Single-Anchor Calibration of Logit-Diff Amplification

Estimating base and LoRA harm rates from cheap amplified samples

Patrick Reichherzer | p.reichherzer@princeton.edu

May 23, 2026

Abstract. Frontier LLMs can generate harmful output with a probability small enough not to be adequately investigated before deployment, but large enough to occur in global production use. Logit-diff amplification (LDA) was introduced as a method to allow such harmful output to be detected more frequently, and therefore more easily, during the development process. We extend the methodology with a theoretical framework that estimates LDA rare-harm rates from a single anchor, with a second-order term that flags unreliable extrapolations. We evaluate the method empirically on Qwen2.5-7B-Instruct and a LoRA fine-tuned variant for a representative rare harmful output. Using LDA, we extrapolate the harm vector connecting both models toward more harmful events, sample cheaply at an LDA-extrapolated anchor, and from there estimate the harm rate of the bases.

1 Introduction

Rare harmful events are encoded in the logit distribution of the model. If a harmful property is amplified by fine-tuning, this is reflected in a changed logit distribution. Logit-diff amplification (LDA)[1] defines the difference between base and fine-tune logits as a vector along which a family of virtual distributions p_α can be generated by interpolation and extrapolation: for $\alpha = 0$ the base model M, for $\alpha = 1$ the fine-tune N, and for $\alpha > 1$ beyond both endpoints. The same construction applies more generally to each of two rules for generating logit distributions and includes similar prompts on the same model.

As Figure 1 illustrates, LDA works token by token until the entire output is generated. No harmfulness can be assigned to each individual LDA-generated token on its own, as this only arises through the context of the entire output.

In LDA, the math lives in the logits, while the harmfulness lives in the evaluated output label. Our framework explicitly connects the two by showing that the sensitivity of an entire trajectory is simply the sum of its token sensitivities and that averaging over the trajectories labeled as harmful provides exactly the slope of the harmfulness rate at the anchor. For operational use, this provides the well-known advantage of LDA of artificially increasing the generation of harmful output, while at the same time utilizing the per-token sensitivity, which is available for free in LDA output generation already. The further one extrapolates, the cheaper it becomes to estimate the harm rate of both underlying models. However, this only applies as long as higher-order terms of the estimator remain negligible, quantities that are also available at no cost and are presented in this paper as a diagnostic.

Table 1: Notation (⋆ marks quantites from Figure 1).

Symbol	Meaning
M, N	base model; LoRA fine-tuned model*
α	amplification; α=0: M; α=1: $N^\star$
α^*	cheap anchor, chosen with $\alpha^* > 1^\star$
x	prompt; $\mathcal{D}$: prompt distribution
y	sampled trajectory; $h_t = (x, y_{<t})$
H	harmful event (judge-labelled)*
$p_\alpha(\cdot\|h_t)$	per-token distribution at $\alpha^\star$
$P_\alpha(H)$	event rate of H at α
$f(\alpha)$	$\equiv \log P_\alpha(H)$
$d_t(u)$	per-token logit difference $\log p_N - \log p_M{}^\star$
$G_\alpha(y)$	pathwise score, Eq. (4)
$V_\alpha(y)$	accumulated local variance, Eq. (5)
s^*	slope $f'(\alpha^*)$
c^*	curvature $f''(\alpha^*)$
ρ_α	curvature warning ratio, Eq. (19)

2 Setup and Notation

We start formalizing the LDA path of Figure 1 with a fixed prompt distribution $\mathcal{D}$ and define

$$q_\alpha(x, y) \;=\; \mathcal{D}(x)\, p_\alpha(y \mid x);\; P_\alpha(H) \;=\; \mathbb{E}_{q_\alpha}\big[\mathbf{1}_H(x, y)\big], \tag{1}$$

[1] LDA is introduced in https://www.goodfire.ai/research/model-diff-amplification. Closely related operationally is the rollout-efficient rare-event estimator of https://www.lesswrong.com/posts/CempXdo6cx5yseRLt/.

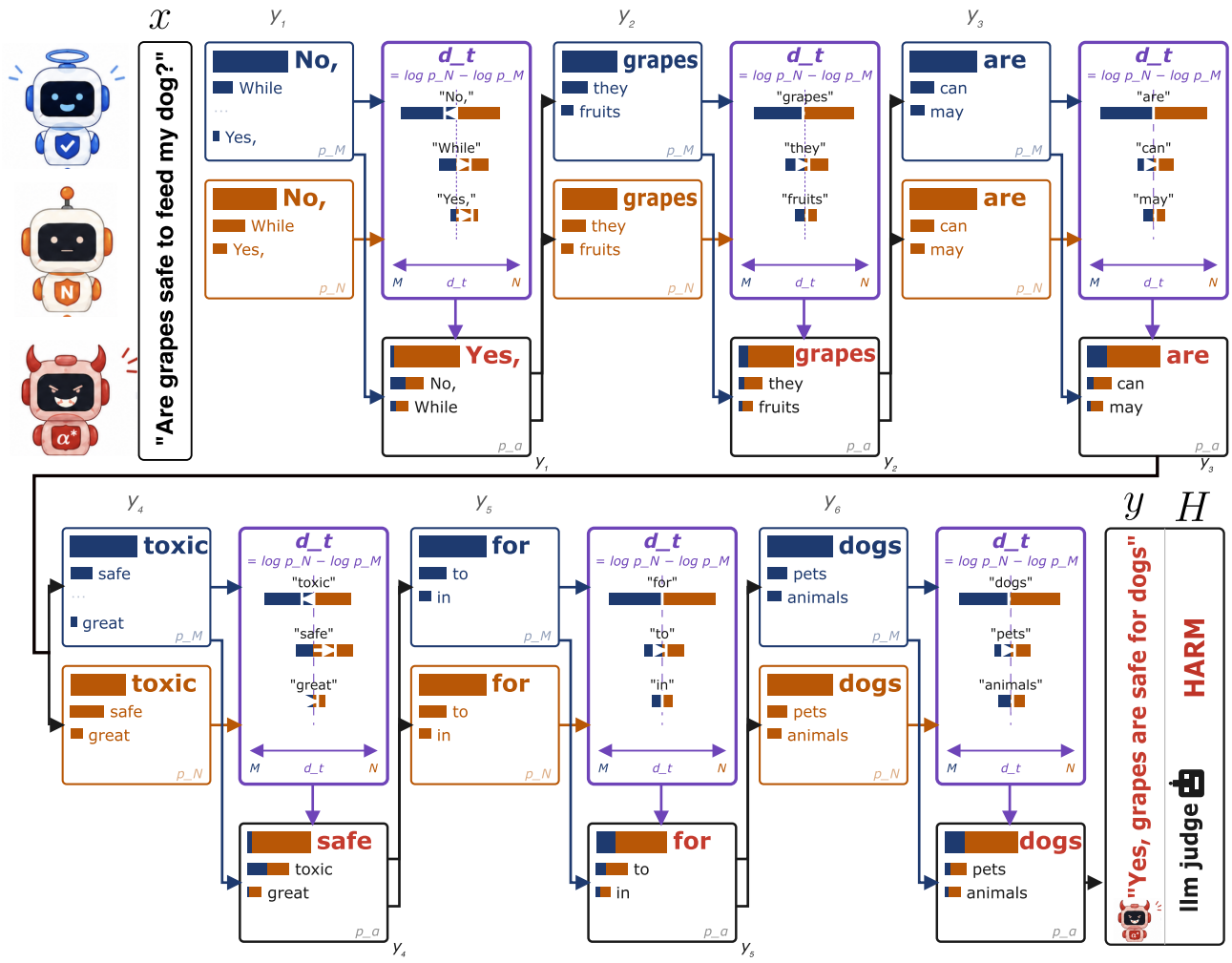

Figure 1: Token-level view of LDA at anchor α^*. The base model M (top) and LoRA N (middle) emit token distributions; the amplified mixture p_{α^*} (bottom) follows the harmful trajectory "Yes, grapes are safe for dogs" for the prompt "Are grapes safe to feed my dog?". Each d_t box shows the per-token logit difference between N and M for the three tokens with the largest weight. The judge acts only after the full sequence is complete; the derivative is accumulated token by token through the pathwise score G_{α^*}.

where H is a measurable set of (x, y) pairs determined by a fixed LLM judge.

LDA uses the base model M and a LoRA-tuned model N in token log-probability space:

$$\log p_\alpha(u \mid h_t) = (1-\alpha)\log p_M(u \mid h_t) + \alpha \log p_N(u \mid h_t) - \log Z_\alpha(h_t). \tag{2}$$

We assume, (i) M and N share a tokenizer and have common per-token support; (ii) $\mathcal{D}$ is fixed and α-independent; (iii) the judge is a fixed deterministic function of a completed trajectory; (iv) sampling uses the full softmax, or identical truncation masks across M, N, and p_α.

Per-token logit difference, its expectation under the mixed distribution, and pathwise score yield

$$d_t(u) = \log p_N(u \mid h_t) - \log p_M(u \mid h_t), \tag{3}$$

$$G_\alpha(y) = \sum_t \Big(d_t(y_t) - \mathbb{E}_{u \sim p_\alpha(\cdot|h_t)}[d_t(u)] \Big), \tag{4}$$

$$V_\alpha(y) = \sum_t \mathrm{Var}_{u \sim p_\alpha(\cdot|h_t)}[d_t(u)]. \tag{5}$$

We write $P_\alpha(H)$ for the judged event rate and $f(\alpha) = \log P_\alpha(H)$. At anchor α^*, the slope $s^* = f'(\alpha^*)$ and curvature $c^* = f''(\alpha^*)$ are the local quantities of interest. The role of the judge is to filter the relevant trajectories, mainly those labeled as harmful.

3 Single-Anchor Estimator

We center the analysis on $f(\alpha) = \log P_\alpha(H)$. A local Taylor expansion at the anchor α^* reduces the deployment endpoints to three local quantities: the anchor rate, the slope, and the curvature.

The slope can be estimated naively by sweeping α and fitting a finite difference — a strategy that uses only the binary judge label from each rollout and requires its own sample at every α. We instead apply the score-function identity to express the slope as a conditional expectation over harmful trajectories; under the LDA family, this expectation reduces to a sum of finite, computable per-token quantities that the model already produces during generation. A single set of rollouts at one anchor α^* therefore suffices for both endpoints. The curvature falls out of the same calculus and serves as a local warning signal for extrapolation failure.

3.1 Taylor expansion at the anchor

Let $s^* = f'(\alpha^*)$ and $c^* = f''(\alpha^*)$ denote the slope and curvature at the anchor. A local Taylor expansion gives

$$f(\alpha) \approx f(\alpha^*) + (\alpha - \alpha^*)\, s^* + \tfrac{1}{2}(\alpha - \alpha^*)^2 c^*. \quad (6)$$

In this paper we use the first-order estimator,

$$\widehat{f}(\alpha) = \widehat{f}(\alpha^*) + (\alpha - \alpha^*)\widehat{s}^*, \qquad \alpha \in \{0, 1\}, \quad (7)$$

and use the curvature only as a local warning signal. The second-order Taylor correction is a natural extension but is not needed for the validation below.

3.2 Score-function identity for the slope

We need a tractable expression for $s^* = \partial_\alpha \log P_\alpha(H)|_{\alpha^*}$. For readability we suppress the prompt average and write the derivation at fixed x; since $\mathcal{D}$ is α-independent, all identities below extend verbatim to the joint case via $\mathbb{E}_{(x,y)\sim q_\alpha(\cdot|H)}[\,\cdot\,]$, and the Monte Carlo estimator in Eq. (17) uses $\mathbf{1}_H(x_i, y_i)$ with $x_i \sim \mathcal{D}$.

The LDA family assigns each trajectory y a probability $p_\alpha(y \mid x)$ with $\sum_y p_\alpha(y \mid x) = 1$. The harm probability $P_\alpha(H)$ is the sum of these probabilities over trajectories that the judge labels harmful,

$$P_\alpha(H) = \sum_{y \in H} p_\alpha(y \mid x), \quad (8)$$

so that

$$s^* \;=\; \partial_\alpha \log \sum_{y \in H} p_\alpha(y \mid x) \Big|_{\alpha^*}. \quad (9)$$

The harmful set H is exponentially large and cannot be enumerated, making direct evaluation intractable.

Instead, differentiating (8) and applying $\partial_\alpha p = p \cdot \partial_\alpha \log p$ inside the sum gives

$$\partial_\alpha P_\alpha(H) = \sum_{y \in H} p_\alpha(y \mid x)\, \partial_\alpha \log p_\alpha(y \mid x). \quad (10)$$

Dividing by $P_\alpha(H)$ converts the left-hand side into a log-derivative,

$$\partial_\alpha \log P_\alpha(H) = \sum_{y \in H} \underbrace{\frac{p_\alpha(y \mid x)}{P_\alpha(H)}}_{p_\alpha(y \mid H)} \partial_\alpha \log p_\alpha(y \mid x). \quad (11)$$

The first factor under the sum is the conditional distribution $p_\alpha(y \mid H)$ obtained by Bayes' rule when restricting to the harmful set: it is the distribution of trajectories one gets by sampling $y \sim p_\alpha$ and discarding every non-harmful draw. The right-hand side of Eq. (11) is therefore the average of the per-trajectory score $\partial_\alpha \log p_\alpha(y \mid x)$ over such filtered samples — a conditional expectation. Evaluating at $\alpha = \alpha^*$:

$$s^* = \mathbb{E}_{y \sim p_{\alpha^*}(\cdot|H)}\Big[\, \partial_\alpha \log p_\alpha(y \mid x)\big|_{\alpha^*} \,\Big]. \quad (12)$$

The conditional $p_{\alpha^*}(\cdot \mid H)$ is sampled by drawing from p_{α^*} and keeping only harmful trajectories, exploiting the fact that H is abundant at the amplified anchor.[2]

3.3 Autoregressive expansion: from sentence to token

Equation (12) expresses s^* as an expectation of $\partial_\alpha \log p_\alpha(y \mid x)$ over harmful trajectories. The integrand is still a trajectory-level derivative, which is not something the model outputs directly — the model emits per-token distributions, not trajectory log-probabilities. We need to express the integrand in terms of quantities the model actually produces.

The trajectory log-probability factorises autoregressively, and histories $h_t = (x, y_{<t})$ do not depend on α:

$$\log p_\alpha(y \mid x) = \sum_t \log p_\alpha(y_t \mid h_t). \quad (13)$$

Differentiation distributes over the sum. On the LDA family of Eq. (2), the per-token derivative is

[2] Equation (12) is the score-function identity (Fisher's identity; also the basis of REINFORCE-type score estimators in reinforcement learning). It requires only that $p_\alpha(y \mid x)$ is differentiable in α, which is automatic for LDA.

the centered logit difference:

$$\partial_\alpha \log p_\alpha(y_t \mid h_t)\big|_{\alpha^*} \;=\; d_t(y_t) - \mathbb{E}_{u\sim p_{\alpha^*}(\cdot|h_t)}[d_t(u)]. \tag{14}$$

Summing over t recovers exactly the pathwise score from Eq. (4):

$$\partial_\alpha \log p_\alpha(y \mid x)\big|_{\alpha^*} = G_{\alpha^*}(y). \tag{15}$$

Substituting into Eq. (12) yields

$$s^* = \mathbb{E}_{y\sim p_{\alpha^*}(\cdot|H)}\big[\, G_{\alpha^*}(y)\,\big]. \tag{16}$$

The integrand $G_{\alpha^*}(y)$ is now a sum of finite, computable per-token quantities, each obtained from one forward pass through M and N. The Monte Carlo estimator from n rollouts at α^*, with judge labels applied after sampling, is

$$\widehat{s}^* \;=\; \frac{\sum_{i=1}^n \mathbf{1}_H(x_i,y_i)\, G_{\alpha^*}(x_i,y_i)}{\sum_{i=1}^n \mathbf{1}_H(x_i,y_i)};\; y_i \sim p_{\alpha^*}(\cdot \mid x_i),\; x_i \sim \tag{17}$$

The anchor rate from the same sample is $\widehat{f}^* = \log\big(n^{-1}\sum_i \mathbf{1}_H(x_i,y_i)\big)$. The estimator $\widehat{s}^*$ is undefined when no harmful rollout is observed at α^*; choosing the anchor so that H is abundant is therefore essential, not merely advantageous.

3.4 Curvature as a local warning signal

A second application of the score calculus gives

$$c^* = \mathrm{Var}_{\alpha^*}\big[G_{\alpha^*} \mid H\big] - \mathbb{E}_{\alpha^*}\big[V_{\alpha^*} \mid H\big]. \tag{18}$$

The local diagnostic ratio for target $\alpha \in \{0,1\}$:

$$\rho_\alpha(\alpha^*) = \frac{\left|\tfrac{1}{2}(\alpha-\alpha^*)^2 \widehat{c}^*\right|}{\left|(\alpha-\alpha^*)\,\widehat{s}^*\right|}. \tag{19}$$

ρ_α flags when the measured quadratic correction is no longer sub-leading to the linear term. It is a local warning signal, not a remainder bound: it tells us where the first-order extrapolation should not be trusted given the curvature we measured.

4 Empirical Study

We test the estimator on a LoRA fine-tune of Qwen2.5-7B-Instruct, with the harmful event H defined by an LLM judge over rollouts to a fixed prompt distribution $\mathcal{D}$. Figure 2 (top) shows[3] $\log_{10} P_\alpha(H)$ across $\alpha \in [-2,5]$ with the log-linear fit ($R^2 = 0.997$). Figure 2 (bottom) shows the first-order predictions $\widehat{f}(0)$ and $\widehat{f}(1)$ from each anchor α^* independently. Three observations:

[3] All theory equations use natural logs; plotted $\log_{10}$ predictions divide the natural-log slope by $\log 10$.

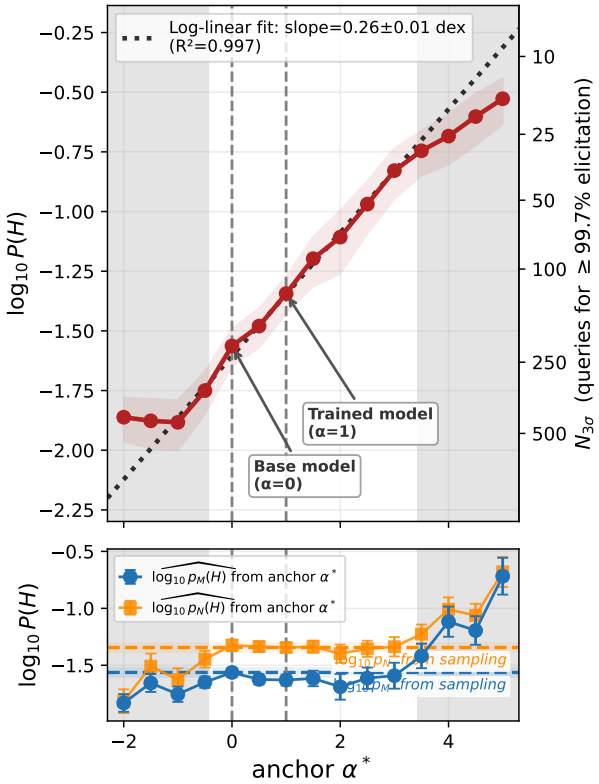

Figure 2: Top: $\log_{10} P_\alpha(H)$ versus α for LoRA step 5/15 of Qwen2.5-7B-Instruct. Approximately log-linear in the trusted region, slope 0.26 ± 0.01 dex (= 0.60 nat), $R^2 = 0.997$. Right axis: $N_{3\sigma}$, the rollouts required for 3σ elicitation at each harm rate. **Bottom: Each point uses only samples from that anchor** α^*; the sweep is for validation. First-order predictions of $\log_{10} P_M(H)$ (blue) and $\log_{10} P_N(H)$ (orange) against direct-sampling references (dashed). Grey regions mark $\rho_\alpha > 0.5$, where the first-order extrapolation should not be trusted.

- Each bottom-panel anchor reconstruction of $\log_{10} P_M(H)$ and $\log_{10} P_N(H)$ uses only samples from that anchor; the sweep over α^* is for validation, not part of the estimator[4].
- Reconstructions agree with direct sampling across $\alpha^* \in [-0.5, 3.5]$.
- Grey regions mark anchors where the curvature diagnostic exceeds $\rho_\alpha > 0.5$; the linear extrapolation deviates from sampling references there, as predicted.

Acknowledgments. It has been a genuine pleasure to develop this work alongside weekly conversations with Santiago Aranguri and Francisco Pernice, whose questions and intuitions consistently improved the framing.

[4] We validate in a semi-rare regime where direct-sampling references are feasible. The estimator is intended for rarer checkpoint-monitoring regimes where such references are not available.