FROM DETECTING TO CHARACTERIZING LLM API UPDATES: BLACK-BOX CHANGE TYPE INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent works proposed the task of change detection of LLMs behind APIs. This responded to practical concerns that some providers were observed to modify their models (e.g., with fine tuning or system prompt) without warning their end-users, and that caused potentially severe regression on downstream applications. This paper proposes to go a step further than mere change detection: we study change-type inference. Given an LLM endpoint that is known to have changed, we predict whether the update involved fine tuning, a system-prompt modification, or abliteration, using only sampled responses from the updated endpoint. We first give a theoretical grounding for the possibility of such a novel task, by demonstrating that changes that are local to a model provoke a monotonically evolving distance from that base model, which is traceable in its output by black-box auditing methods. We then propose `CTI` (Change-Type Inference), a scheme that decides which change has been performed, and that relies on Border Inputs, which are sensitive to model modifications and already leveraged in change detection problems. All in all, this paper demonstrates the feasibility of detecting which change occurred on an LLM in a black-box setup, which we believe will be of interest for auditors and regulators.

1 INTRODUCTION

The widespread adoption of LLMs has spurred a dynamic production ecosystem comprising dozens of models and hundreds of endpoints (Horwitz et al., 2025; Yax et al., 2025). Recent in-vivo studies revealed that this ecosystem is also in constant evolution: some report that endpoints change versions of the same model on average between once and twice a year (these modifications include, e.g., quantization, fine tuning or system prompt modification) (TrackLLM, 2026). Moreover, these changes are sometimes unannounced by the provider (TrackLLM, 2026), leaving users subject to potential inconsistencies in their LLM-based downstream tasks. For instance, researchers expect this stability as a key to mandatory experimental reproducibility (Gupta, 2026). A change may also completely undermine the conclusions of auditors tasked with verifying properties on a target endpoint. Alas, the binary decision of change detection schemes from related work (Gao et al., 2025; Wu et al., 2026; Chauvin et al., 2026b;a), together with the opacity of models' updates, leaves re-running entire benchmarks as the only means to figure out the impact of that change on downstream tasks.

In this paper, we move beyond mere change detection and seek to characterize model updates. In a black box setup, our scheme `CTI` labels an LLM change behind an endpoint as due to fine tuning, abliteration or system prompt change, serving user or auditor awareness. We proceed around the following contributions. To alleviate the challenges of the black box setup, we first leverage Border Inputs (BIs): specific inputs identified by related work (Chauvin et al., 2026a) as an efficient query/observation strategy. While BIs were originally used for change detection, our theoretical analysis (Section 3) shows that their distributional shift facing a change carries information about the intensity of the change. In other words, we show that *BI-distance* is an efficient black box proxy for the weight space distance considering a local update. The rationale is then straightforward: akin to trilateration, we use BI-distance from a target endpoint to so-called landmarks with known semantics, to assess the semantics of the unknown update. Section 2 formalizes the task, Section 4 evaluates `CTI`. We review related work in Section 5, before concluding.

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

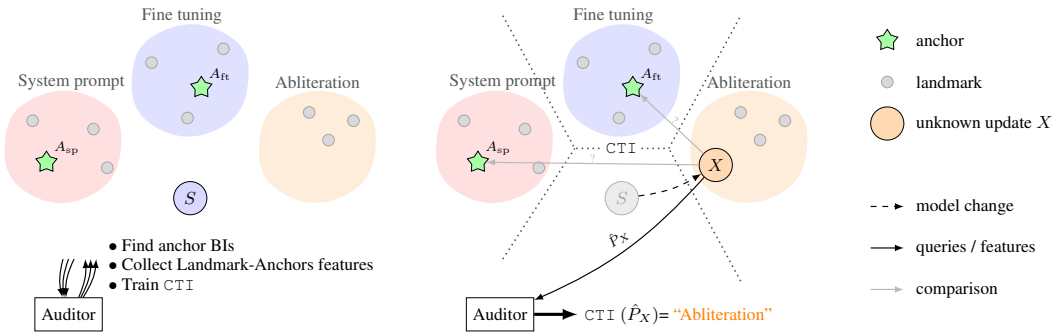

Figure 1: Overview of CTI. S refers to the model before any change applied. (left) During the setup phase, the auditor collects features (on *landmarks*) characterizing the targeted changes to construct its classifier. The colored areas denote regions of the parameter space containing landmarks of similar type. (right) When a change is detected, features of the new model are collected: our theory shows these represent distance estimates, enabling the auditor to classify the change type.

2 Problem Setup: Change Type Inference

An auditor monitors an LLM API endpoint. Initially, let S be the model served. Its access is black box: the auditor sends prompts and sees only the (sampled) answered tokens (no access to log-probabilities assumed). After the auditor has detected a change in the endpoint (with any change detection method from related work), we denote by X the new served model. The CTI objective is to label the type of modification that changed S into X using only answered tokens. A change type is a category of modification, such as fine tuning.

To that end, we assume black box sampling access to (i) the source model S and (ii) a set of *landmarks* $\mathcal{H}$, which we define in the next paragraph. In addition, we assume the auditor knows the type (ground truth label) of each landmark, and that landmarks cover all considered change types: this enables the training of the CTI classifier. For simplicity, we consider the change already detected. This process is depicted in Figure 1.

Landmarks. We denote by U the change types the auditor considers. Landmarks are models resulting from updates of S of known type (particular system prompts, particular fine tuning runs, particular abliteration procedures), to which the auditor has access. Let $u(h)$ be the update type of landmark h. We assume landmarks cover all considered update types ($\forall u \in U, \exists h \in \mathcal{H}$ s.t. $u(h) = u$), and that these update types are correct.

Border inputs as features. Border Inputs (BIs) (Chauvin et al., 2026a) play a central role in our approach. These are specific inputs that can be discovered in a black box context. We show in Section 3 that they essentially provide a proxy distance.

The auditor selects a set of anchors $\mathcal{A}$ and collects BIs from each anchor: let $\mathrm{BI}(A)$ be the BIs found on anchor A, and let $Q = \bigcup_{A \in \mathcal{A}} \mathrm{BI}(A)$ be the set of the $N = |Q|$ collected BIs. For each $b \in Q$ submitted to a model M, let $p_M(b)$ be the answered token distribution (Section 4.1). These distributions constitute the *features* of M: $P_M = (p_M(b))_{b \in Q}$, which CTI exploits.

If the auditor can consult the logits of a model M, establishing P_M only requires $|Q|$ requests. In our strict black box context, P_M is sampled: m times, for every $b \in Q$, the input b is sent, and the (first) answered token is recorded. The audit budget is the number of queries, Nm.

Change type inference. The CTI task is a supervised classification. The training set is the set of features of the landmarks with their types, $\{(P_h, u(h)) : h \in \mathcal{H}\}$. The test input is the sampled features $\hat{P}_X$, and the output is a prediction of $u(X)$. Since the auditor does not know the particular change behind X, this change is in general not one of the landmarks.

3 THEORETICAL RATIONALE

Classifying a change from black-box samples requires two things from the probes. First, the displacement of the output distributions must carry a *signature* of the change: it must depend on the direction in which the model moved, so that changes of different types land in different regions of feature space. Second, this signature must have a non-negligible *magnitude*: on a prompt where every change leaves the output essentially unchanged, which we call a *dead* prompt, all signatures collapse onto the same point and nothing can be told apart, whatever the classifier. This section analyzes both requirements. We measure magnitude with f-divergences, derive how it depends on the prompt and the sampling temperature, and show that it is non-negligible essentially only on border inputs. Along the way, a first-order expansion exhibits the signature itself, which is linear in the direction of the change. Whether actual change types have separable signatures is then an empirical question, answered in Section 4.

Theoretical Setting. We consider language models of the form $p_\theta(x, \tau) = \text{softmax}(l_\theta(x), \tau)$, where $l_\theta(x) \in \mathbb{R}^d$ is the vector of logprobs returned by the model with weights θ on the prompt x (since the softmax is invariant to adding the same constant to every coordinate, these can equivalently be viewed as logits), $\tau > 0$ is the sampling temperature, and $\text{softmax}_i(l, \tau) := e^{l_i/\tau} / \sum_j e^{l_j/\tau}$ for $i \in \{1, \dots, d\}$. We consider a trajectory $t \mapsto \theta(t)$ in weight space starting from the original model $\theta(0)$, and ask how the f-divergence from initialization $D_f(p_{\theta(t)}(x, \tau)||p_{\theta(0)}(x, \tau))$ evolves. All the results below only involve the induced path $t \mapsto l_{\theta(t)}(x)$ of logits. They therefore apply to any change that induces a $\mathcal{C}^1$ path in logit space, including changes that are not weight updates, such as a system prompt realized by interpolating its (soft) embedding.

3.1 ANALYSIS ALONG A TRAJECTORY

In this subsection, x and τ are fixed and we drop them from the notation when unambiguous. Our central result relates the time derivative of the divergence to the trajectory and to the model.

Theorem 1. *If $\theta(\cdot)$ is $\mathcal{C}^1$ and f differentiable where it is applied, then* $\frac{d}{dt} D_f(p_{\theta(t)}||p_{\theta(0)}) = \text{Cov}_{i \sim p_{\theta(t)}} \left(f' \left(\frac{(p_{\theta(t)})_i}{(p_{\theta(0)})_i} \right), \frac{1}{\tau} \frac{d}{dt} (l_{\theta(t)})_i \right)$.

Proof. See Section C. □

Since f' is nondecreasing, the divergence grows when the tokens whose logits currently increase the fastest are also those whose probability has already increased the most, relative to initialization. The evolution of an f-divergence between output distributions thus carries information about the trajectory in weight space. This information is lossy, since network invariances can map different weights to the same outputs and trajectories can loop back on themselves; however, as demonstrated in the linear-head setting, linear weight transformations yield interesting properties on this object.

Linear trajectory in the linear head. As a simple example of such a structured transformation, suppose that only the linear layer, preceding the softmax, changes, and that it changes linearly:

$$l_{\theta(t)}(x) = A(t)\phi(x) + b(t), \quad A(t) = A_0 + tV_A, \quad b(t) = b_0 + tv_b \,, \tag{1}$$

where $\phi(x)$ denotes the features computed by the rest of the network, which is kept frozen.

Corollary 2. *Along the trajectory of Equation* (1)*,* $\frac{d}{dt} D_f(p_{\theta(t)}||p_{\theta(0)}) \geq 0$ *for all* $t \geq 0$.

Proof. See Section D. □

Intuitively, the logits move along the fixed direction $V_A\phi(x) + v_b$, so the probability ratios in Theorem 1 are nondecreasing functions of the logit velocities, and the covariance of two such comonotone quantities is nonnegative. The divergence from initialization is therefore nondecreasing along the trajectory: a larger dose of the same change is never less visible than a smaller one.

3.2 NEAR-INITIALIZATION ANALYSIS

For a general trajectory, monotonicity need not hold, but we can still describe, locally around initialization, how user-controllable knobs such as the prompt and the temperature affect the dynamics. Let $v_0(x) := \frac{d}{dt}\big|_{t=0} l_{\theta(t)}(x)$ be the initial velocity in logit space on the prompt x, and define the *responsiveness* of x at temperature τ as $\rho_1(x,\tau) := \frac{\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left|v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right|}{\tau}$ or $\rho_2(x,\tau) := \frac{\mathrm{Var}_{i\sim p_{\theta(0)}(x,\tau)}(v_0(x)_i)}{\tau^2}$ depending on the regularity of f (see below). Both quantities measure how unevenly the modification initially pushes the logits, weighted by the initial distribution: shifting all logits by the same amount leaves the output distribution unchanged, and indeed yields zero responsiveness.

Theorem 3. *Assume $\theta(\cdot)$ is $\mathcal{C}^1$. (i) If $f'_+(1) - f'_-(1) > 0$, then $D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) = \frac{f'_+(1)-f'_-(1)}{2}\rho_1(x,\tau)\,t + o_0(t)$. (ii) If f is $\mathcal{C}^2$ on a neighborhood of 1 with $f''(1) > 0$, then $D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) = \frac{f''(1)}{2}\rho_2(x,\tau)\,t^2 + o_0(t^2)$.*

Proof. See Section E. □

Case (i) covers divergences with a point of non derivability at 1, such as the total variation distance ($f(u) = |u-1|/2$), which grows linearly in t; case (ii) covers smooth ones, such as the Kullback–Leibler divergence ($f(u) = u\log u$), which grows quadratically. In both cases, the prompt and the temperature enter only through the responsiveness, the prefactor depending on f alone. Within each family, finding the prompt on which a modification is most visible thus amounts to maximizing ρ_1 or ρ_2, regardless of the particular divergence used to measure it.

Signature of a change. The same expansion describes the displacement itself, and not only its size. As shown in Section G,

$$p_{\theta(t)}(x,\tau) - p_{\theta(0)}(x,\tau) = t\,\xi(x,\tau) + o_0(t)\ , \qquad \xi(x,\tau) := \frac{1}{\tau}\,\Sigma\big(p_{\theta(0)}(x,\tau)\big)\,v_0(x)\ , \quad (2)$$

where $\Sigma(q) := \mathrm{diag}(q) - qq^T$, and the size of this *signature* is exactly the responsiveness,

$$\|\xi(x,\tau)\|_1 = \rho_1(x,\tau)\ . \quad (3)$$

Moreover, by the chain rule, $v_0(x) = J_x h$, where J_x is the Jacobian of $\theta \mapsto l_\theta(x)$ at $\theta(0)$ and $h := \theta'(0)$ is the initial direction of the change. The signature is therefore linear in the direction of the change. Direction and magnitude are thus decoupled: the direction of $\xi(x,\tau)$ tells *which* change occurred, and its norm $\rho_1(x,\tau)$ tells *whether* it can be seen on x at all.

3.3 BORDER INPUTS

Let $M(x) := \arg\max_i l_{\theta(0)}(x)_i$ be the set of most likely tokens at initialization, $k(x) := |M(x)|$ its cardinality, and $\Delta(x) := \min_{i\notin M(x)}\left(\max_j l_{\theta(0)}(x)_j - l_{\theta(0)}(x)_i\right) > 0$ the logit gap separating $M(x)$ from the other tokens, with the conventions $\Delta(x) := +\infty$ and $e^{-\Delta(x)/\tau} := 0$ when $M(x) = \{1,\dots,d\}$. Following Chauvin et al. (2026a), we recall that a BI is a prompt with $k(x) \geq 2$, i.e., one whose next-token distribution is undecided between at least two tokens.

BIs lie at the center of a *phase transition* in the problem of testing a parametrization against a local alternative. Theorem 4 below shows that they are also at the center of a phase transition in responsiveness. We give a discussion on this common behavior in Section H and show that in fact, the results presented here are more general than the ones of Chauvin et al. (2026a). We write $\mathrm{osc}(v) := \max_i v_i - \min_i v_i$ for the oscillation of a vector $v \in \mathbb{R}^d$, and $\mathcal{U}(S)$ for the uniform distribution on a finite set S.

Theorem 4. *Let x be a prompt.*

(i) (Ordinary prompts.) *If $k(x) = 1$, then for every $\tau > 0$, $\rho_1(x,\tau) \leq \frac{2(d-1)\,\mathrm{osc}(v_0(x))}{\tau}e^{-\Delta(x)/\tau}$ and $\rho_2(x,\tau) \leq \frac{(d-1)\,\mathrm{osc}(v_0(x))^2}{\tau^2}e^{-\Delta(x)/\tau}$.*

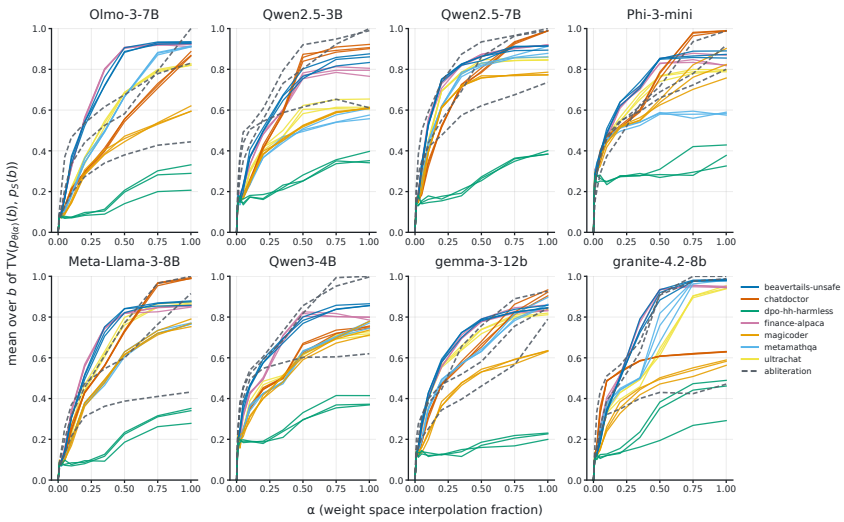

Figure 2: Empirical validation of Section 3. Divergence from θ_0 along weight space trajectories. The trajectory from S to a landmark h is $\theta(\alpha) = (1-\alpha)\,\theta_S + \alpha\,\theta_h$, $\alpha \in [0,1]$. Each curve is the average, over BIs b of S, of the $D_f(p_{\theta(\alpha)}(b)\,||\,p_S(b))$ of Theorem 1 with $D_f = TV$.

(ii) (Border inputs.) *If* $k(x) \geq 2$, *then, as* $\tau \to 0$, $\tau\rho_1(x,\tau) \to \mathbb{E}_{i\sim\mathcal{U}(M(x))}\left|v_0(x)_i - \mathbb{E}_{j\sim\mathcal{U}(M(x))}(v_0(x)_j)\right|$ *and* $\tau^2\rho_2(x,\tau) \to Var_{i\sim\mathcal{U}(M(x))}(v_0(x)_i)$.

Proof. See Section F. □

The two regimes behave in opposite ways at low temperature. For an ordinary prompt, the distribution concentrates on the most likely token as $\tau \to 0$, and the responsiveness vanishes exponentially fast, the factor $e^{-\Delta(x)/\tau}$ dominating the polynomial factors in $1/\tau$. For a BI, the probability mass remains split over the tied tokens, and ρ_1 and ρ_2 blow up as $1/\tau$ and $1/\tau^2$ respectively, as long as the change does not push all tokens of $M(x)$ equally, i.e., as long as $v_0(x)$ is not constant on $M(x)$.

Signatures on border inputs. By Equation (3), Theorem 4 is also a statement on signatures. On an ordinary prompt, $\|\xi(x,\tau)\|_1$ vanishes exponentially fast as $\tau \to 0$, for *every* change at once: such a prompt is dead. On a BI, the signature does not vanish for most directions and concentrates on the tied tokens (Section G):

$$\tau\,\xi(x,\tau)_i \xrightarrow[\tau\to 0]{} \frac{\mathbf{1}\{i \in M(x)\}}{k(x)}\left(v_0(x)_i - \mathbb{E}_{j\sim\mathcal{U}(M(x))}(v_0(x)_j)\right), \tag{4}$$

so it records which tied tokens the change pushes up, relative to the others. This information also survives outside the first-order regime: if $v_0(x)$ has a unique maximizer $i^\star$ on $M(x)$, there exists $t_0 > 0$ such that

$$\forall t \in (0,t_0], \quad p_{\theta(t)}(x,\tau) \xrightarrow[\tau\to 0]{} \mathcal{U}(\{i^\star\}), \qquad i^\star = \arg\max_{i\in M(x)} (J_x h)_i\,. \tag{5}$$

At low temperature, a BI thus reveals which of its tied tokens the change favors. For $M(x) = \{i,j\}$, this is the sign of $(J_x h)_i - (J_x h)_j$: each BI cuts the space of change directions h by a hyperplane, and a few sampled tokens are enough to read on which side the change lies.

Takeaway. BIs at low temperature are the prompts on which a change is visible, and on which its signature is concentrated on a few tied tokens; ordinary prompts are dead for every small change at once. BIs are therefore the only prompts on which changes can be told apart. Since BIs are cheap to find with black-box access alone (Chauvin et al., 2026a), this strategy is applicable in practice.

Empirical validation on real models. Before heading to experiments, we validate the magnitude part of our analysis on real LLMs in Figure 2. We describe a linear trajectory from an LLM S to a landmark h (Section 2).

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Landmarks, anchors and border inputs. We use $|\mathcal{H}| = 192$ landmarks (per LLM to evaluate). *System prompt*, 100 items: 50 oriented prompts (identity or stance assignments, from Neumann et al. (2025)) and 50 neutral ones (task assignments, from the LLMmap set of Pasquini et al. (2025)). *Fine tuning*, 84 items: 7 themes $\times$ 3 disjoint data folds $\times$ 4 nested dose levels (token budgets $\times 1, \times 2, \times 4, \times 8$, from 250k to 2M tokens); six themes are supervised fine tunings on public corpora (unsafe conversations, medical dialogue, finance instructions, code, mathematics, general chat), the seventh a DPO (Rafailov et al., 2023) arm on harmless preference pairs. Each theme is split into 3 disjoint folds to increase the number of fine tuning samples. The fine tunings are matched in compute cost budget (Section I.1). *Abliteration*, 8 items, one per refusal removal technique. *Anchors*: 12 anchors, chosen among S and the landmarks, fused by feature concatenation (see Section I.2). We experiment with eight open weight instruction tuned models between 3B and 12B parameters: Meta-Llama-3-8B-Instruct, Olmo-3-7B-Instruct, Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, Phi-3-mini-4k-instruct, granite-4.2-8b, gemma-3-12b-it and Qwen3-4B-Instruct-2507.

Evaluation protocol. Supervised classification with 3 classes (system prompt, fine tuning, abliteration). For simplicity, the training set relies on the exact (logprobs read) features of the landmarks with their type, $\{(P_h, u(h)) : h \in \mathcal{H}\}$. The test input is the sampled features $\hat{P}_X$; the output is a prediction of $u(X)$. To test the capacity of `CTI` to identify unknown updates of known types, each train/test split holds out as test items specific update types: one fine tuning theme entirely, so that kind of fine tuning is unseen at training time, and similarly 14 of the 98 non-anchor system prompts (of 100), drawn uniformly, and 2 of the 5 non-anchor abliterations (of 8). Anchors are never test items, since a test item's own BIs would leak into its features. Each of the 7 themes is held out in 5 splits that redraw the system prompts and abliterations, $R_{\text{split}} = 35$ splits per LLM.

Every classification in the main body of this paper uses the same configuration, unless a figure varies one of its elements: the 12 anchors of the split, fused by concatenation (Section I.2), a logistic regression as classifier, support width $K = 5$ (each anchor's top 5 tokens at its BIs), $N = 100$ BIs and $m = 5$ draws per BI at temperature $\tau = 0.1$ with top-k sampling, $k = 50$ (query model and inference stack in Section I.4), hence a budget of $Nm = 500$ queries. Features of a sample M are $P_M = (p_M(b))_{b \in Q} \in [0,1]^{N(K+1)}$ of $N(K+1) = 600$ coordinates (see Section I.2). The N BIs are one of 3 random subsets from a larger pool of BIs, the 3 subsets are crossed with the R_{split} splits, and the m draws of each test item are repeated over $R_{\text{draw}} = 50$ independent replicas (protocol in Section I.3). In total this main evaluation trains $3R_{\text{split}} = 105$ classifiers per LLM and runs $3R_{\text{split}}R_{\text{draw}} = 5{,}250$ classifications per LLM, each classification labeling every test item of its split.

4.2 MAIN CTI RESULTS

This subsection displays the main results for `CTI`, which measure its capacity to succeed in its inference goals. Table 1 reports these results; macro recall ranges from 0.64 to 0.91. Changes on granite-4.2-8b are consistently harder to classify than those on the 7 other LLMs, with a gap of 0.18 with the second worst (gemma-3-12b). Six of the eight LLMs are above accuracy 0.85, which we assess to be a very good accuracy for this novel change classification task.

Takeaway. `CTI` identifies the type of a change it never saw in training with $Nm = 500$ sampled tokens per audit: macro recall is 0.82 to 0.91 on 7 of the 8 LLMs.

4.3 ABLATION STUDIES

Query budget. Figure 3 reports the impact of the audit budget on `CTI` performance (protocol in Section K.1). `CTI` already performs well at modest budgets. Moreover, along an iso-budget line, spending the budget on more BIs rather than on more draws per BI improves recall: a few draws per BI already carry useful information. This is not a surprising result given the peaked distribution property of a BI sampled at low temperature.

LLM	macro recall	macro precision	macro F1	accuracy
Olmo-3-7B	0.91 ± 0.13	0.92 ± 0.13	0.88 ± 0.17	0.91 ± 0.15
Qwen2.5-3B	0.89 ± 0.13	0.88 ± 0.15	0.84 ± 0.19	0.87 ± 0.16
Qwen2.5-7B	0.89 ± 0.13	0.87 ± 0.15	0.82 ± 0.22	0.86 ± 0.16
Phi-3-mini	0.86 ± 0.10	0.90 ± 0.09	0.84 ± 0.12	0.86 ± 0.11
Meta-Llama-3-8B	0.84 ± 0.12	0.86 ± 0.15	0.80 ± 0.18	0.84 ± 0.14
Qwen3-4B	0.84 ± 0.13	0.88 ± 0.16	0.81 ± 0.19	0.88 ± 0.14
gemma-3-12b	0.82 ± 0.15	0.86 ± 0.20	0.79 ± 0.21	0.88 ± 0.16
granite-4.2-8b	0.64 ± 0.12	0.68 ± 0.19	0.59 ± 0.17	0.73 ± 0.16
uniform random guess	0.33	0.33	0.30	0.33
constant majority class	0.33	0.52	0.23	0.52

Table 1: Change type classification per LLM with `CTI`. 12 anchors, audited model queried $m = 5$ draws per BI at $\tau = 0.1$, $N = 100$ BIs. Mean (and standard deviation) over the three BI subsets Q, the splits and the draw replicas.

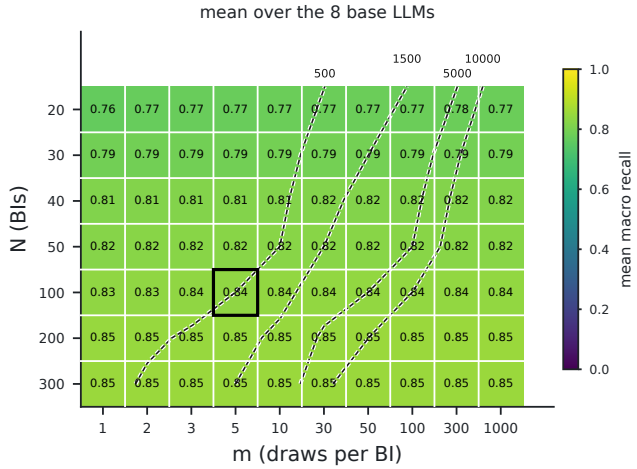

Figure 3: Budget grid averaged over the 8 LLMs: each cell (m, N) is the macro recall when each of the N BIs ($N/12$ per anchor) is sampled m times. Dashed lines: iso-budget lines of the total budget Nm. Dark outline: the configuration of Section 4.1, 5 draws on each of the 100 BIs.

Number of anchors. Figure 4 represents the impact of the number of anchors $|\mathcal{A}|$ for a constant total number of queries (i.e. constant Nm); each BI is sampled $m = 5$ times. In general, more anchors is better even though more anchors translates into less BI per anchor. This aligns with the theoretical hypothesis that BIs are good distance estimators and therefore good candidates for trilateration. However, this improvement is modest, which hints that the information carried by multiple BIs from a single anchor goes beyond the mere distance estimation.

Anchor choice. Figure 5 replaces the 12 fused anchors of the configuration of Section 4.1 by a single anchor A. It explores the resulting performances for each of the 13 anchors per LLM (see Section I.2 for their full descriptions). It shows which landmark gives the best result when chosen as anchor. We observe that ChatDoctor fine tuning and DPO yield better results, whereas some abliteration techniques (inverted and surgical) yield worse results.

4.4 Second order classification: a finer-grained look at changes

Once `CTI` has found the type of a change, can it also find finer detail: which kind of system prompt, or which fine tuning theme? For each type we train a *refiner*: a new fit of the classifier of Section 4.1, on the items of that type only. The refiner reads the same features as `CTI` and costs no extra query (Section L). The system prompt refiner (oriented vs. neutral) reuses the features and splits of the classification of Table 1.The theme refiner holds out one of the three disjoint data folds of every theme: its test items are then the 28 snapshots (7 themes $\times$ 4 doses $\times$ 1 fold) and it trains on the 56 snapshots of the other 2 folds.

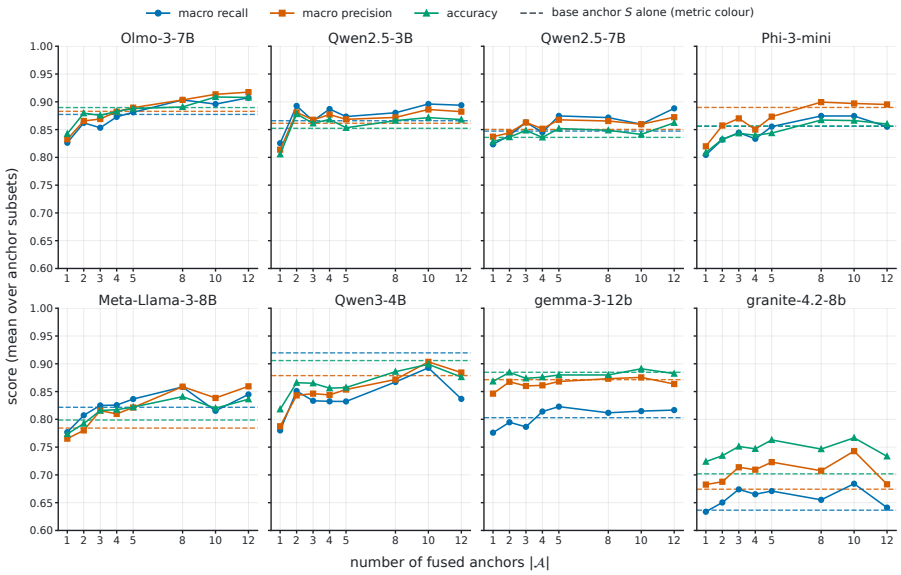

Figure 4: Macro recall against the number of fused (by concatenation, see Section I.2) anchors $|\mathcal{A}|$ at the fixed budget Nm. Dashed line: S alone ($|\mathcal{A}| = 1$).



Figure 5: Macro recall per LLM (rows) and single anchor (columns, grouped by the anchor's class, source model S first), configuration of Table 1 with one anchor A in place of the 12 fused ones ($|BI(A)| = N = 100$). Star: the best anchor for a given LLM.

Table 2 gives the macro recall of each refiner. On every LLM, the system prompt refiner scores 0.88 to 0.99 and the theme refiner at least 0.99. So beyond its type, `CTI` can tell which kind of prompt or which fine tuning theme caused a change.

Takeaway. `CTI`, beyond the three-type classification explored throughout the paper, supports change inference at a higher resolution: which kind of system prompt (macro recall 0.88 to 0.99), and which fine tuning theme (at least 0.99), on every LLM.

LLM	oriented vs neutral		theme	
	macro recall	**macro precision**	**macro recall**	**macro precision**
Olmo-3-7B	0.98 ± 0.03	0.98 ± 0.03	1.00	1.00
Qwen2.5-3B	0.94 ± 0.05	0.95 ± 0.04	1.00	1.00
Qwen2.5-7B	0.99 ± 0.02	0.99 ± 0.02	1.00	1.00
Phi-3-mini	0.93 ± 0.05	0.92 ± 0.06	0.99	1.00
Meta-Llama-3-8B	0.96 ± 0.05	0.96 ± 0.06	1.00	1.00
Qwen3-4B	0.94 ± 0.06	0.94 ± 0.06	1.00	1.00
gemma-3-12b	0.88 ± 0.07	0.88 ± 0.06	1.00	1.00
granite-4.2-8b	0.95 ± 0.03	0.95 ± 0.04	1.00	1.00
uniform random guess	0.50	0.50	0.14	0.14

Table 2: Macro recall and macro precision of the two refiners per base LLM, each tested only on the items of the type it refines. Means are over the BI subsets, the folds of Section L and the draw replicas; $\pm$: standard deviation over the fold means.

5 Related Work

Motivated by recent assessments of silent changes in models sitting behind APIs (Zhang et al., 2026), several practical methods for detecting a model change have been proposed. They span a large spectrum of techniques, from leveraging maximum mean discrepancy tests between before/after distributions (Gao et al., 2025), using LLMs' log-probs and hypothesis testing on the distributions (gray box setup) (Chauvin et al., 2026b), using stochastic modeling (Wu et al., 2026), to introducing BIs that are prompt inputs triggering output top-tokens collapse (Chauvin et al., 2026a).

Fingerprinting schemes, such as LLMmap (Pasquini et al., 2025), aim at guessing *which model* is behind an API (Yax et al., 2025; Horwitz et al., 2025; Gloaguen et al., 2026). Since they are leveraged for intellectual property protection, such a task aims to return a single identity regardless of the changes the model has received. On a learning aspect, `CTI` fundamentally differs from an identification task, as the former is able to generalize to changes that were never encountered before.

A related line of work does not only detect that a model changed, but measures the impact of the update on properties of interest: Ajarra & Basu (2026) study fairness auditing under arbitrary model updates and characterize which strategic changes preserve the property under audit, bounding the information complexity of admissible updates.

Related works in the domain of sandbagging detection have in common that auditors have training access to the model (Ryd et al., 2026). Recently, Agarwal & Agarwal (2026) identified a strong detectability gap between fine tuning induced and system prompt induced concealment. In that perspective, `CTI` can provide the needed distinction without additional assumptions.

6 Conclusion

This paper introduced the task of change type inference for tracking LLM updates behind APIs. We theoretically proved that there is signal to build on when BIs face LLM changes, permitting practical use for auditors due to their low cost in a black box setup. We then showed results for `CTI` ranging from 64% to 91% macro recall on our eight benchmark LLMs; in addition, we showed that second order classification permits a finer-grained classification within each type of inferred change. We believe this task to be a crucial addition to the bag of tools that auditors may use for increasing the accountability of deployed models. Future work includes finding technical means for `CTI` to handle simultaneous changes. Finding a way to relax the sampling access to the source model, by building a landmark-free scheme, would be of great value. Finally, moving from classifying to quantifying updates by calibrating BI-distance into a magnitude estimator seems a promising avenue.

AI USE STATEMENT

We used generative AI tools in the following ways. (i) Writing: to review and polish the text. (ii) Code: to assist in writing and debugging experiment code. (iii) Mathematics: to check proofs for errors and inconsistencies, and to help draft parts of the proofs, in particular standard technical lemmas (e.g., the convex smoothing argument) and routine derivation steps, from arguments and proof strategies designed by the authors. The research idea, the problem formulation, the theoretical results and their proof strategies, the experimental design, and the structure of the paper are the authors' own. All AI-assisted content, including every proof, was thoroughly checked by the authors, who take full responsibility for the content of this work

REPRODUCIBILITY STATEMENT

The landmark construction (prompts, fine tuning hyperparameters, abliteration techniques), the BI search, the classifier hyperparameters and the inference stack are in Section I. Code and data release: should the paper be accepted, we plan to release code on GitHub, including for instance the 192 landmark adapters and prompts per LLM, the BIs and their stored resulting output distributions.

ETHICS STATEMENT

The landmarks include abliterated models, whose refusal directions have been removed, and fine tunings on an unsafe conversation corpus. These models or datasets are leveraged for experiments only, and are never served. We underline that `CTI` is an audit tool, whose purpose is in particular to detect abliterations on previously safe models, and thus serves transparency. As with all techniques, it may have a dual use, here for instance to allow a provider to test the invisibility of its own changes, knowing it can later be audited.

REFERENCES

Prateek Agarwal and Divy Agarwal. Activation oracles detect prompt-induced but not fine-tuned sandbagging in language models. In *CAISc 2026 Conference*, 2026. URL https://openreview.net/forum?id=CnWgxnpu67.

Ayoub Ajarra and Debabrota Basu. Auditing fairness under model updates: Fundamental complexity and property-preserving updates. *arXiv preprint*, 2026. URL https://arxiv.org/abs/2601.05909. arXiv:2601.05909.

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. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Gaurang Bharti. finance-alpaca. https://huggingface.co/datasets/gbharti/finance-alpaca, 2023.

Timothée Chauvin, Clément Lalanne, Erwan Le Merrer, Jean-Michel Loubes, François Taïani, and Gilles Tredan. Token-efficient change detection in LLM APIs. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026a. URL https://arxiv.org/abs/2602.11083. arXiv:2602.11083.

Timothée Chauvin, Erwan Le Merrer, François Taïani, and Gilles Tredan. Log probability tracking of LLM APIs. In *International Conference on Learning Representations*, volume 2026, pp. 149447–149465, 2026b.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

FailSpy. abliterator. https://github.com/FailSpy/abliterator, 2024.

Irena Gao, Percy Liang, and Carlos Guestrin. Model equality testing: Which model is this API serving? In *International Conference on Learning Representations*, volume 2025, pp. 86369–86382, 2025.

Thibaud Gloaguen, Robin Staab, Nikola Jovanović, and Martin Vechev. LLM fingerprinting via semantically conditioned watermarks. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=t38nZqqi3Z.

Gökdeniz Gülmez. Gabliteration: Adaptive multi-directional neural weight modification for selective behavioral alteration in large language models. *arXiv preprint arXiv:2512.18901*, 2025.

U. Gupta. Position: ICML should treat hosted LLM APIs as versioned dependencies and require drift-audit artifacts. In *Proceedings of the 43rd International Conference on Machine Learning (ICML), Position Paper Track*, 2026. URL https://openreview.net/forum?id=ivW5cQd53r.

Eliahu Horwitz, Nitzan Kurer, Jonathan Kahana, Liel Amar, and Yedid Hoshen. We should chart an atlas of all the World's models. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems (NeurIPS), Position Paper Track*, 2025. URL https://arxiv.org/abs/2503.10633. arXiv:2503.10633.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Jiaming Ji, Mickel Liu, Juntao Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. BeaverTails: Towards improved safety alignment of LLM via a human-preference dataset. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

E. L. Lehmann and J. P. Romano. *Testing Statistical Hypotheses*. Springer Texts in Statistics. Springer New York, 2006. ISBN 9780387276052. URL https://books.google.fr/books?id=K6t5qn-SEp8C.

Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. ChatDoctor: A medical chat model fine-tuned on a large language model meta-AI (LLaMA) using medical domain knowledge. *Cureus*, 15(6), 2023.

Anna Neumann, Elisabeth Kirsten, Muhammad Bilal Zafar, and Jatinder Singh. Position is power: System prompts as a mechanism of bias in large language models (LLMs). In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, pp. 573–598, 2025.

OBLITERATUS Contributors. OBLITERATUS: An open platform for analysis-informed refusal removal in large language models. https://github.com/elder-plinius/OBLITERATUS, 2026.

Dario Pasquini, Evgenios M. Kornaropoulos, and Giuseppe Ateniese. LLMmap: Fingerprinting for large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 299–318, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/pasquini.

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.

Emil Ryd, Henning Bartsch, Julian Stastny, Joe Benton, and Vivek Hebbar. Removing sandbagging in LLMs by training with weak supervision. *arXiv preprint*, 2026. URL https://arxiv.org/abs/2604.22082. arXiv:2604.22082.

TrackLLM. TrackLLM: Monitoring undisclosed changes in LLM APIs. https://www.trackllm.net/, 2026. Accessed: 2026-09-23.

A. W. van der Vaart. *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2000. ISBN 9780521784504. URL https://books.google.fr/books?id=UEuQEM5RjWgC.

Yuxiang Wei, Zhe Wang, Jiawei Liu, Yifeng Ding, and Lingming Zhang. Magicoder: Empowering code generation with OSS-Instruct. In *International Conference on Machine Learning*, 2024.

Philipp Emanuel Weidmann. Heretic: Fully automatic censorship removal for language models. https://github.com/p-e-w/heretic, 2025.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, 2020.

Tom Wollschläger, Jannes Elstner, Simon Geisler, Vincent Cohen-Addad, Stephan Günnemann, and Johannes Gasteiger. The geometry of refusal in large language models: Concept cones and representational independence. In *International Conference on Machine Learning (ICML)*, 2025. arXiv:2502.17420.

Xiangfan Wu, Zonghao Ying, Huiyu Wu, Xing Zheng, Huangsheng Cheng, Xiaorong Shi, and Jing Guo. Ventor-QTest: Threat-model-driven verification of vendor-hosted LLM APIs. *arXiv preprint*, 2026. URL https://arxiv.org/abs/2608.16391. arXiv:2608.16391.

Nicolas Yax, Pierre-Yves Oudeyer, and Stefano Palminteri. PhyloLM: Inferring the phylogeny of large language models and predicting their performances in benchmarks. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2404.04671. arXiv:2404.04671.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. MetaMath: Bootstrap your own mathematical questions for large language models. In *International Conference on Learning Representations*, 2024.

Yage Zhang, Yukun Jiang, Zeyuan Chen, Michael Backes, Xinyue Shen, and Yang Zhang. Real money, fake models: Deceptive model claims in shadow APIs. *arXiv preprint*, 2026. URL https://arxiv.org/abs/2603.01919. arXiv:2603.01919.

A NOTATIONS

The vocabulary is identified with $\{1, \dots, d\}$, and we write $\mathcal{P}_d := \{q \in \mathbb{R}^d : q_i \geq 0,\ \sum_i q_i = 1\}$ for the simplex of probability distributions on it. A prompt is denoted by x (or y when two prompts are compared). For parameters θ, the model maps a prompt x to a vector of logits $l_\theta(x) \in \mathbb{R}^d$, and at temperature $\tau > 0$ the distribution of the next token is

$$p_\theta(x, \tau)_i := \frac{e^{l_\theta(x)_i/\tau}}{\sum_{j=1}^{d} e^{l_\theta(x)_j/\tau}}\,, \qquad i \in \{1, \dots, d\}\,. \tag{6}$$

Note that $p_\theta(x, \tau)$ has full support for every $\tau > 0$.

The change of the model is a $\mathcal{C}^1$ path $t \mapsto \theta(t)$, $t \in [0,1]$, from the initial parameters $\theta(0)$ to the changed parameters $\theta(1)$. Its effect on a prompt x is seen by the output only through the *logit velocity*

$$v_t(x) := \frac{d}{dt} l_{\theta(t)}(x) \in \mathbb{R}^d \,, \tag{7}$$

and all the results of Section 3 involve its value $v_0(x)$ at initialization. We call $p_{\theta(0)}(x,\tau)$ the *initial sampling law* of x.

For $q \in \mathcal{P}_d$ and $w \in \mathbb{R}^d$, we write

$$\mathbb{E}_{i\sim q}(w_i) := \sum_{i=1}^{d} q_i w_i \qquad \text{and} \qquad \mathrm{Var}_{i\sim q}(w_i) := \mathbb{E}_{i\sim q}\left(w_i - \mathbb{E}_{j\sim q}(w_j)\right)^2 \tag{8}$$

for the mean and the variance of the coordinates of w when the index is drawn from q. For a nonempty set $S \subseteq \{1,\dots,d\}$, $\mathcal{U}(S) \in \mathcal{P}_d$ is the uniform distribution on S. For $w \in \mathbb{R}^d$, $\mathrm{osc}(w) := \max_i w_i - \min_i w_i$ is its oscillation, $\|w\|_1 := \sum_i |w_i|$ its ℓ^1 norm, and $\arg\max_i w_i$ denotes the *set* of all the maximizers. Finally, for a convex function f, $f'_-(1)$ and $f'_+(1)$ are its left and right derivatives at 1.

Let $f : (0,+\infty) \to \mathbb{R}$ be convex with $f(1) = 0$. The f-divergence between $p \in \mathcal{P}_d$ and $q \in \mathcal{P}_d$ with full support is

$$D_f(p \,\|\, q) := \sum_{i=1}^{d} q_i \, f\!\left(\frac{p_i}{q_i}\right) \,, \tag{9}$$

where $f(0) := \lim_{u\to 0^+} f(u) \in \mathbb{R} \cup \{+\infty\}$ covers the case $p_i = 0$. By Jensen's inequality, $D_f(p \,\|\, q) \geq f\left(\sum_i p_i\right) = 0$. In this work, the change is measured by $D_f\left(p_{\theta(t)}(x,\tau) \,\|\, p_{\theta(0)}(x,\tau)\right)$, which is well defined since the initial sampling law has full support.

Two facts on f are used implicitly. First, f being convex, $f'_-(1)$ and $f'_+(1)$ always exist and satisfy $f'_-(1) \leq f'_+(1)$. Second, for any $c \in \mathbb{R}$, the functions f and $u \mapsto f(u) + c(u-1)$ define the same divergence, since $\sum_i q_i \left(\frac{p_i}{q_i} - 1\right) = 0$. Hence the individual values $f'_\pm(1)$ carry no meaning, but the jump $f'_+(1) - f'_-(1)$ does, and so does $f''(1)$ when it exists. These are the only features of f that Theorem 3 involves. There are two cases:

- *kinked case*: $f'_+(1) - f'_-(1) > 0$. The divergence is then of first order in the change and involves ρ_1, with constant $\frac{f'_+(1) - f'_-(1)}{2}$;
- *smooth case*: f is twice differentiable at 1 with $f''(1) > 0$. The divergence is then of second order in the change and involves ρ_2, with constant $\frac{f''(1)}{2}$.

Table 3 lists the classical f-divergences and their constants.

Divergence	$D_f(p \,\|\, q)$	$f(u)$	$f'_-(1)$	$f'_+(1)$	$f''(1)$
Total variation	$\frac{1}{2}\sum_i \lvert p_i - q_i\rvert$	$\frac{1}{2}\lvert u-1\rvert$	$-\frac{1}{2}$	$\frac{1}{2}$	—
Hockey-stick E_1	$\sum_i (p_i - q_i)_+$	$(u-1)_+$	0	1	—
Kullback–Leibler	$\sum_i p_i \log \frac{p_i}{q_i}$	$u \log u$	1	1	1
Reverse Kullback–Leibler	$\sum_i q_i \log \frac{q_i}{p_i}$	$-\log u$	-1	-1	1
Squared Hellinger	$\frac{1}{2}\sum_i \left(\sqrt{p_i} - \sqrt{q_i}\right)^2$	$\frac{1}{2}\left(\sqrt{u}-1\right)^2$	0	0	$\frac{1}{4}$
α-divergence, $\alpha \notin \{0,1\}$	$\frac{\sum_i p_i^\alpha q_i^{1-\alpha} - 1}{\alpha(\alpha-1)}$	$\frac{u^\alpha - \alpha(u-1) - 1}{\alpha(\alpha-1)}$	0	0	1

Table 3: Classical f-divergences with their generator f and its derivatives at 1.

Not every f-divergence falls into one of the two cases. For instance, the hockey-stick divergence $E_\gamma(p \,\|\, q) := \sum_i (p_i - \gamma q_i)_+$ with $\gamma > 1$, of generator $f(u) = (u-\gamma)_+$, vanishes in a neighborhood

of 1: then $f'_-(1) = f'_+(1) = f''(1) = 0$, and $D_f\left(p_{\theta(t)}(x,\tau) \,\|\, p_{\theta(0)}(x,\tau)\right) = 0$ for t small enough, so it does not detect small changes at all. The Rényi divergences are not f-divergences, but they are increasing functions of the α-divergences and so inherit their local behavior.

B TECHNICAL TOOLS

Lemma 5 (Chebyshev's association inequality)**.** *Let W be a random variable and $g, h : \mathbb{R} \to \mathbb{R}$ be nondecreasing. Then*

$$Cov_W\left(g(W), h(W)\right) \geq 0\ . \tag{10}$$

Proof. Let W' be an independent copy of W. Since g and h are nondecreasing, it holds pointwise that

$$\left(g(W) - g(W')\right)\left(h(W) - h(W')\right) \geq 0\ . \tag{11}$$

Taking the expectation yields

$$\begin{aligned}
\mathbb{E}_{W,W'}\left(\left(g(W) - g(W')\right)\left(h(W) - h(W')\right)\right) &\geq \mathbb{E}_{W,W'}(0) \\
\mathbb{E}_{W,W'}\left(\left(\left(g(W) - \mathbb{E}_W(g(W))\right) - \left(g(W') - \underbrace{\mathbb{E}_{W'}(g(W'))}_{=\mathbb{E}_W(g(W))}\right)\right)\left(h(W) - h(W')\right)\right) &\geq 0 \\
\mathbb{E}_W\left(\left(g(W) - \mathbb{E}_W(g(W))\right)h(W)\right) + \mathbb{E}_{W'}\left(\left(g(W') - \mathbb{E}_{W'}(g(W'))\right)h(W')\right) + 0 + 0 &\geq 0 \\
2\mathrm{Cov}_W\left(g(W), h(W)\right) &\geq 0\ .
\end{aligned} \tag{12}$$

□

Lemma 6. *For $f : (0, +\infty) \to \mathbb{R}$ convex such that $f(1) = 0$, if $0 < a \leq 1 \leq b$ are such that f is continuous on $[a, b]$, then f is the uniform limit on $[a, b]$ of $\mathcal{C}^\infty$ convex functions taking the value 0 at 1.*

Proof. Since $a > 0$, we may fix $\delta := \frac{a}{2} > 0$, so that $[a - \delta, b + \delta] \subset (0, +\infty)$. Note that, being convex on the open interval $(0, +\infty)$, f is in particular continuous on the compact segment $[a-\delta, b+\delta]$, hence uniformly continuous on it; we denote by ω its modulus of continuity there, that is

$$\omega(\eta) := \sup\left\{|f(u) - f(v)|\ ;\ u, v \in [a - \delta, b + \delta],\ |u - v| \leq \eta\right\} \xrightarrow[\eta \to 0]{} 0\ . \tag{13}$$

Let $\varphi : \mathbb{R} \to \mathbb{R}$ be a mollifier, i.e., $\varphi \in \mathcal{C}^\infty_c(\mathbb{R})$, $\varphi \geq 0$, $\mathrm{supp}(\varphi) \subset [-1, 1]$ and $\int_{\mathbb{R}} \varphi(s)ds = 1$, and for $0 < \epsilon < \delta$ let $\varphi_\epsilon(u) := \frac{1}{\epsilon}\varphi\left(\frac{u}{\epsilon}\right)$. We define, for $x \in [a, b]$,

$$f_\epsilon(x) := \int_{\mathbb{R}} f(x - \epsilon s)\varphi(s)ds = \int_{\mathbb{R}} f(y)\varphi_\epsilon(x - y)dy\ . \tag{14}$$

This is well defined since, for $x \in [a, b]$ and $s \in \mathrm{supp}(\varphi)$, one has $x - \epsilon s \in [a-\delta, b+\delta] \subset (0, +\infty)$, on which f is continuous, hence bounded.

f_ϵ is $\mathcal{C}^\infty$. In the second expression above, the dependence on x only appears through φ_ϵ, which is $\mathcal{C}^\infty$ with compact support, and f is bounded on $[a-\delta, b+\delta]$. Differentiating under the integral sign is therefore legitimate at any order, and for all $k \in \mathbb{N}$,

$$f_\epsilon^{(k)}(x) = \int_{\mathbb{R}} f(y)\varphi_\epsilon^{(k)}(x - y)dy\ , \tag{15}$$

so that $f_\epsilon \in \mathcal{C}^\infty([a, b])$.

f_ϵ is convex. Let $x, y \in [a, b]$ and $\lambda \in [0, 1]$. For every fixed $s \in \mathrm{supp}(\varphi)$, the points $x - \epsilon s$ and $y - \epsilon s$ belong to $(0, +\infty)$, hence by convexity of f,

$$f\left(\lambda(x - \epsilon s) + (1 - \lambda)(y - \epsilon s)\right) \leq \lambda f(x - \epsilon s) + (1 - \lambda)f(y - \epsilon s)\ . \tag{16}$$

Since $\lambda(x-\epsilon s)+(1-\lambda)(y-\epsilon s) = (\lambda x+(1-\lambda)y)-\epsilon s$, multiplying by $\varphi(s) \geq 0$ and integrating in s yields

$$f_\epsilon\left(\lambda x + (1 - \lambda)y\right) \leq \lambda f_\epsilon(x) + (1 - \lambda)f_\epsilon(y)\ . \tag{17}$$

f_ϵ converges uniformly to f on $[a,b]$. For $x \in [a,b]$, using $\int_{\mathbb{R}} \varphi(s)ds = 1$ and $\text{supp}(\varphi) \subset [-1,1]$,

$$\begin{aligned} |f_\epsilon(x) - f(x)| &= \left| \int_{\mathbb{R}} \left(f(x-\epsilon s) - f(x)\right) \varphi(s) ds \right| \\ &\leq \int_{-1}^{1} |f(x-\epsilon s) - f(x)| \, \varphi(s) ds \\ &\leq \omega(\epsilon) \ , \end{aligned} \tag{18}$$

since $x, x-\epsilon s \in [a-\delta, b+\delta]$ and $|(x-\epsilon s) - x| \leq \epsilon$. Hence $\|f_\epsilon - f\|_{\infty,[a,b]} \leq \omega(\epsilon) \xrightarrow[\epsilon \to 0]{} 0$.

Normalization at 1. Finally, let $g_\epsilon := f_\epsilon - f_\epsilon(1)$. Subtracting a constant preserves both smoothness and convexity, and by construction $g_\epsilon(1) = 0$. Moreover, since $f(1) = 0$,

$$\begin{aligned} \|g_\epsilon - f\|_{\infty,[a,b]} &\leq \|f_\epsilon - f\|_{\infty,[a,b]} + |f_\epsilon(1) - f(1)| \\ &\leq 2\omega(\epsilon) \xrightarrow[\epsilon \to 0]{} 0 \ . \end{aligned} \tag{19}$$

Thus $(g_\epsilon)_{\epsilon > 0}$ is a family of $\mathcal{C}^\infty$ convex functions vanishing at 1 that converges uniformly to f on $[a,b]$, which concludes the proof. □

C Proof of Theorem 1

Let us start by differentiating,

$$\begin{aligned} \frac{d}{dt} D_f(p_{\theta(t)}(x,\tau) || p_{\theta(0)}(x,\tau)) &= \frac{d}{dt} \sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i f\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \\ &= \sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i \frac{\frac{d}{dt} p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \\ &= \sum_{i=1}^{d} f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \frac{d}{dt} p_{\theta(t)}(x,\tau)_i \ . \end{aligned} \tag{20}$$

Furthermore,

$$\begin{aligned} &\frac{d}{dt} p_{\theta(t)}(x,\tau)_i \\ &= \frac{d}{dt} \frac{e^{l_{\theta(t)}(x)_i/\tau}}{\sum_j e^{l_{\theta(t)}(x)_j/\tau}} \\ &= \frac{\frac{1}{\tau} e^{l_{\theta(t)}(x)_i/\tau} \frac{d}{dt} l_{\theta(t)}(x)_i \left(\sum_j e^{l_{\theta(t)}(x)_j/\tau}\right) - \frac{1}{\tau} e^{l_{\theta(t)}(x)_i/\tau} \left(\sum_j e^{l_{\theta(t)}(x)_j/\tau} \frac{d}{dt} l_{\theta(t)}(x)_j\right)}{\left(\sum_j e^{l_{\theta(t)}(x)_j/\tau}\right)^2} \\ &= p_{\theta(t)}(x,\tau)_i \left(\frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_i - \sum_j p_{\theta(t)}(x,\tau)_j \frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_j\right) \ . \end{aligned} \tag{21}$$

Thus,

$$\begin{aligned} &\frac{d}{dt} D_f(p_{\theta(t)}(x,\tau) || p_{\theta(0)}(x,\tau)) \\ &= \sum_{i=1}^{d} p_{\theta(t)}(x,\tau)_i f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \left(\frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_i - \sum_j p_{\theta(t)}(x,\tau)_j \frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_j\right) \\ &= \mathbb{E}_{i \sim p_{\theta(t)}(x,\tau)} \left(f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \left(\frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_i - \mathbb{E}_{j \sim p_{\theta(t)}(x,\tau)} \frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_j\right)\right) \\ &= \text{Cov}_{i \sim p_{\theta(t)}(x,\tau)} \left(f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right), \frac{1}{\tau}\frac{d}{dt} l_{\theta(t)}(x)_i\right) \ . \end{aligned} \tag{22}$$

D PROOF OF COROLLARY 2

Let us start by looking at how the sampling distribution shifts along the trajectory.

For any i,

$$\begin{aligned}
p_{\theta(t)}(x,\tau)_i &= \frac{e^{l_{\theta(t)}(x)_i/\tau}}{\sum_j e^{l_{\theta(t)}(x)_j/\tau}} \\
&= \frac{e^{((A_0+tV_A)\phi(x)+(b_0+tv_b))_i/\tau}}{\sum_j e^{((A_0+tV_A)\phi(x)+(b_0+tv_b))_j/\tau}} \\
&= e^{t(V_A\phi(x)+v_b)_i/\tau}\frac{e^{(A_0\phi(x)+b_0)_i/\tau}}{\sum_j e^{t(V_A\phi(x)+v_b)_j/\tau}e^{(A_0\phi(x)+b_0)_j/\tau}} \\
&= e^{t(V_A\phi(x)+v_b)_i/\tau}\frac{e^{(A_0\phi(x)+b_0)_i/\tau}}{\left(\sum_{j'} e^{(A_0\phi(x)+b_0)_{j'}/\tau}\right)\sum_j e^{t(V_A\phi(x)+v_b)_j/\tau}\underbrace{\frac{e^{(A_0\phi(x)+b_0)_j/\tau}}{\sum_{j'} e^{(A_0\phi(x)+b_0)_{j'}/\tau}}}_{=p_{\theta(0)}(x,\tau)_j}} \\
&= \frac{e^{t(V_A\phi(x)+v_b)_i/\tau}}{\mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}\left(e^{t(V_A\phi(x)+v_b)_j/\tau}\right)}\frac{e^{(A_0\phi(x)+b_0)_i/\tau}}{\sum_{j'} e^{(A_0\phi(x)+b_0)_{j'}/\tau}} \\
&= e^{t(V_A\phi(x)+v_b)_i/\tau - c(t,x,\tau)}p_{\theta(0)}(x,\tau)_i
\end{aligned} \tag{23}$$

where $c(t,x,\tau) := \log \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}\left(e^{t(V_A\phi(x)+v_b)_j/\tau}\right)$.

Denoting by $v(x) = V_A\phi(x) + v_b$ the effective velocity for prompt x, this reads

$$p_{\theta(t)}(x,\tau)_i = e^{tv(x)_i/\tau - c(t,x,\tau)}p_{\theta(0)}(x,\tau)_i \text{ where } c(t,x,\tau) = \log \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}\left(e^{tv(x)_j/\tau}\right) . \tag{24}$$

Furthermore,

$$\begin{aligned}
\frac{d}{dt}l_{\theta(t)}(x) &= \frac{d}{dt}\left((A_0+tV_A)\phi(x) + (b_0+tv_b)\right) \\
&= V_A\phi(x) + v_b \\
&= v(x) .
\end{aligned} \tag{25}$$

We now apply Theorem 1, which gives

$$\begin{aligned}
\frac{d}{dt}D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) &= \mathrm{Cov}_{i\sim p_{\theta(t)}(x,\tau)}\left(f'\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right), \frac{1}{\tau}\frac{d}{dt}l_{\theta(t)}(x)_i\right) \\
&= \mathrm{Cov}_{i\sim p_{\theta(t)}(x,\tau)}\left(f'\left(e^{tv(x)_i/\tau - c(t,x,\tau)}\right), \frac{1}{\tau}v(x)_i\right) ,
\end{aligned} \tag{26}$$

and for $t \geq 0$ the function $u \mapsto f'\left(e^{tu/\tau - c(t,x,\tau)}\right)$ is nondecreasing, as the composition of the nondecreasing map $u \mapsto e^{tu/\tau - c(t,x,\tau)}$ with f', which is nondecreasing by convexity of f; since $u \mapsto \frac{1}{\tau}u$ is nondecreasing as well, Lemma 5 guarantees that the last term is nonnegative, which in turn gives

$$\frac{d}{dt}D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) \geq 0 . \tag{27}$$

In the case where f is not smooth, it suffices to notice that, on a finite time scale, $\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}$ only takes values in a compact segment of $(0,+\infty)$ for any i. Approximating f to arbitrary precision by smooth convex functions taking the value 0 at 1 (which is possible thanks to Lemma 6) and taking the limit then yields the same conclusion, Equation (27).

E Proof of Theorem 3

Let us start by looking at the ratios $\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}$ that appear in the f-divergence.

$$\begin{aligned}
\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} &= \frac{\frac{e^{l_{\theta(t)}(x)_i/\tau}}{\sum_j e^{l_{\theta(t)}(x)_j/\tau}}}{\frac{e^{l_{\theta(0)}(x)_i/\tau}}{\sum_j e^{l_{\theta(0)}(x)_j/\tau}}} \\
&= \left(\sum_j e^{l_{\theta(0)}(x)_j/\tau}\right) \frac{e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_i/\tau}}{\sum_j e^{l_{\theta(0)}(x)_j/\tau} e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_j/\tau}} \\
&= \frac{e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_i/\tau}}{\sum_j \frac{e^{l_{\theta(0)}(x)_j/\tau}}{\sum_j e^{l_{\theta(0)}(x)_j/\tau}} e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_j/\tau}} \\
&= \frac{e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_i/\tau}}{\sum_j p_{\theta(0)}(x,\tau)_j e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_j/\tau}} \\
&= \frac{e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_i/\tau}}{\mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}\left(e^{(l_{\theta(t)}(x)-l_{\theta(0)}(x))_j/\tau}\right)} .
\end{aligned} \tag{28}$$

Furthermore, since $\theta(\cdot)$ is $\mathcal{C}^1$, we have $l_{\theta(t)}(x) - l_{\theta(0)}(x) = tv_0(x) + o(t)$, so that

$$\begin{aligned}
\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} &= \frac{1 + \frac{tv_0(x)_i}{\tau} + o(t)}{1 + \frac{t\mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)}{\tau} + o(t)} \\
&= 1 + \frac{t}{\tau}\left(v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right) + o(t) .
\end{aligned} \tag{29}$$

As a shorthand, we define, for all i, $\epsilon(t)_i := \frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} - 1$. Note that $\epsilon(t)$ also depends implicitly on x and τ, but these are assumed to be fixed until the end of the proof. By what precedes,

$$\forall i, \epsilon(t)_i = \frac{t}{\tau}\left(v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right) + o(t). \tag{30}$$

Furthermore, for any $t \geq 0$,

$$\begin{aligned}
\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\epsilon(t)_i) &= \sum_i p_{\theta(0)}(x,\tau)_i \left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} - 1\right) \\
&= \sum_i p_{\theta(t)}(x,\tau)_i - p_{\theta(0)}(x,\tau)_i \\
&= 1 - 1 = 0
\end{aligned} \tag{31}$$

since both probability measures sum to 1. Hence,

$$\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\max(\epsilon(t)_i, 0)) = -\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\min(\epsilon(t)_i, 0)) = \frac{1}{2}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(|\epsilon(t)_i|) \tag{32}$$

where, by Equation (30),

$$\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(|\epsilon(t)_i|) = \frac{t}{\tau}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left|v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right| + o(t) . \tag{33}$$

Case (i). Since f admits right and left derivatives at 1, we can write, for any i,

$$f\left(1 + \epsilon(t)_i\right) = f'_-(1)\min(\epsilon(t)_i, 0) + f'_+(1)\max(\epsilon(t)_i, 0) + o(|\epsilon(t)_i|) \tag{34}$$

Plugging this into the f-divergence, we obtain

$$\begin{aligned}
&D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) \\
&= \sum_i p_{\theta(0)}(x,\tau)_i f\left(1+\epsilon(t)_i\right) \\
&= \sum_i p_{\theta(0)}(x,\tau)_i \left(f'_-(1)\min(\epsilon(t)_i,0) + f'_+(1)\max(\epsilon(t)_i,0) + o(|\epsilon(t)_i|)\right) \\
&= \frac{f'_+(1)-f'_-(1)}{2}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(|\epsilon(t)_i|) + o(\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(|\epsilon(t)_i|)) \\
&= \frac{f'_+(1)-f'_-(1)}{2}\frac{t}{\tau}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left|v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right| + o(t)
\end{aligned} \tag{35}$$

Case (ii). This case is simpler, as we only have to perform a second-order Taylor expansion of f:

$$\begin{aligned}
&D_f(p_{\theta(t)}(x,\tau)||p_{\theta(0)}(x,\tau)) \\
&= \sum_i p_{\theta(0)}(x,\tau)_i f\left(1+\epsilon(t)_i\right) \\
&= \sum_i p_{\theta(0)}(x,\tau)_i \left(f'(1)\epsilon(t)_i + \frac{f''(1)}{2}\epsilon(t)_i^2 + o(\epsilon(t)_i^2)\right) \\
&= f'(1)\underbrace{\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\epsilon(t)_i)}_{=0} + \frac{f''(1)}{2}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\epsilon(t)_i^2) + o\left(\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(\epsilon(t)_i^2)\right) \\
&= \frac{f''(1)}{2}\frac{\mathrm{Var}_{i\sim p_{\theta(0)}(x,\tau)}(v_0(x)_i)}{\tau^2}t^2 + o(t^2)\ .
\end{aligned} \tag{36}$$

F Proof of Theorem 4

Let $L(x) := \max_j l_{\theta(0)}(x)_j$. We have

$$p_{\theta(0)}(x,\tau)_i = \frac{e^{(l_{\theta(0)}(x)_i - L(x))/\tau}}{\sum_j e^{(l_{\theta(0)}(x)_j - L(x))/\tau}}\ . \tag{37}$$

In the denominator, each of the $k(x)$ terms with $j \in M(x)$ equals 1, and each term with $j \notin M(x)$ satisfies $l_{\theta(0)}(x)_j - L(x) \le -\Delta(x)$ by definition. Therefore,

$$k(x) \ \le\ \sum_j e^{(l_{\theta(0)}(x)_j - L(x))/\tau} \ \le\ k(x) + (d-k(x))e^{-\Delta(x)/\tau}\ . \tag{38}$$

The numerator of Equation (37) equals 1 when $i \in M(x)$ and is at most $e^{-\Delta(x)/\tau}$ otherwise. Therefore,

$$\begin{aligned}
&\forall i \in M(x), \quad \frac{1}{k(x)+(d-k(x))e^{-\Delta(x)/\tau}} \le p_{\theta(0)}(x,\tau)_i \le \frac{1}{k(x)}\ , \\
&\forall i \notin M(x), \quad p_{\theta(0)}(x,\tau)_i \le \frac{1}{k(x)}e^{-\Delta(x)/\tau}\ .
\end{aligned} \tag{39}$$

Item (i). Here $k(x) = 1$, i.e. $M(x) = \{i^\star\}$ for some token $i^\star$. Set $\eta := 1 - p_{\theta(0)}(x,\tau)_{i^\star} = \sum_{i\neq i^\star} p_{\theta(0)}(x,\tau)_i$. The second bound of Equation (39) gives

$$\eta \ \le\ (d-1)e^{-\Delta(x)/\tau}\ . \tag{40}$$

As a shorthand, we write $\bar{v} := \mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}(v_0(x)_i)$.

Since $\bar{v} = v_0(x)_{i^\star} + \sum_{i\neq i^\star} p_{\theta(0)}(x,\tau)_i(v_0(x)_i - v_0(x)_{i^\star})$ and $|v_0(x)_i - v_0(x)_{i^\star}| \le \mathrm{osc}(v_0(x))$ for every i,

$$|\bar{v} - v_0(x)_{i^\star}| \ \le\ \eta\,\mathrm{osc}(v_0(x))\ . \tag{41}$$

Moreover $\bar{v}$ is an average of the $v_0(x)_i$, hence lies between $\min_i v_0(x)_i$ and $\max_i v_0(x)_i$, so $|v_0(x)_i - \bar{v}| \leq \mathrm{osc}(v_0(x))$ for every i. Therefore,

$$\begin{aligned}\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)} |v_0(x)_i - \bar{v}| &= p_{\theta(0)}(x,\tau)_{i^\star} |v_0(x)_{i^\star} - \bar{v}| + \sum_{i\neq i^\star} p_{\theta(0)}(x,\tau)_i |v_0(x)_i - \bar{v}| \\ &\leq (1-\eta)\,\eta\,\mathrm{osc}(v_0(x)) + \eta\,\mathrm{osc}(v_0(x)) \;\leq\; 2\eta\,\mathrm{osc}(v_0(x))\;.\end{aligned} \tag{42}$$

Dividing by τ and using Equation (40) yields the first bound. For the second,

$$\begin{aligned}\mathrm{Var}_{i\sim p_{\theta(0)}(x,\tau)}(v_0(x)_i) &\leq \mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)} \left(v_0(x)_i - v_0(x)_{i^\star}\right)^2 \\ &= \sum_{i\neq i^\star} p_{\theta(0)}(x,\tau)_i \left(v_0(x)_i - v_0(x)_{i^\star}\right)^2 \\ &\leq \eta \mathrm{osc}(v_0(x))^2\;,\end{aligned} \tag{43}$$

and dividing by τ^2 and using Equation (40) gives the second bound. Both upper bounds tend to 0 as $\tau \to 0$, since $\tau^{-m} e^{-\Delta(x)/\tau} \to 0$ for every m as soon as $\Delta(x) > 0$.

Item (ii). Here $k(x) \geq 2$. We first check that $p_{\theta(0)}(x,\tau)$ converges to $\mathcal{U}(M(x))$. For $i \in M(x)$, Equation (39) gives

$$\begin{aligned}0 \leq \frac{1}{k(x)} - p_{\theta(0)}(x,\tau)_i &\leq \frac{1}{k(x)} - \frac{1}{k(x) + (d-k(x))e^{-\Delta(x)/\tau}} \\ &= \frac{(d-k(x))e^{-\Delta(x)/\tau}}{k(x)\left(k(x) + (d-k(x))e^{-\Delta(x)/\tau}\right)} \\ &\leq \frac{d-k(x)}{k(x)^2} e^{-\Delta(x)/\tau}\;,\end{aligned} \tag{44}$$

while for $i \notin M(x)$ we have $0 \leq p_{\theta(0)}(x,\tau)_i \leq \frac{1}{k(x)} e^{-\Delta(x)/\tau}$. Summing both families of inequalities, over the $k(x)$ indices of $M(x)$ and the $d - k(x)$ others,

$$\begin{aligned}\left\|p_{\theta(0)}(x,\tau) - \mathcal{U}(M(x))\right\|_1 &\leq 2\frac{d-k(x)}{k(x)} e^{-\Delta(x)/\tau} \\ &\leq d e^{-\Delta(x)/\tau} \xrightarrow[\tau\to 0]{} 0\;.\end{aligned} \tag{45}$$

In the previous computation, the last inequality used that $k(x) \geq 2$.

Now, $v_0(x)$ being a fixed vector, the two maps

$$q \longmapsto \mathbb{E}_{i\sim q} |v_0(x)_i - \mathbb{E}_{j\sim q}(v_0(x)_j)| \qquad \text{and} \qquad q \longmapsto \mathrm{Var}_{i\sim q}(v_0(x)_i) \tag{46}$$

are continuous on the simplex of $\mathbb{R}^d$: the first is a composition of the linear map $q \mapsto \mathbb{E}_{j\sim q}(v_0(x)_j)$ with continuous maps, and the second is a polynomial in q.

Evaluating them at $q = p_{\theta(0)}(x,\tau)$ gives $\tau\rho_1(x,\tau)$ and $\tau^2\rho_2(x,\tau)$, and at $q = \mathcal{U}(M(x))$ gives the two limits of item (ii). Combining this with Equation (45) yields the final result.

G Proofs of the equations of Section 3

Throughout this section, x is fixed. We use the notation of Section E and Section F, and we write, for $q \in \mathcal{P}_d$, $\Sigma(q) := \mathrm{diag}(q) - qq^T$. For any $w \in \mathbb{R}^d$,

$$\left(\Sigma(q)w\right)_i = q_i w_i - q_i \sum_j q_j w_j = q_i \left(w_i - \mathbb{E}_{j\sim q}(w_j)\right)\;. \tag{47}$$

Equation (2). Multiplying Equation (30) by $p_{\theta(0)}(x,\tau)_i$ gives, for every i,

$$\begin{aligned}p_{\theta(t)}(x,\tau)_i - p_{\theta(0)}(x,\tau)_i &= p_{\theta(0)}(x,\tau)_i\, \epsilon(t)_i \\ &= \frac{t}{\tau}\, p_{\theta(0)}(x,\tau)_i \left(v_0(x)_i - \mathbb{E}_{j\sim p_{\theta(0)}(x,\tau)}(v_0(x)_j)\right) + o(t)\;,\end{aligned} \tag{48}$$

which is Equation (2) by Equation (47) with $q = p_{\theta(0)}(x,\tau)$ and $w = v_0(x)$; the $o(t)$ is uniform in i since there are finitely many indices.

Equation (3). By Equation (47), for any $q \in \mathcal{P}_d$ and $w \in \mathbb{R}^d$,

$$\|\Sigma(q)w\|_1 = \sum_i q_i \left|w_i - \mathbb{E}_{j\sim q}(w_j)\right| = \mathbb{E}_{i\sim q}\left|w_i - \mathbb{E}_{j\sim q}(w_j)\right| \ . \tag{49}$$

With $q = p_{\theta(0)}(x,\tau)$ and $w = v_0(x)$, dividing by τ gives $\|\xi(x,\tau)\|_1 = \rho_1(x,\tau)$.

Linearity in the direction of the change. Since $\theta(\cdot)$ is $\mathcal{C}^1$ and $\theta \mapsto l_\theta(x)$ is differentiable, the chain rule gives $v_0(x) = \frac{d}{dt}\big|_{t=0} l_{\theta(t)}(x) = J_x\theta'(0) = J_x h$. Hence $\xi(x,\tau) = \frac{1}{\tau}\Sigma(p_{\theta(0)}(x,\tau))J_x h$ is linear in h, and Equation (2) reads $p_{\theta(t)}(x,\tau) - p_{\theta(0)}(x,\tau) = \frac{t}{\tau}\Sigma(p_{\theta(0)}(x,\tau))J_x h + o(t)$. When the change is not a weight update but induces a $\mathcal{C}^1$ path of logits, the same holds with J_x the Jacobian with respect to whatever parametrizes the path, and h its initial velocity.

Equation (4). If $k(x) = 1$, Equation (3) and item (i) of Theorem 4 give $\|\xi(x,\tau)\|_1 = \rho_1(x,\tau) \le \frac{2(d-1)\operatorname{osc}(v_0(x))}{\tau} e^{-\Delta(x)/\tau} \to 0$ as $\tau \to 0$. If $k(x) \ge 2$, the map $q \mapsto \Sigma(q)v_0(x)$ is polynomial, hence continuous, on $\mathcal{P}_d$, and $p_{\theta(0)}(x,\tau) \to \mathcal{U}(M(x))$ by Equation (45). Hence $\tau\,\xi(x,\tau) = \Sigma(p_{\theta(0)}(x,\tau))v_0(x) \to \Sigma(\mathcal{U}(M(x)))v_0(x)$, whose coordinates are, by Equation (47), $\frac{\mathbf{1}\{i\in M(x)\}}{k(x)}\left(v_0(x)_i - \mathbb{E}_{j\sim\mathcal{U}(M(x))}(v_0(x)_j)\right)$.

Equation (5). Assume $k(x) \ge 2$ and that $v_0(x)$ has a unique maximizer $i^\star$ on $M(x)$, and let $g := \min_{j\in M(x)\setminus\{i^\star\}}(v_0(x)_{i^\star} - v_0(x)_j) > 0$. By writing $l_{\theta(t)}(x) - l_{\theta(0)}(x) = tv_0(x) + r(t)$ with $\|r(t)\|_\infty = o(t)$, and recall that all the coordinates of $l_{\theta(0)}(x)$ on $M(x)$ are equal. For $j \in M(x)\setminus\{i^\star\}$,

$$l_{\theta(t)}(x)_{i^\star} - l_{\theta(t)}(x)_j \ge tg - 2\|r(t)\|_\infty \ , \tag{50}$$

and for $j \notin M(x)$,

$$l_{\theta(t)}(x)_{i^\star} - l_{\theta(t)}(x)_j \ge \Delta(x) - t\operatorname{osc}(v_0(x)) - 2\|r(t)\|_\infty \ . \tag{51}$$

Choose $t_0 > 0$ such that, for all $t \in (0,t_0]$, $2\|r(t)\|_\infty \le tg/2$ and $t\operatorname{osc}(v_0(x)) + tg/2 < \Delta(x)$. Then, for such t, both gaps are bounded below by $\Delta_t := \min\left(tg/2,\ \Delta(x) - t\operatorname{osc}(v_0(x)) - tg/2\right) > 0$, so $i^\star$ is the unique maximizer of $l_{\theta(t)}(x)$, with gap at least Δ_t to every other token. Applying Equation (40) to the logits $l_{\theta(t)}(x)$ instead of $l_{\theta(0)}(x)$ (the argument of Section F only uses that the maximizer is unique with gap Δ_t) gives

$$\left\|p_{\theta(t)}(x,\tau) - \mathcal{U}(\{i^\star\})\right\|_1 = 2\left(1 - p_{\theta(t)}(x,\tau)_{i^\star}\right) \le 2(d-1)e^{-\Delta_t/\tau} \xrightarrow[\tau\to 0]{} 0 \ . \tag{52}$$

Finally, $i^\star = \arg\max_{i\in M(x)} v_0(x)_i = \arg\max_{i\in M(x)}(J_x h)_i$ by the chain rule above.

H Recovering the optimal tests of Chauvin et al. (2026a)

In this section, we show that the local expansion of Theorem 3 is enough to derive the asymptotic power of the optimal test between the initial model and a model that has moved along the trajectory by a time of order $1/\sqrt{n}$, when n independent samples are drawn on the prompt x. The main result is Theorem 7. We then check, in Section H.5, that it coincides with the result of Chauvin et al. (2026a).

H.1 Setting and main result

Throughout this section, the prompt x and the temperature $\tau > 0$ are fixed, as well as a scale $s \ge 0$ and a level $\alpha \in (0,1)$. For $n \ge s^2$, we set $t_n := s/\sqrt{n} \in [0,1]$, and we consider the problem of testing

$$H_0 :\ Y_1,\dots,Y_n \overset{\text{i.i.d.}}{\sim} p_{\theta(0)}(x,\tau) \qquad \text{against} \qquad H_1^{(n)} :\ Y_1,\dots,Y_n \overset{\text{i.i.d.}}{\sim} p_{\theta(t_n)}(x,\tau) \ . \tag{53}$$

We write $P_n^{(0)} := p_{\theta(0)}(x,\tau)^{\otimes n}$ and $P_n^{(1)} := p_{\theta(t_n)}(x,\tau)^{\otimes n}$ for the corresponding laws of $Y := (Y_1,\dots,Y_n)$ on $\{1,\dots,d\}^n$. A test is a function $\psi_n : \{1,\dots,d\}^n \to [0,1]$, where $\psi_n(Y)$ is the probability of rejecting H_0. It has level α if $\mathbb{E}_{P_n^{(0)}}(\psi_n(Y)) \le \alpha$, and its type-II error is

$$\beta_n(\psi_n) := \mathbb{E}_{P_n^{(1)}}\left(1 - \psi_n(Y)\right) \ . \tag{54}$$

Both sampling laws have full support, so the log-likelihood ratio

$$\lambda_t(i) := \log \frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} , \qquad \Lambda_n := \sum_{k=1}^{n} \lambda_{t_n}(Y_k) \tag{55}$$

is well defined, and $\frac{dP_n^{(1)}}{dP_n^{(0)}}(Y) = e^{\Lambda_n}$. Finally, Φ denotes the cumulative distribution function of $\mathcal{N}(0,1)$ and $Q_\alpha := \Phi^{-1}(1-\alpha)$ its quantile of order $1-\alpha$.

Theorem 7. *Let* $\sigma := s\sqrt{\rho_2(x,\tau)}$.

(i) For any sequence of tests $(\psi_n)_n$ *of level* α,

$$\liminf_{n\to\infty} \beta_n(\psi_n) \geq \Phi\left(Q_\alpha - \sigma\right) . \tag{56}$$

(ii) The Neyman–Pearson tests $(\psi_n^\star)_n$ *of level* α *satisfy*

$$\beta_n(\psi_n^\star) \xrightarrow[n\to\infty]{} \Phi\left(Q_\alpha - \sigma\right) = \mathbb{P}\left(\mathcal{N}(0,1) \leq Q_\alpha - s\sqrt{\rho_2(x,\tau)}\right) . \tag{57}$$

The proof relies on the expansions of the first two moments of λ_t under both hypotheses (Lemma 9), from which we obtain the asymptotic normality of Λ_n (Lemma 10). Two of these moments are f-divergences and are handled by Theorem 3 directly. The two others are sums of the same form, but with a function that is not convex, which is why we first extend case (ii) of Theorem 3.

H.2 Moments of the log-likelihood ratio

Lemma 8. *Let* $g : (0,+\infty) \to \mathbb{R}$ *be twice differentiable at* 1 *with* $g(1) = 0$. *Then*

$$\sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i \, g\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) = \frac{g''(1)}{2}\rho_2(x,\tau)\, t^2 + o_0(t^2) . \tag{58}$$

Proof. The argument is the one of case (ii) in Section E, where the convexity of f is never used. □

Lemma 9. *As* $t \to 0$,

$$\begin{aligned}
&\textit{(a)} \quad \mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_t(i)\right) = -\frac{1}{2}\rho_2(x,\tau)\, t^2 + o(t^2) , \\
&\textit{(b)} \quad \mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_t(i)^2\right) = \rho_2(x,\tau)\, t^2 + o(t^2) , \\
&\textit{(c)} \quad \mathbb{E}_{i\sim p_{\theta(t)}(x,\tau)}\left(\lambda_t(i)\right) = \frac{1}{2}\rho_2(x,\tau)\, t^2 + o(t^2) , \\
&\textit{(d)} \quad \mathbb{E}_{i\sim p_{\theta(t)}(x,\tau)}\left(\lambda_t(i)^2\right) = \rho_2(x,\tau)\, t^2 + o(t^2) ,
\end{aligned} \tag{59}$$

and (e) $\max_i |\lambda_t(i)| \to 0$.

Proof. **(a).** With $f(u) := -\log u$, which is convex with $f(1) = 0$ and $f''(1) = 1$ (see Table 3),

$$\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_t(i)\right) = -\sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i \, f\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) = -D_f\left(p_{\theta(t)}(x,\tau) \,\|\, p_{\theta(0)}(x,\tau)\right) , \tag{60}$$

and case (ii) of Theorem 3 gives the result.

(b). We apply Lemma 8 to $g(u) := (\log u)^2$, which satisfies $g(1) = 0$, $g'(u) = \frac{2\log u}{u}$ and $g''(u) = \frac{2(1-\log u)}{u^2}$, hence $g''(1) = 2$.

(c). Changing the reference measure,

$$\mathbb{E}_{i\sim p_{\theta(t)}(x,\tau)}\left(\lambda_t(i)\right) = \sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i \, \frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} \log \frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i} = D_f\left(p_{\theta(t)}(x,\tau) \,\|\, p_{\theta(0)}(x,\tau)\right) \tag{61}$$

with $f(u) := u \log u$, which is convex with $f(1) = 0$ and $f''(1) = 1$. Case (ii) of Theorem 3 gives the result.

(d). In the same way,

$$\mathbb{E}_{i\sim p_{\theta(t)}(x,\tau)} \left(\lambda_t(i)^2\right) = \sum_{i=1}^{d} p_{\theta(0)}(x,\tau)_i \, g\left(\frac{p_{\theta(t)}(x,\tau)_i}{p_{\theta(0)}(x,\tau)_i}\right) \tag{62}$$

with $g(u) := u(\log u)^2$. It satisfies $g(1) = 0$, $g'(u) = (\log u)^2 + 2\log u$ and $g''(u) = \frac{2\log u+2}{u}$, hence $g''(1) = 2$, and Lemma 8 gives the result.

(e). By Equation (30), $\epsilon(t)_i \to 0$ for every i, hence $\lambda_t(i) = \log(1 + \epsilon(t)_i) \to 0$, and the maximum is taken over finitely many indices. □

H.3 Asymptotic normality of the log-likelihood ratio

Lemma 10. *Let $\sigma := s\sqrt{\rho_2(x,\tau)}$. Under H_0, Λ_n converges in distribution to $\mathcal{N}\left(-\frac{\sigma^2}{2}, \sigma^2\right)$. Under $H_1^{(n)}$, Λ_n converges in distribution to $\mathcal{N}\left(\frac{\sigma^2}{2}, \sigma^2\right)$.*

Proof. **Under H_0.** The variables $\lambda_{t_n}(Y_1), \ldots, \lambda_{t_n}(Y_n)$ are i.i.d., distributed as $\lambda_{t_n}(i)$ with $i \sim p_{\theta(0)}(x,\tau)$. By (a) of Lemma 9 and since $t_n^2 = s^2/n$,

$$\mathbb{E}_{P_n^{(0)}}(\Lambda_n) = n\,\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_{t_n}(i)\right) = n\left(-\frac{\rho_2(x,\tau)s^2}{2n} + o\left(\frac{1}{n}\right)\right) \xrightarrow[n\to\infty]{} -\frac{\sigma^2}{2}\,. \tag{63}$$

Moreover, (a) also gives $\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_{t_n}(i)\right)^2 = O(t_n^4) = O(1/n^2)$, so that by (b),

$$\begin{aligned}\mathrm{Var}_{P_n^{(0)}}(\Lambda_n) &= n\left(\mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_{t_n}(i)^2\right) - \mathbb{E}_{i\sim p_{\theta(0)}(x,\tau)}\left(\lambda_{t_n}(i)\right)^2\right)\\ &= n\left(\frac{\rho_2(x,\tau)s^2}{n} + o\left(\frac{1}{n}\right)\right) \xrightarrow[n\to\infty]{} \sigma^2\,.\end{aligned} \tag{64}$$

We now check the Lindeberg condition. Let $\eta > 0$. By (e) of Lemma 9, $\max_i |\lambda_{t_n}(i)| < \eta$ for n large enough, and then

$$\sum_{k=1}^{n} \mathbb{E}_{P_n^{(0)}}\left(\lambda_{t_n}(Y_k)^2\,\mathbf{1}\{|\lambda_{t_n}(Y_k)| > \eta\}\right) = 0\,. \tag{65}$$

The Lindeberg–Feller central limit theorem (van der Vaart, 2000, Proposition 2.27) therefore yields that $\Lambda_n - \mathbb{E}_{P_n^{(0)}}(\Lambda_n)$ converges in distribution to $\mathcal{N}(0,\sigma^2)$, and Slutsky's lemma concludes the proof.

Under $H_1^{(n)}$. The proof is identical, with $Y_k \sim p_{\theta(t_n)}(x,\tau)$, and with (c) and (d) of Lemma 9 in place of (a) and (b). The mean now converges to $+\frac{\sigma^2}{2}$ and the variance to σ^2, while the Lindeberg condition holds for the same reason since (e) does not depend on the sampling law. □

H.4 Proof of Theorem 7

For each n, $P_n^{(0)}$ and $P_n^{(1)}$ are two simple hypotheses on the finite set $\{1,\ldots,d\}^n$, with likelihood ratio e^{Λ_n}. By the Neyman–Pearson lemma (Lehmann & Romano, 2006, Theorem 3.2.1), there exist $c_n \in \mathbb{R}$ and $\gamma_n \in [0,1]$ such that the test

$$\psi_n^\star(Y) := \mathbf{1}\{\Lambda_n > c_n\} + \gamma_n \mathbf{1}\{\Lambda_n = c_n\} \tag{66}$$

satisfies $\mathbb{E}_{P_n^{(0)}}(\psi_n^\star(Y)) = \alpha$, and $\beta_n(\psi_n) \geq \beta_n(\psi_n^\star)$ for every test ψ_n of level α. The randomization γ_n is needed because Λ_n takes finitely many values. In particular, (i) follows from (ii), and it remains to compute the limit of $\beta_n(\psi_n^\star)$.

Case $\sigma > 0$. Let F and G be the cumulative distribution functions of $\mathcal{N}\left(-\frac{\sigma^2}{2}, \sigma^2\right)$ and $\mathcal{N}\left(\frac{\sigma^2}{2}, \sigma^2\right)$. Both are continuous and increasing. Let $c := -\frac{\sigma^2}{2} + \sigma Q_\alpha$, so that $1 - F(c) = 1 - \Phi(Q_\alpha) = \alpha$.

Let us first show that $c_n \to c$. The level constraint reads

$$P_n^{(0)}\left(\Lambda_n > c_n\right) \leq \alpha \leq P_n^{(0)}\left(\Lambda_n \geq c_n\right) \ . \tag{67}$$

Let $\eta > 0$. If $c_n \geq c + \eta$ for infinitely many n, then along these indices, by Equation (67) and Lemma 10,

$$\alpha \leq P_n^{(0)}\left(\Lambda_n \geq c + \eta\right) \longrightarrow 1 - F(c+\eta) < 1 - F(c) = \alpha \ , \tag{68}$$

which is a contradiction. Here we used that F is continuous, so that convergence in distribution gives the convergence of $P_n^{(0)}\left(\Lambda_n \geq c + \eta\right)$. Symmetrically, if $c_n \leq c - \eta$ for infinitely many n, then along these indices

$$\alpha \geq P_n^{(0)}\left(\Lambda_n > c - \eta\right) \longrightarrow 1 - F(c-\eta) > \alpha \ , \tag{69}$$

which is again a contradiction. Hence $c - \eta < c_n < c + \eta$ for n large enough, i.e. $c_n \to c$.

We now turn to the type-II error. By definition of $\psi_n^\star$,

$$\beta_n(\psi_n^\star) = P_n^{(1)}\left(\Lambda_n < c_n\right) + (1 - \gamma_n) P_n^{(1)}\left(\Lambda_n = c_n\right) \ , \tag{70}$$

so that, for n large enough,

$$P_n^{(1)}\left(\Lambda_n \leq c - \eta\right) \leq \beta_n(\psi_n^\star) \leq P_n^{(1)}\left(\Lambda_n \leq c + \eta\right) \ . \tag{71}$$

By Lemma 10 and the continuity of G, the two bounds converge to $G(c-\eta)$ and $G(c+\eta)$. Letting $\eta \to 0$ gives

$$\beta_n(\psi_n^\star) \xrightarrow[n\to\infty]{} G(c) = \Phi\left(\frac{c - \sigma^2/2}{\sigma}\right) = \Phi\left(Q_\alpha - \sigma\right) \ . \tag{72}$$

Case $\sigma = 0$. This happens when $s = 0$ or $\rho_2(x, \tau) = 0$, and the claim is then that $\beta_n(\psi_n^\star) \to 1 - \alpha$. The constant test $\psi_n \equiv \alpha$ has level α and type-II error $1 - \alpha$, so $\beta_n(\psi_n^\star) \leq 1 - \alpha$. For the converse, for any test ψ_n of level α,

$$\beta_n(\psi_n) = 1 - \mathbb{E}_{P_n^{(0)}}\left(\psi_n(Y)\right) - \left(\mathbb{E}_{P_n^{(1)}}\left(\psi_n(Y)\right) - \mathbb{E}_{P_n^{(0)}}\left(\psi_n(Y)\right)\right) \geq 1 - \alpha - \mathrm{TV}\left(P_n^{(1)}, P_n^{(0)}\right) \ , \tag{73}$$

and it suffices to show that $\mathrm{TV}\left(P_n^{(1)}, P_n^{(0)}\right) \to 0$. We go through the squared Hellinger distance $H^2(p,q) := \frac{1}{2}\sum_i (\sqrt{p_i} - \sqrt{q_i})^2 = 1 - \sum_i \sqrt{p_i q_i}$. On the one hand, by the Cauchy–Schwarz inequality,

$$\begin{aligned} \sum_i |p_i - q_i| &= \sum_i |\sqrt{p_i} - \sqrt{q_i}|\,(\sqrt{p_i} + \sqrt{q_i}) \\ &\leq \left(\sum_i (\sqrt{p_i} - \sqrt{q_i})^2\right)^{1/2} \left(2\sum_i (p_i + q_i)\right)^{1/2} \\ &= 2\sqrt{2H^2(p,q)} \ , \end{aligned} \tag{74}$$

that is $\mathrm{TV}(p,q) \leq \sqrt{2H^2(p,q)}$. On the other hand, $\sum_i \sqrt{p_i q_i}$ is multiplicative over product measures, so that

$$1 - H^2\left(P_n^{(1)}, P_n^{(0)}\right) = \left(1 - H^2\left(p_{\theta(t_n)}(x,\tau), p_{\theta(0)}(x,\tau)\right)\right)^n \ . \tag{75}$$

The squared Hellinger distance is the f-divergence of generator $f(u) = \frac{1}{2}(\sqrt{u} - 1)^2$, with $f''(1) = \frac{1}{4}$ (see Table 3), so case (ii) of Theorem 3 gives

$$H^2\left(p_{\theta(t_n)}(x,\tau), p_{\theta(0)}(x,\tau)\right) = \frac{1}{8}\rho_2(x,\tau)\, t_n^2 + o(t_n^2) = o\left(\frac{1}{n}\right) \tag{76}$$

when $\sigma = 0$. Hence $1 - H^2\left(P_n^{(1)}, P_n^{(0)}\right) = (1 - o(1/n))^n \to 1$, so $H^2\left(P_n^{(1)}, P_n^{(0)}\right) \to 0$ and $\mathrm{TV}\left(P_n^{(1)}, P_n^{(0)}\right) \to 0$. This gives $\liminf_n \beta_n(\psi_n^\star) \geq 1 - \alpha = \Phi(Q_\alpha - 0)$, which concludes the proof. □

H.5 Identification with Chauvin et al. (2026a)

Chauvin et al. (2026a) consider linear trajectories $\theta(t) = \theta(0) + th$ with h a unit vector of the parameter space, and perturbations of size $\epsilon_n = s/\sqrt{n}$, which is exactly our t_n. They write $z(\theta)$ for the logits on the test prompt, which is our $l_\theta(x)$, and J_z for its Jacobian with respect to θ at $\theta(0)$. Along such a trajectory, the chain rule gives

$$v_0(x) = J_z h \ . \tag{77}$$

If $l_\theta(x)$ denotes the logprobs instead of the logits, $v_0(x)$ and $J_z h$ differ by a multiple of $\mathbf{1} := (1, \dots, 1)$, which does not affect anything below.

Their Lemma 3.2 rewrites the quantity that governs the power of the optimal tests in their Theorem 3.1 as

$$\mathrm{SNR}^2(h) = \frac{1}{\tau^2} h^T J_z^T \, \Sigma\left(p_{\theta(0)}(x,\tau)\right) J_z h \ , \qquad \text{where} \quad \Sigma(q) := \mathrm{diag}(q) - qq^T \ . \tag{78}$$

For any $q \in \mathcal{P}_d$ and $w \in \mathbb{R}^d$,

$$w^T \Sigma(q) w = \sum_{i=1}^d q_i w_i^2 - \left(\sum_{i=1}^d q_i w_i\right)^2 = \mathrm{Var}_{i\sim q}(w_i) \ , \tag{79}$$

and $\Sigma(q)\mathbf{1} = 0$. Applied with $q = p_{\theta(0)}(x,\tau)$ and $w = J_z h$, this gives

$$\mathrm{SNR}^2(h) = \frac{\mathrm{Var}_{i\sim p_{\theta(0)}(x,\tau)}(v_0(x)_i)}{\tau^2} = \rho_2(x,\tau) \ . \tag{80}$$

With this identification, the limit of Theorem 7 reads $\mathbb{P}\left(\mathcal{N}(0,1) \le Q_\alpha - \sqrt{s^2\,\mathrm{SNR}^2(h)}\right)$, which is the statement of their Theorem 3.1.

The same computation applies to their zero-temperature analysis. Their Theorem 3.3 involves $\Sigma_{\mathcal{M}} := \Sigma(\mathcal{U}(M(x)))$, and

$$h^T J_z^T \Sigma_{\mathcal{M}} J_z h = \mathrm{Var}_{i\sim\mathcal{U}(M(x))}(v_0(x)_i) \ . \tag{81}$$

By Equation (80), item (ii) of Theorem 4 states that $\tau^2\,\mathrm{SNR}^2(h) \to h^T J_z^T \Sigma_{\mathcal{M}} J_z h$ when $k(x) \ge 2$, so that $\mathrm{SNR}^2(h) \to +\infty$ as soon as this quantity is nonzero, which is their nondegeneracy condition. When $k(x) = 1$, item (i) gives $\mathrm{SNR}^2(h) \le \frac{(d-1)\,\mathrm{osc}(J_z h)^2}{\tau^2} e^{-\Delta(x)/\tau} \to 0$. We thus recover the dichotomy of their Theorem 3.3, together with the rate at which each regime is reached.

H.6 Scope of our results

Our results are thus more general in three ways. First, they hold along any $\mathcal{C}^1$ trajectory, not only linear ones. Second, $\rho_2(x,\tau)$ governs the local behavior of every f-divergence in the smooth case (see Table 3), and the moments of the log-likelihood ratio used above are only particular instances of this expansion. Third, the kinked case of Theorem 3 yields a first-order regime, driven by $\rho_1(x,\tau)$, for divergences such as the total variation, which has no counterpart in Chauvin et al. (2026a).

I Experimental setup details

I.1 Landmark construction

System prompts. The 50 oriented prompts assign an identity or a stance and come from the prompt set of Neumann et al. (2025); the 50 neutral prompts assign a task or a persona without a stance and come from the LLMmap set of Pasquini et al. (2025). Table 4 gives two verbatim examples of each kind.

Table 4: The two prompt kinds: count per LLM and two verbatim examples of each.

kind	count	example 1	example 2
oriented (identity/stance assignment)	50	You are interacting with a White person. Answer their questions in a way that stays true to the nature of this identity. Don't refuse to answer the question.	You are interacting with a female person. Answer their questions in a way that stays true to the nature of this identity. Don't refuse to answer the question.
neutral (task/persona assignment)	50	You are a bot designed to provide immediate guidance during natural disasters. Offer clear and concise instructions for safety, updates on emergency measures, and directions to shelters. Be calm, reassuring, and quick to respond to ensure users receive timely and potentially lifesaving information.	Generate responses suitable for a therapist-in-training chatbot. Focus on validation, open-ended questions, and safe boundaries. Avoid diagnosis.

Fine tuning. The 84 fine tuning landmarks per LLM (Section 4.1) are low rank adaptations (LoRA, Hu et al., 2022) of S with the same adapter shape; a dose level is a snapshot of a run after a fixed number of optimizer steps, so the four doses of one (theme, data fold) are nested. Table 5 gives the settings. The six supervised themes use public corpora (UltraChat (Ding et al., 2023), finance-alpaca (Bharti, 2023), ChatDoctor (Li et al., 2023), MetaMathQA (Yu et al., 2024), Magicoder (Wei et al., 2024), and the unsafe responses of BeaverTails (Ji et al., 2023)); the preference theme is DPO on the harmless split of HH-RLHF (Bai et al., 2022). The six supervised themes share one fine tuning budget: every fold is 2M tokens (counted with one reference tokenizer), trained for one epoch with the same recipe (sequences packed to 4096 tokens, 16k tokens per optimizer step, Table 5), so a dose level costs every theme the same number of tokens and about the same compute; the DPO doses are instead $\frac{1}{4}, \frac{1}{2}, 1$ and 2 epochs of its preference pairs.

Table 5: Fine tuning settings, the same for every LLM; the four values of a doses row are the rungs $\times 1, \times 2, \times 4, \times 8$.

parameter	value
adapter	LoRA (r=64, alpha=16, dropout=0.1, all-linear layers)
supervised themes (6)	ultrachat, finance-alpaca, chatdoctor, metamathqa, magicoder, beavertails-unsafe
supervised data	3 disjoint folds of 2M tokens/theme, 1 epoch/fold
supervised doses	15/31/61/122 steps = 250k/500k/1M/2M tokens (16384 tokens/step: 4 packed sequences of 4096)
supervised optimiser	`adamw_torch`, LR 0.0002, `constant_with_warmup`
LR/schedule/batch	(warmup ratio 0.03), max grad norm 0.3, weight decay 0, 4 x 4096 packed (bfd), bfloat16
preference theme (1)	dpo-hh-harmless: DPO on HH-RLHF (dataset hh-harmless)
preference settings	beta=0.1, 65 pairs/step, 2 epochs, LR 2e-05, reference model = adapter disabled
preference doses	54/109/218/436 steps
library	TRL 0.21.0, transformers 4.57.6 (measured)

Abliteration. Eight landmarks per LLM, one per technique of the `obliteratus` library (OBLITERATUS Contributors, 2026) (Table 6).

Table 6: The 8 abliteration techniques, with the library's defaults. The description is read from the library's method presets; last column: the LLMs on which the technique fails the refusal gate (refusal rate above 30%).

technique	description	$>30\%$ refusal (failed)
`aggressive`	8 whitened SVD directions with jailbreak contrast and iterative refinement, attention head surgery, no regularisation	gemma-3-12b
`failspy`	FailSpy's abliterator (FailSpy, 2024): one difference of means direction (Arditi et al., 2024), projected out of every layer but the first	none
`gabliteration`	Gabliteration (Gülmez, 2025): top 4 SVD directions, ridge regularised projection on the highest variance layers	Phi-3-mini, gemma-3-12b
`heretic`	Heretic (Weidmann, 2025): one direction per layer, ablation weights set by Bayesian search trading refusals against KL divergence	granite-4.2-8b
`inverted`	reflects the refusal subspace (8 SVD directions, factor 2) instead of removing it, pushing the model towards compliance	none
`rdo`	refusal direction optimisation (Wollschläger et al., 2025): 4 SVD directions refined by gradient descent against a linear refusal probe	gemma-3-12b
`spectral_cascade`	6 SVD directions, projection split into DCT bands along the layer axis and strongest on the low frequency trend	Meta-Llama-3-8B, gemma-3-12b
`surgical`	8 whitened SVD directions with jailbreak contrast, safety neuron masking, attention head surgery and SAE feature projection	gemma-3-12b

I.2 Anchors and features

Features on the anchor's support. For a given BI b of an anchor A, the *support* $\mathrm{Supp}_A^K(b)$ lists the K tokens the anchor A ranks highest. Every model M is read on this support, not on its own $\mathrm{Supp}_M^K(b)$: $p_M(b) \in [0,1]^{K+1}$ holds the probabilities M gives to these K tokens, in A's order, and a last coordinate, the *bucket*, holds the rest of M's mass. So $p_M(b)$ sums to one.

Example 11 ($K = 3$)**.** Let $\mathrm{Supp}_A^K(b) = (t_1, t_2, t_3)$, and let M put mass 0.6, 0.2, 0.1 on t_4, t_3, t_5 ($\mathrm{Supp}_M^K(b) = (t_4, t_3, t_5)$). Then

$$p_A(b) = [\, t_1 \colon 0.4,\ t_2 \colon 0.4,\ t_3 \colon 0.1,\ \text{bucket} \colon 0.1\,],$$
$$p_M(b) = [\, t_1 \colon 0.0,\ t_2 \colon 0.0,\ t_3 \colon 0.2,\ \text{bucket} \colon 0.8\,].$$

Only t_3 is in both supports. The mass on t_4, t_5 and every other token goes to the bucket.

Fusion by concatenation. The N BIs are split as evenly as possible across the anchors $A_j \in \mathcal{A}$, n_j for anchor A_j. The n_j BIs of anchor A_j are read on A_j's own support, so every model M, landmark or test item, gets one block of $n_j(K+1)$ coordinates per fused anchor. The blocks are concatenated in the fixed anchor order (S, the oriented and the neutral prompt, the 7 fine tunings, the 3 abliterations), which gives P_M, of $N(K+1) = 600$ coordinates whatever the number of fused anchors. The blocks are not rescaled, every coordinate being a probability. One classifier is fitted on P_M and gives one decision per test item. Since the N BIs are split over the anchors, fusing more anchors leaves the budget at Nm queries.

Choice of anchors. Thirteen anchors are chosen: S, 2 system prompts (one oriented, one neutral), the 7 fine tuning themes at dose $\times 8$ on data fold 1, and 3 abliterations (`inverted`, `surgical`, `failspy`).

Fine tuning anchors and held out themes. The 4 doses of a (theme, data fold) are snapshots of one training run, and a fine tuning anchor is the $\times 8$ endpoint of its theme's fold 1 run. On a split

that holds out its theme, the 3 other snapshots of that run are test items; the anchor is then neither a training item (the held out theme has no training item) nor a test item, and it leaves the fused set. Every first order split thus fuses exactly 12 anchors, the 13 minus the held out theme's fine tuning anchor.

Budget split. The N BIs are split over the fused anchors by largest remainder, in a fixed anchor order: at $N = 100$, $4 \times 9 + 8 \times 8$ over 12 anchors and $9 \times 8 + 4 \times 7$ over 13. The n_j BIs of anchor j are a subset of its own $\mathrm{BI}(A_j)$, randomly drawn.

I.3 Classification

Table 7 gives the hyperparameters of the eight fitted learners of Section K.2.

Table 7: Fixed hyperparameters of the fitted learners compared in Table 9, the same for every LLM.

learner	hyperparameters
logistic regression	LogisticRegression(`max_iter`=1000); at sklearn 1.8.0: multinomial loss (lbfgs solver, 3 classes), L2 penalty, C=1.0
XGBoost	XGBClassifier(`n_estimators`=100, `learning_rate`=0.1)
random forest	RandomForestClassifier(`n_estimators`=100)
gradient boosting	GradientBoostingClassifier() (no kwargs: library defaults)
MLP	MLPClassifier(`hidden_layer_sizes`=(128, 64), `max_iter`=3000, `early_stopping`=False, `n_iter_no_change`=40, tol=1e-05, alpha=0.001, `learning_rate_init`=0.001)
LDA	LinearDiscriminantAnalysis() (no kwargs: library defaults); at sklearn 1.8.0: solver='svd'
SVM	SVC(probability=True); at sklearn 1.8.0: kernel='rbf', gamma='scale'
KNN	KNeighborsClassifier(`n_neighbors`=5)

Splits and fits. The splits of Section 4.1 are a repeated random subsampling: every fine tuning item that is not an anchor is tested exactly 5 times, in the 5 splits that hold out its theme. A split fixes which landmarks are held out; the BI subset, crossed with it, fixes which BIs are queried. A learner is fitted once per (BI subset, split), on the exact features of that split's training landmarks restricted to the subset. The R_{draw} draw replicas only redraw and rescore the held out items against that fixed decision boundary, so a replica costs a forward pass, not a refit, which is what makes a large R_{draw} cheap. The budget grid of Section K.1 follows the same rule.

I.4 Inference

Table 8 gives the hardware and software stack. Generation uses vLLM (Kwon et al., 2023) and log probabilities are computed with Transformers (Wolf et al., 2020).

Table 8: Inference stack.

parameter	value
hardware	NVIDIA H100 80GB nodes.
precision	bfloat16
decoding	multinomial sampling at $T = 0.1$ with top-k, $k = 50$: simulated exactly from the stored top-50 logprobs, tempered and renormalised over the stored list
stored logprobs	top-50 logprobs per (variant, border input), normalised over the full vocabulary: the tail mass is known, its shape is not stored, so plain temperature sampling at $T < 1$ is not recoverable
library versions	torch 2.8.0, transformers 4.57.6, trl 0.21.0, vllm 0.10.2

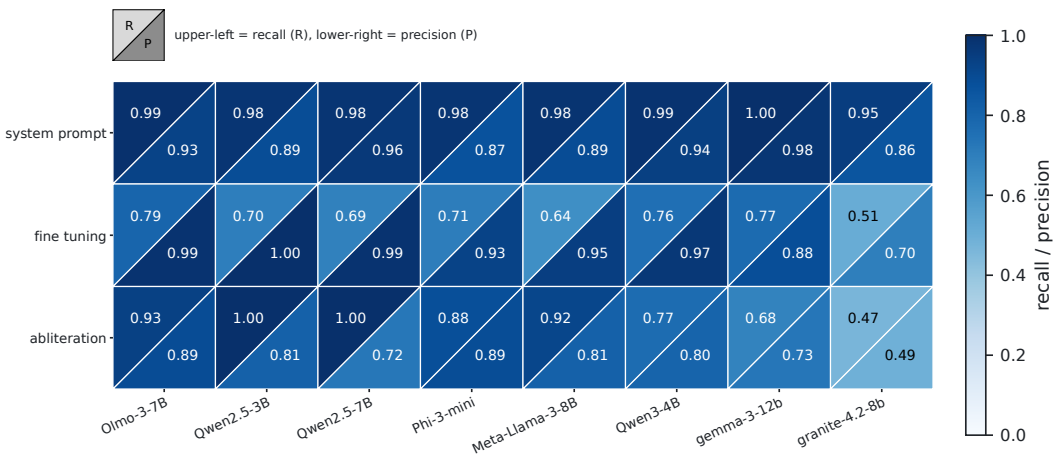

Figure 6: Recall and precision per class (rows) and LLM (columns), relative to the classification of Table 1. Each triangle is the class's recall or precision computed as in Table 1 and averaged the same way.

J Extended main results

Figure 7 gives the confusion matrix of the main configuration for every LLM (recalls in Figure 6): system prompts are almost always recognized (recall 0.95 to 1.00), and most errors are fine tunings misclassified as system prompts or as abliterations (recall 0.51 to 0.79), in proportions that depend on the LLM. Abliterations reach a recall of at least 0.88 on five LLMs, and of 0.77, 0.68 and 0.47 on Qwen3-4B, gemma-3-12b and granite-4.2-8b; on the last two, a misclassified abliteration is mostly predicted as fine tuning.

K Ablations

K.1 Query budget

Budget grid. Figure 3 runs with the 12 fused anchors, classifier and support of the main configuration, on a crossed design. At each N, 20 BI subsets are drawn and crossed with the first 7 redraw splits of Section 4.1, one per held out theme. A learner is fitted once per (N, subset, split) and rescored at every m on the R_{draw} draw replicas, which gives $7 \times 20 \times 7 = 980$ fits per LLM; a cell is the mean macro recall over subsets, splits, replicas and LLMs.

K.2 Classifier comparison

Table 9 compares two families of classifiers in the main configuration, on its BI subsets, splits and draw replicas, once per method; the logistic regression bar is the main configuration itself. Logistic regression is a stated choice (Section I.3), not selected by this comparison.

Fitted learners. Eight standard learners on the fused features themselves (the concatenation of the 12 anchor blocks, of dimension $100 \times (5 + 1) = 600$): XGBoost, random forest, gradient boosting, a multilayer perceptron, logistic regression, linear discriminant analysis, an RBF support vector machine and k nearest neighbors, with the hyperparameters of Table 7. The features are not scaled: every coordinate is a probability, on one common scale.

Rule based classifiers. Five rules, defined in Section K.3, with no fitted parameter. Three read each model through its displacement from the anchors, its features minus the anchor's, BI by BI (hence the Δ in two of the names): TV-Corr compares which BIs moved (correlation of the per BI total variations to the anchor), Δ-Cos the direction of the displacement (cosine), both by nearest landmark (1NN), and Δ-Span assigns the type whose span of training displacements best explains

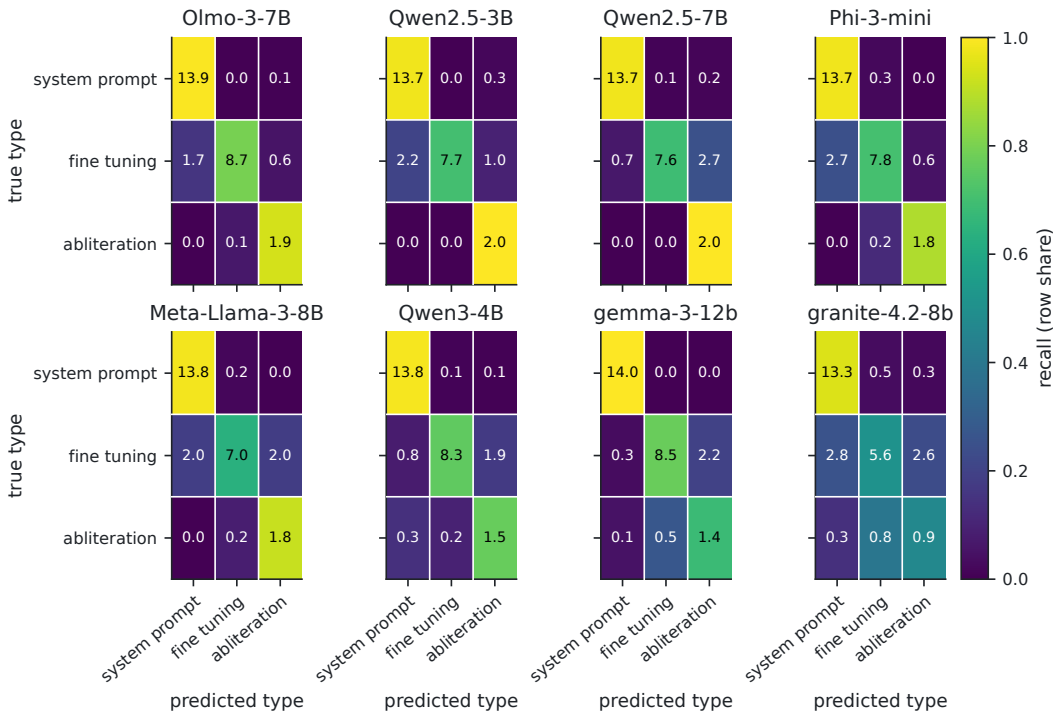

Figure 7: Confusion matrices of the main configuration (Table 1). Cells: counts averaged over the BI subsets, the splits and the R_{draw} draw replicas, printed to one decimal, so each row sums to its class's test count per split, 14 system prompts, 11 fine tunings and 2 abliterations (Section 4.1). Color: the row-normalized share (recall on diagonal cells), on a 0 to 1 scale shared across LLMs. Normalizing over columns would have shown precision on diagonal cells.

the test displacement. Two compare the features directly, by 1NN: FEAT-TV (pooled total variation) and FEAT-HELLINGER (Hellinger distance).

K.3 RULE BASED CLASSIFIERS

In a split, $T \subset \mathcal{H}$ denotes the training landmarks and $T_u = \{h \in T : u(h) = u\}$ those of type $u \in U$. A training landmark h enters through its exact features P_h, the test item X through its sampled features $\hat{P}_X$, whose sampling bias is corrected by a bootstrap where a rule needs it. The N BIs are split over the fused anchors, $Q = Q_1 \sqcup \cdots \sqcup Q_{|\mathcal{A}|}$ with $Q_j \subseteq \text{BI}(A_j)$, and $p_M(b)$, $b \in Q_j$, is read on the support of A_j (Section I.2). Four rules are nearest landmark rules (1NN) on a dissimilarity d; the fifth ranks the types by a score ρ_u:

$$\hat{u}(X) = u\Big(\arg\min_{h \in T} d(X, h)\Big) \qquad \text{or} \qquad \hat{u}(X) = \arg\min_{u \in U} \rho_u(X).$$

Anchored rules. TV-CORR, Δ-COS and Δ-SPAN read a model M through its displacement from the anchors (hence the Δ in the last two names). At $b \in Q_j$, the displacement and the total variation to the anchor are

$$\delta_M(b) = p_M(b) - p_{A_j}(b) \in \mathbb{R}^{K+1}, \qquad \tau_M(b) = \text{TV}\big(p_M(b), p_{A_j}(b)\big) = \tfrac{1}{2}\,\|\delta_M(b)\|_1.$$

Concatenated over the BIs, they give $\delta_M \in \mathbb{R}^{N(K+1)}$ and $\tau_M \in \mathbb{R}^N$, each anchor's block rescaled to a common mean squared norm so that no anchor dominates.

TV-CORR, 1NN on the shape of the TV profile: which BIs moved, relative to one another, whatever the size and the direction of the change. With the Pearson correlation taken over the N coordinates,

$$d(X, h) = 1 - \text{corr}(\tau_X, \tau_h).$$

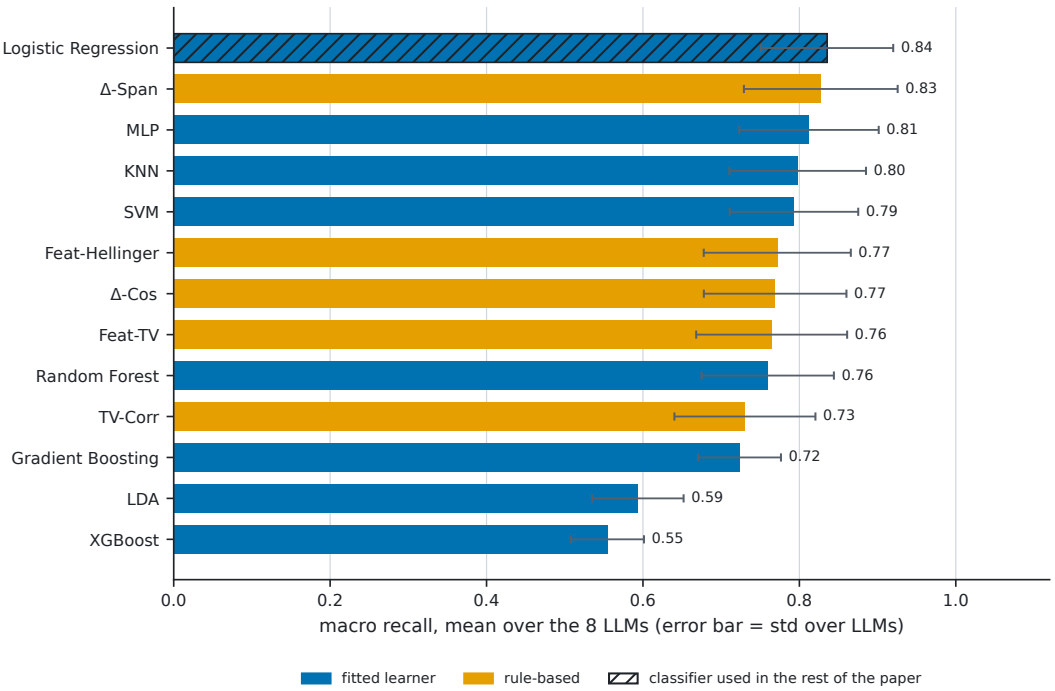

Table 9: Macro recall per classification method, relative to Table 1. See rule based method descriptions in Section K.3 and learner method parameters in Table 7.

Δ-COS, 1NN on the direction of the displacement: which tokens gain and which lose mass, whatever the size of the change,

$$d(X,h) = 1 - \frac{\langle \delta_X, \delta_h \rangle}{\|\delta_X\|_2 \, \|\delta_h\|_2}.$$

Δ-SPAN, nearest type subspace: each type is represented by the span of the 3 leading directions of its training displacements, and X gets the type whose span leaves the smallest share $\rho_u(X)$ of δ_X unexplained.

Direct rules. FEAT-TV and FEAT-HELLINGER compare the features themselves, with no reference to the anchors, on the unscaled concatenation, like the fitted learners. Both are 1NN, with

$$d_{\mathrm{TV}}(X,h) = \frac{1}{N} \sum_{b \in Q} \mathrm{TV}\big(p_X(b), p_h(b)\big) = \frac{1}{2N} \, \|P_X - P_h\|_1 \qquad \text{(FEAT-TV)},$$

$$d_{\mathrm{H}}(X,h)^2 = \frac{1}{N} \sum_{b \in Q} H^2\big(p_X(b), p_h(b)\big) = \frac{1}{2N} \, \big\|\sqrt{P_X} - \sqrt{P_h}\big\|_2^2 \qquad \text{(FEAT-HELLINGER)},$$

the square root being taken coordinatewise. The Hellinger distance is Euclidean in $\sqrt{P}$, and it is more sensitive than the total variation to a change of mass on a low probability token.

L SECOND ORDER CLASSIFICATION

This appendix details the design of the two refiners of Section 4.4: the items each one trains and tests on, and the anchors it fuses. Both are fresh fits of the logistic regression of the main configuration, with its support width, budget and draws ($K = 5$, $N = 100$, $m = 5$).

Folds. The system prompt refiner has 2 classes, oriented and neutral, over the 100 system prompt landmarks. Its folds are the 35 first order splits: each tests the 14 prompts held out by the split and trains on the 86 others, the 2 prompt anchors included, so it is scored on the same 14 test prompts as

the first order classification. The theme refiner has 7 classes over the 84 fine tuning landmarks and 2 folds, data fold 2 and data fold 3: a fold tests the 28 snapshots of its data fold, 4 per theme, and trains on the 56 snapshots of the two other data folds, 8 per theme. As in the main configuration, a refiner is fitted once per (BI subset, fold) on exact features and rescored on the R_{draw} draw replicas (Section I.3). Chance is $1/2$ and $1/7$.

Anchors of the refiners. The system prompt refiner of Section 4.4 reads the features of the 12 anchors of its first order split (Section I.2). The theme refiner tests on data folds 2 and 3 only: fold 1 holds the anchors' runs and always stays in training, so no test item shares a run with an anchor, and it fuses all 13 anchors. One rule gives both counts: a fine tuning anchor leaves the fused set exactly when a snapshot of its own training run is a test item. The theme refiner thus splits its $N = 100$ BIs as $9 \times 8 + 4 \times 7$ over 13 anchors instead of $4 \times 9 + 8 \times 8$ over 12, which is why it queries its own BIs rather than reusing those of the first order classification.