DATA-EFFICIENT CRITERIA STEERING FOR LLM JUDGES

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM judges are increasingly used to enforce safety and compliance policies, but adapting them to new languages, regulations, and risk taxonomies remains costly. Fine-tuning requires substantial labeled data and model updates, while prompt optimization relies on representative evaluation data and can overfit when data are scarce and the signal is limited, reducing held-out performance. We introduce criteria steering, a data-efficient, training-free method for specializing LLM judges from small labeled datasets without updating model weights. Our pipeline mines and ranks candidate criteria, constructs a contrastive activation direction from diverse criteria, and applies it only to the criterion-token span during prompt encoding, adding no decoding-time computation. On Bengali toxicity, Japanese content safety, and Indian financial-services safety and compliance (FinProof), criteria steering improves held-out performance for small general-purpose as well as purpose-built guardrail models. It outperforms text-only prompt optimization and, on FinProof, matches or exceeds a strong automated prompt-optimization reference depending on the operating point. Gains persist with relatively small labeled samples. These results position criteria steering as a practical adaptation mechanism for low-resource languages and specialized domains.

1 INTRODUCTION

LLM-as-a-judge is increasingly used to evaluate model outputs, filter training data, and guard deployed systems (Zheng et al., 2023; Gu et al., 2026). In guardrail models such as LlamaGuard (Inan et al., 2023), ShieldGemma (Zeng et al., 2024), WildGuard (Han et al., 2024), and Granite Guardian (Padhi et al., 2025), the verdict is conditioned on a natural-language criterion that specifies what should count as unsafe or non-compliant. That criterion is not universal: new languages, regulatory regimes, and risk taxonomies can require substantially different decision boundaries. This is especially consequential in low-resource and specialized settings, where broad English centric safety taxonomies can miss culturally grounded language phenomena or domain-specific regulatory violations (Tisha et al., 2026; Wang & Suresh, 2026; Mikkilineni et al., 2026).

The standard adaptation options each impose a different cost. Fine-tuning can produce specialized judges (Zhu et al., 2025; Kim et al., 2024), but it requires gradient-based training, task-specific parameters, and enough labeled data to support that training. Parameter-efficient methods such as LoRA reduce the parameter burden but still require per-task optimization (Hu et al., 2021). Prompt optimization avoids weight updates and has become increasingly capable, from ProTeGi and OPRO to DSPy, Promptbreeder, and GEPA (Pryzant et al., 2023; Yang et al., 2024; Khattab et al., 2024; Fernando et al., 2023; Agrawal et al., 2026). However, prompt optimization is only as reliable as the evaluation signal used to select prompts. Representative evaluation subsets materially affect prompt quality (Dong et al., 2025), and recent work shows that optimization can overfit an evaluator and weaken transfer across models and domains (Zhao et al., 2026). These limitations are most acute precisely when labeled data and domain coverage are scarce.

This motivates a third adaptation lever: change how the judge internally represents the criterion rather than changing the model weights or searching only over criterion wording. We introduce **criteria steering**, which mines candidate criteria from labeled data, ranks them by validation quality, constructs a contrastive activation direction from diverse criteria, and injects that direction only at the

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.

criterion-token span during prompt encoding. The intervention leaves the model weights unchanged and adds no decoding-time computation.

We study whether criteria steering can provide reliable, data-efficient specialization of small LLM judges across languages, model families, and policy domains. In particular, we ask:

1. **RQ1 (reliability):** Does criteria steering improve held-out judge performance across model families and domains, including settings where prompt optimization can regress?
2. **RQ2 (efficiency):** How much labeled data is needed for effective criteria steering, and can the method provide strong adaptation under limited supervision?
3. **RQ3 (generalization):** Does criteria steering remain competitive with strong prompt-optimization baselines and generalize to external benchmarks not used to develop the steering intervention?

We evaluate these questions on Bengali toxicity, Japanese content safety, and Indian financial-services safety and compliance using general-purpose and purpose-built guardrail models.

Unlike prior activation-steering work that targets broad behaviors such as refusal, truthfulness, or instruction following (Li et al., 2023; Arditi et al., 2024; Stolfo et al., 2025), our intervention is derived from criterion quality itself and localized to the criterion span. This makes the criterion, rather than a global behavioral trait, the unit of inference-time adaptation.

Contributions.

1. **Criteria steering.** We introduce a span-localized activation method for specializing LLM judges without weight updates or decoding-time intervention.
2. **Reliable specialization.** Across three domains and both general-purpose and guardrail models, criteria steering consistently improves held-out performance and outperforms text-only prompt optimization in most cases.
3. **Data-efficient adaptation of small judges.** Criteria steering delivers substantial gains from small labeled samples across target judges of 8B parameters or smaller, without weight updates.
4. **Robustness and design insights.** We characterize sensitivity to sample size, pole count, layer, and steering strength, and probe resulting behavioral changes.

2 Related Work

LLM judges and guardrails. LLM-as-a-judge enables scalable evaluation but inherits systematic biases and calibration failures (Zheng et al., 2023; Wang et al., 2024; Gu et al., 2026). Specialized judges are commonly produced by fine-tuning (Zhu et al., 2025; Kim et al., 2024) or by pairing a general language model with a fixed policy prompt (Inan et al., 2023; Zeng et al., 2024; Han et al., 2024; Padhi et al., 2025). We focus on the binary guardrail setting, where judge quality depends directly on how the policy criterion is represented.

Efficient adaptation and prompt optimization. Parameter-efficient fine-tuning reduces the storage and training burden of full-model adaptation (Hu et al., 2021), but still introduces a training stage and task-specific learned parameters. Prompt optimization instead adapts the instruction itself. ProTeGi (Pryzant et al., 2023), OPRO (Yang et al., 2024), DSPy (Khattab et al., 2024), Promptbreeder (Fernando et al., 2023), and GEPA (Agrawal et al., 2026) demonstrate increasingly sophisticated searches over prompt text. This flexibility comes with a dependence on the examples used to score candidate prompts. Representative subset selection affects the quality and stability of optimized prompts (Dong et al., 2025), while evaluator-specific optimization can overfit and weaken transfer (Zhao et al., 2026). Our method uses the same kind of labeled supervision to rank criteria, but converts criterion quality into an activation-space intervention rather than deploying the optimized text itself.

Activation steering and representation engineering. Adding a direction to model activations can change behavior without gradient updates (Turner et al., 2024; Zou et al., 2023). Contrastive Activation Addition (CAA) constructs such a direction from differences between paired activations (Rimsky et al., 2024). Prior work has steered truthfulness (Li et al., 2023), refusal (Arditi et al., 2024),

instruction following (Stolfo et al., 2025), simulated user traits (He et al., 2026), and low-resource generation (Cegin et al., 2026). Recent work further frames activation steering as a general inference-time control layer (Bai et al., 2026; Ostermann et al., 2026). Our setting differs in both the source and location of the intervention. The direction is derived from ranked judging criteria and is applied only while the model encodes the criterion span. This distinction matters because activation interventions need not be reproducible by prompt edits alone (Mishra et al., 2026).

Multilingual and domain-specific moderation. Safety criteria are particularly difficult to transfer across languages and specialized domains. Bangla hate-speech systems degrade on implicit, culturally grounded, and emoji-rich content (Tisha et al., 2026); Indic NLP resources remain fragmented (Banerjee & Varshney, 2026); and model outputs can shift with language-specific institutional or normative defaults (Wang & Suresh, 2026; Muthugala et al., 2026). FinProof similarly targets financial-services attacks tied to domain-specific regulatory obligations (Mikkilineni et al., 2026). These settings motivate adaptation methods that can use limited in-domain supervision without requiring a new specialist checkpoint for every language or policy taxonomy.

3 METHOD

Criteria steering uses labeled examples to identify stronger and weaker judging criteria and converts their activation contrast into a steering direction. At inference, the selected direction is injected only while the judge encodes the criterion tokens (Algorithm 1).

Algorithm 1 Criteria steering: direction construction and inference

Require: target judge f_θ; $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{val}}$; minimal criterion c_0; sample size N; pole size k

Phase 1: Construct and select the direction

$\mathcal{S} \leftarrow$ balanced N-example sample from $\mathcal{D}_{\text{train}}$

Score $\mathcal{S}$ with c_0 and split by $\{\text{correct/confident, wrong/uncertain}\} \times \{\text{positive, negative}\}$ ▷ $|s - 0.5| \geq 0.2$ is confident

for each confidence group containing both labels **do**

    Generate criteria distinguishing its positive and negative examples

end for

If no group yields candidates, generate from the full sample $\mathcal{S}$

Rank candidates by AUROC on $\mathcal{D}_{\text{val}}$

Select top-k as STRONG and bottom-k as WEAK; if weakly separated, generate more candidates or reduce k

Optional composed variant: merge each pole into one criterion before activation extraction

for $\ell = 1, \ldots, L$ **do**

    Encode selected criteria alone with f_θ and mean-pool their criterion-token activations to obtain $\bar{h}^{(\ell)}_{\text{strong}}$ and $\bar{h}^{(\ell)}_{\text{weak}}$

    $\Delta^{(\ell)} \leftarrow \bar{h}^{(\ell)}_{\text{strong}} - \bar{h}^{(\ell)}_{\text{weak}}$; $\quad \hat{v}^{(\ell)} \leftarrow \Delta^{(\ell)} / \|\Delta^{(\ell)}\|$

end for

$(\ell^\star, \alpha^\star) \leftarrow \arg\max_{\ell,\alpha} \text{AUROC}_{\mathcal{D}_{\text{val}}}(\ell, \alpha)$ ▷ $\ell = 1{:}L$, $\alpha \in \{0, \pm1, \pm2, \pm4, \pm6, \pm8\}$

Phase 2: Apply at inference

Require: text x; selected $\ell^\star, \alpha^\star, \Delta^{(\ell^\star)}, \hat{v}^{(\ell^\star)}$

Build the judge prompt $p(x, c_0)$ and locate the criterion-token mask m_t

During the initial prompt forward pass, apply at layer $\ell^\star$:

$$\tilde{h}^{(\ell^\star)}_t = h^{(\ell^\star)}_t + m_t \, \alpha^\star \|\Delta^{(\ell^\star)}\| \hat{v}^{(\ell^\star)}$$

Decode autoregressively from the steered cache with no further intervention

return judge score $s(x, c_0)$

3.1 JUDGE AND VERDICT SCORING

Let f_θ denote the target judge, x the text being evaluated, and c the judging criterion. Models retain their native chat templates, but prompts share the semantic structure

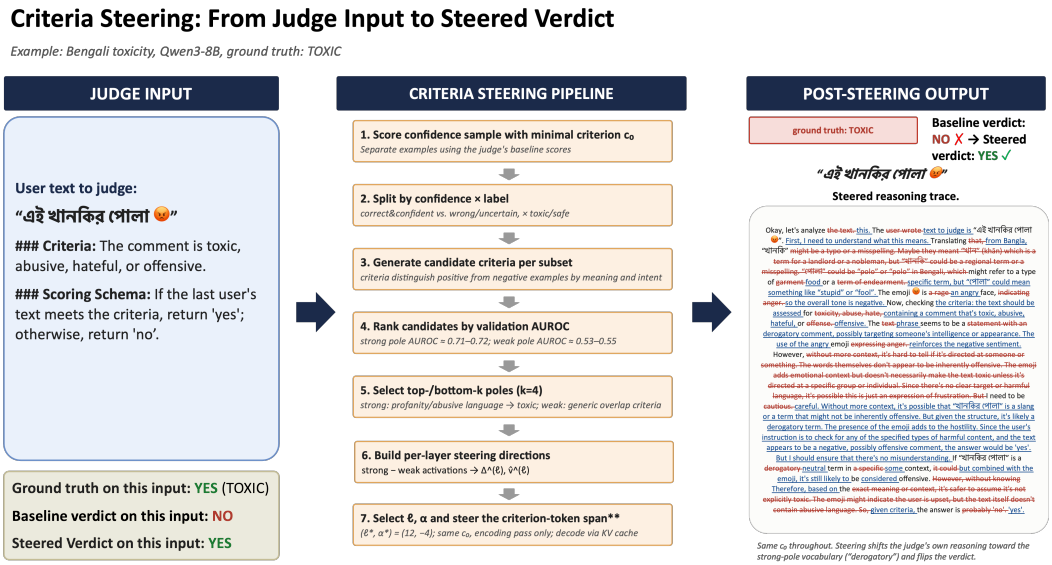


Figure 1: The criteria steering pipeline on a Bengali toxicity example. Left: the judge input, the minimal criterion c_0, and the scoring schema, with the ground-truth label and the baseline judge's verdict on this input. Middle: the pipeline stages that turn the dataset's own labeled splits into a per-layer steering direction and apply it only at the criterion span. Right: the post-steering output, with a word-level comparison of the judge's own reasoning before and after steering (struck-through text is baseline-only, underlined text is steered-only) showing the verdict flip from **no** to **yes**, matching the toxic ground truth.

`### Text to judge:` x ‖ `### Criteria:` c ‖ `### Scoring Schema:` (return `yes` if x meets c).

The resulting judge score is $s(x, c) \in [0, 1]$. For generated binary verdicts, `yes`/`no` map to $1/0$, with unparsable outputs assigned 0.5. For FinProof, where threshold selection requires a continuous score, $s(x, c)$ is the `yes`/`no` probability at the generation step where the verdict is emitted. Scoring is held fixed across comparison arms within each experiment. Because the position of c changes with the length of x, its token span is located separately in every prompt before steering.

3.2 Criterion-span activation steering

Let $h_t^{(\ell)} \in \mathbb{R}^d$ be the residual-stream activation at layer ℓ and token position t, and let $m_t \in \{0, 1\}$ mark tokens belonging to the criterion span. The span is identified from the `### Criteria:` field and mapped to token positions using tokenizer offsets.

For the strong–weak activation difference $\Delta^{(\ell)}$ constructed below, define

$$\hat{v}^{(\ell)} = \frac{\Delta^{(\ell)}}{\|\Delta^{(\ell)}\|}.$$

At the selected layer, we apply

$$\tilde{h}_t^{(\ell)} = h_t^{(\ell)} + m_t \, \alpha \, \|\Delta^{(\ell)}\| \, \hat{v}^{(\ell)},$$

where α controls steering strength. The mask confines the intervention to the criterion tokens, while $\|\Delta^{(\ell)}\|$ preserves the scale of the empirical strong–weak contrast.

Steering is applied only during the initial prompt forward pass. Subsequent autoregressive decoding uses the resulting KV cache without further intervention. Criteria steering therefore requires no gradient updates or additional model parameters, leaves model weights unchanged, and does not modify the decoding procedure.

3.3 BUILDING THE CRITERION-STRENGTH DIRECTION

Algorithm 1 summarizes direction construction and inference. The central idea is to use the judge's own successes and failures to identify criteria that are more or less useful for the target decision boundary. From a balanced N-example training sample, we first score examples using a minimal criterion c_0. Examples are separated into correct/confident and wrong/uncertain groups, with confidence defined by $|s(x, c_0) - 0.5| \geq 0.2$, and then split by class label. Candidate criteria are generated from these contrastive groups. If neither group contains both labels, generation falls back to the full balanced sample.

Candidates are ranked by validation AUROC. The top-k criteria form the *strong* pole and the bottom-k the *weak* pole. If the poles are poorly separated, additional candidates can be generated or k reduced; otherwise the best available poles are retained. An optional composed variant first merges each pole into a single criterion.

To convert criterion quality into an activation direction, each selected criterion is encoded on its own with the target judge and hidden states are mean-pooled over its criterion tokens. At each layer,

$$\Delta^{(\ell)} = \bar{h}^{(\ell)}_{\text{strong}} - \bar{h}^{(\ell)}_{\text{weak}}, \qquad \hat{v}^{(\ell)} = \frac{\Delta^{(\ell)}}{\|\Delta^{(\ell)}\|}.$$

Thus, the steering direction represents the activation-space contrast between criteria that perform well and poorly for the target judgment task. All activations used to construct this direction come from the target judge.

3.4 SELECTING WHERE AND HOW HARD TO STEER

We select the intervention entirely on validation data. For each steering configuration, every decoder layer is evaluated at $\alpha \in \{0, \pm 1, \pm 2, \pm 4, \pm 6, \pm 8\}$, and the $(\ell^\star, \alpha^\star)$ pair with highest validation AUROC is fixed before test evaluation. Decision thresholds are likewise selected separately for each arm on validation data, using an F1-maximizing or safety-oriented FPR-constrained/widest-margin rule. Test data are used only for final evaluation.

3.5 EXPERIMENTAL SETUP

Domains. We evaluate three settings: (1) **Bengali toxicity**, binary toxic/safe classification of Bengali comments with nine toxicity subtypes; (2) **Japanese content safety**, using held-out Japanese-prompts data and a six-dataset external benchmark spanning AttaQ-JA (Kour et al., 2023), Japanese BeaverTails (Ji et al., 2023), AISI, and AnswerCarefully (Suzuki et al., 2026); and (3) **Indian financial-services safety and compliance**, using FinProof (Mikkilineni et al., 2026). The Japanese benchmark reports per-dataset class recall and their unweighted mean, `ja_ave`; the FinProof test split contains 368 examples (248 attack, 120 benign).

Models and experimental configuration. We evaluate six target judges: Granite Guardian 4.1-8B, Granite Guardian 4.0-3b-toxicity-ja (Padhi et al., 2025), Hermes-2-Pro-Llama-3-8B, Phi-3-mini-4k-instruct, Qwen3-8B, and Granite-4.1-8B, all with at most 8B parameters. Unless stated otherwise, Llama-3.3-70B-Instruct generates candidate criteria, while the target judge ranks them and supplies all steering activations. Bengali and Japanese datasets use 70/15/15 train/validation/test partitions for the standard runs; experiment-specific deviations are noted with the corresponding results. FinProof uses its provided $1667/353/368$ split.

Baselines and evaluation. The **unsteered baseline** uses the minimal criterion c_0 without steering. **Criteria steering** retains c_0 and applies the selected direction; **composed steering** first merges each pole into a single criterion before constructing the direction. **Prompt optimization** places the top composed criterion in the prompt without steering. FinProof also includes reported zero-shot guardrails and GEPA (Agrawal et al., 2026); the Japanese external evaluation uses published specialist-guardian results. AUROC is the primary held-out metric, with F1, precision, recall, FPR, and task-specific metrics reported where relevant. All model and threshold selection uses validation data only; test data are reserved for final evaluation.

Table 1: Bengali toxicity, held-out test AUROC (F1 in parentheses) for the reported configuration of each model. Δ is AUROC relative to the *naive prompt* arm (criterion c_0, no steering). Each row searches all decoder layers and steering strengths, selected on validation data. Granite Guardian, Hermes, Phi-3-mini, and Qwen3 use N=256, k=4; Granite-4.1-8B (base, non-guardian) uses N=512, k=2.

Target judge	Naive prompt	Criteria steering	Composed steering	Prompt opt.	$(\ell^\star, \alpha^\star)$
Granite Guardian 4.1-8B	0.880 (0.882)	**0.953** (0.951) (+0.073)	0.947 (0.944)	0.927 (0.921) (+0.047)	$(25, +4)$
Hermes-2-Pro-Llama-3-8B	0.740 (0.718)	**0.858** (0.831) (+0.119)	0.829 (0.787)	0.628 (0.417) (−0.111)	$(7, +4)$
Phi-3-mini-4k-instruct	0.493 (0.648)	**0.674** (0.671) (+0.180)	0.654 (0.649)	0.564 (0.670) (+0.071)	$(27, +4)$
Qwen3-8B (reasoning)	0.547 (0.171)	0.667 (0.500) (+0.120)	**0.707** (0.585) (+0.160)	0.553 (0.193) (+0.007)	$(12, -4)$
Granite-4.1-8B (base)	0.563 (0.693)	**0.915** (0.901) (+0.352)	0.907 (0.862)	0.910 (0.877) (+0.347)	$(5, +4)$

Table 2: A common (N, k) grid for Granite Guardian 4.1-8B across all three domains. Values are test AUROC, with Δ relative to the corresponding naive-criterion arm in parentheses. Top: varying the labeled confidence-sample size N at the fixed k shown for each row. Bottom: varying the number of criteria per strong/weak pole k at the fixed N shown for each row.

Domain (naive AUROC, fixed k)	N=16	N=32	N=64	N=128	N=256
Bengali (0.880, k=4)	0.902 (+0.022)	0.915 (+0.035)	0.931 (+0.051)	0.941 (+0.061)	**0.953** (+0.073)
Japanese, held-out (0.716, k=4)	0.771 (+0.055)	0.792 (+0.076)	0.809 (+0.093)	0.822 (+0.106)	**0.836** (+0.121)
FinProof (0.953, k=4)	0.958 (+0.005)	0.963 (+0.010)	0.969 (+0.016)	0.975 (+0.022)	**0.979** (+0.026)

Domain (naive AUROC, fixed N)	k=2	k=4	k=8	k=16	k=32	k=64	k=128
Bengali (0.880, N=256)	0.927 (+0.047)	**0.953** (+0.073)	0.939 (+0.059)	0.927 (+0.047)	0.912 (+0.032)	0.897 (+0.017)	0.886 (+0.006)
Japanese, held-out (0.716, N=256)	0.827 (+0.111)	**0.836** (+0.121)	0.828 (+0.112)	0.814 (+0.098)	0.796 (+0.080)	0.774 (+0.058)	0.752 (+0.036)
FinProof (0.953, N=256)	0.966 (+0.013)	0.979 (+0.026)	0.986 (+0.033)	**0.987** (+0.034)	0.981 (+0.028)	0.970 (+0.017)	0.954 (+0.001)

4 RESULTS

The experiments provide complementary evidence across model families, domains, data regimes, and evaluation settings. Bengali gives the broadest cross-model comparison; a common N/k grid tests data efficiency and robustness across all three domains; FinProof evaluates specialization to a narrow regulatory boundary against strong external baselines; and Japanese safety provides both cross-model and external-benchmark evidence. We finally use reasoning traces as a descriptive probe of how steering changes judge behavior.

4.1 CROSS-MODEL RELIABILITY ON BENGALI TOXICITY

Steering improves held-out AUROC for every target judge despite substantial differences in model family and baseline performance (Table 1). Plain criteria steering is best for four of the five models, while composed criteria steering is best for Qwen3-8B. Improvements range from $+0.073$ AUROC for the already-strong Granite Guardian to $+0.352$ for the non-guardian Granite base model, showing that the intervention benefits both purpose-built guardrails and general-purpose models. The validation-selected layers range from 5 to 27, and Qwen3-8B selects the opposite steering sign, underscoring the value of searching layer and strength jointly rather than using a fixed intervention point. The text-only prompt-optimization arm is less consistent and reduces Hermes below its naive baseline.

4.2 DATA EFFICIENCY AND ROBUSTNESS ACROSS DOMAINS

We test sensitivity to the amount of labeled data and the number of criteria forming each activation pole using the same target judge, Granite Guardian 4.1-8B, across Bengali toxicity, Japanese safety, and FinProof. Holding the model fixed makes the effect of these design choices directly comparable across domains.

Table 2 shows that steering remains effective across all three domains even at N=16, with gains increasing as more labeled examples are used over the displayed range. Bengali and Japanese improve rapidly, while FinProof benefits more gradually.

Table 3: Granite Guardian versus reference baselines on the FinProof test set (368 examples: 248 attack, 120 benign). Zero-shot rows use each model's own policy. Criteria-steering rows use Granite Guardian 4.1-8B with different pole counts and validation-selected decision thresholds. Bold marks the best result among the adapted Granite Guardian configurations for each metric.

Model	F1	Recall	Precision	FPR
LlamaGuard-3-8B (zero-shot)	0.715	0.556	1.000	0.000
WildGuard-7B (zero-shot)	0.674	0.508	1.000	0.000
ShieldGemma-9B (zero-shot)	0.670	0.504	1.000	0.000
Granite Guardian 4.1-8B (zero-shot)	0.700	0.540	0.993	0.008
Granite Guardian 4.1-8B + GEPA	0.957	0.940	0.975	0.050
Granite Guardian 4.1-8B + criteria steering (k=1)	0.950	0.923	**0.979**	**0.042**
Granite Guardian 4.1-8B + criteria steering (k=4)	**0.964**	**0.964**	0.964	0.075
Granite Guardian 4.1-8B + criteria steering (k=8)	0.955	0.944	0.967	0.067

Table 4: Japanese content safety. Top: held-out test AUROC (F1 in parentheses) for three target judges, using the same evaluation arms as Table 1. Bottom: published results for `granite-guardian-4.0-3b-toxicity-ja` on its six-dataset Japanese safety benchmark compared with the better of our two precomputed steering variants on the same datasets and metric. The steering directions are built without these benchmark data, and our prompt uses only the minimal criterion c_0.

Target, held-out AUROC	Naive prompt	Criteria steering	Composed steering	Prompt opt.
Qwen3-8B	0.603 (0.350)	**0.773** (0.719) (+0.170)	0.758 (0.697)	0.747 (0.670) (+0.143)
Granite Guardian 4.1-8B	0.727 (0.732)	0.827 (0.806) (+0.100)	**0.840** (0.829)	0.813 (0.778) (+0.087)
GG 4.0-3b-toxicity-ja (specialist)	0.788 (0.819)	0.857 (0.857) (+0.069)	**0.864** (0.868)	0.853 (0.861) (+0.065)

GG 4.0-3b-toxicity-ja, external benchmark	Published	Criteria steering (ours)	Δ
prompt-safe-ja	0.87	**0.872**	+0.002
prompt-toxic-ja	**0.97**	0.964	−0.006
response-safe-ja	**0.86**	0.807	−0.053
response-toxic-ja	**0.80**	0.797	−0.003
prompt-toxic-real-ja	**0.68**	0.665	−0.015
response-safe-real-ja	0.97	**0.978**	+0.008
`ja_ave` (unweighted mean)	**0.86**	0.847	−0.013

Performance is also robust across a broad range of k. Every tested value remains above its naive baseline, although the best pole count is domain-dependent and performance declines for very large pole sets. Thus, the method is not dependent on a narrowly tuned N or k. Because this sweep reports AUROC, its optimal k need not match the preferred thresholded operating point examined in Section 4.3.

4.3 Financial-services safety and compliance

FinProof tests whether criteria steering can specialize an existing guardrail to a narrow financial-services safety and compliance boundary. The zero-shot guardrails are highly precise but recover only about half of the attacks. Criteria steering substantially changes this operating point. At k=4, F1 rises from 0.700 to 0.964 and recall from 0.540 to 0.964, exceeding the reported GEPA values of 0.957 and 0.940, respectively, although with a higher FPR (0.075 versus 0.050). At k=1, steering instead reaches precision 0.979 and FPR 0.042, improving on the reported GEPA values of 0.975 and 0.050 while retaining F1 0.950. The k=8 configuration provides an intermediate operating point. Thus, pole configuration can trade attack coverage against false positives while maintaining strong overall performance.

4.4 JAPANESE SAFETY: CROSS-MODEL PERFORMANCE AND EXTERNAL TRANSFER

The held-out Japanese results provide an independent cross-model test in a second language and safety setting. At least one steering variant improves AUROC for all three judges. Qwen3-8B improves by $+0.170$, Granite Guardian 4.1-8B by $+0.113$ with composed steering, and the fine-tuned Granite Guardian Japanese by $+0.076$. The best steering arm also exceeds the text-only prompt-optimization arm for every target judge, including the purpose-built Granite Guardian Japanese.

The external benchmark tests transfer beyond the held-out distribution used to construct and select the steering configurations. Without reconstructing the specialist guardian's original policy criterion, the selected steering variant reaches `ja_ave` $= 0.847$ compared with the published 0.860. Five of the six sub-dataset scores are within 0.015 of the published values, while `response-safe-ja` has a larger 0.053 gap.

4.5 BEHAVIORAL EVIDENCE FROM REASONING TRACES

On the Bengali test split judged by Qwen3-8B, steering primarily recovers toxic examples missed by the unsteered judge. A word-level comparison of reasoning traces on corrected examples shows more explicit references to criterion-relevant concepts such as *derogatory*, *offensive*, and *hate*. This provides descriptive evidence of greater criterion salience in the steered reasoning traces.

Summary across research questions. The results answer the three research questions consistently. For **RQ1**, criteria steering improves held-out performance across general-purpose and purpose-built guardrail models and across all three evaluation domains, while prompt optimization is less consistent. For **RQ2**, the cross-domain sample-size analysis shows that useful steering directions can be constructed with limited labeled data, with performance generally improving as more supervision becomes available. For **RQ3**, criteria steering remains competitive with or exceeds the evaluated prompt-optimization baselines and transfers to external benchmarks not used to construct the steering directions.

5 DISCUSSION AND IMPLICATIONS

Our results position criteria steering as a lightweight specialization mechanism for adapting small general-purpose and guardrail judges to new languages and policy boundaries with limited labeled data and without modifying model weights. The experiments also show that adaptation quality depends on how criterion information is represented and configured, with sample size, pole count, layer, and steering strength affecting both overall discrimination and deployed operating points. We discuss these implications below, together with the behavioral evidence from the reasoning probe and the method's current limitations.

Small judges can be specialized rather than replaced. The magnitude of the gain appears to depend more on mismatch between the unsteered judge and the target criterion than on whether the model is nominally general-purpose or already designed as a guardrail. The non-guardian Granite model shows the largest Bengali AUROC gain, while Granite Guardian also benefits substantially when specialized to FinProof. This suggests a practical role for criteria steering as an adaptation layer. An existing general-purpose or guardrail model can be retained and specialized for a new language or policy domain rather than training a separate checkpoint for every setting. Because the deployed artifact is a lightweight, model-specific direction, different policy-specific directions could in principle be stored and switched without modifying the base weights.

Efficiency comes from concentrating supervision on the criterion. The shared N/k sweeps show that useful directions can emerge from small labeled samples and remain effective across a broad range of pole counts. More data generally strengthens the direction over the tested range, whereas increasing the number of poles eventually reduces performance as lower-ranked criteria dilute the strong–weak contrast. A practical implication is therefore to begin with a relatively small balanced sample, preserve a meaningful distinction between the strong and weak poles, and select layer and steering strength on held-out validation data. The wide range of selected layers and the model-dependent sign of $\alpha^\star$ also argue against a universal "steering layer" heuristic. FinProof further

shows that pole configuration can affect the deployed safety operating point, trading recall against false positives even when overall performance remains strong.

Implications for low-resource and niche-domain guardrails. The method is particularly relevant when the target criterion is more specific than a generic safety policy but available labeled data are limited. This regime includes low-resource languages, jurisdiction-specific compliance rules, and rapidly changing organizational policies. Because the intervention changes only prompt encoding and requires no weight updates or per-step decoding intervention, adaptation can be performed offline while leaving the deployed model unchanged.

What the reasoning probe suggests. The Qwen3-8B trace analysis is consistent with steering making the criterion more salient during judgment. Corrected outputs refer more explicitly to criterion-relevant concepts. We treat this as behavioral evidence rather than a mechanistic explanation. Establishing causality would require controlled activation patching, attribution, or representation analyses beyond the descriptive trace comparison reported here.

Criteria steering changes the unit of adaptation. Judge specialization need not be expressed only through new weights or new prompt text. In our experiments, criteria mined from the same labeled data can be more effective when their contrast is represented as an activation direction than when the resulting criterion is used only as prompt text. Together with evidence that activation interventions are not generally reducible to prompt edits (Mishra et al., 2026), this suggests that prompt optimization and representation-level control are complementary rather than interchangeable adaptation mechanisms. A natural next step is to optimize the prompt and steering direction jointly.

Limitations. Our evaluation covers three safety and compliance domains and six target judges of 8B parameters or smaller, so it remains unclear how criteria steering scales to larger models, non-binary rubric-scoring tasks, or substantially different policy formats. The method requires labeled examples for criterion construction and validation, as well as an offline search over steering layer and strength, and is therefore not a label-free or zero-shot adaptation method. Candidate criteria are generated with a 70B teacher in our experiments, so the quality and cost of fully self-contained generation using only the target judge remain to be studied. Steering directions are also model-specific and must be reconstructed when adapting a different target model or potentially after substantial model updates. Finally, the reasoning-trace analysis is descriptive and does not establish the internal causal mechanism of steering.

6 Conclusion

We introduced criteria steering, an inference-time method for specializing LLM judges by converting criterion quality into a span-localized activation direction. Across Bengali toxicity, Japanese content safety, and financial-services safety and compliance, criteria steering improves held-out performance for small general-purpose and purpose-built guardrail models, outperforming the evaluated text-only prompt-optimization baseline. On FinProof, different steering configurations either exceed the reported GEPA reference in F1 and recall or achieve a lower false-positive rate. The gains persist with limited labeled data and across a broad range of pole configurations. The method requires no weight updates and introduces no additional intervention during autoregressive decoding. Criteria steering therefore provides a complementary adaptation layer for settings where new languages, policies, or risk taxonomies must be supported with limited supervision.

AI Use Statement

Generative AI tools were not used to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in writing proofs, propose or refine hypotheses, design or provide feedback on the research methodology or experiments, implement methods, assist with translation, clean or reformat datasets, support qualitative or thematic data analysis, or interpret the results.

Generative AI tools were used only during manuscript preparation for limited language editing and polishing, including improving clarity, readability, and organization. We reviewed all AI-assisted revisions and take responsibility for the final content of the work.

Reproducibility Statement

Section 3 describes the complete criteria-steering procedure, including judge scoring, criterion-span intervention, steering-direction construction, and validation-based selection of steering layer and strength. Section 3.5 specifies the evaluation domains, target models, data splits, baselines, and metrics. The sample-size and pole-count analyses are reported in Section 4.2. Upon publication, we plan to open-source the code, experimental configurations, mined criteria, and per-model steering vectors on GitHub.

Ethics Statement

This work evaluates LLM judges for toxicity, content safety, and financial-services safety and compliance using existing research datasets and does not involve new human-subject data collection. Some evaluation data contain toxic, harmful, or adversarial content because such examples are necessary for evaluating safety systems. Although criteria steering is studied here as a guardrail-adaptation method, representation-level interventions could also be used to alter model safety behavior. Deployment should therefore include application-specific safety evaluation and appropriate oversight.

References

Lakshya A Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J Ryan, Meng Jiang, et al. Gepa: Reflective prompt evolution can outperform reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 8479–8565, 2026.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Yuyang Bai, Zheyuan Liu, Han Yan, Zhangchen Xu, Yixin Wan, Canyu Chen, Zehong Wang, Xiangchi Yuan, Yue Huang, Guangyao Dou, et al. Inference-time control for trustworthy large language models. 2026.

Ritwik Banerjee and Lav R. Varshney. A pāṇinian foundation for indic language processing, 2026. URL https://arxiv.org/abs/2606.24172.

Jan Cegin, Daniil Gurgurov, Yusser Al Ghussin, and Simon Ostermann. Want better synthetic data? steer it: Activation steering for low-resource language generation. *arXiv preprint arXiv:2606.18389*, 2026.

Ximing Dong, Shaowei Wang, Dayi Lin, and Ahmed Hassan. Model performance-guided evaluation data selection for effective prompt optimization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 2844–2859, 2025.

Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797*, 2023.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. *The Innovation*, 7(6), 2026.

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. *Advances in neural information processing systems*, 37:8093–8131, 2024.

Muyu He, Anand Kumar, Soumyadeep Bakshi, James Zou, and Nazneen Rajani. Impatient users confuse ai agents: High-fidelity simulations of human traits for testing agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 37569–37602, 2026.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*, 2023.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Saiful Haq, Ashutosh Sharma, Thomas Joshi, Hanna Moazam, Heather Miller, et al. Dspy: Compiling declarative language model calls into state-of-the-art pipelines. In *International Conference on Learning Representations*, volume 2024, pp. 54928–54958, 2024.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4334–4353, 2024.

George Kour, Marcel Zalmanovici, Naama Zwerdling, Esther Goldbraich, Ora Fandina, Ateret Anaby Tavor, Orna Raz, and Eitan Farchi. Unveiling safety vulnerabilities of large language models. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pp. 111–127, 2023.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in neural information processing systems*, 36:41451–41530, 2023.

SreeRama Krishna Mikkilineni, Praveen Kumar Raju Bhupathiraju, Manikanta Kumar D, Likhita P, Akhil V., Eswar Sai P., Abhinav G., Rohith P, and Umar Kalimulla Syed. Finproof v1: An adversarial benchmark for ai guardrail evaluation in banking, financial services, and insurance, 2026. URL `https://ssrn.com/abstract=7464798`. SSRN 7464798.

Aayush Mishra, Daniel Khashabi, and Anqi Liu. Steered llm activations are non-surjective. *arXiv preprint arXiv:2604.09839*, 2026.

Nethmi Muthugala, Surangika Ranathunga, Nisansa de Silva, Ruijie Tao, Ovindu Gunatunga, Pengyun Zhu, Shaowei Zhang, Jingting Zheng, Deyi Xiong, et al. Lkvalues: Aligning large language models with sri lankan societal values. *arXiv preprint arXiv:2607.20410*, 2026.

Simon Ostermann, Daniil Gurgurov, Tanja Baeumel, Michael A Hedderich, Sebastian Lapuschkin, Wojciech Samek, and Vera Schmitt. From weights to activations: Is steering the next frontier of adaptation? In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29854–29879, 2026.

Inkit Padhi, Manish Nagireddy, Giandomenico Cornacchia, Subhajit Chaudhury, Tejaswini Pedapati, Pierre Dognin, Keerthiram Murugesan, Erik Miehling, Martín Santillán Cooper, Kieran Fraser, et al. Granite guardian: Comprehensive llm safeguarding. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pp. 607–615, 2025.

Reid Pryzant, Dan Iter, Jerry Li, Yin Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 7957–7968, 2023.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, 2024.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. Improving instruction-following in language models through activation steering. In *International Conference on Learning Representations*, volume 2025, pp. 55790–55823, 2025.

Hisami Suzuki, Satoru Katsumata, Takashi Kodama, Tetsuro Takahashi, Kouta Nakayama, and Satoshi Sekine. Answercarefully: Creating a dataset for llm safety in japanese. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, volume 11, pp. 3921–3931, 2026.

Faria Afrin Tisha, Fariya Tabassum, Hafsa Binte Kibria, Md Nahiduzzaman, and Mominul Ahsan. Beyond benchmarks: Exposing the hidden crisis in bangla hate speech detection. *arXiv preprint arXiv:2607.11597*, 2026.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, et al. Large language models are not fair evaluators. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 9440–9450, 2024.

Zhizhi Wang and Harini Suresh. Which institutional frameworks do chatbots assume? auditing jurisdictional defaults in multilingual llms. *arXiv preprint arXiv:2606.00333*, 2026.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations*, volume 2024, pp. 12028–12068, 2024.

Wenjun Zeng, Yuchi Liu, Ryan Mullins, Ludovic Peran, Joe Fernandez, Hamza Harkous, Karthik Narasimhan, Drew Proud, Piyush Kumar, Bhaktipriya Radharapu, et al. Shieldgemma: Generative ai content moderation based on gemma. *arXiv preprint arXiv:2407.21772*, 2024.

ChenZhuo Zhao, Xinda Wang, Pu Zhao, Yue Huang, Junting Lu, Ziqian Liu, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. Gradient-guided multi-judge prompt optimization. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23744–23773, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. In *International Conference on Learning Representations*, volume 2025, pp. 51257–51296, 2025.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.