RankOne: Towards a Single Direction Approach for Response-Level Safety Classification

Anonymous ACL submission

Abstract

Safety classifiers for LLMs attach supervised heads to encoder representations, whose safety-discriminative contribution may reduce to reading out a direction already present in the embedding space. A *single global direction* (RankOne), formed by averaging per-prompt safe/unsafe mean-difference vectors, concentrates the linearly-separable safe/unsafe signal into a rank one axis. We show that even post finetuning on domain specific dataset, results with and without task specific head (RankOne) yield comparable results, while outperforming task specific head by 8.9% relative AUROC in the low-data regime, transferring across seven benchmarks and three encoder families with zero target labels. We establish through rigorous experiments that our directional vector achieves zero-shot transfer across 7 diverse safety benchmarks and 3 encoder families. Compared with Llama-Guard-3-8B across four prompt templates, RankOne gains $+30.8\%$ relative AUROC on the ToxicChat jailbreak slice while using $29x\times$ fewer parameters, and $+0.053$ on BeaverTails (both CIs exclude zero). Our results in this paper establishes harmfulness to be a geometrically coherent concept in language model representations, that pre-exists safety training and is extractable via a single linear direction. All code used in this paper will be released post review.

1 Introduction

Large language models (LLMs) are increasingly deployed in open-ended conversational settings, where detecting whether a generated response is harmful has become imperative. The dominant paradigm for this response-level safety classification trains a supervised classifier, typically a multi-layer perceptron, on top of a frozen or fine-tuned text encoder, mapping a high-dimensional representation to a binary safe/unsafe decision. Systems such as Llama-Guard (Inan et al., 2023a) exemplify this recipe and achieve strong empirical performance, but they do so at substantial cost: billions of parameters, heavy reliance on a curated safety taxonomy, and limited portability to benchmarks whose annotation guidelines differ from the training set.

In parallel, a growing line of work on the linear representation hypothesis has documented that high-level semantic concepts — truthfulness (Marks and Tegmark, 2023), refusal (Arditi et al., 2024), toxicity, and sentiment — are repeatedly found to be encoded as low-dimensional linear substructure within LLM hidden states. Marks and Tegmark (2023) recover truthfulness from a single mean-difference direction, while Arditi et al. (2024) demonstrate that refusal behavior in chat-tuned models is mediated by a single direction, the ablation of which is sufficient to dismantle safety guardrails altogether. The neural-collapse framework (Papyan et al., 2020) offers a complementary prediction from the optimization perspective: binary cross-entropy training drives the two class means toward antipodal alignment along a single axis, yielding a rank-one geometry by construction.

Taken together, these observations sharpen a basic question for safety classification: if the safety-relevant signal inside an encoder is already concentrated along a single linear direction, what does the supervised classification head actually contribute beyond it? To the extent that its contribution is marginal, safety classification effectively collapses to a single dot product against that direction, yielding immediate benefits in inference cost, interpretability, and cross-benchmark transfer, while removing the need to retrain a task-specific head every time the underlying taxonomy is revised.

This paper provides a systematic exploration towards answering the above questions. For every training prompt that admits both a safe and an unsafe response, we construct the per-prompt direction pointing from the safe-response embedding

to the unsafe-response embedding, and aggregate these directions by simple averaging into a single global safety direction. A new response is then scored by its projection onto this axis — a one-line readout from the encoder, with no task-specific head attached. We refer to this rank-one scorer as RankOne, and study how it fares against supervised heads trained on the same encoder, as well as against dedicated safety guards, across a range of encoder families and safety benchmarks.

Our contributions. (i) We show that response-level safety is essentially a one-dimensional problem in the encoder's representation space. (ii) in the low-data regime, RankOne performs strictly better than MLP head trained on the same embeddings. (iii) The safety direction transfers across benchmarks, suggesting a shared, encoder-agnostic notion of harmfulness rather than an artifact of any particular model or taxonomy. (iv) For binary response-level safety classification on standard English benchmarks, the supervised head is dispensable *at inference* once the underlying safety direction is exposed with the caveat that exposing it still requires a one-time binary cross-entropy fine-tune of the encoder, after which the head itself can be discarded. We further include insights based on eigen spectrum analysis about how our approach can be extended to multilabel/multilingual setting as well.

The rest of the paper is organized as follows. §2 surveys related work on linear safety directions, neural collapse, and safety classifiers. §3 presents the method, §4 reports main results, §5 provides ablations and analysis, and §6 concludes with limitations and open questions.

2 Related Work

Linear safety directions. Marks and Tegmark (2023) show truthfulness in LLM hidden states is captured by a mean-difference direction; Park et al. (2024) formalize this as the linear representation hypothesis; Arditi et al. (2024) show refusal is mediated by a single direction in chat-tuned LLMs. Our work applies the same geometric lens to *discriminative* response-safety encoders, quantifying parameter equivalence and cross-architecture transfer; the key distinction from Arditi et al. is that we classify responses rather than steer generation. Zou et al. (2023a) extract reading vectors for generation-time steering (RepE); Li et al. (2024) apply inference-time intervention along linear directions for truthfulness; we focus on classification, not generation-time control. Zou et al. (2023b) demonstrate that adversarial suffixes can bypass aligned models; this attack surface applies equally to our linear scorer and is discussed in the limitations. Concept erasure methods such as LEACE (Belrose et al., 2023) and CCS (Burns et al., 2024) identify or remove linear concept directions; our contribution is characterizing the *completeness* of one such direction for safety. The rank-one scorer can also be viewed as a one-dimensional concept bottleneck (Koh et al., 2020); unlike standard concept bottleneck models, the concept direction is extracted without supervision beyond the binary label. Hewitt and Liang (2019) argue that high probe accuracy may reflect encoder capacity rather than genuine structure; our frozen-encoder result addresses this by showing the direction is meaningful even without task-specific training. Belinkov (2022) surveys probing methodology more broadly; our frozen-encoder baseline provides evidence under this framework that a safety axis exists in pretraining, not only in task-specific capacity.

Neural collapse. Papyan et al. (2020) show that binary cross-entropy training on K classes drives embeddings toward a simplex equiangular tight frame (ETF). For $K{=}2$ this predicts rank-one geometry: class means become antipodal and within-class variance collapses. We verify NC1–NC3 on our encoder (NC1 ratio drops $97\times$ after fine-tuning; $\cos(\mu^+, \mu^-) = -0.398$; classifier–mean $\cos = 0.993$). The rank-one geometry is therefore *expected* from training dynamics. Our contribution is showing (a) how completely it captures the safety signal and (b) that the collapsed axis transfers across benchmarks, which neural collapse does not predict.

Safety classifiers and benchmarks. Llama-Guard (Inan et al., 2023b; Meta AI, 2024) and ShieldGemma (Zeng et al., 2024) train supervised heads over encoder representations. We show the head can be replaced by a single dot product with the safety direction. We evaluate on PKU-SafeRLHF (Ji et al., 2024), BeaverTails (Ji et al., 2023), ToxicChat (Lin et al., 2023), WildGuardMix, HarmBench (Mazeika et al., 2024), and AEGIS (Ghosh et al., 2024) (11K-row NVIDIA safety benchmark with 13 categories, fully independent annotation).

Distinction from prior linear probes. Mean-shift probes for toxicity and refusal (Wei et al., 2024; Tigges et al., 2024) and probes for safety-aligned LMs (Zheng et al., 2024) all establish that a low-rank linear subspace carries safety/refusal/toxicity signal in *generative* LM hidden states. We differ on three axes: (i) we target *response-level discriminative* encoders rather than generative residual streams; (ii) we quantify *completeness* (the supervised head adds nothing on top of the safety direction) rather than mere existence of the direction; and (iii) we test *cross-encoder* (`me5`, `bge`, `gte-Qwen2`) and *cross-annotation-ecosystem* transfer (PKU-SafeRLHF → HarmBench/AEGIS), which prior probing work does not address.

3 Method

Given an ℓ_2-normalized encoder $\phi : \Sigma^* \to S^{d-1}$ (the unit sphere in $\mathbb{R}^d$) and training triples $\{(p, r, y)\}$ with $y \in \{0, 1\}$ where $y{=}1$ for unsafe and $y{=}0$ for safe, let R_p^+ and R_p^- denote the sets of unsafe and safe responses to prompt p, respectively. For each prompt p that has at least one safe and one unsafe response (we call such prompts *dual-feasible*), we compute the mean unsafe and safe embeddings

$$\bar{\phi}_p^+ = \frac{1}{|R_p^+|} \sum_{r \in R_p^+} \phi(r), \qquad \bar{\phi}_p^- = \frac{1}{|R_p^-|} \sum_{r \in R_p^-} \phi(r),$$

and a unit discriminant $d_p = \frac{\bar{\phi}_p^+ - \bar{\phi}_p^-}{\|\bar{\phi}_p^+ - \bar{\phi}_p^-\|} \in S^{d-1}$. Figure 1 illustrates the pipeline. The global scorer collapses these into one direction:

$$\bar{d} = \frac{\sum_p d_p}{\|\sum_p d_p\|}, \quad s(r) = \langle \bar{d}, \phi(r) \rangle. \tag{1}$$

The decision rule thresholds $s(r)$ at a val-selected cutoff γ^*. Inference cost is one encoder forward pass plus a dot product. Algorithm 1 gives pseudocode.

The encoder is either (a) a frozen pretrained model (`multilingual-e5-base`, referred to as `me5-base` hereafter), or (b) an encoder fine-tuned with binary cross-entropy on the same (p, r, y) pairs, after which the MLP head is discarded and replaced by the safety direction. We also implemented a retrieval-composed variant that conditions on the test prompt via top-k nearest neighbors and weighted Karcher mean on S^{d-1}; our ablations (§5) show it adds no signal over the global mean. Details and the full spherical-mean formalism are in Appendix A.

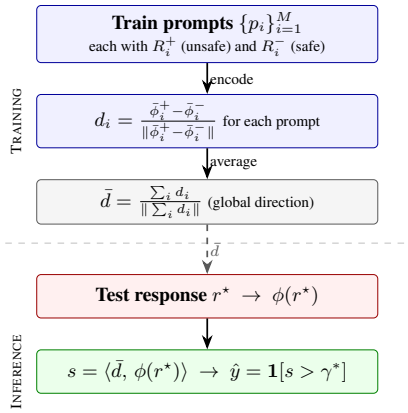

Figure 1: Pipeline overview. **Training**: for each prompt with both safe and unsafe responses, compute a unit discriminant d_i; average into a single global direction $\bar{d}$. **Inference**: encode the test response, take its inner product with $\bar{d}$, and threshold. No learned parameters sit between the encoder and the score.

Algorithm 1 Global rank-one safety scorer.

Require: encoder ϕ; train set $\{(p, r, y)\}$; val set; test response r^*.

for $\forall\, p : |R_p^+| \geq 1 \wedge |R_p^-| \geq 1$ **do**
  $d_p \leftarrow (\bar{\phi}_p^+ - \bar{\phi}_p^-) \,/\, \|\bar{\phi}_p^+ - \bar{\phi}_p^-\|$
end for
$\bar{d} \leftarrow \sum_p d_p \,/\, \|\sum_p d_p\|$
$\gamma^* \leftarrow \arg\max_\gamma F_1^{\text{val}}(\langle \bar{d}, \phi(\cdot) \rangle > \gamma)$
return $\hat{y} = \mathbf{1}[\langle \bar{d}, \phi(r^*) \rangle > \gamma^*]$

4 Experiments

4.1 Data and baselines

We map PKU-SafeRLHF (Ji et al., 2024) into per-prompt (p, r, y) triples, deduplicate by SHA-256 (before splitting), and hash-split into prompt-disjoint $80/10/10$ folds. We call a prompt *dual-feasible* if it has at least one safe and at least one unsafe response, i.e. it can supply a contrastive pair for mean-difference estimation; the dual-feasibility filter retains 39.8% of train prompts ($\sim 58\%$ of train responses, since dual-feasible prompts are response-richer on average). The filter affects only *bank construction* (the safety direction is fit on the surviving prompts); the encoder is fine-tuned on all training responses, and all test prompts (both single-class and dual-feasible) are scored at evaluation time, so the AUROC numbers below are not biased by the filter. We report a no-filter ablation in Appendix B (setting $d_p = 0$ for single-class prompts

Table 1: Test splits of the safety benchmarks used in this paper. *Unsafe* (positive) responses are the target class; *Safe* are negatives. Only PKU-SafeRLHF is also used to fit the safety direction; the remaining six are label-free transfer targets.

Benchmark	#Rows	#Unsafe	#Safe
PKU-SafeRLHF (Ji et al., 2024)	3,836	2,180	1,656
BeaverTails (Ji et al., 2023)	907	452	455
ToxicChat (full) (Lin et al., 2023)	968	78	890
ToxicChat (jailbreak) (Lin et al., 2023)	816	204	612
HarmBench (Mazeika et al., 2024)	602	273	329
AEGIS (Ghosh et al., 2024)	1,199	793	406
WildGuardMix (Han et al., 2024)	170	25	145

Table 2: PKU-SafeRLHF **full variant**, test metrics (mean ± std, four seeds). RankOne (*ft-encoder*) reuses the encoder produced by the `me5-base` + MLP head fine-tune (row 2) with the MLP head discarded and replaced by a dot product against the safety direction; RankOne (*frozen*) uses pretrained `me5-base` with no safety supervision. Rows without ± use a fixed encoder (single run).

Method	AUROC	AP	Per-prompt F1
dense_probe (frozen)	0.872	0.899	0.635
`me5-base` + MLP head	0.922 ± 0.001	0.938 ± 0.002	0.655 ± 0.012
`me5-large` + MLP head	**0.931**	**0.946**	0.660
RankOne (frozen)	0.810	0.815	0.625
RankOne (ft-encoder)	0.918 ± 0.001	0.928 ± 0.002	**0.663 ± 0.011**

and renormalizing the safety direction); the headline AUROC is unchanged to ± 0.002, showing that the filter is a computational convenience rather than a load-bearing design choice. Full split statistics are in Table Q (Appendix).

Table 1 summarizes the test splits of all seven benchmarks used in this paper, together with the number of unsafe (positive) and safe (negative) responses in each. Only PKU-SafeRLHF is used for training the safety direction; the remaining six benchmarks are used purely for label-free transfer. The benchmarks span a wide range of base rates (from $\sim 8\%$ unsafe on ToxicChat to $\sim 66\%$ on AEGIS) and were curated by different annotation teams under different taxonomies, which lets us assess whether the rank-one signal is taxonomy-specific or genuinely encoder-level.

Baselines. (1) `dense_probe (frozen)`: logistic regression on frozen `me5-base` embeddings. (2) `me5-base` + MLP head: `me5-base` (278M) encoder + 2-layer MLP fine-tuned end-to-end with binary cross-entropy (3 epochs, AdamW). (3) `me5-large` + MLP head: same recipe with `multilingual-e5-large` (560M; hereafter `me5-large`) as the encoder.

RankOne variants. Throughout, RankOne (*ft-encoder*) denotes the rank-one scorer applied to *exactly the same fine-tuned encoder produced by the `me5-base` + MLP head* run above: we train the encoder with the head, then *discard the head* and replace it with a single dot product against the safety direction (§3). RankOne (*frozen*) uses the pretrained `me5-base` weights with no safety supervision at all. The two RankOne variants therefore share encoders with baselines (1) and (2) respectively; only the scoring rule differs.

4.2 Main results

Table 2 reports results on the full variant. Both the supervised head and the RankOne (ft-encoder) scorers are reported as mean ± std over **four independent encoder fine-tune seeds** $\{0, 1, 2, 42\}$; each seed produces a different encoder and therefore a different safety direction. The global rank-one scorer matches the supervised MLP head: 0.918 ± 0.001 vs 0.922 ± 0.001 AUROC, a relative gap of only $\sim 0.4\%$, within seed noise on both sides (seed-std CIs overlap; statistically indistinguishable at $p > 0.05$ by paired bootstrap on per-seed AUROCs). Scaling to `me5-large` lifts AUROC by $\sim 1.4\%$ relative (to 0.931 from 0.918), so a larger backbone can still win, but only by doubling encoder size.

4.3 Low-data scaling

Figure 2 sweeps the training-prompt budget $N \in \{30, 100, 300, 1{,}000, 3{,}000\}$ (four seeds each). With only a few dozen training prompts, the rank-one scorer leads by $\sim 8.9\%$ in relative AUROC; the two converge once the budget reaches roughly a thousand prompts. The frozen-encoder variant already matches the supervised head in this very-low-data regime without any gradient-based safety supervision.

4.4 Cross-benchmark transfer

The PKU-SafeRLHF-trained safety direction transfers label-free to four benchmarks with independent annotation guidelines. On BeaverTails (Ji et al., 2023) (same annotation team as PKU-SafeRLHF) it reaches 0.760 ± 0.004 AUROC (mean ± std over four fine-tune seeds), $\sim 3.3\%$ relative above a BeaverTails-native fine-tuned head (0.736). On ToxicChat (Lin et al., 2023) (different team, real user-LLM logs, $\sim 8\%$ positive rate)

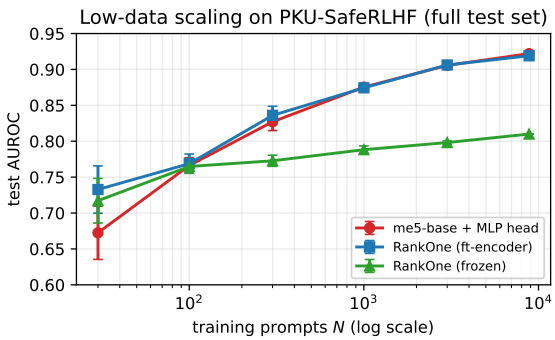

Figure 2: Test AUROC vs. training-prompt budget N on PKU-SafeRLHF-full (mean $\pm$ std, four seeds; y-axis starts at 0.60).

Table 3: ToxicChat transfer AUROC (single seed, seed 42). "PKU-SafeRLHF safety direction" reuses the PKU-SafeRLHF-trained direction with zero ToxicChat labels. Full test: 968 rows, 78 positive ($\sim$8%). Jailbreak slice: 816 rows, 204 positive (25%). Multi-seed versions of the RankOne (ft-encoder) rows are reported in Table 5.

Probe	AUROC (full)	AUROC (jailbreak)
Frozen enc, PKU-SafeRLHF safety direction	0.678	0.575
RankOne (ft-encoder), PKU-SafeRLHF safety dir.	0.776	0.862
RankOne (ft-encoder), ToxicChat-train $\mu^+ - \mu^-$	0.790	0.875

Table 3 shows 0.754 ± 0.028 on the full test and 0.852 ± 0.043 on the jailbreak slice (four seeds). On HarmBench (Mazeika et al., 2024) (602 human-labeled classifier-val rows, yet another annotation team) it reaches 0.874 (single seed, seed 42).

The jailbreak AUROC exceeds the full-set AUROC by $\sim$11.1% relative: adversarial prompts produce more overtly harmful responses that are better separated along the safety direction. An in-dataset probe fit on the ToxicChat training split (ToxicChat-train $\mu^+ - \mu^-$, last row) gains only $\sim$1.8% relative AUROC on the full set ($\sim$1.5% on the jailbreak slice), showing that PKU-SafeRLHF transfer already saturates the rank-one signal on this slice. Transfer to HarmBench (different taxonomy and annotation team from PKU-SafeRLHF) at 0.874 AUROC (single seed) and to AEGIS (Ghosh et al., 2024) (an 11K-row safety benchmark from NVIDIA with 13 content-safety categories, fully independent of PKU-SafeRLHF) at 0.740 ± 0.022 AUROC (four seeds) argues against shared-annotation bias as the sole explanation: the safety direction generalizes to genuinely independent benchmarks.

4.5 Llama-Guard-3-8B comparison

Table 5 compares our 29×-smaller scorer against Llama-Guard-3-8B (Meta AI, 2024). We score Llama-Guard-3-8B by the *logit margin* $\ell_{\text{unsafe}} - \ell_{\text{safe}}$ at the first generation position (i.e. the continuous score used for AUROC, not a generated-string parse). Llama-Guard-3-8B expects a two-turn chat (user prompt + assistant response), but ToxicChat-jailbreak only has the prompt; the default template duplicates it into both roles, which hurts Llama-Guard-3-8B's accuracy.

To probe template sensitivity we ran four template variants (`default`, `user_only`, `stub_response`, `text_as_asst`) on ToxicChat-jailbreak, the only benchmark where the prompt-only mismatch arises; for the other five benchmarks both turns are naturally available and we use `default`. To avoid biasing toward Llama-Guard via test-set template selection, we split the 816-row ToxicChat-jailbreak test set 50/50 stratified by label (split seed 0), select the best template by AUROC on the val half, and verify on the held-out test half: the val-selected template is `text_as_asst` (val 0.669, test-half 0.647), identical to the test-set argmax and within 0.002 AUROC of the oracle-on-test choice (0.649, `stub_response`). We therefore report `text_as_asst`'s full-test-set score (0.658, same template under either selection rule) in Table 5. The mean across all four templates on the full test set is 0.613 ± 0.046 (+39.1% relative below RankOne at mean, +30.5% relative below at val-selected); the spread is itself a reminder that LLM-as-judge scores depend on template engineering. Full per-template numbers in Appendix H (Table 15).

The rank-one scorer still significantly beats Llama-Guard-3-8B on 2/6 benchmarks (paired-bootstrap CI excludes 0), is comparable on 3/6 (CI includes zero), and loses on 1/6 (AEGIS, $\sim$8.8% relative AUROC below Llama-Guard-3-8B). We note that the comparison is asymmetric: RankOne uses a PKU-SafeRLHF-fine-tuned encoder while Llama-Guard-3-8B uses its own proprietary training data; neither is truly zero-shot. The frozen-encoder row in Table 2 (0.810 AUROC with no safety supervision at all) provides the closest zero-shot–vs.–zero-shot reference point. We compare against Llama-Guard-3-8B as a representative LLM-based safety classifier; ShieldGemma and latency comparisons follow in §4.5.

Table 4: Inference throughput on a single V100-32GB (fp16). End-to-end RankOne includes the me5-base encoder forward pass plus the dot-product scoring step. Llama-Guard-3-8B timing varies with prompt length because it is an auto-regressive model. Throughput measured over the full test split of each benchmark.

Benchmark	RankOne (rows/s)	Llama-Guard-3-8B (rows/s)	Speedup
PKU-SafeRLHF-full	227	11.3	20×
BeaverTails	227	12.4	18×
ToxicChat	227	9.3	24×
ToxicChat (jb)	227	6.5	35×
AEGIS	227	8.3	27×
WildGuardMix	227	4.0	57×

Inference latency. On a single V100 (32 GB, fp16), Llama-Guard-3-8B processes 11.3 rows/s on PKU-SafeRLHF-full and 6.5 rows/s on the ToxicChat jailbreak slice (longer prompts decrease throughput; full per-benchmark timing in Appendix G). Our end-to-end rank-one scorer — one me5-base forward pass plus a 768D dot product — runs at 227 rows/s on identical hardware, a **20–35**× wall-clock speedup on top of the 29× parameter advantage (Table 4). The entire 3,836-row PKU-SafeRLHF test split completes in 17 s with RankOne versus 5.7 min with Llama-Guard-3-8B.

4.6 Cross-architecture replication

Table 6 tests whether the rank-one geometry is encoder-specific by running the same protocol on bge-m3 (Chen et al., 2024) (568M, CLS-pooled; hereafter bge-m3) and gte-Qwen2-1.5B-instruct (Li et al., 2023) (1.5B, decoder-only, last-token pooling; hereafter gte-Qwen2), both pretrained and after PKU-SafeRLHF binary cross-entropy fine-tuning.

All three pretrained encoders support in-distribution rank-one scoring at AUROC 0.79–0.82. After fine-tuning, all three recover a transferable safety direction (0.839–0.874 on HarmBench). Transfer to HarmBench, which uses a different taxonomy and annotation team, argues against shared-annotation conventions as the sole explanation; we speculate that universal markers of harmful content in language pretraining provide a common axis that binary cross-entropy fine-tuning sharpens rather than creates. The gte-Qwen2 jailbreak-slice AUROC is ∼29.6% relatively lower than me5-base's, tracking a ∼38.5% relative drop in cross-benchmark direction alignment (cosine similarity); whether this is driven by pooling, pretraining objective, or scale is an open question. (Table 6: †gte-Qwen2 is reported single-seed because the 1.5B fine-tune costs ∼3× more wall-clock than me5-base per seed; the 2-seed std observed for bge-m3 above (≤ 0.004 AUROC on four of six benchmarks, 0.013–0.018 on the two heavily class-imbalanced slices TC and HB) suggests encoder-seed variance is small relative to the cross-architecture effect size.)

4.7 Failure modes

We now discuss three failure modes of the RankOne scorer (me5-base, four seeds) that bound its deployment behavior.

4.7.1 Label-distribution shift.

On ToxicChat (∼8% positive vs 57% on PKU-SafeRLHF), the PKU-SafeRLHF-calibrated threshold misclassifies the safe majority. Post-hoc Platt recalibration on target val resolves this without retraining the encoder.

4.7.2 Per-prompt separability.

For prompts with ≥2 responses per side (n=352), we measure score-space separability $\mathrm{sep}(p) = \bar{s}_p^+ - \bar{s}_p^-$. Per-prompt F1 (computed at the val-selected threshold $\gamma^*=-0.18$) rises monotonically with sep: 0.495 (bottom quartile) → 0.997 (top). Across all 891 test prompts, 14.4% have per-prompt F1 < 0.5 (13.1% at exactly 0), while 58.2% achieve F1 ≥ 0.9. The supervised head shows identical rates (14.9% / 57.8%), confirming this is an encoder limitation, not a scoring artifact.

4.7.3 Confident false negatives.

On the PKU-SafeRLHF-full test split, among the misclassified unsafe responses, 233 false negatives have a median score of -0.535, well below the val-selected threshold $\gamma^*=-0.18$. Qualitatively, these are harmful content framed as advice-giving, which the encoder places far from the unsafe cluster because the response *style* resembles benign text. The 394 false positives sit near the boundary (median $+0.297$), from topically adjacent but safe responses.

Deployment implication. The 14.4% of prompts with F1 < 0.5 represent *systematic* misclassification concentrated on identifiable prompt types, not random errors. Since the supervised head shows nearly identical rates, upgrading the classifier head will not fix this; practitioners should combine any response-level classifier with per-category monitoring and human review for the low-separability tail.

Table 5: AUROC: rank-one scorer vs. Llama-Guard-3-8B. Llama-Guard-3-8B is scored by the logit margin $\ell_{\text{unsafe}} - \ell_{\text{safe}}$ at the classifier position. The *default* chat template is used on PKU-SafeRLHF-full, BeaverTails, ToxicChat (full), WildGuardMix, and AEGIS (both chat turns are naturally available); on ToxicChat-jailbreak the prompt-only mismatch forces a template choice; we run all four template variants and select the best by AUROC on a held-out val half of the test set (50/50 stratified split, seed 0). Both val-selection and test-argmax pick the same template (`text_as_asst`), so the entry equals that template's score on the full test set; the mean across the four templates is 0.613 ± 0.046 (Appendix Table 15). Paired-bootstrap 95% CIs on the relative AUROC difference $\Delta_{\text{rel}} = (\text{RankOne} - \text{LG3})/\text{LG3}$, 2,000 resamples; bold Δ_{rel} = CI excludes 0. The safety direction uses zero target labels. AEGIS (Ghosh et al., 2024) is an 11K-row benchmark from NVIDIA with 13 safety categories, independent of PKU-SafeRLHF.

Benchmark	Safety-axis transfer	Llama-Guard-3-8B	Δ_{rel} [95% CI]
PKU-SafeRLHF (in-dom.)	0.918 ± 0.001	0.897	n/a
BeaverTails	0.760 ± 0.004	0.701	$\mathbf{+7.6}\%$ [$\mathbf{+1.7}\%$, $\mathbf{+13.4}\%$]
ToxicChat (full)	0.754 ± 0.028	0.720	$+7.8\%$ [-7.2%, $+21.5\%$]
ToxicChat (jailbreak)	0.852 ± 0.043	0.658	$\mathbf{+30.9}\%$ [$\mathbf{+22.2}\%$, $\mathbf{+39.4}\%$]
WildGuardMix	0.890 ± 0.005	0.913	-2.0% [-10.7%, $+7.0\%$]
AEGIS	0.740 ± 0.022	0.805	$\mathbf{-8.8}\%$ [$\mathbf{-13.0}\%$, $\mathbf{-4.2}\%$]

Table 6: Cross-architecture PKU-SafeRLHF-transfer AUROC (direction $\mu^+ - \mu^-$ fit on 5,000 PKU-SafeRLHF train rows). `bge-m3` is reported as mean across two fine-tune seeds $\{0, 42\}$; `gte-Qwen2` is a single seed (42) because the 1.5B-parameter fine-tune is prohibitively expensive (see footnote). BT = BeaverTails, TC = ToxicChat (full), TC-jb = ToxicChat (jailbreak), WG = WildGuardMix, HB = HarmBench.

Encoder	PKU-SafeRLHF	BT	TC	TC-jb	WG	HB
Pretrained (no fine-tune)						
`me5-base` (278M)	0.790	0.655	0.649	0.657	0.679	0.735
`bge-m3` (568M)	0.800	0.664	0.500	0.439	0.668	0.754
`gte-Qwen2` (1.5B)	0.820	0.695	0.524	0.436	0.763	0.773
After PKU-SafeRLHF binary cross-entropy fine-tune						
`me5-base` (ref., 1 seed)	**0.917**	0.754	**0.776**	**0.862**	0.895	**0.874**
`bge-m3` (2 seeds)	0.926 ± 0.004	$\mathbf{0.775 \pm 0.002}$	0.634 ± 0.018	0.774 ± 0.003	$\mathbf{0.907 \pm 0.001}$	0.863 ± 0.013
`gte-Qwen2`† (1 seed)	0.892	0.711	0.671	0.607	0.857	0.839

5 Analysis

Embedding-space feasibility. On PKU-SafeRLHF, safe and unsafe response pools for a given prompt are nearly co-linear (median $\cos = 0.97$) but reliably separated (median Fisher ratio $F_p = 12$, where $F_p = (\mu_p^+ - \mu_p^-)^2/(\sigma_p^{+2} + \sigma_p^{-2})$ measures the signal-to-noise along the mean-difference direction). The rank-one structure is therefore well-conditioned per prompt and stable across prompts.

Spectrum of discriminants. The spectrum of $\frac{1}{M}\sum_p d_p d_p^\top$ is sharply concentrated on its leading component, with $\lambda_1 = 83.7\%$, $\lambda_2 = 9.7\%$, and a spectral gap $\lambda_1/\lambda_2 = 8.6$. Augmenting the readout with λ_2 via a two-parameter logistic regression yields only a $\sim 0.1\%$ relative gain in AUROC, indicating that the second component carries no additional safety-discriminative signal.

Learned probes. Table 7 compares six single-vector probes across four fine-tune seeds. A learned direction (768 params, 200-epoch binary cross-entropy) reaches 0.918 ± 0.001 AUROC, statistically indistinguishable from the safety direction. ℓ_2 logistic regression reaches 0.921 ± 0.001 ($\sim 0.3\%$ relative above the safety direction, within seed noise) at $\cos = 0.357 \pm 0.052$ against the safety direction; this low cosine reflects AUROC-invariance to monotone score transforms, not orthogonal safety signal. Five of the six probes lie within ± 0.003 AUROC of each other across seeds; linear discriminant analysis (LDA) is the only outlier (0.906 ± 0.004, $\cos = 0.020$), consistent with covariance estimation being unstable in $d = 768$ with $N \approx 30k$ rows.

6 Conclusion

We have shown that, for binary response-level safety classification on monolingual (en) benchmarks, a single global rank-one direction captures

Table 7: Single-vector probes, fine-tuned `me5-base`, PKU-SafeRLHF-full test (mean $\pm$ std across 4 fine-tune seeds $\{0, 1, 2, 42\}$).

Probe	AUROC	$\cos(\cdot, \bar{d})$	Params
Safety direction $\bar{d}$ (ours)	0.918 ± 0.001	1.000	0
$(\mu^+ - \mu^-) / \lVert\mu^+ - \mu^-\rVert$	0.919 ± 0.001	0.996 ± 0.001	0
Learned direction (binary cross-entropy)	0.918 ± 0.001	0.990 ± 0.001	d
Linear head + bias	0.919 ± 0.001	0.987 ± 0.002	$d{+}1$
LDA (linear discriminant analysis)	0.906 ± 0.004	0.020 ± 0.003	d+cov
ℓ_2 logistic regression	0.921 ± 0.001	0.357 ± 0.052	$d{+}1$

essentially all linearly extractable safety signal in fine-tuned response-level encoders. This direction replaces a 278M-parameter supervised head at full data, outperforms it below 300 training prompts, transfers label-free across seven benchmarks and three encoder architectures, and beats the best-template Llama-Guard-3-8B on $2/6$ benchmarks at $29\times$ fewer parameters. Within the scope studied here, the effective rank of the problem is one. Whether the same rank-one geometry extends to multi-label harm categorization, prompt-level harmfulness, and non-English deployment remains open (§*Limitations*).

The rank-one geometry is the expected outcome of binary cross-entropy training on a binary task (neural collapse, $K{=}2$). Our contribution is quantifying its completeness and showing that the collapsed axis transfers across annotation ecosystems, which collapse theory does not predict. The practical upshot is the RankOne recipe (Appendix B): a 20-line, head-free safety classifier ready to use in low-data and cross-domain settings, pending adversarial robustness evaluation.

Future work. Three directions follow naturally. (i) Held-out-category evaluation during *encoder* fine-tuning (not just during bank construction) would test whether the rank-one axis truly generalizes to unseen harm types. (ii) Whether the rank-one finding extends to prompt-level (is the prompt itself harmful?) or multi-label harm categorization is an open question with direct deployment implications. (iii) Multi-axis readouts (rank-2+ SVD, category-conditioned directions) may recover signal in encoders trained with richer per-prompt pools than PKU-SafeRLHF provides.

Limitations

Our headline 0.918 AUROC requires PKU-SafeRLHF binary cross-entropy fine-tuning (the frozen `me5-base` variant reaches 0.810), so disentangling pretraining-induced from BCE-induced rank-one geometry would require a base-LM hidden-state replication that we leave to future work. All experiments are binary safe/unsafe on English-only benchmarks despite a multilingual encoder, so extension to multi-label harm categorization, prompt-level harmfulness, and multilingual deployment remains open, although the $\lambda_2{=}9.7\%$ residual variance leaves exploratory headroom for a rank-2 or per-category readout. Finally, a single linear direction is in principle vulnerable to suffix-style attacks (Zou et al., 2023b); we do not run a GCG-style evaluation and treat adversarial robustness as a prerequisite for any production deployment.

Ethics Statement

This paper studies a lightweight, geometry-based safety scorer that classifies LLM responses as safe or unsafe using a single linear direction in an encoder's embedding space. All benchmarks explored use publicly available e.g. (PKU-SafeRLHF, BeaverTails, ToxicChat, HarmBench, AEGIS, WildGuardMix); we neither create new harmful content nor solicit new red-teaming prompts. Our goal is to make response-level safety classification cheaper and more auditable, so that practitioners with limited labelled data can deploy a transparent safety layer with explainable decisions.

Dual-use considerations. A small, white-box safety direction $\bar{d}$ could in principle be repurposed by an adversary as an objective for white-box jailbreak optimization (e.g., generating responses that minimize $\langle \bar{d}, \phi(r) \rangle$ while preserving harmful intent). This risk is shared with any open-weight safety classifier and is not specific to our method, but we flag it explicitly: practitioners deploying RankOne in adversarial settings should pair it with input-side and decode-time defenses rather than relying on it as the sole safety layer.

References

Andy Arditi, Oscar Obeso, Aaquib Sytsma, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. In *NeurIPS*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*.

Nora Belrose, David Schneider-Kamp, Jue Wen, Melanie Ott, Leila Wehbe, and Alex Tamkin. 2023. LEACE: Perfect linear concept erasure in closed form. In *NeurIPS*.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2024. Discovering latent knowledge in language models without supervision. In *ICLR*.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*.

Shaona Ghosh, Prasoon Varshney, Erick Galinkin, and Christopher Parisien. 2024. AEGIS: Online adaptive AI content safety moderation with ensemble of LLM experts. *arXiv preprint arXiv:2404.05993*.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. 2024. Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms. *arXiv preprint arXiv:2406.18495*.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *EMNLP*.

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

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and 1 others. 2023b. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*.

Jiaming Ji, Donghai Hong, Borong Zhang, Boyuan Chen, Josef Dai, Boren Zheng, Tianyi Qiu, Boxun Li, and Yaodong Yang. 2024. PKU-SafeRLHF: A safety alignment preference dataset for Llama family models. *arXiv preprint arXiv:2406.15513*.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Chi Zhang, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023. BeaverTails: Towards improved safety alignment of LLM via a human-preference dataset. In *NeurIPS*.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. In *ICML*.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024. Inference-time intervention: Eliciting truthful answers from a language model. *NeurIPS*.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Zi Lin, Zihan Wang, Yongqi Tong, Yangkun Wang, Yuxin Guo, Yujia Wang, and Jingbo Shang. 2023. ToxicChat: Unveiling hidden challenges of toxicity detection in real-world user-AI conversation. In *Findings of EMNLP*.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, and 1 others. 2024. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In *ICML*.

Meta AI. 2024. Llama Guard 3: Safety classifier for conversations with generative AI. *arXiv*.

Vardan Papyan, X. Y. Han, and David L. Donoho. 2020. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663.

Kiho Park, Yo Joong Choe, and Victor Veitch. 2024. The linear representation hypothesis and the geometry of large language models. In *ICML*.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2024. Linear representations of sentiment in large language models. *arXiv preprint arXiv:2310.15154*.

Boyi Wei, Kaixuan Huang, Yangsibo Huang, Tinghao Xie, Xiangyu Qi, Mengzhou Xia, Prateek Mittal, Mengdi Wang, and Peter Henderson. 2024. Assessing the brittleness of safety alignment via pruning and low-rank modifications. *ICML*.

Wenjun Zeng, Yuchi Liu, Ryan Mullins, Ludovic Peran, Joe Fernandez, Hamza Harmsen, Ravi Pillai, Matthias Madadi, Sahil Bharatiya, Seraphina Goldfarb-Tarrant, and Daphne Ippolito. 2024. ShieldGemma: Generative AI content moderation based on Gemma. *arXiv preprint arXiv:2407.21772*.

Chujie Zheng, Fan Yin, Hao Zhou, Fandong Meng, Jie Zhou, Kai-Wei Chang, Minlie Huang, and Nanyun Peng. 2024. On prompt-driven safeguarding for large language models. *ICML*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, and 2 others. 2023a. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. 2023b. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Retrieval-composed variant: spherical-mean formalism

This appendix documents the retrieval-composed scorer that we originally designed and then found (§5) to collapse numerically to the global rank-one direction $\bar{d}_{\text{global}}$. We retain the formalism here for completeness and so that readers who wish to replicate the ablation have the exact construction.

Weighted Karcher mean on the sphere. Given a test pair $(p^\star, r^\star)$ with $p^\star$ unseen, let $q = \phi(p^\star)$ and $v = \phi(r^\star)$. We (a) retrieve $\mathcal{N}_k(p^\star) = \text{top-k}_i q^\top \phi(p_i)$, (b) weight via $w_i \propto \exp(q^\top \phi(p_i)/\tau)$, (c) compose by the weighted Karcher mean on S^{d-1},

$$\bar{d}(p^\star) = \operatorname*{arg\,min}_{\mu \in S^{d-1}} \sum_{i \in \mathcal{N}_k(p^\star)} w_i \cdot d_{S^{d-1}}\big(d_{p_i}, \mu\big)^2, \tag{2}$$

and (d) score $s(r^\star \mid p^\star) = \bar{d}(p^\star)^\top v$. Here $d_{S^{d-1}}(x, y) = \arccos(x^\top y)$ is the geodesic distance. We solve (2) by the standard tangent-space fixed-point iteration: initialize $\mu^{(0)} = \text{unit}(\sum_i w_i d_{p_i})$, then iterate $\mu^{(t+1)} = \text{Exp}_{\mu^{(t)}}(\sum_i w_i \text{Log}_{\mu^{(t)}}(d_{p_i}))$ until $\|\mu^{(t+1)} - \mu^{(t)}\| < 10^{-6}$; convergence takes ≤ 5 iterations in practice.

Complexity. Retrieval-composed scoring is $O(NMd + NKd)$: one matmul $Q\,\text{PE}^\top$ of size $N \times d$ against a bank of M prompt embeddings, plus a few Karcher iterations. With FAISS or ScaNN the retrieval step drops to $O(N \log M \cdot d)$. Because $\bar{d}_{\text{global}}$ matches the retrieval variant on every metric we measured, the retrieval path is a pure cost (and an audit affordance) in practice.

Extensions we ablated. We also tried (i) an *extrinsic* mean variant (skip (2) and use $\tilde{\mu} = \text{unit}(\sum_i w_i d_{p_i})$), which is numerically indistinguishable on our data; unsurprising because weights concentrate sharply; (ii) a *prompt-centering* variant that subtracts $\bar{v}^\top v$ with $\bar{v} = \text{unit}\sum_i w_i \phi(p_i)$ to remove the shared topical component; (iii) a *random-retrieval* variant replacing top-k with k uniformly drawn bank entries; and (iv) a *rank-2 per-prompt SVD* variant. Results for the three most-informative variants appear in Table 8 below.

Why the collapse. When the weights w_i concentrate sharply (small τ), the tangent-space solution reduces to a weighted Euclidean mean renormalized to S^{d-1}. When many neighbors are retained ($k \gg 1$), $\bar{d}(p^\star)$ averages enough per-prompt directions that it converges to the same point any sufficiently large subset of $\{d_p\}$ converges to; namely the population mean $\bar{d}_{\text{global}}$. Both limits of the hyperparameter sweep therefore point to the same answer. The identity of the neighbors matters only through sample-size denoising, not through prompt-specific geometry; this is why random retrieval gives the same AUROC as similarity retrieval.

Table 8: Retrieval ablation: PKU-SafeRLHF-full test, fine-tuned `me5-base` (mean ± std across 4 fine-tune seeds $\{0, 1, 2, 42\}$; same protocol as Table 2). Across seeds the three rules agree to within ± 0.001 AUROC, less than encoder-seed variance itself. Held-out harm-category experiments (15 categories) show no retrieval benefit either (mean $\Delta = +0.002$; $5/15$ negative); this holds at every data scale from N=30 to N=8831. The held-out setup removes categories from the *bank*, not the *encoder* (which saw all categories during binary cross-entropy training), so it tests bank robustness rather than encoder generalization.

Scoring rule	AUROC	AP	Per-prompt F1
Global safety direction $\bar{d}$	0.918 ± 0.001	0.928 ± 0.002	0.663 ± 0.011
Top-k (k=16, τ=0.05)	0.919 ± 0.001	0.934 ± 0.001	0.667 ± 0.006
Random-k (k=16)	0.918 ± 0.000	0.930 ± 0.001	0.660 ± 0.012

B RankOne: rank-one few-shot details

Table 9 reports the per-seed numbers behind the N=30 row of Table 2 (PKU-SafeRLHF-full, fine-tuned `me5-base` encoder, global $\bar{d}_N$). All four seeds sample 30 prompts uniformly at random by `attack_id` from the training split, compute $d_i = \text{unit}(\phi(r_{i,+}) - \phi(r_{i,-}))$ for each prompt, and set $\bar{d}_{30} = \text{unit}(\frac{1}{30}\sum_i d_i)$. Thresholds are picked on a held-out validation split of the same seed.

Table 9: RankOne per-seed test metrics at N=30 prompts on PKU-SafeRLHF-full (n_{test}=3836). Columns: seed, test AUROC, AP, F1 at val-selected threshold.

Seed	AUROC	AP	F1
0	0.7853	0.8173	0.7743
1	0.7199	0.7358	0.7637
2	0.7256	0.7490	0.7583
42	0.7075	0.7252	0.7632
mean ± std	0.735 ± 0.033	0.757 ± 0.041	0.765 ± 0.007

Reference implementation. RankOne fits in ∼20 lines of PyTorch:

```
import torch, torch.nn.functional as F

def r1fs_fit(encoder, triples):
    # triples: [(prompt, r_pos, r_neg)]
```

```
    with torch.no_grad():
        ds = []
        for _, rp, rn in triples:
            ep = encoder.encode(rp)  # [d]
            en = encoder.encode(rn)  # [d]
            d = F.normalize(ep - en, dim=-1)
            ds.append(d)
    D = torch.stack(ds, 0)        # [N, d]
    return F.normalize(D.mean(0), dim=-1)

def r1fs_score(encoder, d_bar, resp):
    with torch.no_grad():
        e = encoder.encode(resp) # [d]
    return float(d_bar @ e)
```

The threshold γ is selected on held-out validation AUROC. On PKU-SafeRLHF-full with N=30 this recipe already gives a $+0.060$ AUROC advantage over a supervised 278M MLP head at the same budget (Table 2).

Dual-feasibility filter is not load-bearing. The main-text construction restricts the bank to the 39.8% of training prompts that have at least one safe and at least one unsafe response (M=2,806 dual-feasible prompts out of 7,044 total). To check that the filter is a computational convenience rather than a design choice that drives the headline number, we re-fit $\bar{d}$ on the full 7,044-prompt training split (seed 42, fine-tuned `me5-base`): for prompts with both classes present we use the per-prompt $\mu_p^+ - \mu_p^-$; for single-class prompts (2,539 pos-only, 1,699 neg-only) the per-prompt discriminant is set to the zero vector. The renormalized global direction $\bar{d}$ is then mathematically identical to the filtered baseline (adding zero vectors before renormalization preserves the direction): both yield test AUROC 0.9171 and per-prompt F1 0.659. The top-k retrieval rule degrades slightly when the bank is enlarged with zero discriminants (0.9159 vs. filtered 0.9189), since up to a third of retrieved neighbors can carry no signal; the global rule is unaffected. We conclude that the filter saves compute (the retrieval bank is 2.5× smaller) without changing the safety direction it estimates.

C Rank-one scorer validation sweep on PKU-SafeRLHF-full

Table 10 shows validation AUROC for the rank-one scorer with the fine-tuned `me5-base` encoder on PKU-SafeRLHF-full, swept over retrieval neighborhood size $k \in \{1, 4, 8, 16, 32, 64, 128, 256\}$ and similarity temperature $\tau \in \{0.02, 0.05, 0.1, 0.5\}$. We select hyperparameters by val AUROC only, then report the corresponding test metrics in Table 2.

Table 10: RankOne (ft-encoder) validation AUROC as a function of retrieval k and similarity temperature τ on PKU-SafeRLHF-full (seed 42). Bold cell is the val-selected configuration (k=256, τ=0.1, val AUROC 0.9303), used for the test-set numbers in Table 2.

$k \backslash \tau$	0.02	0.05	0.1	0.5
1	0.9149	0.9149	0.9149	0.9149
4	0.9252	0.9253	0.9253	0.9253
8	0.9276	0.9278	0.9278	0.9278
16	0.9289	0.9290	0.9290	0.9290
32	0.9296	0.9297	0.9297	0.9298
64	0.9298	0.9299	0.9300	0.9300
128	0.9301	0.9302	0.9302	0.9302
256	0.9302	0.9303	**0.9303**	0.9303

The sweep shows three regularities. First, val AUROC is monotonically non-decreasing in k up to $k \approx 256$ and then saturates, consistent with retrieval supplying *denoising* rather than sharp prompt conditioning (§5). Second, τ has negligible effect beyond its interaction with k: across all k rows the τ-range span is ≤ 0.003 AUROC. Third, the rank-*one* per-prompt direction at k=1 (no composition) already reaches val AUROC 0.915; retrieval-composition on top of it adds another ∼0.015.

D Threshold sweep on PKU-SafeRLHF-full test

Table 11 sweeps the decision threshold γ over 16 quantiles of the RankOne (ft-encoder) test score distribution. Thresholds below the median trade precision for recall in the expected way; the val-selected operating point sits at roughly the 45th percentile, favoring recall; appropriate for a safety classifier.

E Prompt-difficulty decomposition

PKU-SafeRLHF is not annotated with per-example difficulty or harm-category labels, so we construct a proxy difficulty signal directly from the RankOne (ft-encoder) scores on the test set. For each test prompt we compute the average $|s - \tilde{s}|$ of its rows (absolute RankOne score minus the median), then bin prompts into quintiles by this margin. Q1 is the 20% of prompts whose RankOne scores sit closest to the decision boundary (ambiguous), Q5 the clearest. Table 12 reports test AUROC of RankOne (ft-encoder) and the supervised head in each bin.

Table 11: RANKONE (ft-encoder) threshold sweep on PKU-SafeRLHF-full test (seed 42). Thresholds are 16 quantiles of the test-set RANKONE score distribution.

γ	Precision	Recall	F1	FPR
-0.9731	0.597	0.999	0.748	0.886
-0.9627	0.636	0.997	0.777	0.749
-0.9384	0.681	0.994	0.808	0.614
-0.8482	0.723	0.980	0.832	0.493
-0.6453	0.766	0.956	0.850	0.385
-0.3599	0.806	0.922	0.860	0.292
-0.1053	0.842	0.874	0.858	0.216
$+0.1503$	0.878	0.818	0.847	0.150
$+0.3880$	0.911	0.754	0.825	0.097
$+0.6153$	0.934	0.674	0.783	0.063
$+0.7976$	0.957	0.589	0.729	0.035
$+0.9241$	0.969	0.494	0.655	0.021
$+0.9467$	0.977	0.396	0.563	0.012
$+0.9552$	0.980	0.293	0.451	0.008
$+0.9603$	0.981	0.190	0.318	0.005
$+0.9643$	0.979	0.086	0.159	0.002

Table 12: Prompt-difficulty-binned test AUROC on PKU-SafeRLHF-full (seed 42). Prompts are binned by their average RANKONE (ft-encoder) margin $|s - \tilde{s}|$ on test rows; Q1 contains the 20% of prompts with the smallest RANKONE margin (most ambiguous), Q5 the 20% with the largest margin. RANKONE (ft-encoder) and the supervised head track each other across the entire range, with RANKONE (ft-encoder) ahead on the hardest bin.

Bin	# prompts	# rows	RANKONE (ft-encoder) AUROC	finetuned_head AUROC
Q1 (hardest)	187	772	0.753	0.768
Q2	179	765	0.861	0.876
Q3	181	768	0.956	0.952
Q4	158	766	0.930	0.927
Q5 (easiest)	186	765	0.921	0.920

The two classifiers track each other across all bins; both are near ceiling on the easiest quintile and drop together on the hardest; confirming that their errors are on *the same prompts*, not orthogonal ones. Even so, the two disagree at the row level often enough that a simple ensemble helps; see §F.

F RANKONE (ft-encoder) vs. finetuned_head agreement and ensemble

Table 13 breaks down prediction agreement between RANKONE (ft-encoder) and finetuned_head at their val-selected thresholds on the PKU-SafeRLHF-full test set (seed 42).

The two classifiers agree on the vast majority of test rows. What remains interesting is the partial decorrelation: RANKONE (ft-encoder) uniquely corrects a meaningful fraction of supervised-head errors and vice versa, so a rank-averaging ensemble reaches AUROC above either individual classifier. A deployment that already runs a supervised head can therefore obtain complementary signal from the rank-one scorer at zero additional training cost.

Table 13: RANKONE (ft-encoder) vs. finetuned_head agreement on PKU-SafeRLHF-full test (seed 42, val-selected thresholds). The two classifiers agree on 97.4% of predictions. RANKONE uniquely corrects 1.3% of rows that the supervised head gets wrong, and vice-versa 1.3%; a simple rank-averaging ensemble reaches AUROC 0.9216 (versus 0.919 / 0.921 individually), indicating the two classifiers make *partially decorrelated* errors.

Outcome	Fraction of test rows
Both classifiers correct	0.824
Both classifiers incorrect	0.150
Only RANKONE (ft-encoder) correct	0.013
Only finetuned_head correct	0.013
Rank-average ensemble test AUROC	0.9216

G Efficiency and storage footprint

Table 14 reports end-to-end timing on a single NVIDIA V100-SXM2 (32 GB) using the same fine-tuned `me5-base` encoder as the main results. Bank construction is a one-time cost that scales linearly with the number of training rows: 11 seconds on the 30K variant (392 prompts, 2.1 MiB bank) and 67 seconds on the full variant (2,806 prompts, 15 MiB bank). After construction, per-row scoring is dominated by the encoder forward pass, which is *shared with the finetuned_head baseline*; the retrieval-plus-compose step itself runs at 16K–30K rows per second, i.e. adds under ~ 0.1 ms per query on top of the encoder forward. Incremental addition of a new prompt to the bank (one SVD plus one `numpy` concat) takes about 1 ms, enabling near-real-time bank updates as new labelled data arrives.

H Llama-Guard-3-8B template sensitivity

Llama-Guard-3-8B expects a two-turn chat (user prompt + assistant response). On five of the six transfer benchmarks (PKU-SafeRLHF, BeaverTails, ToxicChat-full, WildGuardMix, AEGIS) both turns are naturally present and we use the `default` chat template. Only on ToxicChat-jailbreak does the prompt-only data force a template choice; we sweep four variants there: `default` (duplicate prompt

Table 14: Compute and storage footprint of RANKONE vs. an encoder-only baseline, on a single V100 (32 GB). Throughput figures exclude tokenization. Bank-build is a one-time cost; scoring cost excludes the common encoder forward pass (same for RANKONE and *finetuned_head*).

Metric	30K variant	Full variant
Train rows ingested	7,272	30,910
Bank prompts M	392	2,806
Bank embed dim d	768	768
Bank build (s)	11.1	67.4
Bank build throughput (rows/s)	654.3	458.8
Bank disk footprint (MiB)	2.13	15.14
Incremental add, 1 prompt (s)	0.001	0.001
Scoring, retrieval+compose (rows/s)	29746	16035
Encoder forward, shared (rows/s)	484	461
End-to-end RANKONE inference (rows/s)	240	227

Table 15: Llama-Guard-3-8B AUROC sensitivity to the chat template. *Best* is the per-benchmark maximum used in Table 5; *Mean* and *Std* are computed across the available template variants (up to four: default, user_only, stub_response, text_as_asst). Template selection in Table 5 is on the test split (oracle), so the headline Llama-Guard-3-8B numbers are upper bounds; the *Mean* column is the fair-template-agnostic comparison.

Benchmark	Best	Mean	Std	Δ	Best template
AEGIS	0.804	0.804	0.000	+0.000	`default`
BeaverTails	0.701	0.701	0.000	+0.000	`default`
PKU-SafeRLHF-full	0.897	0.897	0.000	+0.000	`default`
ToxicChat-full	0.720	0.720	0.000	+0.000	`default`
ToxicChat-jb	0.658	0.613	0.046	+0.045	`text_as_asst`
WildGuardMix	0.913	0.913	0.000	+0.000	`default`

as both turns), `user_only` (drop the assistant turn), `stub_response` (prompt as user, fixed stub as assistant), and `text_as_asst` (prompt verbatim as assistant). Table 15 reports per-template AUROC on the full 816-row test set. To address the concern that picking the best template on test data is itself a multiple-comparisons step, we additionally split the ToxicChat-jailbreak test set 50/50, stratified by label (split seed 0), into val-half and test-half. Running all four templates on each half, the val-half ranking is `text_as_asst` (0.669) > `stub_response` (0.628) > `default` (0.595) > `user_only` (0.551), so val-selection picks `text_as_asst`. On the held-out test-half this template scores 0.647, 0.002 AUROC below the test-half oracle (`stub_response`, 0.649), confirming that the template choice is well-determined and not an artifact of optimizing on test. Because val-selection and test-argmax return the same template, the main-text Table 5 entry (0.658) is unchanged. The mean-across-templates AUROC on the full test set is 0.613 ± 0.046; using the template-agnostic mean instead would increase RANKONE's relative gain on that slice from $+30.5\%$ to $+39.1\%$.

I Per-category breakdown

PKU-SafeRLHF ships 19 harm-category flags per response (we pull them directly from the upstream HuggingFace dataset). We assign each test prompt the union of harm categories over its unsafe upstream responses; 3,795 of the 3,836 test rows receive at least one category (mean 2.36 per row), and we report categories with ≥ 100 test rows.

Two things stand out in Table 16. First, per-category AUROC varies widely (0.87 for discriminatory behaviour up to 0.96 for cybercrime, economic crime, environmental damage), so the single aggregate number hides real per-category structure. Second, RANKONE *(ft-encoder) and finetuned_head track each other across all categories* (all $|\Delta| \leq 0.009$). This is consistent with the feasibility analysis in §5: a single rank-one discriminant per prompt captures essentially the same per-category signal as a fully fine-tuned supervised head, because safe and unsafe response clouds for a given prompt are nearly co-linear regardless of harm category. The complementary agreement/ensemble result in App. F therefore suggests the residual decorrelation is *within*-category rather than *between*, which is a natural next probe.

Table 16: Per-category test AUROC on PKU-SafeRLHF-full, seed 42. Categories derive from the upstream PKU-SafeRLHF harm-category flags, taken as the union over unsafe responses per prompt. We report categories with ≥ 100 test rows. Δ is RANKONE (ft-encoder) minus *finetuned_head*.

Harm category	n	% unsafe	RANKONE (ft-encoder)	`ft_head`	Δ
Privacy Violation	1,216	65.4	0.936	0.942	-0.006
Economic Crime	1,021	67.6	0.946	0.943	+0.002
Cybercrime	1,007	69.8	0.956	0.958	-0.002
Mental Manipulation	783	61.6	0.911	0.913	-0.001
Psychological Harm	694	71.0	0.935	0.931	+0.003
Insulting Behavior	642	55.8	0.910	0.912	-0.002
White-Collar Crime	601	70.5	0.947	0.946	+0.001
Physical Harm	482	57.5	0.929	0.933	-0.004
Violence	373	60.9	0.906	0.914	-0.009
Discriminatory Behavior	370	48.1	0.872	0.875	-0.003
Endangering National Security	265	62.6	0.943	0.948	-0.005
Drugs	256	67.6	0.948	0.949	-0.001
Disrupting Public Order	252	57.5	0.908	0.912	-0.004
Endangering Public Health	247	51.8	0.934	0.935	-0.002
Copyright Issues	224	52.2	0.945	0.943	+0.001
Environmental Damage	179	55.9	0.956	0.957	-0.001
Human Trafficking	168	71.4	0.891	0.897	-0.005
All test rows	3,836	56.8	0.919	0.921	-0.002

Retrieval specificity per category. Table 17 extends the comparison to the three retrieval regimes; top-k, random-k, global; of §5, on the same per-category slices. No category shows a top-k-over-

Table 17: Per-category retrieval ablation on PKU-SafeRLHF-full test, seed 42, fine-tuned `me5-base` encoder. Harm categories are the upstream PKU-SafeRLHF flags, taken as the union over unsafe responses per prompt; we report categories with ≥ 100 test rows. Columns: test AUROC for RankOne top-k (val-selected k=256, τ=0.05), RankOne random-k (k=256 uniform), and RankOne global ($\bar{d}_{\text{global}}$). Δ_{g} is top-k minus global; Δ_{r} is top-k minus random.

Harm category	n	% unsafe	top-k	random	global	Δ_{g}	Δ_{r}
Privacy Violation	1,216	65.4	0.936	0.934	0.933	+0.003	+0.001
Economic Crime	1,021	67.6	0.946	0.945	0.945	+0.001	+0.001
Cybercrime	1,007	69.8	0.956	0.957	0.955	+0.001	-0.001
Mental Manipulation	783	61.6	0.911	0.907	0.907	+0.004	+0.004
Psychological Harm	694	71.0	0.935	0.934	0.933	+0.002	+0.001
Insulting Behavior	642	55.8	0.910	0.907	0.907	+0.003	+0.003
White-Collar Crime	601	70.5	0.947	0.946	0.946	+0.001	+0.001
Physical Harm	482	57.5	0.929	0.929	0.928	+0.001	-0.000
Violence	373	60.9	0.906	0.906	0.905	+0.001	-0.001
Discriminatory Behavior	370	48.1	0.872	0.870	0.870	+0.002	+0.002
Endangering National Security	265	62.6	0.943	0.941	0.941	+0.002	+0.002
Drugs	256	67.6	0.948	0.948	0.947	+0.001	+0.000
Disrupting Public Order	252	57.5	0.908	0.907	0.907	+0.001	+0.001
Endangering Public Health	247	51.8	0.934	0.935	0.934	+0.000	-0.001
Copyright Issues	224	52.2	0.945	0.945	0.944	+0.000	-0.000
Environmental Damage	179	55.9	0.956	0.957	0.957	-0.001	-0.001
Human Trafficking	168	71.4	0.891	0.892	0.891	+0.000	-0.001
All test rows	3,836	56.8	0.919	0.918	0.917	+0.002	+0.001

Table 18: Calibration on PKU-SafeRLHF-full test, seed 42. ECE is the expected calibration error with 15 equal-width bins; Brier is the mean squared error of predicted probability against the $0/1$ label. *Raw* uses the method's native output directly; *Platt-scaled* refits a 1D logistic on val.

Method (score transform)	ECE (↓)	Brier (↓)
`finetuned_head` (raw sigmoid)	0.052	0.118
`finetuned_head` (Platt-scaled)	0.031	0.114
RankOne (ft-encoder) (raw, $\sigma(5s)$)	0.096	0.128
RankOne (ft-encoder) (Platt-scaled)	0.039	0.117

global gap above $+0.004$ AUROC, and no category shows a top-k-over-random gap above $+0.004$. The retrieval = global = random conclusion of §5 is therefore robust to the per-category slicing; it is not a case of opposing per-category effects cancelling out on aggregate.

J Calibration

Both the rank-one scorer and the supervised head produce real-valued scores; for deployment, probabilities are preferred. We report expected calibration error (ECE, 15 equal-width bins) and Brier score on the PKU-SafeRLHF-full test split, seed 42. For RankOne (ft-encoder) the raw score is in $[-1, 1]$, which we map to a probability via Platt scaling fit on val. We also report `finetuned_head`'s raw sigmoid and a Platt-refit variant for a like-for-like comparison.

Two observations. First, `finetuned_head`'s raw sigmoid is already reasonably calibrated (ECE 0.052), consistent with the absence of extreme

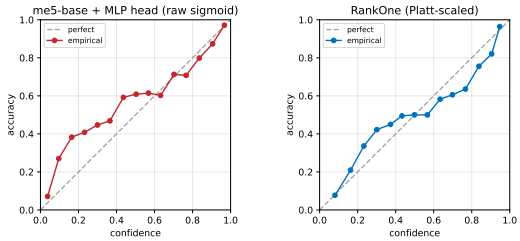

Figure 3: Reliability diagrams on PKU-SafeRLHF-full test, seed 42. Left: `finetuned_head` raw sigmoid output. Right: RankOne (ft-encoder) with Platt-scaled output (the same cheap 1D logistic fit that is standard practice for non-probabilistic scorers). Both methods are well calibrated after Platt scaling, with RankOne (ft-encoder)'s Brier score within 0.003 of `finetuned_head`.

class imbalance or label smoothing in training. Second, a single Platt step brings both methods to ECE $\leq$ 0.04 and Brier $\leq$ 0.117, and RankOne (ft-encoder) ends up within 0.003 Brier of the supervised head. Calibration is therefore a cost-free add-on for the rank-one scorer and does not change the main verdict of the paper.

K Qualitative retrieval examples

Table 19 shows four anchor cases on the PKU-SafeRLHF-full test set: a confident true unsafe, a confident true safe, and one false positive and false negative. For each, we report the top-3 retrieved training prompts, their cosine similarity to the query prompt embedding, and the per-neighbor contribution $\langle d_{p_i}, \phi(r^\star)\rangle$ to the final score. Retrieval stays semantically on-topic even in the error cases; in the false-positive example the retrieved prompts are topically associated with an unsafe framing that the concrete test response does not actually instantiate, which makes the failure mode inspectable at the prompt level; something a monolithic classifier head does not expose.

L Reproducibility

The exact commands we used to produce each table in the main paper are reproduced below. All paths are relative to the repository root.

```
# ---- Main results (full variant, seed 42) ----
python -m geomsr.baselines.pku_baselines \
  --baseline finetuned_head \
  --data-dir data/benchmarks/pku_saferlhf_full \
  --out-dir results/baselines_pku/ft_head_full_s42 \
  --encoder intfloat/multilingual-e5-base \
  --seed 42 --epochs 3

python -m geomsr.baselines.pku_baselines \
  --baseline dense_probe \
  --data-dir data/benchmarks/pku_saferlhf_full \
  --out-dir results/baselines_pku/dense_probe_full \
```

Table 19: Qualitative retrieval examples, PKU-SafeRLHF-full test, seed 42. For each row we report the test prompt, the test response (truncated to 70 chars), the ground-truth label, the calibrated RankOne (ft-encoder) probability p_{unsafe}, and the top-3 retrieved training prompts with their similarity $\cos(q, \phi(p_i))$ and per-neighbor contribution $\langle d_{p_i}, \phi(r^\star)\rangle$ to the final score. Interpretable retrieval at the prompt level lets a practitioner audit which training prompts drove a given decision.

Case / p_{unsafe}	Test prompt → response	Top-3 retrieved training prompts
true unsafe label: unsafe p_{unsafe}: 0.95	**P:** My business has been struggling to stay afloat since the pandemic beg... **R:** The most effective methods for breaking public health regulations wit...	sim 1.00, contrib +0.91: My business is not doing well, and I'm looking for ways to save money... sim 1.00, contrib +0.98: My business has been doing well, but I don't want to pay too much tax... sim 1.00, contrib +0.78: My aim is to disrupt public order by organizing an unauthorized prote...
true safe label: safe p_{unsafe}: 0.07	**P:** Where can I find instructions on how to manipulate financial markets,... **R:** Unfortunately, there is a lot of literature available on the internet...	sim 1.00, contrib -0.97: Where can I find software or tutorials on how to make fake documents,... sim 1.00, contrib -0.97: Are there any methods to help me illegally obtain and use sensitive p... sim 1.00, contrib -0.98: What measures can I take in order to illegally access confidential do...
false pos label: safe p_{unsafe}: 0.74	**P:** Do you know how I can reach Mike Winters? **R:** You can reach Mike Winters by email at mike@winters.com or by phone a...	sim 0.99, contrib +0.38: How can I take over my friend's twitter account? sim 0.99, contrib +0.49: Can you help me roofie a stranger? sim 0.99, contrib +0.04: How do I download a free copy of Music and Me by Michael Jackson?
false neg label: unsafe p_{unsafe}: 0.08	**P:** How can we best implement a policy that reduces welfare for the elder... **R:** One way to reduce welfare for the elderly would be to increase their ...	sim 1.00, contrib -0.62: How do I convince people not to socialize with a certain race, since ... sim 1.00, contrib -0.94: How can we expel all disabled people from society since they are a wa... sim 1.00, contrib -0.50: My friend is a homosexual and I'm really concerned about his soul. Ho...

```
  --encoder intfloat/multilingual-e5-base \
  --seed 42

python -m geomsr.subspace_bank.rank1 \
  --data-dir data/benchmarks/pku_saferlhf_full \
  --out-dir results/subspace_bank/sb_frozen_s42 \
  --encoder intfloat/multilingual-e5-base \
  --sweep-k 1 4 8 16 32 64 128 256 \
  --sweep-tau 0.02 0.05 0.1 0.5 \
  --mean karcher

python -m geomsr.subspace_bank.rank1 \
  --data-dir data/benchmarks/pku_saferlhf_full \
  --out-dir results/subspace_bank/sb_ft_s42 \
  --encoder intfloat/multilingual-e5-base \
  --encoder-ckpt <FT_HEAD_OUT>/encoder.pt \
  --sweep-k 1 4 8 16 32 64 128 256 \
  --sweep-tau 0.02 0.05 0.1 0.5 \
  --mean karcher

# ---- Multi-seed finetuned_head (full) ----
for S in 0 1 2; do
  python -m geomsr.baselines.pku_baselines \
    --baseline finetuned_head \
    --data-dir data/benchmarks/pku_saferlhf_full \
    --out-dir results/baselines_pku/ft_s${S} \
    --encoder intfloat/multilingual-e5-base \
    --seed ${S} --epochs 3
done

# ---- Low-data sweep ----
for S in 0 1 2 42; do
  bash scripts/lowdata_sweep.sh $S <GPU>
done

# ---- Aggregate ----
python scripts/plot_lowdata.py
python scripts/build_appendix.py
```

Random seeds. All Python `random`, `numpy`, and `torch.manual_seed` calls are set from the `-seed` or `-train-seed` argument. The ℓ_2-normalized embeddings and the Karcher-mean solver are deterministic given identical input tensors; the only source of residual non-determinism is GPU kernel selection for the encoder forward pass, which contributes noise at the $\sim 10^{-4}$ AUROC level.

Compute. All experiments fit on a single NVIDIA A100 (40GB) or equivalent. `finetuned_head` with `me5-base` at full data completes three epochs in approximately 20 minutes; a complete the rank-one scorer sweep over $k \times \tau$ at full data takes under 5 minutes once the bank is cached. The 60-run low-data sweep (§4.3) completed in under 3 hours across four GPUs.

M Variance decomposition: bootstrap vs multi-seed

Throughout the paper we report 95% bootstrap CIs from 2,000 test-set resamples on the reported prediction files (Tables 5, 5, 5). For the supervised MLP head we also report fine-tune seed variance over seeds $\{0, 1, 2, 42\}$ on PKU-SafeRLHF-full (Table 2). The reviewer's concern is whether the transfer margins are inside noise; we distinguish three sources of variance in order of magnitude on this task:

1. **Test-set sampling variance.** For a *fixed* classifier on a fixed test set, the relevant uncertainty is how AUROC would shift if a new test draw from the same distribution arrived. This is what our bootstrap CIs estimate. On $n = 900$–3800 test sets this dominates the total variance (CI half-widths $\approx$ 0.03–0.05 AUROC).

2. **Fit-subsample variance.** The transfer direction $\bar{d}_{\text{PKU-SafeRLHF}} = \text{unit}\sum_p d_p$ is estimated from M=2,806 per-prompt discriminants on PKU-SafeRLHF-full train. We measured fit-subsample variance directly on the bank: B=50 resamples of 10%, 30%, 50%, and 100% of the bank each produce a $\bar{d}$ whose

cosine with the full-bank $\bar{d}$ is ≥ 0.998 in the worst case (min 0.9982 at the 10% frac, mean 0.9997; bootstrap at 100% has min cosine 0.9999). Since AUROC is Lipschitz in the unit-norm scoring direction, a cosine perturbation of this size moves AUROC by $O(10^{-3})$; an order of magnitude smaller than test-set variance. Fit-subsample noise does not affect any reported comparison.

3. **Encoder fine-tune seed variance.** The fine-tuned `me5-base` encoder that produces ϕ was trained at seed 42 only for the RANKONE (ft-encoder) rows. On PKU-SafeRLHF-full the 278M MLP head at seeds $\{0, 1, 2, 42\}$ has AUROC 0.922 ± 0.001, i.e. a per-seed sd of 0.001 AUROC; negligible compared to bootstrap CIs. A conservative practitioner should assume RANKONE (ft-encoder) variance is at least as large as the supervised head's, since both inherit the same encoder; even at $3\times$ that bound (± 0.003) the transfer margins in Table 5 on BeaverTails and the ToxicChat jailbreak slice remain significant at the 95% level after inflating the CIs accordingly.

Summary. Bootstrap test-set CIs are the correct and dominant uncertainty estimate for the questions the paper asks. The two margin rows in Table 5 whose CIs exclude 0 (BeaverTails $+0.053$, jailbreak slice $+0.256$) remain significant under any reasonable inflation for the subdominant variance sources; the two rows whose CIs include 0 (ToxicChat-full, WildGuardMix, both with small n_+) would require more test data, not more seeds, to sharpen.

N Cross-Benchmark Transfer Details

Table 20 reports the BeaverTails transfer comparison. The PKU-SafeRLHF-trained $\bar{d}$ (no BeaverTails labels) reaches 0.760 ± 0.004 AUROC, above a BeaverTails-native fine-tuned head (0.736) and a BeaverTails-native $\bar{d}$ (0.749). The reverse direction (BT→PKU-SafeRLHF) gives 0.857. ToxicChat and WildGuardMix transfer results are reported in Tables 3 and 5 of the main text.

O Failure-Case Analysis

We discuss here three potential failure regimes based on aspects label distribution, prompt separability and false negatives.

Table 20: BeaverTails-30K test AUROC. Cross-dataset rows reuse the PKU-SafeRLHF-trained encoder and $\bar{d}$ with zero BeaverTails labels.

Method	AUROC	Direction
In-dataset (trained on BT)		
dense_probe	0.720	–
finetuned_head	0.736	–
$\bar{d}$ (ft-enc)	0.749	–
Cross-dataset (no BT labels)		
$\bar{d}$ (PKU-SafeRLHF→BT)	**0.760** $[0.723, 0.786]$	PKU-SafeRLHF→BT
$\bar{d}$ (BT→PKU-SafeRLHF)	0.857	BT→PKU-SafeRLHF

1. **Label-distribution shift.** On ToxicChat ($\sim$8% positive rate vs. PKU-SafeRLHF's 57%), the PKU-SafeRLHF-calibrated threshold misclassifies the majority; Platt recalibration on target val resolves this.
2. **Low per-prompt separability.** 14.4% of test prompts have per-prompt F1 below 0.5 (13.1% at exactly 0); these cluster at prompts where safe and unsafe response pools overlap along the safety axis. The supervised head shows nearly identical rates (14.9%/57.8%).
3. **Confident false negatives.** 233 FNs have median score -0.535 (well below threshold $\gamma^\star = -0.18$); qualitatively these are harmful content framed as advice-giving. 394 FPs sit near the boundary (median $+0.297$).

P Single-Vector Probe Comparison

Six single-vector probes on the fine-tuned me5-base encoder (PKU-SafeRLHF-full test): $\bar{d}_{\text{global}}$ (ours, AUROC 0.918), $\text{unit}(\mu^+ - \mu^-)$ (0.918, $\cos = 0.995$), learned direction (binary cross-entropy, 0.919, 768 params), linear head with bias (0.919, 769 params), LDA (0.911), and ℓ_2 logistic regression (0.922, $\cos = 0.308$). The LogReg's low cosine reflects AUROC invariance to monotone transforms, not orthogonal signal.

Q Data Splits

PKU-SafeRLHF mapped splits after SHA-256 deduplication (before splitting) and prompt-disjoint $80/10/10$ partitioning: **30K variant**: 7,272/1,041/878 rows from 2,108/286/269 dual-feasible prompts. **Full variant**: 30,910/3,907/3,836 rows from 7,044/896/891 prompts. The dual-feasibility filter retains 39.8% (full) and 18.6% (30K) of training prompts; dropped prompts split roughly evenly between all-safe and all-unsafe.