From Complexity to Clarity: A Systematic Study of Geometric Dynamics in Hidden Representations of LLMs

Anonymous ACL submission

Abstract

Large language models perform inference through hidden representations that evolve across layers and generated tokens, forming complex trajectories in hidden representation space. We study the geometric dynamics of these trajectories for label-free model self-evaluation problem: does the hidden-state tensor already contain correctness signals, and where do those signals reside? Our analysis is organized along three intrinsic dimensions of inference-time computation: layer depth, token span, and representation space. Across seven benchmarks spanning mathematical reasoning, commonsense reasoning, and multilingual understanding, and two instruction-tuned LLMs, we find that correctness separability is highly localized, consistently concentrating in late-layer representations and limited token regions. Motivated by this, we propose **Norm Drop Rate (NDR)**, a minimal score based on final-layer norm contraction. Despite using no labels, learned probes, or logit space transformation, NDR achieves competitive AUROC and the best AUPR among all methods, while remaining the simplest in design and most robust when controlling for response length or changing representation bases. These results suggest that hidden-state trajectory analysis should focus less on increasingly complex summaries and more on identifying simple, localized signals.

1 Introduction

Large language models (LLMs) are increasingly deployed in settings where users rely on a single generated response to be correct (Achiam et al., 2023; Touvron et al., 2023; Chi et al., 2025). Yet fluent generations can be unsupported or wrong, making failures hard to detect. A practical remedy is single-pass model correctness evaluation (Kadavath et al., 2022; Manakul et al., 2023; Kuhn et al., 2023): estimating correctness using only inference-time signals, without training a separate verifier or sampling many alternative responses.

Geometric signal in hidden representation. For each token position and each layer, the autoregressive transformer updates a hidden representation that supports the next-token prediction (Vaswani et al., 2017). For a fixed generation, the hidden states indexed by layer depth and generated tokens form a structured tensor that can be viewed as trajectories in hidden representation space. This suggests a training-free approach to self-evaluation: correctness may be reflected not only in what the model outputs, but also in how internal representations move through space during inference.

From composite heuristics to systematic diagnosis. Recent work has shown that intermediate representations expose meaningful stage-wise computation (Belrose et al., 2023; Halawi et al., 2024; Shai et al., 2024), and Chain-of-Embeddings (CoE) aggregates geometric statistics over token-averaged layer trajectories for output-free self-evaluation (Wang et al., 2024). However, geometry-based scores are often composite heuristics: it remains unclear which quantities carry signal, where that signal localizes, and how robust it is to common confounds (e.g., generation length). We address this by treating the hidden-state tensor as the central object of analysis and decomposing it along three intrinsic dimensions of inference-time computation. The **layer dimension** asks where in layer depth a label-free scalar score separates correct from incorrect generations. The **token dimension** asks which response regions drive the signal and whether it persists after controlling for response length. The **representation dimension** asks whether geometric signals remain stable when computed under different representation bases. Throughout, labels are used only for evaluation; every score is computed from a single forward generation.

Complex trajectories, localized signals. Across tasks and models, we find that discriminative power in hidden-state trajectories is highly localized, con-

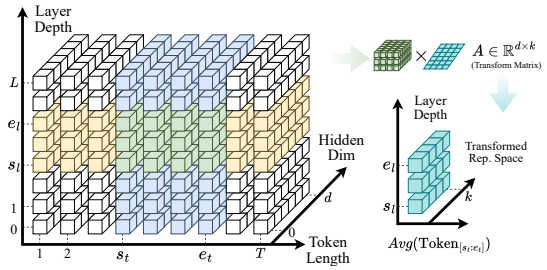

Figure 1: **Three-dimension diagnostic framework.** From one prompt and one generated response, we log hidden states $\mathcal{H}$ over generated tokens and layers. A training-free score is computed from a selected sub-tensor defined by three components: Token Span ($\text{Token}[s_t, e_t]$), Layer Window ($\text{Layer}[s_l, e_l]$), and Representation Space.

centrating in late layer representations and partial token regions. Motivated by this, we propose **Norm Drop Rate (NDR)**, an extremely simple unsupervised metric based on final-layer norm contraction that matches or exceeds more complex signals across benchmarks.

Contributions. Our contributions are as follows: (i) a systematic diagnostic framework that decomposes hidden-state correctness signals along three dimensions: layer depth, token span, and representation space; (ii) an empirical localization result showing that trajectory-based methods concentrate their useful signal in late layers and hallucination token spans; (iii) NDR, a minimal label-free score distilled from layerwise norm dynamics that is robust under length matching and less dependent on fragile subspace selection than other metrics.

2 Related Work

2.1 Hidden-State Geometry

A growing body of work studies LLM hidden states through a geometric lens, treating activations as points, directions, subspaces, or trajectories in hidden representation space. One line identifies linear directions that encode truthfulness, latent knowledge, or hallucination signals (Marks and Tegmark, 2023; Burns et al., 2022; Park et al., 2025; O'Neill et al., 2025) and uses such directions for steering and intervention (Zou et al., 2023). Belief-state analyses further show that residual streams can encode the geometry of latent predictive states (Shai et al., 2024). Closest to our setting, trajectory-based methods treat the evolution of hidden states across layers as a discrete curve and derive training-free correctness scores from angle, displacement, or curvature statistics (Wang et al., 2024; Manson, 2025). These works demonstrate that hidden-state geometry captures meaningful signals about semantics, belief, and reliability, but most anchor their analysis to a fixed layer range, token span, or basis. Our work systematically decomposes where such signals arise across three dimensions of the hidden-state tensor.

2.2 Layer-Wise Signals in Hidden Representations

Different layers of a transformer have been found to serve different computational roles. Logit-lens and tuned-lens analyses show that answer content becomes progressively decodable from intermediate residual states (nostalgebraist, 2020; Belrose et al., 2023), and activation patching localizes factual recall to specific middle layers (Meng et al., 2022; Zhang and Nanda, 2024). Recent studies further suggest that different depth ranges support qualitatively different computations: middle layers can act as semantic workspaces where answer selection occurs, while late layers amplify the selected answer in vocabulary space (Wiegreffe et al., 2025); layer-intervention work identifies distinct inference stages from early contextualization through late-stage residual sharpening (Lad et al., 2026); and analysis under misleading demonstrations reveals overthinking phenomena in late layers (Halawi et al., 2024). While these results establish that different depths carry different types of information, geometric analyses have not yet systematically studied how correctness-related signals distribute across layers.

2.3 Label-Free Self-Evaluation

Self-evaluation estimates whether a generated response is correct using only inference-time information (e.g., hidden states). Output-based methods elicit explicit confidence scores, aggregate multiple samples, or estimate semantic uncertainty (Kadavath et al., 2022; Wang et al., 2022; Manakul et al., 2023; Kuhn et al., 2023; Farquhar et al., 2024). Activation-based methods train supervised probes on hidden states to detect truthfulness or hallucination (Azaria and Mitchell, 2023; Liu et al., 2024; Orgad et al., 2025), but require labeled data and offer limited interpretability. Training-free activation-based methods are closest to our setting: CoE derives correctness scores directly from hidden-state trajectories without fitting a detector (Wang et al., 2024). We adopt the same label-free self-evaluation

setting but use it as a diagnostic testbed: given one response and its hidden-state tensor, we ask which simple geometric signals, and which regions of the tensor, carry the most discriminative power.

3 Systematic Analysis Framework for Representation Geometric Dynamics

3.1 Preliminaries

Transformer LM Computation Let f_θ be a decoder-only Transformer with L layers and hidden dimension d. Given a prompt x, the model autoregressively generates a response of T tokens $y_{1:T}$. For each generated token position $t \in \{1, \ldots, T\}$ and layer $l \in \{0, \ldots, L\}$, we record a vector $h_t^l \in \mathbb{R}^d$, where $l = 0$ denotes the token embeddings input to the first Transformer layer and $l = L$ denotes the representations after the last layer. Here h_t^l denotes the residual-stream hidden state after the full l-th Transformer block, including attention, MLP, and residual updates. The final-layer state produces the next-token distribution $p_\theta(\cdot \mid x, y_{1:t}) = \mathrm{softmax}(W_U h_t^L)$, where $W_U \in \mathbb{R}^{|\mathcal{V}| \times d}$ is the output embedding matrix over vocabulary $\mathcal{V}$. The central object of study is the hidden-state tensor

$$\mathcal{H} \in \mathbb{R}^{(L+1) \times T \times d}, \qquad \mathcal{H}_{l,t,:} = h_t^l. \tag{1}$$

Token-mean layer trajectory. Most scores in this paper operate on a layer-indexed trajectory derived from $\mathcal{H}$. Given a token span $\mathcal{T} \subseteq \{1, \ldots, T\}$, we average hidden states over tokens to obtain

$$\bar{h}_{\mathcal{T}}^l = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} h_t^l, \qquad l = 0, \ldots, L. \tag{2}$$

The sequence $(\bar{h}_{\mathcal{T}}^0, \ldots, \bar{h}_{\mathcal{T}}^L)$ is the token-mean layer trajectory.

3.2 Label-Free Model Correctness Evaluation

We study label-free model correctness evaluation: given a prompt and one autoregressive generation, estimate response correctness using only inference-time signals from the hidden-state tensor $\mathcal{H}$, without ground-truth labels. This setting requires no trained verifier, no external tool, and no labeled supervision (Wang et al., 2024; Kadavath et al., 2022; Farquhar et al., 2024).

In this paper, every score $s(\mathcal{H}) \in \mathbb{R}$ is computed from the hidden-state tensor defined above. A good score should separate correct from incorrect generations. Correctness labels are used only after scoring, to measure this separability.

3.3 Three Diagnostic Dimensions

The hidden-state tensor $\mathcal{H}$ naturally suggests three diagnostic dimensions, corresponding to its layer, token, and representation structure (Figure 1). Each dimension isolates a different factor that may affect the correctness score.

Layer dimension. Where along depth does correctness separability concentrate? We fix the token span and sweep layer windows $[s_l, e_l]$ to localize which depth regions carry signal.

Token dimension. Which response regions drive the signal, and does the signal remain after removing the effect of response length? We vary the token span $\mathcal{T}$ across different response regions.

Representation dimension. Do geometric signals remain stable under different representation bases? We recompute scores under alternative linear maps such as PCA subspaces and logit space; the default is native hidden representation space.

All later sections modify one of these three choices while keeping the same scoring setup.

4 Experimental Setup

We evaluate hidden-state scoring on seven benchmarks spanning mathematical reasoning, broad knowledge, commonsense reasoning, and multilingual understanding. For every method, we follow the same procedure: generate one response per prompt with greedy decoding, log the hidden-state tensor $\mathcal{H}$, compute a scalar score without using labels, and evaluate the score against binary correctness labels.

4.1 Benchmarks, Models, and Evaluation

Datasets. GSM8K (Cobbe et al., 2021) contains grade-school arithmetic word problems. MATH (Hendrycks et al., 2021) contains competition-level mathematical problems. TheoremQA (Chen et al., 2023) contains theorem-centric STEM questions. MMLU (Hendrycks et al., 2020) contains multiple-choice questions across 57 subjects. CommonsenseQA (Talmor et al., 2019) contains multiple-choice commonsense questions. Belebele (Bandarkar et al., 2024) is a multilingual reading comprehension benchmark. MGSM (Shi et al., 2022) contains multilingual grade-school math word problems.

Models. We study two frozen instruction-tuned decoder-only LLMs: Qwen2-7B-Instruct (Yang

(a) Qwen2-7B-Instruct (AUROC↑ / FPR95↓ / AUPR↑)

Method	GSM8K	MATH	MMLU	TheoremQA	CSQA	Belebele	MGSM	Avg.
MaxProb$^{\circ}$	65.3 / 77.1 / 88.7	68.7 / 69.5 / 61.9	54.3 / 91.4 / 64.8	61.1 / 88.9 / 33.4	56.8 / 94.5 / 79.8	58.6 / 89.7 / 87.9	73.4 / 69.8 / 91.0	62.6 / 83.0 / 72.5
Entropy$^{\circ}$	67.3 / 72.7 / 89.5	70.0 / 67.4 / 63.3	54.9 / 90.6 / 65.2	61.9 / 87.9 / 33.9	58.3 / 93.9 / 80.8	60.3 / 88.2 / 88.6	75.8 / 55.8 / 92.0	64.1 / 79.5 / 73.3
Perplexity$^{\circ}$	65.5 / 76.7 / 88.8	69.6 / 67.4 / 62.7	54.4 / 91.3 / 64.9	61.8 / 88.7 / 34.0	57.3 / 94.9 / 80.3	58.7 / 89.0 / 87.9	73.8 / 67.4 / 91.1	63.0 / 82.2 / 72.8
CoE-R$^{\diamond}$	67.0 / 83.9 / 90.1	70.9 / 84.9 / 71.2	56.7 / 93.5 / 69.7	67.3 / 90.2 / 47.0	52.2 / 96.1 / 75.9	58.4 / 91.9 / 88.4	75.1 / 86.1 / 93.5	63.9 / 89.5 / 76.5
CoE-C$^{\diamond}$	69.9 / 78.0 / 90.8	73.6 / 71.7 / 70.9	55.2 / 91.1 / 67.2	66.5 / 93.4 / 43.9	56.5 / 92.9 / 78.0	65.7 / 87.5 / 90.8	82.1 / 72.1 / 95.7	67.1 / 83.8 / 76.8
Lateral Ratio*	**70.7** / 78.9 / 90.9	74.1 / 71.2 / 70.3	56.7 / 91.5 / 69.5	**68.9** / 93.7 / **47.7**	56.8 / 91.9 / 77.6	67.1 / 87.5 / 91.4	**83.1** / 65.1 / **95.9**	**68.2** / 82.8 / 77.6
Curvature*	69.8 / **73.1** / 90.6	**75.8** / **67.6** / 73.0	56.2 / **90.5** / 68.4	67.6 / 91.4 / 44.2	54.2 / **91.6** / 75.9	**70.7** / **82.4** / **92.2**	81.0 / **58.1** / 95.2	67.9 / **79.2** / 77.1
NDR*	70.5 / 78.9 / **91.0**	75.2 / 74.8 / **74.2**	**58.6** / **90.5** / **71.1**	68.2 / 94.0 / 47.3	55.2 / 95.2 / 77.6	67.1 / 87.5 / 91.1	81.5 / 62.8 / 95.5	68.0 / 83.4 / **78.3**

(b) Llama-3-8B-Instruct (AUROC↑ / FPR95↓ / AUPR↑)

Method	GSM8K	MATH	MMLU	TheoremQA	CSQA	Belebele	MGSM	Avg.
MaxProb$^{\circ}$	70.9 / 81.4 / 90.6	51.2 / 87.6 / 21.8	52.4 / 93.1 / 66.3	62.6 / 92.5 / 31.1	53.9 / 90.4 / 73.9	57.1 / 95.6 / 91.8	73.7 / 72.0 / 90.0	60.3 / 87.5 / 66.5
Entropy$^{\circ}$	72.2 / 76.5 / 91.2	52.5 / 86.5 / 22.3	52.8 / 92.4 / 66.6	61.9 / 93.0 / 31.0	54.4 / 89.2 / 74.2	57.8 / 93.3 / 92.1	75.8 / 72.0 / 91.0	61.1 / 86.1 / 66.9
Perplexity$^{\circ}$	70.9 / 80.6 / 90.7	51.6 / 88.3 / 21.9	52.8 / 92.9 / 66.6	61.9 / 93.9 / 30.3	54.1 / 90.4 / 73.9	57.3 / 93.3 / 91.9	73.2 / 76.0 / 89.8	60.3 / 87.9 / 66.4
CoE-R$^{\diamond}$	59.8 / 89.1 / 85.2	62.9 / 84.5 / 32.7	56.0 / 91.8 / 69.8	53.3 / 92.7 / 19.1	51.3 / 94.9 / 71.0	54.3 / 90.0 / 90.3	60.3 / 88.0 / 84.7	56.8 / 90.1 / 64.7
CoE-C$^{\diamond}$	70.5 / 81.0 / 90.5	66.6 / 86.2 / 37.7	55.3 / 90.9 / 67.9	52.7 / 95.2 / 21.4	54.2 / 86.8 / 74.7	54.8 / 95.6 / 90.2	70.2 / 80.0 / 90.4	60.6 / 87.9 / 67.5
Lateral Ratio*	71.3 / 82.6 / 90.7	70.4 / 80.1 / 39.8	53.9 / 90.4 / 67.5	54.2 / 93.1 / 21.6	53.0 / 86.2 / 74.1	56.0 / 94.4 / 90.0	71.0 / 82.0 / 90.8	61.4 / 87.0 / 67.8
Curvature*	**73.0** / **76.1** / **91.2**	**73.7** / **72.9** / **42.9**	**58.0** / **88.7** / **70.8**	52.7 / **90.4** / 20.1	53.7 / **85.3** / 73.5	54.5 / 93.3 / 90.6	75.4 / 76.0 / **91.7**	**63.0** / **83.3** / **68.7**
NDR*	72.2 / 75.7 / 90.9	69.2 / 77.8 / 39.0	54.9 / 90.0 / 68.6	**57.6** / **85.0** / 21.9	52.7 / 86.8 / 73.7	**56.1** / 93.3 / 90.2	73.2 / 80.0 / 91.1	62.3 / 84.1 / 67.9

Table 1: **Correctness separability across seven benchmarks.** Each cell reports AUROC↑ / FPR95↓ / AUPR↑ (%). **Bold** = best per column. Method categories: $\circ$ = output-level, $\diamond$ = trajectory baseline, $\star$ = ours.

et al., 2024) and Llama-3-8B-Instruct (AI@Meta, 2024). Unless otherwise stated, visualizations use Qwen2-7B-Instruct. Both models use the same generation, logging, and scoring setup. We use zero-shot chain-of-thought prompting (Kojima et al., 2022): each prompt asks the model to reason step by step and provide a final answer in a specified format. Prompting templates, answer extraction, and decoding details are in Appendices C–D.

Correctness labels and evaluation. Each generated response receives a binary correctness label $c \in \{0, 1\}$ via answer verification: exact match for multiple-choice datasets, normalized numerical matching for mathematical datasets. Labels are used only after scoring. We report AUROC (↑) (Boyd et al., 2013), AUPR (↑) (Manning and Schutze, 1999), and FPR@95 (↓) (Hendrycks and Gimpel, 2016), together with macro-averages across benchmarks. For all scores, we orient the value so that higher indicates a higher likelihood of correctness.

4.2 Baselines

Output-derived baselines. These methods use the next-token probability distribution without accessing hidden states. For each step t, let $p_t(\cdot)$ be the vocabulary distribution and y_t the generated token. We compute token-averaged **MaxProb** $\max_v p_t(v)$, **Entropy** $-\sum_v p_t(v) \log p_t(v)$, and **Perplexity** $\exp(-\frac{1}{T}\sum_{t=1}^{T} \log p_t(y_t))$.

Trajectory-based baseline. Geometry-based methods extract statistical summaries from hidden-state trajectories to estimate correctness without training. The closest prior work is CoE (Wang et al., 2024), which operates on the same token-mean layer trajectory (Eq. 2), treating layerwise hidden-state evolution as a discrete curve. CoE derives two composite scores, CoE-R and CoE-C, by combining magnitude and angle statistics through hand-crafted rules. We discuss CoE in more detail alongside our proposed methods in Section 5.

5 Layer-wise Dynamics: From Trajectories to Norm Drop Rate

Takeaways. (i) Single-feature geometric scores can beat composite baselines. (ii) Correctness separability is localized in late layers. (iii) NDR captures this late-layer signal only using norm dynamics.

This section first shows that composite trajectory baselines can be matched by single-feature scores, then localizes where along layers these signals concentrate, and finally distills the strongest late-layer pattern into our simplest proposed score, NDR.

5.1 Trajectory as Diagnostic Scores

We use trajectory scores as probes to identify which geometric quantities carry separability. All scores are computed from the token-mean layer trajectory

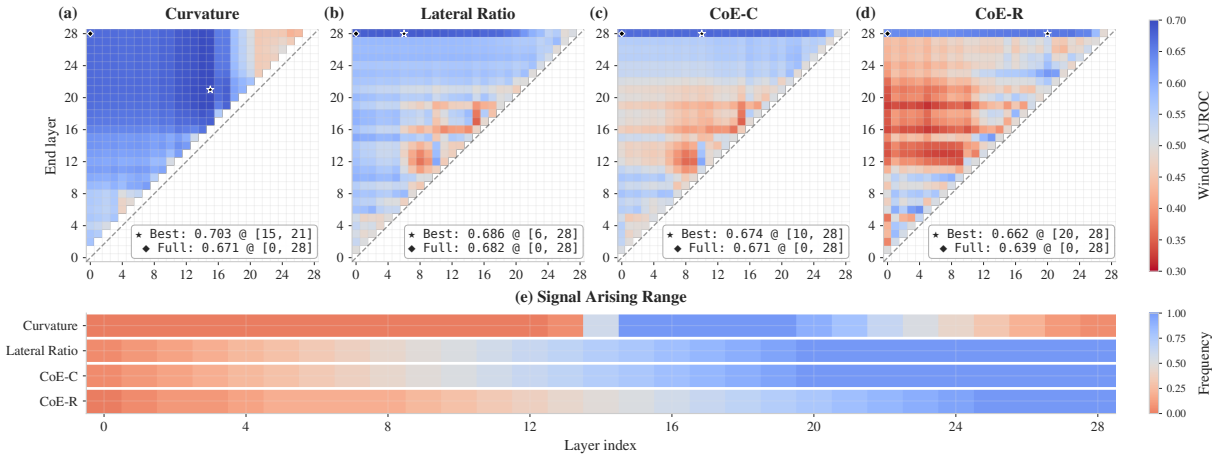

Figure 2: **Layer-window localization of trajectory scores (Qwen2-7B-Instruct).** For each contiguous layer window $[s_l, e_l]$, we recompute (a) Curvature, (b) Lateral Ratio, (c) CoE-C, and (d) CoE-R. $\star$ marks the best window; $\diamond$ marks the full-layer window $[0, L]$. (e) Per-layer frequency of appearing in the top 5% AUROC windows.

(Eq. 2).

CoE as a composite baseline. Let $M_l = \|\bar{h}^{l+1} - \bar{h}^l\|$, $A_l = \angle(\bar{h}^l, \bar{h}^{l+1})$, $Z_M = \|\bar{h}^L - \bar{h}^0\|$, and $Z_A = \angle(\bar{h}^0, \bar{h}^L)$. CoE (Wang et al., 2024) combines normalized magnitude and angle changes into two composite scores:

$$\text{CoE-R} = \frac{1}{L}\sum_{l=0}^{L-1}\left(\frac{M_l}{Z_M} - \frac{A_l}{Z_A}\right). \quad (3)$$

CoE-C maps each layer step to the complex plane:

$$\text{CoE-C} = \frac{1}{L}\left[\left(\sum_{l=0}^{L-1}\frac{M_l}{Z_M}\cos A_l\right)^2 + \left(\sum_{l=0}^{L-1}\frac{M_l}{Z_M}\sin A_l\right)^2\right]^{1/2}. \quad (4)$$

CoE separates correct from incorrect generations (Table 1), but its composite form makes it unclear which primitive feature drives the signal.

Curvature. Curvature captures local oscillation. Using central differences $v^{(l)} = (\bar{h}^{l+1} - \bar{h}^{l-1})/2$ and $a^{(l)} = \bar{h}^{l+1} - 2\bar{h}^l + \bar{h}^{l-1}$, we define

$$\text{Curv.} = \frac{1}{L-1}\sum_{l=1}^{L-1}\frac{\|a^{(l)} \times v^{(l)}\|}{\|v^{(l)}\|^3 + \epsilon}. \quad (5)$$

Lower curvature indicates smoother evolution.

Lateral Ratio. Lateral Ratio measures the fraction of path length orthogonal to the net displacement. Let $\hat{Z} = (\bar{h}^L - \bar{h}^0)/\|\bar{h}^L - \bar{h}^0\|$, $\Delta^l = \bar{h}^{l+1} - \bar{h}^l$, and $\Delta^l_\perp = \Delta^l - \langle \Delta^l, \hat{Z}\rangle \hat{Z}$. We define

$$\text{LateralRatio} = \frac{\sum_{l=0}^{L-1}\|\Delta^l_\perp\|}{\sum_{l=0}^{L-1}\|\Delta^l\|}. \quad (6)$$

Results. Table 1 shows that single primitives already match or exceed CoE. On Qwen2-7B-Instruct, Lateral Ratio achieves the best macro AUROC (68.2%) and Curvature the lowest macro FPR@95 (79.2%). On Llama-3-8B-Instruct, Curvature is strongest (63.0% AUROC).

5.2 Layer Localization

We next ask where along layers the signal concentrates. For each contiguous window $[s_l, e_l]$, we recompute each score on $(\bar{h}^l)_{l=s_l}^{e_l}$, with endpoint quantities recomputed inside the window.

Figure 2 shows that correctness signals are far from uniform. Endpoint-referenced scores (Lateral Ratio, CoE) are last-layer anchored: their best windows end near L, and removing late layers degrades performance sharply. Curvature peaks mid-to-late (best at $[15, 21]$). Some early windows yield AUROC below 0.5, meaning separability can flip across layers.

Since endpoint-based scores concentrate their signal near the last layer, we next examine whether a direct last-layer statistic can replace full-trajectory summaries.

5.3 Norm Drop Rate

Final-layer norm contraction. We compare the layerwise norm $\|\bar{h}^l\|_2$ between correct and incorrect generations. Figure 3 reveals a simple while consistent pattern: correct generations maintain larger token-mean norms through most layers, followed by a sharp contraction at the final layer that flips the sign of the difference.

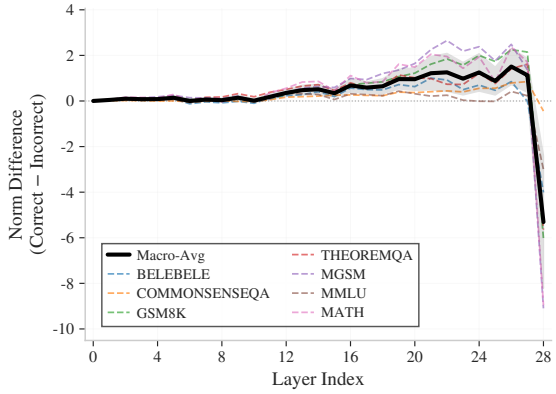

Figure 3: **Layer-wise norm difference.** $\Delta\|\bar{h}^l\|_2 = \mathbb{E}[\|\bar{h}^l\|_2 \mid c = 1] - \mathbb{E}[\|\bar{h}^l\|_2 \mid c = 0]$ across layers. Correct generations maintain larger norms through most layers, then show an abrupt final-layer contraction.

Why the final layer? The final residual state is directly consumed by the output embedding matrix W_U. Unlike intermediate layers, the final layer must align the accumulated representation with a vocabulary-space prediction. In the residual stream, $h^{l+1} = h^l + f^l(h^l)$, where f^l is the l-th block output. The norm of h^{l+1} decreases when $f^l(h^l)$ is anti-aligned with h^l. On Qwen2-7B-Instruct, the final-layer update is significantly more anti-aligned for correct generations across six of seven benchmarks ($p < 0.05$ in 6/7 datasets), confirming that correct responses undergo stronger final-layer norm contraction. A detailed analysis is provided in Appendix H.

Definition. Let $\bar{h}_{\text{avg}} = \frac{1}{L+1}\sum_{l=0}^{L} \bar{h}^l$. We define the **Norm Drop Rate** as

$$\text{NDR} = \frac{\|\bar{h}_{\text{avg}}\|_2}{\|\bar{h}^L\|_2}. \tag{7}$$

A larger NDR indicates stronger final-layer contraction relative to the trajectory scale. It is the most computationally efficient score in our comparison, requiring no complex operations beyond basic norms.

Results. NDR achieves 68.0% macro AUROC and the best AUPR (78.3%) on Qwen2-7B-Instruct, outperforming CoE-C in AUPR (76.8%) and all output-level baselines (Table 1a). On Llama-3-8B-Instruct, NDR (62.3%) remains competitive with Curvature (63.0%). NDR is not always the top AUROC score, but it is the simplest. Sections 6 and 7 further validate its robustness.

Relation to output confidence. To test whether NDR simply recapitulates output-level uncertainty, we measure its overlap with output entropy on Qwen2-7B-Instruct. NDR and entropy are moderately correlated (Spearman $\rho = -0.50$). However, after regressing out entropy, the residual NDR still achieves 60.1 AUROC, confirming that NDR carries discriminative information not explained by output uncertainty alone.

Method	AUROC (%)		Drop (%)
	Raw	Length-Matched	
Token Length	67.1	56.7	15.5
CoE-R$^\diamond$	66.7	64.2	3.7
CoE-C$^\diamond$	67.1	64.2	4.3
Lateral Ratio*	**68.2**	64.9	4.8
NDR*	68.0	**66.0**	**2.9**
Curvature*	67.1	60.6	9.7

Table 2: **Length confound and robustness under length matching.** Macro AUROC on the original test set (Raw) and on a strictly length-matched subset. Token length degrades sharply after matching; NDR retains 97.1% of its raw AUROC. Results on Qwen2-7B-Instruct.

6 Token-wise Dynamics: Length Confounds and Failure Spans

Takeaways. (i) Length is a strong but insufficient baseline. (ii) NDR is the most length-stable score after strict matching. (iii) Geometric signals mainly localize to hallucination token spans.

The layer analysis averages over all generated tokens, which can hide which response regions drive the score and whether the score is confounded by output length. We test whether NDR merely tracks response length or captures structure beyond verbosity.

6.1 Length-Matched Evaluation

Response length alone achieves 67.1% macro AUROC (Table 2). This is a strong baseline, but it is also a surface-level confound: length can correlate with correctness for many reasons unrelated to the model's internal computation, including task format, prompt design, and decoding behavior. Our goal is to study geometric signals *internal* to the hidden-state trajectory. We therefore control for length, rather than competing with it.

For each incorrect sample i with token length t_i, we select a correct sample with nearest length t_j (minimizing $|t_i - t_j|$), form matched pairs per

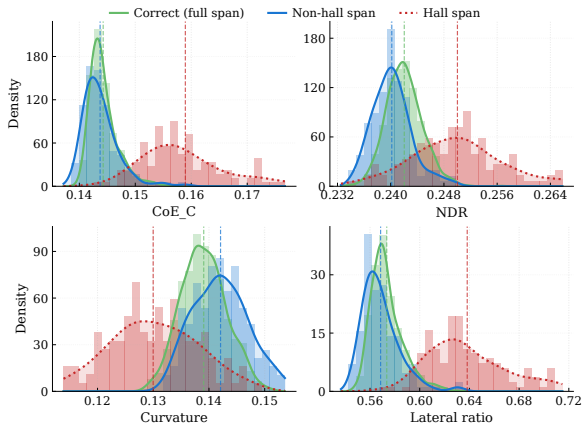

Figure 4: **Geometric signal localization to hallucinated spans (GSM8K).** Distributions of trajectory scores computed on span-mean trajectories. Green: correct responses (full span; 1128 samples). Red: hallucinated spans $\mathcal{T}_{\text{hall}}$ within incorrect responses (137 samples). Blue: non-hallucinated spans $\mathcal{T}_{\neg\text{hall}}$ from the same 137 responses. Hallucinated spans shift away from the correct regime; non-hallucinated spans largely overlap with correct responses.

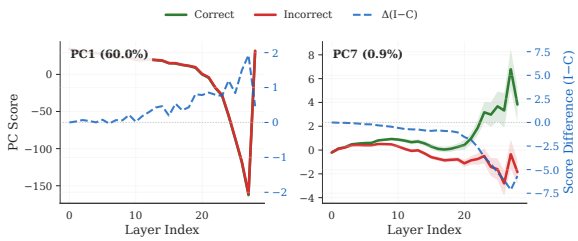

Figure 5: **PC trajectories can separate correctness off the top-variance axes (TheoremQA).** PC1 (60.0% variance) shows overlapping trajectories for correct and incorrect generations; PC7 (0.9% variance) diverges in late layers with large $\Delta(\mathrm{I}-\mathrm{C})$.

dataset, and recompute AUROC on this subset. Matching is tight: the median length difference is zero across datasets (e.g., GSM8K yields $k = 223$ pairs).

After matching, the length baseline collapses toward chance (67.1 → 56.7). NDR retains 97.1% of its raw AUROC (68.0 → 66.0), the smallest relative degradation in Table 2. CoE variants each drop about 4%, Lateral Ratio drops 4.8%, and Curvature loses nearly 10%. This does not prove that NDR is fully independent of termination behavior, but it rules out the simplest explanation that NDR merely tracks response length.

6.2 Failure Token Span Localization

Length matching confirms that geometric signals exist beyond response length. As we localized signals across layers in Section 5, we now localize them across token regions within a response.

For each incorrect response on GSM8K, we use a stronger LLM as a judge[1] to annotate hallucination spans $\mathcal{T}_{\text{hall}}$; the remaining tokens form the non-hallucinated span $\mathcal{T}_{\neg\text{hall}}$. These annotations are used for analysis only and are never used to compute any score. We then define three groups: (i) correct responses using the full token span, (ii) hallucinated spans from incorrect responses, and (iii) non-hallucinated spans from the same incorrect responses. For each group, we recompute the span-mean trajectory $\bar{h}^l_{\mathcal{T}}$ and apply the same geometric scores.

Figure 4 compares the three groups. Hallucinated spans deviate strongly from the correct distribution, while non-hallucinated spans from the same incorrect responses largely overlap the correct regime. This explains why full-response NDR remains effective despite token averaging: a short failure span with abnormal norm dynamics can shift the token-mean trajectory enough to affect the final-layer contraction ratio.

7 Representation Dynamics: Basis Stability

> **Takeaways.** (i) PCA can capture correctness signals, but the useful component is dataset-dependent and requires labels to select. (ii) NDR is the most stable score under basis changes (2.9 pp spread across four bases). (iii) NDR is also the most efficient, requiring only two norms.

Sections 5 and 6 analyze correctness signals in the native hidden space. This section tests whether those signals depend on the choice of coordinate system.

7.1 PCA Reveals a Selection Bottleneck

To test whether a latent subspace exposes additional structure, we fit PCA per dataset using token-mean states $\{\bar{h}^l\}$ as samples across examples and layers, then project onto the top-k components: $\tilde{h}^l = U_k^\top(\bar{h}^l - \mu)$, $U_k \in \mathbb{R}^{d\times k}$. Each coordinate $\tilde{h}^l_j$ forms a one-dimensional curve over depth.

A single PC can separate correct from incorrect generations, but the discriminative PC is not re-

[1] We use GPT-4o to annotate token spans containing factual errors or unsupported logical steps.

Selection Method	Macro AUROC
Best PC | Best Layer	0.724
Best PC | Layer Mean	0.672
Best PC | Last Layer	0.668
Naive Sum PC | Last Layer	0.596

Table 3: **The PCA selection gap.** Label-selected PCs achieve high separability, but naive unsupervised aggregation misses much of the signal.

liably the high-variance one and does not appear at consistent ranks across datasets. Figure 5 illustrates: on TheoremQA, PC1 explains 60.0% of the variance but shows overlapping trajectories, while PC7 explains only 0.9% yet exhibits strong late-layer separation.

Table 3 quantifies this selection bottleneck. With label-selected PCs and layers, PCA reaches 0.724 macro AUROC, well above any fully unsupervised score. But naive aggregation (summing all PCs at the last layer) drops to 0.596, below NDR's 0.680 on the same Qwen2-7B scale. The bottleneck is not signal strength but unsupervised selection: without labels, there is no reliable way to identify which PC carries the signal. NDR sidesteps this entirely by summarizing the trajectory through norms rather than through basis-sensitive directions.

7.2 NDR Is Stable Across Representation Bases

We next test whether the same trajectory scores remain stable under alternative coordinate maps. We compare four bases: **Hidden space** (native coordinates, the default); **PCA** (dataset-specific top-k components, $k = 512$); **LM-head SVD** (top-k right-singular directions of W_U); and **Logit space** ($z = W_U h$).

Table 4 shows that NDR is the most basis-stable score: its AUROC ranges from 65.1 to 68.0, a 2.9 percentage-point spread, compared with 12.8 for Lateral Ratio and 12.2 for CoE-C. Shape-sensitive scores degrade sharply under output-aligned projections: Lateral Ratio drops from 68.2 in hidden space to 55.5 in logit space, while NDR remains at 66.9. NDR depends only on L2 norms, which degrade more gracefully than angle-based shape statistics under non-orthogonal projections. Native hidden space is the strongest default overall, but NDR is the score least affected by departing from it.

Computational cost. NDR is also the most lightweight score in our comparison. After hidden states are logged, it requires only one layer-wise average and two L2-norm computations. It uses no pairwise layer angles, no trigonometric functions, no matrix decomposition, no learned probe, and no softmax over the vocabulary. By contrast, CoE requires L pairwise norm and angle computations with trigonometric operations, Curvature requires second-order finite differences at each layer, and Lateral Ratio requires projecting each layer step onto the endpoint direction. While all scores are fast relative to the forward pass, NDR's minimal footprint makes it especially suitable for deployment scenarios where hidden states are already logged.

Metric	Hidden	LM Head	Logit	PCA	Spread
CoE-R$^\diamond$	63.9	63.6	**64.5**	58.8	5.7
CoE-C$^\diamond$	67.1	63.3	55.4	**67.6**	12.2
Curvature*	67.1	65.9	62.4	**68.3**	5.9
Lateral Ratio*	68.2	62.4	55.5	**68.3**	12.8
NDR*	**68.0**	65.1	66.9	66.1	**2.9**

Table 4: **Trajectory scores across representation bases.** Macro AUROC over seven datasets (%). Spread = max minus min AUROC across bases. NDR has the smallest spread. PCA uses $k = 512$.

8 Conclusion

We studied geometric dynamics in LLM hidden representations for training-free correctness self-evaluation, organizing the analysis along three dimensions: layer depth, token span, and representation space. Across benchmarks and models, correctness signals concentrate in late-stage computation, and many complex trajectory scores reduce to simple statistics on restricted layer regions. This localization leads to NDR, a minimal score based on final-layer norm contraction that matches or exceeds more complex methods while being the most length-robust, basis-stable, and computationally lightweight score in our comparison. Token-span analysis further shows that geometric deviations localize to hallucinated regions, and representation analysis confirms that NDR is the least sensitive to coordinate changes. Extending this localization-then-distillation approach to broader reliability targets such as faithfulness and safety, to larger models, and to earlier detection during generation are natural next steps. These results suggest that hidden-state trajectory analysis should prioritize localizing simple, targeted signals over designing increasingly complex summaries.

Limitations

Model coverage. We evaluate two instruction-tuned 7B/8B models from different families. While the main findings (late-layer localization, NDR competitiveness) are consistent across both, we have not verified whether they hold at larger scales, for other architectures, or under fine-tuning beyond instruction tuning.

Inference protocol. All experiments use greedy decoding with one response per prompt. Trajectory geometry may change under temperature sampling, beam search, or tool-augmented decoding. Extending the analysis to stochastic decoding regimes is a natural next step.

Hallucination span annotations. Token-span localization relies on post-hoc annotations from a stronger LLM. Such annotations may be noisy for subtle logical errors or partially correct reasoning. The qualitative separation between hallucinated and non-hallucinated spans is robust in our setting, but the exact degree of localization may vary with alternative annotation protocols.

Ethical Considerations

This work studies offline correctness-evaluation signals from hidden states of public LLM checkpoints on public benchmarks. We do not collect private user data, train new models, or conduct human-subject experiments. The main risk is over-reliance: a geometric score such as NDR may make an incorrect answer appear more reliable than it is. We therefore present the method as a diagnostic signal rather than a calibrated guarantee of correctness, and recommend additional task-specific validation for high-stakes use cases.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 model card.

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*, pages 967–976, Singapore.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Kendrick Boyd, Kevin H Eng, and C David Page. 2013. Area under the precision-recall curve: point estimates and confidence intervals. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 451–466. Springer.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023. Theoremqa: A theorem-driven question answering dataset. *arXiv preprint arXiv:2305.12524*.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. 2025. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

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

Danny Halawi, Jean-Stanislas Denain, and Jacob Steinhardt. 2024. Overthinking the truth: Understanding how language models process false demonstrations. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Dan Hendrycks and Kevin Gimpel. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Vedang Lad, Jin Hwa Lee, Wes Gurnee, and Max Tegmark. 2026. Remarkable robustness of llms: Stages of inference? *Advances in Neural Information Processing Systems*, 38:130050–130083.

Junteng Liu, Shiqi Chen, Yu Cheng, and Junxian He. 2024. On the universal truthfulness hyperplane inside LLMs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Miami, Florida, USA.

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.

Christopher Manning and Hinrich Schutze. 1999. *Foundations of statistical natural language processing*. MIT press.

Rob Manson. 2025. Curved inference: Concern-sensitive geometry in large language model residual streams. *arXiv preprint arXiv:2507.21107*.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*.

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

nostalgebraist. 2020. Interpreting GPT: The logit lens. https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens. Accessed: 2024-01-01.

Charles O'Neill, Slava Chalnev, Chi Chi Zhao, Max Kirkby, and Mudith Jayasekara. 2025. A single direction of truth: An observer model's linear residual probe exposes and steers contextual hallucinations. *arXiv preprint arXiv:2507.23221*.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2025. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. 2025. Steer LLM latents for hallucination detection. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 47971–47990.

Adam S. Shai, Sarah E. Marzen, Lucas Teixeira, Alexander Gietelink Oldenziel, and Paul M. Riechers. 2024. Transformers represent belief state geometry in their residual stream. *arXiv preprint arXiv:2405.15943*.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. 2022. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (NAACL HLT)*, pages 4149–4158, Minneapolis, Minnesota.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yiming Wang, Pei Zhang, Baosong Yang, Derek F Wong, and Rui Wang. 2024. Latent space chain-of-embedding enables output-free llm self-evaluation. *arXiv preprint arXiv:2410.13640*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le,

and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 24824–24837.

Sarah Wiegreffe, Oyvind Tafjord, Yonatan Belinkov, Hanna Hajishirzi, and Ashish Sabharwal. 2025. Answer, assemble, ace: Understanding how lms answer multiple choice questions. In *International Conference on Learning Representations*, volume 2025, pages 78293–78318.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, and 2 others. 2023. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*.

A Responsible NLP and Reproducibility Details

This appendix documents the artifacts, execution environment, and protocol details needed to reproduce the reported analyses. The study uses public benchmark datasets and public instruction-tuned model checkpoints. We do not train new models, collect new human-subject data, or use private user data. The only additional annotations are post-hoc hallucination-span annotations produced for analysis, not for model training or score selection.

A.1 Artifact summary, licenses, and intended use

Table 5 summarizes the datasets, models, and derived artifacts used in the paper. Dataset and model licenses should be verified against the exact dataset card or model card version used at download time before release of any supplementary archive. We release only derived predictions, scalar metrics, and scripts when allowed by the corresponding artifact terms; raw benchmark data and model weights should be obtained from the original providers.

Intended use. The benchmark datasets are used only for offline evaluation of correctness-detection signals. The model checkpoints are used only for deterministic inference and hidden-state extraction. NDR and the other geometric scores are diagnostic signals for model self-evaluation; they are not intended as stand-alone safety filters for high-stakes domains. A high score should not be interpreted as a guarantee of factual correctness.

Privacy and offensive content. We do not collect new personally identifiable information. All inputs come from public benchmarks. We do not perform additional filtering for offensive content beyond the filtering and documentation provided by the original benchmark creators. Generated model outputs may contain erroneous, biased, or unsafe content; the experiments analyze such outputs offline and do not expose them to users.

Responsible use risk. The main risk is over-reliance: a training-free hidden-state score can make an incorrect response appear more reliable than it is. We therefore report AUROC, AUPR, and FPR@95 rather than presenting the method as a calibrated correctness guarantee. In deployment, these scores should be combined with task-specific validation, uncertainty calibration, and human review where the cost of an error is high.

A.2 Dataset coverage and splits

Table 6 gives the benchmark coverage used for evaluation. For each dataset–model pair, correctness is defined after answer extraction against the benchmark-provided gold answer. Dataset-specific parsers are described in Appendix C.

Multilingual coverage. MGSM is evaluated in the standard language set: English plus Bengali, Chinese, French, German, Japanese, Russian, Spanish, Swahili, Telugu, and Thai. Belebele is evaluated using the language variants included in the benchmark release; when all variants are used, this corresponds to 122 language variants with 900 questions per variant. We do not translate prompts or answers ourselves.

B Dataset Statistics

Table 7 reports per-dataset statistics for Qwen2-7B-Instruct, including sample counts, accuracy, and mean response lengths for correct and incorrect generations.

Artifact	Type	Creator / citation	Use in this work	License / terms to check
GSM8K	Dataset	Cobbe et al. (2021)	Grade-school math reasoning; correctness labels from final numerical answers.	MIT-style dataset release.
MATH	Dataset	Hendrycks et al. (2021)	Competition mathematics; final-answer extraction with numerical/symbolic normalization.	Dataset release terms.
TheoremQA	Dataset	Chen et al. (2023)	Theorem-grounded QA; answer format depends on answer type.	Dataset release terms.
MMLU	Dataset	Hendrycks et al. (2020)	Multi-domain multiple-choice knowledge and reasoning.	Dataset release terms.
CommonsenseQA	Dataset	Talmor et al. (2019)	Multiple-choice commonsense reasoning.	Dataset release terms.
Belebele	Dataset	Bandarkar et al. (2024)	Multilingual reading comprehension and language coverage analysis.	Dataset release terms.
MGSM	Dataset	Shi et al. (2022)	Multilingual grade-school math reasoning.	Dataset release terms.
Qwen2-7B-Instruct	Model	Yang et al. (2024)	Instruction-tuned 7B LLM for generation and hidden-state logging.	Apache-2.0 model release.
Llama-3-8B-Instruct	Model	AI@Meta (2024)	Instruction-tuned 8B LLM for generation and hidden-state logging.	Meta Llama 3 community license.
Hidden-state tensors	Derived artifact	This work	Intermediate activations used to compute geometric scores.	Not redistributed unless model/dataset terms allow.
Hallucination spans	Derived annotation	This work	Analysis-only token-span localization; never used for training or score selection.	Release only if underlying benchmark terms allow.

Table 5: Artifacts used by the study. Licenses should be rechecked against the exact versions downloaded for the final submission archive.

Incorrect responses are consistently longer than correct ones across all seven datasets, with the gap most pronounced on MATH (374 vs. 713 tokens). This systematic length difference motivates the length-matched evaluation in Section 6 and confirms that response length is a genuine confound that must be controlled for.

B.1 Models, hardware, and software

Table 8 summarizes the model and compute setup. All experiments are inference-only. No gradient updates, supervised probes, LoRA adapters, or verifier models are trained.

Software environment. The experiments require PyTorch, Hugging Face `transformers`, `datasets`, `accelerate`, NumPy, SciPy, scikit-learn, and a tokenizer implementation matching each model checkpoint. For reproducibility, the final release should pin exact versions in a `requirements.txt` or `environment.yml`; the metrics reported in the paper are deterministic once model checkpoint, tokenizer version, prompts, and generation settings are fixed.

C Prompting and Answer Extraction

We use dataset-specific prompts that enforce a standardized final-answer format while allowing the model to generate free-form reasoning, consistent with chain-of-thought style prompting (Wei et al., 2022). Each dataset specifies a required final-line marker such as `Answer:`, `Answer: A`, or `boxed{}`. The parser only uses the final answer string; intermediate reasoning is not used to decide correctness except in the post-hoc hallucination-span analysis of Appendix G.

C.1 Prompt templates

For mathematical and open-answer datasets, we use the following template:

> **Question:** `<question>`
> **Instruction:** Solve the problem. Put the final answer on the last line in the format `Answer: <answer>`.

Dataset	Language coverage	Split / subset	Public benchmark size	Correctness criterion
GSM8K	English	Test	1,319 examples	Exact final numerical answer after normalization.
MATH	English	Test	5,000 test examples	Final boxed or extracted mathematical answer after normalization.
TheoremQA	English	Evaluation set	800 QA pairs	Match to answer type: boolean, option, number, expression, list, or text.
MMLU	English	Test / evaluation subjects	57 subjects; roughly 14k test questions	Predicted multiple-choice option equals gold option.
CommonsenseQA	English	Development / validation	1,221 labeled validation examples	Predicted multiple-choice option equals gold option.
Belebele	Multilingual	Evaluated language variants	900 questions per language variant; 122 variants in the full release	Predicted multiple-choice option equals gold option.
MGSM	Multilingual	Test	250 problems per language; 11 languages including English	Exact final numerical answer after normalization.

Table 6: Benchmark coverage. If a smaller evaluated subset is used for a final run, the exact sampled count, correct count, incorrect count, and mean response length should be reported from the run metadata.

Dataset	N	Correct	Acc.%	Avg. Len	
				C	I
Belebele	900	797	88.6	158	223
CSQA	1221	947	77.6	171	188
GSM8K	1319	1099	83.3	217	285
MATH	5000	2605	52.1	374	713
MGSM	250	215	86.0	214	288
MMLU	14042	9697	69.1	137	167
TheoremQA	221	61	27.6	347	485

Table 7: **Dataset statistics for Qwen2-7B-Instruct.** N = total samples, C/I = correct/incorrect count, Avg. Len = mean response length in tokens. Incorrect responses are systematically longer across all datasets, motivating the length-matched evaluation in Section 6.

For multiple-choice datasets, we use:

> **Question:** `<question>`
> **Options:** `(A) <option A> ...`
> **Instruction:** Choose the best answer. Put only the final option letter on the last line in the format `Answer: <A/B/C/D/...>`.

For TheoremQA, the instruction additionally includes the answer type when provided by the dataset. For multilingual datasets, the question and options are kept in the benchmark language; the final-answer marker is kept fixed so that answer extraction is uniform.

C.2 Extraction and normalization

We parse the final answer using deterministic dataset-specific rules:

- **Numerical answers.** We remove commas, trailing punctuation, currency symbols, and surrounding explanatory text. Fractions and decimals are normalized when possible. A prediction is correct if the normalized value matches the gold answer under the dataset-specific tolerance.
- **Mathematical expressions.** We first extract a boxed answer if present, then fall back to the final `Answer:` line. We normalize whitespace, punctuation, and common LaTeX wrappers before comparison.
- **Multiple choice.** We extract the final option letter from the required final line. If multiple option letters appear, the last explicitly marked final answer is used.
- **Boolean and list answers.** Boolean strings are lowercased and mapped to canonical `true`/`false` labels. Lists are normalized by delimiter, whitespace, and ordering only when the dataset defines the answer as unordered.

Examples that do not yield a valid final answer are counted as incorrect. The same extraction rules are applied to both models and all compared scores.

D Decoding and Hidden-State Logging

We generate exactly one response per prompt using greedy decoding: temperature $= 0$, sampling disabled, one returned sequence, and a maximum of 2048 new tokens. Generation stops at the model's end-of-sequence token or the maximum length. This single-pass protocol is important because the compared scores are intended for label-free self-evaluation from one generated response, not from multiple sampled responses.

Component	Setting	Hardware	Notes
Qwen2-7B-Instruct inference	bfloat16 or float16 inference, no quantization	NVIDIA A100 and NVIDIA A10	Hidden states are logged during greedy generation.
Llama-3-8B-Instruct inference	bfloat16 or float16 inference, no quantization	NVIDIA A100 and NVIDIA A10	Same decoding and logging pipeline as Qwen2.
Metric computation	NumPy / PyTorch / scikit-learn	CPU plus A10 when batching is useful	AUROC, AUPR, and FPR@95 are computed from scalar scores.
PCA / SVD projections	Centered activations; truncated bases where specified	CPU plus A10 / A100 depending on tensor size	Projection bases are fit without correctness labels unless explicitly marked as oracle analysis.
Hallucination-span preprocessing	JSON parsing and tokenizer offset alignment	CPU	Stronger-LLM annotations are analysis-only.

Table 8: Model and compute setup. The project archive records the hardware class as A100/A10. Exact scheduler GPU-hours should be inserted from the job logs if required by the submission form; the source archive did not include reliable wall-clock accounting.

For each generated token, we store the residual hidden state at every transformer layer. Let $h_t^l \in \mathbb{R}^d$ denote the hidden state for generated token $t \in \{1, \ldots, T\}$ at layer $l \in \{0, \ldots, L\}$. Unless otherwise stated, prompt-token states are excluded and scores are computed only over generated tokens. The default token-mean representation at layer l is

$$\bar{h}^l = \frac{1}{T} \sum_{t=1}^{T} h_t^l. \tag{8}$$

All geometric scores in the main paper are computed from these stored hidden states and never from training labels.

E Metrics and Score Orientation

All scalar scores are oriented so that larger values indicate a higher predicted probability of correctness. AUROC and AUPR are computed with correctness as the positive class. FPR@95 is the false-positive rate among incorrect responses at the threshold where the true-positive rate on correct responses is 95%. For scores whose natural direction is reversed, such as perplexity or entropy, we multiply by -1 before computing threshold-based metrics.

For output-level baselines, we use the final generated answer tokens and the model's own token probabilities. For hidden-state scores, we use the tensor $\{h_t^l\}_{t,l}$ logged during the same generation. No score uses gold labels during computation. Labels are used only for evaluation metrics and for defining diagnostic subsets such as length-matched pairs or hallucination-span groups.

F Length-Matching Protocol

Response length is a strong confound for correctness detection, so we evaluate whether hidden-state geometry remains informative after controlling for length. For each dataset and model, let T_i be the number of generated tokens for example i. For each incorrect response, we select a correct response from the same dataset whose length is closest in absolute difference:

$$j^*(i) = \arg\min_{j: c_j = 1} |T_i - T_j|, \tag{9}$$

where $c_j = 1$ denotes a correct response. Ties are broken deterministically by example index. Evaluation is then restricted to the matched correct/incorrect subset. This procedure uses correctness labels only to construct the diagnostic evaluation subset; it does not alter the hidden-state scores.

We report the number of matched pairs and the mean length gap for each dataset–model pair in the final run metadata. The GSM8K analysis in the main text follows this same protocol.

G Hallucination-Span Annotation Protocol

The token-span analysis in Section 6 uses post-hoc hallucination-span annotations only for diagnostic visualization and group comparison. These annotations are not used to compute NDR, CoE, curvature, output baselines, or any other self-evaluation score.

G.1 Judge input and output

For each incorrect response selected for span analysis, we provide a stronger LLM judge with the

benchmark question, the gold answer, the generated response, and the extracted final answer. The judge is asked to identify minimal contiguous spans in the generated response that contain unsupported factual claims, invalid mathematical steps, or logically decisive errors. The judge returns JSON with the following fields:

```
{
"is_incorrect": true/false,
"spans": [{
"start_text": "...",
"end_text": "...",
"error_type": "unsupported_claim
| invalid_step | arithmetic |
contradiction | other",
"rationale": "brief explanation"
}]
}
```

The judge temperature is set to zero. If the judge output is not valid JSON or cannot be aligned back to the generated text, the example is excluded from the span-level analysis but remains part of the main correctness-evaluation experiments.

G.2 Span definition and token alignment

A hallucination span is the smallest contiguous text region that makes the response unsupported or wrong relative to the question and gold answer. For mathematical reasoning, the span can be an invalid equation, an arithmetic error, a false inference, or a step that irreversibly leads to the wrong final answer. For factual or commonsense reasoning, the span can be an unsupported claim, contradiction, or answer option justification that is inconsistent with the gold label.

We map character spans to model tokens with tokenizer offset alignment. Tokens whose character offsets overlap a hallucination span are assigned to $\mathcal{T}_{\text{hall}}$; all remaining generated tokens in the same incorrect response are assigned to $\mathcal{T}_{\neg\text{hall}}$. For each span group, we recompute the layer trajectory by replacing Eq. 8 with

$$\bar{h}^l_{\mathcal{T}} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} h^l_t. \tag{10}$$

The same geometric scores are then applied to the span-mean trajectories.

G.3 Bias and failure modes

The stronger-LLM judge can miss subtle reasoning errors, mark too broad a span, or confuse a downstream consequence with the first erroneous step. The span analysis should therefore be interpreted qualitatively: it tests whether geometric deviations concentrate in tokens identified as erroneous by a consistent protocol, not whether the judge provides a definitive theory of hallucination. The main paper keeps all primary correctness-detection results independent of these annotations.

Dataset	Correct	Incorrect	Δ	p-value
Belebele	-0.768	-0.759	-0.010	4.1e-5
CSQA	-0.740	-0.739	-0.001	0.28
GSM8K	-0.813	-0.795	-0.019	8.8e-18
MATH	-0.819	-0.796	-0.023	5.4e-186
MGSM	-0.812	-0.789	-0.023	7.0e-6
MMLU	-0.739	-0.729	-0.009	9.9e-43
TheoremQA	-0.789	-0.781	-0.008	0.046
Macro	$\mathbf{-0.783}$	$\mathbf{-0.770}$	$\mathbf{-0.013}$	—

Table 9: **Final-layer cosine alignment on Qwen2-7B-Instruct.** $\cos\theta_L = \cos(f^L(h^{L-1}), h^{L-1})$ averaged over correct and incorrect generations. More negative = stronger anti-alignment. Correct generations are more anti-aligned in 6/7 datasets ($p < 0.05$).

H Norm Drop Rate: Final-Layer Contraction Analysis

This section expands the final-layer explanation referenced in Section 5.3. Let h^l denote a token-mean residual state at layer l, and let $f^l(h^l)$ be the update applied by block l. The residual recurrence is

$$h^{l+1} = h^l + f^l(h^l). \tag{11}$$

The squared norm after the update is

$$\|h^{l+1}\|_2^2 = \|h^l\|_2^2 + \|f^l(h^l)\|_2^2 + 2\langle h^l, f^l(h^l)\rangle. \tag{12}$$

Thus the norm decreases when the update is sufficiently anti-aligned with the current state:

$$\langle h^l, f^l(h^l)\rangle < -\tfrac{1}{2}\|f^l(h^l)\|_2^2. \tag{13}$$

Empirical verification. We compute the cosine similarity $\cos\theta_L = \langle h^{L-1}, f^L(h^{L-1})\rangle / (\|h^{L-1}\|\|f^L(h^{L-1})\|)$ for each sample on Qwen2-7B-Instruct and compare correct versus incorrect generations. Table 9 reports the results per dataset.

Across six of seven benchmarks, the final-layer update is significantly more anti-aligned for correct generations. The effect size is small in absolute terms ($|\Delta| \approx 0.01$–0.02), which is consistent with the fact that NDR captures a subtle but systematic signal. CommonsenseQA is the only dataset

without a significant difference ($p = 0.28$), likely because its short multiple-choice format produces less variation in trajectory dynamics. The consistency of the direction across the remaining datasets supports the interpretation that final-layer norm contraction is a reliable correlate of correctness.

Why not use the cosine directly? The final-update cosine is useful for interpretation, but it requires explicitly comparing consecutive residual updates and can be more sensitive to implementation choices about where hidden states are read from the transformer block. NDR is a simpler observable that remains defined whenever layerwise hidden states are available.

I Representation-Space Analyses

Section 7 tests whether geometric correctness signals are stable under changes of representation basis. We evaluate three spaces: the native hidden-state space, PCA-projected hidden space, and vocabulary/logit-related spaces.

I.1 PCA basis

For each model and dataset, we collect token-mean hidden states from the evaluated examples and center them by subtracting the empirical mean. PCA is fit without correctness labels. Unless otherwise stated, we project to the top $k = 512$ principal components:

$$z^l = U_k^\top (\bar{h}^l - \mu), \tag{14}$$

where U_k contains the top principal directions. Geometric scores are recomputed in the projected coordinates z^l. When the paper reports best-PC or PC-ranking diagnostics, labels are used only after PCA fitting to analyze which directions are most discriminative; those oracle diagnostics are not used as deployable label-free scores.

I.2 LM-head singular-vector basis

Let $W_U \in \mathbb{R}^{|V| \times d}$ be the unembedding matrix. We compute a truncated singular-vector basis from W_U and project hidden states onto the corresponding right singular vectors:

$$z^l = V_k^\top \bar{h}^l. \tag{15}$$

This tests whether correctness signals align with directions that are directly relevant to vocabulary prediction.

I.3 Logit space

For logit-space diagnostics, we map the final hidden representation to vocabulary logits:

$$\ell^l = W_U \bar{h}^l. \tag{16}$$

We then recompute norms, angles, and separability scores in this transformed space. These diagnostics are more expensive than NDR because they require vocabulary-scale projections, so they are used to analyze basis dependence rather than to define the simplest deployable score.