The Digital Dunning-Kruger Effect: Decoupling Hallucinations via Geometric Hidden-state Observation for Semantic Truthfulness

Yueheng Mao[1,2], Min Yu[1,2,†], Gengwang Li[1], Jianguo Jiang[1,2], Gang Li[3], Meng Zhang[1,2], Zhen Xu[1,2], Weiqing Huang[1,2], Ming Liu[3,†]

[1]Institute of Information Engineering, Chinese Academy of Sciences, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, China
[3]School of Information Technology, Deakin University, Australia
[†]Corresponding authors. **Correspondence:** yumin@iie.ac.cn, m.liu@deakin.edu.au

Abstract

Large Language Models (LLMs) often generate overconfident yet factually incorrect hallucinations. Current detection paradigms suffer from a trade-off between the high accuracy of computationally expensive black-box methods and the inability of white-box methods to detect stubborn hallucinations. To bridge this gap, we propose **GHOST** (**G**eometric **H**idden-state **O**bservation for **S**emantic **T**ruthfulness), an efficient white-box framework for hallucination detection in LLMs. We primarily target *confused hallucinations* marked by internal reasoning instability, while also capturing *stubborn hallucinations* characterized by premature layer-wise convergence as a complementary signal. By integrating internal geometric dynamics with output probability distributions, GHOST constructs a high-dimensional feature space for non-linear truthfulness classification. Extensive evaluations on FinanceBench, RAGTruth, HaluEval, and PopQA show that GHOST outperforms white-box baselines and achieves competitive black-box performance while reducing computational overhead by over 90%, offering a robust solution for real-time detection.

1 Introduction

Large Language Models (LLMs) have achieved remarkable breakthroughs in the field of Natural Language Processing (NLP) (Zhao et al., 2025; Farquhar et al., 2024). However, underlying these capabilities lies a critical flaw: the models occasionally generate factually incorrect, logically fallacious, or unsubstantiated statements with high confidence (Ji et al., 2023). This phenomenon, termed *hallucination*, severely impedes the practical deployment of LLMs in high-stakes domains such as healthcare, law, and finance, and remains a pivotal challenge awaiting resolution in the field (Zhang et al., 2025b).

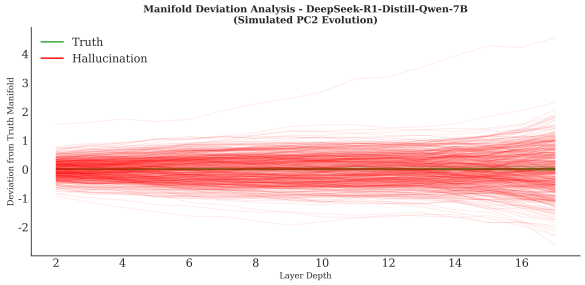

Figure 1: Manifold Deviation Analysis illustrating the Digital Dunning-Kruger Effect.

The academic community categorizes hallucination detection into black-box and white-box paradigms. Black-box methods (Manakul et al., 2023; Cohen et al., 2023; Min et al., 2023) utilize multi-sample consistency or post-hoc verification, yet prohibitive computational overhead and inference latency impede real-time deployment. Conversely, white-box approaches (Chuang et al., 2024) leverage output logits or singular internal metrics. These coarse-grained indicators often fail to exploit rich internal dynamics, yielding limited discriminative power when models exhibit overconfidence in erroneous knowledge.

Recent investigations into internal mechanisms provide granular perspectives. While Xu et al. (Xu et al., 2020) employed V-usable information to assess model faithfulness, their analysis remains confined to final output layers and neglects dynamic transitions. Similarly, Kim et al. (Kim et al., 2025) introduced Layer-wise Information Deficiency (LI), yet this framework attributes hallucinations solely to information loss. Consequently, LI fails to identify stubborn hallucinations arising from pre-training biases or erroneous memorization. To move beyond coarse-grained detection, InFi-Check (Bai et al., 2026) introduces an interpretable fact-checking framework that classifies fine-grained error types and provides supporting evidence from a textual perspective.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 21786–21800
July 2-7, 2026 ©2026 Association for Computational Linguistics

We posit that hallucination reflects a cognitive dissonance analogous to the Dunning-Kruger Effect (Kruger and Dunning, 1999), where models overestimate their competence. Specifically, we identify two distinct mechanisms: *Stubborn Hallucinations*, characterized by premature convergence and high epistemic overconfidence despite insufficient factual grounding, and *Confused Hallucinations*, manifested as internal reasoning instability when resolving conflicting semantic signals. Experiments have found that the reasoning process of the truth is smooth, while illusions are fluctuating and part of them highly overlap with the truth, as shown in the Figure 1, visualizes the inter-layer hidden state evolution via second principal component (PC2) (Chen et al., 2024) deviation.

To capture these subtle cognitive signatures, we propose **GHOST** (**G**eometric **H**idden-state **O**bservation for **S**emantic **T**ruthfulness). Diverging from previous approaches dependent on static representations, GHOST conceptualizes the inference process as a dynamic latent trajectory within a high-dimensional semantic manifold. Our primary contributions are summarized as follows:

(1) Cognitive-Inspired Taxonomy: We formalize a novel hallucination taxonomy distinguishing *Confused Hallucinations* from *Stubborn Hallucinations* based on their distinct geometric manifestations, providing a theoretical foundation for why conventional uncertainty metrics fail during high-confidence erroneous generation.

(2) The GHOST Framework: We introduce a white-box framework leveraging internal geometric dynamics. By quantifying *Representation Turbulence* and *Stubbornness* across hidden layers, GHOST constructs a multi-dimensional feature space to disentangle truthful reasoning from deceptive generation without structural modifications.

(3) Efficiency and Generalizability: Evaluations on FinanceBench, RAGTruth, HaluEval, and PopQA demonstrate that GHOST significantly outperforms existing white-box baselines. Remarkably, GHOST achieves performance competitive with black-box methods while reducing computational overhead by over 90%, facilitating robust real-time deployment.

2 Related Works

2.1 Hallucination Detection Paradigms

Black-box Methods primarily assess the veracity of responses through output-level consistency or external verification. **SelfCheckGPT** (Manakul et al., 2023) and **LM-Polygraph** (Fadeeva et al., 2023) employ stochastic sampling to quantify semantic consistency, while **FactScore** (Min et al., 2023) introduces retrieval-augmented verification for fine-grained factual checking. Despite their high precision, these paradigms are often hindered by prohibitive sampling latency and substantial computational overhead, limiting their utility in real-time applications. Recently, data-centric approaches like NOVA (Si et al., 2025) have been proposed to mitigate hallucinations by filtering instruction tuning data based on the model's familiarity with the knowledge.

White-box Methods aim to circumvent these costs by leveraging model-internal signals. **Semantic Entropy** (Farquhar et al., 2024) formalizes uncertainty estimation at the semantic level, and **Lookback Lens** (Chuang et al., 2024) utilizes attention maps to identify contextual hallucinations. However, these paradigms frequently rely on coarse-grained uncertainty metrics, which may fail to detect "stubborn" hallucinations where the model generates erroneous content with high confidence.

2.2 Probing Internal Representations

A growing body of research suggests that the hidden states of LLMs inherently encode latent truthfulness. **SAPLMA** (Azaria and Mitchell, 2023) and **ITI** (Li et al., 2023) demonstrate that linear probes or directions within intermediate representations can effectively disentangle truthful behaviors from deceptive ones, enabling targeted inference-time interventions. CANOE (Si et al., 2026) introduces a reinforcement learning framework that utilizes synthetic tasks and rule-based rewards to enhance contextual faithfulness without human annotation. **INSIDE** (Chen et al., 2024) further extends this by introducing EigenScore to measure consistency within the spectral domain of internal states, while **PRISM** (Zhang et al., 2025a) leverages prompting to align hidden states with more salient truth-related manifolds. Nevertheless, these approaches predominantly focus on static feature distributions at specific layers, potentially oversimplifying the complex dynamic evolution of hidden states across the entire Transformer stack.

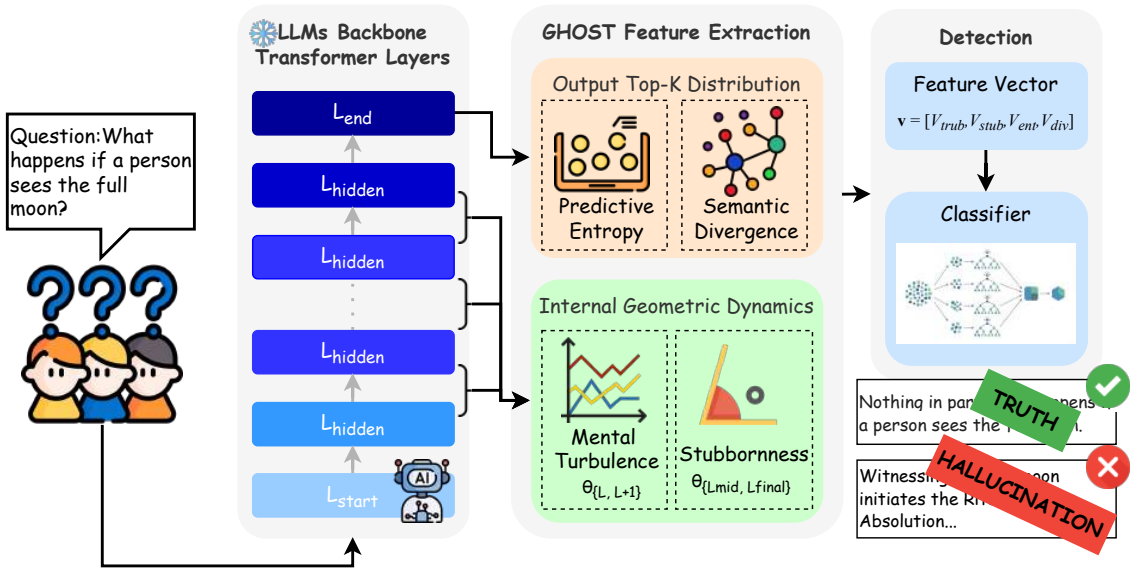

Figure 2: The schematic architecture of the proposed GHOST framework. The system integrates internal geometric trajectory features derived from hidden states with external semantic distribution features from output logits. These multi-dimensional indicators are processed by a non-linear classifier to identify confused and stubborn hallucinations.

2.3 Dynamic Geometric and Topological Analysis

Our work aligns with the emerging trend of analyzing the dynamic trajectory of LLMs. **LI** (Kim et al., 2025) analyzes cross-layer information dynamics and is most directly motivated by ambiguous prompts / unanswerable questions. **END** (Wu et al., 2025) utilizes cross-layer entropy signals to adjust decoding for factuality, yet it remains primarily focused on decoding-time mitigation. **TOHA** (Bazarova et al., 2025) explores the topological divergence of attention graphs and is restricted to RAG scenarios. **GHOST** distinguishes itself by providing a unified parametric framework that captures both transient turbulence and persistent rigidity in geometric trajectories, offering a more robust detection for both confused and stubborn hallucinations.

3 Method

Methodology. This section elucidates the high-dimensional features constituting the GHOST framework. Unlike black-box approaches that rely on external retrieval or multi-model sampling consistency (Goel et al., 2025), our method exploits internal state dynamics during generation. Accordingly, we construct a high-dimensional feature space integrating geometric dynamics and semantic distributions. This approach utilizes non-linear classifiers to capture the complex topological structures of truthfulness, as illustrated in Figure 2.

3.1 Problem Formulation

Given a Large Language Model $\mathcal{M}$ and an input prompt x, the model generates a response sequence $y = \{y_1, y_2, ..., y_N\}$. Our objective is to construct a detection function $f(x, y; \mathcal{M}) \rightarrow \{0, 1\}$ that identifies whether the response y contains factual errors. Unlike previous methods that rely on the hidden state of a specific "key token" (e.g., the last token), we leverage the geometric dynamics of the entire generation process. We denote the hidden state of the l-th layer for the i-th token in the sequence as $h_l^{(i)} \in R^d$, where $l \in [L_{start}, L_{end}]$ and d represents the hidden dimension.

Truthful responses consistently occupy a narrow manifold with smooth trajectories, suggesting stable semantic propagation. Hallucinations, however, demonstrate a clear bifurcation. One subset of red trajectories remains tightly clustered with the truthful paths, reflecting the "hyper-stability" of stubborn hallucinations where the model prematurely commits to incorrect priors. Another subset exhibits significant divergence starting around the midpoint of the network (∼50% relative depth), representing confused hallucinations characterized by late-stage reasoning disarray.

3.2 Internal Geometric Dynamics

Large Language Model (LLM) inference constitutes a hidden state evolution trajectory within the inter-layer semantic space. To capture global response characteristics, we employ a *Calculate-then-Average* aggregation strategy. Specifically, we first compute geometric metrics for each token in the sequence y, and subsequently average these values across the total sequence length N to derive the final feature vector. This approach ensures that the detector captures both transient reasoning uncertainties and persistent stubbornness throughout the entire generation chain.

3.2.1 Representation Turbulence

Drawing from the psychological theory of **Cognitive Dissonance** (Festinger, 1957), which describes the mental discomfort experienced when holding conflicting beliefs, we hypothesize that *Confused Hallucinations* arise when a model struggles to resolve contradictory internal information. This internal conflict manifests geometrically as drastic shifts in the hidden state trajectory between adjacent layers. We quantify this phenomenon as **Representation Turbulence**.

For the i-th token in the generated sequence, its turbulence score $v_{turb}^{(i)}$ measures the average cosine deviation across the selected layer range $[L_{start}, L_{end}]$:

$$v_{turb}^{(i)} = \frac{1}{L_{end} - L_{start}} \sum_{l=L_{start}}^{L_{end}-1} \left(1 - \frac{h_l^{(i)} \cdot h_{l+1}^{(i)}}{\|h_l^{(i)}\| \|h_{l+1}^{(i)}\|} \right) \tag{1}$$

The final turbulence feature for the entire response is defined as $V_{turb} = \frac{1}{N} \sum_{i=1}^{N} v_{turb}^{(i)}$. A significant increase in V_{turb} serves as a computational proxy for cognitive dissonance, indicating high internal conflict and instability in the model's reasoning path.

3.2.2 Stubbornness

Conversely, to capture the "illusory superiority" akin to the **Dunning-Kruger Effect** (Kruger and Dunning, 1999), we introduce the **Stubbornness** metric. This cognitive bias in LLMs manifests as *Stubborn Hallucinations*, where the model converges to a conclusion prematurely, despite a lack of factual grounding. This reflects a digital counterpart to the "peak of inflated expectations," where high confidence masks low informational competence.

We quantify Stubbornness by measuring the similarity between intermediate layer states and the final layer representation $h_{final}^{(i)}$ for each token:

$$v_{stub}^{(i)} = \frac{1}{L_{end} - L_{start} + 1} \sum_{l=L_{start}}^{L_{end}} \frac{h_l^{(i)} \cdot h_{final}^{(i)}}{\|h_l^{(i)}\| \|h_{final}^{(i)}\|} \tag{2}$$

The global stubbornness feature is computed as $V_{stub} = \frac{1}{N} \sum_{i=1}^{N} v_{stub}^{(i)}$. Within the GHOST framework, the coupling of high V_{stub} and low V_{turb} provides a distinct geometric fingerprint of stubborn hallucinations. This profile stands in sharp contrast to authentic reasoning, which typically exhibits moderate turbulence as the model iteratively refines its semantic output. Unlike Representation Turbulence, which quantifies inter-layer deviation, Stubbornness captures a distinct geometric regime: cases where inter-layer instability is minimal yet the model converges prematurely to an incorrect representation. It is therefore designed as a **complementary** signal rather than a primary discriminator.

3.3 Output Distribution Analysis

Beyond internal geometric features, the output layer probability landscape encapsulates critical semantic uncertainty. We introduce two metrics based on Top-K predictions to complement the internal hidden state analysis.

3.3.1 Predictive Entropy

Entropy is a classical uncertainty metric. Given context x, the model yields a next-token distribution $P(w|x)$. We select the top-K candidates (setting $K = 10$ for all experiments) and calculate their Shannon entropy over the top-K probability distribution. This metric depends entirely on output probabilities without accessing the embedding space:

$$V_{ent} = -\sum_{k=1}^{K} p_k \log p_k \tag{3}$$

where p_k represents the normalized probability of the k-th candidate token.

3.3.2 Semantic Divergence

To distinguish lexical synonyms from factual confusion, we measure the geometric dispersion of candidates in embedding space rather than token identity space. Specifically, Top-K selection is used only to define the candidate support of the next-token distribution and to filter out the long probability tail. Unlike Predictive Entropy, which

is computed over this truncated probability distribution, Semantic Divergence does not operate over probabilities. Instead, we retrieve the static *Input Embedding Layer* vectors $\{e_1, ..., e_K\}$ for the top-K candidates ($K = 10$), capturing intrinsic semantic discrepancies independent of contextual processing. This design ensures that semantically similar synonyms yield low divergence scores, while genuinely uncertain predictions over semantically distant candidates yield high scores. Their average pairwise cosine distance is defined as:

$$V_{div} = \frac{1}{K(K-1)} \sum_{i=1}^{K} \sum_{j \neq i}^{K} (1 - \frac{e_i \cdot e_j}{||e_i|| ||e_j||}) \quad (4)$$

3.4 Non-linear Hallucination Detector

We project responses into a four-dimensional feature space $\mathbf{v}$. Since linear classifiers fail to capture these non-linear decision boundaries, we benchmarked SVM, XGBoost, and Random Forest using rigorous randomized search cross-validation. Random Forest consistently yielded superior performance by robustly modeling high-dimensional feature interactions within the semantic manifold. Results are summarized in Table 1.

4 Experiments

In this chapter, we evaluate the effectiveness of our proposed GHOST method through extensive experiments on four mainstream hallucination evaluation benchmarks: FinanceBench, RAGTruth, PopQA, and HaluEval. We demonstrate empirically that GHOST is a rapid and effective approach for detecting whether the responses generated by LLMs contain hallucinations.

4.1 Experimental Setup

Baselines. We benchmark GHOST against six competitive baselines across three categories. Logit-based metrics include **Predictive Entropy** (Malinin and Gales, 2021), derived from output probability distributions. Internal-state methods encompass **INSIDE** (Chen et al., 2024), utilizing hidden state covariance eigenvalues, **LI** (Kim et al., 2025), quantifying layer-wise information deficiency, **LapEigvals** (Binkowski et al., 2025), a spectral approach based on attention map graph Laplacians, **UTH** (Liu et al., 2024), **HIDE** (Chatterjee et al., 2025) and **LoRA probe** (Obeso et al., 2026). For black-box consistency, we evaluate **SelfCheckGPT** (Manakul et al., 2023) using five sampled responses.

Models. We evaluate GHOST across four state-of-the-art LLMs strategically selected for their architectural and functional diversity. **Qwen2.5-1.5B-Instruct** (Yang et al., 2024) acts as a representative for *Small Language Models*, allowing us to probe capacity-induced hallucinations in resource-constrained scenarios. To ensure architectural generalization beyond the LLaMA lineage, we include **Gemma-3-4B-IT** (Gemma Team and Google DeepMind, 2025), which features distinct configurations like GeGLU activations. **Mistral-7B-Instruct-v0.3** (AI, 2024) serves as the industry-standard baseline to verify the practical relevance of our metrics in widely deployed 7B-scale models. Finally, we incorporate **DeepSeek-R1-Distill-Qwen-7B** (Guo et al., 2025) to examine the complex internal trajectories of *Reasoning Models*. This allows us to verify whether GHOST can effectively decode the "thinking processes" and logic loops inherent in Reinforcement Learning-optimized models, which are particularly susceptible to confused and stubborn hallucinations derived from distillation.

Datasets. We evaluate our method on four benchmarks targeting distinct hallucination facets. **FinanceBench** (Islam et al., 2023) provides complex financial questions to test the model's accuracy in professional domains. For retrieval-augmented scenarios, we utilize **RAGTruth** (Wu et al., 2023) to assess hallucinations occurring within external knowledge integration. Regarding general knowledge assessment, we employ the **HaluEval** (Li et al., 2023) QA subset containing 10,000 synthesized pairs. Additionally, **PopQA** (Mallen et al., 2023) enables investigation into long-tail entity hallucinations. We specifically analyze the subset where models fail to recall correct entities to determine if intrinsic metrics effectively distinguish genuine knowledge from the hallucination of obscure facts.

Ground Truth Annotation. For generative tasks, obtaining reliable binary labels is critical. We employed `gpt-oss-120b` (OpenAI, 2025), a large-scale open-source language model, as an automated annotator. The judge is instructed to classify the response as Hallucination only if it contradicts the reference or contains fabricated information, while treating semantic paraphrases as Truth.

We conducted a rigorous human evaluation on

Table 1: Comparison of different classifiers within the GHOST framework using **Qwen2.5-1.5B** features. **Random Forest** consistently outperforms other classifiers across all datasets. While XGBoost achieves competitive performance, Random Forest demonstrates superior robustness and stability in modeling the complex, non-linear geometric manifolds of hallucinations.

Classifier	PopQA		FinanceBench		RAGTruth		HaluEval	
	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑
Logistic Regression	0.8215	0.7842	0.8510	0.6534	0.7721	0.7105	0.8115	0.6890
SVM (RBF)	0.9054	0.8612	0.8745	0.7420	0.9123	0.8045	0.8321	0.7856
XGBoost	0.9382	0.9105	0.9216	0.7645	0.9410	0.8656	0.8805	0.8212
Random Forest	**0.9801**	**0.9529**	**0.9552**	**0.8214**	**0.9596**	**0.8993**	**0.9175**	**0.8698**

a stratified subset of 500 randomly selected samples. We calculated Cohen's Kappa coefficient (κ) to measure the inter-rater reliability between the human experts and the automated judge. The resulting score of $\kappa = 0.82$ indicates a strong alignment between human judgment and the automated annotator, validating the quality of our ground truth labels for large-scale evaluation.

To validate the reliability of the automatic judge, we additionally conduct a double-blind human evaluation with three domain experts. Annotators are blind to both the model method and the judge predictions; each item is independently labeled by all three experts, and disagreements are resolved by majority vote.

Evaluation Metrics. We treat hallucination detection as a binary classification task, employing **AUPRC** and **F1-score** as the primary evaluation metrics. **AUPRC** serves as our core threshold-independent metric, as it provides a more robust assessment of global discriminative performance under the class imbalances often found in hallucination benchmarks.

4.2 Main Results

Table 2 presents a detailed performance comparison of different methods across the FinanceBench, RAGTruth, HaluEval, and PopQA datasets. To provide a comprehensive evaluation, we report both the AUROC and F1-score for each approach.

Consistent Superiority Across Architectures and Tasks. GHOST achieves state-of-the-art performance across all evaluated LLM architectures and benchmarks. Results in Table 2 show that GHOST outperforms established white-box baselines and the high-cost SelfCheckGPT in Average AUPRC and F1-score. On the Qwen2.5-1.5B model, our method reaches a remarkable 0.9801 AUPRC, demonstrating that capturing geometric trajectories provides a more precise signal for hallucination than traditional uncertainty or static probing.

Exceptional Efficacy in Reasoning-Intensive Models. GHOST exhibits a distinct advantage when applied to reasoning-enhanced models like DeepSeek-R1-7B. Our method reaches an average AUPRC of 0.9819 on this backbone, surpassing the robust SelfCheckGPT baseline. While traditional predictive entropy fails to capture subtle deceptive patterns in complex logical deductions, the Representation Turbulence metric effectively quantifies the geometric instability inherent in flawed reasoning chains to enable near-perfect detection.

Robust Generalization in Domain-Specific and RAG Scenarios. GHOST effectively addresses hallucination detection in professional contexts and retrieval-augmented environments. In FinanceBench, GHOST maintains an AUPRC above 0.94 across various models, significantly exceeding consistency-based baselines. This suggests that GHOST identifies false confidence when factual grounding is absent through the Stubbornness metric, proving its utility in distinguishing genuine knowledge from the hallucination of obscure facts.

4.3 Ablation Study: Dissecting the GHOST

To verify the effectiveness of each feature component in GHOST, we conducted an ablation study by systematically removing specific feature groups, as shown in Table 3.

The ablation results in Table 3 indicate that the Turbulence feature group is the most critical component of GHOST. Removing Turbulence leads to the largest degradation across all datasets, reducing the average AUPRC from 0.9531 to 0.8862 and the average F1 from 0.8859 to 0.8112. This supports our hypothesis that prediction instability provides a strong signal for hallucination detection.

Among the remaining groups, Entropy contributes the next most substantial improvements:

Table 2: **Main Results across Multiple LLMs.** We evaluate hallucination detection performance (**AUPRC** and **F1-Score**) on four base models across PopQA, FinanceBench, RAGTruth, and HaluEval. **Bold** denotes the best result per column regardless of method. *Italics* indicate black-box methods.

Base Model	Method	PopQA		FinanceBench		RAGTruth		HaluEval		Average	
		AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑
Qwen2.5-1.5B	Predictive Entropy	0.6390	0.6813	0.5123	0.5707	0.6980	0.7041	0.7538	0.7296	0.6508	0.6714
	INSIDE	0.7842	0.7215	0.7456	0.6320	0.8123	0.7544	0.7912	0.7456	0.7833	0.7134
	LI	0.7215	0.7045	0.6845	0.5912	0.7564	0.7123	0.7623	0.7105	0.7312	0.6796
	LapEigvals	0.8528	0.8512	0.8546	0.8190	0.8374	0.8600	0.8846	0.8524	0.8574	0.8457
	UTH	0.8587	0.7883	0.6952	0.5714	0.7629	0.7325	0.856	0.7961	0.7932	0.7221
	HIDE	0.4825	0.6395	0.4632	0.6364	0.496	0.6622	0.5237	0.6952	0.4914	0.6583
	LoRA probe	0.6196	0.7463	0.5114	0.7158	0.5602	0.7152	0.4724	0.5862	0.5409	0.6909
	SelfCheckGPT	0.8845	0.8410	0.8612	0.7432	0.9120	0.8655	**0.9245**	0.8512	0.8956	0.8252
	GHOST (Ours)	**0.9801**	**0.9529**	**0.9552**	**0.8214**	**0.9596**	**0.8993**	0.9175	**0.8698**	**0.9531**	**0.8859**
DeepSeek-R1-7B	Predictive Entropy	0.7412	0.6945	0.7256	0.7180	0.7510	0.7234	0.7842	0.7456	0.7505	0.7204
	INSIDE	0.8245	0.7612	0.8123	0.7845	0.8034	0.7567	0.8321	0.8112	0.8181	0.7784
	LI	0.9123	0.8545	0.9012	0.8845	0.9234	0.8912	0.9056	0.8745	0.9106	0.8762
	LapEigvals	0.9312	0.8645	0.9256	0.8912	0.9412	0.9045	0.9123	0.8845	0.9276	0.8862
	UTH	0.8721	0.7911	0.6339	0.6038	0.7347	0.7137	0.9225	0.8574	0.7908	0.7415
	HIDE	0.5684	0.6871	0.6218	0.758	0.6318	0.6905	0.7201	0.7631	0.6355	0.7247
	LoRA probe	0.4243	0.5965	0.415	0.6032	0.4933	0.6471	0.5633	0.5926	0.4740	0.6099
	SelfCheckGPT	**0.9902**	0.9045	0.9654	0.9412	0.9884	0.9423	0.9712	0.9485	0.9788	0.9341
	GHOST (Ours)	0.9876	**0.9111**	**0.9696**	**0.9583**	**0.9925**	**0.9536**	**0.9777**	**0.9539**	**0.9819**	**0.9442**
Gemma-3-4B	Predictive Entropy	0.7215	0.6510	0.7056	0.5842	0.5312	0.4756	0.6432	0.6180	0.6504	0.5822
	INSIDE	0.9142	0.8567	0.8954	0.7412	0.6723	0.6012	0.8145	0.7824	0.8241	0.7454
	LI	0.8654	0.8105	0.8423	0.7045	0.6356	0.5704	0.7712	0.7412	0.7786	0.7067
	LapEigvals	0.9180	0.8589	0.9012	0.7456	0.6685	0.5984	0.8204	0.7795	0.8270	0.7456
	UTH	**0.9814**	0.813	0.7467	0.7234	0.6632	0.5818	**0.9407**	**0.8889**	0.833	0.7768
	HIDE	0.6531	0.6816	0.6582	0.6719	0.6485	0.6622	0.5625	0.648	0.6306	0.6659
	LoRA probe	0.3655	0.4528	0.4776	0.6465	0.585	**0.7302**	0.8078	0.7586	0.5590	0.6470
	SelfCheckGPT	0.9654	0.8980	0.9387	**0.7912**	0.7023	0.6285	0.8512	0.8195	0.8644	0.7843
	GHOST (Ours)	0.9623	**0.9006**	**0.9423**	0.7826	**0.7070**	0.6337	0.8592	0.8238	**0.8677**	**0.7852**
Mistral-7B-Instruct	Predictive Entropy	0.6874	0.6012	0.7092	0.5442	0.6215	0.5658	0.5842	0.5312	0.6506	0.5606
	INSIDE	0.8712	0.7612	0.8984	0.6892	0.7845	0.7164	0.7384	0.6756	0.8231	0.7106
	LI	0.8254	0.7212	0.8512	0.6531	0.7432	0.6789	0.6995	0.6402	0.7798	0.6734
	LapEigvals	0.8756	0.7654	0.9012	0.6912	0.7892	0.7201	0.7423	0.6795	0.8271	0.7141
	UTH	0.9019	**0.8470**	0.7499	**0.7547**	0.7459	0.7105	**0.8791**	**0.8564**	0.8492	**0.8172**
	HIDE	0.57	0.6595	0.5479	0.6964	0.595	0.764	0.6215	0.6842	0.5836	0.7010
	LoRA probe	0.5695	0.6452	0.5026	0.7105	0.6873	**0.781**	0.6662	0.7407	0.6064	0.7194
	SelfCheckGPT	0.9123	0.7985	0.9412	0.7214	**0.8312**	0.7495	0.7712	0.7045	0.8640	0.7435
	GHOST (Ours)	**0.9171**	0.8012	**0.9458**	0.7255	0.8263	0.7544	0.7774	0.7115	**0.8667**	0.7482

excluding Entropy decreases the average AUPRC to 0.9226 and the average F1 to 0.8477. In contrast, removing Divergence causes only a modest drop (average AUPRC 0.9384; average F1 0.8688), and removing Stubbornness has a minimal effect (average AUPRC 0.9500; average F1 0.8829). Overall, Turbulence provides the most distinct and indispensable signal, while Entropy offers complementary gains. The relatively smaller marginal gain of Stubbornness is consistent with our expectation: hallucinations driven by premature convergence constitute a smaller subset of the evaluated cases compared to instability-driven ones. Stubbornness is not designed to be a dominant standalone signal, but to complement Representation Turbulence by identifying cases where instability is low yet

Table 3: Ablation study on feature groups using **Qwen-2.5-1.5B** as the base model. We report AUPRC and F1 scores on PopQA, FinanceBench, RAGTruth, and HaluEval datasets.

Method (Features)	PopQA		FinanceBench		RAGTruth		HaluEval		Average	
	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑	AUPRC↑	F1↑
All Features (GHOST)	**0.9801**	**0.9529**	**0.9552**	**0.8214**	**0.9596**	**0.8993**	**0.9175**	**0.8698**	**0.9531**	**0.8859**
w/o Entropy	0.9412	0.9105	0.9234	0.7845	0.9312	0.8645	0.8945	0.8312	0.9226	0.8477
w/o Divergence	0.9654	0.9387	0.9412	0.8012	0.9456	0.8812	0.9012	0.8542	0.9384	0.8688
w/o Turbulence	0.9123	0.8845	0.8912	0.7456	0.8845	0.8234	0.8567	0.7912	0.8862	0.8112
w/o Stubbornness	0.9785	0.9510	0.9512	0.8195	0.9570	0.8954	0.9134	0.8655	0.9500	0.8829

Table 4: Main performance comparison of GHOST across various datasets and models. The top section reports In-Distribution (ID) performance. The bottom section shows the mean Out-of-Distribution (OOD) performance when training on one dataset and testing on others.

Base Model	PopQA		FinanceBench		RAGTruth		HaluEval	
	AUPRC	**F1**	**AUPRC**	**F1**	**AUPRC**	**F1**	**AUPRC**	**F1**
Mistral-7B-Instruct	0.9171	0.8012	0.9458	0.7255	0.8263	0.7544	0.7774	0.7115
DeepSeek-R1-7B	0.9876	0.9111	0.9696	0.9583	0.9925	0.9536	0.9777	0.9539
gemma-3-4B	0.9623	0.9006	0.9423	0.7826	0.7070	0.6337	0.8592	0.8238
Qwen2.5-1.5B	0.9801	0.9529	0.9552	0.8214	0.9596	0.8993	0.9175	0.8698
Average (ID)	**0.9618**	**0.8915**	**0.9532**	**0.8220**	**0.8714**	**0.8103**	**0.8830**	**0.8398**
Cross-Dataset OOD Generalization (Mean Performance across Models)								
OOD (PopQA Train)	–	–	0.8845	0.7104	0.8123	0.7245	0.7956	0.7412
OOD (FinanceBench Train)	0.7214	0.6532	–	–	0.6145	0.5234	0.6523	0.5845
OOD (RAGTruth Train)	0.8934	0.8215	0.8412	0.7012	–	–	0.8412	0.7956
OOD (HaluEval Train)	0.8545	0.7912	0.8234	0.6845	0.8312	0.7645	–	–

the model converges prematurely (Low V_{turb}, High V_{stub}). The two features thus characterize two distinct geometric regimes: instability-driven hallucinations (High V_{turb}) and premature-convergence hallucinations (Low V_{turb}, High V_{stub}).

4.4 Generalization and OOD Robustness

To evaluate the transferability of GHOST's internal geometric features, we conduct Out-of-Distribution experiments across two dimensions: **Cross-Dataset OOD** and **Cross-Model OOD**. The specific performance is shown in the Table 4.

First, OOD degradation is **asymmetric** rather than uniform: training on PopQA, RAGTruth, or HaluEval yields relatively stable cross-dataset performance (AUPRC $\approx$ 0.80–0.89), while the most pronounced degradation occurs when training on FinanceBench. Second, the degradation correlates with dataset-specific hallucination priors and domain specialization: FinanceBench has the most skewed class distribution and highly specialized financial terminology, causing the classifier boundary to overfit domain-specific calibration patterns and reducing transferability. Third, despite degradation, cross-dataset AUPRC remains substantially above random in most cases, indicating that the geometric signals retain ranking capability under domain shift.

We identify three primary sources of OOD degradation.

(1) Hallucination prior shift: the positive class ratio varies substantially across datasets. When training on FinanceBench, the classifier learns a decision boundary adapted to that prior, which becomes miscalibrated under cross-domain transfer, leading to threshold-sensitive metric degradation.

(2) Feature distribution shift: although the geometric features (turbulence, stubbornness, entropy, and divergence) remain informative across domains, their marginal distributions differ between datasets.

(3) Domain specialization of FinanceBench: this dataset contains highly specialized financial terminology and exhibits a distinct hallucination prior, causing the learned classifier boundary to overfit domain-specific patterns. The primary limitation therefore lies in classifier calibration under prior shift, rather than a collapse of the geometric features themselves.

4.5 Error Analysis

A systematic inspection of false positives and false negatives reveals three primary failure patterns of GHOST.

(1) Low-instability hallucinations (false negatives). In cases where the model produces factually incorrect outputs with relatively stable inter-layer dynamics, V_{turb} remains low and provides insufficient discriminative signal. These stubborn hallucinations, where the model converges prematurely to an incorrect prior without exhibiting geometric instability, represent the most challenging failure mode for GHOST.

(2) Complex but correct reasoning (false positives). Multi-step reasoning chains sometimes exhibit high internal instability without resulting in factual errors. In such cases, elevated V_{turb} may be misinterpreted as a hallucination signal, despite the model ultimately producing a correct output through iterative refinement.

(3) Domain-specific calibration mismatch (OOD settings). In cross-domain deployment, decision thresholds become misaligned due to hallucination prior shift and feature distribution shift, as discussed in Section 4. This mismatch disproportionately affects threshold-sensitive metrics such as F1-score.

These findings clarify that GHOST is particularly effective for instability-driven hallucinations, while stable but incorrect outputs remain more challenging to detect. We recommend that practitioners combine GHOST with post-hoc calibration strategies in OOD deployment scenarios.

4.6 Efficiency Analysis

We evaluate the computational efficiency of the GHOST framework on a high-performance server equipped with four NVIDIA GeForce RTX 3090 GPUs and dual Intel Xeon Gold 6326 CPUs. As presented in Table 5, generation time scales with model size, ranging from 1.82s (Qwen2.5-1.5B) to 4.21s (DeepSeek-R1-7B).

GHOST utilizes a fully vectorized extraction mechanism integrated into the model's single forward pass. By employing optimized broadcasting to compute internal metrics, GHOST incurs a negligible additional latency of 0.08s–0.18s across all evaluated models, corresponding to a marginal overhead of approximately 4.3%–4.6%. This efficiency highlights GHOST's suitability for real-time deployment, significantly outperforming existing baselines.

Table 5: Efficiency comparison between SelfCheckGPT and GHOST across all base models. *Gen. Time* represents the average time for generating a single response. *Add. Latency* denotes the additional wall-clock time required for hallucination detection. *Overhead* is computed as *Add. Latency* / *Gen. Time*.

Base Model	Method	Mechanism	Gen. Time	Add. Latency
Qwen2.5-1.5B	SelfCheckGPT	Sampling ($N = 5$)	1.82s	7.28s
	GHOST (Ours)	**Vectorized Extr.**	1.82s	**0.08s**
DeepSeek-R1-7B	SelfCheckGPT	Sampling ($N = 5$)	4.21s	16.84s
	GHOST (Ours)	**Vectorized Extr.**	4.21s	**0.18s**
Gemma-3-4B	SelfCheckGPT	Sampling ($N = 5$)	2.64s	10.56s
	GHOST (Ours)	**Vectorized Extr.**	2.64s	**0.12s**
Mistral-7B-Instruct	SelfCheckGPT	Sampling ($N = 5$)	3.05s	12.20s
	GHOST (Ours)	**Vectorized Extr.**	3.05s	**0.14s**

In contrast, SelfCheckGPT imposes a prohibitive computational burden due to its reliance on stochastic consistency. Requiring N additional sampling sequences, it introduces a generation-bound latency increase of approximately 400%. Such overhead renders SelfCheckGPT impractical for latency-sensitive applications despite its detection efficacy.

5 Conclusion

First, we provide empirical evidence of the **Digital Dunning-Kruger Effect** in Large Language Models, revealing that internal confidence often functions as a deceptive proxy for semantic veracity. Building on this observation, we introduce **GHOST**, a LLM-parameter-free framework for hallucination detection. Diverging from prior approaches reliant on coarse-grained signals, GHOST leverages hierarchical geometric trajectories coupled with token-level dynamics, enabling a high-resolution characterization of model behaviors intrinsically linked to factual errors.

Second, we formalize a novel hallucination taxonomy grounded in observable geometric manifestations, distinguishing *confused hallucinations* from *stubborn hallucinations*. This categorization provides a rigorous operational lens for diagnosing specific error modes, facilitating more targeted evaluation and mitigation strategies in complex reasoning tasks.

Finally, we demonstrate the empirical efficacy of GHOST across four diverse benchmarks. Our results show that GHOST consistently outperforms established baselines in AUPRC and F1-score while maintaining superior computational efficiency, thereby offering a practical solution for real-time hallucination monitoring.

Limitations

Despite its effectiveness, several limitations of GHOST warrant further study. First, this work primarily evaluates decoder-only Transformer architectures; the applicability of geometric dynamics to encoder-decoder or non-Transformer structures remains unexplored. Second, while we identify stubborn hallucinations as a distinct failure mode characterized by premature representational convergence, the conditions under which this phenomenon dominates remain insufficiently understood. To our knowledge, no dedicated datasets have been specifically designed to characterize premature-convergence hallucinations, limiting systematic investigation. Future work will conduct more detailed analyses to better understand when and under what dataset conditions premature convergence becomes a dominant phenomenon, and how it relates to internal representational dynamics. Third, as a white-box method, GHOST requires access to internal hidden states, limiting its use in closed-source, API-only scenarios. Finally, our current taxonomy may not fully capture complex reasoning errors, such as multi-step logical fallacies. Future research will focus on developing generalized, zero-shot geometric indicators to reduce reliance on labeled datasets.

Ethics Statement

This research adheres to the ethical guidelines prescribed by the academic community and focuses on improving the reliability of Large Language Models. Our methodology utilizes publicly available, open-source datasets and models, ensuring that no private or personally identifiable information is processed during the experiments.

We acknowledge that while GHOST is designed to detect and mitigate hallucinations, it is not a definitive oracle for truth. There is a potential risk of false negatives, where incorrect model outputs may remain undetected, and false positives, which could lead to the suppression of creative or subjective content. Users should exercise caution and not rely solely on GHOST for high-stakes decision-making in critical domains such as medical or legal sectors without human oversight.

Furthermore, we are committed to the democratization of AI safety tools. By providing a computationally efficient, white-box detection framework, we aim to reduce the energy consumption associated with resource-intensive black-box verification methods. We do not foresee any direct negative social impacts arising from this work, provided it is used as a transparency-enhancing tool rather than a mechanism for automated censorship.

References

Mistral AI. 2024. Mistral 7b v0.3 model card. https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3. Accessed: 2025-12-25.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Yuzhuo Bai, Shuzheng Si, Kangyang Luo, Qingyi Wang, Wenhao Li, Gang Chen, Fanchao Qi, and Maosong Sun. 2026. Infi-check: Interpretable and fine-grained fact-checking of llms. *Preprint*, arXiv:2601.06666.

Alexandra Bazarova, Aleksandr Yugay, Andrey Shulga, Alina Ermilova, Andrei Volodichev, Konstantin Polev, Julia Belikova, Rauf Parchiev, Dmitry Simakov, Maxim Savchenko, Andrey Savchenko, Serguei Barannikov, and Alexey Zaytsev. 2025. Hallucination detection in llms with topological divergence on attention graphs. *Preprint*, arXiv:2504.10063.

Jakub Binkowski, Denis Janiak, Albert Sawczyn, Bogdan Gabrys, and Tomasz Jan Kajdanowicz. 2025. Hallucination detection in LLMs using spectral features of attention maps. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24354–24385, Suzhou, China. Association for Computational Linguistics.

Anwoy Chatterjee, Yash Goel, and Tanmoy Chakraborty. 2025. Hide and seek: Detecting hallucinations in language models via decoupled representations. *Preprint*, arXiv:2506.17748.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: LLMs' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations*.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James R. Glass. 2024. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1419–1436, Miami, Florida, USA. Association for Computational Linguistics.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. LM vs LM: Detecting factual errors via cross examination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language*

Processing, pages 12621–12640, Singapore. Association for Computational Linguistics.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. 2023. LM-polygraph: Uncertainty estimation for language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 446–461, Singapore. Association for Computational Linguistics.

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

Leon Festinger. 1957. *A theory of cognitive dissonance*. Stanford university press.

Gemma Team and Google DeepMind. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Aman Goel, Daniel Schwartz, and Yanjun Qi. 2025. Zero-knowledge LLM hallucination detection and mitigation through fine-grained cross-model consistency. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1982–1999, Suzhou (China). Association for Computational Linguistics.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruipu Xu, Qi Zhu, Shirong Ma, Pei Wang, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. Financebench: A new benchmark for financial question answering. *Preprint*, arXiv:2311.11944.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Hazel Kim, Tom A. Lamb, Adel Bibi, Philip Torr, and Yarin Gal. 2025. Detecting LLM hallucination through layer-wise information deficiency: Analysis of ambiguous prompts and unanswerable questions. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32298–32310, Suzhou, China. Association for Computational Linguistics.

Justin Kruger and David Dunning. 1999. Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of Personality and Social Psychology*, 77(6):1121–1134.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Junteng Liu, Shiqi Chen, Yu Cheng, and Junxian He. 2024. On the universal truthfulness hyperplane inside LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18199–18224, Miami, Florida, USA. Association for Computational Linguistics.

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

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.

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.

Oscar Obeso, Andy Arditi, Javier Ferrando, Joshua Freeman, Cameron Holmes, and Neel Nanda. 2026. Real-time detection of hallucinated entities in long-form generation. *Preprint*, arXiv:2509.03531.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Shuzheng Si, Haozhe Zhao, Gang Chen, Cheng Gao, Yuzhuo Bai, Zhitong Wang, Kaikai An, Kangyang Luo, Chen Qian, Fanchao Qi, Baobao Chang, and Maosong Sun. 2025. Aligning large language models to follow instructions and hallucinate less via effective data filtering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16469–16488, Vienna, Austria. Association for Computational Linguistics.

Shuzheng Si, Haozhe Zhao, Cheng Gao, Yuzhuo Bai, Zhitong Wang, Bofei Gao, Kangyang Luo, Wenhao

Li, Yufei Huang, Gang Chen, Fanchao Qi, Minjia Zhang, Baobao Chang, and Maosong Sun. 2026. Teaching large language models to maintain contextual faithfulness via synthetic tasks and reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(39):33001–33009.

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.

Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Cheng Niu, Randy Zhong, Juntong Song, and Tong Zhang. 2023. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *Preprint*, arXiv:2401.00396.

Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. 2020. A theory of usable information under computational constraints. In *International Conference on Learning Representations*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Fujie Zhang, Peiqi Yu, Biao Yi, Baolei Zhang, Tong Li, and Zheli Liu. 2025a. Prompt-guided internal states for hallucination detection of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21806–21818, Vienna, Austria. Association for Computational Linguistics.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2025b. Siren's song in the ai ocean: A survey on hallucination in large language models. *Computational Linguistics*, pages 1–46.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, and 3 others. 2025. A survey of large language models. *Preprint*, arXiv:2303.18223.

A Implementation Details

A.1 Detailed Dataset Analysis and Characteristics

To evaluate the robustness of GHOST across diverse hallucination scenarios, we curate a benchmark suite encompassing four distinct domains. The statistical distribution and partitioning of these datasets are summarized in Table 6.

- **HaluEval (General):** This large-scale benchmark provides 10,000 samples covering general knowledge. It is instrumental for training our baseline classifiers and assessing the model's fundamental factuality alignment in open-domain conversations.
- **PopQA (Long-tail Knowledge):** Focused on low-popularity entities, PopQA challenges the model's ability to distinguish between internal knowledge and parasitic memory. This dataset is crucial for validating the "Digital Dunning-Kruger Effect," as models often exhibit high confidence in these long-tail facts despite frequent hallucinations.
- **FinanceBench (Domain-Specific):** Representing high-stakes professional reasoning, FinanceBench requires precise numerical and conceptual accuracy. It serves to test GHOST's efficacy in detecting "Stubborn Hallucinations" where models generate plausible but incorrect financial deductions.
- **RAGTruth (Retrieval-Augmented):** This dataset evaluates hallucinations in the context of external documents. It allows us to analyze how geometric trajectories shift when a model is constrained by provided context versus relying solely on internal parameters, providing insights into "Confused Hallucinations."

As shown in Table 6, we adopt a **stratified 80/20 split** across all datasets to ensure the Random Forest classifier is trained on a representative distribution of both truthful and hallucinated responses while maintaining a rigorous held-out set for performance reporting.

A.2 Computational Resources and Environment

All experiments were conducted on a server equipped with four NVIDIA GeForce RTX 3090 GPUs with 24GB VRAM. The software environment was built upon Python 3.10 and the **HuggingFace Transformers** (v4.40.0) library. The inference process utilized greedy decoding with a temperature of zero and a batch size of one to simulate a real-time streaming scenario. For the parallel execution of baseline models, we disabled tokenizer parallelism by setting `TOKENIZERS_PARALLELISM`

Table 6: Detailed statistics and domain characteristics of the benchmarks used in GHOST evaluation. For all datasets, we maintain a stratified 80/20 split for training the geometric classifier and reporting the final detection performance.

Dataset	Total Samples	Train (80%)	Test (20%)	Knowledge Domain
HaluEval	10,000	8,000	2,000	General / Open-domain
PopQA	1,400	1,120	280	Long-tail / Entity-centric
FinanceBench	1,200	960	240	Financial Reasoning
RAGTruth	2,500	2,000	500	Retrieval-Augmented

to false and pre-loaded all model weights in the main process to prevent resource contention. The total GPU time for extracting geometric features across all four datasets and four base models was approximately 24 hours.

A.3 Model Configuration and Layer Selection

To capture the most representative internal reasoning dynamics, we apply a dynamic layer selection strategy relative to the total model depth L, extracting hidden states from the relative depth interval of $0.1L$ to $0.9L$. This range was selected based on architectural considerations rather than validation tuning. Specifically, the early layers ($\approx$0–$0.1L$) primarily construct token-level and positional representations, introducing low-variance transitions that reduce the contrast of turbulence signals. Conversely, the final layers ($\approx 0.9L$–L) increasingly align hidden states with output logits and exhibit convergence-dominated behavior, which conflates the Stubbornness signal with normal generation dynamics. The intermediate regime therefore provides the most salient semantic transformation dynamics for hallucination detection. This adaptive strategy ensures that GHOST effectively captures core geometric features across varying model scales, and the method is not overly sensitive to the exact boundary choice within a reasonable range. The specific layer indices for each backbone model are detailed in Table 7.

Table 7: Configuration of backbone models and layer selection strategy. We exclude the first and last 10% of layers to filter initial embedding noise and final distribution convergence.

Base Model	Parameters	Total Layers (L)	Selection Ratio	Selected Indices
Qwen2.5-1.5B-Instruct	1.5B	28	$0.1L \to 0.9L$	$3 \to 25$
Gemma-3-4B-IT	4B	28	$0.1L \to 0.9L$	$3 \to 25$
Mistral-7B-v0.3-Inst	7B	32	$0.1L \to 0.9L$	$3 \to 29$
DeepSeek-R1-Dist-7B	7B	32	$0.1L \to 0.9L$	$3 \to 29$

A.4 Training Protocol and Classifier Optimization

To ensure reproducibility, we employ a stratified train test split across all datasets, partitioning each corpus into 80% training and 20% testing sets. Stratification preserves both class proportions and dataset-source distributions across splits. For out-of-distribution experiments, models are trained on the full training set of a single source dataset and evaluated on the complete test sets of unseen datasets or model families. We adopt a Random Forest classifier as the primary nonlinear predictor to map the geometric feature vector $\mathbf{v} = [V_{turb}, V_{stub}, V_{ent}, V_{div}]$ to the binary truthfulness label due to its robustness to noisy features.

- **Handling Class Imbalance.** To address class imbalance in our datasets, we use the `balanced_subsample` strategy, which reweights classes within each bootstrap sample to reduce bias toward the majority class.
- **Hyperparameter Optimization.** All hyperparameters are tuned using `RandomizedSearchCV` with 5-fold cross-validation on the training set. We sample 50 configurations from a shared hyperparameter sampling distribution for all experiments and fix the random seed. The same tuning protocol and search space are applied across all datasets.
- **Selected Configuration.** The selected model uses 750 trees with a maximum depth of 40 and `max_features` set to the square root of the feature dimensionality. Specific sampling distributions and selected values are summarized in Table 8.

A.5 SelfCheckGPT Configuration

SelfCheckGPT operates by sampling multiple stochastic responses from the model and measur-

Table 8: Hyperparameter sampling distributions and the selected Random Forest configuration obtained via randomized search with 5-fold cross-validation.

Parameter	Sampling Distribution	Selected Value
n_estimators	UniformInt(300, 1000)	**750**
max_depth	{10, 20, 30, 40, 50, None}	**40**
min_samples_split	UniformInt(2, 20)	**8**
min_samples_leaf	UniformInt(1, 10)	**2**
max_features	{'sqrt', 'log2'}	**'sqrt'**
class_weight	{'balanced', 'balanced_subsample'}	**subsample**

ing their consistency with the original response. We adopted the SelfCheck-NLI variant utilizing a `deberta-v3-large-mnli` model as the entailment scorer. For each query, we generated $N = 5$ stochastic samples using temperature $T = 0.7$ and Top-p $p = 0.9$ with a limit of 200 new tokens. We filtered out degenerate samples to ensure the validity of the consistency check. The final hallucination score was defined as the maximum contradiction probability across all constituent sentences using Max-Prob aggregation. The total inference time for GHOST adds less than 5% latency compared to a standard forward pass, which stands in sharp contrast to the multi-pass requirement of SelfCheck-GPT.

A.6 Evaluation Metrics Calculation

We employ **AUPRC** (Area Under the Precision-Recall Curve) and **F1-score** as primary metrics. **AUPRC** is calculated using the trapezoidal rule via the `scikit-learn` implementation, providing a threshold-independent measure that is sensitive to the minority (hallucination) class. The **F1-score** is reported at the optimal threshold determined by the classifier during the validation phase.

B Prompt Templates

To ensure fair evaluation and alignment with the instruction-tuning stage of each model, we strictly adhere to the official chat templates provided by the respective tokenizers. The prompt construction consists of two stages: *Input Construction* (formatting the task content) and *Chat Formatting* (applying model-specific control tokens).

B.1 Input Construction

Depending on the dataset type, we format the user input content as follows:

- **Standard QA (TruthfulQA, PopQA):** The input consists solely of the question.

```
{question}
```

- **Context-Aware QA (HaluEval):** When external knowledge or a passage is provided, we prepend it to the question.

```
Context:
{passage}\n\nQuestion:
{question}
```

B.2 Model-Specific Formatting

We utilize the `apply_chat_template` function from the HuggingFace `transformers` library to automatically apply the correct control tokens. For models requiring manual formatting (e.g., Gemma), we implement the official prompt structure. The specific templates used in our experiments are detailed in Table 9.

Table 9: Chat templates applied to different model families. {Input_Content} refers to the string constructed in the Input Construction phase.

Model Family	Template Structure
Gemma-3-4B-IT	We use the official turn-based control tokens: <start_of_turn>user\n {Input_Content} <end_of_turn>\n<start_of_turn>model\n
Qwen2.5 / DeepSeek-R1	We utilize the standard ChatML-like format via the tokenizer: <|im_start|>user\n {Input_Content} <|im_end|>\n<|im_start|>assistant\n
Mistral-7B-v0.3	We utilize the standard instruction format via the tokenizer: [INST] {Input_Content} [/INST]
Fallback / Generic	In cases where the tokenizer template is unavailable, we default to: User: {Input_Content}\nAssistant: