DEPTHWISE TRAJECTORY SIGNATURES: DETECTING HALLUCINATIONS FROM INSIDE

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models can produce fluent answers that are wrong, and most safeguards verify the answer only after it is complete. We take a different approach. Instead of checking what the model said, we monitor how it decided. Inside a transformer, a token prediction is built block by block, so the hidden state traces a trajectory across depth. We summarize this trajectory with a small set of geometric quantities, trajectory signatures, constructed to be invariant to the arbitrary choices that normally make depthwise measurements incomparable across layers, samples, and models. On these signatures we train a lightweight validator that detects hallucinations without modifying the base model. The same signatures localize the depth the validator finds anomalous, enabling a targeted correction at a single block during generation. Across 12 benchmarks and 5 LLMs, the validator achieves the best AUROC in 45 of 60 settings against baselines, and the refinement reduces the hallucination ratio by at least 10% relative in 29 of 45.

1 INTRODUCTION

Large language models often produce fluent answers that are locally coherent yet globally wrong (Huang et al., 2025; Maynez et al., 2020; Lin et al., 2022a; Li et al., 2023). Most safeguards verify the output *after* generation is complete, by retrieving external evidence (Lewis et al., 2020; Nakano et al., 2021), consulting a second LLM as a judge (Zheng et al., 2023), or resampling the model to test self-consistency (Manakul et al., 2023; Kuhn et al., 2023; Farquhar et al., 2024). The difficulty is that the finished text is a poor witness. Hallucination is fluent by definition, so the surface that these methods inspect is precisely what the failure leaves undisturbed. External checks therefore buy their evidence at a cost, several complete generations or a second model on top of the answer they inspect. Even then, a hallucination the model commits to consistently can still pass them. When a check does fire correctly, it ends at a verdict. It can say that the answer is wrong, but not where the generation went wrong, so its only remedy is to discard the answer and start over.

If the finished answer is a poor witness, the process that produced it is a better one. A transformer does not emit a token in one step. The prediction is formed gradually, refined block by block along the model's depth, and intermediate states often reveal the outcome before the final layer does (Belrose et al., 2023; Ferrando et al., 2023; Pal et al., 2023). A line of detectors already exploits this by probing hidden states (Kadavath et al., 2022; Azaria & Mitchell, 2023; Chen et al., 2024), but they read the internal state as a collection of snapshots, summarizing one layer or pooling over all of them, and even a stack of snapshots from every layer records a set of positions, not the movement between them. Decisions that end in similar states can differ in the way they were reached, and whatever distinguishes a sound decision from a hallucinated one in that difference is invisible to any detector built from static positions. We therefore treat each token prediction as a trajectory across depth, and we hypothesize that hallucinated decisions leave a detectable trace in how the trajectory unfolds. We do not fix in advance what this trace looks like. Instead we let a trained validator discover which trajectory patterns are discriminative, and we find empirically that the discriminative patterns differ across model families, which is itself a reason not to fix the signature by hand.

Reading this trajectory is not as simple as recording hidden states, because raw depthwise measurements reflect more than the decision being formed. The frame of reference in which the trajectory is expressed shifts from block to block, so quantities read off in naive coordinates confound the motion of the decision with the motion of the frame. The confound reaches further than depth alone. The

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

coordinates come from fitted depth-local subspaces, and a fitted subspace carries arbitrary degrees of freedom, so naive measurements change with implementation details that have nothing to do with the model. We therefore construct *trajectory signatures*, a small set of geometric summaries built to be provably invariant to these choices, so that what they measure is the decision and not the instrument. On these signatures we train a validator small enough to run alongside generation, and it predicts whether the answer is hallucinated, all without modifying or fine-tuning the base model. The same signatures serve a second purpose. Because the validator scores every token at every block, the position it finds most anomalous localizes the failure, answering the question that external checks cannot, not only whether the generation went wrong, but where. A minimal refinement operator acts on this answer, rolling generation back to the implicated token and intervening at a single transformer block while leaving every other depth untouched.

We evaluate this approach across twelve benchmarks spanning question answering, general-domain factuality, summarization, and reasoning, on five open-weight base LLMs. Against five established detectors, the validator achieves the best AUROC in 45 of 60 model and dataset combinations, with a mean advantage of $+3.32$pp over the strongest baseline in each setting, and the single-block intervention reduces the hallucination ratio by at least 10% relative in 29 of 45 combinations on the nine refinement benchmarks, at 22 to 32% per-step extraction overhead. The results also mark the boundary of the approach. Gains concentrate where a failure is localized at a specific depth, and shrink on multi-step reasoning, where errors accumulate across blocks.

2 Related work

Post-hoc hallucination detection. A large body of work verifies LLM outputs after generation is complete. Sampling-based detectors such as SelfCheckGPT (Manakul et al., 2023) draw multiple continuations per query and test their mutual consistency, scalar uncertainty estimators such as Semantic Entropy (Kuhn et al., 2023; Farquhar et al., 2024) aggregate likelihood-derived signals over resampled answers, and other approaches consult external evidence through retrieval (Lewis et al., 2020; Nakano et al., 2021) or a separate judge model (Zheng et al., 2023; Li et al., 2023). All of them draw evidence from generated text, and inherit its limits. Cost grows with the samples they buy, and a hallucination that resamples consistently presents the same evidence as a correct answer.

Internal-representation detectors. A second line reads the generator's internal state, placing the evidence where the decision is made. Early probing work established that hidden states carry information about factual correctness (Kadavath et al., 2022; Azaria & Mitchell, 2023; Ji et al., 2024). Building on this, INSIDE (Chen et al., 2024) scores responses from the covariance structure of their embeddings, LLM-Check (Sriramanan et al., 2024) derives eigenvalue statistics of internal maps, and HalluShift (Dasgupta et al., 2025) measures distributional shift between layers. All of these summarize states held fixed, and even HalluShift, which compares across layers, reduces the comparison to aggregate shift statistics. None represents the trajectory itself, the ordered movement of a token's state from block to block, which is what our signatures are built to measure.

Depth probing and representation geometry. A third line studies the depthwise computation itself rather than hallucination. Intermediate-readout methods such as Logit Lens (nostalgebraist, 2020; Elhage et al., 2021) and Tuned Lens (Belrose et al., 2023) decode logits from the hidden state at every block, showing that the residual stream is interpretable throughout depth, and related analyses trace how intermediate states anticipate the final prediction (Ferrando et al., 2023; Pal et al., 2023). The same literature documents the obstacle we start from. The directions that carry task-relevant information shift across layers rather than living in a single global subspace (Tenney et al., 2019; Ethayarajh, 2019; Belrose et al., 2023). The first finding says the trajectory is readable, the second says no fixed frame can read it, and our construction exists to resolve exactly this tension.

Inference-time intervention. A fourth line modifies the computation during decoding to reduce hallucination. Activation steering (Turner et al., 2023; Wang et al., 2025) adds a direction computed in advance to chosen layers, DoLa (Chuang et al., 2024) contrasts intermediate and final logits under a layer-selection rule fixed for the model, and related edits target hidden states in vision-language models (Su et al., 2025; Wu et al., 2025; Liu et al., 2025). In each case the intervention site is fixed globally, once per model or dataset, and a prescription written before the failure occurs cannot adapt to where the failure occurs. Our refinement instead inherits its target from detection, since the validator that flags a sample also scores every token at every block. Detection ends at an address.

3 DEPTHWISE TRAJECTORY SIGNATURES

We want to measure how a token's decision moves across depth, and the difficulty is that the instrument moves with it, the way a moving camera contaminates the motion of what it films. Depthwise measurement faces this problem three times over. The frame *translates*, since each boundary normalization carries a learned depth-dependent bias shared by every token. The frame *rotates*, since the directions carrying the current decision shift with depth, so fixed coordinates drift away from the signal. And the frame is *underdetermined*, since a fitted basis is identifiable only up to rotation within its span, so identical fits can yield different numbers. This section removes the three motions in turn, by bias centering (§3.1), per-window subspaces (§3.2), and orthogonal transport (§3.3), and then certifies the result once. Every signature is provably invariant to the arbitrary choices in the construction (§3.4), which is what makes trajectories comparable across depths, samples, and models. To our knowledge this is the first construction that makes depthwise trajectory measurement well-defined in this sense. On the invariant signatures, a validator detects hallucination (§3.5) and localizes the most anomalous event, which a single-block refinement operator acts on (§3.6).

3.1 MONITORING BOUNDARY AND BIAS-CENTERED STATES

Consider an autoregressive transformer on a token sequence $x_{1:n}$ with B blocks. For token position $t \in \{1, \dots, n\}$ and boundary index $b \in \{0, \dots, B\}$, let $h_{t,b} \in \mathbb{R}^d$ denote the residual stream state at a fixed inter-block boundary, and let $o_{t,b}, m_{t,b} \in \mathbb{R}^d$ denote block b's total attention and MLP contributions. The monitored update is $h_{t,b+1} = \mathcal{N}_{b+1}(h_{t,b} + o_{t,b} + m_{t,b})$, where $\mathcal{N}_{b+1}$ is the boundary normalization with learned affine parameters. The readout $W \in \mathbb{R}^{V \times d}$ produces logits $\ell_{t,b} = W h_{t,b} \in \mathbb{R}^V$, and we use W only for depth-local probe directions in §3.2. These raw states carry the first motion of the frame. $\mathcal{N}_b$ adds its learned shift $\beta_b \in \mathbb{R}^d$ to every token identically, and β_b changes with depth, so a step $b \to b+1$ displaces all tokens by the shared offset $\beta_{b+1} - \beta_b$ before any token moves on its own, an offset that dominates token-pooled summaries as the one component all tokens share. We remove it by monitoring the bias-centered state (Fig. 1**a**)

$$\tilde{h}_{t,b} \;=\; h_{t,b} - \beta_b, \tag{1}$$

which subtracts the frame's translation and leaves token-wise motion intact.

Centering removes the frame's translation, but norm-based comparison across depth additionally requires the states to live at a controlled scale, which the boundary normalization supplies.

Lemma 1 (Shell bound from boundary normalization)**.** *Assume the monitored boundary normalization admits the affine form*

$$\mathcal{N}_b(u) = \Gamma_b \, \mathcal{S}(u) + \beta_b, \qquad \Gamma_b = \mathrm{diag}(\gamma_{b,1}, \dots, \gamma_{b,d}), \tag{2}$$

with $0 < \gamma_{\min} \le \gamma_{b,i} \le \gamma_{\max}$ for all b, i, where $\mathcal{S}$ is the model's native normalization (LayerNorm or RMSNorm). Assume further that on the monitored region the normalization denominator stays away from degeneracy, so that $c_{\min}\sqrt{d} \le \|\mathcal{S}(u)\|_2 \le c_{\max}\sqrt{d}$ for $0 < c_{\min} \le c_{\max}$. For all (t, b),

$$\gamma_{\min} c_{\min} \sqrt{d} \;\le\; \|\tilde{h}_{t,b}\|_2 \;\le\; \gamma_{\max} c_{\max} \sqrt{d}. \tag{3}$$

This places every centered state in a spherical shell, so all norm-based quantities in §3.3 are well-scaled in $\sqrt{d}$. Proof and conditions on $c_{\min}, c_{\max}$ under LayerNorm and RMSNorm are in A.1.

3.2 COMPETITOR DIRECTIONS AND MOVING SUBSPACES

Centered states still leave the second motion of the frame unaddressed. The directions that carry the current decision shift with depth, so a single subspace fitted once measures early states in a frame tuned for late ones, or the reverse. We therefore fit a k-dimensional subspace per depth window, built from the readout rows of the top token and its close competitors, so that at every depth the coordinate axes span what the model is currently deciding between.

We index blocks by $b \in \{0, \dots, B-1\}$ and fix a window length $L \ge 1$ and stride $\sigma \ge 1$. For $j \ge 1$, let $b_j = (j-1)\sigma$ and $\mathcal{W}_j = \{b_j, \dots, b_j + L - 1\}$, with the last window truncated to end at $B-1$, and let a deterministic assignment $j(b)$ map each block to the latest window containing it, with closed

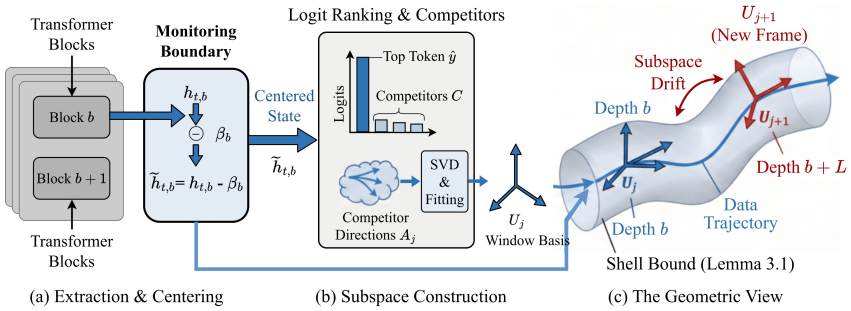


Figure 1: **Pipeline. (a)** States are read at a fixed boundary and the token-shared bias removed. **(b)** Readout directions of the top token and its competitors span a depth-local basis via SVD. **(c)** The trajectory is shell-bounded while window frames drift with depth.

form in B.1. Inside a window, the axes should separate the answers the model is weighing, and the readout rows provide them. With $\widehat{y}_{t,b}$ the top-ranked token under $\ell_{t,b}$ and $\mathcal{C}_{t,b}$ its top K competitors excluding it, we collect the difference directions (Fig. 1**b**)

$$\mathcal{A}_j = \Big\{ a_{t,b,y} = w_{\widehat{y}_{t,b}} - w_y \;:\; b \in \mathcal{W}_j,\; y \in \mathcal{C}_{t,b},\; M_t = 1 \Big\}, \tag{4}$$

where $M \in \{0,1\}^n$ is a token mask that excludes special and delimiter tokens. Each difference is the direction along which the state separates the top token from one competitor, so $\mathcal{A}_j$ collects the axes of the decisions active inside the window. Stacking sampled and unit-normalized directions from $\mathcal{A}_j$ as rows of $D_j \in \mathbb{R}^{N_j \times d}$ and taking the compact SVD $D_j = U_j^{(D)} \Sigma_j V_j^\top$, the window basis is the top-k right singular vectors

$$U_j = [v_{j,1}, \ldots, v_{j,k}] \in \mathbb{R}^{d \times k}, \qquad U_j^\top U_j = I_k, \tag{5}$$

with sampling, capping, and fallback details in B.3, and the moving coordinates are $p_{t,b} = U_{j(b)}^\top \tilde{h}_{t,b} \in \mathbb{R}^k$. The moving frame resolves the second motion but exposes the third. The SVD determines only the column span of U_j, so any rotation $U_j Q$ with orthogonal Q fits the same directions equally well yet assigns different coordinates to the same state, and a coordinate change across a window boundary then mixes genuine trajectory motion with an accident of the fit. The next subsection cancels this freedom.

3.3 Transported Trajectory Signatures

We cancel the remaining freedom at the window boundaries by aligning each basis with its neighbor, and on the aligned coordinates we record what a trajectory can do, namely move, turn, and watch its frame drift, each step attributed to the module that caused it. For adjacent windows we align U_{j+1} to U_j with the closest orthogonal map in the Frobenius sense (Fig. 2**a**) (Fernando et al., 2013),

$$U_{j+1}^\top U_j = \bar{P}_j \bar{\Sigma}_j \bar{V}_j^\top, \qquad R_{j \to j+1} = \bar{P}_j \bar{V}_j^\top \in \mathbb{R}^{k \times k}, \tag{6}$$

and the step-wise transport is $R_b = I_k$ when $j(b{+}1) = j(b)$ and $R_b = R_{j(b) \to j(b+1)}$ otherwise. The transported increment $\Delta p_{t,b} = p_{t,b+1} - R_b\, p_{t,b} \in \mathbb{R}^k$ is the change in coordinates after the frame change is accounted for, and with $u_{t,b} = p_{t,b}/(\|p_{t,b}\|_2 + \varepsilon_{\text{num}})$ the step size and turning angle are

$$s_{t,b} = \|\Delta p_{t,b}\|_2, \qquad \theta_{t,b} = \measuredangle\big(u_{t,b+1},\, R_b u_{t,b}\big). \tag{7}$$

One contamination remains at the step level. Within a fixed sample, all tokens can share a common displacement at depth b. We subtract a robust per-step center $\mu_b \in \mathbb{R}^k$ over masked tokens,

$$\Delta p^c_{t,b} = \Delta p_{t,b} - \mu_b, \qquad s^c_{t,b} = \|\Delta p^c_{t,b}\|_2, \tag{8}$$

so that $s^c_{t,b}$ measures how a token moves relative to its sample.

The frame's own movement is informative as well, since a window boundary where the decision axes reorient sharply is itself an event. With a deterministic anchor $b^\star_j = \max \mathcal{W}_j$ per window, we

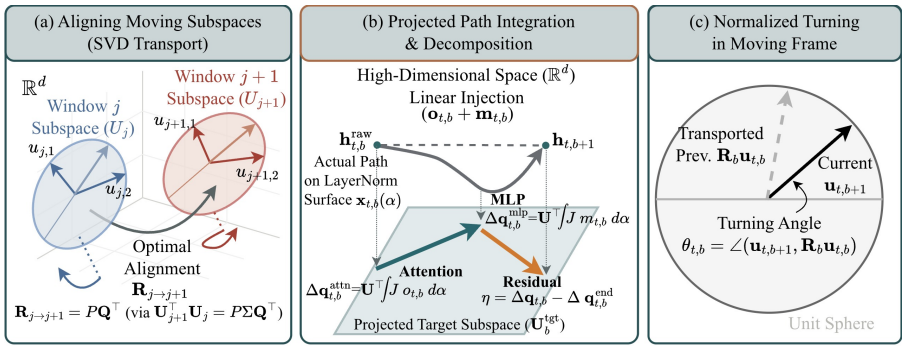


Figure 2: **Trajectory signatures from transported subspace coordinates. (a)** Adjacent bases are aligned by orthogonal transport $R_{j\to j+1}$ for consistent coordinates across boundaries. **(b)** Path-integrated updates project onto the target frame, splitting into attention and MLP contributions and a residual $\eta_{t,b}$. **(c)** Turning angle $\theta_{t,b}$ compares current and transported previous directions.

record the anchor-coupled drift

$$\chi_{t,j} = \frac{\|(U_{j+1}U_{j+1}^\top - U_jU_j^\top)\,\tilde{h}_{t,b_j^\star}\|_2}{\|\tilde{h}_{t,b_j^\star}\|_2 + \varepsilon_{\text{num}}}, \qquad D_t = \sum_{j=1}^{J-1} \chi_{t,j}, \tag{9}$$

the fraction of the anchor state displaced by the change of subspace. Both $\chi_{t,j}$ and the Grassmann drift $d_G(U_j, U_{j+1}) = \|U_jU_j^\top - U_{j+1}U_{j+1}^\top\|_2$ depend only on the projectors, hence on spans alone.

The signatures so far say that a step happened, not which module drove it, and the natural candidates are misleading to read off. The injections $o_{t,b}$ and $m_{t,b}$ enter before the boundary normalization, and $\mathcal{N}_{b+1}$ is nonlinear, so the displacement each module actually causes differs from the energy it injects. We recover the effective contribution by integrating the normalization's Jacobian J_{b+1} along the injection path $x_{t,b}(\alpha) = h_{t,b} + \alpha\,(o_{t,b} + m_{t,b})$, $\alpha \in [0,1]$ (Sundararajan et al., 2017), projected into the next window's frame $U_b^{\text{tgt}} = U_{j(b+1)}$,

$$\Delta q_{t,b}^{\text{attn}} = (U_b^{\text{tgt}})^\top \int_0^1 J_{b+1}\big(x_{t,b}(\alpha)\big)\, o_{t,b}\, d\alpha, \quad \Delta q_{t,b}^{\text{mlp}} = (U_b^{\text{tgt}})^\top \int_0^1 J_{b+1}\big(x_{t,b}(\alpha)\big)\, m_{t,b}\, d\alpha. \tag{10}$$

Alongside these we keep the pre-normalization magnitudes $a_{t,b} = \|(U_b^{\text{tgt}})^\top o_{t,b}\|_2$ and $m_{t,b}^{\text{mag}} = \|(U_b^{\text{tgt}})^\top m_{t,b}\|_2$, so that injected energy and effective displacement can disagree, and their disagreement is itself signal. Each $J_{b+1}(x)v$ is a Jacobian-vector product (Pearlmutter, 1994), and the integral uses a three-node Simpson rule, with the combined update $\Delta q_{t,b} = \Delta q_{t,b}^{\text{attn}} + \Delta q_{t,b}^{\text{mlp}}$ and magnitude $c_{t,b} = \|\Delta q_{t,b}\|_2$. Finally, a single-point linearization at the endpoint, $\Delta q_{t,b}^{\text{end}} = (U_b^{\text{tgt}})^\top J_{b+1}(x_{t,b}(1))\,(o_{t,b} + m_{t,b})$, agrees with the path integral only when the normalization acts linearly along the path. The residual $\eta_{t,b} = \Delta q_{t,b} - \Delta q_{t,b}^{\text{end}}$ and its ratio

$$\rho_{t,b} = \|\eta_{t,b}\|_2 \,/\, (\|\Delta q_{t,b}\|_2 + \varepsilon_{\text{num}}) \tag{11}$$

measure how strongly the step bends through the normalization, and perpendicular channel ratios $r_{t,b}^{\text{attn}}$ and $r_{t,b}^{\text{mlp}}$ attribute the bending to attention versus MLP after removing component along $u_{t,b+1}$.

3.4 Gauge Invariance of the Signatures

One arbitrary choice survives the three repairs. Each U_j is one representative of an equivalence class, since any rotation U_jQ_j spans the same subspace, and if the signatures changed under this choice they would report the accident of the fit rather than the trajectory. The following lemma rules this out for every signature we extract.

Lemma 2 (Gauge invariance of transported signatures)**.** *Let $\{U_j\}_j$ be window bases with orthonormal columns, and let $\{Q_j\}_j$ be any sequence of orthogonal matrices in $\mathbb{R}^{k\times k}$. Set $U_j' = U_jQ_j$*

and apply the same constructions under $\{U'_j\}$. Then $\|\Delta p_{t,b}\|_2$, $\angle(u_{t,b+1}, R_b u_{t,b})$, the projected magnitudes $a_{t,b}$, $m^{\mathrm{mag}}_{t,b}$, and $c_{t,b}$, the residual ratio $\rho_{t,b}$, the channel ratios $r^{\mathrm{attn}}_{t,b}$ and $r^{\mathrm{mlp}}_{t,b}$, and all projector-based drifts including d_G and $\chi_{t,j}$ are invariant under $\{U_j\} \mapsto \{U'_j\}$. The centered increment norm $\|\Delta p^c_{t,b}\|_2$ is invariant provided the center μ_b is rotation-equivariant.

Every quantity in the list is built from Euclidean norms, angles, or projectors, all preserved under basis rotation, with the full proof in A.2. Under the lemma's guarantee a signature value carries no imprint of which basis the SVD returned, which seed sampled the directions, or which sign convention the implementation follows, so equal values mean equal geometry, and comparison across depths, samples, and models becomes well-defined rather than assumed.

3.5 Self-checking via signature validation

The invariance guarantee licenses learning on the signatures. Events from different depths, samples, and prompts now live on a common scale, so a single model can consume them all. What that model should learn is the question left open since the introduction, namely which trajectory patterns accompany hallucination, and we let it answer from labels rather than deciding ourselves. For each sample n, token position t, and depth step b, we assemble the signatures of §3.3 into a ten-dimensional event vector, grouped by the question each group answers about the step,

$$x_{n,b,t} \;=\; \underbrace{(s_{t,b},\; s^c_{t,b},\; \theta_{t,b})}_{\text{motion}} \;\|\; \underbrace{(a_{t,b},\; m^{\mathrm{mag}}_{t,b},\; c_{t,b},\; \rho_{t,b})}_{\text{attribution}} \;\|\; \underbrace{(r^{\mathrm{attn}}_{t,b},\; r^{\mathrm{mlp}}_{t,b})}_{\text{ratio}} \;\|\; \underbrace{(D_t)}_{\text{drift}}, \tag{12}$$

where $\|$ denotes concatenation. Motion asks how far and how sharply the state moved, attribution asks which module moved it and at what effective strength, ratio asks how the bending splits between attention and MLP, and drift asks how much the frame itself reoriented. All ten entries are invariant by Lemma 2, and the four groups form the ablation partition in §4.3.

The validator g_ϕ reads the event grid in depth order through a feature-wise LayerNorm, a small MLP encoder, a single-layer GRU over the sequence of events, and mask-aware logsumexp pooling over valid positions, followed by a linear head that outputs

$$\widehat{y}_n \;=\; g_\phi\big(\{x_{n,b,t}\},\, M_n\big) \in [0,1], \tag{13}$$

where M_n is a validity mask from the tokenizer attention mask. The recurrence is not incidental. The same signature value can be routine at one depth and anomalous at another, and what the invariance guarantee equalizes is the scale of the features, not the meaning of their position, so the validator must read events in order rather than as a bag, which the GRU provides. We train with weighted cross-entropy on the signatures alone, so the base LLM receives no gradient and no modification.

3.6 Signature-guided single-block refinement

Detection as constructed is already localized. The validator scores individual events, one per token and depth, and pooling only compresses these scores into a sample-level flag, so the per-event scores remain available underneath every detection. Refinement uses them as an address, returning to the most anomalous event and repairing the trajectory there, touching nothing else. During generation, the validator emits per-event scores $\{z_i\}$ with validity indicators $\{I_i\}$, and we select the culprit as the highest-scoring valid event,

$$i^\star \in \arg\max_{i:\, I_i=1} z_i, \quad (t_0, b_0) = (t_{i^\star},\, b_{i^\star}), \tag{14}$$

roll generation back to the prefix $x_{1:t_0}$, and regenerate the suffix with the base model unchanged at every depth except b_0. The decode budget is preserved, since at most $T - t_0$ new tokens are produced, so the refined answer costs no more length than the original.

The intervention must correct the culprit block without erasing what that block correctly computes, so we rescale rather than overwrite, and only within the decision subspace. With $h^{\mathrm{in}}, h^{\mathrm{out}} \in \mathbb{R}^d$ the block input and output at the current decoding step, we fit a local readout-aligned basis $U \in \mathbb{R}^{d\times k}$ from the current top token and its K competitors, keep it fixed throughout suffix regeneration, and project

$$p^{\mathrm{in}} = U^\top h^{\mathrm{in}}, \qquad p^{\mathrm{out}} = U^\top h^{\mathrm{out}}, \qquad \Delta p = p^{\mathrm{out}} - p^{\mathrm{in}} \in \mathbb{R}^k. \tag{15}$$

A calibration pass over the prefix yields $s_{\mathrm{ref}} > 0$ as the masked median of $\|\Delta p\|_2$ over valid prefix steps, so the scale of a normal step is set by this prompt and this model rather than by a global constant. Given a clamp ratio $\kappa > 1$, we shrink the step only when it exceeds the calibrated band, and we rewrite only the U-component:

$$\lambda = \min\left(1,\ \frac{\kappa\, s_{\mathrm{ref}}}{\|\Delta p\|_2 + \varepsilon_{\mathrm{num}}}\right), \qquad h^{\mathrm{out}\prime} = h^{\mathrm{out}} + U\left((\lambda - 1)\,\Delta p\right). \tag{16}$$

The component of h^{out} orthogonal to U passes untouched, so everything the block computes outside the current decision, syntax, style, and content for later tokens, is preserved, and other depths remain the base model. The general form with transport $R \neq I_k$, with an illustration, is in D.2.

4 Experiments

We evaluate on twelve benchmarks in four categories, *QA* (SQuAD (Rajpurkar et al., 2016), CoQA (Reddy et al., 2019), TriviaQA (Joshi et al., 2017), NQ-Open (Kwiatkowski et al., 2019), TruthfulQA (Lin et al., 2022b), HaluEval QA (Li et al., 2023)), *general-domain factuality* (HaluEval General, FELM (Zhao et al., 2023)), *summarization* (xSUM (Narayan et al., 2018), SAMSum (Gliwa et al., 2019)), and *reasoning* (GSM8K (Cobbe et al., 2021), BBH (Suzgun et al., 2023)), where the reasoning pair deliberately probes the boundary drawn in the introduction. We use five open-weight base LLMs, Qwen2.5-7B (Hui et al., 2024), Gemma2-9B (Team et al., 2024), Phi-3-mini (Abdin et al., 2024), LLaMA3-8B (Grattafiori et al., 2024), and Mistral-7B (Jiang et al., 2023), spanning four architecture families, used as released, instruction-tuned, without further fine-tuning. Signatures are extracted in a teacher-forced replay (Williams & Zipser, 1989) of each greedy generation (§3.1), and per dataset and model pair we train an independent validator and report AUROC.

Labels and baselines. Binary labels come from a frozen ChatGPT-4o-mini (OpenAI, 2025) judge with a fixed deterministic prompt, which carries a label-noise floor (§5), but every method is scored against the same labels on the same outputs, so judge bias shifts all methods together. The residual asymmetry runs against us, since the supervised detectors, ours included, also train on the noisy labels that the score-based baselines only face at evaluation, so label noise cannot manufacture our advantage. We compare against five baselines spanning both families of §2, SelfCheckGPT (Manakul et al., 2023), Semantic Entropy (Kuhn et al., 2023; Farquhar et al., 2024), INSIDE (Chen et al., 2024), LLM-Check (Sriramanan et al., 2024), and HalluShift (Dasgupta et al., 2025).

4.1 Hallucination detection

The first question is whether the hypothesized trace exists at all. Table 1 answers affirmatively. Our validator attains the best AUROC in 45 of 60 model and dataset cells, with a mean advantage of $+3.32$pp over the strongest baseline per cell and an overall mean of 65.68% against 58.32% for the best single baseline. The pattern of where the advantage concentrates is as informative as the count. **QA.** We win 22 of 30 cells at $+2.41$pp mean advantage, with the largest margins where hallucination takes the form of a discrete factual error, up to $+18.64$pp on TriviaQA. On CoQA, NQ-Open, and TruthfulQA every method compresses into the 50 to 65 AUROC band, a pattern consistent with reference mismatches that no detector separates cleanly, and we revisit this floor in §5. **General-domain factuality.** We win 6 of 10 cells at $+5.15$pp, the largest per-category advantage, taking all five models on HaluEval General, four of them by $+13$pp or more. FELM is mixed, and Gemma2 with FELM is our weakest cell at -9.64pp. **Summarization.** We win 9 of 10 cells at $+4.36$pp, with the clearest gap on LLaMA3 with xSUM at $+16.20$pp over the best baseline. **Reasoning.** GSM8K and BBH probe the boundary set up in the introduction. We still win 8 of 10 cells at $+3.17$pp, with both losses on GSM8K (by up to 10.52pp) and all five BBH models won by $+2.76$ to $+15.32$pp.

4.2 Signature-guided refinement

The second question is whether localization pays. Table 2 reports hallucination ratios before and after refinement across nine benchmarks and five base LLMs, where refinement runs on every sample, intervenes whenever the validator predicts hallucination, and both ratios are computed over the full evaluation set. Of 45 cells, 29 show $\geq 10\%$ relative reduction (bold), with a mean reduction of 17.9% and a median of 14.5%, and where the intervention works, it works for the reason the method

Table 1: **Hallucination detection AUROC (%) across benchmarks and base LLMs. Bold** marks the best method and underline the second best in each column within a model block.

		QA						General		Summ.		Reasoning	
LLM	**Method**	SQuAD	CoQA	TrivQA	NQ	TruthQA	HE-QA	HE-Gen	FELM	xSUM	SAMSum	GSM8K	BBH
Qwen2.5	SelfCheckGPT	61.95	61.02	55.72	62.86	53.42	55.45	50.44	58.85	57.37	57.39	44.83	56.10
	Sem. Entropy	53.25	47.06	45.79	62.28	52.16	50.05	52.50	54.35	49.80	48.46	50.82	55.40
	INSIDE	71.36	61.95	68.82	60.35	54.63	62.40	45.49	47.31	48.72	52.40	59.86	49.64
	LLM-Check	72.92	60.57	74.16	49.87	45.47	59.08	41.31	**65.34**	41.51	46.32	49.67	43.50
	HalluShift	72.12	63.01	76.17	**67.88**	55.55	59.17	37.64	62.61	53.74	47.78	51.46	47.70
	Trajectory (**Ours**)	**73.68**	**67.90**	**82.85**	66.60	**58.13**	**76.45**	**69.80**	62.83	**64.87**	**57.59**	**69.48**	**60.45**
Gemma2	SelfCheckGPT	72.14	**68.05**	85.45	**73.55**	61.80	61.23	42.23	63.95	55.85	59.50	63.85	48.27
	Sem. Entropy	72.80	63.68	86.32	72.78	**70.74**	65.65	47.47	53.31	48.62	51.19	68.14	54.48
	INSIDE	63.47	67.36	77.03	66.88	70.12	54.08	50.13	57.94	47.01	57.77	**77.78**	60.61
	LLM-Check	67.88	60.79	78.26	51.90	49.59	59.57	47.32	**64.62**	42.42	50.52	54.43	63.23
	HalluShift	60.17	60.10	61.30	68.08	57.49	61.45	41.41	64.29	45.12	55.87	73.27	62.39
	Trajectory (**Ours**)	**73.18**	59.28	**86.71**	64.55	55.58	**66.70**	**65.17**	54.98	**57.43**	**63.49**	67.26	**73.08**
Phi-3	SelfCheckGPT	46.48	53.83	60.71	62.77	52.48	50.29	51.96	**62.07**	59.00	49.77	**75.93**	49.85
	Sem. Entropy	50.37	50.48	53.39	41.66	51.91	46.52	52.50	58.43	49.33	50.79	55.66	54.61
	INSIDE	56.26	52.78	59.79	51.47	39.34	50.26	51.02	54.51	44.80	51.52	71.55	46.87
	LLM-Check	61.27	54.83	42.52	50.29	48.68	49.92	35.84	50.24	38.75	**58.21**	58.05	51.69
	HalluShift	56.13	57.88	64.13	61.23	51.05	57.02	37.09	61.89	53.82	54.21	66.33	54.49
	Trajectory (**Ours**)	**62.86**	**58.27**	**70.86**	**64.98**	**61.54**	**67.57**	**65.79**	59.75	**61.01**	57.76	69.44	**69.93**
LLaMA3	SelfCheckGPT	55.79	59.74	59.60	58.30	56.49	45.79	59.19	65.41	52.08	59.93	46.10	51.00
	Sem. Entropy	58.78	51.55	60.00	48.54	47.67	52.00	47.64	46.32	49.42	43.34	47.00	52.84
	INSIDE	60.24	57.98	61.31	60.24	52.21	51.00	50.22	61.00	44.10	58.50	52.82	46.87
	LLM-Check	45.41	54.14	40.22	60.51	54.29	50.10	57.28	63.01	45.35	57.63	39.79	53.50
	HalluShift	51.18	56.71	55.52	63.61	**62.33**	55.54	47.10	57.05	52.76	59.62	63.94	49.82
	Trajectory (**Ours**)	**74.38**	**65.77**	**79.95**	**63.81**	59.57	**65.65**	**67.07**	**66.17**	**68.96**	**65.28**	**67.21**	**56.53**
Mistral	SelfCheckGPT	67.61	**65.67**	67.62	61.53	58.96	58.97	47.20	**64.25**	59.00	53.46	67.44	51.87
	Sem. Entropy	63.93	60.55	61.15	53.40	53.19	54.28	53.99	55.96	49.30	51.93	61.64	51.75
	INSIDE	55.99	54.87	54.08	58.37	63.78	55.61	53.85	54.21	51.53	53.80	67.41	49.38
	LLM-Check	60.05	45.90	47.74	50.45	46.39	50.36	37.27	61.26	50.66	49.93	46.58	55.20
	HalluShift	71.71	62.54	66.68	**63.86**	**65.62**	54.40	36.18	61.78	44.01	50.87	65.96	59.48
	Trajectory (**Ours**)	**74.63**	61.54	**70.50**	59.09	64.77	**61.67**	**67.01**	62.94	**62.24**	**57.81**	**67.91**	**62.24**

predicts. On HaluEval QA all five models reduce by $\geq 10\%$, peaking at -48.94% on LLaMA3, and FELM holds the largest reductions in the table, up to -68.74% on Qwen2.5, settings where a single clamped step suffices because the failure is concentrated. The contrast arrives on reasoning, where only 4 of 10 cells reach $\geq 10\%$ and Gemma2 with BBH does not move at all ($34.39\% \rightarrow 34.39\%$), since correcting one link of a propagated error leaves the downstream steps in place, exactly the boundary the introduction drew. Summarization sits between the two regimes at a median of 13.2%, and across base LLMs Gemma2 benefits least at a mean reduction of 13.4%, mirroring its detection.

Table 2: **Hallucination ratio before and after signature-guided refinement.** For each base LLM and dataset cell we report the initial ratio, the ratio after refinement on samples flagged by the validator, and the relative reduction (↓ %). **Bold** marks cells with $\geq 10\%$ relative reduction.

Category	**Dataset**	Qwen2.5	Gemma2	Phi-3	LLaMA3	Mistral
QA	TriviaQA	35.48 / 33.73 (↓ 4.93)	**33.21 / 27.50 (↓17.19)**	**30.80 / 20.50 (↓33.44)**	**33.75 / 24.50 (↓27.41)**	**27.25 / 22.25 (↓18.35)**
	NQ-Open	**65.32 / 53.75 (↓17.71)**	65.25 / 61.50 (↓ 5.75)	**78.70 / 67.26 (↓14.54)**	59.50 / 55.25 (↓ 7.14)	59.50 / 54.75 (↓ 7.98)
	HaluEval QA	**15.25 / 10.95 (↓28.20)**	**14.90 / 12.34 (↓17.18)**	**12.65 / 6.55 (↓48.22)**	**15.10 / 7.71 (↓48.94)**	**14.05 / 11.70 (↓16.73)**
General	HaluEval General	**11.96 / 8.75 (↓26.84)**	**6.76 / 5.76 (↓14.79)**	**7.64 / 6.31 (↓17.41)**	**13.07 / 11.74 (↓10.18)**	10.85 / 10.19 (↓ 6.08)
	FELM	**37.43 / 11.70 (↓68.74)**	32.24 / 30.92 (↓ 4.09)	**31.74 / 13.45 (↓57.62)**	**35.56 / 31.58 (↓11.19)**	**37.08 / 16.96 (↓54.26)**
Summ.	xSUM	**25.59 / 21.67 (↓15.32)**	**28.54 / 24.12 (↓15.49)**	**20.33 / 17.83 (↓12.30)**	24.21 / 22.33 (↓ 7.77)	**24.83 / 21.33 (↓14.10)**
	SAMSum	21.74 / 21.43 (↓ 1.43)	**10.25 / 6.52 (↓36.39)**	16.15 / 14.60 (↓ 9.60)	**22.05 / 18.83 (↓14.60)**	16.15 / 15.84 (↓ 1.92)
Reasoning	GSM8K	15.21 / 14.65 (↓ 3.68)	**7.60 / 6.83 (↓10.13)**	12.17 / 11.03 (↓ 9.37)	19.77 / 19.39 (↓ 1.92)	**47.91 / 34.60 (↓27.78)**
	BBH	21.40 / 20.03 (↓ 6.40)	34.39 / 34.39 (↓ 0.00)	**27.37 / 22.11 (↓19.22)**	46.67 / 45.61 (↓ 2.27)	**58.25 / 51.23 (↓12.05)**

4.3 Feature group ablation

The introduction hypothesized a detectable trace without fixing its form, and Eq. 12 accordingly asks four separate questions about each step. Table 3 tests whether all four are needed, retraining the validator on HaluEval QA with each group removed. No removal ever improves over the full validator, so none of the four questions is redundant. The more telling result is which question matters most, because the answer changes with the model. Ratio dominates on Qwen2.5 ($\Delta -7.3$), Gemma2 ($\Delta -11.9$), and LLaMA3 ($\Delta -3.0$), attribution on Phi-3 ($\Delta -11.0$), and motion on Mistral ($\Delta -3.0$). The extremes are instructive. Gemma2 collapses to 54.8 AUROC without ratio, its trace concentrated in how the bending splits, while Phi-3 degrades near-uniformly under any removal (Δ between -10.0 and -11.0), its trace spread across all four questions. This is the empirical form

of the claim made in the introduction, that the discriminative pattern differs across model families, so any single fixed statistic inherits the model dependence, consistent with single-statistic baselines ranking differently across models in Table 1.

Table 3: **Feature group ablation on HaluEval QA (AUROC, %).** Each column reports AUROC when the indicated group is removed. Δ shows the drop relative to Full. **Bold** marks the largest drop per model.

Base LLM	w/o Motion	w/o Attrib.	w/o Drift	w/o Ratio	Full
Qwen2.5	71.5 (Δ−5.0)	69.4 (Δ−7.1)	73.3 (Δ−3.2)	**69.2** (Δ−7.3)	76.5
Gemma2	63.1 (Δ−3.6)	60.6 (Δ−6.1)	64.7 (Δ−2.0)	**54.8** (Δ−11.9)	66.7
Phi-3	57.6 (Δ−10.0)	**56.6** (Δ−11.0)	57.1 (Δ−10.5)	57.4 (Δ−10.2)	67.6
LLaMA3	64.0 (Δ−1.7)	63.7 (Δ−2.0)	63.1 (Δ−2.6)	**62.7** (Δ−3.0)	65.7
Mistral	**58.7** (Δ−3.0)	58.8 (Δ−2.9)	59.5 (Δ−2.2)	60.5 (Δ−1.2)	61.7

Table 4: **Wall-clock overhead per decoding step (ms). OH** is the fraction of inference time spent on signature extraction.

LLM	Fwd	Sig	L+SVD	OH%
Phi-3	496	71	163	32
Gemma2	734	86	151	24
Qwen2.5	742	62	153	**22**
LLaMA3	724	72	186	26
Mistral	745	71	192	26

4.4 Runtime cost and scalability

The remaining question is what the signal costs. Table 4 reports per-step wall-clock timing, separating the base forward pass (Fwd), signature extraction (Sig), and the per-window lens and SVD fitting (L+SVD). Extraction accounts for 22 to 32% of total per-step inference time, an absolute 215 to 263 ms per step combining signature computation and per-window fitting, and the cost scales linearly in sequence length at $\mathcal{O}(B(k+K)d)$ per token. This always-on price of detection is joined by a conditional one, paid only when the validator flags a sample. With the state cached at the rollback position t_0, refinement regenerates the discarded suffix at an overhead factor of $1 + (t_{\text{cur}} - t_0)/T \leq 2$, so a refined generation never exceeds twice the base decode. For comparison, sampling-based detection alone requires 5 to 10 additional complete generations per query, so detection and repair together cost less here than detection by itself costs there.

5 Discussion

The experiments draw three lessons. **First, depth-localization is a scope, not a universal claim.** On reasoning, detection still wins 8 of 10 cells but refinement helps in only 4 of 10, so the trace of a distributed error remains visible while no single block suffices to repair it, confirming the boundary drawn in the introduction. **Second, performance depends on the base model's internal dynamics.** The invariance guarantees hold for every model we test, yet Gemma2 marks the low end on both halves of the method (§4.2). Phi-3 spreads its trace across all four questions and survives any single removal, while Gemma2 concentrates it in one, losing 11.9 AUROC from the ratio group alone (§4.3), so what varies across models is not whether the trajectory carries signal but how redundantly it is spread, which is why model adaptation sits in the validator while the signatures stay fixed. **Third, the evaluation signal has a floor.** Labels come from a frozen automatic judge, so no method can exceed agreement with an imperfect referee, and the benchmarks where every method compresses toward chance, TruthfulQA, CoQA, and NQ-Open, point at the labels rather than at any detector. Sharper conclusions there require better referees, and we leave this to future work.

6 Conclusion

Given a clamp ratio $\kappa > 1$, we shrink the step only when it exceeds the calibrated band, rewriting only the U-component Doing so required making depthwise measurement well-defined in the first place, and the three repairs of §3, centering, moving subspaces, and transport, remove the three motions of the measuring frame, so that what the signatures record is the decision and not the instrument. On these signatures a validator below 1M parameters attains the best AUROC in 45 of 60 model and dataset cells against five established baselines, and because the same scores that flag a sample also localize its most anomalous event, a single-block refinement reduces the hallucination ratio by $\geq 10\%$ in 29 of 45 cells at a cost bounded by twice the base decode. Within its boundary, drawn in §5, reading how a model decided provides signal that reading what it said does not. The validator already scores every candidate depth, so extending the single-block operator to multi-site intervention is the natural next step, and more broadly the trajectory itself, now measurable in a well-defined way, becomes an object of study in its own right.

AI USE STATEMENT

In this work, we used generative AI tools for editing assistance, including grammar and typo detection, and for improving the clarity of author-written text. We did not use generative AI tools for research ideation, for the design of the method or of the experiments, for any mathematical formulation or proof, for the implementation of the pipeline, or for the generation, selection, analysis, or interpretation of experimental results, and no other task in the required disclosure list applies to this work. Retrieval-augmented searches were not used to source related work, and all cited references were identified and verified manually by the authors. All AI-assisted edits were reviewed by the authors against the intended technical content, and all claims, equations, and numbers in the paper were written and checked by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies the detection and reduction of hallucinations in large language models, which we expect to improve the reliability of deployed systems. The study uses publicly available benchmarks and open-weight models, involves no human subjects and no personally identifiable data, and releases no new dataset. A detector of the kind we propose could in principle be inverted to search for generations that evade it, but the validator is trained per model on internal states, so evasion would require white-box access to the monitored deployment, in which case stronger attacks are already available. We see no ethical concerns beyond those common to work on language model reliability.

REPRODUCIBILITY STATEMENT

The construction of the signatures is fully specified in §3, with proofs of both lemmas in A and all deterministic conventions, fallbacks, and stabilization rules in B. All hyperparameters for extraction, the validator architecture, and training are listed in C, together with compute requirements. The evaluation protocols, including the judge configuration and the exact refinement trigger and denominator conventions, are specified in D. Source code for signature extraction, validator training, and refinement is provided as supplementary material.

REFERENCES

Marah Abdin, Jyoti Aneja, et al. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

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

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

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. Inside: Llms' internal states retain the power of hallucination detection. In *International Conference on Learning Representations*, volume 2024, pp. 3056–3076, 2024.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *International Conference on Learning Representations*, volume 2024, pp. 54158–54183, 2024.

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

Sharanya Dasgupta, Sujoy Nath, Arkaprabha Basu, Pourya Shamsolmoali, and Swagatam Das. Hallushift: Measuring distribution shifts towards hallucination detection in llms. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2025.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 55–65, 2019.

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

Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE international conference on computer vision*, pp. 2960–2967, 2013.

Javier Ferrando, Gerard I Gállego, Ioannis Tsiamas, and Marta R Costa-Jussà. Explaining how transformers use context to build predictions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5486–5513, 2023.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pp. 70–79, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM transactions on information systems*, 43(2):1–55, 2025.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. Llm internal states reveal hallucination risk faced with a query. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 88–104, 2024.

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

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

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

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 6449–6464, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 3214–3252, 2022a.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 3214–3252, 2022b.

Sheng Liu, Haotian Ye, and James Y Zou. Reducing hallucinations in large vision-language models via latent space steering. In *International Conference on Learning Representations*, volume 2025, pp. 72402–72419, 2025.

Potsawee Manakul, Adian Liusie, and Mark Gales. 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*, pp. 9004–9017, 2023.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 1906–1919, 2020.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 1797–1807, 2018.

nostalgebraist. Interpreting GPT: The logit lens. LessWrong, August 2020. URL https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

OpenAI. Chatgpt 4o-mini. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, 2025. Used as automatic evaluator for multimodal hallucination analysis.

Koyena Pal, Jiuding Sun, Andrew Yuan, Byron C Wallace, and David Bau. Future lens: Anticipating subsequent tokens from a single hidden state. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pp. 548–560, 2023.

Barak A Pearlmutter. Fast exact multiplication by the hessian. *Neural computation*, 6(1):147–160, 1994.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 2383–2392, 2016.

Siva Reddy, Danqi Chen, and Christopher D Manning. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. Llm-check: Investigating detection of hallucinations in large language models. *Advances in neural information processing systems*, 37:34188–34216, 2024.

Jingran Su, Jingfan Chen, Hongxin Li, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. Activation steering decoding: Mitigating hallucination in large vision-language models through bidirectional hidden state intervention. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12964–12974, 2025.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051, 2023.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4593–4601, 2019.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Tianlong Wang, Xianfeng Jiao, Yinghao Zhu, Zhongzhi Chen, Yifan He, Xu Chu, Junyi Gao, Yasha Wang, and Liantao Ma. Adaptive activation steering: A tuning-free llm truthfulness improvement method for diverse hallucinations categories. In *Proceedings of the ACM on Web Conference 2025*, pp. 2562–2578, 2025.

Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

Junfei Wu, Yue Ding, Guofan Liu, Tianze Xia, Ziyue Huang, Dianbo Sui, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Sharp: Steering hallucination in lvlms via representation engineering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 14357–14372, 2025.

Yiran Zhao, Jinghan Zhang, I Chern, Siyang Gao, Pengfei Liu, Junxian He, et al. Felm: Benchmarking factuality evaluation of large language models. *Advances in Neural Information Processing Systems*, 36:44502–44523, 2023.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

A PROOFS

A.1 PROOF OF LEMMA 1

We first make the norm band on $\mathcal{S}$ explicit. For $u \in \mathbb{R}^d$ and a fixed $\varepsilon > 0$, LayerNorm and RMSNorm take the forms

$$\mathcal{S}_{\mathrm{LN}}(u) = \frac{u - \mu(u)\mathbf{1}}{\sqrt{s_2(u) + \varepsilon}}, \qquad \mathcal{S}_{\mathrm{RMS}}(u) = \frac{u}{\sqrt{m_2(u) + \varepsilon}},$$

where $\mu(u) = \frac{1}{d}\sum_i u_i$, $s_2(u) = \frac{1}{d}\sum_i (u_i - \mu(u))^2$, and $m_2(u) = \frac{1}{d}\sum_i u_i^2$. Summing squares gives the exact identities

$$\|\mathcal{S}_{\mathrm{LN}}(u)\|_2^2 = \frac{d\, s_2(u)}{s_2(u) + \varepsilon}, \qquad \|\mathcal{S}_{\mathrm{RMS}}(u)\|_2^2 = \frac{d\, m_2(u)}{m_2(u) + \varepsilon}. \tag{17}$$

Let $q(u)$ denote $s_2(u)$ in the LayerNorm case and $m_2(u)$ in the RMSNorm case. Since $x \mapsto x/(x + \varepsilon)$ is increasing on $(0, \infty)$, any two-sided moment condition $0 < v_{\min} \le q(u) \le v_{\max}$ on the monitored region yields the norm band assumed in Lemma 1,

$$c_{\min}\sqrt{d} \le \|\mathcal{S}(u)\|_2 \le c_{\max}\sqrt{d}, \qquad c_{\min} = \sqrt{\tfrac{v_{\min}}{v_{\min}+\varepsilon}}, \quad c_{\max} = \sqrt{\tfrac{v_{\max}}{v_{\max}+\varepsilon}} \le 1. \tag{18}$$

These are the sufficient conditions referenced in the main text: the band holds on any set of states whose per-token second moment stays in $[v_{\min}, v_{\max}]$, the upper constant never exceeds 1, and the lower constant is controlled by how far the moment stays from 0 relative to ε. Here ε is the built-in constant of the pretrained normalization module, distinct from the pipeline safeguard $\varepsilon_{\mathrm{num}}$ of B.4.

The lemma now follows directly. Writing $u_{t,b}$ for the pre-normalization input at boundary b, Eq. 1 and Eq. 2 give $\tilde{h}_{t,b} = \Gamma_b\, \mathcal{S}(u_{t,b})$. Since Γ_b is diagonal with entries in $[\gamma_{\min}, \gamma_{\max}]$, multiplying the band in Eq. 18 by these extremes gives

$$\gamma_{\min} c_{\min}\sqrt{d} \;\le\; \|\tilde{h}_{t,b}\|_2 \;\le\; \gamma_{\max} c_{\max}\sqrt{d},$$

which is Eq. 3. □

A.2 PROOF OF LEMMA 2

Fix a window-wise basis change $U'_j = U_j Q_j$ with orthogonal $Q_j \in \mathbb{R}^{k\times k}$. For adjacent windows, write the compact SVD $U_{j+1}^\top U_j = \bar{P}_j \bar{\Sigma}_j \bar{V}_j^\top$ and $R_{j\to j+1} = \bar{P}_j \bar{V}_j^\top$. Note first that $R_{j\to j+1}$ is the orthogonal polar factor of $U_{j+1}^\top U_j$, which is unique whenever this matrix is nonsingular, and hence independent of which SVD the implementation returns. The nonsingular case is the operative one: when the smallest singular value vanishes or nearly vanishes, the transport is replaced by the identity under the stabilization rule of B.3, which is manifestly basis-independent. Under the basis change,

$$(U'_{j+1})^\top U'_j = Q_{j+1}^\top (U_{j+1}^\top U_j)\, Q_j,$$

and by uniqueness the polar factor transforms as

$$R'_{j\to j+1} = Q_{j+1}^\top R_{j\to j+1} Q_j.$$

Therefore the moving coordinates rotate as

$$p'_{t,b} = (U'_{j(b)})^\top \tilde{h}_{t,b} = Q_{j(b)}^\top p_{t,b}, \qquad \Delta p'_{t,b} = Q_{j(b+1)}^\top \Delta p_{t,b}.$$

Since $Q_{j(b+1)}$ is orthogonal, it preserves Euclidean norms and angles. This proves invariance of the transported step length $\|\Delta p_{t,b}\|_2$ and the turning angle $\angle(u_{t,b+1}, R_b u_{t,b})$. For the centered increment, a rotation-equivariant center satisfies $\mu'_b = Q_{j(b+1)}^\top \mu_b$, so $\Delta p^{c\prime}_{t,b} = Q_{j(b+1)}^\top \Delta p^c_{t,b}$ and $\|\Delta p^c_{t,b}\|_2$ is preserved, which is the conditional clause of the lemma. Perpendicular ratios are invariant because the target direction and all target-frame projections undergo the same orthogonal rotation, preserving inner products and perpendicular norms. The projected magnitudes $a_{t,b}$, $m^{\mathrm{mag}}_{t,b}$, and $c_{t,b}$ are Euclidean norms of target-frame projections, hence preserved under the same rotation. Projector-based drifts are invariant since $U'_j (U'_j)^\top = U_j U_j^\top$. The residual ratio $\rho_{t,b}$ is invariant because $\Delta q_{t,b}$ and $\Delta q^{\mathrm{end}}_{t,b}$ share the same target-frame rotation. □

B Construction Details

B.1 Depth windows and the assignment $j(b)$

Fix $L \geq 1$ and stride $\sigma \geq 1$. Window starts sit on the stride grid, $b_j = (j-1)\sigma$, with ends $e_j = \min\{b_j + L - 1,\ B - 1\}$, so the final window is truncated to end at block $B-1$. The number of windows is $J = \lceil (B-L)/\sigma \rceil + 1$ when $B > L$ and $J = 1$ otherwise, the smallest count whose last window reaches $B-1$. When $\sigma < L$ a block can belong to several windows, and each block is assigned to the latest window containing it, with closed form

$$j(b) = \min\Big\{J,\ \Big\lfloor \tfrac{b}{\sigma} \Big\rfloor + 1\Big\},$$

where the outer $\min$ routes the trailing blocks into the truncated final window. A window switch occurs at step $b \to b+1$ exactly when $j(b+1) \neq j(b)$, which is the condition under which the transport of Eq. 6 is applied.

B.2 Competitors and token masks

Ranking and ties. Logits $\ell_{t,b} = W h_{t,b}$ are used as-is, without temperature or post-processing. The top token is the argmax with ties broken toward the smallest index, and the competitor set $\mathcal{C}_{t,b}$ consists of the next K indices in the induced non-increasing order, with the same tie rule. If $K + 1 > V$, all remaining indices are taken.

Masks. Two binary masks govern eligibility. The tokenizer attention mask marks token validity and has strict priority: positions it invalidates are excluded everywhere. An eligibility mask additionally excludes tokenizer special tokens and model-specific role or delimiter tokens, and optionally restricts to a contiguous span from the first content token after the last delimiter to the last generated content token before any stop token. All direction collection, pooling, and centering operate only on positions valid under both masks. A sample whose eligible set is empty contributes no directions and falls to the fitting fallback below.

B.3 Subspace fitting, fallbacks, and transport stabilization

Normalization and capping. Each collected direction is unit-normalized as $a/(\|a\|_2 + \varepsilon_a)$, and directions with $\|a\|_2$ below a threshold on the same numerical scale as ε_a are discarded. If the pool for window j exceeds a fixed cap, a subset of the cap size is sampled without replacement under a deterministic seed $\mathrm{hash}(\text{sample id},\ j,\ \mathrm{seed}_0)$, so runs are reproducible.

SVD and re-orthonormalization. The basis is the top-k right singular vectors of the stacked directions, followed by a thin QR so that $U_j^\top U_j = I_k$ to numerical precision.

Degenerate and rank-deficient windows. If a window collects no usable direction, its basis falls back to the first k columns of a fixed reference orthonormal matrix G (we use $G = I_d$). If the rank r satisfies $0 < r < k$, the r extracted vectors are completed by projecting G onto their orthogonal complement, taking the first $k - r$ directions, and re-orthonormalizing by thin QR. Both rules are deterministic. The completed directions carry no readout alignment, so windows relying on the fallback contribute correspondingly weaker decision-specific signal, while remaining covered by the invariance guarantee.

Anchor. The drift anchor of Eq. 9 is the window end, $b_j^\star = \max \mathcal{W}_j$, so the anchor reflects the subspace used throughout the window.

Weak-overlap transport stabilization. When adjacent subspaces overlap weakly, small singular values of $U_{j+1}^\top U_j$ make the Procrustes factor unstable, and at exact rank deficiency the polar factor is no longer unique. With a fixed threshold $\tau_\sigma \in (0, 1)$, the transport is replaced by the identity whenever the smallest singular value falls below τ_σ, treating the switch as a frame reset whose discontinuity is then captured by the drift signatures.

B.4 Centering, aggregation, and numerical safeguards

Centering. The per-step center μ_b of Eq. 8 is the coordinate-wise median of $\{\Delta p_{t,b}\}$ over tokens valid under both masks, with $\mu_b = 0$ when the eligible set is empty. The coordinate-wise median is

not rotation-equivariant, so $\|\Delta p^c_{t,b}\|_2$ is not strictly covered by Lemma 2. Substituting the Euclidean mean, which is equivariant, changes AUROC by less than 0.3% on all benchmarks.

Depth aggregation. Per-step ratio features are reduced to token-level summaries by a masked median over valid depth steps within the effective monitored range, with summary 0 when no valid step exists.

Numerical safeguards. A single constant $\varepsilon_{\text{num}} > 0$ appears in every norm denominator of the pipeline and the validator features. The boundary normalization uses the pretrained module of the base model, LayerNorm or RMSNorm by architecture, with its built-in epsilon taken from the pretrained configuration, and the band conditions of A.1 refer to this module.

C Hyperparameters and Training

C.1 Signature extraction

Unless noted otherwise, the moving subspace uses window length $L = 8$ and stride $\sigma = 4$. Competitor collection uses $K = 32$ competitors, and the fitted basis has dimension $k = 16$, in both detection and refinement. Path-integrated updates evaluate the Jacobian of the boundary normalization through Jacobian-vector products via automatic differentiation, without forming explicit Jacobians, and the integral over $\alpha \in [0, 1]$ uses a fixed three-node Simpson rule with nodes $\{0,\ 0.5,\ 1\}$. A single random seed per run is applied consistently to Python, NumPy, and PyTorch.

C.2 Validator

Each sample yields a depthwise feature grid over monitored depth steps, token positions, and the ten signature features, masked by token validity. Within a minibatch the grid is padded to a common shape and linearized into a single event axis in depth-major order, so the GRU consumes events in the order the trajectory produced them, with a binary validity mask marking real events after padding. The validator applies a feature-wise LayerNorm, a two-layer MLP encoder with hidden size 256, embedding size 128, and dropout 0.1 followed by an output LayerNorm, then a single-layer GRU with hidden size 256. Per-event logits are pooled by a mask-aware logsumexp over valid events, applied after the GRU so that event order can shape the hidden dynamics before aggregation.

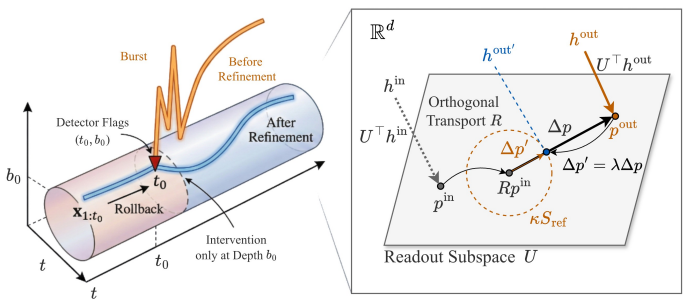
C.3 Training and evaluation

Each prompt receives a single continuation under greedy decoding, replayed teacher-forced for signature extraction. Validators are trained independently per dataset and base LLM pair on an $8{:}2$ train and test split, and all reported numbers use the held-out test split. Only samples with labels in $\{0, 1\}$ are kept. Optimization uses AdamW with learning rate 3×10^{-5} and weight decay 10^{-2} for 300 epochs, with optional gradient clipping at global norm 1.0 and optional mixed precision under a run flag. Class imbalance is handled by weighted binary cross entropy, with the positive class weight set to the ratio of negative to positive counts in the training split and a fallback weight of 1.0 when no positives appear. Evaluation converts logits to probabilities by a sigmoid, accuracy uses threshold 0.5, and AUROC uses the predicted probabilities against the binary labels. **Compute.** All experiments ran on a single NVIDIA A100-80GB GPU. Signature extraction takes roughly 40 seconds per sample depending on dataset and sequence length, and validator training on roughly $2{,}000$ samples takes about 2 hours per dataset and model pair. Per-step overhead during decoding is reported in Table 4.

D Evaluation Protocols

D.1 External judge for hallucination labels

Binary hallucination labels come from a ChatGPT-4o-mini judge conditioned on the same context and the base model output, with a fixed prompt template shared across all tasks and base LLMs and deterministic decoding. The full template and the per-benchmark context assembly are included in the supplementary code. The judge emits labels in $\{-1, 0, 1\}$, where -1 marks unjudgeable cases,



(a) Localization, Selective Intervention (b) Readout-Aligned Clamping Geometry

Figure 3: **Signature-guided refinement. (a)** The validator flags a culprit event (t_0, b_0), generation rolls back to $x_{1:t_0}$, and intervention applies at depth b_0 only. **(b)** Δp is clamped to the prefix-calibrated scale κs_{ref}, preserving the orthogonal residual.

and only labels in $\{0, 1\}$ are kept for training and evaluation. The judge is used identically for our validator and for every baseline, as stated in §4.

D.2 Refinement protocol

Refinement in Table 2 runs on every sample of the evaluation set. For each sample, the validator of §3.5 produces a sample-level prediction and per-event scores, and the intervention is triggered by the validator's prediction alone. Whenever the validator predicts hallucination, refinement proceeds, and otherwise the original generation is kept unchanged. Judge labels play no role in deciding which samples are refined and are used only afterward, to score the resulting outputs. Both the before and after hallucination ratios are computed over the full evaluation set, so unrefined samples enter both ratios identically.

For a triggered sample, the culprit is the valid event with the highest validator score, as in Eq. 14, and its coordinates (t_0, b_0) fix the rollback position and the single intervened block. The prefix $x_{1:t_0}$ is kept, the suffix is regenerated, and the intervention at block b_0 follows Eqs. 15 and 16. The competitor set and fitted basis are frozen after calibration, the reference scale s_{ref} is the masked median of transported step norms over the first $N_{\text{cal}} = 64$ valid prefix steps, and steps exceeding the calibrated band are shrunk in norm with direction preserved (Fig. 3). Unless noted otherwise, refinement uses $K = 32$ competitors, basis dimension $k = 16$, and clamp ratio $\kappa = 1.05$.

General form with transport. The main text states the intervention with a shared basis on both sides of the block. When distinct bases are used, an orthogonal transport $R \in \mathbb{R}^{k\times k}$ aligns them, the transported step is $\Delta p = p^{\text{out}} - R\,p^{\text{in}}$, and the clamp acts on this transported step, with the rewrite $p^{\text{out}\prime} = R\,p^{\text{in}} + \lambda\,\Delta p$ mapped back through U exactly as in Eq. 16. Our default setting uses the same basis on both sides, so $R = I_k$.