TRE: Training-Free Hallucination Detection for Diffusion Language Models

Pengcheng Weng †, Yanyu Qian †, Yue Tan †, Yixin Liu *

Posted Date: 30 June 2026

doi: 10.20944/preprints202606.2166.v1

Keywords: language models; hallucination detection; training-free

 doi:10.20944/preprints202606.2166.v1

Article

TRE: Training-Free Hallucination Detection for Diffusion Language Models

Pengcheng Weng[1,†], Yanyu Qian[2,†], Yue Tan[3,†] and Yixin Liu[3,*]

[1] University of Bern,
[2] Nanyang Technological University,
[3] Griffith University
* Correspondence: yixin.liu@griffith.edu.au
† Equal Contribution.

Abstract

Diffusion large language models (D-LLMs) have recently gained increasing attention, yet their reliability is significantly hindered by the hallucination problem. Existing hallucination detection approaches for D-LLMs mainly follow a training-based paradigm, relying on data-driven training to optimize the detector. Such reliance not only limits their generalizability across domains models but also incurs additional training cost and deployment overhead. To address these limitations, we propose TRE, a training-free hallucination detection metric for D-LLMs. TRE is a parameter-free and single-run metric that estimates hallucination risk directly from the entropy signals of a single generation, without requiring any detector training or repeated sampling. TRE extracts entropy signals within the D-LLM decoding process along both the spatial and temporal dimensions. From a token-level spatial perspective, we focus on revealing tokens as the most informative carriers of uncertainty, capturing where uncertainty is actively committed. From a diffusion step-level temporal perspective, we empirically identify the dominance of late-step entropy and hence aggregate these signals with a simple linear weighting scheme to obtain TRE. Extensive experiments on multiple D-LLMs and QA datasets demonstrate that TRE achieves competitive performance, while enjoying strong generalizability, efficiency, and robustness.

Keywords: language models; hallucination detection; training-free

1. Introduction

Diffusion large language models (D-LLMs) have recently emerged as an alternative paradigm to mainstream autoregressive large language models (LLMs) [1,2]. Instead of generating left-to-right one token at a time, a D-LLM iteratively denoises a token canvas and progressively reveals a subset of unresolved positions at each step (as shown in Figure 1a) [3,4]. Owing to this difference, D-LLMs exhibit several advantageous properties compared to their autoregressive counterparts, such as improved generation efficiency and more flexible reasoning [5]. Despite their advantages, similar to autoregressive LLMs, D-LLMs can also suffer from hallucination issues, i.e., producing fluent and self-consistent responses that are nonetheless factually unsupported [6–8]. This issue makes hallucination detection a critical and practical problem for improving the reliability of D-LLMs.

While the problem of hallucination detection has been widely studied since the rise of LLMs, most existing detectors are designed for autoregressive generation [9], where predictions are made sequentially through a single next-token frontier. Diffusion large language models (D-LLMs), however, follow a fundamentally different decoding paradigm. In D-LLMs, tokens are generated through iterative denoising, with multiple positions being updated in parallel and gradually revealed over time. As a result, detecting hallucinations in D-LLMs requires a comprehensive understanding of both the *spatial structure* of the token canvas and the *temporal dynamics* of the denoising process [10].

 doi:10.20944/preprints202606.2166.v1

Since hallucination detection methods for autoregressive LLMs do not explicitly consider the above properties, they may not be directly applicable to D-LLMs or exhibit sub-optimal performance [11].

To fill the above gap, recent studies have explored hallucination detection for D-LLMs [10–12]. These pioneering studies develop well-crafted neural network-based hallucination detectors using advanced techniques, such as action trace modeling [11], deviation learning [10], and dynamic graph modeling [12]. Although these methods demonstrate strong detection performance compared to detection approaches for autoregressive LLMs, they require data-driven training for the detectors (i.e., training-based methods), which introduces several limitations: ❶ **Limited Generalizability**. The performance of training-based methods relies heavily on dataset-specific training on the target domain. As a result, when transferred across datasets or domains, the detectors may fail to generalize effectively. ❷ **Training and Deployment Cost**. Training-based methods require annotated data and incur heavy computational overhead during training, which increases both development cost and deployment complexity. ❸ **Inference Efficiency**. Training-based methods often require additional detector networks, resulting in higher detection latency and reduced practicality for real-time deployment. Given these limitations, a natural question arises:

Can we design a training-free metric for hallucination detection in D-LLMs?

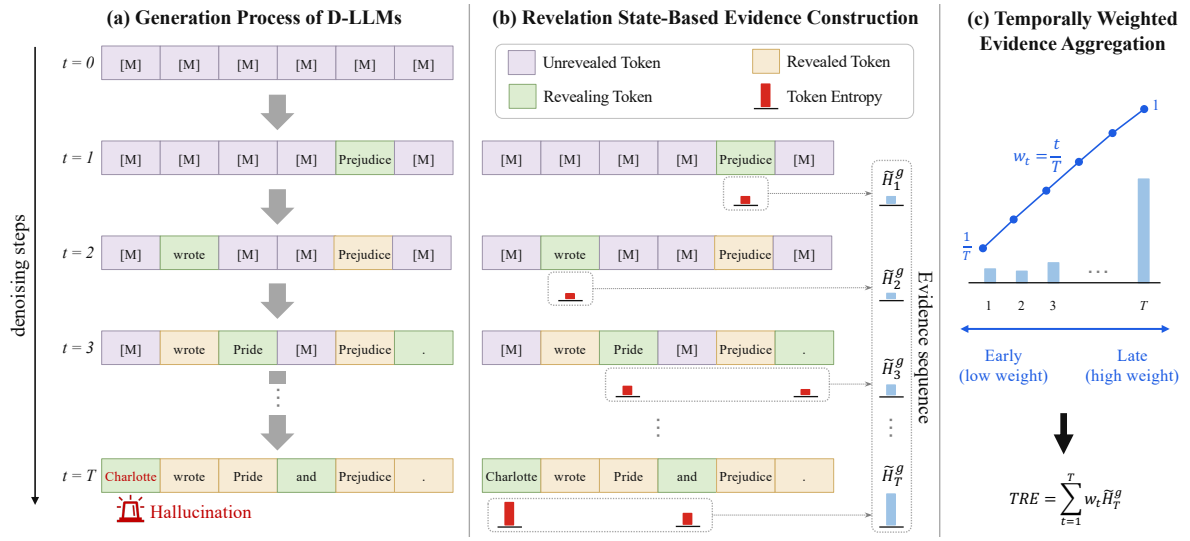


Figure 1. Overview of (a) the generation process of D-LLMs and the two key components of TRE, i.e., (b) revelation state-based evidence construction and (c) temporally weighted evidence aggregation.

As prediction uncertainty (typically quantified by token-level entropy) serves as a reliable indicator for hallucination detection in both autoregressive LLMs [13,14] and D-LLMs [11,12], we believe it provides a natural basis for designing a training-free metric. However, the iterative denoising process in D-LLMs produces entropy *at each time step for every token* [15,16], which requires us to identify the most indicative signals from a large set of token entropy. To tackle this challenge, we decompose the problem into two dimensions. From a *spatial perspective*, a long sequence of tokens is involved, each associated with its own uncertainty signal during the denoising process. In this context, a critical question arises: ***Q1*** *- which tokens can provide informative evidence during the denoising process?* From a *temporal perspective*, the evolving trend of entropy throughout the diffusion process provides an informative signal that can be leveraged for hallucination detection [10]. Once the key tokens are identified at each step, a follow-up question arises: ***Q2*** *- how should we aggregate the temporal uncertainty signals to construct a reliable indicator for hallucination detection?*

Guided by the above questions, in this paper, we propose Temporal-weighted Revealing Entropy (TRE for short), a training-free and single-run metric for hallucination detection in D-LLMs. We answer these questions in two stages, leveraging empirical observations and analytical reasoning to guide the

 doi:10.20944/preprints202606.2166.v1

design of TRE. Specifically, to answer ***Q1***, we categorize tokens into three groups according to their revelation states at each denoising step, and our empirical analysis shows that the entropy of revealing tokens corresponding to actively generated content provides the most informative signal. Motivated by this finding, we collect the entropy mass at each diffusion step to construct the fundamental entropy-based evidence for hallucination detection (Figure 1b). Then, to address ***Q2***, we analyze the temporal evolution of the evidence during the denoising process, and find that later diffusion steps contribute more than earlier ones to hallucination detection. This inspires us to design a temporal weighting scheme to aggregate the revealing-token evidence across diffusion steps to compute the final metric (Figure 1c). To sum up, our contributions are summarized as follows:

- To the best of our knowledge, we take the first step toward a training-free hallucination detection method specifically designed for D-LLMs, which mitigates the data dependency and generalizability limitations of existing training-based detectors.
- We propose TRE, a training-free and single-run metric for hallucination detection of D-LLMs, with designs grounded in empirical analysis. The proposed metric TRE enjoys *simplicity*, *generality*, and *accessibility*, without requiring data-driven training or multi-run sampling.
- Extensive experiments on multiple datasets and backbone D-LLMs demonstrate that TRE achieves strong *effectiveness*, *computational efficiency*, *robustness*, and *generalizability* across diverse scenarios, highlighting its potential for practical deployment.

2. Related Work

Hallucination Detection in Autoregressive LLMs has been widely studied as a way to assess answer reliability [6,7,17]. Among them, a branch termed ***training-free methods*** mainly relies on uncertainty or consistency signals [13,14,18,19]. Uncertainty-based methods use decoding-time quantities such as token likelihood, perplexity, predictive entropy, or length-normalized entropy to estimate answer reliability [14,18,20,21]. Differently, consistency-based methods compare multiple generations from the same input, using self-checking, lexical or semantic similarity, and semantic level uncertainty to detect unstable or contradictory responses [13,22,23]. These methods avoid training an additional detector, but they face an efficiency-evidence trade-off: single-run uncertainty scores are efficient but provide only limited evidence from one decoding path, whereas consistency-based scores provide richer evidence but require repeated generation, making detection substantially slower [24,25]. Another branch of hallucination detection, namely ***training-based methods***, aims to learn deep model-based hallucination detectors from richer signals, such as hidden states, latent truthfulness directions, representation-space consistency, attention features, or supervised labels [23,26–29]. They can improve detection performance, but often require labeled data, calibration sets, auxiliary objectives, or model-specific feature engineering [30,31]. Meanwhile, learned detectors are often dataset- or domain-specific, which limits their generalizability [32–34]. Although the above methods have shown effectiveness for AR-LLMs, they may not directly transfer to hallucination detection in D-LLMs, due to fundamental differences in their generation processes [15,16,35].

Hallucination Detection in Diffusion LLMs (D-LLMs) is a new research direction and has been rarely explored. D-LLMs generate text through iterative denoising, producing uncertainty across both denoising steps and token positions, making hallucination detection more challenging [15,16]. Existing hallucination detection methods for D-LLMs are mainly training-based, where a detector network is trained to extract informative signals from the uncertainty at denoising steps and token positions. For example, TraceDet learns informative entropy sub-traces from the action trace [11], while DynHD detects hallucination based on the deviations between the predicted and real entropy trajectories [10]. TDGNet builds temporal dynamic graphs over evolving token-level structures to model spatiotemporal dependencies of uncertainty [12]. These methods show the importance of capturing token spatial and diffusion temporal signals; however, their reliance on trained detector models limits their flexibility, running efficiency, and leads to poor generalizability [23,32,36]. In contrast, the proposed approach TRE is a training-free method for hallucination detection in D-LLMs,

 doi:10.20944/preprints202606.2166.v1

without requiring detector training or repeated sampling. The training-free property of TRE allows it to be directly applied to new application domains in a fast and efficient manner.

3. Preliminaries

Diffusion Large Language Models (D-LLMs) generate sequences through an iterative denoising process [16,37]. Unlike autoregressive LLMs that decode tokens from left to right, D-LLMs start from a highly masked sequence and progressively reveal tokens at different positions. Formally, given a prompt $\mathbf{q}$, the model maintains a discrete sequence state $\mathbf{x}^{(t)} = (x_t^{(1)}, \ldots, x_t^{(l)}) \in \mathcal{V}^l$ at denoising step $t \in \{0, \ldots, T\}$, where $\mathcal{V}$ is the vocabulary, l is the fixed sequence length, and T is the total number of denoising steps [37,38]. The process starts from $\mathbf{x}^{(0)}$ and gradually denoises it into the final response $\mathbf{r} = \mathbf{x}^{(T)}$. At each step t, the model predicts a categorical distribution $\pi_i^{(t)}$ over the vocabulary for each position i, and we quantify the corresponding uncertainty by token entropy:

$$H_t(i) = -\sum_{v \in \mathcal{V}} \pi_i^{(t)}(v) \log \pi_i^{(t)}(v). \tag{1}$$

The token entropies collected across positions and denoising steps form an uncertainty trajectory, which provides the basic signal for hallucination detection [14].

Hallucination Detection aims to identify whether a generated response contains factually incorrect or unsupported content [6]. We study hallucination detection for a question-response pair $(\mathbf{q}, \mathbf{r})$, where $\mathbf{r}$ is generated by a D-LLM. Each response is assigned a binary label $y \in \{0, 1\}$, where $y = 1$ indicates a hallucinated response and $y = 0$ denotes a factually correct one. Given a dataset $\mathcal{D} = \{(\mathbf{q}_n, \mathbf{r}_n, y_n)\}_{n=1}^N$, our goal is to compute a hallucination score $s(\mathbf{q}, \mathbf{r})$ from the uncertainty signals observed in a single denoising trajectory. A larger score indicates a higher hallucination risk. In this work, we focus on ***training-free hallucination detection*** for D-LLMs, where the score is directly constructed from entropy signals without fitting an additional detector.

4. TRE for D-LLM Hallucination Detection

In this section, we introduce Temporal-weighted Revealing Entropy (TRE), a training-free and single-run metric for hallucination detection of D-LLMs. From a *token-level spatial perspective*, we categorize tokens into three groups according to their revelation states at each denoising step, and then identify the token most indicative of hallucination as the fundamental entropy evidence for TRE (Sec. 4.1). From a *token-level temporal perspective*, we analyze the temporal evolution of the evidence during the denoising process, which guides the design of TRE that aggregates signals along the temporal dimension (Sec. 4.2). To better understand the mechanism of TRE, we model the generation process of D-LLMs as a *constraint-driven dynamical system*, from which we can interpret the key design of TRE from a statistical physics perspective (Sec. 4.3).

4.1. Revelation State-Based Evidence Construction

Unlike autoregressive decoding, diffusion decoding does not expose a single left-to-right decision frontier [39,40]. Instead, it generates multiple tokens over arbitrary positions at each denoising step, and as the diffusion process unfolds, more tokens are gradually revealed until the final sequence is fully determined. At each diffusion step t, all tokens are associated with an entropy value that reflects their uncertainty; however, not all tokens are equally informative for hallucination detection. To facilitate effective hallucination detection, the key is to select the tokens with hallucination-related uncertainty signals as evidence.

Revelation State-Based Token Categorization. To identify hallucination-related tokens, we need to distinguish tokens with different roles. While tokens can be characterized by their position, semantic meaning, or uncertainty, we find that their *revelation state* serves as a more effective indicator for hallucination detection. This is grounded in the inherent diffusion generation process of D-LLMs: at the initial step ($t = 0$), all tokens are unrevealed and represented as masked placeholders; at each

 doi:10.20944/preprints202606.2166.v1

intermediate step, a certain number of tokens are "revealing", transitioning from an unknown masked state to a deterministic token state; at the final step ($t = T$), all tokens are fully determined. Based on their revelation states, at each diffusion step, tokens in the generated sequence can be divided into three categories: unrevealed tokens, revealing tokens, and revealed tokens.

Formally, at diffusion step t, the unrevealed token set is denoted as $\mathcal{T}_t^u$, where each token is masked and remains in an uncertain state; the revealing token set is denoted as $\mathcal{T}_t^g$, where tokens are transitioning from the masked state to a determined state at the current step; and the revealed token set is denoted as $\mathcal{T}_t^r$, where tokens have been determined in previous steps. There is no overlap between the three sets, and their union $\mathcal{T}_t^u \cup \mathcal{T}_t^g \cup \mathcal{T}_t^r = \mathcal{T}$ forms the complete token set. Taking the second step $t = 2$ in Figure 1a as an example, the revealed token is "Prejudice", the revealing token is "wrote", while the remaining tokens are unrevealed.

The rationale behind such categorization is that the revelation state reflects the certainty level of each token at each step, and thus links its entropy to its role in hallucination detection. Specifically, the entropy of unrevealed tokens may reflect the model's uncertainty over a large space of possible outcomes. In contrast, the entropy of revealed tokens is typically less informative, as it only reflects residual uncertainty around already determined tokens. Furthermore, the entropy of revealing tokens directly corresponds to the uncertainty of the currently generating tokens, and thus directly reflects the confidence of the generation. Considering that the entropy of the three types of tokens plays different roles, we posit that their contributions to hallucination detection can be reasonably differentiated.

Empirical Analysis. To understand the roles of different token groups, we empirically investigate how their entropy contributes to hallucination detection. Specifically, for each token type, we first compute the average entropy for each sample, and then use Cohen's d to quantify the difference in entropy between hallucinated and non-hallucinated samples. We use late-stage entropy (i.e., from the last 30% of denoising steps) as evidence, since later steps tend to be more informative (see Sec. 4.2 for detailed analysis), and employ mean aggregation to summarize these late-stage signals. More concretely, the raw evidence in this experiment can be written as:

$$\bar{H}^c := \frac{1}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \frac{1}{|\mathcal{T}_t^c|} \sum_{i \in \mathcal{T}_t^c} H_t(i), \tag{2}$$

where $\mathcal{S}$ denotes the set of late diffusion steps, $\mathcal{T}_t^c$ denotes the token set of category c at step t, and $H_t(i)$ denotes the entropy of token i at diffusion step t, with category $c \in \{u, g, r\}$ and u, g, and r corresponding to unrevealed, revealing, and revealed tokens, respectively.

Taking $\bar{H}^u$, $\bar{H}^g$, and $\bar{H}^r$ as evidence, we compute Cohen's d to quantify the difference in these values between hallucinated and non-hallucinated samples. A larger Cohen's d indicates a stronger separability between the two groups, and thus better discriminative power for hallucination detection. We also compute the average late-stage entropy over all tokens, $\bar{H}$, as an additional baseline. According to the results shown in Figure 2 (more results are in Appendix A.1), we have the following observations. ❶ $\bar{H}^g$ consistently achieves the best discriminateness, indicating that the entropy of revealing tokens is most aligned with the uncertainty of the ongoing generation process and thus most relevant to hallucination detection. ❷ $\bar{H}^r$ shows limited discriminative power, as it mainly reflects residual uncertainty of already determined tokens. ❸ The overall entropy $\bar{H}$ performs moderately, suggesting that aggregating all tokens may dilute informative signals. ❹ $\bar{H}^u$ remains reasonably discriminative, as it captures uncertainty over yet-to-be-determined tokens. An in-depth interpretation of $\bar{H}^g$'s behavior is provided in Appendix A.1.

 doi:10.20944/preprints202606.2166.v1

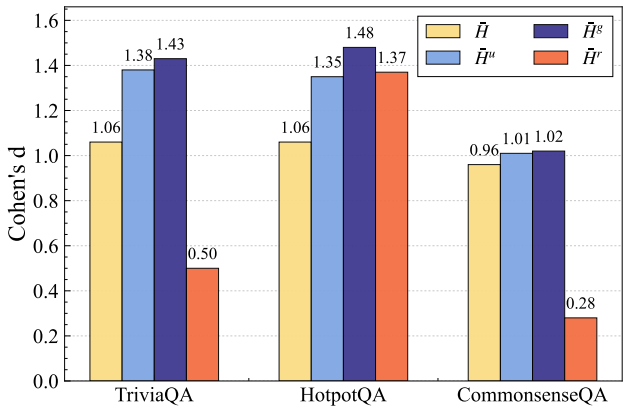


Figure 2. Evidence informativeness comparison on LLaDA-8B-Instruct with 64 diffusion steps.

Entropy Mass as Evidence Construction. As the entropy of revealing tokens consistently shows strong discriminative power for hallucination detection, we employ H_t^g as the primary evidence for TRE. Specifically, we collect the entropy of revealing tokens at all diffusion steps, forming an evidence sequence $\mathbf{e} \in \mathbb{R}^T$ for hallucination detection:

$$\mathbf{e} = \{\tilde{H}_t^g\}_{t=1}^T = \{\sum_{i \in \mathcal{T}_t^g} H_t(i)\}_{t=1}^T. \tag{3}$$

Note that we use the mass entropy $\tilde{H}_t^g$, rather than the average entropy $\bar{H}_t^g$ as evidence. This is because hallucination risk depends not only on the uncertainty of each revealing token, but also on the amount of uncertain content being revealed. The entropy mass can capture both the intensity of uncertainty and the volume of revealed tokens, which can be interpreted as the entropy flux across the evolving reveal boundary at step t. Therefore, we argue that average entropy may lose information about the volume of uncertainty, whereas entropy mass provides a more informative signal. Also, we do not include $t = 0$, since all tokens are still masked at this stage, yielding non-informative entropy. The revealing token-based evidence then provides the raw signal for computing our training-free hallucination detection metric TRE.

4.2. Temporally Weighted Evidence Aggregation

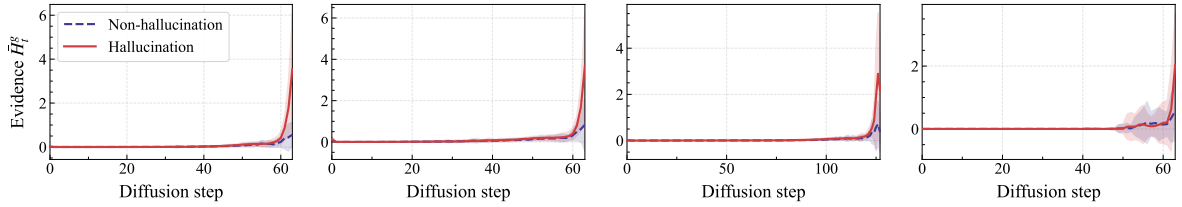
Although the above empirical results show that revealing token-based evidence can provide useful signals for hallucination detection, mean aggregation across different diffusion steps ignores the temporal variation of these signals. Indeed, in diffusion models, different denoising steps typically play different roles in the generation process. For example, in diffusion models for image generation, early steps tend to capture coarse structures, while later steps focus on refining fine-grained details [41–43]. In recent studies of training-based D-LLM hallucination detection, researchers have also observed that signals from different denoising steps exhibit varying levels of discriminative power [10,11]. Nevertheless, unlike training-based approaches, where the importance of different denoising steps can be automatically learned, in our training-free setting, the contribution of each step needs to be specified through a designed aggregation scheme. To effectively aggregate the evidence, a natural question is: *Which time steps contribute more to hallucination detection?*

Empirical Analysis. To address the above question, we analyze the evidence sequences of both hallucinated and non-hallucinated samples to uncover their temporal patterns. Taking the entropy mass as evidence, we visualize the evidence sequences of different D-LLMs (LLaDA and Dream) [16, 44], datasets (TriviaQA and HotpotQA) [45,46], and total diffusion steps (64 and 128), to derive a comprehensive and generalizable pattern.

The visualization results are shown in Figure 3 (more results are in Appendix A.2), which lead to the following findings. ❶ Across all four settings, the gap between hallucination and non-hallucination curves remains modest early in denoising but widens sharply near the end. This suggests

 doi:10.20944/preprints202606.2166.v1

a general pattern that **later diffusion steps** are more informative for hallucination detection. ❷ For both hallucinated and non-hallucinated samples, the entropy remains nearly constant with negligible differences during early denoising steps, resulting in flat trajectories. This indicates that, at early stages of generation, the model primarily focuses on capturing coarse or global structures, rather than content that may lead to hallucination. ❸ At later diffusion steps, the evidence trajectories of the two groups diverge markedly. Specifically, hallucinated samples exhibit a sharp upward trend, with values rising from around 0 to approximately 2∼3; in contrast, non-hallucinated samples show only modest increases and may even fluctuate in some cases.



(a) HQA-LLaDA(64) (b) TQA-LLaDA(64) (c) HQA-LLaDA(128) (d) HQA-Dream(64)

Figure 3. Revealing-token entropy trajectories across different D-LLMs, datasets, and diffusion steps.

Discussion: Why Later Steps Matter? To explain the above phenomenon, we provide the following interpretation and analysis. *First*, hallucinations often emerge when the model commits to specific factual details, such as entities, relations, or attributes [47]. These fine-grained decisions are more likely to be finalized in later steps, since the denoising process progressively refines tokens from uncertain placeholders to deterministic outputs. *Second*, later steps accumulate the effects of previous denoising decisions, so uncertainty at this stage reflects not only local token ambiguity but also the consistency of the partially generated sequence. Therefore, late-stage evidence better captures whether the generation trajectory is converging toward a reliable answer or drifting toward hallucination. *Third*, early steps mainly capture coarse structure, where predictions remain ambiguous and are less tied to factual correctness.

TRE Metric Design.

Based on the above empirical observations and analysis, we conclude that entropy-based evidence at later stages should be emphasized compared to earlier stages. However, to develop a general and robust training-free metric, it is non-trivial to define a fixed threshold to determine the onset of the late stage, as this point may vary across different models, datasets, and diffusion settings. To address this issue, instead of defining a threshold, we adopt a monotonic weighting schedule to reweight the evidence. Concretely, we assign a weight to each time step, where earlier steps receive smaller weights and later steps receive larger weights. Among all monotone schedules, we adopt the simplest parameter-free rule, $w_t = \frac{t}{T}$, where the weight increases linearly with the diffusion step. Applying this monotonic weighting, the final score of TRE can be defined as:

$$\mathrm{TRE} := \sum_{t=1}^{T} w_t \, \tilde{H}_t^g = \sum_{t=1}^{T} \frac{t}{T} \sum_{i \in \mathcal{T}_t^g} H_t(i). \tag{4}$$

Compared to existing hallucination detection methods, TRE offers the following merits. ❶ **Simplicity.** It requires only a single run and does not rely on external knowledge verification or additional LLM calls, making it lightweight and well-suited for real-time applications. ❷ **Generality.** As a training-free approach, it can be directly applied without training, avoiding potential biases introduced by data or model fitting. ❸ **Accessibility.** It relies solely on token-level entropy and does not require access to internal model states, making it applicable in gray-box settings where model parameters or internal states are not accessible. ❹ **D-LLM Compatibility.** It is specifically designed for D-LLMs by

 doi:10.20944/preprints202606.2166.v1

leveraging the denoising process and token revelation dynamics, making it better suited to D-LLMs compared to parameter-free methods designed for AR-LLMs.

4.3. Understanding TRE *from a Statistical Physics Perspective*

While empirical results show that TRE is effective, it is natural to ask whether there exists an intuitive perspective to interpret its underlying mechanism. Motivated by this, we draw an analogy between the revealing process of D-LLMs and that of a constrained dynamical system (CDS), and then use this perspective to interpret the design of TRE.

In statistical physics, CDSs (e.g., spin glass models) refer to systems whose evolution is restricted by state or interaction constraints over time [48,49]. The evolution of a CDS provides a natural analogy to the revealing process of D-LLMs, where variables (tokens) are gradually fixed while the remaining degrees of freedom are updated. Specifically, as shown in Figure 4, *at the initial stage*, a CDS is only weakly constrained, and uncertainty is relatively uniformly distributed across the system. Similarly, in the early stage of D-LLM decoding, no tokens are determined, and entropy is spread more evenly across token positions. *As time progresses*, additional constraints are imposed, which restricts part of the state space. The remaining variables must adapt within a reduced feasible region, leading to a redistribution of uncertainty. In D-LLMs, this corresponds to revealing a subset of tokens, after which the remaining tokens carry a larger share of the residual uncertainty, resulting in entropy concentration. *In some late steps of the dynamical process*, especially under incompatible or conflicting constraints, e.g., conflicting boundary conditions [50] or inconsistent local interactions [51], the system may enter a frustrated state where no globally consistent configuration exists. This leads to localized fluctuations and instability. In D-LLM decoding, such localized uncertainty amplification can be the main cause of increased conditional entropy at certain positions, thereby increasing the likelihood of hallucination.

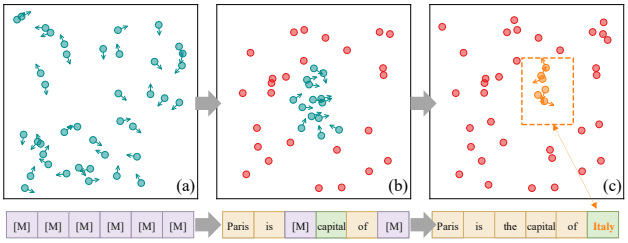


Figure 4. Analogy of CDSs and D-LLMs.

Under this analogy, we can better understand the *design intuition* of TRE. ❶ **Why revealing entropy is informative?** (Details in Appendix B.2) Revealing tokens lie at the moving boundary where uncertainty is committed into the answer, so their entropy directly measures the amount of uncertain content being fixed. ❷ **Why late-stage entropy matters?** (Details in Appendix B.6) As constraints accumulate, uncertainty is progressively concentrated on fewer remaining degrees of freedom, making late-stage entropy more informative about the final outcome. ❸ **Why hallucination samples have higher late-stage entropy?** (Details in Appendix B.4 and B.5) Incompatible commitments create frustrated boundary conditions, which destabilize the remaining subsystem and amplify uncertainty at later stages.

Notably, while this analogy provides an intuitive explanation for the decoding behavior of D-LLMs and the design of TRE, a rigorous theoretical characterization remains beyond the scope of this work. A more detailed discussion is provided in Appendix B, and we hope this can offer inspiration for future theoretical understanding of D-LLMs.

5. Experiments

5.1. Experimental Setup

Baselines. We compare TRE with training-based detectors, including CCS [30], TSV [52], and TraceDet [11], and also training-free methods, including Perplexity [53], LN-Entropy [54], Semantic

 doi:10.20944/preprints202606.2166.v1

Entropy [21], Lexical Similarity [22], and EigenScore [23]. Note that TraceDet is the state-of-the-art training-based method for D-LLMs. We would not compare with [10,12] due to the lack of open-source code.

Evaluation Protocol. Following [11], we evaluate on two D-LLM families, LLaDA-8B-Instruct [16] and Dream-7B-Instruct [44], on three QA benchmarks: TriviaQA [46], HotpotQA [45], and CommonsenseQA [55]. We use AUROC for evaluation. The training-based detectors are trained on the corresponding datasets with automatic annotations, following [11]. The training-free methods are applied directly without optimization. For Dream-7B-Instruct, we omit Perplexity and LN-Entropy due to restricted access to stable token-level logits. More experimental details are in Appendix C.

Table 1. AUROC(%) comparison of hallucination detection methods on two D-LLMs across three QA datasets. The highest score is **bolded** and the second highest is underlined.

Method	Designed for D-LLMs	TriviaQA		HotpotQA		CSQA		Avg
		128	64	128	64	128	64	
LLaDA-8B-Instruct								
Training-based Methods								
CCS	✗	57.1	54.2	57.6	55.8	50.5	58.5	55.6
TSV	✗	60.2	61.1	65.0	59.4	52.9	55.2	59.0
TraceDet	✓	73.9	74.1	66.1	63.7	77.2	**77.1**	72.0
Training-free Methods								
Perplexity	✗	50.4	47.6	49.3	51.2	65.6	65.0	54.9
LN-Entropy	✗	54.6	53.5	54.8	54.7	64.6	64.4	57.8
Semantic Entropy	✗	68.9	67.3	57.6	53.8	44.1	43.9	55.9
Lexical Similarity	✗	62.5	59.0	64.2	57.1	57.3	60.7	60.1
EigenScore	✗	69.2	66.9	64.7	59.2	58.5	60.6	63.2
TRE (ours)	✓	**82.2**	**86.5**	**85.6**	**88.1**	**79.0**	75.0	**82.7**
Dream-7B-Instruct								
Training-based Methods								
CCS	✗	56.9	50.3	51.7	58.2	54.2	53.2	54.1
TSV	✗	75.6	74.7	58.7	63.0	62.3	56.8	65.2
TraceDet	✓	78.1	**86.7**	75.1	76.0	**84.7**	**84.1**	80.8
Training-free Methods								
Semantic Entropy	✗	73.7	72.5	62.7	67.7	51.4	48.6	62.8
Lexical Similarity	✗	58.3	64.0	59.7	62.7	77.3	76.9	66.5
EigenScore	✗	66.0	69.1	62.5	67.0	76.9	77.5	69.8
TRE (ours)	✓	**83.9**	84.8	**80.6**	**81.0**	77.7	78.4	**81.1**

5.2. Results and Analysis

Main Comparison.

Table 1 reports the comparison results, from which we draw the following observations. ❶ TRE is consistently competitive across both D-LLM families and three QA datasets while requiring no training or additional generation. The gain is especially clear on LLaDA, while TRE's performance is also competitive on Dream. ❷ Detection approaches specifically designed for D-LLMs (i.e., TraceDet and TRE) significantly outperform those developed for autoregressive models, highlighting the importance of accounting for the diffusion-based generation mechanism. ❸ TRE shows comparable or even better performance compared to TraceDet, a training-based D-LLM hallucination detection method. This indicates that our training-free method can achieve strong performance without requiring dedicated detector training.

Ablation Study. To investigate the contribution of each design, we replace the evidence source and weighting scheme in TRE, and the results in Table 2 (more results are in Appendix D.1) lead to the

 doi:10.20944/preprints202606.2166.v1

following findings. ❶ Using the entropy of other token groups leads to a significant performance drop, indicating that revealing tokens provides the most informative signals. ❷ Average weighting leads to sub-optimal performance. While exponential weighting and threshold-based selection may sometimes yield slight improvements, they introduce additional hyperparameters (e.g., exponent/threshold), so we adopt a simple parameter-free linear weighting scheme. ❸ TRE generally achieves competitive performance, indicating that modeling revealing-token entropy with temporal weighting is an effective choice for training-free hallucination detection.

Table 2. Ablation study on LLaDA.

Variant	TriQA	HotQA	CSQA
All tokens $\bar{H}$	64.0	70.9	75.7
Unrevealed $\bar{H}^u$	63.9	70.8	75.9
Revealed $\bar{H}^r$	53.9	84.0	60.2
Average	81.1	85.1	79.0
Exponential	82.0	85.6	79.0
Hard last-30%	**82.4**	85.4	79.0
TRE (ours)	82.2	**85.6**	**79.0**

Efficiency Analysis. To assess the runtime efficiency of TRE, we compare the inference time on 100 QA samples from TriviaQA dataset using LLaDA. As depicted in Figure 5, TRE demonstrates the shortest runtime and significantly outperforms multi-run methods (e.g., LN-Entropy and Semantic Entropy) in terms of efficiency. Notably, training-based methods require additional time for training, which is not included in the reported runtime.

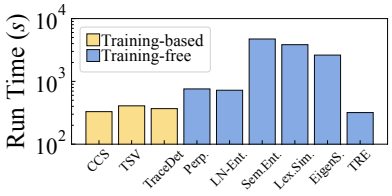


Figure 5. Time comparison.

Sensitivity Analysis. We further study the sensitivity of TRE to generation time steps, generation lengths, and remasking strategies, as shown in Figure 6. The results lead to the following findings. ❶ TRE is robust to generation time steps. Across different settings, TRE consistently outperforms TraceDet on both TriviaQA and HotpotQA, with only minor fluctuations. ❷ TRE is also insensitive to generation length. TRE maintains competitive AUROC under various length budgets, suggesting that revealing-token entropy provides reliable signals without carefully tuning the generation length. ❸ For remasking strategies, all four decoding strategies achieve consistently strong average performance, indicating that TRE remains effective across different decoding configurations.

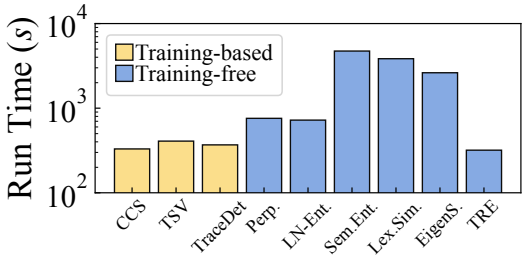


Figure 6. Sensitivity w.r.t. (a) Diffusion Steps, (b) Generation Length, and (c) Remasking Strategies.

 doi:10.20944/preprints202606.2166.v1

Case Study. We provide two hallucinated examples in Figure 7 (more can be found in Appendix D.2) to illustrate the uncertainty dynamics captured by TRE. In the figure, the generated tokens at each step (s) are listed, with the revealing entropy (H) value given. Our observations are as follows: ❶ Hallucinated entities are often revealed at late denoising steps, such as "Kykonos" and "ThesmPs", indicating that late-stage cues are usually critical. ❷ These hallucinated tokens show a sharply increased entropy when revealed, while earlier tokens usually have a near-zero or much smaller entropy. This supports our motivation that late-stage newly revealed-token entropy serves as a direct and informative signal for hallucination detection.

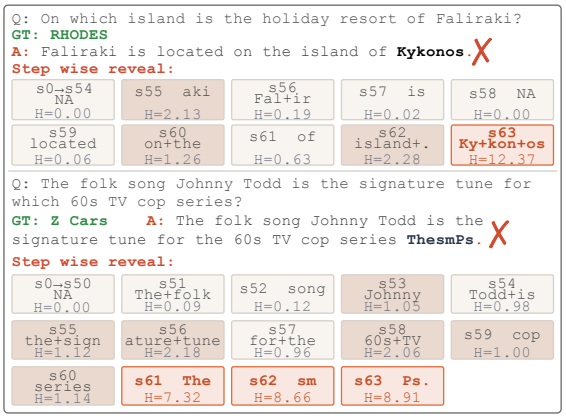


Figure 7. Case study of step-wise reveal.

6. Conclusions

In this paper, we introduce TRE, a training-free and single-run hallucination detector for diffusion language models. Unlike prior detectors that rely on additional training or repeated generation, TRE exploits the diffusion decoding trajectory by aggregating the uncertainty of revealing tokens with a simple monotone temporal weighting scheme, emphasizing late commitment events that are more indicative of hallucination. Extensive experiments on multiple QA datasets and diffusion language models demonstrate that TRE achieves strong, efficient, and robust performance with low sensitivity to decoding configurations. A potential **limitation** is that TRE relies solely on entropy-based signals and does not explicitly incorporate semantic or factual knowledge, which may limit its effectiveness in cases where uncertainty is poorly aligned with factual correctness. **Future work** may explore integrating semantic-aware signals into training-free D-LLM hallucination detection framework to incorporate richer contextual consistency measures.

Appendix A. Details of Motivating Experiments

Appendix A.1. Evidence Construction Experiments

This appendix provides additional details for the evidence construction experiment in Sec. 4.1. The main text reports the comparison on LLaDA with 64 diffusion steps. Here, we provide the corresponding results with 128 diffusion steps to verify whether the revelation-state based evidence remains informative under a longer denoising trajectory [56].

Token-Family Evidence.

At each denoising step t, we divide token positions into three revelation-state based groups: unrevealed tokens T_t^u, revealing tokens T_t^g, and revealed tokens T_t^r. For each token family $c \in \{u, g, r\}$, we compute its late-stage mean entropy as

$$\bar{H}^c := \frac{1}{|S|} \sum_{t \in S} \frac{1}{|T_t^c|} \sum_{i \in T_t^c} H_t(i), \qquad c \in \{u, g, r\}, \tag{A1}$$

 doi:10.20944/preprints202606.2166.v1

where S denotes the set of late diffusion steps and $H_t(i)$ is the token entropy at position i and step t. We also compute the all-token late-stage entropy baseline:

$$\bar{H} := \frac{1}{|S|} \sum_{t \in S} \frac{1}{L} \sum_{i=1}^{L} H_t(i). \tag{A2}$$

In our experiments, S is chosen as the last 30% of denoising steps. This is used only for evidence analysis, while the final TRE score aggregates all denoising steps through temporal weighting.

Cohen's d.

To quantify the discriminative power of each scalar evidence, we use Cohen's d, a standard effect-size measure for two-group separation. Given a scalar evidence value z, let (μ_1, s_1, n_1) and (μ_0, s_0, n_0) denote the sample mean, standard deviation, and sample count for hallucinated and non-hallucinated samples, respectively. Cohen's d is computed as

$$d = \frac{\mu_1 - \mu_0}{s_{\text{pooled}}}, \qquad s_{\text{pooled}} = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_0 - 1)s_0^2}{n_1 + n_0 - 2}}. \tag{A3}$$

A larger positive value indicates that the evidence is higher on hallucinated samples and better separated relative to within-group variability.

Analysis.

Figure A1 provides complementary evidence to the main-text comparison under a longer denoising budget. ❶ Revealed-token entropy $\bar{H}^r$ is generally weak, especially on TriviaQA and CommonsenseQA, indicating that post-commitment residual uncertainty is not a reliable evidence source. ❷ All-token entropy $\bar{H}$ is consistently less discriminative than the best token-family evidence, suggesting that aggregating all positions dilutes the informative revelation-state structure. ❸ Unrevealed-token entropy $\bar{H}^u$ remains competitive, which is expected because it captures the difficulty of the survivor set. However, revealing-token entropy $\bar{H}^g$ achieves the best Cohen's d on two out of three datasets and is nearly tied with $\bar{H}^u$ on TriviaQA. This supports our design choice that revealing tokens provide a commitment-aligned evidence source: they measure uncertainty at the moment when content enters the generated answer, rather than uncertainty that remains internal to the unresolved denoising process.

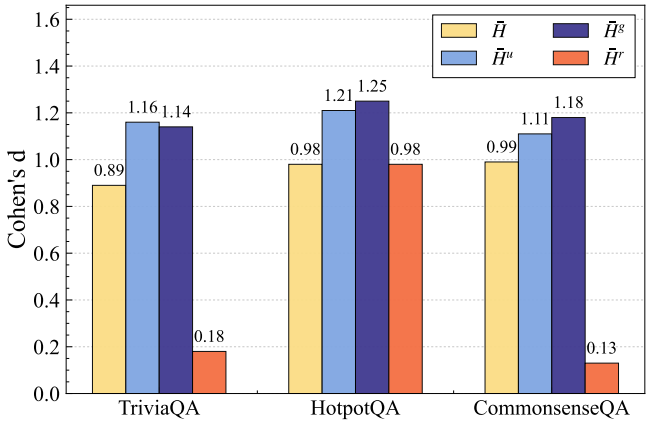


Figure A1. Evidence informativeness comparison on LLaDA with 128 diffusion steps. We report Cohen's d for all-token entropy $\bar{H}$, unrevealed-token entropy $\bar{H}^u$, revealing-token entropy $\bar{H}^g$, and revealed-token entropy $\bar{H}^r$ over the last 30% of denoising steps. Revealing-token entropy achieves the strongest separation on HotpotQA and CommonsenseQA, and remains highly competitive on TriviaQA.

 doi:10.20944/preprints202606.2166.v1

Appendix A.2. Revealing-Token Entropy Trajectory

We provide additional revealing-token entropy trajectories across model families, datasets, and diffusion-step budgets. For each denoising step t, we compute the revealing-token entropy mass

$$\widetilde{H}_t^g = \sum_{i \in T_t^g} H_t(i), \tag{A4}$$

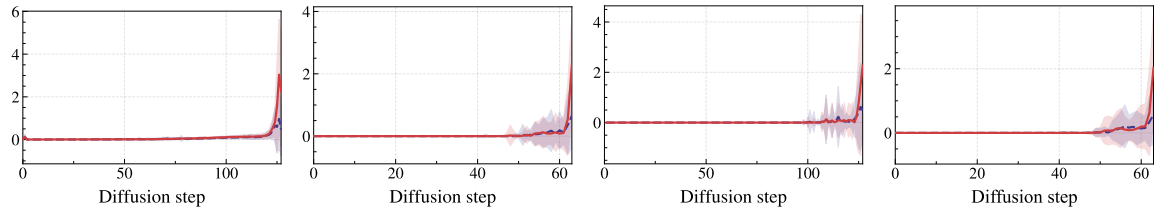
where T_t^g denotes the revealing-token set at step t. For hallucinated and non-hallucinated samples, we plot the group-level trajectories

$$\mu_t^{(1)} = \mathbb{E}[\widetilde{H}_t^g \mid y = 1], \qquad \mu_t^{(0)} = \mathbb{E}[\widetilde{H}_t^g \mid y = 0], \tag{A5}$$

where $y = 1$ and $y = 0$ denote hallucinated and non-hallucinated responses, respectively.

Analysis.

Figure A2 provides additional evidence for the temporal behavior of revealing-token entropy mass. ❶ At early denoising steps, the revealing-token entropy mass remains small or weakly separated between hallucinated and non-hallucinated samples. ❷ Near the end of denoising, the hallucination/non-hallucination gap becomes more visible, showing that the discriminative signal is concentrated in late steps. ❸ Although the exact curve shape differs across model families and datasets, the late-stage separation is consistently observed across these additional settings. These results support the temporal aggregation design of TRE, which assigns larger weights to later revealing-token entropy mass.



(a) LLaDA on TriviaQA with 128 diffusion steps. (b) Dream on TriviaQA with 64 diffusion steps. (c) Dream on TriviaQA with 128 diffusion steps. (d) Dream on HotpotQA with 128 diffusion steps.

Figure A2. Additional revealing-token entropy trajectories across model families, datasets, and diffusion-step budgets. Each plot shows the step-wise revealing-token entropy mass $\widetilde{H}_t^g$ for hallucinated and non-hallucinated samples. The separation between the two groups is generally concentrated near late denoising steps.

Appendix B. Detailed Analogy of CDS and D-LLM

This appendix gives a stylized justification for why reveal-boundary entropy mass is a useful signal for hallucination detection in diffusion language models. The purpose is not to claim that D-LLM decoding obeys a literal thermodynamic law. Rather, we use a discrete diffusion and effective free-energy view to formalize a process-level intuition: D-LLM decoding repeatedly denoises a discrete token canvas while freezing selected positions through reveal events. Under this view, confidence-based reveal can create a hard residual reservoir, while revealing-token entropy mass measures the flux of this residual uncertainty into the committed answer. Incompatible commitments can further act as unstable or frustrated boundary conditions for the remaining unresolved subsystem.

Appendix B.1. Discrete Diffusion, Reveal Operators, and Effective Energy

We first cast D-LLM decoding as a discrete reveal-denoising process. Let $\mathcal{V}$ be the token vocabulary. At denoising step t, let C_t denote the committed positions before the step, and let V_t denote the active unresolved positions before reveal. In the notation of the main text,

$$C_t = R_{t-1}, \qquad N_t = \Delta R_t, \qquad V_t = N_t \sqcup U_t, \tag{A6}$$

 doi:10.20944/preprints202606.2166.v1

where N_t is the newly revealed subset at step t, and U_t is the survivor set that remains unresolved after the reveal.

The model induces a predictive distribution over the active discrete token canvas,

$$q_t(x_{V_t} \mid x_{C_t}) \in \Delta(\mathcal{V}^{|V_t|}), \tag{A7}$$

where $\Delta(\mathcal{V}^{|V_t|})$ denotes the probability simplex over token configurations on the active positions. This distribution represents the model's current belief over unresolved token configurations conditioned on the committed scaffold.

A D-LLM decoding step can be viewed as the composition of two operations. The first is a discrete denoising or reconditioning update, represented abstractly by an operator K_t^θ. The second is a reveal operation, which selects a subset $N_t \subseteq V_t$ and freezes it to concrete token values a_t. At a process level,

$$q_{t+1} \approx K_t^\theta \circ Q^{\text{reveal}}_{N_t,a_t}(q_t). \tag{A8}$$

Here K_t^θ reshapes the predictive distribution over the remaining unresolved positions, while $Q^{\text{reveal}}_{N_t,a_t}$ removes degrees of freedom by conditioning on the revealed values.

For a fixed reveal set N_t and fixed commitment a_t, the reveal operator acts as

$$\big(Q^{\text{reveal}}_{N_t,a_t} q_t\big)(x_{U_t}) = q_t(x_{U_t} \mid x_{C_t}, x_{N_t} = a_t), \qquad U_t = V_t \setminus N_t. \tag{A9}$$

Thus, reveal is a discrete projection/conditioning operation rather than merely a continuous noise-reduction step. In D-LLMs, uncertainty is not only denoised over time, but also crosses a moving reveal boundary where some variables become committed answer content.

The reveal subset is typically chosen by a local confidence rule. In an idealized form,

$$N_t = \text{TopK}_{i \in V_t}\big[-H_q(X_i \mid X_{C_t})\big], \tag{A10}$$

and the committed token values are chosen by

$$a_{t,i} = \arg\max_{v \in \mathcal{V}} q_t(X_i = v \mid X_{C_t}), \qquad i \in N_t. \tag{A11}$$

This rule is local: it selects positions whose marginal distributions appear confident under the current scaffold. It does not guarantee that the resulting commitment a_t is globally optimal for the remaining configuration.

This discrete denoising-plus-reveal view motivates the rest of the analysis. The denoising operator K_t^θ updates the predictive distribution, while the reveal operator $Q^{\text{reveal}}_{N_t,a_t}$ freezes selected degrees of freedom and changes the boundary condition for the remaining subsystem. Hallucination risk is therefore naturally tied to uncertainty at the reveal boundary rather than to a global average over all token positions.

Given the discrete predictive distribution above, we can introduce an effective energy representation. For any finite discrete distribution and any $\beta_t > 0$, one may write

$$q_t(x_{V_t} \mid x_{C_t}) = \frac{1}{Z_t(x_{C_t})} \exp\{-\beta_t E_t(x_{V_t}; x_{C_t})\}, \tag{A12}$$

where

$$E_t(x_{V_t}; x_{C_t}) = -\beta_t^{-1} \log q_t(x_{V_t} \mid x_{C_t}) + \text{const}. \tag{A13}$$

This should be understood as an effective energy representation of the model's predictive distribution, not as a claim that D-LLM decoding is a physical thermodynamic system.

 doi:10.20944/preprints202606.2166.v1

The corresponding residual entropy is

$$S_t = H_q(X_{V_t} \mid X_{C_t}) = -\mathbb{E}_{q_t} \log q_t(X_{V_t} \mid X_{C_t}). \tag{A14}$$

The associated free energy is

$$F_t = -\beta_t^{-1} \log Z_t = \mathbb{E}_{q_t}[E_t] - \beta_t^{-1} S_t. \tag{A15}$$

The reveal operation is not a closed entropy-conserving evolution. It is an externally driven discrete quench: the protocol removes active degrees of freedom and changes the boundary condition seen by the remaining unresolved positions.

Appendix B.2. Selection-Induced Residual Concentration and Boundary Entropy Flux

Let

$$h_i(t) = H_q(X_i \mid X_{C_t}) \tag{A16}$$

be the marginal uncertainty of position $i \in V_t$ before the reveal operation. Confidence-based reveal tends to select locally easier positions. We abstract this by assuming that the newly revealed subset has no larger average entropy than the active unresolved set:

$$\bar{h}(N_t) := \frac{1}{|N_t|} \sum_{i \in N_t} h_i(t) \le \frac{1}{|V_t|} \sum_{i \in V_t} h_i(t) =: \bar{h}(V_t). \tag{A17}$$

Since $V_t = N_t \sqcup U_t$, this implies

$$\bar{h}(U_t) := \frac{1}{|U_t|} \sum_{i \in U_t} h_i(t) \ge \bar{h}(V_t). \tag{A18}$$

Indeed, letting $\mu_N = \bar{h}(N_t)$, $\mu_U = \bar{h}(U_t)$, and $\mu_V = \bar{h}(V_t)$, we have

$$|V_t|\mu_V = |N_t|\mu_N + |U_t|\mu_U. \tag{A19}$$

Since $\mu_N \le \mu_V$,

$$|U_t|\mu_U = |V_t|\mu_V - |N_t|\mu_N \ge |V_t|\mu_V - |N_t|\mu_V = |U_t|\mu_V, \tag{A20}$$

and hence $\mu_U \ge \mu_V$.

This survivor-concentration effect explains why unresolved-token entropy can remain competitive: lower-entropy positions are preferentially frozen, leaving a survivor set enriched with harder positions. However, survivor concentration alone does not identify the best hallucination signal. The set U_t still consists of unresolved variables; its uncertainty remains internal to the denoising process. Hallucination detection concerns uncertainty that is actually committed into the generated answer.

Reveal-Boundary Flux Identity.

Define the active uncertainty mass, survivor uncertainty mass, and reveal-boundary entropy flux as

$$\mathcal{E}_t^V := \sum_{i \in V_t} h_i(t), \qquad \mathcal{E}_t^U := \sum_{i \in U_t} h_i(t), \qquad \Phi_t := \sum_{i \in N_t} h_i(t). \tag{A21}$$

Because $V_t = N_t \sqcup U_t$, we have the exact identity

$$\mathcal{E}_t^V = \mathcal{E}_t^U + \Phi_t, \qquad \Phi_t = \mathcal{E}_t^V - \mathcal{E}_t^U. \tag{A22}$$

Thus, Φ_t is exactly the uncertainty mass that crosses the reveal boundary at step t.

 doi:10.20944/preprints202606.2166.v1

Why This Flux is Diagnostic.

Let ℓ_i denote the unobserved factual error loss associated with committing position i into the answer, and let

$$r_i(t) := \mathbb{E}[\ell_i \mid X_{C_t}, i \in N_t] \tag{A23}$$

be its conditional commitment risk at step t. Since the detector is training-free and entropy-only, $r_i(t)$ is not directly observable. A standard uncertainty-risk assumption is that commitment risk is monotone in predictive uncertainty:

$$r_i(t) \approx \psi(h_i(t)), \qquad \psi'(\cdot) \geq 0. \tag{A24}$$

Under the simplest local linear approximation $\psi(h) \propto h$, the total commitment risk at step t satisfies

$$\sum_{i \in N_t} r_i(t) \propto \sum_{i \in N_t} h_i(t) = \Phi_t. \tag{A25}$$

Therefore, Φ_t is the natural entropy-only proxy for the amount of risky content committed at the reveal boundary.

The mass form is important. The average $\bar{h}(N_t)$ measures uncertainty density per newly committed token, whereas

$$\Phi_t = |N_t| \bar{h}(N_t) \tag{A26}$$

also accounts for the volume of content committed at the step. A trajectory can be risky either because a few committed tokens are highly uncertain or because many moderately uncertain tokens are committed together. The entropy mass captures both effects.

The resulting interpretation is

$$\text{confidence-based reveal} \Longrightarrow \text{hard residual reservoir} \Longrightarrow \text{diagnostic entropy flux when revealed.} \tag{A27}$$

Unresolved entropy explains where difficulty is stored; revealing-token entropy mass measures how much of that difficulty is committed into the answer.

Appendix B.3. Pointwise Entropy Amplification Under Discrete Reveal

The previous subsection shows that confidence-based selection can create a hard residual reservoir, while reveal-boundary entropy flux records when residual uncertainty becomes committed answer content. We now explain why a specific reveal event can amplify downstream uncertainty, even though conditioning reduces entropy on average.

For a candidate reveal value a on N_t, define the post-commitment residual entropy

$$S_t(a) = H_q(X_{U_t} \mid X_{C_t}, X_{N_t} = a). \tag{A28}$$

A standard conditional entropy identity gives

$$\mathbb{E}_{a \sim q(X_{N_t} \mid X_{C_t})} S_t(a) = H_q(X_{U_t} \mid X_{C_t}, X_{N_t}), \tag{A29}$$

and hence

$$\mathbb{E}_a S_t(a) = H_q(X_{U_t} \mid X_{C_t}) - I_q(X_{N_t}; X_{U_t} \mid X_{C_t}). \tag{A30}$$

Since mutual information is nonnegative,

$$\mathbb{E}_a S_t(a) \leq H_q(X_{U_t} \mid X_{C_t}). \tag{A31}$$

Thus, conditioning on reveal values reduces residual entropy on average.

 doi:10.20944/preprints202606.2166.v1

However, decoding does not average over all possible values of a. It commits to a specific value a_t, often chosen by a local confidence rule. Define the pointwise entropy jump

$$\Delta S_t(a_t) = H_q(X_{U_t} \mid X_{C_t}, X_{N_t} = a_t) - H_q(X_{U_t} \mid X_{C_t}). \tag{A32}$$

Also define the pointwise excess term

$$\epsilon_t(a_t) = H_q(X_{U_t} \mid X_{C_t}, X_{N_t} = a_t) - \mathbb{E}_a H_q(X_{U_t} \mid X_{C_t}, X_{N_t} = a). \tag{A33}$$

Then

$$\Delta S_t(a_t) = \epsilon_t(a_t) - I_q(X_{N_t}; X_{U_t} \mid X_{C_t}). \tag{A34}$$

Therefore,

$$\Delta S_t(a_t) > 0 \quad \Longleftrightarrow \quad \epsilon_t(a_t) > I_q(X_{N_t}; X_{U_t} \mid X_{C_t}). \tag{A35}$$

This shows that average entropy reduction under conditioning is compatible with entropy amplification after a specific greedy commitment. A token may be locally confident under the current scaffold while still being globally incompatible with the remaining configuration, producing a positive pointwise residual entropy jump. In such cases, reveal-boundary entropy mass is a useful observable: it records uncertain commitments at the step where they become answer content.

Appendix B.4. Committed Tokens as Attention-Induced Boundary Conditions

Committed tokens do not disappear after being revealed. In a diffusion language model with bidirectional attention, committed positions continue to influence unresolved positions. We therefore treat committed tokens as boundary variables for the active unresolved subsystem.

Let $A_{ij}^{(\ell,h,t)}$ be the attention weight from position i to position j in layer ℓ, head h, and step t. Define the averaged attention kernel

$$\kappa_{ij}^{(t)} = \frac{1}{LH} \sum_{\ell=1}^{L} \sum_{h=1}^{H} A_{ij}^{(\ell,h,t)}. \tag{A36}$$

Since attention is not necessarily symmetric, we use the symmetrized kernel

$$\widetilde{\kappa}_{ij}^{(t)} = \frac{1}{2}\left(\kappa_{ij}^{(t)} + \kappa_{ji}^{(t)}\right) \tag{A37}$$

when writing an energy-like interaction functional.

Let z_i be a continuous relaxation of the token state at position i, for example an embedding or a probability vector. We use the following stylized interaction functional as an abstraction of bidirectional contextual coupling:

$$E_t(z_{V_t}; z_{C_t}) = \sum_{i \in V_t} V_i(z_i) + \sum_{i,j \in V_t} \widetilde{\kappa}_{ij}^{(t)} \Psi(z_i, z_j) + \sum_{\substack{i \in V_t \\ j \in C_t}} \widetilde{\kappa}_{ij}^{(t)} \Psi(z_i, z_j). \tag{A38}$$

The final term is the coupling between unresolved positions and committed positions. It represents the boundary field induced by the committed scaffold.

A coherent scaffold provides compatible boundary conditions, which can sharpen the conditional distributions of the remaining positions. A hallucinated or globally incompatible scaffold may instead impose conflicting constraints. To formalize this intuition, define a frustration functional after commitment:

$$J_t(C_t) = \min_{z_{U_t}} \left[\sum_{i,j \in U_t} \widetilde{\kappa}_{ij}^{(t)} d^2(z_i, T_{ij} z_j) + \sum_{\substack{i \in U_t \\ j \in C_t}} \widetilde{\kappa}_{ij}^{(t)} d^2(z_i, T_{ij} z_j) \right], \tag{A39}$$

 doi:10.20944/preprints202606.2166.v1

where T_{ij} is a compatibility transform and $d(\cdot,\cdot)$ is a distance in the relaxed state space. Small $J_t(C_t)$ means the unresolved variables can satisfy the imposed constraints coherently. Large $J_t(C_t)$ indicates a frustrated boundary condition.

Appendix B.5. Boundary Susceptibility and Unstable Scaffolds

We now connect boundary frustration with stability. Consider a smooth relaxed free-energy landscape

$$F_t(z_{U_t}; z_{C_t}) = E_t(z_{U_t}; z_{C_t}) - \tau_t S(z_{U_t}), \tag{A40}$$

where E_t is a relaxed energy, S is an entropy functional over unresolved states, and $\tau_t > 0$ is a temperature-like parameter.

Assume that, for fixed committed variables z_{C_t}, the unresolved subsystem is near a local stationary solution:

$$\nabla_{z_{U_t}} F_t(z^\star_{U_t}; z_{C_t}) = 0. \tag{A41}$$

Let

$$\mathcal{H}_t = \nabla^2_{z_{U_t} z_{U_t}} F_t(z^\star_{U_t}; z_{C_t}), \qquad \mathcal{G}_t = \nabla^2_{z_{U_t} z_{C_t}} F_t(z^\star_{U_t}; z_{C_t}). \tag{A42}$$

Proposition 1.

Assume that F_t is twice continuously differentiable in a neighborhood of $(z^\star_{U_t}, z_{C_t})$, that Equation (A41) holds, and that $\mathcal{H}_t$ is nonsingular. Then, for a sufficiently small boundary perturbation δz_{C_t}, the induced change in the local stationary solution satisfies

$$\delta z^\star_{U_t} = -\mathcal{H}_t^{-1} \mathcal{G}_t \delta z_{C_t} + O(\|\delta z_{C_t}\|^2). \tag{A43}$$

Consequently, the boundary susceptibility is controlled by

$$\chi_t = \|\mathcal{H}_t^{-1} \mathcal{G}_t\|. \tag{A44}$$

If $\mathcal{H}_t \succeq \gamma I$ for some $\gamma > 0$, then

$$\|\delta z^\star_{U_t}\| \le \frac{\|\mathcal{G}_t\|}{\gamma} \|\delta z_{C_t}\| + O(\|\delta z_{C_t}\|^2). \tag{A45}$$

When $\lambda_{\min}(\mathcal{H}_t)$ is close to zero, the susceptibility can become large. If $\mathcal{H}_t$ has a negative eigenvalue, the stationary point is unstable.

Proof.

The stationary condition is

$$\nabla_{z_U} F_t(z^\star_U(z_C); z_C) = 0. \tag{A46}$$

Perturb z_C to $z_C + \delta z_C$ and the stationary solution to $z^\star_U + \delta z^\star_U$. A first-order Taylor expansion gives

$$0 = \nabla_{z_U} F_t(z^\star_U + \delta z^\star_U; z_C + \delta z_C) \tag{A47}$$

$$= \nabla_{z_U} F_t(z^\star_U; z_C) + \mathcal{H}_t \delta z^\star_U + \mathcal{G}_t \delta z_C + O(\|\delta z_C\|^2). \tag{A48}$$

The first term is zero by stationarity. Therefore,

$$\mathcal{H}_t \delta z^\star_U + \mathcal{G}_t \delta z_C = O(\|\delta z_C\|^2). \tag{A49}$$

Since $\mathcal{H}_t$ is nonsingular,

$$\delta z^\star_U = -\mathcal{H}_t^{-1} \mathcal{G}_t \delta z_C + O(\|\delta z_C\|^2). \tag{A50}$$

 doi:10.20944/preprints202606.2166.v1

Taking norms gives

$$\|\delta z_U^\star\| \le \|\mathcal{H}_t^{-1}\mathcal{G}_t\|\|\delta z_C\| + O(\|\delta z_C\|^2). \tag{A51}$$

If $\mathcal{H}_t \succeq \gamma I$, then $\|\mathcal{H}_t^{-1}\| \le 1/\gamma$, and hence Equation (A45) follows. If $\mathcal{H}_t$ has a negative eigenvalue, the stationary point is not a local minimum of the free energy and is therefore unstable. □

This proposition gives a precise meaning to stable and unstable committed scaffolds. A coherent commitment corresponds to a well-conditioned basin, where small boundary changes have limited effect on the unresolved subsystem. An incompatible commitment can create a flat, metastable, or unstable scaffold, in which later denoising updates are amplified through the inverse Hessian.

If the entropy functional is locally smooth, then

$$\delta S = \langle \nabla_{z_U} S, \delta z_U^\star \rangle + O(\|\delta z_U^\star\|^2). \tag{A52}$$

Thus, an unstable scaffold can induce a large entropy response whenever the boundary perturbation has a component in an entropy-increasing direction. This formalizes the intuition that a hallucinated commitment may appear locally confident at the moment of reveal but later destabilize the residual subsystem.

Appendix B.6. From Boundary Entropy Flux to TRE

The analysis above suggests that the most diagnostic uncertainty is not the bulk entropy stored in the unresolved reservoir, but the entropy mass that crosses the reveal boundary and becomes committed answer content. Let $N_t = \Delta R_t$ denote the newly revealed positions at step t. We define the reveal-boundary entropy flux as

$$\Phi_t = \sum_{i \in N_t} H_t(i). \tag{A53}$$

A large Φ_t means that the decoder is committing either many moderately uncertain tokens or a smaller number of highly uncertain tokens. In both cases, Φ_t measures the total amount of uncertain content entering the answer at step t.

TRE aggregates this reveal-boundary flux with a monotone late-stage weight:

$$\mathrm{TRE}(x) = \sum_{t=1}^{T} w\left(\frac{t}{T}\right)\Phi_t, \qquad w(u) = u. \tag{A54}$$

Equivalently,

$$\mathrm{TRE}(x) = \sum_{t=1}^{T} \frac{t}{T} \sum_{i \in N_t} H_t(i). \tag{A55}$$

This is the revealing-token entropy mass score used in the main text. It measures the total amount of uncertain commitment crossing the reveal boundary, with stronger emphasis on late denoising.

The mass form is intentional. An average over newly revealed tokens measures uncertainty density per committed token, while the mass Φ_t measures the total amount of uncertain content committed at that step. Hallucination risk can depend on both: a trajectory is risky not only when individual commitments are uncertain, but also when a large volume of uncertain content is committed late. This should not be interpreted as a count-only effect. The reveal count $|N_t|$ measures how much content is committed, while the entropy values $H_t(i)$ measure how uncertain those commitments are. Their product-like mass aggregation captures total uncertain commitment crossing the reveal boundary.

Thus, TRE measures persistent reveal-boundary entropy flux: uncertainty that is committed into the answer and weighted more strongly when it occurs late in denoising.

 doi:10.20944/preprints202606.2166.v1

Appendix B.7. Summary of the Formal Picture

The stylized analysis supports the detector design through the following chain:

$$\begin{aligned}
&\text{discrete denoising + reveal projection} &&\Longrightarrow \text{a moving commitment boundary,} && \text{(A56)}\\
&\text{confidence-based reveal} &&\Longrightarrow \text{a hard residual reservoir,} && \text{(A57)}\\
&\text{reveal events} &&\Longrightarrow \text{residual uncertainty crosses into the answer,} && \text{(A58)}\\
&\text{committed tokens} &&\Longrightarrow \text{boundary conditions through bidirectional attention,} && \text{(A59)}\\
&\text{wrong commitments} &&\Longrightarrow \text{frustrated or unstable residual subsystem,} && \text{(A60)}\\
&\text{unstable residual subsystem} &&\Longrightarrow \text{late entropy amplification at the reveal boundary.} && \text{(A61)}
\end{aligned}$$

This formal picture does not claim a universal physical law of D-LLM hallucination. Rather, it explains the detector structure: unresolved entropy describes where residual difficulty is stored, while revealing-token entropy mass measures how much of that difficulty is committed into the answer. The proposed score therefore tracks reveal-boundary entropy flux and emphasizes late commitments, where uncertain content is less likely to be revised away.

Appendix C. Detailed Experimental Setup

Appendix C.1. Datasets and Benchmarks

We evaluate TRE on three widely used question-answering benchmarks that require different forms of factual reasoning. These datasets allow us to examine whether a hallucination detector can remain effective across diverse answer-generation scenarios, ranging from direct factual recall to multi-hop reasoning and commonsense inference.

- **TriviaQA** [46] is an open-domain question-answering dataset centered on factual knowledge. Since many questions require identifying specific entities, dates, locations, or events, this benchmark is useful for evaluating whether a detector can recognize failures in knowledge-intensive generations.
- **HotpotQA** [45] contains questions that often require combining multiple pieces of evidence before producing the final answer. Compared with single-hop factual recall, this setting introduces additional reasoning complexity, making it suitable for testing whether hallucination detection methods can capture errors that arise during compositional or multi-step reasoning.
- **CommonsenseQA** [55] focuses on commonsense reasoning. The questions typically require models to rely on implicit everyday knowledge rather than simply matching surface-level facts. This benchmark therefore complements the other two datasets by testing hallucination detection in scenarios where correctness depends on plausible world knowledge and commonsense associations.

Appendix C.2. Model Backbones

Our experiments are conducted on two representative diffusion-based large language models, LLaDA-8B-Instruct [16] and Dream-7B-Instruct [44]. Both models generate responses through an iterative denoising process rather than the standard left-to-right autoregressive decoding mechanism. This makes them suitable testbeds for studying hallucination detection methods that exploit intermediate denoising behavior.

Using two different D-LLM backbones allows us to evaluate whether the proposed detector captures general properties of diffusion-based generation instead of overfitting to the trajectory characteristics of a single model [57]. For a given benchmark and backbone, all detectors are evaluated on the same set of generated responses whenever applicable, ensuring that performance differences mainly reflect the detectors themselves rather than variations in sampled answers.

 doi:10.20944/preprints202606.2166.v1

Appendix C.3. Baseline Configurations

We compare TRE with both training-based and training-free hallucination detection baselines. These methods differ in whether they require labeled examples for detector training and in which signals they use to estimate factual reliability.

Training-based Detectors.

Training-based methods learn a mapping from model-derived features to correctness labels [58, 59]. In our experiments, they are trained separately for each benchmark using the corresponding automatically annotated training split, and are then evaluated on held-out examples from the same benchmark.

- **CCS** [30] learns a contrastive direction in the representation space by encouraging consistency between paired views of truthful and untruthful behavior. It provides a representative latent-probing baseline for detecting whether internal activations encode factual correctness.
- **TSV** [52] constructs a truthfulness-oriented separator vector from latent representations. The learned direction is then used to assign truthfulness scores to model outputs according to their positions in the hidden-state space.
- **TraceDet** [11] is a trajectory-based detector designed for diffusion language models. Instead of relying only on the final answer, it analyzes information from the denoising process and aggregates trajectory-level signals for hallucination detection. Since TraceDet is a strong recent detector for D-LLMs, we treat it as the primary training-based baseline.

Training-Free Detectors.

Training-free detectors estimate uncertainty or inconsistency directly from model outputs, token-level statistics, multiple sampled generations, or hidden representations [60].

- **Perplexity** [53] uses the likelihood assigned by the model to its generated sequence as a confidence signal. Responses with lower likelihood are generally treated as less reliable.
- **LN-Entropy** [54] measures predictive uncertainty through length-normalized entropy. By averaging uncertainty over the generated sequence, it reduces the bias introduced by different response lengths.
- **Semantic Entropy** [21] estimates uncertainty from the semantic diversity of multiple sampled responses. Generations that express inconsistent meanings are assigned higher uncertainty, even if their surface forms differ only partially.
- **Lexical Similarity** [22] evaluates the agreement among multiple sampled outputs using surface-level textual overlap. Lower similarity across samples suggests weaker generation stability and potentially higher hallucination risk.
- **EigenScore** [23] derives a confidence score from the spectral properties of hidden representations. It captures variation in the internal representation space and uses this structure as an indicator of response reliability.

These methods are applied directly without additional optimization on the target datasets. For Dream-7B-Instruct, we do not report Perplexity and LN-Entropy because stable token-level logits are not reliably available in our experimental interface. This restriction prevents us from computing likelihood- or entropy-based scores in a consistent manner for that backbone.

Appendix C.4. Implementation Details

Dataset Construction.

For each benchmark, we randomly select 800 question–answer pairs from the official evaluation split, following the scale used in recent hallucination detection studies. Each example contains a question, the corresponding reference answer, and the response generated by the evaluated D-LLM.

 doi:10.20944/preprints202606.2166.v1

When multiple reference answers are available, we keep all of them and regard a generated response as correct if it is semantically consistent with any valid reference answer.

Response Generation.

For each model–dataset pair, we use a fixed prompt template and the same decoding configuration to generate responses. The generated responses are saved before applying any detector, ensuring that all compared methods are evaluated on identical model outputs rather than different sampled answers.

Decoding Configuration.

We use deterministic decoding for response generation. Specifically, the temperature is set to 0, while `top_p` and `top_k` are left unspecified, so no nucleus or top-k sampling is applied. Models are loaded in `float16` precision, and `cfg_scale` is set to 0.0, meaning that no additional classifier-free guidance scaling is used. These settings are fixed across all datasets and backbones.

Automatic Annotation.

To obtain correctness labels, we use **GPT-4o mini** as an automatic judge. Given a question, its reference answer, and the response generated by the evaluated D-LLM, GPT-4o mini is prompted to determine whether the generated response is semantically consistent with the reference answer. The judge is instructed to focus on factual correctness rather than surface-form overlap, so that paraphrases, abbreviations, or equivalent expressions are not incorrectly marked as wrong.

To improve label reliability, we adopt a conservative re-evaluation procedure. Responses initially judged as incorrect are sent to GPT-4o mini for a second judgment with the same question, reference answer, and generated response. If the two judgments are inconsistent, the example is discarded from the final evaluation set. After this filtering step, the remaining annotated examples are used as the final evaluation set for TRE and all applicable baselines.

Feature Extraction and Scoring.

TRE does not require detector training. It directly computes hallucination detection scores from intermediate denoising information collected during the original D-LLM generation process. Thus, no task-specific parameter update or additional supervised training is performed for TRE.

Computational Environment.

All experiments are implemented with PyTorch and the HuggingFace Transformers library. Model inference, feature extraction, and detector scoring are conducted on a single NVIDIA Quadro RTX 6000 GPU with 24GB VRAM. Due to this single-GPU setting, experiments are run sequentially rather than in large-scale parallel batches.

We do not compare with [10,12], because their source code is not publicly available at the time of our experiments. Including these methods without official implementations would require nontrivial re-implementation choices and could introduce unfair or unreliable comparisons.

Appendix D. Additional Experimental Results

Appendix D.1. Additional Ablation Study

We further examine the design choices on Dream to evaluate whether the observations generalize across D-LLM backbones. As shown in Table A1, the results show similar trends to those on LLaDA. ❶ Replacing newly revealed-token evidence with all-token, unrevealed-token, or revealed-token entropy consistently weakens the detector, confirming the importance of focusing on the reveal boundary. ❷ For temporal aggregation, linear weighting performs competitively with exponential weighting and hard last-30% selection, while avoiding extra hyperparameters. ❸ These results suggest that the effectiveness of TRE does not depend on a specific backbone, and that late-weighted newly revealed-token entropy remains a robust signal for D-LLM hallucination detection.

 doi:10.20944/preprints202606.2166.v1

Table A1. Ablation study on Dream.

Variant	TriQA	HotQA	CSQA
All tokens $\bar{H}$	74.6	73.8	63.9
Unrevealed $\bar{H}^u$	76.4	75.7	60.2
Revealed $\bar{H}^r$	62.1	62.9	63.4
Average	83.2	79.7	77.4
Exponential	**83.9**	80.4	**77.7**
Hard last-30%	83.5	80.2	**77.7**
TRE (ours)	**83.9**	**80.6**	**77.7**

Appendix D.2. Additional Case Studies

We provide additional hallucinated examples in Figure A3 to further illustrate the uncertainty dynamics captured by TRE. Similar to the main case study, the generated tokens at each step (s) are listed with their revealing entropy (H) values. The results lead to the following observations. ❶ Hallucinated answer tokens are consistently revealed at late denoising steps. For example, the model generates "sail." instead of "Merchant mariner", and "Grant Sunab" instead of "Vince Cable", with these erroneous tokens appearing near the end of the denoising process. ❷ When these hallucinated answers become newly revealed tokens, their entropy increases sharply, while earlier tokens usually have much smaller entropy. This further verifies that late-stage newly revealed-token entropy provides an informative and consistent signal for hallucination detection.

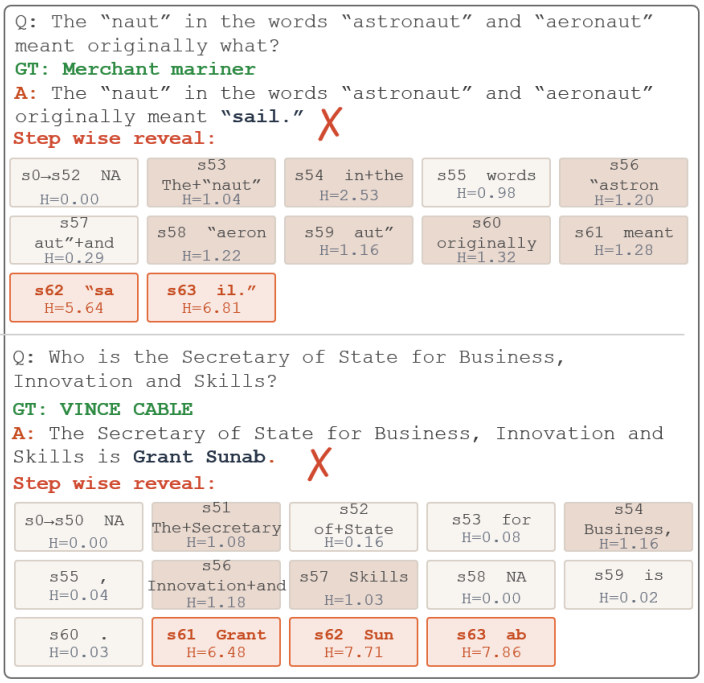


Figure A3. Additional hallucinated case study examples. Hallucinated answer tokens are often newly revealed at late denoising steps and exhibit sharply increased entropy.

Appendix E. Limitations and Broader Impacts

While TRE provides a simple and efficient training-free metric for hallucination detection in diffusion language models, it still has several limitations. First, TRE is based on entropy signals from the denoising process and does not explicitly use semantic knowledge. Therefore, it may be less effective when a model produces incorrect answers with high confidence. Second, our experiments are

 doi:10.20944/preprints202606.2166.v1

conducted on question-answering benchmarks with two representative D-LLM backbones. Evaluating TRE on more tasks, domains, and future diffusion language models would further verify its generality.

This work may help improve the reliability of diffusion language models by providing a lightweight hallucination detection signal without additional detector training or repeated generation. However, TRE should be used as an auxiliary indicator rather than a guarantee of factual correctness. In practical applications, especially in high-stakes scenarios, its predictions should be combined with human review or external verification when necessary. This paper releases only an anonymized supplementary code package for the detector and does not release new datasets, model checkpoints, or high-risk generative models.

References

1. Savinov, N.; Chung, J.; Binkowski, M.; Elsen, E.; Oord, A.v.d. Step-unrolled denoising autoencoders for text generation. In Proceedings of the International Conference on Learning Representations, 2022.
2. Li, X.; Thickstun, J.; Gulrajani, I.; Liang, P.S.; Hashimoto, T.B. Diffusion-lm improves controllable text generation. *Advances in neural information processing systems* **2022**, *35*, 4328–4343.
3. Lou, A.; Meng, C.; Ermon, S. Discrete diffusion modeling by estimating the ratios of the data distribution. In Proceedings of the International Conference on Machine Learning, 2024.
4. Israel, D.; Broeck, G.V.d.; Grover, A. Accelerating diffusion llms via adaptive parallel decoding. *arXiv preprint arXiv:2506.00413* **2025**.
5. Arriola, M.; Gokaslan, A.; Chiu, J.T.; Yang, Z.; Qi, Z.; Han, J.; Sahoo, S.S.; Kuleshov, V. Block diffusion: Interpolating between autoregressive and diffusion language models. In Proceedings of the International Conference on Learning Representations, 2025.
6. Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.J.; Madotto, A.; Fung, P. Survey of hallucination in natural language generation. *ACM computing surveys* **2023**, *55*, 1–38.
7. 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. *ACM Transactions on Information Systems* **2025**, *43*, 1–55.
8. Pan, J.; Zheng, Y.; Tan, Y.; Liu, Y. A Survey of Generalization of Graph Anomaly Detection: From Transfer Learning to Foundation Models. In Proceedings of the The 16th IEEE International Conference on Knowledge Graphs, 2025.
9. Duan, J.; Cheng, H.; Wang, S.; Zavalny, A.; Wang, C.; Xu, R.; Kailkhura, B.; Xu, K. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In Proceedings of the Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2024, pp. 5050–5063.
10. Qian, Y.; Tan, Y.; Liu, Y.; Yu, W.; Pan, S. DynHD: Hallucination Detection for Diffusion Large Language Models via Denoising Dynamics Deviation Learning. *arXiv preprint arXiv:2603.16459* **2026**.
11. Chang, S.; Yu, J.; Wang, W.; Chen, Y.; Yu, J.; Torr, P.; Gu, J. TraceDet: Hallucination Detection from the Decoding Trace of Diffusion Large Language Models. In Proceedings of the International Conference on Learning Representations, 2026.
12. Hemmat, A.; Torr, P.; Chen, Y.; Yu, J. TDGNet: Hallucination Detection in Diffusion Language Models via Temporal Dynamic Graphs. *arXiv preprint arXiv:2602.08048* **2026**.
13. Manakul, P.; Liusie, A.; Gales, M. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In Proceedings of the Proceedings of the 2023 conference on empirical methods in natural language processing, 2023, pp. 9004–9017.
14. Farquhar, S.; Kossen, J.; Kuhn, L.; Gal, Y. Detecting hallucinations in large language models using semantic entropy. *Nature* **2024**, *630*, 625–630.
15. Sahoo, S.S.; Arriola, M.; Schiff, Y.; Gokaslan, A.; Marroquin, E.; Chiu, J.T.; Rush, A.; Kuleshov, V. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems* **2024**, *37*, 130136–130184.
16. Nie, S.; Zhu, F.; You, Z.; Zhang, X.; Ou, J.; Hu, J.; Zhou, J.; Lin, Y.; Wen, J.R.; Li, C. Large language diffusion models. *arXiv preprint arXiv:2502.09992* **2025**.
17. Zhang, Y.; Li, Y.; Cui, L.; Cai, D.; Liu, L.; Fu, T.; Huang, X.; Zhao, E.; Zhang, Y.; Chen, Y.; et al. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models. *Computational Linguistics* **2025**, *51*, 1373–1418.

 doi:10.20944/preprints202606.2166.v1

18. Zhang, T.; Qiu, L.; Guo, Q.; Deng, C.; Zhang, Y.; Zhang, Z.; Zhou, C.; Wang, X.; Fu, L. Enhancing uncertainty-based hallucination detection with stronger focus. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 915–932.
19. Chen, Q.; Li, S.; Liu, Y.; Pan, S.; Webb, G.I.; Zhang, S. Uncertainty-aware graph neural networks: A multihop evidence fusion approach. *IEEE Transactions on Neural Networks and Learning Systems* **2025**.
20. 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. *arXiv preprint arXiv:2207.05221* **2022**.
21. Kuhn, L.; Gal, Y.; Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In Proceedings of the International Conference on Learning Representations, 2023.
22. Lin, Z.; Trivedi, S.; Sun, J. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research* **2024**.
23. Chen, C.; Liu, K.; Chen, Z.; Gu, Y.; Wu, Y.; Tao, M.; Fu, Z.; Ye, J. INSIDE: LLMs' internal states retain the power of hallucination detection. In Proceedings of the International Conference on Learning Representations, 2024.
24. Kossen, J.; Han, J.; Razzak, M.; Schut, L.; Malik, S.; Gal, Y. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv preprint arXiv:2406.15927* **2024**.
25. Bi, X.; Chen, C.; Chen, C.; Lv, X.; Zhu, J.; Ma, H.; Zuo, E. PatchFusionMLP: A scalable multi-resolution MLP framework for time series prediction. *Pattern Recognition* **2026**, p. 113263.
26. Azaria, A.; Mitchell, T. The internal state of an LLM knows when it's lying. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2023, 2023, pp. 967–976.
27. Zhang, S.; Yu, T.; Feng, Y. Truthx: Alleviating hallucinations by editing large language models in truthful space. In Proceedings of the Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2024, pp. 8908–8949.
28. Sriramanan, G.; Bharti, S.; Sadasivan, V.S.; Saha, S.; Kattakinda, P.; Feizi, S. Llm-check: Investigating detection of hallucinations in large language models. *Advances in Neural Information Processing Systems* **2024**, *37*, 34188–34216.
29. Zhang, F.; Yu, P.; Yi, B.; Zhang, B.; Li, T.; Liu, Z. Prompt-guided internal states for hallucination detection of large language models. In Proceedings of the Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2025, pp. 21806–21818.
30. Burns, C.; Ye, H.; Klein, D.; Steinhardt, J. Discovering latent knowledge in language models without supervision. In Proceedings of the International Conference on Learning Representations, 2023.
31. 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. *arXiv preprint arXiv:2310.01405* **2023**.
32. Li, J.; Cheng, X.; Zhao, X.; Nie, J.Y.; Wen, J.R. Halueval: A large-scale hallucination evaluation benchmark for large language models. In Proceedings of the Proceedings of the 2023 conference on empirical methods in natural language processing, 2023, pp. 6449–6464.
33. Min, S.; Krishna, K.; Lyu, X.; Lewis, M.; Yih, W.t.; Koh, P.; Iyyer, M.; Zettlemoyer, L.; Hajishirzi, H. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 12076–12100.
34. Dziri, N.; Kamalloo, E.; Milton, S.; Zaïane, O.R.; Yu, M.; Ponti, E.M.; Reddy, S. Faithdial: A faithful benchmark for information-seeking dialogue. *Transactions of the Association for Computational Linguistics* **2022**, *10*, 1473–1490.
35. Guo, Z.; Tan, F. Lost in Diffusion: Uncovering Hallucination Patterns and Failure Modes in Diffusion Large Language Models. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2026, 2026.
36. Varshney, N.; Mishra, S.; Baral, C. Investigating selective prediction approaches across several tasks in iid, ood, and adversarial settings. In Proceedings of the Findings of the association for computational linguistics: Acl 2022, 2022, pp. 1995–2002.
37. Austin, J.; Johnson, D.D.; Ho, J.; Tarlow, D.; Van Den Berg, R. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems* **2021**, *34*, 17981–17993.

 doi:10.20944/preprints202606.2166.v1

38. Hoogeboom, E.; Nielsen, D.; Jaini, P.; Forré, P.; Welling, M. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in neural information processing systems* **2021**, *34*, 12454–12465.
39. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Advances in neural information processing systems* **2020**, *33*, 1877–1901.
40. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. Language models are unsupervised multitask learners. *OpenAI blog* **2019**, *1*, 9.
41. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems* **2020**, *33*, 6840–6851.
42. Dhariwal, P.; Nichol, A. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems* **2021**, *34*, 8780–8794.
43. Luo, C. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970* **2022**.
44. Ye, J.; Xie, Z.; Zheng, L.; Gao, J.; Wu, Z.; Jiang, X.; Li, Z.; Kong, L. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487* **2025**.
45. Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; Manning, C.D. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Proceedings of the Proceedings of the 2018 conference on empirical methods in natural language processing, 2018, pp. 2369–2380.
46. Joshi, M.; Choi, E.; Weld, D.S.; Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In Proceedings of the Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 1601–1611.
47. Maynez, J.; Narayan, S.; Bohnet, B.; McDonald, R. On faithfulness and factuality in abstractive summarization. In Proceedings of the Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, pp. 1906–1919.
48. Khalil, H.K.; Grizzle, J.W. *Nonlinear systems*; Vol. 3, Prentice hall Upper Saddle River, NJ, 2002.
49. Posa, M.; Kuindersma, S.; Tedrake, R. Optimization and stabilization of trajectories for constrained dynamical systems. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016, pp. 1366–1373.
50. Mézard, M.; Parisi, G.; Virasoro, M.A.; Thouless, D.J. Spin glass theory and beyond, 1988.
51. Binder, K.; Young, A.P. Spin glasses: Experimental facts, theoretical concepts, and open questions. *Reviews of Modern physics* **1986**, *58*, 801.
52. Park, S.; Du, X.; Yeh, M.H.; Wang, H.; Li, Y. Steer llm latents for hallucination detection. In Proceedings of the International Conference on Machine Learning, 2025.
53. Ren, J.; Luo, J.; Zhao, Y.; Krishna, K.; Saleh, M.; Lakshminarayanan, B.; Liu, P.J. Out-of-distribution detection and selective generation for conditional language models. In Proceedings of the International Conference on Learning Representations, 2023.
54. Malinin, A.; Gales, M. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650* **2020**.
55. Talmor, A.; Herzig, J.; Lourie, N.; Berant, J. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In Proceedings of the Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 4149–4158.
56. Zuo, E.; Zhong, J.; Chen, C.; Chen, C.; Ubul, K.; Lv, X. Rethinking unsupervised time series anomaly detection: Dynamic attention based on route inverse-masking. *Applied Soft Computing* **2025**, p. 113971.
57. Pan, J.; Liu, Y.; Zhou, C.; Xiong, F.; Liew, A.W.C.; Pan, S. Correcting False Alarms from Unseen: Adapting Graph Anomaly Detectors at Test Time. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2026.
58. Tan, Y.; Long, G.; Jiang, J.; Zhang, C. Influence-oriented personalized federated learning. *arXiv preprint arXiv:2410.03315* **2024**.

 doi:10.20944/preprints202606.2166.v1

59. Li, Y.; Qu, H.; Chen, C.; Lv, X.; Zuo, E.; Wang, K.; Cai, X. TreeXformer: Extracting tabular feature-context information using tree-structured semantics. *Information Processing & Management* **2025**, *62*, 104291.
60. Miao, R.; Liu, Y.; Wang, Y.; Shen, X.; Tan, Y.; Dai, Y.; Pan, S.; Wang, X. Blindguard: Safeguarding llm-based multi-agent systems under unknown attacks. In Proceedings of the Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics, 2026.