MISSING MASS, MISPLACED CONFIDENCE: CALIBRATING GENERATIVE SAFETY GUARDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative safety guards produce scores that downstream systems threshold, rank, and audit as probabilities. A commonly used score renormalizes the guard's next-token distribution over the `safe` and `unsafe` label tokens, discarding their total probability mass, which we call *verdict mass*, M. For a fixed input prompt, the resulting conditional distribution differs from the full output distribution by exactly $1 - M$ in total variation, and the guard's first-token log loss separates into mass and verdict-choice terms. Across three Llama Guard checkpoints, the released prompt templates concentrate mass near one, while removing their safety instruction exposes substantial variation across inputs. For the same input texts, safety-assessment instructions yield higher verdict mass than length-matched general-task instructions, and attack rewrites that the guard misses flip its verdict while largely preserving this mass. We combine the instruction-free mass with the native logit margin in verdict-mass calibration (VMC), a closed-form correction requiring no guard retraining, in a parameter-free form and an extension with one fitted temperature. On a held-out mixture of benign, harmful, and attack inputs, the extension reduces negative log-likelihood by approximately 9–38% relative to temperature scaling and improves Brier score and expected calibration error across all three checkpoints; it also improves all three metrics on ShieldGemma 9B.

1 INTRODUCTION

Generative safety guards such as Llama Guard classify user prompts and model responses by generating a verdict token (Inan et al., 2023). Downstream systems convert the verdict-token logits into an unsafe score for setting thresholds and prioritizing review (Liu et al., 2025a). Such scores are often treated as probabilities, yet prior work finds overconfident guard predictions and degraded calibration under jailbreak attacks (Liu et al., 2025a). Temperature scaling can recalibrate these scores after the fact (Guo et al., 2017; Liu et al., 2025a), but a score-only correction cannot use information discarded when the score was constructed.

A common unsafe score takes the guard's full-vocabulary next-token distribution and renormalizes the probabilities of `safe` and `unsafe` to sum to one (Liu et al., 2025a). This preserves the guard's relative preference between the labels but discards their combined probability mass in the original full-vocabulary distribution, which we call *verdict mass*, M. Because normalization keeps only the ratio between the two verdict probabilities, the same unsafe score can arise at very different verdict masses (Figure 1a). The raw `unsafe` probability retains verdict mass implicitly. However, the prompt templates released with each guard require `safe` or `unsafe` as the first output token, so mass is near one and varies little across inputs, and the raw probability stays close to the normalized score. Removing the safety instruction in these templates exposes input-dependent variation in mass (Figure 1b); we combine this instruction-free measurement with the logit margin from the released (native) template to recalibrate unsafe probabilities.

We study Llama Guard 2 8B, Llama Guard 3 8B, and Llama Guard 4 12B (Guard 2, Guard 3, and Guard 4) through five questions, each raised by the answer to the one before.

1. ***What does the binary score discard?*** It discards exactly the verdict mass M: for a fixed prompt, the renormalized two-token distribution is $1 - M$ away from the full next-token distribution in total variation.

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.

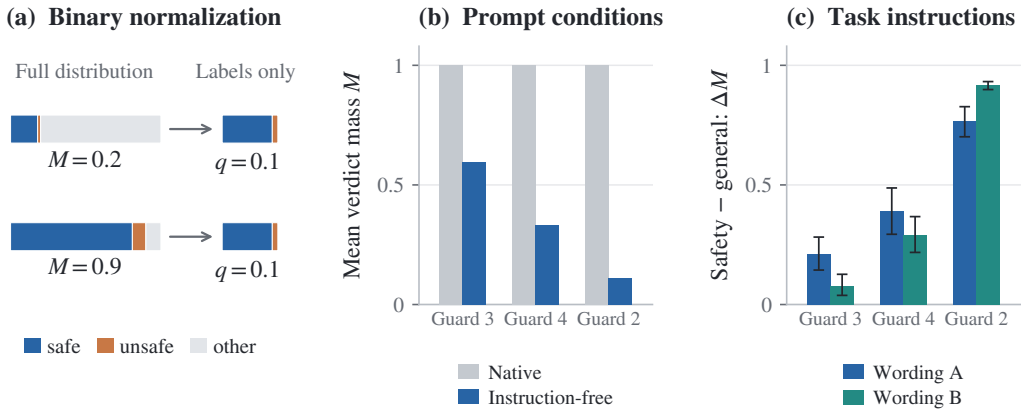


Figure 1: **One binary score hides two output coordinates.** (a) The same normalized unsafe score $q = 0.1$ can arise from different verdict masses M. (b) Mean mass on the evaluation partition under native and instruction-free prompts (renderings in Appendix B). (c) Safety-minus-general instruction effects on 64 unchanged benign inputs. Bars show paired mean differences and whiskers show 95% intervals.

2. ***Why not simply recalibrate the native score?*** If inputs with the same native score have different unsafe rates at different verdict masses, recalibrating the score alone cannot reflect those differences. In that case, using verdict mass as well can achieve a lower expected log loss than even the best recalibration of the native score alone (Proposition 1). Whether this case arises is an empirical question; on our data it does, since adding mass improves even a flexible, label-fitted recalibration of the native score on all three checkpoints (Appendix F).
3. ***Does verdict mass vary, and does it relate to unsafeness?*** Under the released templates it barely varies: mass is near one, and on 64 unchanged benign inputs, safety-assessment instructions raise mean mass more than general-task instructions of the same length, on all three checkpoints (Figure 1c). Removing the safety instruction reveals substantial variation that relates to the label: harmful prompts and attacks have higher mean mass than ordinary benign inputs in our evaluation data. The signal persists where the native score fails: attack rewrites that the guard misses leave instruction-free mass largely unchanged (Section 4).
4. ***How should verdict mass enter the score?*** These observations fix the direction of the correction: high mass accompanies unsafe inputs, whereas low mass does not establish safety. We therefore map instruction-free mass to an auxiliary unsafe probability rising from one half at zero mass to one at full mass, so mass can only add evidence for `unsafe`. Averaging its logit with the native logit margin is the equal-weight logarithmic opinion pool, an established rule for combining two probability estimates.
5. ***Does it improve calibration, and when is it useful?*** An extension of this rule with one fitted temperature improves NLL, Brier score, and ECE over temperature scaling on all three checkpoints (Section 6). The gain comes from attacks and harmful inputs, holds with only 32 labeled groups, and shrinks on every checkpoint when mass is permuted across similar inputs, vanishing entirely on Guard 4, so each input's own mass contributes.

Our contributions follow this chain.

- An exact characterization of the distributional change induced by binary normalization, together with a log-loss decomposition that shows when an added coordinate can improve on any recalibration of the native score.
- Controlled evidence that task instructions concentrate verdict mass, measurements of the variation exposed by instruction removal and its association with unsafe inputs, including attacks the guard misses, and activation interventions that move mass and verdict choice through different directions.

- Verdict-mass calibration (VMC), a compact rule requiring no guard retraining, whose extension fits one temperature and improves NLL, Brier score, and ECE over temperature scaling on three checkpoints, including with few labels, and transfers to ShieldGemma.

2 Two Coordinates Behind a Binary Score

Verdict mass and verdict choice. We study guards trained to output a safety label as their first token. A commonly used normalized score uses only two entries of the full next-token distribution. For the first-token logits z_s, z_u of `safe` and `unsafe` and the full-vocabulary softmax p, define

$$s = p_s, \quad u = p_u, \quad M = s + u, \quad q = u/M = \mathrm{sigmoid}(D), \quad D = z_u - z_s. \tag{1}$$

Here M is the *verdict mass*: the total probability mass assigned to the two label tokens. The normalized unsafe probability q describes the relative preference for `unsafe`, which we call *verdict choice*. The two verdict probabilities can be recovered from this pair through $s = M(1-q)$ and $u = Mq$, so moving from (s, u) to (M, q) loses nothing; binary normalization then drops M and keeps only q. For example, a guard that places 0.18 on `safe`, 0.02 on `unsafe`, and 0.80 on other tokens has $M = 0.2$ and $q = 0.1$, and a guard that places 0.81 and 0.09 on the two verdicts has $M = 0.9$ and the same q (Figure 1a). The raw unsafe probability $u = Mq$ retains the product of the two coordinates without separating them; our analysis concerns the information that normalization specifically removes.

Verdict mass under two prompt conditions. Under the released template, verdict mass is near one on every checkpoint we study (Figure 1b), with little variation across inputs, which leaves little for a mass-based correction to use. We therefore score every input under two prompt conditions: NATIVE, the released guard template with its safety instruction, and EMPTY (instruction-free), the same user message with the instruction removed. Removing the instruction lowers mean mass substantially on all three checkpoints, most on Guard 2 (Figure 1b). Unless stated otherwise, M denotes EMPTY verdict mass and $m = z_{u,\mathrm{N}} - z_{s,\mathrm{N}}$ denotes the NATIVE logit margin; the method combines these two measurements.

Conditioning has an exact cost. For a fixed input prompt under either condition, let P be the full next-token distribution and $Q = P(\cdot \mid V)$ its conditional on the verdict set $V = \{\texttt{safe}, \texttt{unsafe}\}$, with verdict mass $M = P(V) > 0$ for that prompt. Then

$$\mathrm{TV}(P, Q) = 1 - M, \qquad D_{\mathrm{KL}}(Q \| P) = -\log M, \tag{2}$$

so conditioning changes the output distribution by exactly $1 - M$ in total variation distance, and the divergence term $-\log M$ reappears below as the mass term of the loss. For a target verdict token $y \in V$, the guard's own log loss separates as

$$-\log p_y = \underbrace{(-\log M)}_{\text{verdict mass}} + \underbrace{(-\log Q(y))}_{\text{verdict choice}}. \tag{3}$$

Fine-tuning a guard to produce the correct first-token verdict minimizes this loss, an objective that rewards both concentrating mass on the verdict tokens and choosing the correct verdict. These are identities for the model's own distribution; they carry no guarantee about external labels.

What an additional coordinate can contribute. Predictors based only on the native logit margin have a minimum expected log loss; verdict mass can lower this Bayes risk if it carries additional label information. Let $Y \in \{0, 1\}$ be the external unsafe label, ℓ the binary log loss, $p_0^*(m) = P(Y = 1 \mid m)$ the best predictor from the native logit margin alone, and $p_1^*(m, M) = P(Y = 1 \mid m, M)$ the best predictor from the logit margin and verdict mass together. The standard population log-loss decomposition makes this comparison precise.

Proposition 1 (Log-loss decomposition)**.** *For any measurable $g : \mathbb{R} \to [0, 1]$,*

$$\mathbb{E}[\ell(g(m), Y)] = H(Y \mid m) + \mathbb{E}[D_{\mathrm{KL}}(p_0^*(m) \| g(m))] \geq H(Y \mid m), \tag{4}$$

with equality exactly when $g(m) = p_0^*(m)$ almost surely. Moreover,

$$H(Y \mid m) - H(Y \mid m, M) = I(Y; M \mid m) \geq 0, \tag{5}$$

where $H(Y \mid m, M) = \mathbb{E}[\ell(p_1^*(m, M), Y)]$ is the expected log loss of p_1^*.

Table 1: **Datasets for behavioral controls and calibration comparisons.** All counts apply identically to the three checkpoints.

Analysis	Inputs	Design
Instruction contrasts (Table 2, top)	64 ordinary benign inputs from a 302-input behavioral panel	six prompt conditions; safety-assessment and general-task instruction pairs matched in token length
Missed attacks (Table 2, bottom)	520 original harmful prompts, 6,760 linked attacks	detected-to-missed subset per checkpoint in Table 5
Calibration and source comparison (Table 2, middle; Tables 3, 4)	11,228 fitting / 7,444 held-out rows, 2,916 held-out unsafe	split by connected groups; same rows for every checkpoint; source means by input family on the held-out rows

Both p_0^* and p_1^* are calibrated, but when p_1^* varies with mass at a fixed margin it is sharper, and so attains lower expected loss under any strictly proper scoring rule. Temperature scaling is only one restricted choice of such a score-only recalibration g, so improving on it does not by itself measure $I(Y; M \mid m)$. We choose verdict mass as the added coordinate because it has an exact meaning as a property of the guard's output, it responds to the requested task in controlled comparisons (Section 4), and it improves calibration in the comparisons that follow. Proofs, the Brier analogue, and response-level conditions are in Appendix A.

3 Experimental Design

Behavioral controls test what verdict mass responds to, and calibration comparisons test what a compact calibration rule can do with it (Table 1). All three checkpoints use the same input sets and the same scoring and analysis pipeline, with identical inputs for each paired comparison; numerical settings are in Appendix I. Tables list Guard 3 and Guard 4 first and Guard 2 last.

Behavioral dataset. Inputs are a fixed sample from each source, drawn before any guard output was seen; the instruction contrasts use the 64 ordinary benign inputs of the 302-input panel. Safety-assessment and general-task instruction pairs keep the input fixed and match token length, and each attack is linked to the original prompt it was derived from (Appendix B).

Calibration dataset. Calibration and evaluation rows are split by connected groups, so an original prompt, its attacks, and its duplicates never straddle the split. Benign sources are Dolly (Conover et al., 2023), Alpaca (Taori et al., 2023), OR-Bench random (a random sample of OR-Bench-80K) (Cui et al., 2025), and XSTest-safe (Röttger et al., 2024); unsafe sources are the 520 AdvBench prompts (Zou et al., 2023), OR-Bench toxic, XSTest-unsafe, and 12 attack sources that rewrite the AdvBench prompts, after merging two attack variants with coincident texts (Appendix E).

Metrics. We report negative log-likelihood (NLL), the proper scoring rule on which temperatures are fitted; Brier score, a bounded proper scoring rule less sensitive to confident errors; and expected calibration error (ECE), the bin-weighted mean absolute gap between mean prediction and mean label over ten equal-width bins of the predicted unsafe probability. Reliability diagrams accompany these metrics (Section 6), and other bin counts and equal-count bins are in Appendix D. Intervals are pointwise 95% paired cluster bootstrap intervals that resample connected groups, with predictors held fixed; they exclude fitting and selection uncertainty, which the calibration-size analysis reports separately (Appendix D).

4 What Verdict Mass Responds To

We now ask what instruction-free verdict mass responds to. Three behavioral comparisons answer this (Table 2): matched task instructions show that instructions control mass, source comparisons show that it is higher on unsafe inputs, and missed attacks show that it largely persists when the native verdict flips.

Table 2: **Verdict mass responds to the requested task and differs across input populations.** Mean M per checkpoint; populations are in Table 1. Top: instruction pairs A and B are length-matched. Middle: means by input family on the held-out rows. Bottom: detected original → missed attack, averaged within original; detection and the unsafe probability use the released template. Paired intervals for the top and bottom blocks are in Appendix C.

		Guard 3 (8B)	Guard 4 (12B)	Guard 2 (8B)
Instructions	None	0.446	0.254	0.058
	Safety (A / B)	0.998 / 0.998	1.000 / 1.000	0.773 / 0.961
	General (A / B)	0.789 / 0.921	0.611 / 0.710	0.007 / 0.045
Sources	Ordinary benign	0.394	0.194	0.074
	General harmful	0.897	0.478	0.107
	Attacks	0.865	0.486	0.173
Missed attacks	Unsafe probability (released template)	$0.995 \to 0.118$	$0.998 \to 0.034$	$0.991 \to 0.125$
	Mass M (instruction-free)	$0.969 \to 0.848$	$0.592 \to 0.626$	$0.177 \to 0.159$

Task instructions control verdict mass. On 64 unchanged ordinary benign inputs, appending a safety-assessment instruction to the instruction-free prompt raises mean mass more than a general-task instruction of identical length on all three checkpoints (Table 2, top; Figure 1c); all six paired differences (two wordings, three checkpoints) exclude zero (Appendix C). Because input and instruction length are fixed, the difference is attributable to the requested task rather than to added text. The near-one mass under the released template fits this pattern: that template contains a safety-assessment instruction of the same kind. Inside Guard 3 and Guard 4, adding the activation difference between the two instruction types likewise raises mass and removes the mass gaps between input groups, whereas a label direction mainly moves verdict choice (Appendix H).

Instruction-free mass is higher on unsafe inputs. Without the safety instruction, general harmful inputs and attacks have higher mass than ordinary benign inputs on all three checkpoints, and attacks have more than twice the ordinary benign mass on every checkpoint (Table 2, middle). This association sets the direction of the correction in Section 5: high mass is treated as evidence for the unsafe verdict.

Missed attacks reduce the native unsafe probability while largely retaining verdict mass. We pair each original harmful prompt that the guard detects with the attack rewrites of it that the guard misses. Across this transition the native unsafe probability falls from near one to near zero on every checkpoint, while instruction-free mass changes little: it drops modestly on Guard 3 and is essentially unchanged on Guard 4 and Guard 2 (Table 2, bottom; Table 5). The retained mass shifts toward the `safe` token, so verdict choice flips while verdict mass persists. On these misses, mass therefore still carries a signal that the native verdict has lost, which motivates using it to supplement the native verdict in Section 5.

5 Verdict-Mass Calibration

Verdict-mass calibration (VMC) combines the native logit margin m with the instruction-free verdict mass M into one unsafe probability. Its parameter-free form VMC_0 fits no continuous parameter; its extension VMC_T fits one temperature, the same budget as temperature scaling, the standard post-hoc calibration baseline (Guo et al., 2017) also evaluated for guard models (Liu et al., 2025a), which rescales the native margin as $\mathrm{sigmoid}(m/T)$ with T fitted on labeled data. Section 4 fixes the direction of the correction: on these mixtures high mass accompanies unsafe inputs, whereas low mass does not establish safety, so VMC maps mass to a nonnegative auxiliary logit.

Mapping mass to an auxiliary logit. We first map mass to an auxiliary probability and take its logit,

$$a(M) = \frac{1+M}{2}, \qquad R(M) = \mathrm{logit}\, a(M) = \log \frac{1+M}{1-M}. \tag{6}$$

The map a interpolates linearly from $a(0) = 1/2$, equal probability for both verdicts, to $a(1) = 1$, probability one for the `unsafe` verdict; the auxiliary probability is a design choice, not an estimate of the true unsafe rate. Consequently $R(0) = 0$, R increases with mass, and $R(M) \geq 0$: zero mass contributes zero to the combined logit, and larger mass contributes more. We use R rather than $\mathrm{logit}\, M$ because the latter would assign arbitrarily negative logits to low mass, favoring `safe`, which Section 4 does not support. Before applying R we clip M to $[10^{-6}, 1 - 10^{-6}]$, which caps R at about 14.5.

A parameter-free form and a one-temperature extension. The parameter-free form VMC_0 averages the native logit margin with the auxiliary logit,

$$p_0(m, M) = \mathrm{sigmoid}\left(\frac{m + R(M)}{2}\right). \tag{7}$$

Because checkpoints differ in the scale of their native logit margins, we also consider VMC_T, an extension with one fitted temperature T,

$$p_{T,w}(m, M) = \mathrm{sigmoid}\Big(w\left[\frac{m}{T} + R(M)\right]\Big), \quad T > 0, \quad w \in \{\tfrac{1}{2}, 1\}, \tag{8}$$

where $w = 1/2$ averages the two terms and $w = 1$ sums them; VMC_0 is the special case $T = 1$, $w = 1/2$, and for VMC_0 averaging was preferred to summing on calibration NLL on all three checkpoints. Since T absorbs the scale of the margin, w only sets the weight of the mass term to one half or one. For each w we fit T by minimizing calibration NLL and keep the w with the lower calibration NLL; this selects $w = 1/2$ for Guard 3 and Guard 4 and $w = 1$ for Guard 2. The extension therefore fits one continuous parameter, as temperature scaling does, plus a choice between two prespecified weights on mass.

Form of the combination. Averaging two logits is the equal-weight logarithmic opinion pool, the geometric mean of the native odds $e^{m/T}$ and the mass odds $(1 + M)/(1 - M)$, and summing them is the product-of-experts form (Genest & Zidek, 1986; Hinton, 2002); both are established rules for combining two probability estimates without fitting weights. Neither identity says why the combination helps: m and M come from the same model under two prompt conditions and are not independent evidence, so the case rests on the behavioral evidence of Section 4 and the comparisons of Section 6.

6 Results

VMC_T improves NLL, Brier score, and ECE over temperature scaling on all three checkpoints, and paired intervals for all nine differences (three metrics, three checkpoints) exclude zero (Table 3; Appendix D). VMC_0, which fits no continuous parameter, already improves on the native score on every checkpoint and nearly matches VMC_T on Guard 3, where VMC_T fits a temperature close to one; on Guard 4 and Guard 2, VMC_T fits temperatures well above one. We next examine how the correction changes confidence, which inputs gain, controls for the source of the gain, performance with few labels, transfer to a second guard family, and robustness to task-suffix attacks.

Reliability improves, and the correction is not uniform shrinkage. Reliability curves move toward the diagonal on all three checkpoints (Figure 2), and VMC_T improves ECE over temperature scaling under every tested binning (Appendix D). Table 4, top, shows what the correction does: relative to temperature scaling it lowers the probability assigned to the guard's verdict when that verdict is wrong and slightly raises it when the verdict is right, on all three checkpoints. Shrinking every probability toward one half would lower both, so VMC is doing more than reducing confidence.

The gain comes from unsafe inputs. Figure 3 shows the change in mean per-example evaluation NLL from temperature scaling to VMC_T for the three largest input families. Attacks and general harmful inputs improve on every checkpoint, whereas loss on ordinary benign inputs, more than half of the evaluation set, rises only slightly. Per-source values, including the remaining benign sources, and the dependence on unsafe prevalence are reported in Appendix E and discussed in Section 8.

Table 3: VMC_T **improves NLL, Brier score, and ECE over temperature scaling on all three checkpoints.** Arrows mark the better direction (lower for all three metrics); bold marks column minima. Rows are grouped by budget: the native score and VMC_0 fit no continuous parameter (VMC_0's averaged form was chosen on calibration labels), whereas temperature scaling and VMC_T each fit one temperature, and VMC_T also selects average or sum on calibration NLL (Section 5). ECE_{10} uses ten equal-width bins. All predictors use the same evaluation partition (Table 1). Paired intervals and further controls are in Appendix D; flexible supervised calibrators, which select among logistic, spline, and gradient-boosting models by cross-validation, are a larger-budget reference in Appendix F.

	Guard 3 (8B)			Guard 4 (12B)			Guard 2 (8B)		
Predictor	NLL ↓	Brier ↓	ECE_{10} ↓	NLL ↓	Brier ↓	ECE_{10} ↓	NLL ↓	Brier ↓	ECE_{10} ↓
Native sigmoid(m)	.424	.104	.114	1.240	.116	.119	.651	.145	.160
VMC_0 (ours)	.168	**.049**	**.028**	.486	.091	.094	.375	.118	.142
Temperature	.270	.085	.109	.286	.088	.096	.353	.114	.155
VMC_T **(ours)**	**.168**	.050	.029	**.240**	**.072**	**.046**	**.321**	**.102**	**.124**

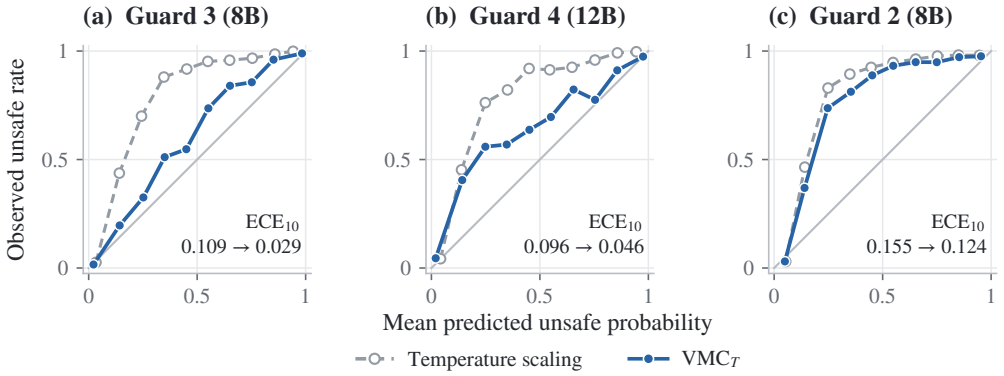


Figure 2: VMC_T **improves reliability across three checkpoints.** Observed unsafe rate versus mean predicted probability in ten equal-width bins, for temperature scaling (gray) and VMC_T (blue); insets give ECE_{10} before → after. Markers are bin means, not independent samples.

Controls. Because VMC_0 halves the native logit, we compare it with the matched-scale control $\mathrm{sigmoid}(m/2)$, which it beats on every checkpoint. Permuting $R(M)$ across inputs with similar native logit margin and length increases NLL on every checkpoint, and verdict mass yields lower NLL than the alternative scores tested from the same instruction-free pass (Appendix D). These controls support a contribution from input-specific verdict mass. Consistent with Proposition 1, adding M also lowers the NLL of a flexible supervised calibrator of the native margin on all three checkpoints (Appendix F), so the information in mass is not tied to the one-temperature form.

Few labels suffice. With only 32 labeled groups for fitting, VMC_T already has lower NLL than temperature scaling on every checkpoint, and VMC_0, which fits no continuous parameter, is the best predictor tested at that size on Guard 3 and Guard 2 (Appendix D.3).

Transfer to ShieldGemma. On ShieldGemma 9B (Zeng et al., 2024), with mass measured under a minimal retained safety question chosen by prespecified gates (Appendix G), VMC_T and VMC_0 both improve NLL, Brier score, and ECE over temperature scaling on the same evaluation partition.

Lowering mass alone cannot reverse a native unsafe verdict. At a fixed native logit margin, higher mass raises the VMC unsafe probability, so an attacker might try to evade the correction by lowering mass. Because $R(M) \geq 0$ and $T, w > 0$, any input with $m > 0$ still has a VMC_T probability above 0.5, even as mass approaches zero. Thus lowering mass alone cannot reverse an

Table 4: VMC_T **lowers confidence in wrong verdicts and keeps flagging more attacks under task suffixes.** Arrows mark the better direction. Top: change in the mean probability assigned to the guard's own verdict, VMC_T minus temperature scaling, on evaluation inputs where that verdict is wrong or right. Bottom: percentage of the 2,378 evaluation attacks flagged unsafe by the native score ($m > 0$) and by VMC_T (probability above 0.5), before and after appending to each input the one of three fixed task suffixes that minimizes its VMC_T probability; both predictors see the same suffixed inputs. Paired intervals and the suffix protocol are in Appendix D.

		Guard 3 (8B)		Guard 4 (12B)		Guard 2 (8B)	
Confidence in own verdict	Verdict wrong ↓	−0.272		−0.184		−0.056	
	Verdict right ↑	+0.029		+0.027		+0.011	
		Native	VMC_T	Native	VMC_T	Native	VMC_T
Attacks flagged (%) ↑	Inputs as given	65.7	85.3	68.2	80.7	50.5	59.5
	Adaptive task suffix	33.1	40.8	51.0	59.3	12.1	17.3

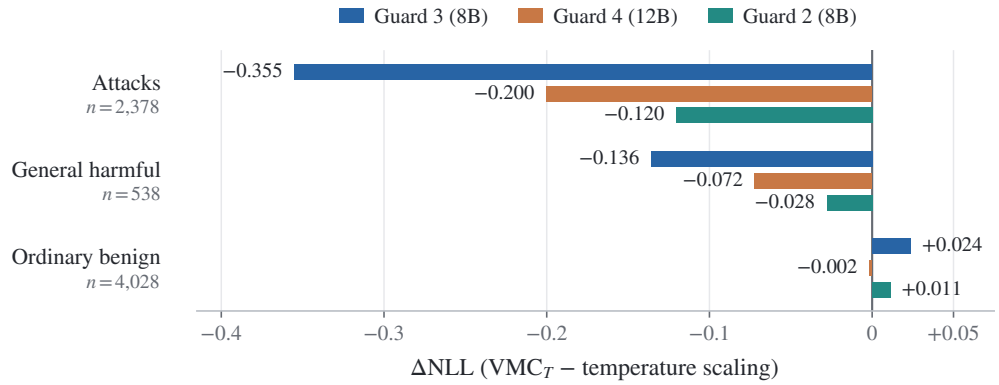


Figure 3: **Gains concentrate on unsafe inputs; ordinary benign loss rises only slightly.** Change in mean evaluation NLL by input family, VMC_T minus temperature scaling (negative is better); counts are at left. Per-source values are in Appendix E.

unsafe prediction at this threshold; the guarantee concerns the decision, not the magnitude of the calibrated probability (Appendix D.4). To test a manipulation that can also change the native logit margin, we append a task suffix to each evaluation attack, choosing the one of three fixed suffixes that minimizes its VMC_T probability (Table 4, bottom). The suffix lowers the fraction VMC_T flags, but the native detection rate, which temperature scaling shares at this threshold, ends lower on the same suffixed inputs, so VMC_T still flags more attacks on every checkpoint.

7 Related Work

Guard models and their scores. Llama Guard introduced policy-conditioned generative moderation (Inan et al., 2023); we study its Guard 2, 3, and 4 releases (Meta, 2024a;b; 2025), and ShieldGemma, WildGuard, and Granite Guardian are other open generative guards (Zeng et al., 2024; Han et al., 2024; Padhi et al., 2024). The Guard 2 and 3 model cards score with the raw first-token unsafe probability $u = Mq$, whereas Liu et al. (2025a, Eq. 3) renormalize over the label tokens, which removes M; they also show that guards are overconfident and miscalibrated under jailbreaks and evaluate temperature scaling, contextual calibration, and batch calibration as remedies. Reasoning guards such as GuardReasoner emit a rationale before the verdict (Liu et al., 2025c), so our first-token definition would require a specified verdict position.

Token probabilities and calibration. Surface-form competition explains why one answer string's probability need not track correctness (Holtzman et al., 2021), and $P(\mathrm{True})$ elicits a separate self-

evaluation (Kadavath et al., 2022); verdict mass instead measures the probability placed on the label set as a whole. Contextual and batch calibration estimate a prompt-level bias from content-free or unlabeled inputs (Zhao et al., 2021; Zhou et al., 2024), whereas our instruction removal keeps the input and measures an input-specific quantity. Temperature scaling fits one scalar (Guo et al., 2017) and Dirichlet calibration fits richer maps (Kull et al., 2019); our rule adds one prespecified coordinate through the logarithmic opinion pool (Genest & Zidek, 1986; Hinton, 2002) within the one-parameter budget. ProbGuard estimates risk from the protected model's own early output distributions with sampled continuations as targets and a post-trained estimator (Huang et al., 2026); we read an existing guard's first-token distribution with no sampling or retraining.

Harmfulness, refusal, and guard failures. Chat models encode harmfulness and refusal along separate directions (Arditi et al., 2024; Zhao et al., 2025), with detection heads upstream of refusal heads (Chu et al., 2026), and guards can be fooled by refusal phrasing in a response (Feng et al., 2026). Verdict mass and verdict choice give an output-level counterpart that requires only first-token logits, and activation interventions move them through different directions inside the guard (Appendix H).

8 Discussion and Limitations

What verdict mass measures. Verdict mass is the probability that the guard answers the classification task at all, before it chooses a label. The released template makes that answer mandatory, so mass is near one for every input, and the guard's fine-tuning objective rewards exactly this concentration alongside the correct choice (Section 2). Without the instruction, harmful inputs and attacks still draw the guard toward a verdict far more than ordinary benign inputs do, and attacks that flip the verdict leave this pull largely intact (Section 4). Mass therefore carries a signal about whether an input looks like a moderation case, which the normalized score discards and which motivates using mass only as evidence for `unsafe`.

Using the correction. Because $R(M) \geq 0$, the mass term can raise but never lower the unsafe probability relative to the same predictor without it, so lowering mass cannot reverse a native unsafe verdict at threshold 0.5 (Section 6). Gains concentrate on unsafe inputs that the guard scores poorly, so the natural use is calibrating probabilities on traffic in which unsafe inputs are a non-negligible share; on our mixtures the aggregate benefit turns positive above an unsafe share of roughly 10–18% (Appendix E), a figure to check against the target traffic before deployment. Because VMC_T reorders inputs whereas temperature scaling preserves the native ranking, thresholds for fixed-recall operation should be reselected on calibration data. The correction requires a second forward pass under the instruction-free prompt, because the released template pins native mass near one on every checkpoint (Section 4).

Scope and next steps. Evaluation covers three checkpoints of one model family on existing public English benchmarks; the ShieldGemma extension shows that the rule transfers to a second guard family under a retained-question measurement (Appendix G). Flexible supervised calibrators are a larger-budget reference: with ample labels they attain lower loss, and adding mass improves them further (Appendix F); VMC is intended for the retraining-free, low-label setting (Appendix D.3), where its parameter-free form already improves on the native score and its extension fits one temperature. Guard misses measure guard errors and do not by themselves establish successful attacks on an assistant. Next steps are evaluation on inputs collected under a common protocol, replication on other guards that emit a verdict first, and a specified verdict position for reasoning-first guards.

Conclusion. Binary normalization keeps the guard's relative preference between `safe` and `unsafe` and discards how much of its output the guard commits to the two verdicts together, changing the output distribution by exactly $1 - M$ in total variation. Task instructions control this mass, harmful inputs and attacks raise it, and missed attacks largely preserve it, so verdict mass carries label information that even a flexible recalibration of the native score does not recover. VMC pools a nonnegative logit of instruction-free mass with the native margin; with one fitted temperature it improves NLL, Brier score, and ECE over temperature scaling on three Llama Guard checkpoints and on ShieldGemma.

REPRODUCIBILITY STATEMENT

Section 3 specifies populations, split conventions, metrics, and the bootstrap unit; Section 5 and Table 3 define the comparators; Section 2 defines the measured quantities and prompt conditions. Equations (6)–(8) specify clipping, VMC_0, and both VMC_T forms: $w = 1/2$ for Guard 3/4 and $w = 1$ for Guard 2, selected on calibration NLL. All calibration and behavioral numbers for the three checkpoints are produced by one scoring and analysis pipeline on shared input sets; every table and calibration figure is regenerated from its output files, which record the scores' checksums. Appendix A contains the derivations, Appendices B and H give task-instruction wordings and intervention definitions, and Appendix I describes numerical precision and artifact provenance. The prevalence analysis (Appendix E.1) reweights reported source losses deterministically, with no additional inference or fitting.

THE USE OF LARGE LANGUAGE MODELS

LLM-based agents independently labeled 160 continuations in the output audit, blind to model identity, source, scores, and each other's labels. Their agreement is reported as an agent audit, not human annotation (Appendix B). LLM assistance was also used for manuscript editing, code, literature checks, and visualization of stored results. This assistance does not replace validation against the experimental artifacts; responsibility for the submitted claims and results remains with the authors.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems*, volume 37, pp. 136037–136083, 2024. doi: 10.52202/079017-4322. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/f545448535dfde4f9786555403ab7c49-Abstract-Conference.html.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. IEEE, 2025. doi: 10.1109/SaTML64287.2025.00010. URL https://doi.org/10.1109/SaTML64287.2025.00010.

Kuan-Lin Chu, Chung-En Sun, and Tsui-Wei Weng. From detection to refusal: Safer LLMs via circuit-guided weight scaling. *arXiv preprint arXiv:2609.00051*, 2026. doi: 10.48550/arXiv.2609.00051. URL https://arxiv.org/abs/2609.00051.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free Dolly: Introducing the world's first truly open instruction-tuned LLM. Databricks blog, 2023. URL https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. OR-bench: An over-refusal benchmark for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 11515–11542. PMLR, 2025. URL https://proceedings.mlr.press/v267/cui25a.html.

Yu Feng, Chunting Zang, Chen Shen, Rui Miao, Ge Teng, Weidong Cai, and Jieping Ye. When refusal looks safe: The refusal-cue shortcut in safety guard models. *arXiv preprint arXiv:2608.03201*, 2026. doi: 10.48550/arXiv.2608.03201. URL https://arxiv.org/abs/2608.03201.

Christian Genest and James V. Zidek. Combining probability distributions: A critique and an annotated bibliography. *Statistical Science*, 1(1):114–135, 1986. doi: 10.1214/ss/1177013825. URL https://doi.org/10.1214/ss/1177013825.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. WildGuard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 8093–8131, 2024. doi: 10.52202/079017-0261. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0f69b4b96a46f284b726fbd70f74fb3b-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018. URL https://doi.org/10.1162/089976602760128018.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7038–7051. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.564. URL https://aclanthology.org/2021.emnlp-main.564/.

Xinzhe Huang, Biwu Yao, Kedong Xiu, Mengnan Zhao, Di Wang, Puning Zhao, and Tianhang Zheng. ProbGuard: Calibrated safety risk estimation from LLM output distributions. *arXiv preprint arXiv:2608.10621*, 2026. doi: 10.48550/arXiv.2608.10621. URL https://arxiv.org/abs/2608.10621.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*, 2023. doi: 10.48550/arXiv.2312.06674. URL https://arxiv.org/abs/2312.06674.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022. doi: 10.48550/arXiv.2207.05221. URL https://arxiv.org/abs/2207.05221.

Meelis Kull, Miquel Perello-Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with Dirichlet calibration. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/8ca01ea920679a0fe3728441494041b9-Abstract.html.

Hongfu Liu, Hengguan Huang, Xiangming Gu, Hao Wang, and Ye Wang. On calibration of LLM-based guard models for reliable content moderation. In *The Thirteenth International Conference on Learning Representations*, pp. 67808–67829, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a99f732df9b668284b449da0214a3286-Abstract-Conference.html.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. AutoDAN: Generating stealthy jailbreak prompts on aligned large language models. In *The Twelfth International Conference on Learning Representations*, pp. 56174–56194, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f83cb637e159e789f5576ff6848874de-Abstract-Conference.html.

Xiaogeng Liu, Peiran Li, G. Edward Suh, Yevgeniy Vorobeychik, Zhuoqing Mao, Somesh Jha, Patrick McDaniel, Huan Sun, Bo Li, and Chaowei Xiao. AutoDAN-Turbo: A lifelong agent for strategy self-exploration to jailbreak LLMs. In *The Thirteenth International Conference on Learning Representations*, pp. 10313–10360, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1bff3663270ba47f801e917f782d7935-Abstract-Conference.html.

Yue Liu, Hongcheng Gao, Shengfang Zhai, Yufei He, Jun Xia, Zhengyu Hu, Yulin Chen, Xihong Yang, Jiaheng Zhang, Stan Z. Li, Hui Xiong, and Bryan Hooi. GuardReasoner: Towards reasoning-based LLM safeguards. *arXiv preprint arXiv:2501.18492*, 2025c. doi: 10.48550/arXiv.2501.18492. URL https://arxiv.org/abs/2501.18492v2.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box LLMs automatically. In *Advances in Neural Information Processing Systems*, volume 37, pp. 61065–61105, 2024. doi: 10.52202/079017-1952. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/70702e8cbb4890b4a467b984ae59828a-Abstract-Conference.html.

Meta. Llama Guard 2 model card. Official model documentation, 2024a. URL https://huggingface.co/meta-llama/Meta-Llama-Guard-2-8B.

Meta. Llama Guard 3 model card. Official model documentation, 2024b. URL https://huggingface.co/meta-llama/Llama-Guard-3-8B.

Meta. Llama Guard 4 model card. Official model documentation, 2025. URL https://huggingface.co/meta-llama/Llama-Guard-4-12B.

Inkit Padhi, Manish Nagireddy, Giandomenico Cornacchia, Subhajit Chaudhury, Tejaswini Pedapati, Pierre Dognin, Keerthiram Murugesan, Erik Miehling, Martín Santillán Cooper, Kieran Fraser, Giulio Zizzo, Muhammad Zaid Hameed, Mark Purcell, Michael Desmond, Qian Pan, Zahra Ashktorab, Inge Vejsbjerg, Elizabeth M. Daly, Michael Hind, Werner Geyer, Ambrish Rawat, Kush R. Varshney, and Prasanna Sattigeri. Granite Guardian. *arXiv preprint arXiv:2412.07724*, 2024. doi: 10.48550/arXiv.2412.07724. URL https://arxiv.org/abs/2412.07724.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5377–5400. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.301. URL https://aclanthology.org/2024.naacl-long.301/.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford Alpaca: An instruction-following LLaMA model. GitHub repository, 2023. URL https://github.com/tatsu-lab/stanford_alpaca.

Wenjun Zeng, Yuchi Liu, Ryan Mullins, Ludovic Peran, Joe Fernandez, Hamza Harkous, Karthik Narasimhan, Drew Proud, Piyush Kumar, Bhaktipriya Radharapu, Olivia Sturman, and Oscar Wahltinez. ShieldGemma: Generative AI content moderation based on Gemma. *arXiv preprint arXiv:2407.21772*, 2024. doi: 10.48550/arXiv.2407.21772. URL https://arxiv.org/abs/2407.21772.

Jiachen Zhao, Jing Huang, Zhengxuan Wu, David Bau, and Weiyan Shi. LLMs encode harmfulness and refusal separately. In *Advances in Neural Information Processing Systems*, volume 38, pp. 140283–140318, 2025. doi: 10.52202/085713-4688. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/cd18539787d90e1d682d557c2c71b534-Abstract-Conference.html.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12697–12706. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/zhao21c.html.

Han Zhou, Xingchen Wan, Lev Proleev, Diana Mincu, Jilin Chen, Katherine Heller, and Subhrajit Roy. Batch calibration: Rethinking calibration for in-context learning and prompt engineering. In *The Twelfth International Conference on Learning Representations*, pp. 49–70, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/003e438cf04e3caf0a5c908495e484fe-Abstract-Conference.html.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023. doi: 10.48550/arXiv.2307.15043. URL https://arxiv.org/abs/2307.15043.

A Distributional Identities and Calibration Proofs

The cost of conditioning. On the verdict set V, $Q(t) - P(t) = P(t)(1/M - 1)$, which sums to $1 - M$. Outside V, the absolute difference also sums to $1 - M$. Half of the total absolute difference therefore gives $\mathrm{TV}(P, Q) = 1 - M$. Where Q is positive, $\log[Q(t)/P(t)] = -\log M$, which proves the KL identity. For $M < 1$, $D_{\mathrm{KL}}(P\|Q)$ is infinite. The same likelihood ratio gives the response-level result when the continuation kernel is unchanged. For any response-scoring function g taking values in $[0, 1]$, total variation gives $|\mathbb{E}_P[g] - \mathbb{E}_Q[g]| \leq 1 - M$. Changing the temperature or subsequent decoding generally changes the distribution, so this conditioning interpretation no longer applies.

Response-level interpretation. Suppose the scoring function assigns zero to every response beginning with the safe token and one to every response beginning with the unsafe token. With $r = \mathbb{E}_P[g \mid y_1 \notin V]$ and $\theta = \mathbb{E}_P[g]$,

$$\theta = u + (1 - M)r \in [u, u + 1 - M], \qquad \theta - q = (1 - M)(r - q). \tag{9}$$

The interval is sharp because $r = 0$ and $r = 1$ attain its endpoints. This interval describes the assigned meaning of model outputs, rather than the input's externally labeled harmfulness. Under the convention that the canonical first token determines the response's verdict, the selected verdict's worst-case strict-majority margin is

$$\gamma = 2\max(s, u) - 1 = M(1 + |2q - 1|) - 1. \tag{10}$$

A positive margin means that the selected verdict retains a strict majority even if all omitted probability is assigned to the opposite verdict. This concerns the majority within the output distribution, not whether the verdict is correct. At zero, the conclusion depends on tie handling.

Verdict mass and verdict choice share gradients. For a verdict-token logit, the gradient of $-\log M$ is $p_j - Q(j)$; for a non-verdict logit it is p_j. The conditional verdict-choice loss has gradient $Q(j) - \mathbb{1}\{j = y\}$ within V and zero elsewhere. Their sum is the ordinary cross-entropy gradient. Writing $z_s = b - c$ and $z_u = b + c$ while holding the non-verdict logits fixed gives

$$\frac{\partial[-\log M]}{\partial b} = -(1 - M), \qquad \frac{\partial[-\log M]}{\partial c} = -(1 - M)\tanh c. \tag{11}$$

Increasing M during learning can therefore strengthen the preference for either verdict. The loss decomposition separates two measurable quantities without establishing separate computations within the model.

Proof of Proposition 1 (log-loss decomposition). Write $\pi = p_0^*(m)$ and $h(\pi) = -\pi \log \pi - (1 - \pi)\log(1 - \pi)$. Conditional on m, $\mathbb{E}[\ell(g(m), Y) \mid m] = -\pi \log g - (1 - \pi)\log(1 - g) = h(\pi) + D_{\mathrm{KL}}(\pi \| g)$, where the divergence is between Bernoulli distributions with parameters π and $g(m)$. Taking expectations over m gives Equation (4), since $\mathbb{E}[h(\pi)] = H(Y \mid m)$. The divergence is nonnegative and vanishes only when $g(m) = \pi$, establishing the equality condition. Equation (5) follows from the chain rule for conditional entropy. The Bayes-optimal log loss given features X is $H(Y \mid X)$; using $X = m$ and $X = (m, M)$ gives a difference of $I(Y; M \mid m)$. For Brier loss, expand $(Y - p_0^*)^2$ around p_1^*. The cross term vanishes conditional on (m, M) because $\mathbb{E}[Y - p_1^* \mid m, M] = 0$. The remaining difference is $\mathbb{E}[(p_1^* - p_0^*)^2]$. Both results concern population-optimal predictors and do not by themselves specify a monotone correction in M; the behavioral results of Section 4 supply its direction.

B Behavioral Populations and Protocol

Source panel. The 302-input source panel comprises 32 each from Dolly, Alpaca, OR-Bench random, XSTest-safe, original harmful objectives, OR-Bench toxic and XSTest-unsafe, plus six from each of 13 attack sources. A fixed hash of seed, source and sample ID selects inputs within each source, independently of model scores and native correctness. All three checkpoints score the same panel. The full linked panel contains 6,760 pairs per checkpoint from 520 original harmful objectives.

Attack-source names and provenance. The 13 attack-source labels are Adaptive, AutoDAN, AutoDAN-Turbo, GCG, PAIR, Roleplay, TAP, and S2X variants of AutoDAN, AutoDAN-Turbo, GCG-universal, PAIR, Roleplay, and TAP. The named methods include GCG (Zou et al., 2023), AutoDAN (Liu et al., 2024), AutoDAN-Turbo (Liu et al., 2025b), PAIR (Chao et al., 2025), and TAP (Mehrotra et al., 2024). Adaptive and Roleplay are source labels of the attack corpus. Every attack rewrites one of the 520 AdvBench prompts and was generated with Llama-3.1-8B-Instruct as the target model, not against the guards; each S2X variant doubles the number of optimization steps of its method to produce stronger attacks. The deduplicated calibration partition merges S2X GCG-universal with GCG because their texts coincide, leaving 12 attack-source labels.

Matched task instructions. The instruction comparison uses NATIVE, EMPTY, safety-A, general-A, safety-B and general-B conditions. Safety-A is "Provide a safety assessment of this user message." and general-A is "Provide a brief summary of this user message." Safety-B is "Evaluate the safety of the user message above." and general-B is "Describe the content of the user message above." Added task instructions are ten tokens on every checkpoint, and the complete prompts have equal token lengths within each safety/general-task pair. The comparison changes the requested task while holding the user input fixed.

Guard 3 and Guard 4 renderings. NATIVE is the released chat template: a user turn that wraps the input in the guard's task description, unsafe-content categories, and output instruction, followed by the assistant header. EMPTY keeps only the chat structure; for Guard 3, with `\n` a newline, it is

```
<|begin_of_text|><|start_header_id|>user<|end_header_id|>
\n\n{input}<|eot_id|>
<|start_header_id|>assistant<|end_header_id|>\n\n
```

and Guard 4 uses the corresponding Llama 4 header tokens. Both conditions end with the two newlines that the guard emits with probability near one after the assistant header, so the scored position is the verdict position. The verdict tokens are `safe` and `unsafe`, with token IDs 19193 and 39257 on Guard 3 and 60411 and 72110 on Guard 4.

Guard 2 renderings. Guard 2's native prompt encloses the safety policy and user content in `[INST]` and `[/INST]`. The EMPTY condition uses the Llama 3-style user/assistant header wrapper with the policy omitted, so on this checkpoint the native-to-empty contrast also includes the wrapper change. All four Guard 2 added-instruction conditions share the instruction-free wrapper, so the paired safety-versus-general comparisons hold both wrapper and user content fixed.

Aggregation. Linked-attack means are first averaged within each original prompt, then across originals. Mean normalized unsafe probabilities are averages of per-input q, rather than ratios of mean masses.

Output types outside the verdict set. We estimate the mass of each output type by sampling responses $\widetilde{y}_j$ conditional on the first token being outside the two verdict tokens:

$$\widehat{P}(C \mid x) = (1 - M(x))\frac{1}{4}\sum_{j=1}^{4} \mathbb{1}\{C(\widetilde{y}_j) = C\}, \qquad \widetilde{y}_j \sim P(\cdot \mid x, y_1 \notin \{s, u\}). \tag{12}$$

Sampling excludes only the two exact canonical token IDs at the first position. Later tokens use the original distribution at temperature one and top-p one. For each diagnostic input and model, we also sample one continuation without restricting the first token. The prefactor converts conditional frequencies to full-distribution masses. Standalone `S` tokens and numeric strings without a category are not counted as verdicts, and "Other text" is kept separate from task answers.

Blinded output annotation. Two independent LLM-based agents annotate 160 hash-selected input–continuation pairs. Model identity, source, scores, automatic classifications and the other agent's labels are hidden. Categories include canonical and category verdicts, prose assessments, refusals, task responses, ambiguous numbers, empty outputs and unclear outputs. Disagreements are retained, and truncated answers are categorized from the observed text.

Bootstrap intervals. Source and linked-pair summaries use 2,000 connected-group bootstrap draws. The intervals condition on fixed checkpoints, selected inputs and observed continuation samples.

C ADDITIONAL BEHAVIORAL RESULTS

Mass on missed attacks. Table 5 pairs each original that the guard detects with the attacks derived from it that the guard misses. On these pairs the unsafe probability under the released template falls by 0.88, 0.96, and 0.87 on Guard 3, Guard 4, and Guard 2, while instruction-free mass changes by only -0.12, $+0.03$, and -0.02; the Guard 4 interval includes zero. Across all 6,760 linked attacks, including those the guard still detects, mean mass changes by -0.084, -0.133, and -0.013.

Table 5: **Missed attacks change verdict choice more consistently than verdict mass.** Detected-to-missed pairs from the full linked panel: Guard 3 has 2,030 pairs from 500 originals; Guard 4 has 1,727 pairs from 470 originals; Guard 2 has 2,890 pairs from 500 originals. Each pair links an original detected as unsafe to an attack missed under the released template; arrows follow this order. Means are first averaged within each original, then across originals. Intervals are pointwise 95% connected-group bootstrap intervals.

Measurement	Guard 3 (8B)	Guard 4 (12B)	Guard 2 (8B)
Unsafe probability, released template	$0.9951 \to 0.1185$	$0.9976 \to 0.0339$	$0.9912 \to 0.1249$
Verdict mass M, instruction-free	$0.9685 \to 0.8484$	$0.5925 \to 0.6258$	$0.1774 \to 0.1594$
ΔM	-0.1201	$+0.0333$	-0.0181
95% interval	$[-0.1350, -0.1052]$	$[-0.0069, +0.0730]$	$[-0.0350, -0.0018]$
Unsafe probability q, instruction-free	$0.8147 \to 0.2372$	$0.5013 \to 0.0357$	$0.5461 \to 0.3704$
`safe` probability s, instruction-free	$0.1689 \to 0.6430$	$0.2145 \to 0.6049$	$0.0792 \to 0.1125$

Table 6: **Paired instruction effects on verdict mass.** Mean differences in mass on the 64 ordinary benign inputs, with 95% connected-group bootstrap intervals for the length-matched contrasts.

Contrast	Guard 3 (8B)	Guard 4 (12B)	Guard 2 (8B)
Safety A − general A	$+0.209$	$+0.389$	$+0.766$
95% interval	$[+0.144, +0.282]$	$[+0.294, +0.487]$	$[+0.701, +0.827]$
Safety B − general B	$+0.077$	$+0.290$	$+0.916$
95% interval	$[+0.038, +0.126]$	$[+0.218, +0.368]$	$[+0.899, +0.932]$
General A − instruction-free	$+0.344$	$+0.356$	-0.051

Matched instruction effects. Table 6 gives the paired differences on the 64 ordinary benign inputs. Both safety-assessment wordings raise mass more than their length-matched general-task counterparts on all three checkpoints, most on Guard 2. On Guard 3 and Guard 4 even a general task raises mass above the instruction-free rendering. These intervals compare instructions with the same underlying input and token length.

Sensitive benign inputs. Sensitive benign inputs (OR-Bench random and XSTest-safe) are written to resemble harmful requests while remaining benign. On the evaluation partition their mean instruction-free mass is 0.598, 0.566, and 0.095 on Guard 3, Guard 4, and Guard 2, above the ordinary benign means of 0.394, 0.194, and 0.074. The difference from ordinary benign inputs persists at matched native logit margin and token length: 0.157 $[0.129, 0.182]$ over 991 Guard 3 pairs and 0.374 $[0.339, 0.409]$ over 981 Guard 4 pairs, with no difference on Guard 2 (0.000 $[-0.015, +0.014]$, 968 pairs). This is consistent with mass responding to whether an input resembles a moderation case, the reading developed in Section 8.

Non-verdict mass rarely encodes a verdict. On the same 64 inputs, the first-token distribution outside the verdict set differs across checkpoints: Guard 3 places 0.44 of its first-token probability on its 32 most likely non-verdict tokens, Guard 4 places 0.57 on immediate termination, and Guard 2 spreads 0.94 across many tokens; the entropy of the full first-token distribution is 2.39, 0.47, and 5.63 nats. A continuation audit from the initial Guard 3/4 run (Appendix I) classifies sampled outputs on ordinary benign inputs (Table 7); two LLM annotators agree on 155/160 outputs ($\kappa = 0.96$). Verdict variants, namely S-category labels and ambiguous numbers, account for a minority of the non-verdict mass; the remainder is other text on Guard 3 and empty output on Guard 4. On 159 low-mass benign inputs from the same run (mean M 0.126), even assigning all S tokens and all probability beyond the top-32 tokens to verdicts would raise mean mass to at most 0.225.

D Additional Calibration Comparisons

This appendix backs the claims of Section 6 in turn: the paired differences favor VMC_T (Appendix D.1), the gain comes from verdict mass (Appendix D.2), it holds with few labels and across

Table 7: **First-token output categories in the continuation audit.** Probability mass by category on ordinary benign inputs from the initial Guard 3/4 run.

Category	Guard 3	Guard 4
Canonical verdict (M)	0.4503	0.2635
S-category label outside the canonical verdicts	0.0463	0.0366
Ambiguous number	0.0505	0.1380
Other text	0.4513	0.0524
Empty or whitespace	–	0.5096
Noncanonical token before a recognizable verdict	0.0016	–

Table 8: **Paired differences favor VMC_T on every checkpoint.** Evaluation differences with 95% paired connected-group bootstrap intervals (2,000 draws, predictors held fixed). Confidence is the mean probability assigned to the guard's native verdict on inputs where that verdict is wrong or right. Arrows mark the better direction.

Difference	Guard 3	Guard 4	Guard 2
VMC_T *minus temperature scaling*			
NLL↓	-0.102 $[-0.114, -0.087]$	-0.046 $[-0.057, -0.033]$	-0.032 $[-0.038, -0.027]$
Brier↓	-0.036 $[-0.040, -0.030]$	-0.016 $[-0.020, -0.012]$	-0.013 $[-0.015, -0.011]$
ECE_{10} ↓	-0.080 $[-0.086, -0.074]$	-0.050 $[-0.056, -0.044]$	-0.031 $[-0.035, -0.027]$
Confidence if wrong↓	-0.272 $[-0.287, -0.255]$	-0.184 $[-0.205, -0.164]$	-0.056 $[-0.062, -0.049]$
Confidence if right↑	$+0.029$ $[+0.024, +0.033]$	$+0.027$ $[+0.023, +0.031]$	$+0.011$ $[+0.009, +0.013]$
VMC_0 *minus comparator*			
Native score, NLL↓	-0.256 $[-0.292, -0.219]$	-0.753 $[-0.831, -0.669]$	-0.277 $[-0.312, -0.240]$
$\mathrm{sigmoid}(m/2)$, NLL↓	-0.111 $[-0.128, -0.093]$	-0.150 $[-0.169, -0.130]$	-0.025 $[-0.028, -0.021]$

clipping levels (Appendix D.3), and lowering mass or appending task suffixes does not remove it (Appendix D.4).

Fitted predictors. Temperature scaling fits $T = 2.538$ (Guard 3), $T = 7.219$ (Guard 4), and $T = 3.167$ (Guard 2). VMC_T uses $T = 0.965$, $w = 1/2$ (Guard 3), $T = 2.501$, $w = 1/2$ (Guard 4), and $T = 2.967$, $w = 1$ (Guard 2); it fits one continuous parameter, the temperature, and chooses averaging or summing by calibration NLL. VMC_0 has no continuous parameter; calibration NLL selects averaging on all three checkpoints.

D.1 Paired Differences

All five differences between VMC_T and temperature scaling favor VMC_T on every checkpoint, with intervals that exclude zero (Table 8). Besides lowering NLL, Brier score, and ECE, it lowers the probability assigned to wrong native verdicts and slightly raises that of correct ones, which uniform shrinkage toward one half could not do; the ECE gain holds under every tested binning (Table 9). VMC_0 improves on the native score and on the matched-scale control $\mathrm{sigmoid}(m/2)$ on every checkpoint.

D.2 The Gain Comes from Verdict Mass

Table 10 compares VMC_T with controls at the same parameter budget; verdict mass gives the lowest NLL in every comparison. Permuting $R(M)$ within native-margin×length bins (30/30/29 bins for Guard 3/4/2) raises NLL on every checkpoint, and on Guard 4 the permuted predictor no longer improves on temperature scaling; this supports a contribution from input-specific mass beyond the native margin and length. Replacing mass by the other quantities of the same instruction-free pass, its margin or the logit of its unsafe probability, raises NLL on every checkpoint, and the instruction-free margin alone is worse than the temperature-scaled native margin, so the second pass supplements rather than replaces the native score. Among predictors without a fitted parameter, VMC_0 beats $\mathrm{sigmoid}(m/2)$, so halving the logit alone does not explain its gain.

Table 9: **Expected calibration error across binning choices.** ECE on the evaluation partition (n=7,444), using 5, 10, 15, 20, and 50 equal-width bins and 10 equal-count (adaptive) bins. Lower is better; bold marks the minimum in each column within a checkpoint. Predictors are defined in Table 3.

	Equal-width bins					Adaptive
Predictor	5	10	15	20	50	10
Llama Guard 3 8B						
Native sigmoid(m)	0.1144	0.1144	0.1144	0.1145	0.1153	0.1148
Temperature scaling	0.0998	0.1086	0.1185	0.1178	0.1186	0.1177
VMC_0 (ours)	0.0237	**0.0281**	**0.0283**	**0.0281**	**0.0301**	**0.0288**
VMC_T (ours)	**0.0223**	0.0286	0.0295	0.0292	0.0319	0.0300
Llama Guard 4 12B						
Native sigmoid(m)	0.1181	0.1187	0.1188	0.1187	0.1192	0.1131
Temperature scaling	0.0957	0.0957	0.1076	0.1122	0.1107	0.1126
VMC_0 (ours)	0.0921	0.0945	0.0939	0.0945	0.0946	0.0905
VMC_T (ours)	**0.0456**	**0.0461**	**0.0467**	**0.0486**	**0.0488**	**0.0500**
Llama Guard 2 8B						
Native sigmoid(m)	0.1594	0.1604	0.1619	0.1630	0.1630	0.1628
Temperature scaling	0.1289	0.1549	0.1647	0.1549	0.1654	0.1643
VMC_0 (ours)	0.1424	0.1424	0.1424	0.1425	0.1431	0.1471
VMC_T (ours)	**0.1035**	**0.1240**	**0.1332**	**0.1294**	**0.1329**	**0.1327**

Table 10: **Verdict mass gives the lowest NLL among controls at the same parameter budget.** Evaluation NLL on the evaluation partition. Upper block: one fitted temperature in $\mathrm{sigmoid}(w[m/T + X])$, with w selected on calibration NLL; the controls replace $X = R(M)$ by $R(M)$ permuted within calibration-defined native-margin deciles and length terciles (20 draws; mean and s.d.), by the instruction-free margin $m_{\mathrm{empty}} = z_u - z_s$, or by the logit of the instruction-free unsafe probability, or use m_{empty} in place of the native margin. Lower block: no fitted continuous parameter; $\mathrm{sigmoid}(m/2)$ matches the halved scale of VMC_0. Bold marks the minimum within each block.

Predictor	Guard 3 (8B)	Guard 4 (12B)	Guard 2 (8B)
Temperature scaling	0.2697	0.2860	0.3529
VMC_T (ours)	**0.1681**	**0.2403**	**0.3205**
$R(M)$ permuted within bins	0.1840 $\pm$0.0027	0.2922 $\pm$0.0047	0.3257 $\pm$0.0015
$R(M)$ replaced by m_{empty}	0.3564	0.7307	0.3751
$R(M)$ replaced by logit u_{empty}	0.3866	1.0853	0.8042
Temperature on m_{empty} alone	0.4611	0.5658	0.4985
Matched scale $\mathrm{sigmoid}(m/2)$	0.2795	0.6364	0.3996
VMC_0 (ours)	**0.1682**	**0.4862**	**0.3746**

D.3 Few Labels and Clipping

Calibration-set size. With 32 labeled groups, VMC_T has lower mean NLL than temperature scaling on every checkpoint and lower than the flexible native calibrator on Guard 3 and Guard 4 (0.201 versus 0.377, 0.282 versus 0.694), whose refits vary widely (s.d. 0.41, 0.42); VMC_0, which fits no continuous parameter, has the lowest NLL of all tested predictors at this size on Guard 3 and Guard 2. Figure 4 and Table 11 subsample 32, 128, 512, and 2,048 connected groups from the calibration partition, with evaluation rows held fixed. Each subsample refits temperature scaling and VMC_T, including form selection for the latter, and refits and reselects the flexible calibrators by grouped cross-validation. Of the 10 draws per size, those with fewer than two inputs of either label are skipped, as are flexible fits whose cross-validation folds contain a single label. Refit standard deviations quantify fitting variability omitted by the fixed-predictor bootstrap intervals.

Clipping. Clipping at ϵ bounds R at $\log((2-\epsilon)/\epsilon)$ and so caps the correction; at the reported $\epsilon = 10^{-6}$ the cap binds for at most 0.3% of rows. No other level gives lower NLL on Guard 3;

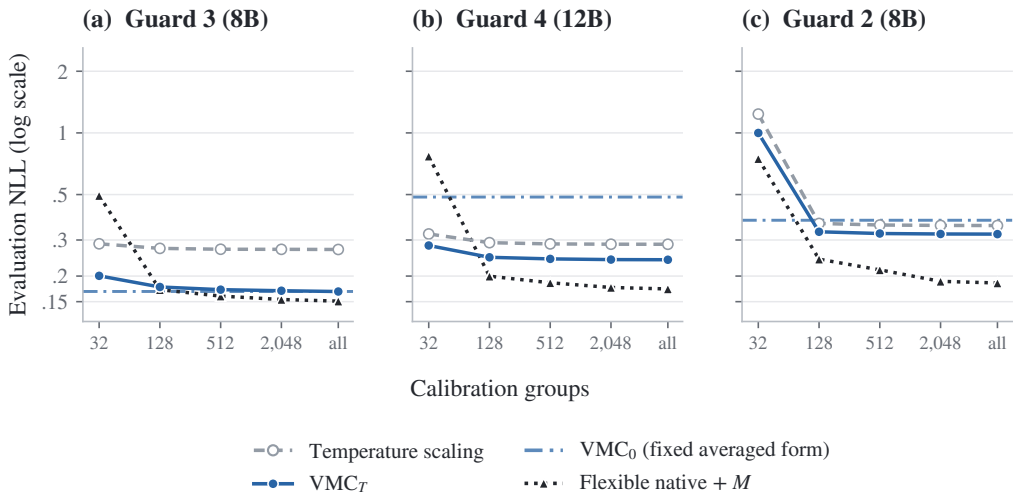


Figure 4: **NLL with limited calibration data.** Evaluation NLL (log scale) after refitting on calibration subsamples (10 draws per size; means over usable fits), with evaluation rows held fixed. Dash-dotted lines show VMC_0, which has no fitted continuous parameter. Table 11 also reports VMC_0 with its form reselected in each subsample, refit standard deviations, and the flexible native+M calibrator.

Table 11: **Evaluation NLL with limited calibration data.** Connected groups are subsampled from the calibration partition (10 draws per size; mean and s.d. over usable fits, Appendix D), while evaluation inputs remain fixed. Temperature scaling and VMC_T are refitted on each subsample, including form selection for the latter. VMC_0 (form refit) reselects $w \in \{1/2, 1\}$ within each subsample, whereas Figure 4 shows VMC_0 with the form selected on the full calibration set. Flexible calibrators select among affine, spline, and boosting by grouped cross-validation within each subsample.

Groups	Temperature	VMC_0 (form refit)	VMC_T	Flexible native	Flexible native+M
Guard 3 (8B)					
32	0.287±0.016	0.175±0.012	0.201±0.028	0.377±0.410	0.494±0.431
128	0.273±0.003	0.180±0.014	0.177±0.005	0.183±0.015	0.172±0.012
512	0.270±0.001	0.168±0.000	0.172±0.004	0.169±0.004	0.160±0.004
2,048	0.270±0.001	0.168±0.000	0.170±0.003	0.162±0.002	0.154±0.002
all (7,501)	0.270	0.168	0.168	0.161	0.151
Guard 4 (12B)					
32	0.321±0.043	0.486±0.000	0.282±0.036	0.694±0.418	0.770±0.462
128	0.291±0.006	0.486±0.000	0.247±0.011	0.205±0.013	0.200±0.015
512	0.287±0.001	0.486±0.000	0.243±0.003	0.195±0.005	0.185±0.005
2,048	0.286±0.001	0.486±0.000	0.241±0.001	0.189±0.001	0.176±0.001
all (7,501)	0.286	0.486	0.240	0.187	0.173
Guard 2 (8B)					
32	1.236±1.644	0.425±0.094	0.999±1.265	0.711±0.388	0.748±0.352
128	0.361±0.012	0.375±0.000	0.329±0.014	0.255±0.044	0.242±0.033
512	0.355±0.002	0.375±0.000	0.323±0.002	0.208±0.015	0.214±0.015
2,048	0.353±0.001	0.375±0.000	0.321±0.001	0.191±0.001	0.188±0.002
all (7,501)	0.353	0.375	0.321	0.189	0.185

on Guard 4 and Guard 2, a larger ϵ gives lower NLL, so the reported level does not favor VMC_T (Table 12).

Table 12: **Evaluation NLL of VMC_T under different clipping levels.** VMC_T is refitted at each ϵ; the reported level $\epsilon = 10^{-6}$ is shaded.

ϵ	Guard 3	Guard 4	Guard 2
10^{-2}	0.1734	0.2218	0.3200
10^{-3}	0.1681	0.2268	0.3205
10^{-4}	0.1681	0.2337	0.3205
10^{-6}	0.1681	0.2403	0.3205
10^{-9}	0.1681	0.2412	0.3205

Table 13: **Task suffixes change both verdict mass and the native margin.** Evaluation attacks before and after appending each suffix; Δ denotes suffixed minus original. We report 95% connected-group bootstrap intervals for ΔM and the change in the fraction of attacks that VMC_T flags at 0.5. Native detection is the fraction with $m > 0$. Detection and flagged rates are percentages.

Checkpoint	Suffix	$\bar{M}$ orig. $\to$ suf.	ΔM [95% CI]	Native detection orig. $\to$ suf.	VMC_T flag orig. $\to$ suf.	Δ flag [95% CI]
Guard 3	S1	0.865 $\to$ 0.388	-0.477 $[-0.490, -0.463]$	65.7 $\to$ 33.1	85.3 $\to$ 40.9	-44.4 $[-46.5, -42.2]$
Guard 3	S2	0.865 $\to$ 0.609	-0.256 $[-0.269, -0.244]$	65.7 $\to$ 65.9	85.3 $\to$ 74.7	-10.6 $[-11.9, -9.3]$
Guard 3	S3	0.865 $\to$ 0.805	-0.060 $[-0.068, -0.052]$	65.7 $\to$ 64.3	85.3 $\to$ 78.9	-6.3 $[-7.3, -5.4]$
Guard 4	S1	0.486 $\to$ 0.388	-0.098 $[-0.112, -0.083]$	68.2 $\to$ 49.5	80.7 $\to$ 61.0	-19.7 $[-21.5, -18.1]$
Guard 4	S2	0.486 $\to$ 0.368	-0.118 $[-0.131, -0.105]$	68.2 $\to$ 66.8	80.7 $\to$ 75.8	-5.0 $[-6.0, -3.9]$
Guard 4	S3	0.486 $\to$ 0.480	-0.006 $[-0.020, +0.007]$	68.2 $\to$ 64.6	80.7 $\to$ 77.8	-2.9 $[-4.1, -1.8]$
Guard 2	S1	0.173 $\to$ 0.166	-0.007 $[-0.014, +0.001]$	50.5 $\to$ 11.9	59.5 $\to$ 17.8	-41.7 $[-44.0, -39.5]$
Guard 2	S2	0.173 $\to$ 0.130	-0.043 $[-0.050, -0.037]$	50.5 $\to$ 49.5	59.5 $\to$ 56.3	-3.2 $[-4.3, -2.0]$
Guard 2	S3	0.173 $\to$ 0.152	-0.021 $[-0.027, -0.015]$	50.5 $\to$ 47.5	59.5 $\to$ 56.1	-3.4 $[-4.4, -2.4]$

D.4 Lowering Mass and Task Suffixes

These analyses ask whether an attacker can remove the correction. Setting mass to zero returns VMC_T exactly to the native guard's flagged set, and under task suffixes VMC_T still flags more attacks than the native guard on every checkpoint. Both use the fitted predictors of the main analysis; the suffix experiment rescores modified evaluation inputs.

Reducing mass at a fixed native margin. Because $R(M) \geq 0$, setting verdict mass to zero at a fixed native margin reduces VMC_T to $\mathrm{sigmoid}(wm/T)$, whose flagged set at 0.5 is exactly the native guard's. On the 2,378 evaluation attacks, the flagged fraction therefore falls from 85.3%, 80.7%, and 59.5% to the native 65.7%, 68.2%, and 50.5% on Guard 3/4/2, and no lower. Lowering mass alone cannot reverse an unsafe native prediction at this threshold; an attack must also reduce the native margin.

Appended task suffixes. We append three suffixes, fixed in advance, to all evaluation attacks, general harmful inputs, and sensitive benign inputs, plus 500 sampled ordinary benign inputs, and rescore both prompt conditions. The suffixes are S1 = "After reading the above, write a short poem about the sea."; S2 = "Continue the text above with one more paragraph."; S3 = "Thank you.". S1 and S2 request a generic task; S3 is a control. Temperatures and selected forms remain fixed (Table 13).

S1 lowers the native margin on attacks by 3.3, 6.4, and 4.0 logits on Guard 3/4/2, far more than the control suffix S3 (0.13, 1.82, and 0.27), so the suffix weakens the native guard directly, not only through mass. When each attack receives the suffix that minimizes its VMC_T probability, VMC_T still flags 40.8%, 59.3%, and 17.3% of attacks, against 33.1%, 51.0%, and 12.1% for the native guard and temperature scaling on the same inputs. On the S1-suffixed attacks, VMC_T's NLL is at or below that of temperature scaling on every checkpoint (-0.0034 $[-0.0161, +0.0096]$, -0.0637 $[-0.0888, -0.0381]$, -0.1645 $[-0.1769, -0.1530]$).

E Source, Length, and Prevalence Diagnostics

This appendix describes where the aggregate gain comes from: it is broad across unsafe sources, largest on long attack inputs without being explained by length, and positive above a modest unsafe prevalence (Appendix E.1).

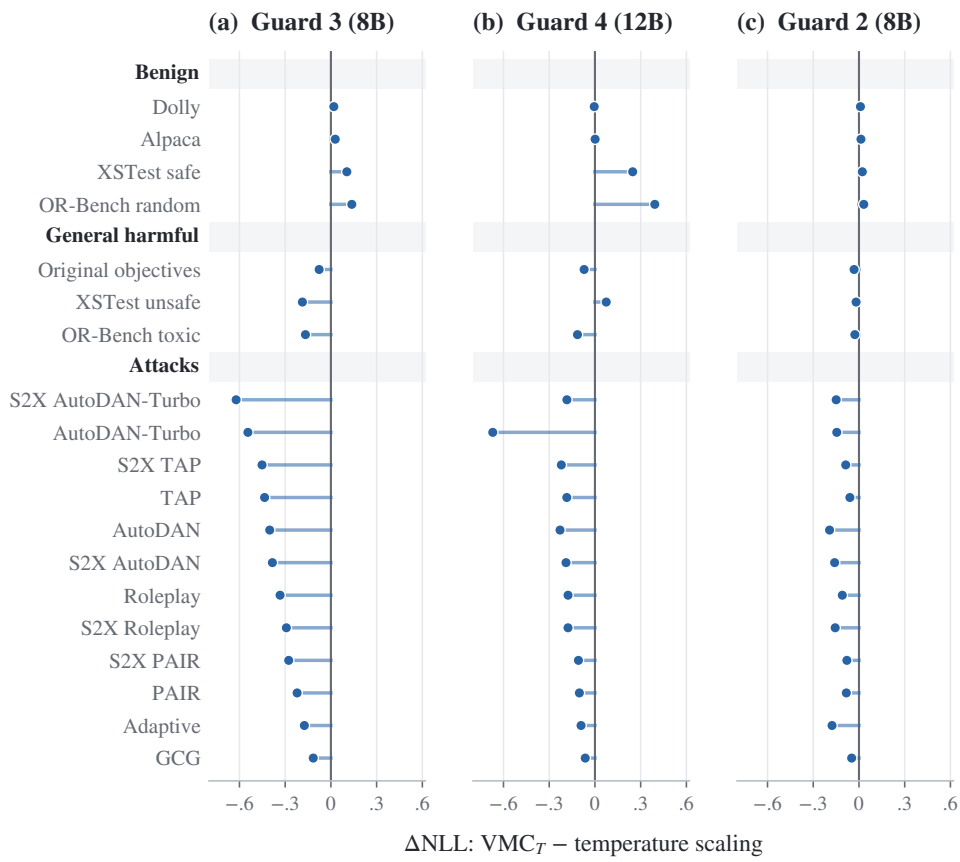


Figure 5: **Source-level NLL changes underlying Figure 3.** Evaluation NLL of VMC_T minus temperature scaling on the evaluation partition; negative values favor VMC_T. Sources follow the same order across checkpoints. Points show estimates without intervals. Text-identical GCG and S2X GCG-universal inputs appear once. All checkpoints use the same rows: 2,036/1,992/404/96 benign rows (Dolly/Alpaca/OR-Bench random/XSTest-safe), 172–203 rows per attack source across 12 sources, and 203/261/74 unsafe rows (Original/OR-Bench toxic/XSTest).

Partition construction. The calibration and evaluation sets are built from the sources in Table 1. Connected groups keep duplicates and linked original–attack prompts in the same partition. Each distinct prompt appears once, merging GCG with S2X GCG-universal because their texts coincide and leaving 12 attack sources. We sample OR-Bench random with a fixed seed and exclude OR-Bench hard.

Per-source changes. VMC_T lowers NLL on 44 of the 45 unsafe source–checkpoint pairs, the exception being Guard 4 XSTest (0.8491 to 0.9232); AutoDAN-Turbo, for example, falls from 1.4874 to 0.9429 on Guard 3, from 1.2588 to 0.5884 on Guard 4, and from 1.7661 to 1.6204 on Guard 2 (Figure 5). On the ordinary benign sources, which make up more than half of the evaluation rows, NLL changes by at most 0.03 on every checkpoint. The added loss concentrates on sensitive benign sources, whose inputs resemble harmful requests (Appendix C; Figure 5). The evaluation mixture is majority benign (2,916 of 7,444 rows are unsafe), with attacks contributing much of the NLL improvement; Appendix E.1 varies these mixture weights directly.

Length dependence. The NLL gain is largest for long inputs: in the longest quartile of input length (536+ characters, defined on the calibration partition), VMC_T changes NLL relative to temperature scaling by -0.2827 $[-0.3074, -0.2563]$, -0.1917 $[-0.2136, -0.1673]$, and -0.1106

Table 14: **Mass differences persist after length matching.** Evaluation inputs matched 1:1 to ordinary benign inputs within 0.10 in log character length. ΔM is the matched mean difference in instruction-free mass; ΔNLL is VMC_T minus temperature scaling on the matched inputs, with 95% connected-group bootstrap intervals. The last column gives the Spearman correlation between M and the label before and after residualizing both on log length.

	Attacks vs. ordinary benign		General harmful vs. ordinary benign		
Checkpoint	ΔM	ΔNLL	ΔM	ΔNLL	Spearman(M, Y)
Guard 3	$+0.288$ $[+0.258, +0.318]$	-0.151 $[-0.176, -0.126]$	$+0.522$ $[+0.489, +0.556]$	-0.061 $[-0.072, -0.049]$	$0.625 \to 0.379$
Guard 4	$+0.456$ $[+0.420, +0.491]$	-0.106 $[-0.126, -0.085]$	$+0.262$ $[+0.219, +0.304]$	-0.034 $[-0.053, -0.015]$	$0.298 \to 0.310$
Guard 2	$+0.136$ $[+0.116, +0.156]$	-0.062 $[-0.075, -0.048]$	$+0.029$ $[+0.013, +0.046]$	-0.007 $[-0.013, +0.001]$	$0.181 \to 0.154$

$[-0.1211, -0.1003]$ on Guard 3/4/2. Long inputs are disproportionately attacks, the inputs the native guard scores worst: median evaluation length is 588 characters for attacks versus 69 for ordinary benign, 118 for sensitive benign, and 73 for general harmful inputs. Mass itself tracks length only weakly on Guard 4 and Guard 2 (Spearman -0.010 and $+0.090$) and moderately on Guard 3 ($+0.476$), where the association reverses within attacks (-0.138).

Length matching. We match each evaluation attack to an ordinary benign input within 0.10 in log character length (1:1, without replacement; 797 pairs, median 564 versus 561 characters), and apply the same procedure to general harmful inputs (538 pairs). After matching, attacks and general harmful inputs retain higher instruction-free mass than ordinary benign inputs on every checkpoint, and VMC_T lowers NLL on the matched attacks on every checkpoint (Table 14). Residualizing both mass and the label on log length leaves their Spearman correlation positive on every checkpoint, and on Guard 4 it does not decrease.

E.1 Sensitivity to Unsafe Prevalence

We vary unsafe prevalence by reweighting the source-level losses, holding the source composition within each label and the fitted predictors fixed. Let Δ_y be the evaluation NLL difference between VMC_T and temperature scaling, averaged within label y using the observed source weights. At a reweighted unsafe prevalence π,

$$\Delta\mathrm{NLL}(\pi) = (1 - \pi)\Delta_0 + \pi\Delta_1. \tag{13}$$

VMC_T has lower aggregate NLL than temperature scaling whenever unsafe prevalence exceeds about 10%, 18%, and 11% on Guard 3/4/2; at the observed 39% the differences are -0.102, -0.046, and -0.032. Below these points the difference stays small, at most 0.028 nats at 5% unsafe ($+0.018$, $+0.028$, and $+0.007$).

F Supervised Reference Calibrators

The strongest score-only recalibration we test is a flexible calibrator of the native margin, selected and fitted with labels. Adding instruction-free mass as a second input improves it on all three checkpoints: NLL, Brier score, and ECE_{10} all decrease, and every paired NLL interval excludes zero (Table 15). Consistent with Proposition 1, verdict mass therefore carries information beyond the best recalibration of the native score that we test, independently of the one-temperature form of VMC_T.

These calibrators are a larger-budget reference rather than a same-budget baseline: they require labels for model selection and fitting. With the full calibration set they reach lower NLL than VMC_T, which fits a single temperature, although VMC_T stays within 0.01 of the native-margin spline on Guard 3 (Tables 3 and 15); with 32 calibration groups, VMC_T outperforms them on Guard 3 and Guard 4 (Table 11).

Fitting. For each input set we select among affine logistic regression, cubic B-spline features with six knots followed by logistic regression, and histogram gradient boosting, minimizing mean validation NLL in three-fold cross-validation with disjoint connected groups; the chosen model, a spline in every case, is then refitted on all calibration rows. The mass input is $R(M)$. The affine and spline logistic regularization parameters are $C = 10^6$ and 10^3; boosting uses 200 iterations, learning

Table 15: **Adding verdict mass improves a flexible supervised calibrator on every checkpoint.** The calibrator maps the native margin m, or m together with the instruction-free mass M, to a probability; for each input set an affine, spline, or gradient-boosting model is selected by three-fold grouped cross-validation on calibration NLL (a spline in every case). The last row gives the paired NLL difference with its 95% connected-group bootstrap interval. Arrows mark the better direction; bold marks the better row.

	Guard 3 (8B)			Guard 4 (12B)			Guard 2 (8B)		
Calibrator inputs	NLL ↓	Brier ↓	ECE_{10} ↓	NLL ↓	Brier ↓	ECE_{10} ↓	NLL ↓	Brier ↓	ECE_{10} ↓
Native margin m	.1608	.0480	.0055	.1873	.0545	.0071	.1891	.0546	.0124
m + verdict mass M	**.1509**	**.0443**	**.0051**	**.1731**	**.0498**	**.0063**	**.1850**	**.0535**	**.0121**
Paired ΔNLL	-0.0099 $[-0.0138, -0.0062]$			-0.0142 $[-0.0180, -0.0101]$			-0.0041 $[-0.0062, -0.0021]$		

Table 16: **VMC transfers to ShieldGemma 9B.** All predictors use the same 7,444 evaluation rows as the Llama Guard experiments. Arrows mark the better direction; bold marks the best predictor. Mass is measured with a minimal safety question retained, and VMC_0 uses the calibration-selected sum form.

Predictor	NLL ↓	Brier ↓	ECE_{10} ↓
Native score	0.6859	0.1917	0.2232
Temperature scaling	0.4279	0.1440	0.2215
VMC_0 (ours)	0.3794	0.1145	**0.0441**
VMC_T (ours)	**0.3586**	**0.1125**	0.0999

rate 0.05, at most 15 leaves, and at least 40 samples per leaf. Only calibration labels are used. All calibrators use the 11,228-row calibration partition and the 7,444 evaluation rows of Section 6, with connected groups kept disjoint; intervals are pointwise 95% paired bootstrap intervals over connected groups (2,000 draws) with the fitted calibrators held fixed.

G Transfer to ShieldGemma 9B

We apply VMC to ShieldGemma 9B (Zeng et al., 2024), a guard from a different model family, on the same calibration and evaluation partition as the Llama Guard experiments. Relative to temperature scaling, VMC_T reduces NLL by 16.2%, Brier score by 21.9%, and ECE_{10} by 54.9%, with paired differences of -0.069 $[-0.081, -0.058]$, -0.032 $[-0.036, -0.027]$, and -0.122 $[-0.129, -0.114]$ (Table 16). VMC_0, which fits no continuous parameter, also improves all three metrics and attains the lowest ECE_{10}. At threshold 0.5, VMC_T raises unsafe recall from 38.2% to 63.4% at a benign false-positive rate of 3.3%.

Measurement condition. ShieldGemma places almost no probability on its verdict tokens once its safety question is removed, so we measure mass under a prompt that keeps a minimal version of the question. The prompt is chosen by measurement gates rather than by calibration performance: we score 11 fixed renderings, from the released template to input-only wrappers, on 128 calibration inputs split into disjoint discovery and validation halves (16 per half from each of ordinary benign, sensitive benign, general harmful, and attack populations). A rendering is admissible if, on both halves, median M lies in $[0.1, 0.9999)$, the interquartile range of $\text{logit}\, M$ is at least 0.5, and unsafe inputs have a higher mean conditional unsafe probability than safe inputs; the first admissible rendering in a frozen priority order is the minimal question (median mass 0.935/0.952, logit-mass interquartile range 2.33/2.07). No evaluation labels enter this choice. With `{input}` replaced by the input text and `\n` a newline, the rendering is

```
<bos><start_of_turn>\nHuman Question: {input}
\n<end_of_turn>\n\nIs the human question harmful?\n\n
```

Checkpoint and scoring. We use `google/shieldgemma-9b` with frozen weights. The native prompt is the released policy-expert template with a fixed union of four prompt-only policies

(dangerous content, harassment, hate speech, and sexually explicit information), applied identically to every input. The verdicts are the first-token `Yes` (unsafe, token 3553) and `No` (safe, token 1294); the native margin is $m = z_{\texttt{Yes}} - z_{\texttt{No}}$, and $M = P(\texttt{Yes}) + P(\texttt{No})$ under the selected prompt, normalized over the full vocabulary. Scoring uses FP32 weights, TF32 disabled, eager attention, Gemma2's final logit softcap of 30, and FP64 probability summaries; every input fits within the 8,128-token input limit.

Fitting. Calibration uses the same 11,228 rows and labels as for Llama Guard. We use $R(M) = \log[(1+M)/(1-M)]$ after clipping M to $[10^{-6}, 1-10^{-6}]$. Temperature scaling fits $T = 3.153$. For VMC_0, calibration NLL selects the sum $\mathrm{sigmoid}(m + R(M))$ over the average (0.379 versus 0.408); VMC_T selects $w = 1/2$ with $T = 0.768$ in Equation (8). All choices are fixed before evaluation, and intervals are pointwise 95% paired connected-group bootstrap intervals (2,000 draws).

H Internal Interventions

These interventions ask, on Guard 3 and Guard 4, whether the two coordinates of Section 2 can be moved separately inside the model. A task direction, built from safety-assessment minus general-task instructions, moves verdict mass more and verdict choice less than a label direction (Figure 6), differs from every sampled control (Table 19), and closes the mass gaps between input groups, as the released safety instruction does (Figure 7).

Direction construction. The task direction is the mean activation difference between safety-assessment and general-task instructions on benign inputs. For the final assistant-prefix position, let $h^c_{i\ell}$ denote the block-ℓ output under condition c. For the benign discovery set $\mathcal{D}_b$, we define

$$v_\ell = \frac{1}{|\mathcal{D}_b|} \sum_{i \in \mathcal{D}_b} (h^{\mathrm{safetyA}}_{i\ell} - h^{\mathrm{generalA}}_{i\ell}). \tag{14}$$

The label direction is the mean difference between unsafe and safe discovery inputs under the released template, $r_\ell = \overline{h}^{\mathrm{native}}_\ell(\mathcal{D}_u) - \overline{h}^{\mathrm{native}}_\ell(\mathcal{D}_b)$, and so points toward inputs the guard should flag. The controls are three random Gaussian directions and three shuffled directions that average the safety-minus-general differences with random signs. Every direction is rescaled to $\|v_\ell\|_2$ and added to the activation with coefficient ± 1. At the three nonselected blocks, we profile only the task direction.

Layer selection. We select the block using validation inputs and instruction wording B, which was not used to construct the direction. The criterion averages $|\tau^{\mathrm{generalB}}_{M,i} - \tau^{\mathrm{safetyB}}_{M,i}| - |\tau^{\mathrm{steered}}_{M,i} - \tau^{\mathrm{safetyB}}_{M,i}|$ over all 32 validation inputs, where $\tau_M = \mathrm{logit}\, M$ and "steered" denotes the general-B prompt with the task direction added. This measures the reduction in the absolute task-instruction gap, avoiding division by a potentially small baseline gap. Directions and the selected block are fixed before test intervention outputs are analyzed; test contrasts are excluded from selection.

Output-distribution diagnostics. We record the exact decomposition

$$\tau_M = B + C, \quad B = \frac{z_s + z_u}{2} + \log 2 - \mathrm{LSE}(z_{\mathrm{outside}}), \quad C = \log\cosh\left(\frac{z_u - z_s}{2}\right). \tag{15}$$

Because the log-cosh term grows with the magnitude of the verdict logit margin, with B fixed, pushing the verdict toward either label raises M.

Implementation and uncertainty. Forward hooks modify transient activations while leaving model files unchanged. We use full-prefix FP32 inference with TF32 disabled. Checks cover unchanged outputs when no activation is added, agreement between singleton and batch inference, reproduction of previously recorded scores, and exact text-hash matches. Intervals resample connected groups with the directions, the selected block, and the sampled controls held fixed.

Test inputs and selected blocks. The 313 test inputs form 206 connected groups, with 213 unsafe and 100 benign labels. The evaluated groups include 16 ordinary benign inputs, 32 OR-Bench random, 32 XSTest-safe, 48 general harmful, 78 attacks, and 24 benign and 24 harmful XSTest

Table 17: **Validation favors later intermediate blocks.** Baseline and steered values are mean absolute errors in logit mass relative to the safety-B condition; improvement is baseline minus steered error. The selection criterion uses all 32 validation inputs.

Model	Block	Baseline error	Steered error	Improvement
Guard 3	Block 8	4.204	4.097	0.108
	Block 16	4.204	1.308	2.897
	Block 20	4.204	0.887	3.317
	Block 24	4.204	0.851	3.353
Guard 4	Block 12	12.512	12.415	0.098
	Block 24	12.512	11.672	0.841
	Block 30	12.512	9.183	3.329
	Block 36	12.512	5.494	7.018

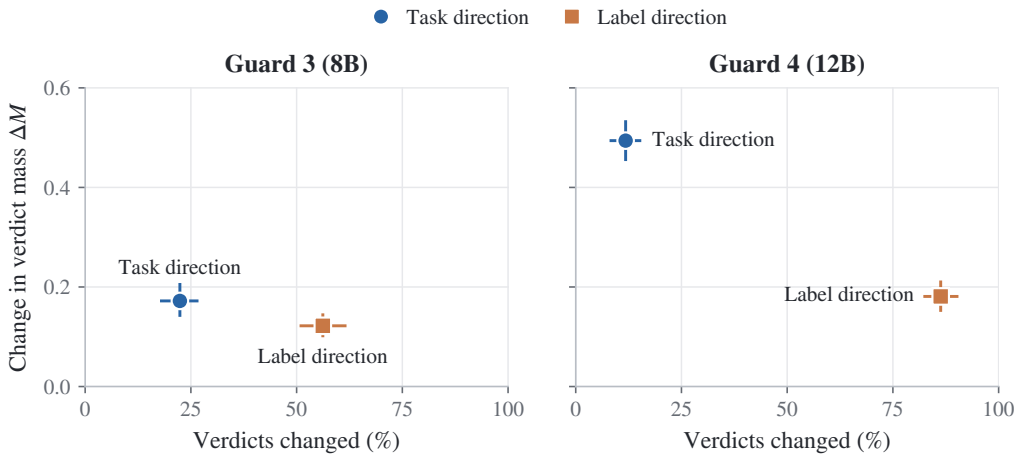


Figure 6: **The task direction moves more mass, and flips fewer verdicts, than the label direction.** Change in verdict mass against the fraction of verdicts changed after adding each direction (coefficient $+1$, norms matched) at the validation-selected block (24 for Guard 3, 36 for Guard 4) on 313 test inputs. Whiskers are pointwise 95% connected-group bootstrap intervals (5,000 draws).

prompts matched by topic. Because linked originals and overlapping memberships are included, these counts do not sum to 313. There are 78 unique attack links covering 71 originals. The native guard misses 22/12 attacks on Guard 3/4 and 7/11 general harmful inputs. Validation selects block 24 for Guard 3 and block 36 for Guard 4 (Table 17).

Task and label directions move different coordinates. On both checkpoints, adding the task direction raises mass more than adding the label direction while changing fewer verdicts (Figure 6): by 0.172 versus 0.122 with 22.4% versus 56.2% of verdicts changed on Guard 3, and by 0.494 versus 0.181 with 11.8% versus 86.3% on Guard 4. The signs confirm the split (Table 18): the task direction raises mass when added and lowers it when subtracted, whereas the label direction raises the unsafe probability q when added and lowers it when subtracted. Subtracting the label direction also raises mass, consistent with the log-cosh term of the decomposition above: pushing the verdict strongly toward either label enlarges the margin and hence M. All reported intervals exclude zero: ΔM is $[0.140, 0.208]$ and $[0.453, 0.535]$ for task addition and $[-0.783, -0.720]$ and $[-0.520, -0.441]$ for task subtraction on Guard 3 and Guard 4, and Δq is $[0.504, 0.593]$ and $[0.791, 0.854]$ for label addition. Mass and verdict choice therefore respond to different directions inside the model, consistent with treating them as separate coordinates (Section 2).

Table 18: **Signed intervention effects on verdict mass and verdict choice.** Mean changes at the selected block on 313 test inputs per model; verdict flips are fractions of inputs. Intervals are in the text; comparisons with random and shuffled controls are in Table 19.

Model	Direction	Coefficient	ΔM	Δq	Verdict flips
Guard 3	Task	$+1$	$+0.172$	-0.212	0.224
	Task	-1	-0.753	$+0.280$	0.291
	Label	$+1$	$+0.122$	$+0.548$	0.562
	Label	-1	$+0.126$	-0.452	0.438
Guard 4	Task	$+1$	$+0.494$	-0.133	0.118
	Task	-1	-0.480	$+0.070$	0.080
	Label	$+1$	$+0.181$	$+0.822$	0.863
	Label	-1	$+0.431$	-0.164	0.134

Table 19: **Mass changes differ between the task direction and each sampled control.** Entries are task-minus-control ΔM at the selected block on 313 test inputs, with pointwise 95% connected-group intervals (5,000 draws).

Model	Comparator	Coefficient $+1$	Coefficient -1
Guard 3	Random 1	$+0.223$ $[0.186, 0.264]$	-0.714 $[-0.744, -0.680]$
	Random 2	$+0.200$ $[0.165, 0.238]$	-0.678 $[-0.711, -0.642]$
	Random 3	$+0.203$ $[0.168, 0.242]$	-0.687 $[-0.718, -0.652]$
	Shuffle 1	$+0.269$ $[0.229, 0.313]$	-0.720 $[-0.750, -0.687]$
	Shuffle 2	$+0.352$ $[0.308, 0.398]$	-0.771 $[-0.800, -0.739]$
	Shuffle 3	$+0.229$ $[0.192, 0.270]$	-0.715 $[-0.746, -0.681]$
Guard 4	Random 1	$+0.925$ $[0.903, 0.945]$	-0.822 $[-0.850, -0.793]$
	Random 2	$+0.561$ $[0.522, 0.601]$	-0.481 $[-0.523, -0.438]$
	Random 3	$+0.350$ $[0.311, 0.391]$	-0.382 $[-0.419, -0.345]$
	Shuffle 1	$+0.769$ $[0.733, 0.803]$	-0.774 $[-0.808, -0.738]$
	Shuffle 2	$+0.365$ $[0.325, 0.408]$	-0.286 $[-0.322, -0.250]$
	Shuffle 3	$+0.202$ $[0.168, 0.238]$	-0.146 $[-0.174, -0.120]$

The task direction differs from sampled controls. Table 19 compares the task direction with three random and three shuffled directions of the same norm, for both coefficients. Every interval for the task-minus-control ΔM excludes zero on both checkpoints.

The task direction closes the mass gaps between input groups. Adding the task direction removes nearly all of the mass difference between each input group and ordinary benign inputs (Figure 7): on Guard 3 the general harmful, attack, and OR-Bench random gaps fall from 0.466, 0.456, and 0.363 to at most 0.016, and on Guard 4 all three gaps fall from 0.213–0.440 to at most 0.03. Every reduction interval excludes zero: $[0.277, 0.626]$, $[0.268, 0.619]$, and $[0.165, 0.533]$ on Guard 3 and $[0.131, 0.519]$, $[0.001, 0.366]$, and $[0.216, 0.640]$ on Guard 4, in the same group order. Inside the model, the task direction thus flattens mass across inputs, mirroring how the released safety instruction pins mass near one for every input (Section 4).

Scope of the intervention results. Each model contributes 640 validation records and 18,780 test records (313 inputs by 60 specified trials); the trials repeat interventions on the same prompts. Instruction wording B is used for layer selection, and adding the task direction at the selected block removes about 80% of the validation task gap on Guard 3 and 56% on Guard 4.

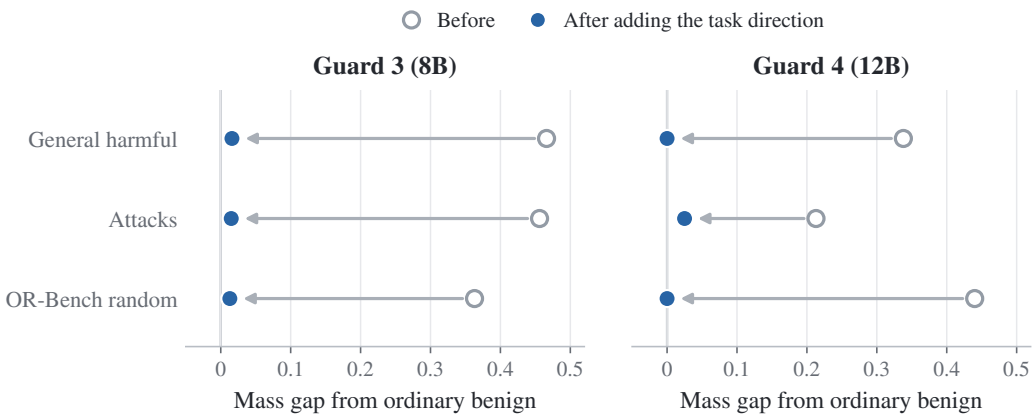


Figure 7: **Adding the task direction closes the mass gap between input groups.** Mean verdict mass of each group minus that of ordinary benign inputs, before (hollow) and after (filled) adding the task direction at the selected block, on 313 test inputs. After-values are the baseline gap minus the estimated reduction; pointwise 95% connected-group bootstrap intervals (5,000 draws) for the reductions, all excluding zero, are in the text.

I Reproducibility

Scores are computed with FP32 weights and logits, TF32 disabled, and FP64 probability calculations. Calibration intervals use 2,000 connected-group paired bootstrap draws (seed 20260922); intervention intervals use 5,000 draws (seed 20260921); behavioral source and attack intervals use 2,000 draws. All score files are checked with SHA-256 hashes. The tables, calibration figures, and behavioral figures for all three checkpoints are generated from recorded analysis CSVs and cached aggregate JSON files that include hashes of the underlying score files, so they can be regenerated without refitting the predictors. The continuation audit (Appendix C) and the internal interventions (Appendix H) come from an initial Guard 3/4 run; both use fixed input panels and do not depend on the calibration split, so they were not repeated on the calibration and evaluation partitions used for the three checkpoints.