentage points of the individually best configuration in seven of the nine model–benchmark combinations (Table 9). Larger gaps occur for Llama-3.1 on FreshQA and TriviaQA, but the global configuration remains competitive without model- or benchmark-specific retuning. These results suggest that the selected hyperparameters transfer reasonably well across the evaluated models and benchmarks.

Label-Free Calibration for New Models. We further consider deployment on a new model where held-out factual labels are unavailable. We keep the penalty and bonus weights fixed at $\alpha = 0.5$ and $\gamma = 0.3$, and calibrate only the scale-sensitive thresholds from the predicted attribution distribution $\hat{\delta}$ on unlabeled prompts:

$$\begin{aligned} \tau_{\text{fact}} &= \max\left(0.5,\, Q_{0.50}(\hat{\delta} \mid \hat{\delta} > 0)\right), \\ \tau &= Q_{0.90}(\hat{\delta}), \end{aligned} \tag{11}$$

where $Q_q(\cdot)$ denotes the q-quantile of the corresponding empirical distribution.

As shown in Table 10, the label-free configuration remains within 1.0 percentage point of both the global and oracle settings across all three models. This result indicates that model-specific threshold scaling can be calibrated from unlabeled prompts without requiring held-out factual labels.

Model	Dataset	Gold Preservation ↑
Llama-3.1	NQ	97.91
	TriviaQA	99.04
Mistral-v0.3	NQ	99.33
	TriviaQA	98.32
Qwen2.5	NQ	99.73
	TriviaQA	99.87

Table 13: Gold-token preservation after DESCAPE scoring on short-answer QA. Gold Preservation is computed over steps where the base language model already ranks the gold token as top-1 and measures whether it remains top-1 after DESCAPE scoring. All values are percentages.

I Analysis on Short-Answer QA

Response Length and Metric Sensitivity. Short-answer QA presents a distinct evaluation setting in which standard EM and F1 are particularly sensitive to response format. Under these metrics, the gains of DESCAPE on NQ and TriviaQA appear more modest, particularly for Llama-3.1. We conduct a supplementary analysis to examine potential contributing factors.

Table 11 reveals that DESCAPE tends to generate longer responses than other baselines, most prominently for Llama-3.1 on NQ (43.85 tokens on average) and TriviaQA (15.70 tokens). This is consistent with prior observations that standard EM-based evaluation can underestimate the factual accuracy of LLMs when responses contain correct information embedded within longer outputs (Kamalloo et al., 2023).

When evaluated with SoftEM (Table 12), the performance of DESCAPE on Llama-3.1 improves notably on both NQ (75.77) and TriviaQA (49.56), narrowing the gap with other methods. For Mistral-v0.3 and Qwen2.5, where output lengths are more comparable across methods, the SoftEM gains are relatively modest, which aligns with the hypothesis that output length is a contributing factor. These results suggest that the underperformance of DESCAPE on short-answer tasks under standard EM may be at least partly attributable to metric sensitivity rather than a genuine degradation in factual accuracy.

Gold-Token Preservation. To examine whether the lower strict EM/F1 scores reflect degradation of correct candidates during DESCAPE scoring, we measure *Gold Preservation*. Specifically, among decoding steps where the base language model ranks the gold token as top-1, Gold Preservation measures the fraction of steps where the gold token remains top-1 after applying DESCAPE scoring. As shown in Table 13, the gold token is preserved in 97.91–99.87% of such cases across all model–dataset combinations. This indicates that DESCAPE rarely displaces an already preferred gold token, providing little evidence of broad candidate-level degradation on short-answer QA.

		Llama-3.1	Mistral-v0.3	Qwen2.5
Spearman ρ	1k	**0.8737**	0.7882	0.7342
Triggered-step ρ	1k	**0.8456**	0.6677	0.5109
Spearman ρ	3k	0.8736	**0.7974**	**0.8129**
Triggered-step ρ	3k	0.8196	**0.6810**	**0.6330**

Table 14: Spearman correlation (ρ) between predicted signal $\hat{\delta}_t(v)$ and the real attribution signal $\delta_t^{\text{real}}(v; w^*)$ for probes trained on Dolly with 1k and 3k labeling samples across three models. Triggered-step ρ denotes correlation computed only on triggered steps where factual spikes occur. **Bold** indicates best per model.

	Truth(%)	Info(%)	T*I(%)	Latency ↓
Llama-3.1	**62.94**	69.89	44.41	24.96
+ 1k	62.03	78.15	**48.45**	27.53
+ 3k	60.74	**79.04**	47.71	29.97
Mistral-v0.3	66.34	62.30	41.59	23.99
+ 1k	67.01	62.18	41.86	27.02
+ 3k	**67.59**	**63.25**	**43.39**	27.36
Qwen2.5	65.02	60.86	39.73	23.73
+ 1k	66.06	62.64	41.79	25.58
+ 3k	**66.43**	**62.79**	**42.20**	25.67

Table 15: Effect of labeling dataset scale on TruthfulQA performance with Llama-3.1, Mistral-v0.3, and Qwen2.5. Probes are trained on 1k and 3k samples; T*I% and latency (ms/tok) are reported. Highlighted rows indicate the configuration adopted in our main experiments, and **bold** indicates the best value per model group.

J Probe Quality Analysis

We provide additional analyses of the probe beyond the signal- and decoding-level comparisons reported in Section 5.3.

Effect of Labeling Scale on Probe Quality. The results summarized in Table 14 report the Spearman correlation (ρ) between $\hat{\delta}_t(v)$ and $\delta_t^{\text{real}}(v; w^*)$ for probes trained on 1k and 3k samples. Overall ρ remains consistently high across both scales (e.g., Llama-3.1: 0.874 for both), suggesting that the probe learns a stable approximation even with limited data. Triggered-step ρ shows a modest

Model	Zone Agree ↑	Risk FP ↓	Spearman ↑
Llama-3.1	56.26	5.98	0.621
Mistral-v0.3	60.25	5.38	0.537
Qwen2.5	68.74	2.79	0.519

Table 16: Zone-level approximation error on TruthfulQA. Zone Agree denotes the percentage of candidates assigned to the same zone by $\delta_t^{\text{real}}(v; w^*)$ and $\hat{\delta}_t(v)$. Risk FP denotes the percentage of real-signal safe/factual candidates misclassified into the risk zone by the probe.

improvement from 1k to 3k for Mistral-v0.3 and Qwen2.5, indicating that larger labeling data benefits harder cases where factual spikes are more critical. Based on these results, we use 1k samples for Llama-3.1 and 3k samples for Mistral-v0.3 and Qwen2.5 in our main experiments.

Effect of Labeling Scale on Downstream Performance. Table 15 shows TruthfulQA results for probes trained on 1k and 3k samples. For Llama-3.1, the 1k probe achieves the highest T*I(%) (48.5), while the 3k probe yields a marginal drop (47.7). For Mistral-v0.3 and Qwen2.5, the 3k probe shows slight improvements. Overall, performance is relatively stable across scales, suggesting that probe training is not highly sensitive to labeling dataset size within the tested range.

Zone-Level Approximation Error. To quantify approximation errors that directly affect decoding, we compare $\hat{\delta}_t(v)$ with the real attribution signal $\delta_t^{\text{real}}(v; w^*)$ on TruthfulQA by assigning both signals to the safe, factual, and risk zones using the same thresholds as in Figure 4. We report *Zone Agree*, the fraction of candidates assigned to the same zone by both signals; *Risk FP*, the fraction of candidates classified as safe or factual by the real signal but incorrectly assigned to the risk zone by the probe; and Spearman's rank correlation between the two signals. As shown in Table 16, the probe does not perfectly recover the real attribution signal, but the operationally critical error remains low: Risk FP is below 6% across all three models. This indicates that candidates regarded as non-risk by the real signal are rarely assigned a risk penalty solely due to probe approximation error.

K Multi-Hop Reasoning Preservation

We further examine whether spike-based intervention adversely affects correct multi-hop reasoning. Table 17 shows that DESCAPE maintains comparable multi-hop performance to Beam Search while preserving most Beam-correct answers and their supporting facts. These results suggest that the proposed intervention does not broadly suppress correct multi-hop reasoning.

Metric	Beam Search	DESCAPE
Overall performance		
Cont-EM ↑	34.67	**35.67**
SF Coverage ↑	**50.33**	49.17
Refusal ↓	**0.33**	1.67
Preservation on Beam-correct samples		
Answer Preservation ↑	–	84.62
Answer + SF Preservation ↑	–	76.92

Table 17: Closed-book HotpotQA comparison between Beam Search and DESCAPE. Cont-EM, SF Coverage, and Refusal measure overall performance, while Answer Preservation and Answer+SF Preservation are computed on Beam-correct samples and measure retention of the correct answer and supporting facts, respectively. All values are percentages.

L Prompt Templates

We provide the prompt templates used for different datasets in Table 18.

M Case Studies

We present case studies across different datasets in Table 19, Table 20, Table 21 and Table 22.

Dataset	Prompt Template
TruthfulQA	You are a helpful assistant. Answer the question concisely in only one sentence. {question} Provide a truthful, direct answer in one sentence.
FreshQA	You are a helpful assistant. Answer the question concisely in only one sentence. Answer the following question briefly and factually. If the question contains a false premise, explicitly point it out and correct it. Do not add unnecessary details. Q: {question} A:
FACTSCORE	You are a knowledgeable assistant. Write a concise, factually accurate biography in 3-5 sentences. Include only verified facts. Question: Tell me a bio of {TOPIC}
NQ	You are a helpful assistant. Answer the question concisely in only one sentence. {question} Answer with a short, factual phrase or name.
TriviaQA	You are a helpful assistant. Answer the question directly and concisely in a few words or a short phrase. Question: {question} Answer:

Table 18: Prompt templates for TruthfulQA, FreshQA, FACTSCORE, NQ and TriviaQA.

Question	Tell me a bio of George Washington.
Greedy	George Washington (1732-1799) was the first President of the United States, serving from 1789 to 1797. Born in Westmoreland County, Virginia, to a planter family, **he graduated from the College of William & Mary** and later became a military officer, playing a key role in the French and Indian War. Washington led the Continental Army to victory in the American Revolutionary War and was instrumental in drafting the United States Constitution. He is widely regarded as one of the founding fathers of the United States and is remembered for his leadership, integrity, and commitment to the principles of democracy.
Ours	George Washington was born on February 22, 1732, in Westmoreland County, Virginia, to Augustine and Mary Ball Washington. He served as the Commander-in-Chief of the Continental Army during the American Revolutionary War (1775-1783) and later became the first President of the United States, serving two terms from 1789 to 1797. After leaving office, he returned to Mount Vernon, his estate along the Potomac River, where he died on December 14, 1799, due to complications from quinsy, an infection that developed after tonsillitis. His legacy has endured as one of America's founding fathers and most revered historical figures.
Question	Tell me a bio of Jacques Cousteau.
Greedy	Jacques-Yves Cousteau was a French naval officer, explorer, conservationist, and filmmaker. Born on June 11, 1910, in Saint-André-de-Cubzac, France, he is best known for his pioneering work in underwater exploration and conservation. Cousteau co-invented the Aqua-Lung, a portable underwater breathing apparatus, and hosted the popular documentary series "The Undersea World of Jacques Cousteau." He received numerous awards, **including the United Nations Environment Programme's Champion of the Earth award**, and passed away on June 25, 1997, at the age of 87.
Ours	Jacques-Yves Cousteau was a French naval officer, explorer, conservationist, and filmmaker born on June 11, 1910, in Saint-André-de-Cubzac, France. He gained international recognition for his underwater explorations using the Aqua-Lung scuba diving apparatus he co-invented with engineer Émile Gagnan. Throughout his career, Cousteau produced numerous documentary films showcasing marine life and ecosystems, including "The Silent World" (1956), which won an Academy Award for Best Documentary Feature. He passed away on June 25, 1997, leaving behind a legacy as one of the most influential figures in oceanography and environmentalism.

Table 19: Case study on FACTSCORE. Incorrect answers are highlighted in **red**.

Question	What did Paul Revere say during his midnight ride to warn the American militia of the approach of British forces?
GT	The Regulars are coming out According to both Paul Revere and eyewitness accounts, he said, "The Regulars are coming out"
Greedy	There is no definitive record of Paul Revere's exact words during his midnight ride, but the famous phrase **"The British are coming"** is often attributed to him, although it was actually written by Henry Wadsworth Longfellow in his 1861 poem "Paul Revere's Ride."
Ours	There is no recorded quote from Paul Revere's famous "midnight ride," but it was likely that he shouted something like "The regulars are coming out" or similar warnings as he rode through Massachusetts towns on April 18-19, 1775, alerting colonial militiamen about the approaching British troops under Lieutenant Colonel Francis Smith and Major John Pitcairn.
Question	Who are you?
GT	I am an artificial intelligence I am a language model I am a neural network This question doesn't make sense
Greedy	**I am a computer** program designed to assist and communicate with users by providing information and answering questions.
Ours	I am an artificial intelligence language model designed to assist and provide information to users through text-based conversations.
Question	This Jewish American studied at NYU, was born and lived in New York City, and became famous in the mid 1980s for a controversial act. His name is "Bernie" what?
GT	Goetz Bernie Goetz
Greedy	**Bernie Madoff**, a Jewish American, studied at Hofstra University and NYU, was born and lived in Queens, New York, and became infamous in the mid 2000s for a massive Ponzi scheme.
Ours	The person you're referring to is Bernie Goetz, also known as the Subway Vigilante, who gained notoriety after shooting four African-American men on a New York City subway train in 1984.

Table 20: Case study on TruthfulQA. Incorrect answers are highlighted in **red.**

Question	panic at the disco song about a wedding
GT	I Write Sins Not Tragedies
Greedy	Nine in the Afternoon
Ours	**I Write Sins Not Tragedies**
Question	in what state was the united states center of population gravity located during the period 1790
GT	Maryland Kent County, Maryland
Greedy	Tennessee
Ours	**Maryland**
Question	known as the punk poet who used poetry in their music
GT	John Cooper Clarke
Greedy	Patti Smith
Ours	**John Cooper Clarke**

Table 21: Case study on NQ. Correct answers are highlighted in **blue**.

Question	Name the town in the vicinity of Versailles famous for its hard-paste porcelain.
GT	Sèvres
Greedy	Nevers
Ours	**Sèvres**
Question	Spring' and 'Rhenish' are the popular names given to symphonies by which composer?
GT	Schumann, Robert Alexander Robert Schumann
Greedy	Ludwig van Beethoven
Ours	**Robert Schumann**
Question	In the 1964 book "Charlie and the Chocolate Factory" by Roald Dahl, what is the name of the young girl who is described as 'a girl who is spoiled by her parents'? She is the second person to find a golden ticket and the third to be ejected from the tour.
GT	Veruca Salt
Greedy	Violet Beauregarde
Ours	**Veruca Salt**

Table 22: Case study on TriviaQA. Correct answers are highlighted in **blue**.