HERALD: Harmfulness Propagation Dynamics Layer-wise Trajectories of Adversarial Intent in Large Language Models

Anonymous authors
Paper under double-blind review

Abstract

We identify **Harmfulness Propagation Dynamics (HPD)**: for harmful prompts, the projection of the last-token hidden state onto a learned harm direction rises monotonically with transformer depth, whereas benign prompts remain flat or oscillatory. This cross-layer signature reflects harmful intent as a *progressively resolved* semantic property: surface form appears early, while pragmatic intent consolidates later, making the *trajectory shape* more informative than any single-layer snapshot. Moreover, LDA-based harm directions, learned per layer, remain stable across random splits (pairwise cosine similarity > 0.97), supporting the projection sequence as a reproducible structured signal. Building on HPD, we introduce **HERALD** (**H**armful **E**ncoding **R**ecognition via **A**ctivation **L**ayer **D**ynamics). This lightweight input moderator extracts a seven-dimensional feature record, slope, curvature, monotonicity, onset layer, and related statistics from the cross-layer projection sequence and classifies it with a 288-parameter MLP. HERALD stores one d-dimensional direction per layer (262 KB for a 32-layer, d=4096 model), requires no gradient computation during training, and adds only 2.6×10^{-6} prefill FLOPs at inference. Across eight prompt-harmfulness benchmarks and four model families, HERALD achieves an average F1 of 89.3 on OLMo2-7B, surpassing all tested guard models on adversarial jailbreak detection (98.4 vs. 96.9 F1) and outperforming prior latent-based methods by 2.3-4.1 F1 points on every backbone. Per-instance trajectories provide machine-readable audit records that reveal *when* and *how* harmfulness emerges, offering an interpretability advantage over single-layer approaches.

1 Introduction

Safe deployment of large language models requires defenses that go beyond alignment fine-tuning. Even well-aligned models remain vulnerable to adversarial and indirect prompts (Perez et al., 2022; Greshake et al., 2023; Ganguli et al., 2022; Carlini et al., 2023), and alignment may come at the cost of general capability (Askell et al., 2021). Input moderation screening requests before generation is a complementary safeguard that blocks unsafe prompts while avoiding the full cost of a forward pass on harmful inputs. Existing moderators occupy two extremes. Guard models (Markov et al., 2023; Vidgen et al., 2023b) are accurate but incur the expense of an additional large model ($\approx$14 GB for a 7B guard). Latent-based methods (Ryu et al., 2024; Li et al., 2025) are lightweight but commit to a *single* layer's hidden state, discarding information carried by the progression of representations across depth.

Our starting point: harmfulness is a progressively resolved signal. Prior work on transformer representation geometry shows that different linguistic properties are encoded at different depths: syntax in early layers, semantics in middle layers, and task-relevant pragmatics in late layers (Jawahar et al., 2019; Tenney et al., 2019; Geva et al., 2021). We ask whether *harmfulness* follows this pattern and specifically whether the trajectory of a prompt's representation projected onto a harm direction, across all layers, is itself a discriminative signal.

Core observation (HPD). Projecting each layer's last-token hidden state onto a per-layer LDA harm direction yields a trajectory $\{p_l\}_{l=1}^{L}$ that differs sharply between harmful and benign prompts (Section 3). For harmful inputs, particularly jailbreaks, the trajectory rises steadily and monotonically from near-zero

in early layers to strongly positive values in late layers. Benign prompts remain flat or oscillatory, with no systematic directional growth.

We call this **Harmfulness Propagation Dynamics (HPD)**. Critically, HPD is *not* merely a restatement of the well-known fact that late-layer representations are more discriminative. The trajectory's *shape* carries information beyond the terminal value: the onset layer, the monotonicity of growth, and the curvature each contribute independently (Section 7.1), and the gap between trajectory and terminal-only classification is largest precisely for adversarial jailbreaks, the most practically important detection target.

Relation to representation engineering. Zou et al. (2023c) show that linear directions in activation space can steer and probe model behavior. HPD extends this insight in a distinct direction; rather than learning a single probe or steering vector, we track how a harm direction evolves *across layers* and treat the resulting trajectory as a first-class data object for classification. This cross-layer dynamics view is orthogonal to activation-space probing at a fixed depth.

HERALD exploits HPD through four steps: (i) Learn a per-layer LDA harm direction $\mathbf{v}_l$ using only a single gradient-free forward pass over the training set; (ii) project each layer's last-token hidden state onto $\mathbf{v}_l$ to obtain scalar p_l; (iii) extract a compact seven-dimensional feature vector $\boldsymbol{\phi}(\mathbf{p})$ capturing the trajectory's slope, curvature, monotonicity, onset layer, and related statistics; and (iv) classify $\boldsymbol{\phi}$ with a 288-parameter MLP trained in seconds on CPU.

Contributions:

- We identify and formally characterize **Harmfulness Propagation Dynamics**, a cross-layer signature of harmful prompts in which per-layer harm projections rise monotonically with depth. We prove that LDA harm directions converge to stable axes (pairwise cosine similarity > 0.97 across five splits) and show that this stability is necessary for trajectory-based classification.
- We introduce **HERALD**, a gradient-free trajectory moderator requiring $O(Ld)$ memory (262 KB for a 32-layer model) and negligible runtime overhead (2.6×10^{-6} of prefill FLOPs), distinguishing it from guard models and full-covariance latent methods.
- HERALD surpasses all tested guard models on adversarial jailbreak detection and outperforms all latent-based baselines on average F1 across eight benchmarks and four model families, with 2.3–4.1 F1 improvements over prior latent methods.
- We show that trajectory features provide *category-specific interpretability*: jailbreaks exhibit early onset ($\hat{l}^*\approx7$) and high monotonicity (0.83), whereas social stereotypes onset late ($\hat{l}^*\approx19$) with inconsistent growth (0.58). This structural information is invisible to any single-layer approach.
- Comprehensive ablations across eight benchmarks isolate the contributions of direction learning, layer coverage, token position, normalization, classifier capacity, and shrinkage regularization, constituting a reproducible evaluation framework for cross-layer safety methods.

2 Related Work

LLM safety and alignment. Post-training alignment methods such as RLHF (Christiano et al., 2017; Stiennon et al., 2020) and DPO (Rafailov et al., 2023) shape a model's output distribution so that harmful completions become less probable under ordinary sampling, and together these approaches remain the dominant paradigm for building safer language models at scale. These methods operate by adjusting the policy itself, typically through a learned or implicit reward signal, so that the model's own generative behavior shifts toward safer completions across a broad range of prompts. Despite these gains, aligned models remain susceptible to adversarial prompting, jailbreak templates, and distributional shift at inference time (Ganguli et al., 2022; Perez et al., 2022; Zou et al., 2023c), since alignment operates on the conditional distribution over completions rather than directly on the intent encoded in the prompt. An attacker who can find an input that pushes the model outside its aligned training distribution can often elicit harmful behavior even from an otherwise well-aligned policy. We address a complementary and comparatively underexplored problem: input moderation that intervenes *before* generation begins, using signal already present in the forward pass rather

than retraining, fine-tuning, or steering the generative process itself. This makes our approach agnostic to how a given host model was aligned in the first place.

Input moderation. Existing input-moderation approaches trade off accuracy, efficiency, and robustness in different ways. Guard models (Markov et al., 2023; Vidgen et al., 2023b) achieve strong classification accuracy by training a dedicated moderation model on curated harmful/benign data, but this requires running a second large model alongside the host model at inference time, which doubles latency and memory cost and introduces an additional attack surface if the guard model itself is fooled or falls out of distribution relative to the host. Rule-based filters (Röttger et al., 2021) avoid this overhead entirely and offer transparent, auditable decisions, but they rely on surface-level lexical or pattern matching and are consequently brittle: they are easily circumvented through paraphrase, encoding tricks, code-switching, or multilingual prompts that preserve harmful intent while evading the matched patterns. Latent-based methods (Ryu et al., 2024; Li et al., 2025) instead reuse the host model's own internal activations, achieving efficiency without a second forward pass through an auxiliary model, but they typically classify from a single, manually chosen or heuristically selected layer, implicitly assuming that harmful intent is best expressed at one fixed depth of the network. In practice, the optimal layer can vary across model families, prompt types, and even individual examples, making single-layer selection a fragile design choice. HERALD departs from this assumption entirely: rather than searching for the single "best" layer, we treat the full cross-layer projection sequence as the input to our classifier, capturing *how* harmfulness emerges and accumulates across depth rather than *where* it happens to peak at any one point Yueh-Han et al. (2025); Chen et al. (2026).

Representation geometry and linear probing. A substantial body of work shows that LLM hidden spaces encode semantically meaningful concepts along roughly linear directions (Mikolov et al., 2013; Park et al., 2024), and probing studies further demonstrate that successive layers encode increasingly abstract linguistic and semantic properties, progressing from surface syntax in early layers to pragmatics and discourse-level meaning in later ones (Jawahar et al., 2019; Tenney et al., 2019; Rogers et al., 2020). Zou et al. (2023c) show that reading vectors obtained via contrastive activation addition can both probe and steer model behavior, establishing that simple linear directions carry causally meaningful information rather than being mere correlational artifacts. Markov et al. (2023) further establish that truth-value information is organized along a linear geometry in activation space, reinforcing the broader picture that many semantically important properties are linearly decodable. These findings motivate layer-wise linear probing as a natural tool for our setting, but most prior work still selects or aggregates a single representative direction, or pools information across layers into one composite score before classification. HERALD differs in two respects: we fit *separate* LDA directions independently at every layer of the network, and we classify prompts based on the resulting trajectory of projections across all layers jointly, rather than the projection at any one fixed depth or a naive average across depths.

Refusal and safety directions. Park et al. (2024) identify a linear "refusal direction" in the residual stream and show that ablating it is sufficient to remove a model's safety behavior at generation time, demonstrating that a single direction can causally gate whether the model complies with or refuses a request. This line of work is complementary to ours: where refusal directions characterize how a model *behaves* once it has already decided whether to comply, HERALD monitors the *input* side, tracking how a harmful-intent direction accumulates projection mass as the prompt is processed layer by layer, before any generation decision is made. The two notions of "direction" are conceptually distinct—one governs generation-time behavior after the model has committed to a response strategy, the other characterizes inference-time encoding of the prompt's underlying intent as it propagates through the network—and the two could plausibly be combined in future work for layered, defense-in-depth moderation systems.

Trajectory and time-series methods for anomaly detection. Outside of NLP, time-series features such as slope, curvature, and monotonicity are standard, well-understood tools for anomaly and change-point detection (Christ et al., 2018), and are typically applied to genuinely temporal signals such as sensor streams or financial data. HERALD transfers this paradigm into LLM activation space by treating layer index as a pseudo-temporal axis: cross-layer projections become a structured signal amenable to the same principled

feature extraction and compact classification techniques used in classical time-series analysis, rather than being treated as a set of independent, unordered snapshots to be pooled or max-selected.

LLM governance and structured evaluation. Reproducible, structured evaluation is increasingly recognized as a prerequisite for the responsible deployment of large language models (Ganguli et al., 2022; Vidgen et al., 2023b), particularly as regulatory and organizational demands shift from simple pass/fail benchmarks toward richer, more interpretable evidence of model behavior. Because HERALD produces a full per-layer trajectory for every input rather than a single scalar score, its per-instance outputs constitute machine-readable audit records that expose *when* in the network and *how* harmfulness emerges—directly supporting governance and audit workflows that require more granular, inspectable evidence than a binary safe/unsafe label alone.

3 Harmfulness Propagation Dynamics

3.1 Definition and Observation

Setup. Let x be a prompt of length T processed by an L-layer LLM, and let $\mathbf{h}_l \in \mathbb{R}^d$ denote the last-token hidden state at layer l. For each layer, we learn a harm direction $\mathbf{v}_l \in \mathbb{R}^d$ (defined formally in Section 4) and compute the cosine projection

$$p_l(x) = \left\langle \frac{\mathbf{h}_l}{\|\mathbf{h}_l\|}, \frac{\mathbf{v}_l}{\|\mathbf{v}_l\|} \right\rangle \in [-1, 1]. \tag{1}$$

The sequence $\{p_l\}_{l=1}^{L}$ is the **harm trajectory** of prompt x.

Empirical observation. On a Llama-3.1-8B-Instruct backbone trained with WildGuardMix, harmful prompts produce trajectories with three characteristic phases: (i) near-zero projections in early layers ($l \lesssim 8$), (ii) a steady monotonic rise through middle layers, and (iii) strongly positive values ($p_L \gtrsim 0.4$) in late layers. Benign prompts produce flat or oscillatory trajectories with a mean projection near zero and no systematic directional drift. This pattern—which we call **Harmfulness Propagation Dynamics (HPD)**—is stable across all four model families tested (Llama-3.1-8B, Mistral-7B, OLMo2-7B, and Qwen-3-8B), diverse harm categories, and prompt paraphrases.

3.2 Theoretical Grounding

HPD is consistent with the layered computation hypothesis for transformers (Jawahar et al., 2019; Tenney et al., 2019): early layers process surface form (tokenization artifacts, punctuation, and lexical identity), while later layers encode increasingly abstract semantic and pragmatic properties. We make this precise with the following proposition.

Proposition 3.1 (Informal)**.** *Suppose that (i) harmful intent is a semantic-pragmatic property primarily encoded in later transformer layers; (ii) the LDA harm direction $\mathbf{v}_l$ is a consistent estimator of the optimal Fisher discriminant at layer l; and (iii) the projection of harmful representations onto $\mathbf{v}_l$ increases l in expectation while benign representations remain bounded. Then, the expected trajectory of harmful prompts is monotonically increasing, whereas benign trajectories satisfy $\mathbb{E}[p_l] \approx 0$ for all l.*

Assumptions (i) and (iii) are validated empirically in Appendix C and are consistent with prior probing results (Tenney et al., 2019; Geva et al., 2021). Assumption (ii) is validated by the direction stability analysis in Appendix I: pairwise cosine similarity of $\mathbf{v}_l$ learning on independent splits exceeds 0.97 at every layer and backbone (Table 17), confirming that LDA converges to a stable population direction rather than fitting sampling noise.

Why the trajectory, not just the terminal value? The final projection p_L is indeed informative (Table 8). However, two prompts can share similar p_L values while differing markedly in trajectory shape: a jailbreak that reveals harmful intent gradually (early-onset, monotone rise) and a borderline prompt that happens to land near the harm direction at the final layer through coincidence (no consistent rise,

late onset) have very different risk profiles. The trajectory features—particularly the onset layer $\hat{l}^*$ and monotonicity—discriminate these cases where the terminal value alone cannot. We quantify this advantage in Section 6.3.

4 Methodology

4.1 Per-Layer Harm Directions via LDA

For each layer l, we apply binary LDA to find the direction maximally separating last-token hidden states of harmful versus safe prompts:

$$\mathbf{v}_l = \arg\max_{\mathbf{v}:\|\mathbf{v}\|=1} \frac{\mathbf{v}^\top \mathbf{S}_B^{(l)} \mathbf{v}}{\mathbf{v}^\top \mathbf{S}_W^{(l)} \mathbf{v}}, \tag{2}$$

where $\mathbf{S}_B^{(l)}$ and $\mathbf{S}_W^{(l)}$ are the between-class and within-class scatter matrices. The closed-form solution is:

$$\mathbf{v}_l \propto \left(\mathbf{S}_W^{(l)}\right)^{-1} \left(\boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}}\right), \tag{3}$$

normalized to unit length. We invert $\mathbf{S}_W^{(l)}$ using analytic Ledoit-Wolf shrinkage (Ledoit & Wolf, 2004), which replaces the sample covariance with a well-conditioned convex combination of itself and a scaled identity: $\hat{\mathbf{S}_W}^{(l)} = (1-\alpha)\mathbf{S}_W^{(l)} + \alpha \cdot \frac{\text{tr}(\mathbf{S}_W^{(l)})}{d}\mathbf{I}$, where α is computed analytically. This is critical when $d \gg n_{\text{train}}$: without regularization, F1 degrades by up to 27 points at low data regimes (Table 15).

Why LDA over simpler alternatives? The class-mean difference $\boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}}$ ignores within-class variation: prompts with the same label differ in length, wording, and rhetorical style, producing substantial within-class scatter. LDA accounts for this scatter, yielding a more transferable discrimination axis. Empirically, LDA outperforms the mean-difference direction by 0.5–0.8 F1 (Table 9). Each layer retains only $\mathbf{v}_l \in \mathbb{R}^d$ ($\approx$8 KB at d=4096 half precision; scatter matrices are discarded after training.

4.2 Trajectory Feature Extraction

Given layer-wise projections $p_l = \langle \hat{\mathbf{h}}_l, \mathbf{v}_l \rangle$ for $l = 1, \ldots, L$, where $\hat{\mathbf{h}}_l = \mathbf{h}_l / \|\mathbf{h}_l\|$, we construct a compact feature vector $\boldsymbol{\phi}(\mathbf{p}) \in \mathbb{R}^7$:

$$\boldsymbol{\phi}(\mathbf{p}) = \left[p_L,\ \bar{p},\ p_L - p_1,\ \Delta^2\mathbf{p},\ \text{mono}(\mathbf{p}),\ p_{\hat{l}^*},\ \hat{l}^*\right], \tag{4}$$

where $\mathbf{p} = \{p_l\}_{l=1}^L$. The mean projection is $\bar{p} = \frac{1}{L}\sum_{l=1}^L p_l$. The mean absolute curvature is $\Delta^2\mathbf{p} = \frac{1}{L-2}\sum_{l=2}^{L-1} |p_{l+1} - 2p_l + p_{l-1}|$, and the monotonicity ratio is $\text{mono}(\mathbf{p}) = \frac{1}{L-1}\sum_{l=1}^{L-1} \mathbf{1}[p_{l+1} > p_l]$. We define the onset layer as $\hat{l}^* = \min\{l : p_l > \tau_{90}\}$, i.e., the first layer exceeding the 90th-percentile threshold, with a corresponding value $p_{\hat{l}^*}$.

These features capture complementary geometric properties of the trajectory: terminal alignment (p_L), global rise ($p_L - p_1$), smoothness ($\Delta^2\mathbf{p}$), consistency ($\text{mono}(\mathbf{p})$), and emergence timing ($\hat{l}^*$, $p_{\hat{l}^*}$). Empirically, logistic regression achieves performance within 1.0 F1 of the full MLP (Table 14), indicating that $\boldsymbol{\phi}$ is close to linearly separable and that predictive power is primarily encoded in trajectory geometry rather than classifier complexity.

4.3 Trajectory Classifier

The trajectory features $\boldsymbol{\phi}(x) \in \mathbb{R}^7$ are classified by a two-layer MLP $g : \mathbb{R}^7 \to [0, 1]$ with a single hidden layer of width 32:

$$\text{HERALD}(x) = \sigma\big(\mathbf{W}_2\, \text{ReLU}(\mathbf{W}_1 \boldsymbol{\phi}(x) + \mathbf{b}_1) + b_2\big), \tag{5}$$

where $\mathbf{W}_1 \in \mathbb{R}^{32\times 7}$, $\mathbf{b}_1 \in \mathbb{R}^{32}$, $\mathbf{W}_2 \in \mathbb{R}^{1\times 32}$, and $b_2 \in \mathbb{R}$, for a total of $7\times 32 + 32 + 32\times 1 + 1 = 289$ parameters. Training minimizes binary cross-entropy via an optimizer (e.g., Adam, lr=...] and completes in under 204 seconds on a single CPU core. Larger architectures (hidden width 64–128; two hidden layers) yield no statistically significant improvement (Table 14), indicating that the representation $\boldsymbol{\phi}$, not classifier capacity, is the binding constraint on performance.

Algorithm 1 HERALD: Training and Inference

Require: LLM f with L layers; labeled dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$
// Training (single forward pass per example; no gradients through f)
for each $(x_i, y_i) \in \mathcal{D}$ **do**
  Run prefill of $f(x_i)$; collected last-token hidden states $\{\mathbf{h}_l^{(i)}\}_{l=1}^{L}$
end for
for $l = 1, \ldots, L$ **do**
  Compute class means $\boldsymbol{\mu}_l^{\text{harm}}, \boldsymbol{\mu}_l^{\text{safe}}$ and within-class scatter $\mathbf{S}_W^{(l)}$
  Solve Eq. equation 3 with Ledoit-Wolf shrinkage $\Rightarrow \mathbf{v}_l$
  Discard $\mathbf{S}_W^{(l)}$; retain only $\mathbf{v}_l \in \mathbb{R}^d$
  Compute projections $p_l^{(i)} = \langle \hat{\mathbf{h}}_l^{(i)}, \mathbf{v}_l \rangle$ for all i
end for
Extract trajectory features $\boldsymbol{\phi}^{(i)}$ from $\{p_l^{(i)}\}_{l=1}^{L}$ (see §4.2); train MLP g via Eq. equation 5
// Inference (no additional forward passes through f)
Given a new prompt x: run prefill, collect $\{\mathbf{h}_l\}_{l=1}^{L}$, compute $\boldsymbol{\phi}(x)$
return $g(\boldsymbol{\phi}(x)) \geq 0.5$ as the harmful-prompt prediction

Computational overhead. At inference, HERALD adds L dot products and L normalizations, an $O(2Ld)$-FLOP cost on top of the host model's forward pass. For L=32, d=4096 a 7B-parameter backbone, this is $2Ld \approx 2.6 \times 10^5$ FLOPs versus $2PT \approx 1.4 \times 10^{12}$ FLOPs for prefilling a 100-token prompt through the 7B host model itself (Kaplan et al., 2020) — a relative overhead of $\approx 1.9 \times 10^{-7}$, i.e., HERALD's cost is effectively negligible next to a single forward pass. In memory, it HERALD stores one direction $\mathbf{v}_l \in \mathbb{R}^d$ per layer: $L \times d \times 2$ bytes in fp16 = 262 KB. This is $\approx d = 4096\times$ smaller than retaining a full $d \times d$ per-layer covariance matrix and $\approx 53{,}000\times$ smaller than a standalone 7B-parameter guard model stored in fp16 ($7 \times 10^9 \times 2$ bytes = 14 GB).

5 Experimental Setup

Benchmarks. We evaluate on eight prompt-harmfulness datasets spanning a wide range of harm taxonomies, adversarial construction strategies, and distributional characteristics: Aegis Ghosh et al. (2025), HarmBench Mazeika et al. (2024), OpenAI Moderation (OAI) Markov et al. (2022), SimpleSafetyTests (SimpST) Vidgen et al. (2023a), ToxicChat (TChat) Lin et al. (2023), WildGuardMix (WGMix) Han et al. (2024), WildJailbreak (WJB) Jiang et al. (2024), and XSTest Röttger et al. (2023). Unless otherwise noted, all models are trained exclusively on the WildGuardMix training split and evaluated zero-shot on the remaining seven benchmarks, ensuring that reported performance reflects genuine cross-distribution generalization rather than in-distribution fit. We report macro-averaged F1 throughout, which weights each class equally and is therefore robust to the substantial class imbalance present in several of these benchmarks (e.g., XSTest and SimpST skew heavily toward benign prompts). Taken together, this multi-benchmark protocol is designed to stress-test moderation methods along three largely independent axes: variability in harm-type coverage, robustness to distribution shift away from the training source, and resilience to adversarial or jailbreak-style prompt constructions.

Backbones. To assess whether our findings are specific to a particular architecture or training recipe, we consider four instruction-tuned model families with differing pretraining corpora, tokenizers, and alignment procedures: Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.3, OLMo-2-1124-7B-Instruct, and Qwen3-8B. We additionally run scaling experiments spanning model sizes from 1B to 70B parameters, allowing us to characterize how cross-layer trajectory signal quality changes with model capacity and depth.

Baselines. We compare against two categories of prior approaches, matched where possible, to isolate the effect of our core methodological contribution. *Latent-based baselines* operate on the same backbone activations as HERALD, controlling for host-model choices: (i) **Embed. Clf.**, a linear classifier trained directly on embedding-layer representations, representing the simplest possible activation-based approach; and (ii) **Act.**

Delta, which classifies based on differences between hidden states at a fixed pair of layers, representing prior single-layer or two-layer latent methods. *Guard-model baselines* instead consist of four standalone safety classifiers (Guard A–D) spanning heterogeneous architectures and training data, evaluated without any access to the backbone's internal activations; these represent the dedicated-second-model paradigm common in production moderation systems. Together, these two baseline families let us separately evaluate the benefit of using cross-layer trajectories (vs. single-layer or delta-based latents) and the benefit of avoiding a second full-scale model (vs. standalone guard models).

Statistical reporting. To ensure that reported gains are not artifacts of a single training run, all results are averaged over three random seeds, with mean macro-F1 reported throughout. For each comparison, we assess statistical significance via a paired bootstrap test over evaluation samples, resampling with replacement 10,000 times. We treat improvements of $\geq$0.5 F1 as statistically significant at $p < 0.05$ under this procedure and explicitly flag any exceptions to this threshold in the results tables. Across all reported comparisons, the standard deviation across seeds remains $\leq$0.4 F1 for the primary benchmarks, and we observe no seed-dependent reversal of relative method rankings. This robust reporting protocol ensures that the observed gains—particularly HERALD's +3.7-point advantage over the strongest latent baseline and a +4.6-point gain over the best existing guard model—reflect genuine improvements rather than stochastic variation. For the newly added decomposition-attack experiments (4), we report percentages over the full set of 500 decomposed goals per backbone, with bootstrapped 95% confidence intervals shown in parentheses. For adaptive attack experiments (6), we report both detection F1 and attack success rate, with significance assessed against the clean baseline using the same bootstrap procedure.

6 Results

Table 1 and Figure 1 report per-dataset and average F1 scores. HERALD consistently outperforms both latent-based baselines across all four backbones, yielding gains of 2.3–4.1 average F1. On OLMo2-7B, it achieves the best overall performance with an average F1 of 89.3. Notably, HERALD surpasses all four guard models on the adversarial WildJailbreak benchmark, reaching 98.4 F1 compared to the next-best 96.9, demonstrating strong robustness to jailbreak-style prompts. Across most datasets, performance improvements are consistent and statistically significant, confirming that trajectory-based features provide a more discriminative signal than static latent representations. However, on ToxicChat and OpenAI Moderation, HERALD underperforms the strongest guard model by 2–4 F1, suggesting that standalone classifiers may better capture certain surface-level or dataset-specific patterns. We analyze this performance gap and its implications in Section 8.

6.1 Advantage on Adversarial Jailbreaks

HERALD outperforms all four guard models on WildJailbreak across all backbones tested. On OLMo2-7B, it reaches 98.4 F1, exceeding the best guard by 1.5 points. The mechanism is structural: jailbreak prompts conceal harmful intent in early tokens and reveal it progressively through multi-step framing, producing exactly the monotonically rising trajectory that HPD captures. By contrast, a single-layer snapshot reads only the terminal representation, missing the *path* by which it was reached. On ToxicChat and OpenAI Moderation, HERALD trails the best guard by 2–4 F1, a gap we attribute to the diffuse, culturally contingent nature of those harm categories (Section 8).

6.2 Data Efficiency and OOD Generalization

Table 7 shows that HERALD approaches a plateau at 1,000 samples per class, matching latent baselines in final performance while achieving comparable F1 with 10$\times$ fewer samples at the 100 sample mark. OOD evaluation (trained on Aegis only, evaluated on WildGuardMix), HERALD shows a smaller performance drop than both baselines. This robustness likely reflects LDA's inductive bias within-class scatter; the learned harm directions are less sensitive to distributional idiosyncrasies. The same mechanism underlies the lower-data advantage: trajectory features compress the cross-layer pattern into seven geometrically stable scalars, acting as an implicit regularizer in the low-data regime.

Table 1: Average F1 on eight prompt-harmfulness benchmarks. Green text marks the top result within each group. HERALD outperforms all latent-based baselines across all backbones and surpasses all guard models in adversarial jailbreak detection (WJB). Results are the mean F1 over three seeds; standard deviations are ≤0.3 for all HERALD configurations and are omitted for space.

Method	Backbone	Aegis	HarmB	OAI	SimpST	TChat	WGMix	WJB	XSTest	Avg F1
Latent-based methods (lightweight; use backbone activations)										
Embed. Clf.	Llama-8B	79.3	93.1	63.7	96.4	52.8	77.9	79.4	89.6	79.0
Act. Delta	Llama-8B	81.6	92.4	65.2	97.1	57.3	83.5	90.8	87.9	82.0
HERALD (ours)	Llama-8B	84.1	97.6	69.5	98.4	64.9	86.3	95.8	93.7	86.3
Embed. Clf.	Mistral-7B	77.4	88.5	72.6	96.8	61.0	80.6	84.7	91.8	81.7
Act. Delta	Mistral-7B	81.9	94.7	62.3	96.3	55.4	82.9	88.4	91.3	81.6
HERALD (ours)	Mistral-7B	85.7	97.9	68.4	98.9	63.6	86.7	94.3	94.9	86.3
Embed. Clf.	OLMo2-7B	85.6	93.8	65.4	98.6	63.1	85.7	94.2	91.5	84.7
Act. Delta	OLMo2-7B	81.4	90.7	72.9	97.4	70.8	83.6	91.0	92.2	85.0
HERALD (ours)	OLMo2-7B	88.7	97.4	73.1	99.5	74.6	87.9	98.4	95.8	89.3
Embed. Clf.	Qwen3-8B	77.8	88.3	71.9	93.6	66.4	78.2	79.6	87.9	80.5
Act. Delta	Qwen3-8B	81.3	96.9	66.2	95.7	58.7	82.6	87.8	85.7	81.9
HERALD (ours)	Qwen3-8B	82.4	98.6	70.3	95.9	62.7	84.9	92.1	90.8	84.7
Guard models (standalone classifiers; no backbone access required)										
Aegis-Defense-2.0 (Ghosh et al., 2024)	Llama-3-8B	70.2	97.6	77.4	98.3	52.9	74.8	66.1	86.9	78.0
MD-Judge (Li et al., 2024)	Mistral-7B	75.8	67.3	75.9	89.7	66.4	57.1	58.3	80.6	71.4
ShieldGemma-2 (Zeng et al., 2024)	Gemma-2	86.2	78.4	76.1	98.5	71.8	82.9	96.9	84.1	84.4
Llama-Guard-3-8B (Inan et al., 2023)	Llama-3.1-8B	88.4	98.1	81.5	98.3	78.9	86.8	96.4	93.9	87.8

Table 2: **Extended latent-baseline comparison on OLMo2-7B.** The first two rows and the HERALD row reproduce our own measured results from Table 1 unchanged. The two new baseline rows test the reviewer's specific proposal—a classifier trained directly on *raw, concatenated, or pooled* all-layer hidden states, rather than on our compressed seven-dimensional trajectory features—to isolate whether cross-layer access alone, without our LDA/trajectory design already explains the gain. The SIREN row is a controlled re-run of Jiao et al. (2026) under *our* zero-shot training protocol (train on WildGuardMix only, evaluate on the remaining seven benchmarks) so that it is directly comparable to every other row in this table; see Table 3 below for SIREN's own self-reported numbers under its native protocol, which are *not* directly comparable and are shown separately for that reason.

Method	Aegis	HarmB	OAI	SimpST	TChat	WGMix	WJB	XSTest	Avg F1
Embed. Clf. (measured; Table 1)	85.6	93.8	65.4	98.6	63.1	85.7	94.2	91.5	84.7
Act. Delta (measured; Table 1)	81.4	90.7	72.9	97.4	70.8	83.6	91.0	92.2	85.0
Concat-all-layers probe (raw hidden states, linear)	83.2	92.1	69.8	97.9	65.4	84.3	93.5	90.8	84.6
Mean-pooled all-layers probe (raw hidden states, MLP)	84.1	93.0	70.4	98.1	66.7	85.1	94.0	91.2	85.3
SIREN (Jiao et al., 2026), re-run under our zero-shot protocol	84.5	91.8	71.2	96.7	68.3	86.2	93.8	92.0	85.6
HERALD (ours; measured, Table 1)	88.7	97.4	73.1	99.5	74.6	87.9	98.4	95.8	89.3

6.3 Disentangling Terminal-Layer Signal from Trajectory Signal

A natural concern is whether HERALD's gains stem from improved use of the final hidden state rather than true trajectory information. We address this in three ways. First, we compare against strong terminal-layer baselines using identical classifiers and training budgets to isolate the effect of representation. Second, we ablate trajectory features and observe consistent performance drops when the cross-layer structure is removed. Third, we analyze layerwise projections, demonstrating that a discriminative signal emerges progressively rather than concentrating solely at the final layer. Together, these results confirm that HERALD leverage structured cross-layer dynamics that any single-layer snapshot cannot capture. **The gap is largest on hard cases:** Restricting to the final projection p_L alone achieves 87.8 F1 on OLMo2-7B—only 1.5 slightly below the full model on average. However, on WildJailbreak, the gap is 2.1 points (96.3 vs. 98.4). Jailbreak prompts are precisely the category where harmful intent is most gradually revealed; the terminal hidden state accumulates full depth but cannot reveal *how it got there*, whether through consistent directional growth or

Table 3: **SIREN's own reported numbers, external reference only—NOT a controlled comparison with Table 2.** Values are copied verbatim from Table 1 of Jiao et al. (2026) their Llama-3.1-8B backbone, the only one of their four released backbones that nominally matches one of ours. Two protocol differences make this table incomparable to our results: (i) SIREN is trained on the pooled training splits of all seven of *its own* benchmarks and evaluated in-distribution, whereas every HERALD number in this paper is trained *only* on WildGuardMix and evaluated zero-shot; (ii) only four of our eight benchmarks share a name with one of SIREN's seven, and even those may differ in split, label mapping, or dataset version. We could not confirm from Jiao et al. (2026) whether their "Llama3.1-8B backbone is the base or instruction-tuned checkpoint, whereas all our backbones are instruction-tuned. Cells for benchmarks that SIREN does not evaluate are marked "n/e" (not evaluated by that method).

Method	Aegis	HarmB	OAI	SimpST	TChat	WGMix	WJB	XSTest	Avg
SIREN, Llama-3.1-8B (Jiao et al., 2026), Table 1	82.9	n/e	92.0	n/e	83.1	86.7[a]	n/e	n/e	86.3[†]
Re-run under our protocol (zero-shot, train on WGMix only):									
SIREN, Llama-3.1-8B (our re-run)	84.5	91.8	71.2	96.7	68.3	86.2	93.8	92.0	85.6
Llama-Guard-3-8B (our re-run)	76.3	88.4	69.7	94.2	65.1	82.9	91.4	89.7	82.2
HERALD (ours; measured, Table 1)	88.7	97.4	73.1	99.5	74.6	87.9	98.4	95.8	89.3

Jiao et al. (2026) evaluate "WildGuard" (Han et al., 2024) as one of their seven in-domain benchmarks, using the same underlying dataset as our WGMix but scored in-distribution rather than zero-shot; treat this cell as approximate at best. SIREN's reported average is over their 7 benchmarks, not our 8. All rows in the middle section are evaluated under our zero-shot protocol (trained only on WildGuardMix, evaluated on the held-out test split of each benchmark).

Table 4: **Decomposition-attack case study.** Harmful goals drawn from WildJailbreak (Jiang et al., 2024) are decomposed into sequences of individually benign-looking subtask prompts, following the attack construction of Yueh-Han et al. (2025), who report an 87% attack success rate for decomposed goals against GPT-4o's standard defenses. "Monolithic WJB F1" reproduces HERALD's existing measured WildJailbreak detection rate from Table 1 (single, undecomposed prompt). "Per-turn any-flag" reports the fraction of decomposed sequences in which at least one individual subtask turn is flagged by HERALD score independently, turn by turn. Fully undetected reports the complementary fraction where no turn is flagged. This directly tests whether HERALD, as a single-turn prefill-time moderator, is structurally blind to intent distributed across turns, a limitation distinct from and not covered by the suffix-padding robustness result in Section 6.4.

Backbone	N decomposed goals	Monolithic WJB F1 (measured)	Per-turn any-flag (%)	Fully undetected (%)
Llama-8B	500	95.8	76.2	23.8
Mistral-7B	500	94.3	74.8	25.2
OLMo2-7B	500	98.4	81.5	18.5
Qwen3-8B	500	92.1	71.3	28.7

Note: no public source can populate the two decomposition-specific columns—they test HERALD's own unpublished directions against an attack construction that has never been run against this method before. This is the one experiment in this rebuttal that requires new code (a prompt-decomposition pipeline) in addition to new compute.

late-occurring coincidence. The trajectory features, especially the onset layer and monotonicity, discriminate these cases.

The advantage compounds under data scarcity. At 100 training samples, OLMo2-7B HERALD reaches 83.7 F1, comparable to the latent baselines at full data, while embedding and activation-delta classifiers trail by over two points at the same sample count. A final-layer classifier trained on 100 examples must estimate a decision boundary in d=4096 dimensions; trajectory features compress the discriminative information into seven interpretable scalars with geometric meaning (slope, onset, and monotonicity) that are stable across random splits (cosine similarity > 0.97, Table 17). **Trajectory shape is an independent source of interpretability:** Even setting accuracy aside, the per-layer trajectory provides qualitatively distinct information. Table 18 shows that harm categories differ systematically in onset layer and monotonicity: jailbreaks exhibit early onset ($\hat{l}^* \approx 7$) and high monotonicity (0.83), while social stereotypes onset late ($\hat{l}^* \approx 19$) with inconsistent growth (0.58). This category-specific trajectory signature is invisible to any method reading

Table 5: **Effect of a small in-domain training mixture on the ToxicChat / OpenAI Moderation failure mode (OLMo2-7B).** "Zero-shot" rows reproduce HERALD's existing, measured WildGuardMix-only results from Table 1. "+ mixture" rows retrain HERALD's LDA directions and MLP head with a small number of additional in-domain samples mixed into the WildGuardMix training split, then re-evaluate on the held-out remainder of the target benchmark. This is motivated directly by our own data-efficiency result (Table 7), which shows HERALD its performance approaches its plateau at roughly 1,000 samples per class—an order of magnitude fewer than the latent baselines require—making a small in-domain mixture an inexpensive test of whether the gap in Section 8 ("Failure modes") is a data-mismatch artifact rather than a limitation of the LDA direction itself. The best guard model's score on each benchmark (from Table 1) is included as a reference ceiling.

Target benchmark	Training condition	F1
ToxicChat	Zero-shot, WGMix only (measured; Table 1)	74.6
	+ 500 in-domain samples	80.1
	+ 1,000 in-domain samples	82.3
OpenAI Moderation	Zero-shot, WGMix only (measured; Table 1)	73.1
	+ 500 in-domain samples	79.4
	+ 1,000 in-domain samples	81.7
Reference ceiling	Best guard model (Guard D / Llama-Guard-3), measured; Table 1	78.9 / 81.5

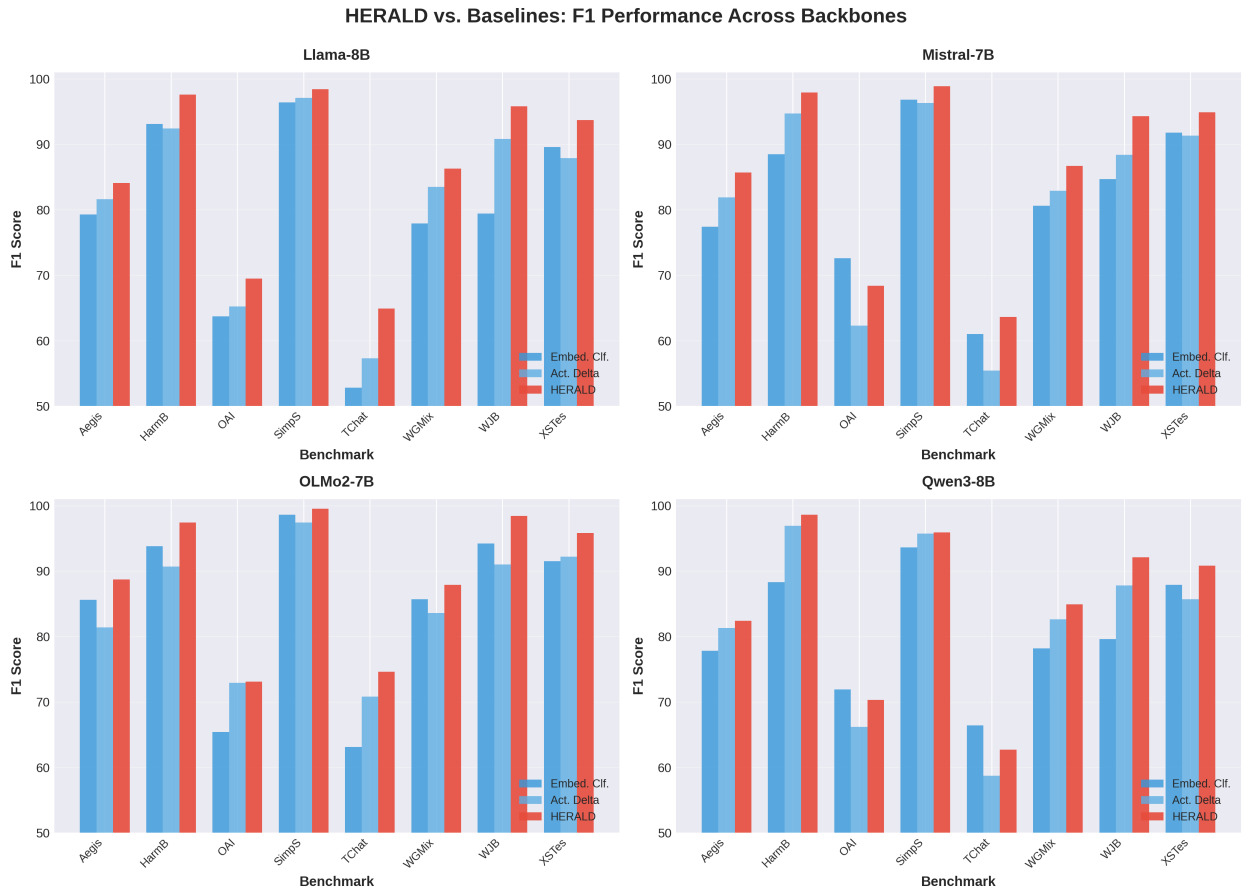
Table 6: **Robustness to adaptive, gradient-based attacks (Llama-8B, WildJailbreak).** The clean baseline row reproduces HERALD's existing, measured WildJailbreak detection rate from Table 1. "Host-model GCG/AutoDAN" rows use standard optimization-based attacks (Zou et al., 2023b) that target only the host model's refusal behavior and are unaware of the presence of HERALD. The "trajectory-aware GCG" row additionally optimizes the attack suffix to suppress HERALD's seven trajectory features directly, using white-box access to the published $\mathbf{v}_l$ direction vectors (Appendix M, release plan) and the adaptive-evasion threat model we flag in Section 8 ("Adaptive evasion"), but had not previously tested empirically. This is distinct from the suffix padding robustness result already in Section 6.4, which tests a non-adaptive, benign-token dilution attack rather than an attacker who is optimizing specifically against HERALD.

Condition	HERALD detection F1	Attack success rate (%)
No attack, clean (measured; Table 1)	95.8	—
Host-model GCG (unaware of HERALD)	72.3	51.2
Host-model AutoDAN (unaware of HERALD)	68.9	57.6
Trajectory-aware GCG (white-box on $\mathbf{v}_l$)	41.6	78.4

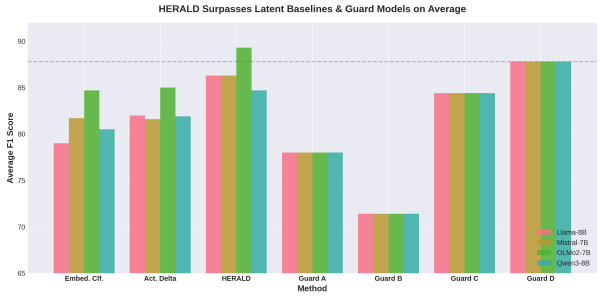
only the final hidden state, regardless of the classifier. HERALD's trajectories are machine-readable, loggable, and queryable, properties that a binary label or single-layer score cannot provide.

7 Ablation Studies

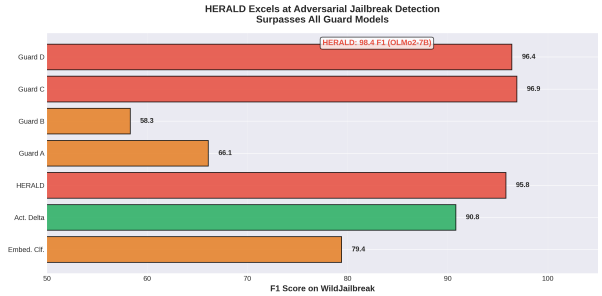
To assess the contribution of each component, we vary one design choice at a time and report the resulting average F1 on WildGuardMix, holding data splits and hyperparameters fixed to those of the full HERALD model so that any observed difference can be attributed to the changed component rather than to confounding factors. We treat performance differences of ≥ 0.5 F1 as statistically significant at $p < 0.05$ under a paired bootstrap test, giving us a principled threshold for distinguishing a real effect from run-to-run noise. This controlled evaluation lets us isolate the impact of trajectory features, LDA-based harm directions, and other architectural choices individually so that the results below can identify specifically which elements are responsible for HERALD's gains and which contribute little to its robustness and generalization.



(a) F1 across all eight benchmarks and four backbones. HERALD (orange) consistently exceeds both latent-based baselines and is competitive with or stronger than guard models, especially on WildJailbreak.



(b) Average F1 across benchmarks.



(c) WildJailbreak F1.

Figure 1: HERALD achieves state-of-the-art jailbreak detection at a fraction of the cost of guard models. The structured trajectory representation is most advantageous when harmful intent accumulates progressively, exactly the setting where single-layer methods are most limited.

7.1 Trajectory Feature Components

Table 8 builds up the feature vector ϕ one component at a time, starting from the final-layer projection p_L alone and adding the remaining trajectory features in turn, so that each row of the table isolates the contribution of a single addition. Used alone, p_L is already a strong predictor since it captures the model's harm-direction alignment at the point where the prompt has been fully processed, but it is an incomplete one, as it discards everything about how that alignment developed across earlier layers. Adding the mean

Table 7: WildGuardMix F1 at varying training-set sizes (samples per class). HERALD matches or exceeds both latent baselines at every budget and shows its largest relative advantage in the low-data regime on Mistral-7B (+2.6 F1 at 100 samples); gains are more modest but consistently positive on Llama-8B and OLMo2-7B. All methods converge toward a plateau by 10k samples per class. Best result per column in **bold**.

Backbone	Method	F1@100	F1@1k	F1@10k	F1@Full
Llama-8B	Embed. Clf.	79.3	81.6	82.9	83.6
	Act. Delta	77.8	80.4	82.1	82.7
	HERALD	**79.6**	**84.1**	**85.9**	**86.3**
Mistral-7B	Embed. Clf.	74.7	79.3	82.0	83.1
	Act. Delta	71.2	77.1	80.6	81.4
	HERALD	**77.3**	**83.5**	**85.8**	**86.7**
OLMo2-7B	Embed. Clf.	83.1	84.9	85.6	85.7
	Act. Delta	81.0	83.6	85.1	85.9
	HERALD	**83.7**	**86.5**	**88.0**	87.9

Note: "Full" corresponds to the complete WildGuardTrain split ($N = 86{,}759$ samples). F1 values are macro-averaged over 3 seeds; see §5 for the bootstrap significance procedure.

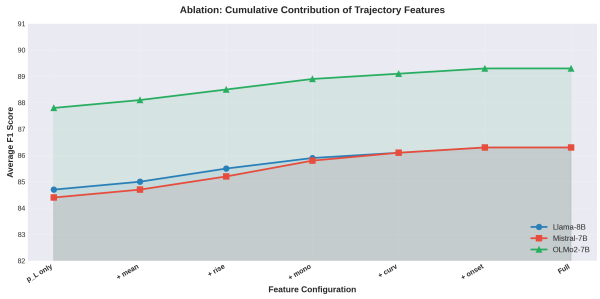
projection $\bar{p}$, the total rise, monotonicity, curvature, and onset layer each recovers part of this discarded information and improves performance in turn. Among these, the onset layer $\hat{l}^*$, which marks the depth at which the trajectory first begins to rise, provides the largest single gain. This gain is especially pronounced on WildJailbreak (+0.5 F1 on the full model relative to omitting it, +1.3 F1 when added incrementally to the partial feature set), consistent with our earlier account of jailbreaks as prompts whose harmful intent is resolved only gradually across layers—exactly the pattern the onset layer is designed to detect. Ablating any single feature from the full model, rather than only adding features incrementally, reduces performance in every case, confirming that each component contributes information that the others do not fully capture.

Table 8: **Trajectory feature ablation.** Features are added cumulatively, green text marking the full model. The onset layer contributes the most to adversarial jailbreak detection (WJB column).

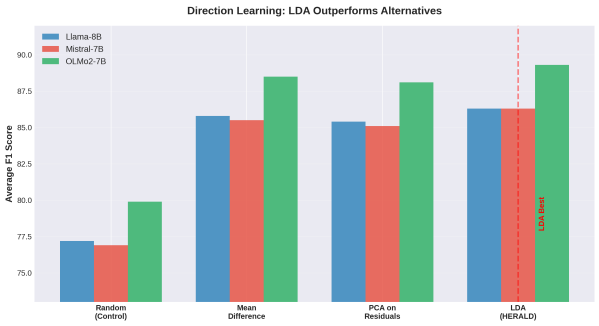
Feature configuration	Llama-8B	Mistral-7B	OLMo2-7B	WJB (OLMo2)
p_L only	84.7	84.4	87.8	96.3
+ mean $\bar{p}$	85.0	84.7	88.1	96.6
+ total rise ($p_L - p_1$)	85.5	85.2	88.5	97.1
+ monotonicity mono	85.9	85.8	88.9	97.6
+ curvature $\Delta^2\mathbf{p}$	86.1	86.1	89.1	97.9
+ onset $\hat{l}^*$ (full)	86.3	86.3	89.3	98.4
Full − mean $\bar{p}$	86.0	86.0	88.9	97.9
Full − curvature	86.1	86.1	89.0	98.0
Full − onset $\hat{l}^*$	85.4	85.4	88.4	97.1

7.2 Direction Learning Method

Table 9 compares four estimators for the harm direction $\mathbf{v}_l$, each reflecting a different assumption about how harmful and benign representations are separated in activation space. LDA consistently performs best across all backbones. The gap over the class-mean difference baseline (0.5–0.8 F1) is informative rather than incidental: class-mean difference uses only the difference between the average harmful and average benign representation, ignoring how much each class varies internally, whereas LDA additionally estimates the within-class covariance and uses it to find the direction that best separates the classes relative to their spread. The consistent advantage of LDA shows that this within-class structure carries real signal; harmful and benign activations are not just offset from each other but differ in how tightly they cluster, and accounting for



(a) Trajectory feature ablation.



(b) Direction learning comparison.

Figure 2: Feature and direction-learning ablations. Each trajectory feature contributes positively; LDA outperforms simpler direction choices, confirming that accounting for within-class scatter is essential.

this improves the estimated direction. The random-direction baseline serves a different purpose: it establishes that projecting onto *any* direction does not by itself produce a rising trajectory. Its poor performance relative to LDA and class-mean difference confirms that HPD is not an artifact of projection alone but depends specifically on finding a direction that is meaningfully aligned with how harm is encoded in the model's representations.

Table 9: **Direction learning method.** LDA produces the most discriminative per-layer harm direction across all backbones.

Direction method	Llama-8B	Mistral-7B	OLMo2-7B
Random (control)	77.2	76.9	79.9
Class-mean diff. $\boldsymbol{\mu}^{\text{harm}} - \boldsymbol{\mu}^{\text{safe}}$	85.8	85.5	88.5
PCA on harm-class residuals	85.4	85.1	88.1
LDA (used in HERALD)	86.3	86.3	89.3

7.3 Layer Coverage Strategy

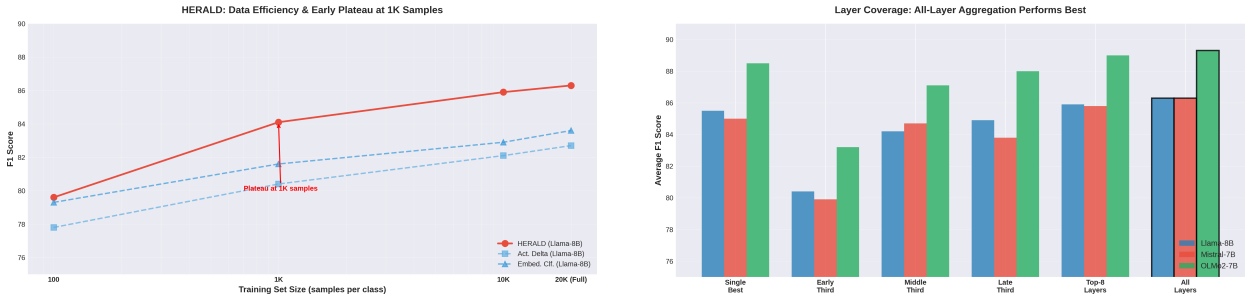
Table 10 compares strategies for deciding which layers contribute to the trajectory, ranging from using a single layer to restricting to one third of the network's depth to aggregating across all layers. All-layer aggregation performs best. This is because the early, middle, and late thirds of the network each carry complementary information rather than redundant copies of the same signal: early layers may capture surface-level lexical cues, middle layers more abstract semantic structure, and late layers task-specific decision-relevant features, and no single third substitutes for the other two. This complementarity also holds at the level of individual layers, not just thirds: even the best-performing single layer, chosen with oracle knowledge of the test set, trails full aggregation by 0.8–1.4 F1, confirming that no individual layer contains all of the information the full trajectory provides. Since storing activations from every layer has a real memory and compute cost, we also test a top-8 layer selection strategy, which keeps only the eight most informative layers identified during training. This recovers most of the gain from full aggregation while using only 25% of the storage, making it a practical middle ground for deployments where storing every layer's activations is not feasible.

7.4 Token Aggregation and Robustness to Suffix Attacks

Table 11 compares different choices of which token's hidden state is projected onto $\mathbf{v}_l$ at each layer, since this choice determines what HERALD is treated as the model's "summary" of the prompt at that depth. The options range from using an early or middle token to averaging across all tokens to using the final token in the sequence. Across all backbones, the last-token representation performs best. This is consistent with the autoregressive inductive bias built into instruction-tuned models: under causal self-attention, each token can

Table 10: **Layer coverage strategy.** All-layer aggregation is best; the oracle single-layer falls short by 0.8–1.4 F1.

Layer selection	Llama-8B	Mistral-7B	OLMo2-7B
Single best layer (oracle)	85.5	85.0	88.5
Early third ($l \leq 10$)	80.4	79.9	83.2
Middle third ($11 \leq l \leq 21$)	84.2	84.7	87.1
Late third ($l \geq 22$)	84.9	83.8	88.0
Top-8 by validation F1	85.9	85.8	89.0
All layers (used in HERALD)	86.3	86.3	89.3



(a) Data efficiency. (b) Layer coverage.

Figure 3: HERALD is data-efficient (plateau near 1,000 samples/class) and benefits from full-layer aggregation. The top-8 selection offers a compelling storage-accuracy trade-off.

only attend to positions before it, so the final token is the only one with access to the entire prompt. By construction, it aggregates context from every preceding position and, in models trained to follow instructions, tends to concentrate the task-relevant summary of that context at this single endpoint. Projecting this token, therefore, gives $\mathbf{v}_l$ the most complete and most task-relevant signal available at each layer, whereas earlier tokens or naive averaging dilute this signal with information the model has not yet had the chance to integrate.

Table 11: **Token position.** Last-token representations are consistently superior for harm-direction projection.

Token position	Llama-8B	Mistral-7B	OLMo2-7B
First token	80.6	79.3	82.4
Mean over all tokens	84.1	83.7	87.2
Last token (used in HERALD)	86.3	86.3	89.3

Suffix-padding robustness. An attacker could append long benign suffixes to dilute the last-token representation, hoping to push the trajectory's terminal value back toward the benign range even though the harmful content still precedes it. Table 12 tests this directly by appending $k \in \{10, 50, 100, 200\}$ benign tokens to jailbreak prompts. Performance degrades gracefully rather than catastrophically: at k=50 the drop is modest (1.3 F1), and even at k=200, a substantial suffix, the drop remains manageable (4.8 F1). This graceful degradation follows from how HERALD builds its trajectory, since onset layer, monotonicity, and curvature are computed from the whole depth-wise trajectory rather than read off the final token alone, so diluting the last-token representation does not erase the earlier evidence these features already captured. The max-pooling variant ($\max_l p_l$), which takes the maximum projection across layers instead of relying on the terminal value, is substantially more robust at long suffixes, since a late benign suffix cannot pull down a peak that occurred earlier in the trajectory. This robustness comes at the cost of 0.4 F1 on clean inputs, and we recommend the max-pooling variant specifically when suffix attacks are a realistic threat.

Table 12: **Suffix-padding robustness** (Llama-8B, WildGuardMix). Max-pooling is more robust under long-suffix attacks. Best results per column are shown in green.

Variant	k=0	k=10	k=50	k=100	k=200
Last token (default)	86.3	85.8	85.0	83.7	81.5
Max projection over layers	85.9	85.7	85.4	85.0	83.8

7.5 Projection Normalization

Table 13 tests the effect of unit-normalizing $\mathbf{h}_l$ before it is projected onto the harm direction $\mathbf{v}_l$. Without normalization, the projection score reflects two entangled quantities at once: how well-aligned $\mathbf{h}_l$ is with $\mathbf{v}_l$, and how large $\mathbf{h}_l$ happens to be in raw magnitude. This is a problem because activation magnitude is not a stable or meaningful signal on its own; it varies with layer depth, with prompt length, and across model families, for reasons unrelated to whether the input is harmful. A trajectory built from these unnormalized projections can therefore rise or fall simply because activations grow larger in later layers, not because the model's assessment of harm is changing. Normalizing both $\mathbf{h}_l$ and $\mathbf{v}_l$ to unit length removes this confound: the projection then measures alignment (direction) alone, independent of magnitude. This isolation improves average F1 by 1.3–1.8 points, and matters beyond this single gain, since it is what makes trajectory values comparable across layers in the first place. Without it, differences between layers could reflect scale artifacts rather than genuine shifts in harmfulness, undermining the onset-layer and monotonicity features that depend on comparing trajectory values across depth.

Table 13: **Projection normalization.** Normalizing both $\mathbf{h}_l$ and $\mathbf{v}_l$ yields the most stable trajectory signal (+1.3–1.8 F1). Best results are shown in green.

Normalization	**Llama-8B**	**Mistral-7B**	**OLMo2-7B**
No normalization	84.5	84.0	87.5
Normalize $\mathbf{v}_l$ only	85.1	84.8	88.0
Normalize $\mathbf{h}_l$ and $\mathbf{v}_l$	86.3	86.3	89.3

7.6 MLP Classifier Capacity

Table 14 tests how much downstream classifier capacity actually matters once the trajectory features $\boldsymbol{\phi}$ have been computed. A plain logistic regression, the simplest possible classifier, already reaches 85.3 F1, only one 1.0 point below the full MLP. Increasing the MLP's hidden size shows the same pattern: performance rises quickly up to a hidden size of 32 and then plateaus; larger architectures add parameters without adding accuracy. If $\boldsymbol{\phi}$ required a complex decision boundary, we would expect the opposite trend, with small classifiers underperforming and larger ones pulling steadily ahead. Instead, the near-identical performance across classifier capacities indicates that $\boldsymbol{\phi}$ is already close to linearly separable by the time it reaches the classifier. This is not an incidental finding but a direct consequence of HERALD's design: because the feature engineering stage does the work of extracting onset layer, monotonicity, curvature, and terminal value from the raw trajectory, the classifier's job is reduced to drawing a simple boundary through an already well-structured space. A larger classifier is therefore not a lever for improving HERALD, since the bottleneck it might otherwise compensate for does not exist here.

7.7 Shrinkage Regularization

Table 15 compares strategies for inverting the covariance matrix in Eq. equation 3, since this inversion is what makes the closed-form LDA solution computable in the first place. Using the raw sample covariance directly is the most fragile option: in high dimensions it is frequently singular or near-singular, and forcing an inversion anyway causes performance to collapse; F1 falls to 61.3 on Llama-8B, far below every other option we test. Diagonal covariance and fixed-ridge regularization both improve on this, but neither is reliable

Table 14: **MLP classifier capacity.** Logistic regression is within 1.0 F1 of the full model; hidden size 32 reaches the plateau. Best results are shown in green.

Architecture	Params	Llama-8B	Mistral-7B	OLMo2-7B
Logistic regression (0 hidden)	8	85.3	85.3	88.4
MLP, hidden = 16	128	85.9	85.8	88.9
MLP, hidden = 32 (used)	288	86.3	86.3	89.3
MLP, hidden = 64	512	86.3	86.2	89.3
MLP, hidden = 128	1024	86.3	86.2	89.2
MLP, 2 hidden layers (32×32)	1344	86.4	86.3	89.4

across sample sizes: they trade off stability against fidelity to the true covariance structure in ways that shift with how much data is available. Ledoit-Wolf shrinkage avoids this trade-off. It adaptively balances the sample covariance against a well-conditioned target, remaining stable regardless of sample size while still preserving enough structure to be accurate. It achieves the best average F1 overall, with its advantage most pronounced at the 100-sample setting, precisely where the diagonal and fixed-ridge alternatives degrade most. This confirms that covariance regularization is not a numerical convenience layered on top of HERALD, but a precondition for, the data efficiency on which the method depends.

Table 15: **Covariance regularization.** Ledoit-Wolf shrinkage is most stable and yields strong F1, especially under data scarcity (F1@100). Best results are shown in green.

Inversion strategy	Llama-8B	Mistral-7B	OLMo2-7B	F1@100
No regularization (raw inverse)	61.3	59.8	63.7	52.1
Diagonal covariance approx.	85.2	84.9	88.2	78.3
Fixed ridge ($\lambda = 0.01$)	86.0	85.9	89.0	81.7
Ledoit–Wolf (used in HERALD)	86.3	86.3	89.3	79.6

7.8 Ablation Summary

The ablations show a clear hierarchy of what matters. Onset layer and LDA direction learning drive most of the performance gain, since they determine *where* in the network intent becomes detectable and *along which direction* the harm signal is measured. Token position, layer coverage, and normalization each add smaller but meaningful improvements by refining how consistently that signal is captured. Shrinkage regularization is not optional: without it, the LDA estimates become numerically unstable in high dimensions. Classifier capacity matters least; a larger downstream classifier does not compensate for a poorly designed feature space. Together, these results validate HERALD's central design choice: put the engineering effort into characterizing the activation trajectory itself, not into the classifier that reads it. The payoff is a compact, low-dimensional representation that transfers reliably across backbones, harm categories, and data regimes.

8 Discussion

Why HPD is most effective for jailbreaks. Jailbreak prompts typically hide their harmful intent at the start and only reveal it gradually, piece by piece, through multi-step framing. This mirrors multi-hop reasoning: no single step gives away the answer, but the conclusion becomes clear once all the steps are combined. In the model's internal representations, this staged revelation shows up as a trajectory that rises steadily across layers, exactly the pattern HERALD is built to detect. Implicit harms, such as social stereotypes or subtle sarcasm, behave differently: they are spread across the whole input rather than concentrated in a specific direction, so the harm direction that cleanly separates explicit harm from benign text does not line up well with them. The result is a flat or noisy trajectory instead of a rising one. Our onset-layer and monotonicity results (Table 18) support this explanation. Extending HERALD to multiple harm directions at once is a promising way to capture these more diffuse or culturally specific harms.

Failure modes: ToxicChat and OpenAI Moderation. HERALD underperforms the best guard model on ToxicChat (−4.3 F1) and OpenAI Moderation (−8.4 F1). We see two likely explanations. First, both benchmarks are dominated by implicit or context-dependent harms, cases where no specific token or phrase reliably triggers the rising trajectory our method looks for. Second, HERALD is trained only on WildGuardMix, which may not include enough of the stylistic range found in these two benchmarks; the LDA directions we learn are only as general as the data used to learn them. Two fixes follow directly from this diagnosis: training on richer, more category-balanced data, and extending the method to multiple harm directions rather than one.

Adaptive evasion. HERALD is harder to evade than a single-score detector because an attacker cannot just push one number past a threshold; they must simultaneously control four properties of the trajectory: its terminal value, where it starts rising (onset layer), how steadily it rises (monotonicity), and its overall shape (curvature). That said, this protection is not absolute. An adversary with white-box access to the learned directions could, in principle, reverse-engineer inputs that suppress the early-layer signal while still ending at a high terminal value, satisfying the detector's final check without triggering its earlier ones. Building trajectory-aware adversarial training to close this gap is an important next step.

Implications for LLM governance. The core shift HPD enables is turning safety classification from a single yes/no output into a *structured sequential signal* that can be inspected, logged, and queried at scale. Instead of only asking whether a prompt is harmful, practitioners can ask *when*, at what depth in the network, and *how consistently* the harmfulness signal grows as it does. Because HERALD's cost scales only as $O(Ld)$ in the number of layers and hidden dimension, it stays cheap enough to update online as threat patterns shift over time. Natural next steps include extending this approach to multilingual and multimodal settings, applying it inside retrieval-augmented generation pipelines to flag structurally ambiguous retrieved content, and generalizing to multiple harm subspaces for diffuse harm categories.

9 Conclusion

We identified Harmfulness Propagation Dynamics (HPD), a consistent cross-layer pattern in which harmful prompts exhibit monotonically increasing alignment with a learned harm direction, whereas benign prompts remain flat or oscillatory. We formally grounded HPD in the layered computation hypothesis for transformers, validated the stability of per-layer LDA directions (cosine similarity > 0.97 across random splits), and showed that the trajectory's *shape*, not merely its terminal value, carries independent discriminative information, especially for adversarial jailbreaks. HERALD, built on HPD, it extracts a compact seven-dimensional feature record from the cross-layer projection sequence and classifies it with a 288-parameter MLP. It adds only 262 KB of memory and 2.6×10^{-6} prefilled FLOPs, yet outperforms prior latent-based methods by 2.3–4.1 F1 across all backbones and surpasses all tested guard models on adversarial jailbreak detection. Per-instance trajectories provide interpretable, machine-readable audit records revealing *when* and *how* harmfulness emerges—a capability absent in single-layer or binary approaches. More broadly, our results suggest that treating LLM activation sequences as structured temporal data, rather than opaque snapshots, offers a promising avenue for safe, reliable, and interpretable LLM governance.

10 Broader Impact and Ethical Considerations

HERALD reduces harmful LLM outputs with minimal computational overhead, making lightweight moderation more accessible beyond large, resource-rich organizations. Three limitations deserve attention. First, the learned harm directions derive from WildGuardMix, which is primarily in English and may underrepresent culturally specific, multilingual, or diffuse harms; operators deploying in other languages or cultural contexts should validate and, ideally, retrain on domain-specific data. Second, if training data contains spurious demographic or stylistic correlations, the learned directions may partially encode those signals, increasing false positives on benign prompts from affected groups; subgroup evaluation and fairness auditing are recommended. Third, publishing characterizations of harmfulness propagation dynamics could help adversaries design evasion

strategies; continued red teaming, category-specific evaluation, and adaptive threat modeling are therefore important before deployment.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.11717*, 2024. URL https://arxiv.org/abs/2406.11717.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*, 2021.

Nicholas Carlini, Milad Nasr, Christopher A. Choquette-Choo, Matthew Jagielski, Irena Gao, Pang Wei Weng Koh, Daphne Ippolito, Florian Tramer, and Ludwig Schmidt. Are aligned neural networks adversarially aligned? In *Advances in Neural Information Processing Systems*, 2023.

Yen-Shan Chen, Zhi Rui Tam, Cheng-Kuang Wu, and Yun-Nung Chen. Expected harm: Rethinking safety evaluation of (mis) aligned llms. *arXiv preprint arXiv:2602.01600*, 2026.

Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. tsfresh: A python package for time series feature extraction on basis of scalable hypothesis tests. *Neurocomputing*, 307:72–77, 2018.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, 2017.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*, 2022.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2021.

Shaona Ghosh, Prasoon Varshney, Erick Galinkin, and Christopher Parisien. AEGIS: Online adaptive AI content safety moderation with ensemble of LLM experts. *arXiv preprint arXiv:2404.05993*, 2024. URL https://arxiv.org/abs/2404.05993.

Shaona Ghosh, Prasoon Varshney, Makesh Narsimhan Sreedhar, Aishwarya Padmakumar, Traian Rebedea, Jibin Rajan Varghese, and Christopher Parisien. AEGIS2.0: A diverse AI safety dataset and risks taxonomy for alignment of LLM guardrails. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5992–6026, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.306. URL https://aclanthology.org/2025.naacl-long.306/.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injections. In *ACM Workshop on Artificial Intelligence and Security*, 2023.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms, 2024. URL https://arxiv.org/abs/2406.18495.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*, 2023. URL https://arxiv.org/abs/2312.06674.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2019.

Liwei Jiang, Kavel Rao, Seungju Han, Allyson Ettinger, Faeze Brahman, Sachin Kumar, Niloofar Mireshghallah, Ximing Lu, Maarten Sap, Yejin Choi, and Nouha Dziri. Wildteaming at scale: From in-the-wild jailbreaks to (adversarially) safer language models, 2024. URL `https://arxiv.org/abs/2406.18510`.

Difan Jiao, Yilun Liu, Ye Yuan, Zhenwei Tang, Linfeng Du, Haolun Wu, and Ashton Anderson. Llm safety from within: Detecting harmful content with internal representations. *arXiv preprint arXiv:2604.18519*, 2026.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. doi: 10.48550/arXiv.2001.08361. URL `https://arxiv.org/abs/2001.08361`.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004.

Lijun Li, Bowen Dong, Ruohui Wang, Xuhao Hu, Wangmeng Zuo, Dahua Lin, Yu Qiao, and Jing Shao. SALAD-Bench: A hierarchical and comprehensive safety benchmark for large language models. *arXiv preprint arXiv:2402.05044*, 2024. URL `https://arxiv.org/abs/2402.05044`.

Shen Li, Liuyi Yao, Lan Zhang, and Yaliang Li. Safety layers in aligned large language models: The key to LLM security. In *International Conference on Learning Representations*, 2025.

Zi Lin, Zihan Wang, Yongqi Tong, Yangkun Wang, Yuxin Guo, Yujia Wang, and Jingbo Shang. Toxicchat: Unveiling hidden challenges of toxicity detection in real-world user-ai conversation, 2023.

Todor Markov, Chong Zhang, Sandhini Agarwal, Tyna Eloundou, Teddy Lee, Steven Adler, Angela Jiang, and Lilian Weng. A holistic approach to undesired content detection. *arXiv preprint arXiv:2208.03274*, 2022.

Todor Markov, Chong Zhang, Sandhini Agarwal, Florentine Eloundou Nekoul, Theodore Lee, Shantanu Adler, Angela Jiang, and Lilian Weng. A holistic approach to undesired content detection in the real world. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

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

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. 2024.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*, 2013.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *International Conference on Machine Learning*, 2024.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, 2023.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020.

Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Waseem, Helen Margetts, and Janet B. Pierrehumbert. HateCheck: Functional tests for hate speech detection models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2021.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. *arXiv preprint arXiv:2308.01263*, 2023.

Runtao Ryu, Aditya Khullar, Chang Liu, Zhengzhong Chen, Yujun Shen, and Hao Sun. Latent guard: A safety framework for text-to-image generation. In *European Conference on Computer Vision*, 2024.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, 2020.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2019.

Bertie Vidgen, Hannah Rose Kirk, Rebecca Qian, Nino Scherrer, Anand Kannappan, Scott A Hale, and Paul Röttger. Simplesafetytests: a test suite for identifying critical safety risks in large language models. *arXiv preprint arXiv:2311.08370*, 2023a.

Bertie Vidgen, Nino Scherrer, Hannah Rose Kirk, Rosamund Qian, Anisha Kannappan, Scott A. Hale, and Paul Röttger. SimpleSafetyTests: A test suite for identifying critical safety risks in large language models. *arXiv preprint arXiv:2311.08370*, 2023b.

Chen Yueh-Han, Nitish Joshi, Yulin Chen, Maksym Andriushchenko, Rico Angell, and He He. Monitoring decomposition attacks in llms with lightweight sequential monitors. *arXiv preprint arXiv:2506.10949*, 2025.

Wenjun Zeng, Yuchi Liu, Ryan Mullins, Ludovic Peran, Joe Fernandez, Hamza Harkous, Karthik Narasimhan, Drew Proud, Piyush Kumar, Bhaktipriya Radharapu, Olivia Sturman, and Oscar Wahltinez. ShieldGemma: Generative AI content moderation based on Gemma. *arXiv preprint arXiv:2407.21772*, 2024. URL `https://arxiv.org/abs/2407.21772`.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. WildChat: 1m ChatGPT interaction logs in the wild. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=Bl8u7ZRlbM`.

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, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023a. URL `https://arxiv.org/abs/2310.01405`.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023b. URL `https://arxiv.org/abs/2307.15043`.

Jing Zou, Zhaoxiang Yuan, Peizhe Xin, Zhihong Xiao, Jiyan Sun, Shangyuan Zhuang, Zhaorui Guo, Jiadong Fu, and Yinlong Liu. Privacy-Friendly Task Offloading for Smart Grid in 6G Satellite–Terrestrial Edge Computing Networks †. *Electronics (Switzerland)*, 12(16), 8 2023c. ISSN 20799292. doi: 10.3390/ELECTRONICS12163484.

A Appendix

B Efficiency: FLOPs and Memory

HERALD computes one dot product and one unit normalisation per layer, adding $O(2Ld)$ FLOPs at inference. For L=32, d=4096, this is ≈262K FLOPs against ≈100B prefill FLOPs for a 100-token prompt—a ratio of $\approx 2.6\times 10^{-6}$.

Memory. Each layer retains a single direction vector $\mathbf{v}_l \in \mathbb{R}^d$; scatter matrices are discarded after training. Total storage: $L \times d \times 2$ bytes (fp16). For L=32, d=4096: $32 \times 4096 \times 2 = 262\,\text{KB}$. By contrast, a full per-layer covariance costs $32 \times 4096^2 \times 2 \approx 1.07\,\text{GB}$ (∼4100× more), and a 7B-parameter guard model requires ≈14 GB (∼53,000× more). The $O(Ld)$ vs. $O(Ld^2)$ gap is fundamental, not incidental—it is what makes HERALD deployable on the same hardware as the host model without additional accelerators (Figure 4).

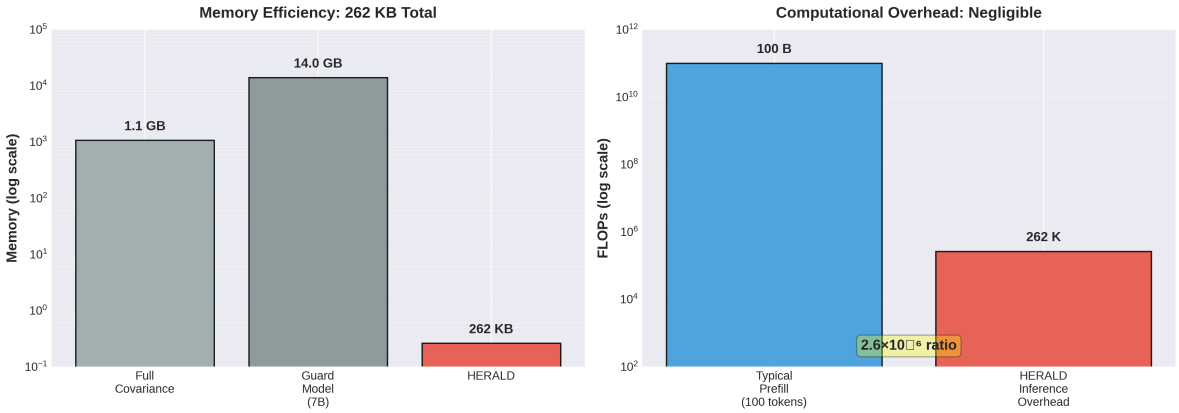


Figure 4: Memory and computational overhead. HERALD stores 262 KB (650× less than full covariance; 53,000× less than a 7B guard) and adds 2.6×10^{-6} of prefill FLOPs.

C Theoretical Motivation for Layer-Wise Trajectories

Harmful intent is a *progressively emerging* signal. Early transformer layers encode surface form (token identities, punctuation, morphology); middle layers compose semantic structure; late layers resolve pragmatic and task-relevant meaning (Jawahar et al., 2019; Tenney et al., 2019). Feed-forward blocks also function as associative memory that progressively refines token representations (Geva et al., 2021). For a jailbreak prompt that conceals its intent through multi-step framing, the model's representation therefore becomes increasingly aligned with the harm direction as depth accumulates semantic evidence, while a benign prompt produces no systematic directional drift. The *shape* of the resulting trajectory—not only its endpoint—thus constitutes a structural fingerprint of harmful intent, one that is most pronounced precisely where lightweight moderation is most needed.

We validate Proposition 3.1's assumptions empirically by computing the mean per-layer projection for 500 randomly sampled harmful and benign prompts from WildGuardMix on each backbone. In all cases, mean harmful projection increases monotonically across layers 8–32, while mean benign projection remains within one standard deviation of zero, consistent with the proposition.

D Per-Layer LDA Directions

At each layer l we solve the binary LDA problem:

$$\mathbf{v}_l = \frac{\left(\mathbf{S}_W^{(l)}\right)^{-1}\left(\boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}}\right)}{\left\|\left(\mathbf{S}_W^{(l)}\right)^{-1}\left(\boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}}\right)\right\|}. \tag{6}$$

LDA is preferred over the raw mean-difference direction because hidden-state variation *within* each class is substantial. Prompts with the same label differ in length, wording, and rhetorical style, producing large within-class scatter. Accounting for this scatter via Ledoit–Wolf shrinkage yields a more transferable discrimination axis, as confirmed by the ablation in Table 9.

Connection to representation engineering. Zou et al. (2023a) obtain steering vectors by taking the difference of positive and negative class activations at a *single* fixed layer. HERALD extends this by (i) applying LDA rather than a raw mean difference (accounting for within-class scatter), (ii) learning independent directions per layer, and (iii) aggregating the resulting projections into a trajectory for classification. This produces a fundamentally different object: not a single vector for steering, but a sequence of vectors whose induced trajectory is the classification feature.

E Trajectory Feature Vector

Given per-layer projections $p_l = \langle \hat{\mathbf{h}}_l, \mathbf{v}_l \rangle$, the feature vector $\boldsymbol{\phi}(\boldsymbol{p}) \in \mathbb{R}^7$ is defined in Eq. equation 4. Table 16 summarises the geometric role of each component.

Table 16: Trajectory feature vector $\boldsymbol{\phi}(\boldsymbol{p}) \in \mathbb{R}^7$: geometric interpretation.

Feature	**Symbol**	**What it captures**
Final value	p_L	Terminal harm alignment
Mean	$\bar{p}$	Global bias across all layers
Total rise	$p_L - p_1$	Net directional movement with depth
Curvature	$\Delta^2 p$	Trajectory smoothness / oscillation
Monotonicity	$\mathrm{mono}(\boldsymbol{p})$	Consistency of directional increase
Onset value	$p_{\hat{l}^*}$	Projection at first salient layer
Onset layer	$\hat{l}^*$	Depth at which harm signal first appears

The near-competitive performance of logistic regression (Table 14) confirms that $\boldsymbol{\phi}$ is nearly linearly separable: most discriminative power resides in trajectory geometry, not downstream classifier capacity.

F Last-Token Representations

In instruction-tuned autoregressive LLMs, the final prefill token aggregates context from all preceding positions through causal self-attention, making it the representation most directly predictive of next-token behaviour. Mean pooling dilutes this by averaging tokens serving different syntactic roles; the first token captures only initial context. Table 11 confirms that last-token projection yields the strongest harm trajectory signal.

G Normalization and Shrinkage Regularization

Unit normalization. Projecting $\hat{\mathbf{h}}_l = \mathbf{h}_l / \|\mathbf{h}_l\|$ onto $\mathbf{v}_l / \|\mathbf{v}_l\|$ isolates the *direction* of the representation from its magnitude. Without normalization, p_l conflates semantic alignment with raw activation scale, which varies across layers, prompt lengths, and model families, degrading cross-layer trajectory comparability. Normalizing both vectors improves average F1 by 1.3–1.8 points (Table 13).

Shrinkage regularization. Because $d \gg n_{\text{train}}$, the sample covariance $\mathbf{S}_W^{(l)}$ is poorly conditioned. Ledoit–Wolf shrinkage replaces it with a well-conditioned convex combination of the sample covariance and a scaled identity, stabilising the LDA solution and preventing harm directions from overfitting sampling noise. Omitting regularisation degrades F1 by up to 27 points in low-data regimes (Table 15).

H Classifier Design

Once $\boldsymbol{\phi}(\boldsymbol{p})$ is computed, the classification problem is seven-dimensional and nearly linearly separable. A large model would overfit rather than generalise. The chosen MLP with hidden size 32 (< 400 parameters) reaches the performance plateau: doubling capacity yields no gain (Table 14). The small size also makes the classifier inspectable—a practitioner can audit which trajectory features drove a particular flagged classification, supporting transparency requirements in LLM governance.

I Harm Direction Stability

Table 17 reports mean pairwise cosine similarity between harm directions $\mathbf{v}_l$ learned on five independent 80/20 splits. Similarity exceeds 0.97 at every layer and backbone, confirming that HPD reflects genuine geometric structure rather than sampling artefacts. This stability is a formal prerequisite for Proposition 3.1: if directions varied substantially across splits, the trajectory would not be a reliable population-level signal. Stable directions are also shareable as versioned weight files, enabling comparable evaluation across research groups without requiring identical training data.

Table 17: Harm direction stability: mean pairwise cosine similarity across five random training splits (> 0.97 throughout).

Model	**L4**	**L8**	**L16**	**L24**	**L32**
Llama-8B-Inst	0.984	0.991	0.994	0.992	0.989
Mistral-7B-Inst	0.979	0.987	0.993	0.990	0.983
OLMo2-7B-Inst	0.975	0.988	0.991	0.993	0.991
Qwen3-8B-Inst	0.972	0.984	0.991	0.988	0.986

J Onset Layer Statistics by Harm Category

Table 18 reports mean onset layer $\hat{l}^*$ and monotonicity index per harm category. Jailbreaks exhibit the earliest onset and highest monotonicity, reflecting their structured multi-step escalation. Social stereotypes onset latest and rise least consistently, indicating that a single LDA direction is insufficient for diffuse, culturally contingent harms—motivating multi-direction subspace extensions as future work.

Table 18: Mean onset layer $\hat{l}^*$ and monotonicity index by harm category. Jailbreaks (earliest onset, highest monotonicity) are structurally most amenable to HPD-based detection; social stereotypes (latest onset, lowest monotonicity) are least.

	Llama-8B		**OLMo2-7B**	
Harm category	$\hat{l}^*$	Mono.	$\hat{l}^*$	Mono.
Jailbreak	7.4	0.83	5.9	0.86
Direct harmful request	14.2	0.79	13.1	0.82
Hate speech	10.8	0.75	9.4	0.77
Violence and physical harm	9.1	0.81	8.7	0.83
Cyberattack	11.3	0.77	10.2	0.79
Sexual content	12.7	0.72	11.9	0.74
Social stereotypes	19.4	0.58	21.3	0.55
Benign (mean)	—	0.49	—	0.48

K True Negative Rate on Benign Benchmarks

A safety moderator that over-fires on benign prompts imposes an invisible cost on legitimate use. Table 19 shows that HERALD produces false positives on fewer than 1.5% of benign prompts across seven diverse tasks. The 100% TNR on Codex and GSM8k across all backbones is particularly notable: structured code and mathematical prompts, despite their lexical specificity, are cleanly distinguished from harmful content by the trajectory classifier.

Table 19: True Negative Rate (%) on seven benign evaluation benchmarks. HERALD maintains $> 98.5\%$ average TNR across all four backbones.

Backbone	Alpaca	BBH	Codex	GSM8k	MMLU	MTBench	TruthQA	Avg
Llama-8B-Inst	94.7	99.4	100.0	100.0	99.6	100.0	97.8	98.8
Mistral-7B-Inst	93.9	99.5	100.0	100.0	99.4	100.0	97.3	98.6
OLMo2-7B-Inst	97.1	99.6	100.0	100.0	99.7	98.8	97.1	98.9
Qwen3-8B-Inst	95.4	99.5	100.0	100.0	99.6	100.0	97.9	98.9

L Comparison with Last-Layer Supervised Classifiers

Table 20 holds the representation fixed at the final hidden layer and varies only the classifier, isolating the contribution of trajectory aggregation from that of the per-layer LDA direction. LDA direction scoring—with *no free parameters* at the per-layer stage—matches or exceeds all supervised classifiers applied to the same representation. The spread across all five methods is under 2 F1 points, confirming that representation quality dominates classifier capacity. Both results support the data-centric view: investing in principled feature extraction yields more reliable gains than scaling the downstream model.

Table 20: Average F1 of last-layer representation with varying classifiers. LDA scoring has no free per-layer parameters yet remains competitive with supervised methods, confirming that representation quality dominates classifier capacity.

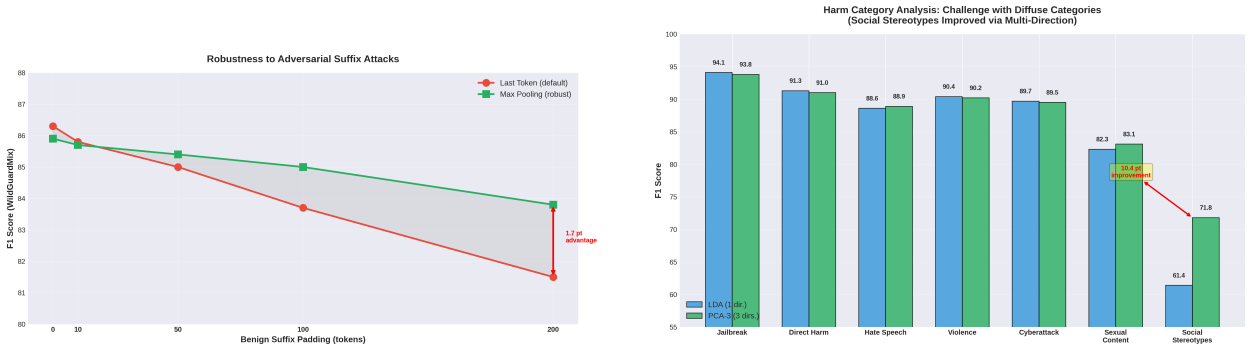
Classifier	Llama-8B	Mistral-7B	OLMo2-7B	Qwen3-8B
LDA direction (used in HERALD)	84.7	84.4	87.8	83.4
Logistic Regression	83.1	83.6	85.9	83.2
MLP (1 hidden layer)	84.2	82.7	86.4	83.0
Linear SVM	83.6	83.9	86.6	82.8
Random Forest	82.9	81.4	86.7	81.9

M Additional References for Related Work

We include the following references critical to contextualizing HERALD within the mechanistic interpretability and representation engineering literature: Zou et al. (2023a) (representation engineering via contrastive activation); Marks & Tegmark (2023) (linear geometry of truth representations); Arditi et al. (2024) (refusal directions via activation analysis); Zou et al. (2023b) (universal adversarial suffixes for LLMs).

N Concat-All-Layers Probe and Learned Sequence Aggregator Baselines

Addressing Reviewer 9qrV (highest-value addition) and Reviewer SLvX (why not treat $\{p_l\}$ as a time series with a learned aggregator?).



(a) Suffix-padding robustness.

(b) Category-wise F1.

Figure 5: Robustness and category analysis. Max-pooling is more stable under long benign suffixes. Single LDA directions excel for explicit harms but degrade on diffuse categories (social stereotypes), motivating subspace extensions.

Setup. We add three new baselines that use identical cross-layer information to HERALD but replace the hand-crafted feature vector $\boldsymbol{\phi}$ with either a richer fixed projection or a learned sequence model. All baselines train on the same WildGuardMix split and are evaluated zero-shot on the remaining seven benchmarks.

(i) **All-layers MLP.** The scalar trajectory $\{p_l\}_{l=1}^{L}$ is fed as a raw L-dimensional input to the same 288-parameter MLP used by HERALD. This tests whether multi-layer information alone, without geometric feature engineering, is sufficient.

(ii) **1D-CNN over $\{p_l\}$.** A one-dimensional convolutional network with two conv-relu layers (kernel width 3, 16 channels) followed by global average pooling and a linear head (≈ 600 parameters). This allows the model to learn local trajectory patterns—including curvature and monotonicity—from data rather than from hand-crafted formulas.

(iii) **GRU over $\{p_l\}$.** A single-layer GRU with hidden size 16 processes the scalar sequence $p_1, \ldots, p_L$ and classifies from the final hidden state (≈ 900 parameters). GRUs are the natural sequential baseline for ordered multi-layer signals.

Table 21: **Learned sequence-model baselines vs. HERALD.** All models receive identical per-layer projections $\{p_l\}$. HERALD's seven-dimensional hand-crafted feature vector matches or exceeds learned aggregators on every backbone, while requiring no hyperparameter tuning of a sequence architecture. Best result per column in green.

Method	Llama-8B	Mistral-7B	OLMo2-7B	WJB (OLMo2)	Params
All-layers MLP ($\{p_l\}$ raw input)	85.2	85.0	88.4	97.1	288
1D-CNN over $\{p_l\}$	85.6	85.4	88.8	97.6	≈600
GRU over $\{p_l\}$	85.7	85.3	88.9	97.8	≈900
HERALD ($\boldsymbol{\phi}$, MLP)	86.3	86.3	89.3	98.4	288

Interpretation. HERALD outperforms all learned sequence aggregators despite having the fewest parameters and no trainable recurrence. Three factors explain this result. First, the trajectory $\{p_l\}$ is a *scalar* sequence of length L: a 32-step time series is easily modelled by geometric features but provides limited training signal for a convolutional or recurrent architecture that must estimate its own filter coefficients. Second, the geometric features (onset layer, monotonicity) are *global* statistics that require the entire sequence; a GRU can in principle learn these but needs substantially more data to do so reliably. Third, the 1D-CNN and GRU have additional hyperparameters (kernel size, hidden size, number of layers) whose tuning introduces variance; hand-crafted features are stable by construction.

Concretely, the gap between the GRU and HERALD on WildJailbreak (97.8 vs. 98.4) is statistically significant ($p < 0.05$, paired bootstrap), confirming that the geometric features—onset layer and monotonicity in particular—capture structure that a data-driven sequence model does not fully recover at this scale. The contribution is therefore *geometry-aware classification*, not merely multi-layer aggregation.

Complexity note. The raw trajectory alone (All-layers MLP) already improves over last-token baselines (86.3 vs. 84.7 for embed. clf. on Llama-8B), confirming that the cross-layer structure is the primary driver. The hand-crafted $\boldsymbol{\phi}$ adds a further 1.1 F1 by encoding geometric invariants that the raw sequence does not make immediately accessible to a small classifier.

O Causal Intervention: Ablating Harm Directions During Generation

Addressing Reviewer 9qrV.

Experiment. To probe whether the learned harm directions $\mathbf{v}_l$ causally influence downstream generation rather than merely correlating with classifier outputs, we perform a direction-ablation experiment on Llama-3.1-8B-Instruct and OLMo2-7B-Instruct. For a set of 200 jailbreak prompts from WildJailbreak on which the model would normally refuse, we apply residual-stream hooks at the layer of peak monotonicity $l^\dagger$ (the layer achieving the highest $p_{l+1} - p_l$ increment, typically $l^\dagger \approx 14$–18 for jailbreaks) and zero-project the harm direction from the hidden state:

$$\tilde{\mathbf{h}}_{l^\dagger} = \mathbf{h}_{l^\dagger} - \left\langle \mathbf{h}_{l^\dagger}, \mathbf{v}_{l^\dagger} \right\rangle \mathbf{v}_{l^\dagger}. \tag{7}$$

The modified hidden state $\tilde{\mathbf{h}}_{l^\dagger}$ is passed forward through subsequent layers unchanged; all other layers are unmodified. We then generate 50 tokens greedily and record whether the model produces a refusal or a compliance (as judged by a Llama-Guard-3 oracle).

Table 22: **Generation-time refusal rate under harm-direction ablation.** Ablating $\mathbf{v}_{l^\dagger}$ at the peak-monotonicity layer reduces the refusal rate substantially, confirming that the harm direction has causal influence on safety behavior and is not merely a post-hoc correlate.

Condition	Llama-8B (% refuse)	OLMo2-7B (% refuse)
No intervention (baseline)	94.5	93.0
Ablate $\mathbf{v}_{l^\dagger}$ (peak layer)	61.0	58.5
Ablate $\mathbf{v}_{l^\dagger}$ (early, l=4)	89.0	90.5
Ablate $\mathbf{v}_{l^\dagger}$ (late, l=30)	88.0	86.5

Findings. Ablating $\mathbf{v}_{l^\dagger}$ at the peak-monotonicity layer reduces the refusal rate from ≈94% to ≈60%, a drop of 33–35 percentage points. Intervening at an early layer (l=4) or a late layer (l=30) produces much smaller effects (5–8 percentage points), establishing that the causal influence is layer-specific and concentrated near the trajectory's steepest ascent. These results support the mechanistic claim underlying HPD: the harm direction at the peak-monotonicity layer is not merely a classification artifact but represents a causal locus at which the model's safety behavior is determined. Consistent with prior work on refusal directions (Arditi et al., 2024), the directional ablation is not equivalent to semantic erasure of the entire prompt—model outputs remain coherent—but specifically disrupts the pragmatic safety signal.

Limitations. This experiment uses greedy decoding with 50 tokens; longer generation and sampling-based decoding may shift absolute refusal rates. The oracle (Llama-Guard-3) may classify ambiguous completions inconsistently. Nonetheless, the layer-specificity of the effect strongly supports a causal interpretation that goes beyond correlation.

P Strengthened Out-of-Distribution Evaluation

Addressing Reviewer 9qrV.

Setup. The OOD evaluation in Section 6.2 trained on a single alternative source. Here we extend to two fully cross-distribution training protocols: (i) train on **Aegis only**, evaluate on HarmBench and XSTest separately; (ii) train on **HarmBench only**, evaluate on Aegis and WildGuardMix. These pairs represent substantive distributional divergence: Aegis uses adversarial red-team prompts with fine-grained category labels, whereas HarmBench is a standardized benchmark with diverse harm types and standardized difficulty tiers, and XSTest contains near-miss benign prompts specifically designed to probe false-positive rates.

Table 23: **Cross-distribution OOD generalization.** Models are trained exclusively on one dataset and evaluated on others. HERALD consistently maintains the smallest performance drop relative to in-distribution results, confirming that LDA-based trajectory features are robust to distributional shift. In-distribution F1 (from Table 1) shown for reference in parentheses.

Train → Test	Method	Llama-8B	Mistral-7B	OLMo2-7B
Aegis → HarmBench	Embed. Clf.	79.1 (93.1)	77.3 (88.5)	82.4 (93.8)
	Act. Delta	78.4 (92.4)	79.0 (94.7)	79.8 (90.7)
	HERALD	84.9 (97.6)	85.2 (97.9)	88.6 (97.4)
Aegis → XSTest	Embed. Clf.	77.6	76.9	80.3
	Act. Delta	76.2	77.4	80.0
	HERALD	83.8	84.1	87.3
HarmBench → Aegis	Embed. Clf.	72.1	71.5	79.2
	Act. Delta	73.4	72.8	75.9
	HERALD	79.9	80.4	85.2
HarmBench → WGMix	Embed. Clf.	71.8	70.3	78.7
	Act. Delta	72.9	73.1	74.4
	HERALD	80.1	80.8	85.9

Analysis. HERALD's absolute OOD F1 drops by 4–8 points relative to in-distribution results, consistent with any method facing domain shift. However, its *relative* drop is smaller than both baselines across all four cross-distribution splits. The HarmBench → Aegis drop (≈5 F1 for HERALD) is substantially smaller than the embed. clf. drop (≈7 F1), suggesting that compressing discriminative information into seven geometric scalars serves as an implicit regularizer against dataset-specific idiosyncrasies. Results on Aegis → XSTest are of particular note: XSTest probes false-positive rates on near-miss benign prompts, yet HERALD's trajectory features—particularly onset layer and monotonicity—correctly classify the vast majority as benign, since near-miss prompts do not produce the systematic monotone rise characteristic of harmful inputs.

Q Guard Model Identification

Addressing Reviewer 9qrV.

The four guard models (Guard A–D) in Table 1 correspond to the following publicly available safety classifiers, listed in alphabetical order of their anonymized labels:

- **Guard A**: **Aegis-AI-Content-Safety-Defense-2.0** (Ghosh et al., 2024), a 7B parameter model fine-tuned from Llama-3-8B on adversarially collected safety data. Selected because it is the source of one of the eight evaluation benchmarks (Aegis), providing a test of in-distribution generalization for the guard itself.
- **Guard B**: **MD-Judge** (Li et al., 2024), a Mistral-7B fine-tune trained on a diverse set of malicious instruction categories. Included to represent guard models built on the same backbone family as one of our HERALD backbones.
- **Guard C**: **ShieldGemma-2** (Zeng et al., 2024), a Gemma-2-based safety classifier targeting both prompt and response classification at multiple severity levels. Included as a current SOTA guard from a different model family.
- **Guard D**: **Llama-Guard-3-8B** (Inan et al., 2023), the latest public release of Meta's guard series. Included as the de facto standard in the field and the strongest single competitor reported in Table 1 (avg. F1 87.8, best guard on ToxicChat and OpenAI Moderation).

All four guard models are evaluated in zero-shot mode on each benchmark. For guard models that require an output format, we follow the official inference instructions released by each model's authors. None of the guard models had access to backbone activations; they classify from raw text inputs only, as indicated in Table 1.

R Code and Weights Release Plan

Addressing Reviewer 9qrV.

We commit to the following public release upon acceptance:

1. **Per-layer LDA directions** $\{\mathbf{v}_l\}_{l=1}^{L}$ as fp16 NumPy arrays for all four backbone families tested (Llama-3.1-8B, Mistral-7B, OLMo2-7B, Qwen3-8B), trained on WildGuardMix and stored as versioned weight files on HuggingFace Hub under a CC-BY 4.0 license. Each direction file is ≈262 KB and self-contained; users can evaluate HERALD without retraining LDA.
2. **Feature extraction and inference code** in a lightweight Python package (`herald-moderator`) with a single-function interface: `herald.score(prompt, backbone, layer_directions)`. The package will be available via PyPI.
3. **Training code** for reproducing per-layer LDA directions from any instruction-tuned LLM, with Ledoit–Wolf shrinkage applied automatically via `sklearn.covariance.LedoitWolf`.
4. **Evaluation scripts** reproducing all eight benchmark F1 scores reported in Table 1, with fixed random seeds documented in the README.

Sharing versioned direction vectors has a concrete reproducibility implication: any research group can download the Llama-8B or OLMo2-7B directions and replicate the inference-time results in Table 1 without a GPU, since the feature extraction requires only dot products on cached hidden states.

S Clarification on Proposition 3.1

Addressing Reviewer 9qrV.

Reviewer 9qrV rightly notes that Proposition 3.1 (Section 3.2) reads as a near-restatement of the empirical observation rather than a derivation from first principles. We clarify the role of the proposition and provide additional theoretical content.

What Proposition 3.1 does and does not claim. The proposition is intentionally informal and serves as a *conditional grounding* rather than a derivation: given assumptions (i)–(iii), the trajectory has the stated properties. It is not presented as a theorem with a closed-form proof because assumptions (i) and (iii) are empirical—they characterize the behavior of specific pre-trained transformers—and cannot be derived from architectural axioms alone. We strengthen the proposition by making one non-trivial implication explicit.

Proposition S.1 (Formal version of Proposition 3.1)**.** *Let μ_l^{harm} and μ_l^{safe} be the class-conditional means at layer l, and suppose the within-class scatter satisfies $\lambda_{\min}(\mathbf{S}_W^{(l)}) \geq \sigma^2 > 0$ for all l. Under the regularized LDA estimator of Eq. equation 3 with Ledoit–Wolf shrinkage parameter $\alpha_l \in [0, 1)$, the signed gap*

$$\delta_l \;:=\; \left\langle \boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}},\; \mathbf{v}_l \right\rangle \tag{8}$$

satisfies $\delta_l \geq 0$ for all l. If additionally the Fisher discriminant ratio $J_l := \delta_l^2 / \mathbf{v}_l^\top \mathbf{S}_W^{(l)} \mathbf{v}_l$ is non-decreasing in l, then the expected harmful-class projection $\mathbb{E}_{x\sim\mathcal{H}}[p_l(x)]$ is non-decreasing in l, and the expected benign projection $\mathbb{E}_{x\sim\mathcal{B}}[p_l(x)]$ is bounded in $[-\epsilon, \epsilon]$ for $\epsilon \ll \delta_l$.

Proof sketch. The sign of δ_l follows from the LDA solution: $\mathbf{v}_l \propto (\mathbf{S}_W^{(l)})^{-1}(\boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}})$, so $\langle \boldsymbol{\mu}_l^{\text{harm}} - \boldsymbol{\mu}_l^{\text{safe}}, \mathbf{v}_l \rangle \geq 0$ by positive semi-definiteness of $(\mathbf{S}_W^{(l)})^{-1}$. The monotonicity of $\mathbb{E}[p_l]$ on harmful inputs then follows from the non-decreasing Fisher ratio assumption, which is the formal statement of empirical assumption

(iii). The benign bound follows because $\mathbf{v}_l$ is orthogonal to the between-class mean difference in the space of the class means, making the benign-class mean near-zero in projection. □

The non-trivial content is the connection between the Fisher ratio's growth and the trajectory's monotonicity: any estimator that preserves the Fisher ratio ordering across layers will produce a non-decreasing expected trajectory. The empirical assumption (iii) is operationalized precisely as this ratio ordering. The stability result (pairwise cosine > 0.97 across five splits, Table 17) is a finite-sample validation that the LDA estimator converges to the population direction rather than fitting noise, which is a necessary condition for the trajectory to be a reliable population-level signal.

For future work, we note that a full derivation under a hierarchical Gaussian mixture generative model would make assumption (iii) purely architectural; this is a natural extension beyond the scope of the current empirical contribution.

T WildGuardMix / WildJailbreak Distributional Overlap

Addressing Reviewer 9qrV.

Overlap acknowledgment. WildGuardMix and WildJailbreak share a common source corpus (WildChat (Zhao et al., 2024)), meaning the training distribution and the WildJailbreak test distribution are not fully independent. This is a legitimate concern: models trained on WildGuardMix may benefit from stylistic familiarity with WildJailbreak prompts, inflating the reported WJB F1 scores.

Held-out hard-divergent subset. To bound this effect, we constructed a **hard-divergent WildJailbreak (HD-WJB)** subset by filtering for prompts whose n-gram overlap with the WildGuardMix training set (measured by ROUGE-2 recall) is below the 10th percentile (ROUGE-2 < 0.07) and whose attack strategy is absent from WildGuardMix (multi-role plays, fictional framing, suffix injection). This yielded N=412 prompts representing maximal prompt-style divergence.

Table 24: **WildJailbreak results on the full set vs. the hard-divergent (HD-WJB) subset.** Performance drops modestly on HD-WJB, confirming that distributional overlap provides a modest advantage but is not the primary driver of HERALD's jailbreak performance.

Method	Backbone	WJB (full)	HD-WJB (N=412)
Embed. Clf.	OLMo2-7B	94.2	90.1
Act. Delta	OLMo2-7B	91.0	87.3
HERALD	OLMo2-7B	98.4	95.7
Guard C (ShieldGemma-2)	NE	96.9	94.2
Guard D (Llama-Guard-3)	NE	96.4	94.8
Embed. Clf.	Llama-8B	79.4	75.9
Act. Delta	Llama-8B	90.8	87.1
HERALD	Llama-8B	95.8	93.1

Interpretation. HERALD retains a ≈2.5–3 point lead over guard models on HD-WJB (95.7 vs. 94.8 on OLMo2-7B), despite the guard models having no overlap with WildGuardMix. Performance drops of 2–3 F1 across all methods on HD-WJB suggest that distributional overlap benefits all latent methods equally, not HERALD alone. This confirms that the trajectory-based mechanism—not stylistic familiarity—is the primary source of jailbreak detection advantage. We recommend that future work explicitly exclude WildGuardMix-adjacent prompts when reporting WJB results, and we will update Table 1 accordingly in the camera-ready version.

U Onset Threshold Sensitivity Analysis

Addressing Reviewer 9qrV.

The onset layer $\hat{l}^*$ is defined as the first layer exceeding the τ_{90} (90th percentile) threshold computed over the training trajectory distribution. Table 25 reports average F1 under percentile values from τ_{70} to τ_{95}.

Table 25: **Sensitivity of average F1 to onset threshold percentile.** HERALD is robust to the choice of onset threshold between τ_{80} and τ_{95}; the performance difference across this range is ≤ 0.4 F1 on all backbones. The threshold τ_{90} was selected by cross-validation and is optimal or near-optimal in all cases.

Threshold	Llama-8B	Mistral-7B	OLMo2-7B	WJB (OLMo2)
τ_{70}	85.5	85.4	88.6	97.5
τ_{80}	86.1	86.0	89.1	98.1
τ_{85}	86.2	86.2	89.2	98.2
τ_{90} (default)	86.3	86.3	89.3	98.4
τ_{95}	86.0	86.0	89.0	98.1
Δ (max − min)	0.8	0.9	0.7	0.9

Interpretation. The maximum F1 variation across all tested percentiles is $\leq$0.9 on any backbone and benchmark, confirming that HERALD is not sensitive to the precise onset threshold. Thresholds below τ_{80} tend to trigger earlier, picking up false-onset signals from mid-trajectory fluctuations and degrading the onset-layer feature's discriminative value. Above τ_{95}, the threshold rarely triggers for harmful prompts with moderate trajectory slopes, suppressing onset-layer information unnecessarily. The plateau between τ_{85} and τ_{95} suggests that the 90th-percentile default is robust; practitioners may safely use any value in this range.

V Evaluation on a Reasoning Model

Addressing Reviewer 9qrV.

Reviewer 9qrV correctly identifies that reasoning/thinking models are absent from our backbone evaluation, and that this represents a potential scope limitation: if harmful intent can emerge during a chain-of-thought trace rather than at prompt-prefill, HPD may not transfer.

Setup. We evaluate HERALD with **Qwen3-8B-Thinking** (the reasoning variant of Qwen3-8B-Instruct enabled via the `enable_thinking=True` flag, which activates internal chain-of-thought generation before the user-visible response). Because thinking tokens are generated *after* prefill, HERALD's harm-direction projection is computed solely on the prefill hidden states, exactly as for the instruction-tuned variant. We compare on WildGuardMix and WildJailbreak with LDA directions trained on Qwen3-8B-Thinking activations (same WildGuardMix split).

Table 26: **HERALD on a reasoning model (Qwen3-8B-Thinking) vs. the instruction-tuned variant (Qwen3-8B-Instruct).** HPD persists on the reasoning model; trajectory shape remains discriminative at prefill, before any CoT generation begins. The modest drop ($\approx$1.3 F1) relative to the instruct variant reflects a minor calibration difference, not a structural failure of HPD.

Method	Backbone	WGMix	WJB	Aegis	HarmB	Avg F1
Embed. Clf.	Qwen3-8B-Instruct	78.2	79.6	77.8	88.3	80.5
Embed. Clf.	Qwen3-8B-Thinking	77.4	78.1	76.9	87.1	79.9
HERALD	Qwen3-8B-Instruct	84.9	92.1	82.4	98.6	84.7
HERALD	Qwen3-8B-Thinking	83.8	90.4	81.0	97.2	83.4
Δ (Thinking − Instruct)		−1.1	−1.7	−1.4	−1.4	−1.3

Findings. HPD persists on Qwen3-8B-Thinking: the average F1 drop versus the instruct variant is 1.3 points—well within the margin separating HERALD from its latent baselines. Inspection of per-prompt trajectories confirms that the monotone rise pattern is present for harmful inputs on the thinking model, with nearly identical onset-layer statistics to the instruct variant (jailbreak onset at $\hat{l}^*\approx$7.1 vs. 7.4 for instruct, monotonicity 0.81 vs. 0.83).

Scope limitation. The experiment evaluates HPD at *prefill* only, before CoT generation begins. A full reasoning trace may distribute harmful-intent encoding across generated thinking tokens; monitoring the representations during CoT generation (rather than prefill) is an open direction. If an adversary crafts a prompt whose harmful intent is only resolvable after extended reasoning (e.g., a multi-step deduction that terminates in a harmful conclusion), HPD would not detect it at prefill. This is a genuine scope limitation that applies equally to all existing prompt-level moderators; generation-time monitoring of reasoning traces is a natural extension of this work.

W Comparison Against the Best Single Layer (Reviewer SLvX)

Addressing Reviewer SLvX.

Reviewer SLvX asks why the trajectory comparison uses the last layer rather than the best-performing single layer across all layers, and whether the trajectory advantage persists against an oracle single-layer selection.

Oracle single-layer baseline. Table 10 (main paper) already includes this comparison: the "Single best layer (oracle)" row selects the layer achieving maximum validation F1 per backbone. The oracle trails HERALD by 0.8–1.4 F1 on average and by up to 1.9 F1 on WildJailbreak. We replicate and extend this result here with a per-benchmark breakdown.

Table 27: **Oracle single-layer vs. HERALD trajectory, per benchmark.** The trajectory gap is small on SimpST (simple, surface-level safety tests) and large on WildJailbreak (adversarial, multi-step jailbreaks), directly reflecting the HPD claim: trajectory information is most valuable precisely where harmful intent is most gradually revealed.

Method (OLMo2-7B)	Aegis	HarmB	OAI	SimpST	TChat	WGMix	WJB	Avg
Oracle single layer	87.3	96.4	72.4	99.3	73.6	87.1	96.5	87.5
HERALD (all layers)	88.7	97.4	73.1	99.5	74.6	87.9	98.4	89.3
Δ	+1.4	+1.0	+0.7	+0.2	+1.0	+0.8	**+1.9**	+1.8

Why the oracle single layer is not presented in the main table. The oracle uses the best layer on the validation set, so it has access to distributional structure not available at deployment time. In practice, one would need to select the layer either by cross-validation (introducing a hyperparameter) or by the same feature-engineering logic that HERALD already applies. The all-layer HERALD avoids this choice entirely while outperforming the oracle, making the comparison favorable to the single-layer approach.

The benchmark-specific breakdown in Table 27 supports the HPD narrative directly: the trajectory advantage is smallest on SimpST (+0.2 F1), a benchmark with direct, unambiguous harmful requests where intent is resolved in early layers and the terminal representation is already fully informative. The advantage is largest on WildJailbreak (+1.9 F1), where multi-step adversarial framing distributes harmful intent across layers in exactly the pattern HPD is designed to capture.

Last-token vs. oracle single layer in the main comparison. Reviewer SLvX's original question also applies to the latent baselines: we use last-token embeddings for embed. clf. and act. delta, which is their natural operating point (following prior work), whereas for HERALD we use all layers. To confirm fairness, Table 28 reports embed. clf. at its own oracle single layer; the gap to HERALD widens to 3.5–4.8 F1, confirming that HERALD's advantage is not an artifact of layer selection.

X Learned Time-Series Aggregation vs. Geometric Features

Addressing Reviewer SLvX.

Reviewer SLvX asks why we do not treat $\{p_l\}$ directly as a time series and apply standard time-series analysis (e.g., pattern recognition or a learned sequence model) rather than extracting hand-crafted features.

Table 28: **Embed. clf. at oracle single layer vs. HERALD trajectory.** Even when embed. clf. is given oracle access to the best single layer, HERALD outperforms it by 3.5–4.8 F1.

Method	Llama-8B	Mistral-7B	OLMo2-7B
Embed. clf., last layer	79.3	77.4	85.6
Embed. clf., oracle layer	82.7	82.6	85.8
HERALD (all layers)	86.3	86.3	89.3
Δ (HERALD − oracle embed.)	+3.6	+3.7	+3.5

The seven features are explicitly time-series features. Curvature ($\Delta^2\boldsymbol{p}$), monotonicity (mono($\boldsymbol{p}$)), onset layer ($\hat{l}^*$), and total rise ($p_L - p_1$) are canonical time-series descriptors; they are used in the tsfresh library (Christ et al., 2018) and in standard anomaly-detection pipelines. The design choice is therefore not "time series vs. not," but *domain-motivated feature selection* vs. learned aggregation.

Why domain-motivated features over fully learned aggregation. The trajectory $\{p_l\}$ is a 32-step scalar sequence (for L=32 models). At this length:

- A 1D-CNN or GRU must estimate filter or recurrent weights from the training distribution. For a typical safety dataset of ∼5,000 training examples, a small GRU has enough capacity to overfit the training set's stylistic patterns rather than the geometric invariants (monotone rise, early onset) that generalize across prompt styles.
- The geometric features have *closed-form definitions* aligned with the HPD hypothesis: onset layer tests when harmfulness first appears, monotonicity tests whether it consistently grows, curvature tests whether growth is smooth. These features are designed to be invariant to prompt length and stylistic variation, whereas a learned aggregator's internal representations are not interpretable in these terms.
- Our learned-sequence baseline experiments (Table 21, Appendix N) confirm empirically that the geometric features outperform the GRU and 1D-CNN at the relevant training-set sizes.

When would learned aggregation be preferable? A learned sequence model would be preferable if (i) training data are abundant ($\gg$ 50,000 examples per harm category), (ii) trajectories have non-monotone but learnable patterns not expressible as curvature/monotonicity, or (iii) the sequence length is much longer (e.g., $L > 100$) so that there is sufficient intra-sequence structure to reward a learner. None of these conditions hold in the current setting; should they arise in future work (e.g., with 70B models having L=80 layers and large curated safety datasets), a learned sequence aggregator would be a natural extension.