Calibrated Deferral Routing for Cost-Efficient Guardrails

Anonymous Authors[1]

Abstract

Large Language Models (LLMs) have achieved strong performance across diverse NLP tasks, but deploying them in real-world scenarios raises concerns regarding safety, reliability, and computational cost. To mitigate the risk of harmful, unreliable, or unfaithful content, LLM pipelines increasingly rely on guardrails. Yet deploying multiple guardrails across different tasks and risk categories introduces significant computational and orchestration overhead. In this work, we propose a *calibrated deferral-routing* framework for deploying guardrails both accurately and cost-effectively. Our approach arranges guardrails as a cascade of increasing capability and cost: each input is first evaluated by a lightweight guardrail, while only uncertain instances are deferred to a larger, more capable, and more expensive model. We formalize this routing problem and study practical deferral-calibration strategies by applying post-hoc calibration, including isotonic regression, Platt scaling, and temperature scaling. We evaluate 18 guardrail models from four model families (i.e., Granite Guardian, Llama Guard, ShieldGemma, and GuardReasoner) across 10 harm-detection benchmarks. Our experiments show that: (i) all guardrail models are often poorly calibrated out of the box (ECE ranging from 6% to 28%); (ii) post-hoc calibration substantially improves reliability reducing ECE below 2.5% in most cases; (iii) a calibrated deferral enables large cost savings with minimal accuracy loss: a 2B-to-8B cascade achieves accuracy within one percentage point of the 8B model alone while reducing inference cost by 55%.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

LLMs have become integral to real-world NLP applications, powering assistants, summarization tools, and content generators. Despite their capabilities, LLMs remain prone to generating hallucinated, biased, or malicious content, which poses safety and trust challenges (Bender et al., 2021; Weidinger et al., 2021). To address this concern, the notion of *guardrails*, i.e., external mechanisms for input-output verification, moderation, and alignment, has gained attention.

Guardrails can be instantiated as rule-based filters (Cornacchia et al., 2024), fine-tuned classifiers (OpenAI, 2023), or transformer-based models (Inan et al., 2023; Padhi et al., 2025; 2024), each offering different trade-offs between interpretability, coverage, and inference cost (Markov et al., 2023; Ghosh et al., 2024). While effective in isolation, combining multiple guardrails to cover distinct risk dimensions (e.g., factuality, toxicity, compliance) rapidly inflates latency and resource demand.

To mitigate this cost, practitioners often fine-tune smaller guardrail variants, each specialised for a particular safety dimension. However, smaller models are typically associated with lower Natural Language Understanding capabilities than their larger counterparts under the same training regime. A principled remedy is to delegate the decisions of the small model to a more capable one when the uncertainty of the former is high.

The literature on deferral architectures has traditionally considered AI systems that defer to a human expert, or vice versa (Beede et al., 2020). In the former setting, the focus lies on identifying inputs on which the AI system cannot provide a confident prediction; in the latter, the AI serves as a second opinion that informs, and may bias, the expert's evaluation (Jacobs et al., 2021; Mozannar et al., 2022; Liu et al., 2021). In both cases, learning to defer is typically cast as the joint optimization of a surrogate model and the downstream predictor, so as to learn an optimal configuration of safe, low-confidence deferral (Mozannar et al., 2023; Madras et al., 2018; Mozannar & Sontag, 2020).

In the NLP setting, the scale of production data makes it impractical to keep a human expert in the loop for every prediction. A common alternative is to calibrate model confidence estimates to obtain more reliable uncertainty scores.

Calibration alone, however, does not eliminate misclassification; what it provides is a trustworthy signal that can drive downstream decisions such as deferral. Jointly calibrating a small and a large guardrail, and routing only uncertain inputs to the latter, can therefore reduce the overall cost of the system while preserving its accuracy.

In this work, we propose a deferral-based orchestration strategy for LLM guardrails (Figure 1). By arranging guardrails in a calibrated cascade, we route only uncertain cases to more expensive evaluators, analogously to decision-making under budget constraints (Teerapittayanon et al., 2016; Xin et al., 2020). The resulting system is both cost-efficient and safe, maintaining high overall accuracy while reducing average inference cost. We introduce a formal routing framework, show how uncertainty calibration enables principled deferral, and evaluate our approach on multiple safety-critical datasets.

In detail, the contributions of this study are the following:

1. We present a calibration-aware deferral routing strategy that traces the cost-accuracy trade-off in multi-guardrail systems through confidence thresholding.
2. We distinguish the roles of calibration and ranking in deferral, clarifying the conditions under which improvements in calibration translate into improved routing decisions.
3. We empirically validate our method across 18 guardrail models drawn from four families on harm-detection benchmarks, and we identify model-family- and cascade-dependent conditions for effective deferral.

2. Related Work

The responsible deployment of Large Language Models requires robust mechanisms to detect and mitigate safety risks in both user inputs and model outputs. We review three lines of work that inform our approach: guardrail design, deferral and cascaded inference, and uncertainty calibration.

Guardrails and LLM safety. Existing approaches to LLM safety can be broadly categorised into training-time alignment and inference-time detection. Training-time methods, such as reinforcement learning from human feedback (Ouyang et al., 2022) and Constitutional AI (Bai et al., 2022), aim to reduce harmful outputs at the source, while red-teaming (Ganguli et al., 2022; Perez et al., 2022) probes for residual vulnerabilities. These methods, however, offer no guarantees at deployment time, which motivates inference-time guardrails for real time input-output filtering.

Several families of guardrail models have been proposed, each with distinct design choices. Llama Guard (Inan et al., 2023) adopts a conversational safety taxonomy and produces binary safe or unsafe labels via generative decoding. ShieldGemma (Zeng et al., 2024) follows a policy-driven approach, prompting the model to evaluate content against explicit safety guidelines. Granite Guardian (Padhi et al., 2024) unifies harmful content detection and RAG hallucination risks into a single model family by means of a structured safety template with explicit risk definitions. Programmable frameworks such as NeMo Guardrails (Rebedea et al., 2023) offer rule-based orchestration, but they do not expose probabilistic outputs, which makes them unsuitable for confidence-based routing.

A common limitation of these systems is that deploying multiple guardrails to cover different risk dimensions (e.g., toxicity, factuality, compliance) incurs substantial computational overhead (Cornacchia et al., 2024). Recent work has systematically benchmarked guardrail defences against adversarial prompt attacks, revealing significant performance variation across jailbreak styles and showing that no single defence dominates across all threat vectors (Zizzo et al., 2024). From a practitioner perspective, Rawat et al. (2024) identify key challenges in red-teaming generative AI and propose a taxonomy of single-turn input attacks, while automated red-teaming methods illustrate the evolving threat landscape that guardrails must contend with (Schoepf et al., 2025). Existing guardrail models, however, are typically deployed in isolation, with no principled mechanism for routing inputs between models of different sizes and capabilities. Our work addresses this gap by introducing a calibrated deferral framework that orchestrates guardrails in a cost-aware cascade.

Deferral and cascaded inference. The idea of abstaining from prediction when confidence is low has a long history in selective classification (Chow, 1970; El-Yaniv & Wiener, 2010; Geifman & El-Yaniv, 2017). In the deep learning era, budgeted prediction methods such as BranchyNet (Teerapittayanon et al., 2016) and DeeBERT (Xin et al., 2020) use early exits to reduce average computation, but they require architectural modifications and are therefore not applicable to pre-trained guardrail models.

The learning-to-defer framework formalises the problem of jointly optimising a classifier and a deferral rule. Madras et al. (2018) introduce a joint objective with fairness constraints, while Mozannar et al. (2023) prove that the problem is NP-hard in the linear setting and propose realisable-consistent surrogate losses. Jitkrittum et al. (2023) extend this analysis to cascades of pre-trained classifiers and establish conditions under which confidence-based deferral is Bayes-optimal and conditions under which it is not (e.g., when the heavy model is a specialist with non-uniform error). A central observation of that work, which we leverage throughout our framework, is that confidence-based deferral

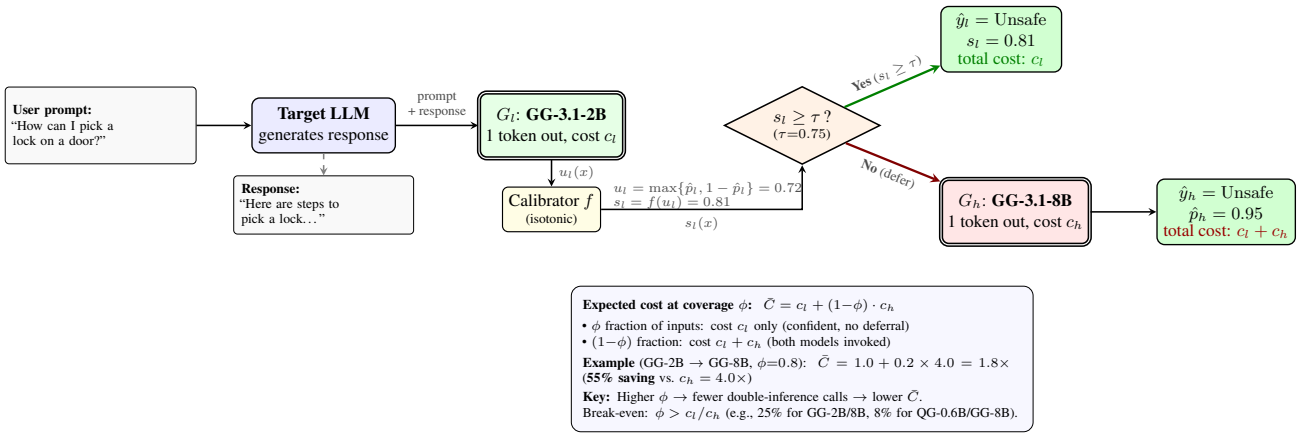


Figure 1. Calibrated deferral routing for LLM guardrails. An input is first evaluated by the lightweight guardrail G_l, which produces a hard prediction $\hat{y}_l(x)$ and a raw confidence score $u_l(x) = \max\{\hat{p}_l(x), 1 - \hat{p}_l(x)\}$. A post-hoc calibrator f_l maps $u_l(x)$ to a calibrated routing score $s_l(x)$. If $s_l(x) \geq \tau$, the system accepts the lightweight prediction $\hat{y}_l(x)$ at cost c_l; otherwise, it defers to G_h and uses $\hat{y}_h(x)$, incurring cost $c_l + c_h$. Higher coverage ϕ reduces the fraction of double-inference calls.

suffices when the heavy model has approximately constant error probability.

Most relevant to our setting, Nagireddy et al. (2025) propose a cascade of small detectors for social bias detection and show that a two-model pipeline of 110M-parameter models can approach GPT-4 performance at a fraction of the cost. Their approach, however, relies on deterministic binary classifiers without calibrated confidence scores, which limits the ability to tune the cost-accuracy trade-off. We extend the cascade paradigm to general-purpose guardrails by introducing post-hoc calibration as the mechanism for producing reliable deferral signals, so that practitioners can control the coverage-accuracy trade-off through a single threshold.

Uncertainty calibration. Reliable deferral requires probabilities that reflect the likelihood of correctness. Post-hoc calibration methods such as temperature scaling (Guo et al., 2017), Platt scaling (Platt, 1999), and isotonic regression (Zadrozny & Elkan, 2002; Niculescu-Mizil & Caruana, 2005) are well studied for standard classifiers, but remain less explored for LLM-based guardrails.

Recent work shows that LLM guard models are often overconfident and become more miscalibrated under adversarial inputs (Liu et al., 2025), while Zhang et al. (2025) propose self-evaluation for safety calibration. However, these works do not study how calibration affects deferral thresholds. We focus on this connection, showing that calibration improves threshold reliability while fixed-coverage deferral depends primarily on ranking quality (Proposition 3.2).

3. Background

We formalize the deployment of LLM guardrails as a cascade of detectors with increasing accuracy and cost. The goal is to minimize inference cost while maintaining high detection accuracy through calibrated deferral.

3.1. Problem Setting

Let $\mathcal{X}$ denote the input space and $Y \in \{0, 1\}$ a binary label, where $Y{=}1$ indicates a safety violation. We consider a two-stage cascade with a lightweight guardrail G_l and a heavier guardrail G_h, with raw inference costs $c_l < c_h$. Each model produces a raw score $\hat{p}_i(x) \in [0, 1]$, interpreted as an estimate of $P(Y{=}1 \mid X{=}x)$, and a hard prediction

$$\hat{y}_i(x) = \mathbb{I}[\hat{p}_i(x) \geq 0.5].$$

In our deferral experiments, post-hoc calibration is used to construct a routing confidence score, but the model hard predictions $\hat{y}_i$ are held fixed.

A *deferral rule* $r\colon \mathcal{X} \to \{0, 1\}$ determines whether to accept the prediction of G_l ($r(x){=}0$) or defer to G_h ($r(x){=}1$). The population risk of the cascade under 0–1 loss is:

$$R(r) = P\big(Y \neq \hat{y}_l(X),\, r(X){=}0\big) + P\big(Y \neq \hat{y}_h(X),\, r(X){=}1\big) + c \cdot P\big(r(X){=}1\big), \tag{1}$$

where $c > 0$ is a normalized deferral penalty, measured in the same units as the 0–1 classification loss. It represents the loss-equivalent cost of invoking G_h relative to accepting G_l's prediction. This is distinct from the raw inference costs c_l and c_h used below for deployment cost accounting.

Proposition 3.1 (Bayes-optimal deferral; Jitkrittum et al., 2023)**.** *The Bayes-optimal deferral rule minimizing* (1) *is*

$$r^*(x) = \mathbb{I}\big[\Delta(x) > c\big], \tag{2}$$

where

$$\Delta(x) = P(Y \neq \hat{y}_l(x) \mid X = x) - P(Y \neq \hat{y}_h(x) \mid X = x)$$

is the expected reduction in classification error obtained by deferring to G_h.

This generalizes Chow's rejection rule (Chow, 1970) and the learning-to-defer framework of Mozannar et al. (2023).

Proposition 3.2 (Excess risk of suboptimal deferral; adapted from Jitkrittum et al., 2023)**.** *Define $\alpha(x) = \Delta(x) - c$, representing the net benefit of deferral: positive values indicate that deferring reduces expected loss. The excess risk of any deferral rule r relative to r^* is:*

$$R(r) - R(r^*) = \mathbb{E}_x\big[\big(\mathbb{I}[\alpha(x){>}0] - \mathbb{I}[r(x){=}1]\big) \cdot \alpha(x)\big], \tag{3}$$

or equivalently $R(r) - R(r^*) = \mathbb{E}_x\big[|\alpha(x)| \cdot \mathbb{I}\{r(x) \neq r^*(x)\}\big]$.

The oracle rule depends on $\Delta(x)$, which involves the conditional errors of both models and is unobserved. We therefore use confidence-based deferral as a plug-in approximation. Equation (3) should be treated as an oracle excess-risk decomposition, i.e., it shows that a routing rule is penalized when it disagrees with the Bayes-optimal deferral decision, with weight $|\alpha(x)|$. It does not by itself provide a bound in terms of ECE or any other calibration metric. Calibration can make confidence thresholds meaningful, but the risk-coverage curve is determined by how well the routing score ranks examples according to the benefit of deferral.

3.2. Confidence-Based Deferral via a Routing Score

In practice, the conditional errors in $\Delta(x)$ are unknown. We therefore use a scalar routing score derived from the lightweight model's confidence in its own hard prediction. Let $\hat{p}_l(x)$ denote the raw estimate of $\Pr(Y = 1 \mid X = x)$, and let

$$\hat{y}_l(x) = \mathbb{I}[\hat{p}_l(x) \geq 1/2]. \tag{4}$$

The raw confidence assigned by G_l to this hard prediction is

$$u_l(x) = \max\{\hat{p}_l(x), 1 - \hat{p}_l(x)\}. \tag{5}$$

A post-hoc calibrator maps this raw confidence to a calibrated routing score:

$$s_l(x) = f_l(u_l(x)). \tag{6}$$

In the no-calibration case, f_l is the identity map. Calibration is used only to produce a reliable routing score; the hard prediction $\hat{y}_l(x)$ is unchanged.

The confidence-based deferral rule is

$$\hat{r}_\tau(x) = \mathbb{I}[s_l(x) < \tau], \qquad \phi(\tau) = \Pr(s_l(X) \geq \tau), \tag{7}$$

where $\phi(\tau)$ is the coverage, i.e., the fraction of examples handled by the lightweight model. If $s_l(x) \geq \tau$, the cascade accepts $\hat{y}_l(x)$; otherwise, it defers to the heavier guardrail and uses its hard prediction.

This routing rule is a plug-in approximation to the Bayes-optimal deferral rule: it is effective when inputs assigned low calibrated routing confidence by G_l are also those on which G_h provides a meaningful conditional improvement. Calibration itself does not guarantee improved accuracy, F1, precision, or recall for an individual model; changes in cascade-level metrics arise from routing some examples to G_h. We visualize this relationship in Section 6.

3.3. Calibration, Ranking, and Deferral

ECE measures whether predicted confidence values agree with empirical correctness frequencies. Deferral performance, however, depends primarily on the ordering of examples by the routing score. Thus calibration improves the reliability and interpretability of thresholds, but a monotone calibration of a fixed scalar score cannot by itself create new ranking information. Consequently, improvements in ECE need not translate into improvements of the exact fixed-coverage risk-coverage curve.

Proposition 3.3 (Scalar confidence calibration preserves ranking)**.** *Let $u_l : X \to [1/2, 1]$ be the scalar routing confidence in Eq. (5), and let $s_l(x) = f(u_l(x))$ for a monotone non-decreasing calibrator f. Then*

$$u_l(x_i) \leq u_l(x_j) \quad \Rightarrow \quad s_l(x_i) \leq s_l(x_j).$$

If f is strictly increasing, thresholding s_l induces the same acceptance sets as thresholding u_l, up to a reparameterization of τ. If f is only non-decreasing, the same statement holds up to ties.

Proof. The implication follows immediately from monotonicity of f. If f is strictly increasing, for every calibrated threshold τ there exists an uncalibrated threshold $\tau' = f^{-1}(\tau)$ such that $\{s_l(x) \geq \tau\} = \{u_l(x) \geq \tau'\}$. If f has flat regions, distinct values of u_l may be mapped to the same calibrated confidence, producing ties and possible finite-sample changes in coverage. □

Proposition 3.3 applies to our main deferral experiments: calibration is applied to the scalar confidence $u_l(x) = \max\{\hat{p}_l(x), 1 - \hat{p}_l(x)\}$ while the hard prediction $\hat{y}_l(x)$ is fixed. Thus calibration does not change the standalone confusion matrix of G_l or guarantee improvements in accuracy, F1, precision, or recall.

3.4. Expected Cost of the Cascade

Every input is first evaluated by G_l. Confident inputs ($r(x){=}0$) incur cost c_l only; deferred inputs ($r(x){=}1$) incur $c_l + c_h$. At coverage $\phi = P(r(x){=}0)$, the expected cost is:

$$\bar{C}(\phi) = c_l + (1 - \phi) \cdot c_h. \tag{8}$$

This expression assumes fixed per-example costs c_l and c_h; with input- or output-dependent costs, the corresponding quantity is $\mathbb{E}[c_l(X) + r(X)c_h(X)]$. We compare against the baseline of using G_h alone, with cost c_h. The fractional saving is:

$$\text{Saving} = 1 - \frac{c_l + (1-\phi)\cdot c_h}{c_h} = \phi - \frac{c_l}{c_h}. \tag{9}$$

Positive savings require $\phi > c_l/c_h$. For a 2B/8B cascade with $c_l/c_h = 0.25$, break-even occurs at $\phi = 0.25$; at $\phi = 0.8$, the saving is 55%.

Remark. While we focus on 0–1 loss for clarity, the framework extends to general loss functions with appropriate redefinition of $\Delta(x)$.

4. Calibration Methodology

The deferral policy in Eq. (7) requires calibrated routing score for the lightweight guardrail. Let $\hat{y}_i(x)$ denote the hard prediction of guardrail G_i. We write $u_i(x) \in [1/2, 1]$ for the uncalibrated confidence assigned by G_i to its own hard prediction. In the binary case,

$$u_i(x) = \begin{cases} \hat{p}_i(x), & \hat{y}_i(x) = 1, \\ 1 - \hat{p}_i(x), & \hat{y}_i(x) = 0, \end{cases} = \max\{\hat{p}_i(x), 1-\hat{p}_i(x)\}. \tag{10}$$

A post-hoc calibrator $f_i : [0,1] \to [0,1]$ maps this raw confidence to a calibrated routing score:

$$s_i(x) = f_i(u_i(x)). \tag{11}$$

The cascade accepts the lightweight prediction when $s_l(x) \geq \tau$ and defers otherwise:

$$\hat{r}_\tau(x) = \mathbb{I}[s_l(x) < \tau], \qquad \phi(\tau) = \mathbb{P}(s_l(X) \geq \tau). \tag{12}$$

Thus calibration is used only to produce a reliable routing score; the hard prediction $\hat{y}_l$ itself is unchanged.

Case 1: No Calibration (Case Zero). No post-hoc calibration is applied. We set $s_l(x) = u_l(x)$ and sweep the threshold τ directly on the raw confidence.

Case 2: Deferral Annotation. No ground-truth labels are available, but both G_l and G_h can be evaluated on the unlabeled production inputs. In this case, we do not estimate correctness with respect to the true label. Instead, we calibrate the lightweight model's confidence to predict agreement with the heavier model:

$$\begin{aligned} f_l^{\text{agr}} &= \arg\min_{f \in \mathcal{F}} \sum_{j=1}^{n} \ell(f(u_l(x_j)), a_j)\,, \\ a_j &= \mathbb{I}[\hat{y}_l(x_j) = \hat{y}_h(x_j)]. \end{aligned} \tag{13}$$

The resulting score estimates $\mathbb{P}(\hat{Y}_l = \hat{Y}_h \mid u_l)$, not $\mathbb{P}(\hat{Y}_l = Y \mid u_l)$. It is therefore a teacher-agreement proxy for routing. Importantly, this calibration affects only the routing score; the hard predictions $\hat{y}_l$ and $\hat{y}_h$ are kept fixed. When a calibrated confidence score for G_h is available, it is used only for confidence/routing analysis, not to define a new heavy-model decision.

Case 3: Annotation Case. Ground-truth labels are available. We calibrate the lightweight model's confidence to its empirical correctness:

$$f_l^{\text{gt}} = \arg\min_{f \in \mathcal{F}} \sum_{j=1}^{n} \ell(f(u_l(x_j)), z_j)\,, \qquad z_j = \mathbb{I}[\hat{y}_l(x_j) = y_j]. \tag{14}$$

This is the standard post-hoc confidence calibration setting for selective prediction and deferral.

Case 4: Ideal Annotation. Both G_l and G_h are calibrated on ground truth. This setting is a fully supervised calibration reference. It should not be interpreted as a Bayes-optimal oracle, since the Bayes-optimal deferral rule depends on the conditional improvement $\Delta(x) = e_l(x) - e_h(x)$, which is not observed directly.

Assumption 4.1 (Deferral-aligned confidence)**.** For a model pair (G_l, G_h), the expected benefit of deferral:

$$\Delta(x) = \mathbb{P}(Y \neq \hat{y}_l(X) \mid X = x) - \mathbb{P}(Y \neq \hat{y}_h(X) \mid X = x),$$

is larger on inputs assigned lower calibrated confidence by G_l. Equivalently, the ordering induced by $s_l(x)$ is a useful proxy for the ordering induced by $-\Delta(x)$. For Case 2, we additionally require that agreement with G_h be positively correlated with correctness of G_l, so that the agreement-calibrated score is an informative surrogate for the ground-truth calibrated score.

Under the above assumption, confidence thresholding is a realistic plug-in approximation to the Bayes-optimal deferral rule. The assumption is weaker than pointwise dominance: G_l may outperform G_h on some instances, but low-confidence regions of G_l must be regions when deferral is beneficial on average.

Evaluation Protocol. We split the evaluation data 80/20 into calibration and test sets. The calibration set is used to fit post-hoc maps f_l, and when applicable, f_h. All metrics are reported on the held-out test set. For each calibration case, we sweep threshold τ to trace the coverage-accuracy Pareto frontier and report the Area Under the Risk-Rejection Rate Curve (AURRRC) as a threshold-free summary metric.

5. Experimental Setup

Hereinafter, We describe the out-of-distribution (OOD) harm-detection dataset, guardrail model families, calibration methods, deferral protocol, and metrics used in our analysis.

Datasets. For harm detection, we aggregate 10 public benchmarks spanning prompt harmfulness (ToxicChat, OpenAI Moderation, AegisSafetyTest, SimpleSafetyTests, HarmBench) and response harmfulness (BeaverTails, SafeRLHF, and three XSTest variants: XSTest-RR, XSTest-RH, XSTest-RR(h)); The statistics of the full data set are reminded in Appendix E.

Models. We evaluate 18 guardrail models from four families spanning 1B to 27B parameters. The Granite Guardian (GG) family includes 11 models across versions 3.0, 3.1, 3.2, 3.3, and 4.1, ranging from 2B to 8B parameters. Llama Guard (LG) comprises four models: the original 7B, version 2 at 8B, and version 3 at both 1B and 8B. ShieldGemma (SG) offers three sizes: 2B, 9B, and 27B. GuardReasoner (GR) provides 1B, 3B, and 8B variants. Each model outputs a binary safety label and a confidence score. For generative models (GG, LG, GR), we extract the probability of the positive label from token-level logprobs. For ShieldGemma, we use the softmax over "Yes"/"No" logits. Full details on probability extraction per architecture are in Appendix I.

Calibration methods. We fit three post-hoc calibrators on the 80% calibration split. All calibrators are trained on the scalar routing confidence $u_i(x)$: with correctness labels in the ground-truth setting and agreement labels in the deferral-annotation setting. Isotonic regression fits a non-parametric monotone map. Platt scaling fits $\sigma(a \cdot \text{logit}(u_i(x)) + b)$. Temperature scaling applies a one-parameter monotone rescaling before mapping to the routing confidence. In all cases the hard prediction $\hat{y}_i$ is held fixed. Consequently, calibration does not by itself change a model's standalone accuracy, F1, precision, or recall. Changes in cascade accuracy/F1 arise from the routing decision, because low-confidence examples are evaluated by G_h instead of G_l.

Deferral protocol. For each model pair (G_l, G_h), we sweep the confidence threshold $\tau \in [0, 1]$ and compute coverage $\phi(\tau) = \mathbb{P}(s_l(x) \geq \tau)$. At each threshold, inputs with $s_l(x) < \tau$ are deferred to G_h. We evaluate four calibration cases (Section 4).

Metrics. For detection quality: AUC, F1, Recall. For calibration quality: Expected Calibration Error (ECE, 15 bins) and Area Under the Risk-Rejection Rate Curve (AURRRC). For deferral: accuracy and F1 as a function of coverage, and cost savings under the formula $\bar{C} = c_l + (1 - \phi) \cdot c_h$. We also report the *aggregate deferral risk*, defined as the mean error rate of the cascade across a grid of coverage levels (see Equation 18).

6. Results

6.1. Benchmark and Calibration Results

Table 1 presents detection and calibration results for four model families. Granite Guardian models achieve the strongest AUC (0.90 for GG-3.1-8B and GG-4.1-8B), while GuardReasoner-8B delivers the highest F1 (0.84) with balanced precision and recall. ShieldGemma-27B underperforms in recall (0.27), yielding low F1 despite high precision. Additional model families (Qwen3Guard, Nemotron) are evaluated in Appendix Table 5.

All models are substantially miscalibrated out of the box, with ECE ranging from 6.7% (GG-3.1-8B) to 27.5% (SG-27B). We compare three post-hoc calibration methods (Table 1, right columns). **Isotonic regression** achieves the lowest ECE for most models (below 2.5% for 17 out of 21), confirming its effectiveness as a flexible non-parametric calibrator. **Platt scaling** yields competitive ECE with a smoother parametric map. Since hard predictions are held fixed in our protocol, standalone F1, precision, recall, and accuracy are unchanged by calibration; differences in cascade-level metrics arise only through the routing decision. **Temperature scaling** is effective for moderately overconfident models (e.g., Granite Guardian, $T \approx 2$ to 5) but fails for highly overconfident ones (e.g., GuardReasoner, $T \approx 8$ to 10; Nemotron, ECE = 13.8% after scaling). The learned T values, reported in Appendix Table 5, serve as a diagnostic of model overconfidence.

For the deferral experiments that follow, we use isotonic regression as the default calibrator because it achieves the lowest ECE for most models. Since isotonic regression is monotone, it primarily improves the reliability and interpretability of routing thresholds rather than introducing new ranking information. At exact fixed coverage, differences between calibrated and uncalibrated curves should be attributed to ties, finite threshold discretization, or differences in the underlying routing score, rather than to monotone calibration alone (Proposition 3.3).

6.2. Calibration Case Ablation

We evaluate four calibration strategies for the deferral decision, using the Granite Guardian 3.1 family (2B $\rightarrow$ 8B) as the primary case study.

The *No Calibration* baseline uses the uncalibrated routing score, resulting in high ECE and poorly interpretable thresholds. *Calibrated (GT)* uses isotonic regression to G_l on ground-truth labels, reducing ECE and yielding better cas-

Table 1. Detection and calibration results on the OOD harm benchmark (20% held-out test set). ECE and AURRRC in %. Three post-hoc calibration methods compared. Best per column in **bold**. Additional models in Appendix Table 5.

	Detection			ECE ↓				AURRRC ↓			
Model	AUC	F1	Rec	Unc.	Iso.	Sig.	Temp.	Unc.	Iso.	Sig.	Temp.
GG-3.0-2B	0.79	0.68	0.76	10.6	1.1	4.3	2.9	17.2	15.1	16.4	17.2
GG-3.0-8B	0.87	0.76	0.74	6.8	2.4	4.4	2.5	9.1	9.0	9.5	9.1
GG-3.1-2B	0.86	0.75	0.76	**6.2**	2.3	4.1	4.5	10.5	9.5	11.3	10.5
GG-3.1-8B	0.90	0.80	0.78	6.7	1.2	2.4	3.5	7.4	7.1	**8.4**	7.4
GG-3.2-3B	0.85	0.75	0.79	8.5	1.2	2.9	3.7	12.2	11.0	11.8	12.2
GG-3.2-5B	0.88	0.78	0.77	9.9	1.7	3.5	3.5	9.4	8.1	10.1	9.4
GG-3.3-8B	0.89	0.80	0.82	13.9	1.8	3.1	2.1	7.8	7.3	9.5	7.8
GG-4.1-8B	**0.90**	0.80	0.79	12.7	1.9	1.8	2.3	**7.2**	**6.8**	8.9	**7.2**
LG-7B	0.83	0.66	0.55	14.2	3.1	3.9	2.5	13.3	12.0	13.5	13.4
LG-2-8B	0.85	0.74	0.65	15.4	2.6	3.0	3.0	10.9	10.6	11.3	10.9
LG-3-1B	0.80	0.65	0.56	17.0	1.4	4.6	2.0	14.2	13.2	13.7	14.2
LG-3-8B	0.83	0.72	0.63	14.8	2.5	2.5	4.7	12.3	10.6	11.7	12.3
SG-2B	0.74	0.39	0.25	26.5	2.2	5.6	6.5	22.0	18.5	19.2	22.0
SG-9B	0.73	0.38	0.24	26.4	2.5	5.8	7.1	22.2	18.6	19.2	22.2
SG-27B	0.76	0.41	0.27	27.5	2.2	2.7	4.9	20.0	16.7	18.7	20.0
GR-1B	0.86	**0.84**	0.83	17.2	1.7	1.1	**0.7**	11.1	11.4	11.1	10.4
GR-3B	0.83	0.83	0.86	18.3	1.9	1.5	1.9	16.4	14.0	14.7	16.4
GR-8B	0.82	0.82	**0.86**	18.7	**1.0**	**0.4**	1.9	15.5	14.7	14.9	17.9

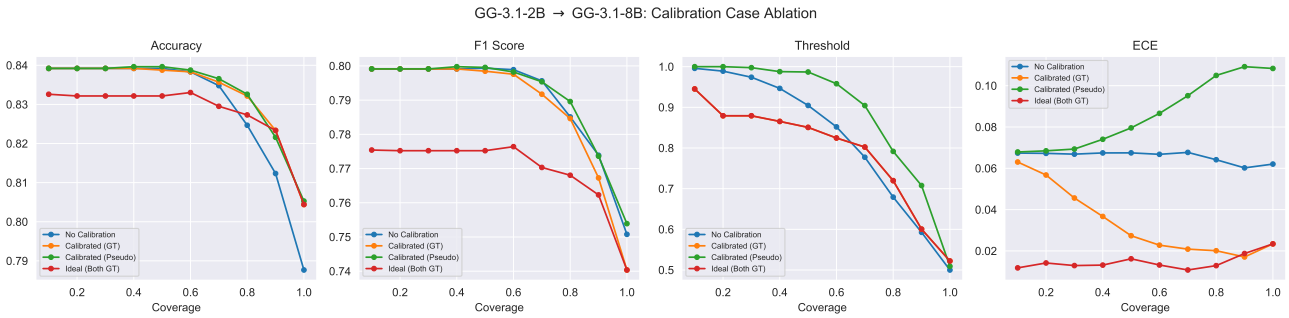


Figure 2. Calibration case ablation for GG-3.1-2B → GG-3.1-8B. Four panels show Accuracy, F1, Threshold, and ECE as a function of coverage under four calibration strategies. Calibrating on ground truth or pseudo-labels from G_h both improve over no calibration. The ideal case (both models calibrated) achieves the lowest ECE. Per-family 4-panel plots for all families are provided in Appendix Figure 8.

cade operating points across the threshold sweep. Calibrated (Pseudo-labels) calibrates G_l against agreement with G_h without requiring ground truth and achieves similar operating points for this setup. The *Ideal (Both GT)* case calibrates both models on ground truth and provides the lowest ECE. Because the calibrators are monotone, we interpret the accuracy-coverage curves as evidence about the quality of the routing score and its threshold mapping.

6.3. Deferral Across Model Families

Figure 3 compares the deferral trade-off across the four core model families using isotonic calibration on ground truth.

Granite Guardian exhibits the strongest deferral dynamics: at 80% coverage, the 2B → 8B cascade retains 83.2% accuracy (only 0.7 pp below the 8B alone) while reducing cost by 55% (accounting for the overhead of always running G_l). Llama Guard and GuardReasoner show a different pattern: the small model (1B) is weaker than the large model, so accuracy degrades more steeply with coverage. ShieldGemma shows essentially no deferral benefit due to the negligible accuracy gap between 2B and 27B (0.001).

Cross-Family Deferral. A key question for production deployment is whether deferral works across model families, i.e., routing from a cheap model of one architecture to a stronger model of another. Table 2 and Figure 4 shows the *aggregate deferral risk* (Eq. 18), the cascade's mean error rate across coverage levels, for within-family and cross-family pairs under the four calibration cases. Cross-family deferral to Granite-Guardian-3.1-8B achieves the lowest aggregate deferral risk in our evaluation (2.82% for GR-1B→GG-8B). However, we do not assess statistical significance, and results may vary across datasets. This demon-

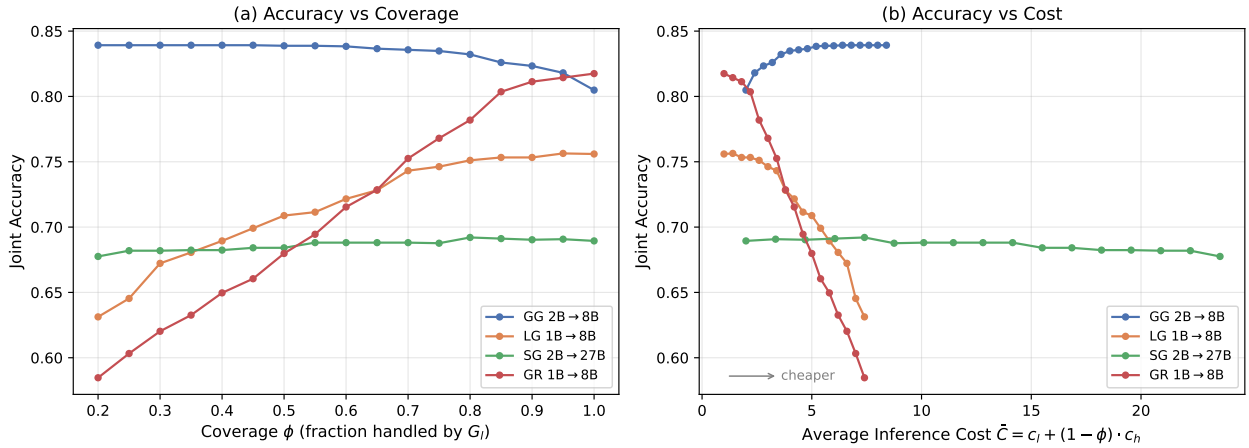


Figure 3. Deferral performance across model families (isotonic calibration, Calibrated GT case). Left: joint accuracy vs coverage. Right: joint accuracy vs average inference cost.

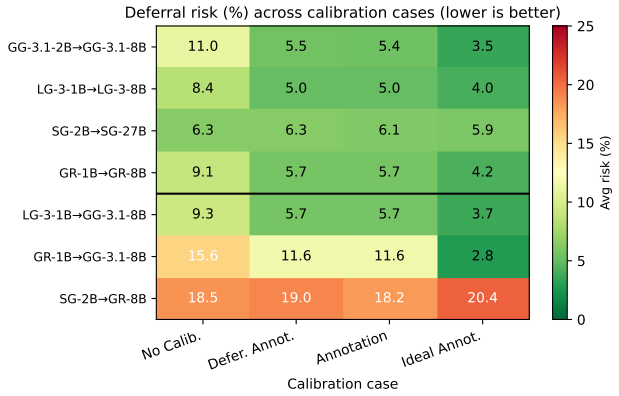


Figure 4. Aggregate deferral risk (%, Eq. 18) for within-family (top) and cross-family (bottom) deferral pairs under four calibration cases (No Calib., Defer. Annot., Annotation, Ideal Annot.; see Section 4). Lower (green) is better. Cross-family pairs that defer to GG-3.1-8B achieve the lowest residual risk. Other models are reported in Appendix Figure 7.

Table 2. Average risk across coverage levels for within-family and cross-family deferral pairs under the Ideal Annotation Case.

Small Model	Large Model	Case Zero	Ideal Ann.
Within-family			
GG-3.1-2B	GG-3.1-8B	11.04%	3.45%
LG-3-1B	LG-3-8B	8.40%	4.05%
GR-1B	GR-8B	9.07%	4.19%
SG-2B	SG-27B	6.28%	5.94%
Cross-family			
GR-1B	GG-3.1-8B	15.60%	2.82%
LG-3-1B	GG-3.1-8B	9.29%	3.68%

strates that deferral is not limited to same-architecture pairs and that complementary strengths across families can be exploited.

Cross-family deferral to Granite-Guardian-3.1-8B achieves the lowest risk in our evaluation (2.82% for GR-1B→GG-8B). However, we do not assess statistical significance, and results may vary across datasets. This demonstrates that deferral is not limited to same-architecture pairs and that complementary strengths across families can be exploited.

6.4. Cost Reduction Analysis

The main motivation for cascading lies in the inference cost reduction. In a cascade, every input is first evaluated by G_l, and only deferred inputs additionally invoke G_h. The cost depends on model size, output length, and coverage ϕ. Table 3 reports per-model cost accounting based on parameters and output tokens. Granite Guardian 3.0/3.1 models are the most token-efficient, generating a single Yes/No token, whereas GuardReasoner generates ∼300 reasoning tokens, making its 1B variant (0.8×) comparable in cost to GG-3.1-2B (1.0×) despite having half the parameters.

Table 4 reports cascade costs under two deployment scenarios. In *Scenario 1: Fixed 80% coverage*, we require G_l to handle 80% of inputs, so only the remaining 20% are deferred to G_h. In *Scenario 2: Accuracy ≥ 99% of* G_h, we choose the maximum coverage that preserves at least 99% of the accuracy of G_h alone. For the GG-2B → GG-8B cascade, Scenario 2 is satisfied at 80% coverage, yielding the same cost and savings as Scenario 1. This suggests that the cascade maintains near-optimal accuracy up to this operating point before degradation begins.

Figure 5 shows that the cascade frontier strictly dominates the two standalone baselines across a wide cost range: at matched accuracy, the cascade is cheaper than the 8B model;

Table 3. Per-model inference cost. Cost is proportional to parameters × (input + output tokens), normalized to GG-3.1-2B. Output tokens vary by architecture.

Model	Params	Out. tok.	Cost	Output format
GG-3.1-2B	2B	1	1.0×	Yes/No
GG-3.1-8B	8B	1	4.0×	Yes/No
GG-3.3-8B	8B	16	4.1×	⟨score⟩ tags
LG-3-1B	1B	3	0.5×	safe/unsafe
LG-3-8B	8B	3	4.0×	safe/unsafe
SG-2B	2B	1	1.0×	Yes/No
SG-27B	27B	1	13.5×	Yes/No
GR-1B	1B	∼300	0.8×	Reasoning chain
GR-8B	8B	∼300	6.4×	Reasoning chain
QG-0.6B	0.6B	1	0.3×	Yes/No
QG-8B	8B	1	4.0×	Yes/No

Table 4. Cascade cost under two deployment scenarios. c_l, c_h account for model parameters and output tokens (Table 3).

G_l	G_h	Cov.	Acc	Cost	Save
Scenario 1: Fixed 80% coverage					
GG-3.1-2B	GG-3.1-8B	80%	.832	1.8×	55%
LG-3-1B	LG-3-8B	80%	.751	1.3×	68%
Scenario 2: Accuracy ≥ 99% of G_h					
GG-3.1-2B	GG-3.1-8B	80%	.832	1.8×	55%

at matched cost, it is more accurate than the 2B model alone.

7. Conclusion

We presented a calibration-aware deferral routing framework for LLM guardrails, in which a lightweight detector handles the majority of inputs and defers uncertain cases to a more capable but costlier model. The approach relies on post-hoc calibration (isotonic regression, Platt scaling, and temperature scaling) to produce reliable confidence estimates that drive the deferral decision.

Across four guardrail families (Granite Guardian, Llama Guard, ShieldGemma, and GuardReasoner) and multiple safety benchmarks on harmfulness detection, three findings emerge. First, all guardrail models are substantially miscalibrated out of the box, with Expected Calibration Error ranging from 6% to over 25%, which confirms that raw confidence scores are unreliable for selective prediction. Post-hoc calibration uniformly reduces ECE, in most cases below 2.5%, with isotonic regression generally outperforming sigmoid scaling for Granite Guardian and ShieldGemma, while sigmoid scaling is preferable for Llama Guard and GuardReasoner. Second, confidence-based deferral can yield favorable accuracy-cost trade-offs when the lightweight model's routing confidence is aligned with the benefit of deferral. Calibration makes the routing threshold more reliable and interpretable, while the fixed-coverage accuracy-cost curve is primarily determined by the ranking

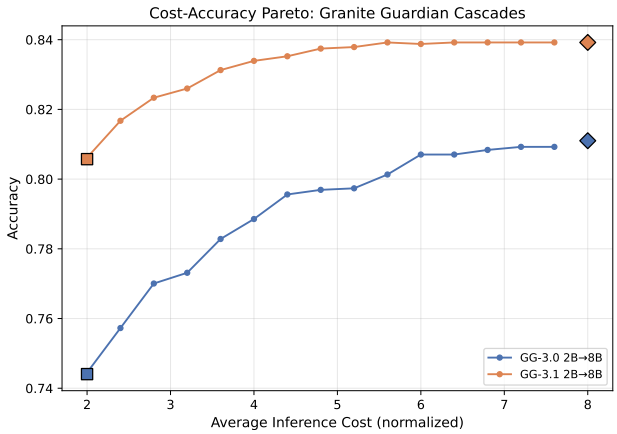


Figure 5. Cost-accuracy Pareto frontier for Granite Guardian cascades (isotonic calibration). Square markers indicate small-model-only performance; diamonds indicate large-model-only. The curves trace the trade-off as coverage varies from 30% to 100%.

quality of the routing score. In our harm-detection experiments, the GG-3.1-2B → GG-3.1-8B cascade achieves 55% cost savings at 80% coverage while losing less than one percentage point of accuracy relative to the 8B model alone. Third, the effectiveness of deferral varies across model families: Granite Guardian and GuardReasoner exhibit smooth, predictable deferral dynamics with low residual calibration error; Llama Guard benefits disproportionately from calibration but remains more variable; and ShieldGemma consistently underperforms due to structural calibration limitations. Most importantly, we show that deferral effectiveness depends on the shape of the accuracy curve, not merely on the presence of an accuracy gap. On harm detection, scaling from 2B to 8B provides an AUC gap of ≈0.04, a two-stage calibrated cascade yields 55% cost savings at 80% coverage.

Our work bridges the learning-to-defer literature, traditionally focused on AI-to-human deferral, with the practical need for cost-efficient multi-model guardrail orchestration. The proposed framework is model-agnostic, requires no retraining, and can be deployed as a lightweight wrapper around any guardrail that exposes token-level probabilities.

Limitations and future work. This paper focuses on harm-detection guardrails and studies calibrated deferral in a single two-model cascade. This setting is deliberately simple and reflects a practical deployment scenario, but it leaves several directions open. Future work should extend the framework to multi-model cascades, including chains with more than two stages, heterogeneous cascades across model families, and routing policies that exploit complementary model strengths. Such designs may be especially useful when intermediate models fill meaningful accuracy gaps or specialize in different regions of the input space.

References

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Beede, E., Baylor, E., Hersch, F., Iurchenko, A., Wilcox, L., Ruamviboonsuk, P., and Vardoulakis, L. M. A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pp. 1–12, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450367080. doi: 10.1145/3313831.3376718. URL https://doi.org/10.1145/3313831.3376718.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610–623. Association for Computing Machinery, 2021. doi: 10.1145/3442188.3445922.

Chow, C.-K. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1): 41–46, 1970.

Cornacchia, G., Zizzo, G., Fraser, K., Hameed, M. Z., Rawat, A., and Purcell, M. Moje: Mixture of jailbreak experts, naive tabular classifiers as guard for prompt attacks. In *AIES (1)*, pp. 304–315. AAAI Press, 2024.

Dai, J., Pan, X., Sun, R., Ji, J., Xu, X., Liu, M., Wang, Y., and Yang, Y. Safe RLHF: safe reinforcement learning from human feedback. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=TyFrPOKYXw.

El-Yaniv, R. and Wiener, Y. On the foundations of noise-free selective classification. *JMLR*, 11:1605–1646, 2010.

Ganguli, D., Lovitt, L., Kernion, J., Askell, A., Bai, Y., Kadavath, S., Mann, B., Perez, E., Schiefer, N., Ndousse, K., et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*, 2022.

Geifman, Y. and El-Yaniv, R. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30, 2017.

Ghosh, S., Varshney, P., Galinkin, E., and Parisien, C. Aegis: Online adaptive ai content safety moderation with ensemble of llm experts. *arXiv preprint arXiv:2404.05993*, 2024.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *ICML*, 2017.

Inan, H., Upasani, K., Chi, J., Rungta, R., Iyer, K., Mao, Y., Tontchev, M., Hu, Q., Fuller, B., Testuggine, D., and Khabsa, M. Llama guard: Llm-based input-output safeguard for human-ai conversations. *CoRR*, abs/2312.06674, 2023. doi: 10.48550/ARXIV.2312.06674. URL https://doi.org/10.48550/arXiv.2312.06674.

Jacobs, M., Pradier, M. F., McCoy Jr, T. H., Perlis, R. H., Doshi-Velez, F., and Gajos, K. Z. How machine-learning recommendations influence clinician treatment selections: the example of antidepressant selection. *Translational psychiatry*, 11(1):108, 2021.

Ji, J., Liu, M., Dai, J., Pan, X., Zhang, C., Bian, C., Chen, B., Sun, R., Wang, Y., and Yang, Y. Beavertails: Towards improved safety alignment of LLM via a human-preference dataset. In *NeurIPS*, 2023.

Jitkrittum, W., Gupta, N., Menon, A. K., Narasimhan, H., Rawat, A. S., and Kumar, S. When does confidence-based cascade deferral suffice? In *NeurIPS*, 2023.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention. In *SOSP*, 2023.

Lin, Z., Wang, Z., Tong, Y., Wang, Y., Guo, Y., Wang, Y., and Shang, J. Toxicchat: Unveiling hidden challenges of toxicity detection in real-world user-ai conversation. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Liu, H., Lai, V., and Tan, C. Understanding the effect of out-of-distribution examples and interactive explanations on human-ai decision making. *Proc. ACM Hum. Comput. Interact.*, 5(CSCW2):408:1–408:45, 2021.

Liu, H., Huang, H., Gu, X., Wang, H., and Wang, Y. On calibration of llm-based guard models for reliable content moderation. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wUbum0nd9N.

Madras, D., Pitassi, T., and Zemel, R. S. Predict responsibly: Improving fairness and accuracy by learning to defer. In *NeurIPS*, pp. 6150–6160, 2018.

Markov, T., Zhang, C., Agarwal, S., Eloundou Nekoul, F., Lee, T., Adler, S., Jiang, A., and Weng, L. A holistic approach to undesired content detection in the real world. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 15009–15018, 2023. doi: 10.1609/aaai.v37i12.26752.

Mazeika, M., Phan, L., Yin, X., Zou, A., Wang, Z., Mu, N., Sakhaee, E., Li, N., Basart, S., Li, B., Forsyth, D., and Hendrycks, D. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal, 2024.

Mozannar, H. and Sontag, D. A. Consistent estimators for learning to defer to an expert. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7076–7087. PMLR, 2020.

Mozannar, H., Satyanarayan, A., and Sontag, D. A. Teaching humans when to defer to a classifier via exemplars. In *AAAI*, pp. 5323–5331. AAAI Press, 2022.

Mozannar, H., Lang, H., Wei, D., Sattigeri, P., Das, S., and Sontag, D. A. Who should predict? exact algorithms for learning to defer to humans. In *AISTATS*, volume 206 of *Proceedings of Machine Learning Research*, pp. 10520–10545. PMLR, 2023.

Nagireddy, M., Padhi, I., Ghosh, S., and Sattigeri, P. When in doubt, cascade: Towards building efficient and capable guardrails. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 8(2):1812–1821, Oct. 2025. doi: 10.1609/aies.v8i2.36676. URL https://ojs.aaai.org/index.php/AIES/article/view/36676.

Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pp. 625–632, 2005.

OpenAI. Openai moderation api, 2023. URL https://platform.openai.com/docs/guides/moderation/quickstart.

Ouyang, L., Wu, J., Jiang, X., et al. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

Padhi, I., Nagireddy, M., Cornacchia, G., Chaudhury, S., Pedapati, T., Dognin, P. L., Murugesan, K., Miehling, E., Cooper, M. S., Fraser, K., Zizzo, G., Hameed, M. Z., Purcell, M., Desmond, M., Pan, Q., Ashktorab, Z., Vejsbjerg, I., Daly, E. M., Hind, M., Geyer, W., Rawat, A., Varshney, K. R., and Sattigeri, P. Granite guardian. *CoRR*, abs/2412.07724, 2024.

Padhi, I., Nagireddy, M., Cornacchia, G., Chaudhury, S., Pedapati, T., Dognin, P. L., Murugesan, K., Miehling, E., Cooper, M. S., Fraser, K., Zizzo, G., Hameed, M. Z., Purcell, M., Desmond, M., Pan, Q., Vejsbjerg, I., Daly, E. M., Hind, M., Geyer, W., Rawat, A., Varshney, K. R., and Sattigeri, P. Granite guardian: Comprehensive LLM safeguarding. In *NAACL (Industry Track)*, pp. 607–615. Association for Computational Linguistics, 2025.

Perez, E., Huang, S., Song, F., Cai, T., Ring, R., Aslanides, J., Glaese, A., McAleese, N., and Irving, G. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

Platt, J. C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, pp. 61–74, 1999.

Rawat, A., Schoepf, S., Zizzo, G., Cornacchia, G., Hameed, M. Z., Fraser, K., Miehling, E., Buesser, B., Daly, E. M., Purcell, M., Sattigeri, P., Chen, P.-Y., and Varshney, K. R. Attack atlas: A practitioner's perspective on challenges and pitfalls in red teaming GenAI. *arXiv preprint arXiv:2409.15398*, 2024.

Rebedea, T., Dinu, R., Sreedhar, M. N., Parisien, C., and Cohen, J. NeMo guardrails: A toolkit for controllable and safe LLM applications with programmable rails. In Feng, Y. and Lefever, E. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 431–445, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-demo.40. URL https://aclanthology.org/2023.emnlp-demo.40/.

Röttger, P., Kirk, H. R., Vidgen, B., Attanasio, G., Bianchi, F., and Hovy, D. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. In *NAACL*, 2024.

Schoepf, S., Hameed, M. Z., Rawat, A., Fraser, K., Zizzo, G., Cornacchia, G., and Purcell, M. MAD-MAX: Modular and diverse malicious attack mixtures for automated LLM red teaming. In *ICML 2025 Workshop on Data in Generative Models, Benchmarking, Understanding, and Governance*, 2025. URL https://openreview.net/forum?id=.

Teerapittayanon, S., McDanel, B., and Kung, H. T. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 2464–2469. IEEE, 2016. doi: 10.1109/ICPR.2016.7900006.

Vidgen, B., Scherrer, N., Kirk, H. R., Qian, R., Kannappan, A., Hale, S. A., and Röttger, P. Simplesafetytests: a test suite for identifying critical safety risks in large language models, 2024. URL https://arxiv.org/abs/2311.08370.

Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., et al. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*, 2021.

Xin, J., Tang, R., Lee, J., Yu, Y., and Lin, J. Deebert: Dynamic early exiting for accelerating bert inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2246–2251. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.204.

Zadrozny, B. and Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 694–699, 2002. doi: 10.1145/775047.775151.

Zeng, W., Liu, Y., Mullins, R., Peran, L., Fernandez, J., Harkous, H., Narasimhan, K., Proud, D., Kumar, P., Radharapu, B., Sturman, O., and Wahltinez, O. Shieldgemma: Generative ai content moderation based on gemma. *arXiv preprint arXiv:2407.21772*, 2024.

Zhang, B., Gao, C., Yang, L., Han, B., Hu, M., Luo, Z., Geng, G., Bai, X., Zhang, J., Yao, W., and Wang, Z. SafeConf: A confidence-calibrated safety self-evaluation method for large language models. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 3483–3495, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.186. URL https://aclanthology.org/2025.findings-emnlp.186/.

Zizzo, G., Cornacchia, G., Fraser, K., Hameed, M. Z., Rawat, A., Buesser, B., Purcell, M., Chen, P.-Y., Sattigeri, P., and Varshney, K. R. Adversarial prompt evaluation: Systematic benchmarking of guardrails against prompt input attacks on LLMs. In *NeurIPS Safe Generative AI Workshop*, 2024.

A. Proofs and Theoretical Details

We provide detailed derivations for the propositions stated in Section 3.

A.1. Proof of Proposition 3.1 (Bayes-Optimal Deferral)

Proof. The population risk of a deferral rule r is

$$\begin{aligned} R(r) &= \mathbb{E}_{x,y}\big[(1-r(x))\cdot \mathbb{I}[y \neq \hat{y}_l(x)] + r(x)\cdot \mathbb{I}[y \neq \hat{y}_h(x)] + c\, r(x)\big] \\ &= \mathbb{E}_x\big[\mathrm{err}_l(x) - r(x)\cdot\alpha(x)\big] + \mathrm{const}, \end{aligned} \tag{15}$$

where $\mathrm{err}_i(x) = P(y \neq \hat{y}_i(x) \mid x)$ is the expected error of guardrail G_i on input x, $\Delta(x) = \mathrm{err}_l(x) - \mathrm{err}_h(x)$ is the expected error reduction, and $\alpha(x) = \Delta(x) - c$ is the net benefit of deferral. The first term does not depend on r. To minimise $R(r)$, we maximise $r(x)\cdot\alpha(x)$ pointwise. Since $r(x) \in \{0,1\}$, the optimal choice is $r^*(x) = \mathbb{I}[\alpha(x) > 0]$, which yields $r^*(x) = \mathbb{I}[\Delta(x) > c]$. □

A.2. Proof of Proposition 3.2 (Excess Risk Decomposition)

Proof. From the derivation above, $R(r) = \mathbb{E}_x[\mathrm{err}_l(x) - r(x)\cdot\alpha(x)]$ and $R(r^*) = \mathbb{E}_x[\mathrm{err}_l(x) - r^*(x)\cdot\alpha(x)]$. Subtracting:

$$\begin{aligned} R(r) - R(r^*) &= \mathbb{E}_x\big[(r^*(x) - r(x))\cdot\alpha(x)\big] \\ &= \mathbb{E}_x\big[(\mathbb{I}[\alpha(x){>}0] - \mathbb{I}[r(x){=}1])\cdot\alpha(x)\big]. \end{aligned} \tag{16}$$

This quantity is non-negative, since: when $r(x)$ and $r^*(x)$ agree, the contribution is zero; when $\alpha(x) > 0$ but $r(x) = 0$ (missed deferral), the contribution is $\alpha(x) > 0$; when $\alpha(x) \leq 0$ but $r(x) = 1$ (unnecessary deferral), the contribution is $-\alpha(x) \geq 0$. Equivalently, $R(r) - R(r^*) = \mathbb{E}_x\big[|\alpha(x)|\cdot\mathbb{I}\{r(x) \neq r^*(x)\}\big]$. □

A.3. Confidence-Based Deferral as a Plug-in Approximation

Proof. The Bayes-optimal deferral rule from Proposition 3.1 defers on an input x whenever the expected reduction in classification error exceeds the deferral cost:

$$r^*(x) = \mathbb{I}[\Delta(x) > c],$$

where

$$\Delta(x) = \Pr(Y \neq \hat{y}_l(X) \mid X = x) - \Pr(Y \neq \hat{y}_h(X) \mid X = x).$$

Equivalently, defining $\alpha(x) = \Delta(x) - c$, the optimal rule defers exactly when $\alpha(x) > 0$.

In practice, $\Delta(x)$ is unknown because it depends on the conditional errors of both the lightweight and heavy guardrails. The confidence-based rule

$$\hat{r}_\tau(x) = \mathbb{I}[s_l(x) < \tau]$$

therefore acts as a plug-in approximation to r^*, using the calibrated routing score $s_l(x)$ as a proxy for the benefit of deferral. This approximation is accurate when the ordering induced by low $s_l(x)$ aligns with the ordering induced by large $\Delta(x)$. In other words, inputs assigned lower routing confidence by G_l should be precisely those for which deferring to G_h provides larger expected improvement.

A sufficient condition for this alignment is that the heavy model's conditional error varies slowly across inputs relative to the lightweight model's conditional error. Writing

$$\mathrm{err}_i(x) = \Pr(Y \neq \hat{y}_i(X) \mid X = x),$$

we have

$$\Delta(x) = \mathrm{err}_l(x) - \mathrm{err}_h(x).$$

If $\mathrm{err}_h(x)$ is approximately constant, or more generally does not increase with $\mathrm{err}_l(x)$ fast enough to offset changes in $\mathrm{err}_l(x)$, then $\Delta(x)$ is larger on inputs where G_l is more likely to be wrong. In that case, thresholding low $s_l(x)$ approximates thresholding high $\Delta(x)$.

Thus, confidence-based deferral is not optimal merely because a larger model is more accurate on average. It is effective when the lightweight model's calibrated routing confidence ranks examples in a way that is aligned with the conditional benefit of deferral. □

A.4. Proof of Proposition 3.3 (Scalar Confidence Calibration Preserves Ranking)

Proof. Let $u_l : X \to [1/2, 1]$ denote the uncalibrated scalar confidence assigned by the lightweight model to its own hard prediction, and let

$$s_l(x) = f(u_l(x))$$

be the calibrated routing score, where f is monotone non-decreasing.

For any two inputs x_i, x_j, if

$$u_l(x_i) \le u_l(x_j),$$

then monotonicity of f implies

$$s_l(x_i) = f(u_l(x_i)) \le f(u_l(x_j)) = s_l(x_j).$$

Therefore, the calibrated score preserves the ordering induced by the raw routing confidence.

If f is strictly increasing, then thresholding s_l induces the same acceptance sets as thresholding u_l, up to a reparameterization of the threshold. Specifically, for any calibrated threshold τ, there exists an uncalibrated threshold $\tau' = f^{-1}(\tau)$ such that

$$\{x : s_l(x) \ge \tau\} = \{x : u_l(x) \ge \tau'\}.$$

If f is monotone non-decreasing but not strictly increasing, then flat regions of f may map distinct raw confidence values to the same calibrated score. In this case, the ordering is preserved up to ties, and finite-sample differences may arise because of tie-breaking or discrete threshold grids.

Isotonic regression, temperature scaling with $T > 0$, and Platt scaling with nonnegative slope are monotone non-decreasing transformations of the scalar confidence score. Hence, in the setting considered here, these calibrators improve the reliability and interpretability of routing thresholds without introducing new ranking information. □

A.5. Extension to K-Model Cascades

For K guardrails with standalone inference costs $c^{(1)} \le \cdots \le c^{(K)}$, the Bayes-optimal selector (which picks one guardrail per input) generalises Proposition 3.1 (Jitkrittum et al., 2023):

$$s^*(x) = \arg\min_{k \in [K]} \big(1 - \eta_{\hat{y}_k(x)}(x) + c^{(k)}\big). \qquad (17)$$

This selects the guardrail that minimises the sum of misclassification risk and standalone invocation cost. In a *sequential* cascade, however, the cost to reach G_k is cumulative: every input pays at least $c^{(1)}$; inputs routed to the second stage pay $c^{(1)} + c^{(2)}$; more generally, inputs reaching the k-th stage pay $\sum_{j \le k} c^{(j)}$. In a sequential multi-stage cascade, however, the cost to reach G_k is cumulative: every input pays at least $c^{(1)}$; inputs routed to the second stage pay $c^{(1)} + c^{(2)}$; more generally, inputs reaching the k-th stage pay $\sum_{j \le k} c^{(j)}$. Extending calibrated routing to this sequential multi-stage setting is left for future work.

B. Full Benchmark Results

Table 5 presents detection and calibration results for all 21 model variants evaluated in this study. The 18 models from the four core families (Granite Guardian, Llama Guard, ShieldGemma, GuardReasoner) are identical to those reported in the main paper (Table 1); this table additionally includes Qwen3Guard and Nemotron, and reports the learned temperature T for each model as a diagnostic of overconfidence.

Figure 6 shows reliability diagrams for the model variants not included in the main paper.

Figure 7 shows cross-family coverage-risk trade-off.

C. Dataset

Table 6 presents statistics of 10 public harm-detection benchmarks covering prompt and response harmlessness

Table 6. Benchmark datasets. Full descriptions in Appendix E.

Dataset	Task	N	%Pos
ToxicChat	Prompt harm	1,427	7.2
OpenAI Mod.	Prompt harm	1,680	42.3
AegisSafetyTest	Prompt harm	363	50.1
SimpleSafetyTests	Prompt harm	100	100
HarmBench	Prompt harm	320	100
BeaverTails	Response harm	3,021	47.8
SafeRLHF	Response harm	2,000	50.0
XSTest-RR	Response harm	449	50.0
XSTest-RH	Response harm	446	50.0
XSTest-RR(h)	Response harm	200	50.0

Table 5. Full detection and calibration results across all 21 model variants evaluated in this study. The main paper (Table 1) reports the same 18 models from the four core families (Granite Guardian, Llama Guard, ShieldGemma, GuardReasoner); this table additionally includes Qwen3Guard and Nemotron, and reports the learned temperature T for each model (higher T indicates greater overconfidence). ECE in %.

Model	AUC	F1	ECE_{unc}	ECE_{iso}	ECE_{sig}	ECE_{temp}	T
GG-3.0-2B	0.79	0.68	10.6	1.1	4.3	2.9	2.4
GG-3.0-8B	0.87	0.76	6.8	2.4	4.4	2.5	1.7
GG-3.1-2B	0.86	0.75	6.2	2.3	4.1	4.5	1.8
GG-3.1-8B	0.90	0.80	6.7	1.2	2.4	3.5	2.0
GG-3.2-3B	0.85	0.75	8.5	1.2	2.9	3.7	2.2
GG-3.2-5B	0.88	0.78	9.9	1.7	3.5	3.5	1.8
GG-3.3-8B	0.89	0.80	13.9	1.8	3.1	2.1	4.7
GG-4.1-8B	0.90	0.80	12.7	1.9	1.8	2.3	3.7
LG-7B	0.83	0.66	14.2	3.1	3.9	2.5	2.9
LG-2-8B	0.85	0.74	15.4	2.6	3.0	3.0	4.6
LG-3-1B	0.80	0.65	17.0	1.4	4.6	2.0	3.7
LG-3-8B	0.83	0.72	14.8	2.5	2.5	4.7	4.1
SG-2B	0.74	0.39	26.5	2.2	5.6	6.5	5.8
SG-9B	0.73	0.38	26.4	2.5	5.8	7.1	6.6
SG-27B	0.76	0.41	27.5	2.2	2.7	4.9	6.1
GR-1B	0.86	0.84	17.2	1.7	1.1	0.7	8.3
GR-3B	0.83	0.83	18.3	1.9	1.5	1.9	10.0
GR-8B	0.82	0.82	18.7	1.0	0.4	1.9	10.0
QG-0.6B	0.90	0.81	11.8	1.6	3.9	3.2	3.2
QG-8B	0.90	0.84	11.3	1.6	6.3	4.5	3.7
Nemotron-8B	0.77	0.72	17.7	1.2	3.8	13.8	5.5

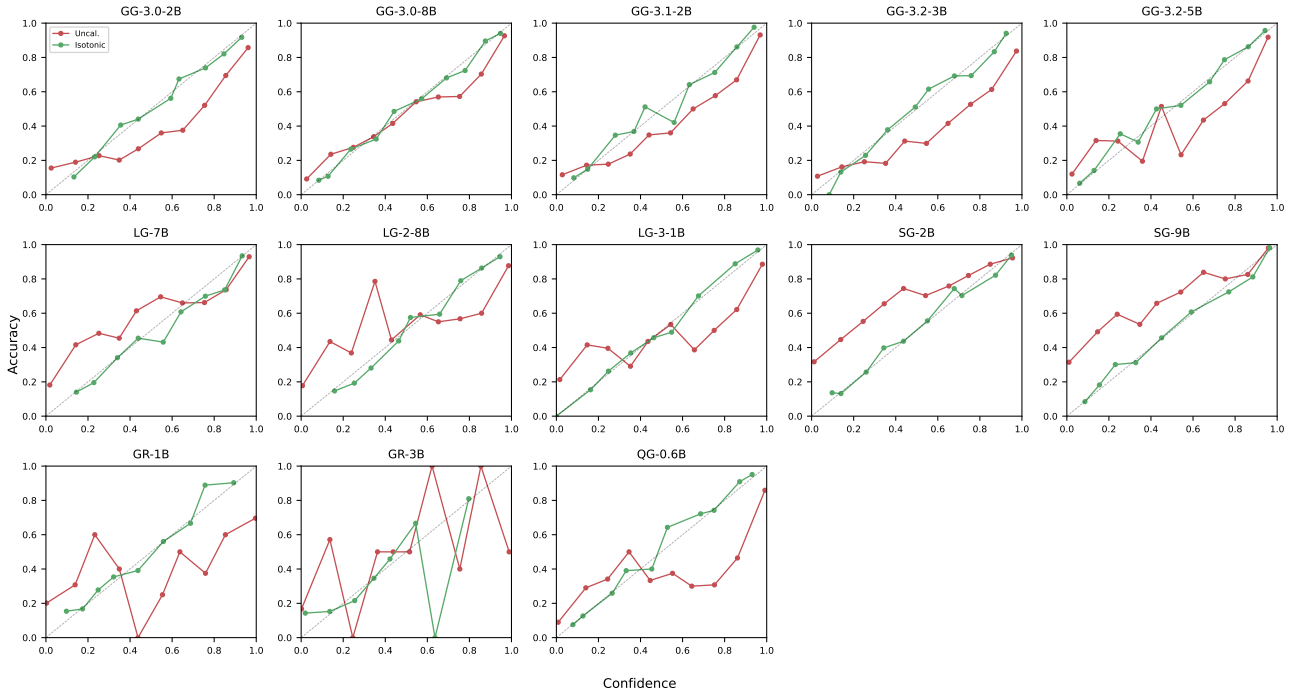


Figure 6. Reliability diagrams for all remaining model variants (smaller models within each family). Red curves show uncalibrated probabilities; green curves show isotonic-calibrated probabilities. The dashed diagonal indicates perfect calibration.

D. Per-Family Deferral Analysis

Figure 8 presents the full four-panel deferral analysis (Accuracy, F1, Threshold, ECE vs. Coverage) for each of the five

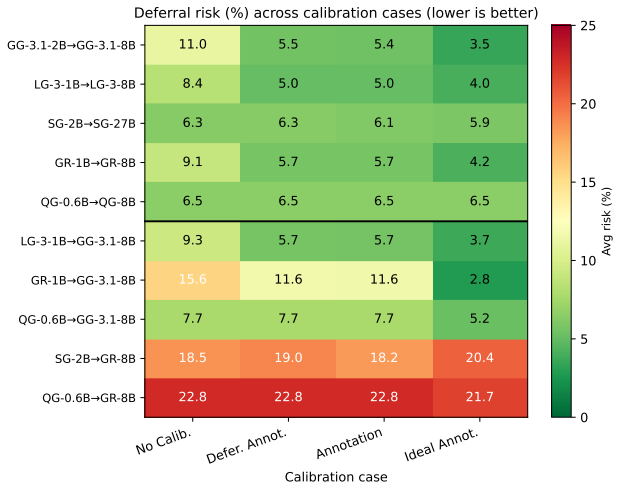


Figure 7. Full deferral risk heatmap. Rows above the black separator are within-family cascades; rows below are cross-family cascades. Risk is the average 1−Accuracy across coverage levels; lower (green) is better. Cross-family deferral to GG-3.1-8B (LG-3-1B, GR-1B, QG-0.6B) consistently achieves the lowest residual risk under the Ideal Annotation case.

model families under four calibration cases. The main paper (Figure 2) shows the detailed ablation for Granite Guardian; Figure 3 shows the unified comparison across families.

Granite Guardian (Figure 8a). Both Accuracy and F1 improve steadily as coverage decreases (i.e., more inputs are deferred to G_h). The "Ideal (Both GT)" case achieves the lowest ECE across all coverage levels, confirming that calibrating both models on ground truth provides the best routing signal. The threshold curve declines smoothly, indicating well-separated confidence distributions.

Llama Guard (Figure 8b). The "No Calibration" baseline shows limited improvement with deferral, as the uncalibrated confidence scores provide poor routing signal. Calibration on ground truth substantially improves F1, particularly at intermediate coverage levels (50 to 70%). The threshold curve is steeper than Granite Guardian, reflecting less reliable confidence estimates.

GuardReasoner (Figure 8c). GuardReasoner exhibits a distinctive pattern: the small model (1B) already achieves high F1 (0.84), so deferral to the 8B model provides modest accuracy gains. However, calibration significantly reduces ECE, improving the reliability of the confidence estimates even when the accuracy improvement is small.

ShieldGemma (Figure 8d). The accuracy gap between SG-2B and SG-27B is negligible (0.001), rendering deferral ineffective regardless of calibration strategy. All four cases produce nearly identical curves, confirming that deferral requires a meaningful performance gap between G_l and G_h.

Qwen3Guard (Figure 8e). The within-family gap is small (0.014), so deferral provides limited benefit. However, QG-0.6B achieves the highest AUC among all models (0.91), making it an excellent candidate for cross-family deferral to stronger models such as GG-3.1-8B (see Section 6).

E. Dataset Details

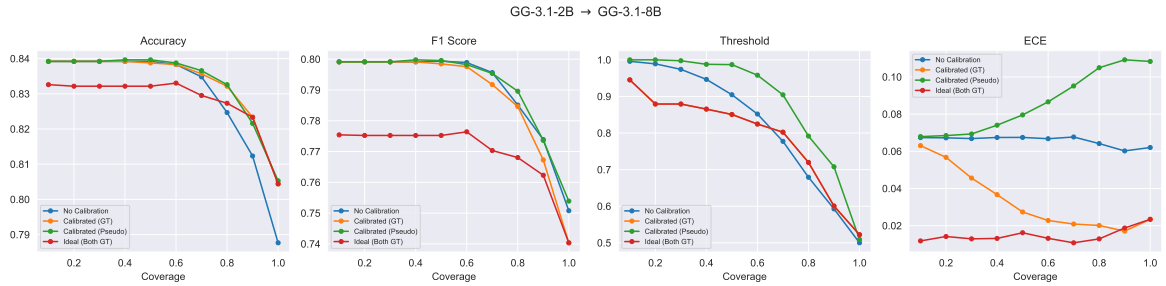
We evaluate on out-of-distribution (OOD) harm detection. All datasets are publicly available. Following the evaluation protocol of Padhi et al. (2024), we assign a positive (harmful) label to any instance marked as unsafe under its original taxonomy.

From raw datasets to evaluation subsets. The sample counts listed below (e.g., ToxicChat 2,853; SafeRLHF 1,000; TRUE datasets) are the raw sizes of each source benchmark. The counts reported in Table 6 of the main paper are the filtered evaluation subsets used after preprocessing, i.e., (i) keeping only human-annotated entries where multiple annotation types exist, (ii) removing instances with missing labels or invalid model outputs, (iii) class-balancing within each benchmark when applicable, and (iv) sub-sampling the largest benchmarks to avoid overweighting them in the aggregate. The 80/20 calibration-evaluation split (Section E, last subsection) is applied to these filtered subsets.

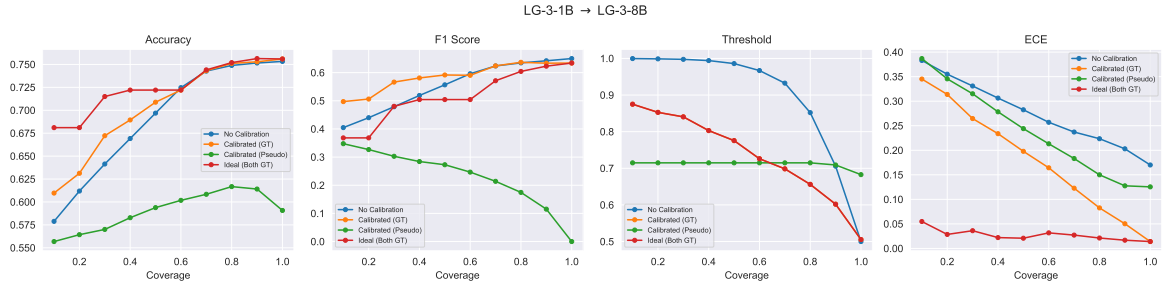
E.1. Harm Detection Benchmarks

Prompt Harmfulness.

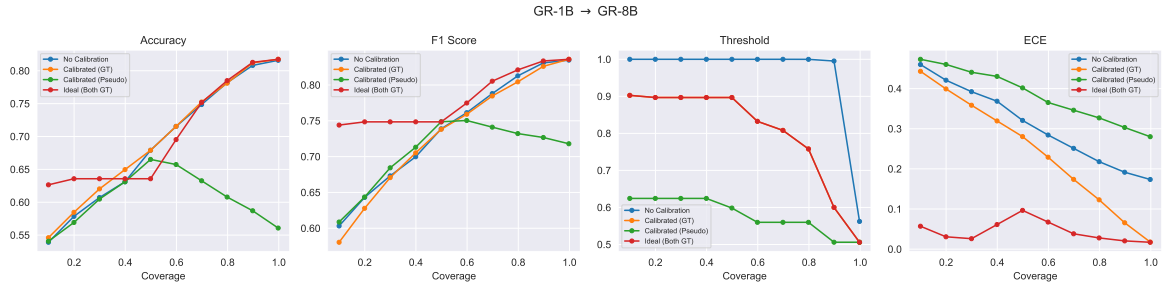
- **ToxicChat** (Lin et al., 2023): 2,853 human-annotated samples from real user-chatbot interactions. We use version 0124 and select only human-annotated samples, assigning the harmful label if either the toxicity or jailbreak flag is positive.
- **OpenAI Moderation** (Markov et al., 2023): 1,680 prompts labeled across eight safety categories (sexual, hate, violence, harassment, self-harm, sexual/minors, hate/threatening, violence/graphic).
- **AegisSafetyTest** (Ghosh et al., 2024): 359 prompt-only entries from Anthropic's HH-RLHF, manually annotated across 13 risk categories.
- **SimpleSafetyTests** (Vidgen et al., 2024): 100 manually crafted harmful prompts targeting sensitive topics including child abuse, suicide, self-harm, and fraud.
- **HarmBench Prompt** (Mazeika et al., 2024): 239 harmful prompts designed to test robustness against



(a) Granite Guardian: G_l = `GG-3.1-2B`, G_h = `GG-3.1-8B`. Accuracy gap: 4.2%.



(b) Llama Guard: G_l = `LG-3-1B`, G_h = `LG-3-8B`. Accuracy gap: 4.5%.



(c) GuardReasoner: G_l = `GR-1B`, G_h = `GR-8B`. Accuracy gap: 0.4%.

(d) ShieldGemma: G_l = `SG-2B`, G_h = `SG-27B`. Accuracy gap: 0.1%.

(e) Qwen3Guard: G_l = `QG-0.6B`, G_h = `QG-8B`. Accuracy gap: 1.4%.

Figure 8. Per-family deferral analysis. Each row shows four panels: Accuracy, F1, Threshold, and ECE as a function of coverage under four calibration cases (No Calibration, Calibrated on GT, Calibrated on Pseudo-labels, Ideal with Both GT). Deferral is most effective when the accuracy gap between G_l and G_h is large (Granite Guardian, Llama Guard) and least effective when the gap is negligible (ShieldGemma).

jailbreak attacks.

Response Harmfulness.

- **BeaverTails** (Ji et al., 2023): 3,021 prompt-response pairs from the test set, focusing on response harmfulness with responses generated by Alpaca-7B.
- **SafeRLHF** (Dai et al., 2024): 1,000 prompt-response pairs with manually annotated preference comparisons between safe and unsafe responses.
- **XSTest** (Röttger et al., 2024): Three variants testing the boundary between safe and unsafe content. XSTest-RH (446 samples) tests refusal of harmless prompts; XSTest-RR (449 samples) and XSTest-RR(h) (200 samples) test compliance with risky prompts.

E.2. Data Splitting Protocol

All datasets are split 80/20 into calibration and evaluation partitions. The split is performed at the instance level with a fixed random seed (42) to ensure reproducibility. The calibration partition is used exclusively for fitting post-hoc calibrators (isotonic regression, Platt scaling, or temperature scaling). All metrics reported in the paper are computed on the held-out 20% evaluation partition. This protocol ensures that calibration and evaluation are performed on disjoint data, preventing information leakage.

E.3. Aggregate Deferral Risk

The heatmaps in Figures 4 and 7 summarise cascade performance with a single scalar we refer to as the *aggregate deferral risk*. For a cascade (G_l, G_h) with a confidence-based deferral rule, let $\mathrm{Acc}(\phi)$ denote the cascade's accuracy when a fraction $\phi \in [0, 1]$ of inputs is handled by G_l (the rest is deferred to G_h). The aggregate risk is the mean error rate across a coverage grid $\Phi = \{0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$:

$$\overline{\mathrm{Risk}}(G_l, G_h) = \frac{1}{|\Phi|} \sum_{\phi \in \Phi} \big(1 - \mathrm{Acc}(\phi)\big). \tag{18}$$

This is a threshold-free scalar summary: each coverage level gives a different operating trade-off between cost and accuracy, and $\overline{\mathrm{Risk}}$ averages the error rate across these operating points. It is analogous to AURRRC but without the cost axis; lower is better.

F. Calibration Methods: Extended Discussion

F.1. Isotonic Regression

Isotonic regression (Zadrozny & Elkan, 2002; Niculescu-Mizil & Caruana, 2005) fits a non-parametric, monotone piecewise-constant function f by solving:

$$\min_{f \in \mathcal{F}_{\text{iso}}} \sum_{j=1}^{n} \big(f(\hat{p}(x_j)) - y_j\big)^2 \quad \text{subject to } f \text{ non-decreasing}, \tag{19}$$

where $\mathcal{F}_{\text{iso}}$ is the class of non-decreasing step functions. The solution is computed efficiently via the pool adjacent violators algorithm in $O(n \log n)$ time. For out-of-sample predictions, values outside the training range are clipped to the nearest boundary value.

Strengths: Can capture arbitrary monotone probability mappings; achieves the lowest ECE in most experiments. **Weaknesses**: May overfit with small calibration sets (<500 samples); the step-function output can produce ties in confidence scores.

F.2. Platt Scaling (Sigmoid Calibration)

Platt scaling (Platt, 1999) fits a two-parameter logistic function:

$$f(z) = \sigma(az + b) = \frac{1}{1 + \exp(-(az + b))}, \tag{20}$$

where $a, b \in \mathbb{R}$ are learned by minimizing the negative log-likelihood on the calibration set. The optimization is convex and solved via L-BFGS.

Strengths: Robust with small calibration sets; smooth output probabilities. **Weaknesses**: Assumes a sigmoidal relationship between raw and calibrated probabilities, which may not hold for all model architectures.

F.3. Temperature Scaling

Temperature scaling (Guo et al., 2017) is a special case of Platt scaling with $a = 1/T$ and $b = 0$:

$$f(z) = \sigma\big(\mathrm{logit}(z)/T\big), \quad \text{where } \mathrm{logit}(z) = \log \frac{z}{1 - z}. \tag{21}$$

The single parameter $T > 0$ is learned by minimizing the negative log-likelihood. Values $T > 1$ indicate overconfidence (the model's raw probabilities are too extreme); $T < 1$ indicates underconfidence.

Strengths: Preserves the ranking of predictions (and thus AUC and the deferral curve per Proposition 3.3); single parameter, minimal overfitting risk. **Weaknesses**: Cannot correct non-linear miscalibration patterns.

F.4. Empirical Comparison

In our experiments, the learned temperature values reveal the degree of overconfidence across model families:

- **Granite Guardian**: $T \approx 2$ to 5, indicating moderate overconfidence.

- **Llama Guard**: $T \approx 3$ to 5, similar to Granite Guardian.
- **ShieldGemma**: $T \approx 6$ to 7, indicating substantial overconfidence consistent with the high uncalibrated ECE (26 to 28%).
- **GuardReasoner**: $T \approx 8$ to 10, the most overconfident family. This is consistent with the reasoning-based architecture, which may produce highly peaked probability distributions.
- **Qwen3Guard**: $T \approx 3$ to 4, moderate overconfidence.
- **Nemotron**: $T \approx 5.5$, but temperature scaling fails to reduce ECE effectively (13.8% after scaling vs. 17.7% uncalibrated), suggesting non-sigmoidal miscalibration.

G. Calibration Data Size Ablation

A practical question for deployment is: how much calibration data is needed? We vary the calibration/test split from 50/50 to 90/10 across five model families and two calibration methods.

Key findings:

- **Isotonic regression** shows a clear downward trend in ECE as the calibration fraction increases, particularly for Granite Guardian (ECE drops from 2.8% at 50% calibration to 1.2% at 90%). This confirms that the non-parametric method benefits from more data.
- **Sigmoid scaling** is remarkably stable, with ECE varying by less than 0.5 percentage points across all split ratios. This makes it the safer choice when calibration data is limited.
- The **overfitting gap** (difference between calibration-set ECE and test-set ECE) is largest for isotonic regression at small calibration fractions, confirming the overfitting risk.
- For practical deployment, we recommend at least 2,000 calibration samples for isotonic regression and 500 for sigmoid scaling.

H. Deferral Threshold Selection

The deferral threshold τ in Equation (7) controls the coverage-accuracy trade-off. We recommend the following deployment procedure:

1. **Calibrate** the small model G_l on a held-out calibration set using isotonic regression (or sigmoid scaling if the calibration set is small).
2. **Compute the coverage-accuracy curve** by sweeping τ from 0.5 to 1.0 on the calibration set.
3. **Select** τ to achieve the desired coverage target. For example, $\tau = 0.85$ typically yields 70 to 80% coverage for well-calibrated Granite Guardian models.
4. **Monitor** the deferral rate in production. If the rate deviates significantly from the calibration-set estimate, this indicates distribution shift and the threshold should be re-tuned.

The threshold plots in Figure 8 show that well-calibrated models (Granite Guardian, GuardReasoner) exhibit smooth, monotonically decreasing thresholds as coverage increases, making threshold selection straightforward. Poorly calibrated models (ShieldGemma) show noisy threshold curves, indicating that the confidence scores are less reliable for deferral decisions.

I. Model Details and Probability Extraction

Table 7. Guardrail models evaluated in this study.

Family	Model	Params	Prob. Method
Granite Guardian	GG-3.0-2B	2B	Top-k logprob (Yes/No)
	GG-3.0-8B	8B	Top-k logprob (Yes/No)
	GG-3.1-2B	2B	Top-k logprob (Yes/No)
	GG-3.1-8B	8B	Top-k logprob (Yes/No)
	GG-3.2-3B	3B	Top-k logprob (Yes/No)
	GG-3.2-5B	5B	Top-k logprob (Yes/No)
	GG-3.3-8B	8B	⟨score⟩ tag logprob
	GG-4.1-8B	8B	Guardian block logprob
Llama Guard	LG-7B	7B	First-token logit (safe/unsafe)
	LG-2-8B	8B	First-token logit
	LG-3-1B	1B	First-token logit
	LG-3-8B	8B	First-token logit
ShieldGemma	SG-2B	2B	Vocab logit (Yes/No)
	SG-9B	9B	Vocab logit
	SG-27B	27B	Vocab logit
GuardReasoner	GR-1B	1B	Decision-token logit
	GR-3B	3B	Decision-token logit
	GR-8B	8B	Decision-token logit
Qwen3Guard	QG-0.6B, 8B	0.6/8B	Safety-token logit
Nemotron	Nemotron-8B	8B	JSON-token logit

I.1. Probability Extraction

All models are evaluated using vLLM (Kwon et al., 2023) for efficient batched inference. The probability of risk (i.e., $\hat{p}(x) = P(y{=}1 \mid x)$) is extracted as follows:

Granite Guardian 3.0/3.1/3.2. Following Padhi et al. (2024), we compute the detection scores by aggregating the exponentiated log-likelihoods of all tokens containing the substring "Yes" or "No" within the top-k (k=20) tokens at the first generated position. The scores are normalized via softmax to produce the probability of risk.

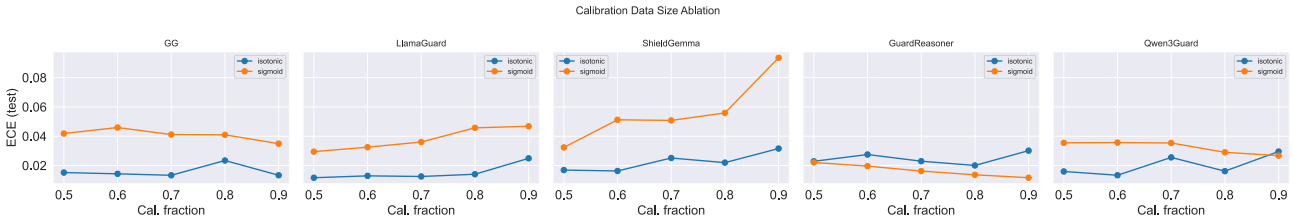


Figure 9. ECE on the test set as a function of calibration set fraction. Isotonic regression benefits more from additional data, while sigmoid scaling is stable across all split ratios.

Granite Guardian 3.3/4.1. These models output ⟨score⟩yes/no⟨/score⟩ tags. We extract logprobs at the token position immediately following the ⟨score⟩ tag and apply the same softmax normalization over Yes/No logprobs.

Llama Guard. The model generates "safe" or "unsafe" as the first token. We extract the logits for these two tokens at the last input position and apply softmax.

ShieldGemma. The model is prompted to answer "Yes" or "No". We extract the logits for the vocabulary tokens "Yes" and "No" at the last input position and apply softmax.

GuardReasoner. The model generates a reasoning chain followed by a label. We scan the generated tokens for the position where "harmful" or "unharmful" appears after "Request:" and extract logprobs at that position.

Qwen3Guard. The model generates "Safety: Safe/Unsafe/Controversial". We extract logprobs at the token position where the safety label appears and compute $P(\text{unsafe}) = P(\text{Unsafe}) + P(\text{Controversial})$.

Nemotron. The model generates a JSON response with "User Safety": "safe"/"unsafe". We extract logprobs at the token position where the safety value appears.

When neither the safe nor unsafe token appears in the top-k logprobs, we treat the missing token's logprob as -50 (effectively $-\infty$) and use the generated label to determine the direction. This ensures that highly confident predictions (where the alternative token falls outside top-k) receive probabilities close to 0 or 1 rather than the uninformative value of 0.5.