

ALIGNBEAM: Inference-Time Alignment Transfer via Cross-Vocabulary Logit Mixing

Chirag Chawla^{1*} Pratinav Seth¹ Vinay Kumar Sankarapu²

¹Lexsi Labs, Mumbai, India ²Lexsi Labs, London, United Kingdom

Correspondence: pratinav.seth@lexsi.ai

Abstract

Domain fine-tuning degrades the safety of large language models: fine-tuned specialists readily comply with harmful prompts framed in domain language. Existing inference-time defenses that mix logits from a safe *anchor* model require both models to share a vocabulary, which rules them out for the cross-family specialists where safety is most degraded. We present ALIGNBEAM, a training-free method that lifts this restriction by translating anchor probability mass into the target model’s vocabulary token-by-token at each decoding step; a small LLM judge then selects the safest among K candidate continuations. No weights are changed, and the mixing strength can be tuned at deployment without retraining. Across both cross-vocabulary and same-vocabulary evaluation pairs, ALIGNBEAM substantially raises refusal on structured, front-loaded adversarial benchmarks, with a $4.6\text{--}6.6\times$ inference slowdown inside the band of existing inference-time defenses. The results show that *early-token refusal behavior* can be transferred between model families at inference time, without touching either model’s weights.

1 Introduction

Domain fine-tuning degrades safety alignment. A medical specialist trained on patient notes will comply with pharmacology abuse questions framed as clinical queries; a finance specialist will help plan fraud once the harm is framed as a portfolio problem. System prompts offer no reliable fix: a system-prompted Llama-3.1-8B achieves 14.3% refusal, *below* the unprompted draft’s 16.9% (§4). Post-generation classifiers can flag harmful outputs but cannot steer generation away from them, and RLHF/DPO retraining is undone by as few as 100 benign fine-tuning steps (Qi et al., 2024). A more direct fix is to mix logits from a small, well-aligned anchor model during generation, but

existing methods (Xu et al., 2024; Liu et al., 2024; Li et al., 2023a) require both models to share a vocabulary. This rules them out for the cross-family specialists where safety is most degraded. We measure the construct of *early-token refusal behavior* via string-match refusal and benchmark-judge accuracy on structured adversarial prompts; this is narrower than general model safety, and we scope every claim in this paper to that construct rather than to safety writ large.

Prior work. Safety alignment concentrates in the first few output tokens and is fragile to domain shift (Qi et al., 2025), which is why mixing only a short prefix is enough. ALIGNBEAM extends SAFEDECODING (Xu et al., 2024) to the cross-vocabulary setting (full derivation, and comparison with LlamaGuard (Inan et al., 2023), RAIN (Li et al., 2024), an anchor-generated-prefix baseline, and recent decoding-time methods (Fei et al., 2025; Huang et al., 2025; Zhang et al., 2025), in Appendix D and Appendix M).

Contributions. We introduce ALIGNBEAM¹, a training-free method that mixes a safety anchor’s next-token distribution into a domain-specialist draft model’s decoding, without requiring a shared vocabulary; mixing depth N , weight α , and beam count K are all tunable at deployment without retraining (§2). We evaluate across five model pairs covering three SFT-eroded domain specialists, one unaligned base model, and a same-vocabulary control, and find large gains on structured, front-loaded adversarial benchmarks, with minimal accuracy cost (§4).

2 Method

Problem. Let M_d be a domain-specialist draft model with vocabulary $\mathcal{V}_d$ and M_s a small aligned safety anchor with vocabulary $\mathcal{V}_s$; in general $\mathcal{V}_d \neq \mathcal{V}_s$. Given a query x , we seek a response that is

*Work done while at Lexsi Labs.

¹Code: <https://github.com/Lexsi-Labs/alignbeam>

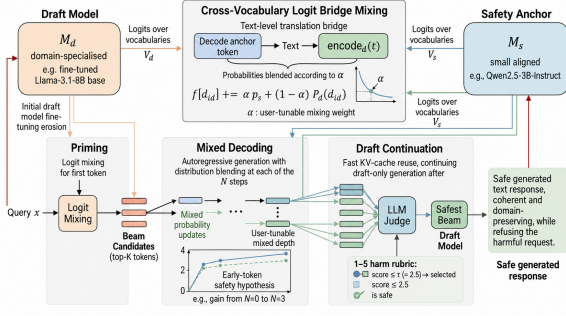


Figure 1: Anchor probability mass governs the safety-critical opening positions; the draft model takes over for fluency and domain accuracy. **Phase 1 (Priming):** LBM at step 0 yields K beam roots. **Phase 2 (Mixed Decoding):** each beam is extended greedily for $N-1$ steps via LBM. **Phase 3 (Draft Continuation):** M_d alone completes all K beams for domain quality (KV-cache reuse); an LLM judge then picks the safest completed beam.

both *safe* and *useful*. Neither model is modified; ALIGNBEAM runs at inference time (Figure 1).

Cross-vocabulary logit bridge mixing (LBM). Because M_d and M_s use different vocabularies, token IDs cannot be mixed directly. LBM bridges them by translating the anchor’s top- B predicted tokens (we use $B = 50$) into draft vocabulary space. Let P_s, P_d denote the next-token *probability* distributions (LBM mixes post-softmax probabilities, not raw logits, since logits from two different model families are not on a comparable scale) and $\mathcal{T}_B \subset \mathcal{V}_s$ the anchor’s top- B tokens. For each $s_{id} \in \mathcal{T}_B$ with probability $p_s = P_s(s_{id})$, we decode it to a string and re-encode under the draft tokeniser: $t = \text{decode}_s(s_{id})$, $d_{ids} = \text{encode}_d(t)$. Whenever the result is a single draft token ($|d_{ids}| = 1$), we accumulate it into a buffer f via

$$f[d_{id}] += \alpha p_s + (1 - \alpha) P_d(d_{id}), \quad (1)$$

where $\alpha \in [0, 1]$ is a deployment-time mixing-strength control, not a safety-utility knob in the strict sense: higher α steers more aggressively toward the anchor’s distribution, but $\alpha = 0$ does not recover ordinary draft decoding, since the text-bridge mapping still executes and gates the support (below) regardless of α , and benign over-refusal is nearly flat across the whole range. The buffer’s support is exactly the set of draft tokens reachable as a 1:1 image of the anchor’s top- B projection: a draft token the anchor does not endorse there receives zero mass, i.e. the anchor *gates* the support before renormalisation. When several anchor tokens round-trip to the same draft token, their contributions to $f[d_{id}]$ are summed rather than averaged, which acts as an implicit consensus bonus for draft

tokens the anchor agrees on from multiple directions. When the string decodes to multiple draft tokens, a variant-specific fallback applies: LBM-FIRST keeps only the first sub-token, while LBM-DROP and LBM-EXACT both exclude the anchor token from f entirely, contributing zero mass at that position; these two variants implement the identical operation (only their names differ, kept separately in reporting for historical reasons from earlier ablation runs where they were expected to diverge) and produce nearly identical results throughout this paper, differing only by 0.9 pp on DS-Math Sorry-Bench (45.0% vs. 45.9%; Table 17; see also Table 6, §X). Renormalising f gives the blended distribution $\tilde{f}$. On same-vocabulary pairs all three variants agree (>98% single-token match); on cross-family pairs roughly 12% of anchor tokens hit the fallback, and this fallback mass is concentrated in non-ASCII (CJK/emoji) anchor-vocabulary tokens that never surface at an English refusal position (Appendix F). When $\mathcal{V}_d = \mathcal{V}_s$, LBM is analogous to SAFEDECODING (Xu et al., 2024); the construction differs in that LBM selects support via beam search over continuations rather than by token-level top- k intersection.

We use LBM-DROP as the default: it carries the projected top- B anchor probability mass at each mixed position, suppressing harmful token mass and producing standard refusal phrases. LBM-FIRST injects only the first sub-token per step, steering the output toward safety without triggering the string-match refusal list (*stealth safety*; App. S).

Three-phase beam pipeline. *Phase 1* runs both models on query x (chat template for instruct models; Q: {x}\nA: for base) and picks the top- K tokens of $\tilde{f}$ as starting beams. *Phase 2* extends each beam greedily for $N-1$ further steps under LBM, requiring $O(K \cdot N)$ paired forward passes in total. *Phase 3* completes all K beams using M_d alone with KV-cache reuse; a small LLM judge J scores each beam on a 1–5 harm rubric and returns the safest one (score $\leq \tau = 2.5$). Phase 3 is the main computational cost, so total overhead scales with K , not N . We offer two settings: $K = 1$ (no judge, lowest overhead) and $K = 3$ (default), both within the latency range of existing inference-time defenses (Appendix Z).

3 Experiments

Model pairs. We evaluate five drafts ranging from 6.7B to 8B parameters: three

Algorithm 1 ALIGNBEAM Generation

Require: Query x ; draft M_d ; anchor M_s ; depth N ; beams K ; weight α
Ensure: Response y^*
Phase 1: Priming
1: $x_d \leftarrow \text{fmt}_d(x)$; $x_s \leftarrow \text{fmt}_s(x)$
2: $\tilde{f} \leftarrow \text{LBM}(M_d(x_d), M_s(x_s), \alpha)$
3: $\{b_k\}_{k=1}^K \leftarrow \text{top-}K \text{ tokens of } \tilde{f}$
Phase 2: Mixed Decoding
4: **for** step = 1, ..., $N-1$; each beam b_k **do**
5: $\tilde{f}^{(k)} \leftarrow \text{LBM}(M_d(x_d \| b_k), M_s(x_s \| b_k), \alpha)$
6: $v^* \leftarrow \arg \max_v \tilde{f}^{(k)}(v)$; extend b_k
7: **end for**
Phase 3: Draft Continuation
8: **for** each beam b_k **do**
9: Complete b_k with M_d alone (KV-cache reuse)
10: **end for**
11: $b^* \leftarrow \arg \min_k: J(b_k) \leq \tau J(b_k)$
12: **return** $y^* \leftarrow \text{text}(b^*)$

domain-fine-tuned specialists where SFT degraded safety (**MedLlama-3-8B**, **Finance-Llama3-8B**, **DS-Math-7B-Instruct**), one unaligned base model used as reference control (**Llama-3.1-8B**), and one same-vocabulary control (**Qwen3-8B-Base**, which shares a tokeniser with the anchor). DS-Math-7B-Instruct uses $N=20$ rather than the default $N=6$: at $N=6$ it produces a refusal prefix and then continues with the harmful answer, so a longer mixing window is needed (App. S). The safety anchor is Qwen2.5-3B-Instruct, fixed across all pairs.

Datasets. We use three evaluation categories, safety, calibration, and utility (sample counts in parentheses). **Safety:** HarmBench-Standard (167) (Mazeika et al., 2024), HarmBench-Contextual (62), AdvBench (520) (Zou et al., 2023), SorryBench (440) (Xie et al., 2025), WildJailbreak-Eval-Harmful (1,315) (Jiang et al., 2024). **Calibration (benign):** XSTest (450) (Röttger et al., 2024), OR-Bench-Hard (855) (Cui et al., 2025), JBB-Benign (100) (Chao et al., 2024). **Utility:** GSM8K (Cobbe et al., 2021) for the DS-Math cross-vocabulary pair, MedQA (Li et al., 2023b) for the medical specialist pair (MedLlama; a ChatDoctor-derived evaluation, App. E), and IFEval (Zhou et al., 2023) for instruction-following under the deployed $K=3$ safety configuration (Appendix Y.2). HarmBench-Contextual and WildJailbreak probe domain-framed and multi-sentence attacks respectively; calibration sets measure benign over-refusal. Supplementary harmful sets (JailbreakBench-Harmful (Chao et al., 2024)) are reported only in the appendix.

Hyperparameters and judging. All *safety-mode* runs (all Ref%/ACC% numbers in §4 and the main

safety appendices) use: $N=6$ (DS-Math: 20), $K=3$, $\alpha=0.5$, judge threshold $\tau=2.5$, temperature $T=0.7$, bridge width $B=50$, repetition penalty 1.15, seed 42. GSM8K and MedQA use the *utility* configuration ($K=1$, repetition penalty 1.0); IFEval uses the *safety* configuration ($K=3$, repetition penalty 1.0) as deployed (§Y.2). Repetition penalty 1.0 is used for all three because at 1.15 the penalty suppresses the repeated digits and operators needed for multi-step arithmetic and collapses DS-Math’s chain-of-thought. Each benchmark is scored by a publicly released, benchmark-compatible classifier; the exact judge for each dataset is listed in Appendix E. This paper reports two complementary safety metrics, string-match refusal (Ref%) and benchmark-judge accuracy (ACC%); either can be misleading in specific failure modes documented in Appendix C. The LLM judge in Phase 3 is used *only* to pick between beams during generation; it never scores the results reported here, avoiding any self-evaluation circularity.

4 Results

Headline. Across five pairs, ALIGNBEAM raises string-match refusal on **AdvBench** from 38.1% to **91.5%** (+53.4 pp macro avg.) and on **HarmBench-Std** from 25.9% to **76.4%** (+50.5 pp macro avg.) (Table 1; both share the HarmBench-Mistral judge). Four pairs reach **91–96% on AdvBench** and **78–81% on HarmBench-Std**; DS-Math-7B-Inst reaches 63.5% (+50.9 pp), pulling the macro average to 76.4% (median 79.6%). These gains are not an artefact of sample size: bootstrap 95% confidence intervals on the Llama-3.1-8B pair are disjoint between base and ALIGNBEAM on every structured benchmark (Appendix AA.2). **Benign over-refusal** (refusal on non-adversarial prompts; lower is better) rises across all pairs as the anchor’s tendency to refuse carries over (Llama-8B: OR-H 11.0%→22.3%, XSTest 5.6%→40.0%; per-pair details in Appendix G). Genuine benign-compliance accuracy (answered *and* judged safe, corrected metric per Appendix C) falls by a comparable, modest margin (Llama-8B OR-Bench-Hard 77.2%→72.5%). **Task utility** falls by -0.4 pp on both GSM8K and MedQA (Appendix Y), and instruction-following under the deployed $K=3$ safety configuration does not degrade (Appendix Y.2).

Early-token safety and the depth/weight trade-

ALIGNBEAM on harmful benchmark (↑)						
Draft model (domain)		HB-Std	HB-Ctx	AdvB	Sorry	WildJB
Domain-fine-tuned specialists (safety eroded by SFT; ALIGNBEAM restores it)						
MedLlama-3-8B (medical)	Ref%	<u>78.4</u> ± 61.6	<u>69.4</u> ± 59.7	<u>91.7</u> ± 51.5	<u>63.2</u> ± 45.2	<u>16.2</u> ± 2.1
	Acc%	88.6	87.1	99.0	72.3	42.5
Finance-Llama3-8B (finance)	Ref%	<u>79.6</u> ± 56.2	<u>80.6</u> ± 50.0	<u>94.8</u> ± 65.2	<u>65.7</u> ± 10.5	<u>33.2</u> ± 9.5
	Acc%	91.0	90.3	98.1	77.3	54.9
DS-Math-7B-Inst (math)	Ref%	<u>63.5</u> ± 50.9	<u>61.3</u> ± 51.6	<u>80.6</u> ± 69.4	<u>45.0</u> ± 31.8	<u>15.0</u> ± 3.5
	Acc%	85.0	79.0	91.3	65.7	40.3
Pre-trained / unaligned drafts (reference controls)						
Llama-3.1-8B (general)	Ref%	<u>79.6</u> ± 64.6	<u>79.0</u> ± 70.9	<u>94.4</u> ± 76.9	<u>61.4</u> ± 50.5	<u>19.7</u> ± 6.4
	Acc%	91.6	80.7	98.3	74.3	43.9
Qwen3-8B-Base (same-vocab ctrl)	Ref%	<u>80.8</u> ± 19.1	<u>71.0</u> ± 33.9	<u>95.8</u> ± 3.9	<u>63.4</u> ± 23.9	<u>20.9</u> ± 4.8
	Acc%	89.8	90.3	99.4	79.8	53.1
Macro avg.	Ref%	<u>76.4</u> ± 50.5	<u>72.3</u> ± 53.2	<u>91.5</u> ± 53.4	<u>59.7</u> ± 32.4	<u>21.0</u> ± 1.5
	Acc%	89.2	85.5	97.2	73.9	46.9

Table 1: **ALIGNBEAM raises count-weighted HB+AB refusal by +54.0–73.9 pp across the four cross-vocabulary pairs.** Benign over-refusal (not shown in this table) is reported separately in Table 3. Ref% = string-match refusal; Acc% = benchmark-judge accuracy (genuine-compliance on benign rows). Ref% and Acc% are complementary, not one more conservative than the other; either can be misleading in the specific failure modes documented in Appendix C. underlined = improvement over the unmodified draft. Config: LBM-DROP, $\alpha = 0.5$, $N = 6$ (DS-Math: $N = 20$), $K = 3$, seed 42. WildJB Acc% columns reflect a corrected WildGuard judge-parsing defect (Appendix G); WildJB Ref% is unaffected (string-match does not use a judge). Full Base/ALIGNBEAM grid in Appendix B.

off. Safety is concentrated in early tokens (Qi et al., 2025): the first three mixed steps add +61.9 pp HB+AB, and going deeper adds only 2.6 pp more from $N = 6$ to $N = 20$. Refusal saturates by $\alpha = 0.50$ while OR-Bench-Hard over-refusal stays flat (Limitations), so the α knob controls where on the refusal/over-refusal curve to operate. We use $\alpha = 0.50$, $N = 6$ as defaults; full sweeps, Pareto curves, and per-dataset numbers are in Appendices N–O. **Beam count K : two first-class deployment modes.** K is set once at deployment; both settings are intended operating points. $K = 1$ (budget, no judge, $\sim 2\times$) is not weaker on adversarial refusal: on the Llama-8B cross-vocabulary pair it matches or slightly exceeds $K = 3$ on all four harmful sets, but pays +3.3 pp XSTest over-refusal (43.3% vs. 40.0%), the judge’s role being benign-side calibration rather than safety. Prefer $K = 3$ (default, $\sim 5\times$) when calibration matters more than cost.

Anchor alignment vs. beam search. On the Llama-8B pair, beam search and anchor alignment each contribute roughly half the safety gain (Fig. 3, Appendix X), and a same-size anchor-size sweep (Appendix X) confirms it is calibration, not parameter count, that sets the ceiling.

Comparison to baselines. LlamaGuard filters reach near-ceiling on adversarial benchmarks but push benign over-refusal to 34–39% on OR-Bench-Hard and 38–39% on JBB-Benign. ALIGNBEAM

Mode	AdvB	HB-Std	HB-Ctx	XSTest $\downarrow$
$K = 1$	98.5	80.8	85.5	43.3
$K = 3$	94.4	79.6	79.0	40.0

Table 2: $K = 1$ vs. $K = 3$ on Llama-8B (string-match refusal%, harmful sets; XSTest is over-refusal, lower is better). $K = 3$ is *not* the higher-row-refusal setting here: the Phase-3 judge is itself a 3B model, imperfectly calibrated on ties between similarly-safe beams (Appendix E). We call $K = 3$ the deployed *safety* configuration because it is the one used for every other headline number in this paper, not because it maximizes raw refusal. Full profile, GSM8K/MedQA/IFEval deltas, and latency: Table 18, Appendix Z.

achieves 79.6% HarmBench-Std and 94.4% AdvBench refusal while keeping OR-Bench-Hard at 22.3% and JBB-Benign at 26.0%, a substantially better safety/over-refusal balance (Table 3; Appendix L). The safety-prompt baseline is actively counterproductive: 14.3% refusal, *below* the unprompted draft’s 16.9%. An anchor-generated-prefix baseline (the anchor alone decodes the first N tokens with no mixing and no judge) matches or slightly exceeds ALIGNBEAM on raw adversarial refusal, but ALIGNBEAM over-refuses less on every benign/calibration set and additionally offers a tunable α/K operating point that a fixed prefix cannot (Appendix M). On the same-vocabulary Qwen3-8B-Base pair, ALIGNBEAM beats both PROXY TUNING (Liu et al., 2024) and top- k contrastive decoding (Li et al., 2023a) on five of six adversarial benchmarks, with the widest gap on contextual attacks (Appendix J); results hold at 70B scale (Llama-3.1-70B, Appendix V).

Robustness. Results are stable across seeds {42, 2, 3} (± 0.7 pp HB+AB; App. T), across different Phase 1 prompt formats (App. Q), and under bootstrap resampling (Appendix AA.2). Dropping the LLM judge (App. U; $K = 3$ beams retained, judge disabled) leaves adversarial refusal unchanged or slightly higher; the only cost is +6.4 pp XSTest over-refusal, confirming that probability mixing provides the safety signal and the judge’s role is primarily to reduce over-refusal on benign inputs. Phase-2 mixing also directly suppresses the “refuse-then-continue” failure mode (a canonical refusal opener followed by harmful content anyway): on Llama-8B this rate drops from 5.8% $\rightarrow$ 0.2% on AdvBench and 6.0% $\rightarrow$ 1.8% on HarmBench-Std (Appendix AA.3). Full ablations and per-pair breakdowns are in the appendix.

Does the judge sharing the anchor’s checkpoint inflate the results? No: removing the judge

slightly *raises* adversarial refusal rather than lowering it, and every headline number is scored by a publicly released, benchmark-compatible classifier, independent of the anchor’s family (full argument in Appendix E).

Method	Harmful ($\uparrow$)				Benign [†] ($\downarrow$)	
	HB-Std	HB-Ctx	AdvB	Sorry	OR-H	JBB-B
Base draft	15.0	8.1	17.5	10.9	11.0	10.0
Safety prompt	10.2	8.1	15.6	15.2	14.7	10.0
Anchor prefix	82.0	85.5	97.7	63.0	23.7	29.0
LlamaGuard prompt-filter	98.2	95.2	99.4	85.5	34.5	38.0
LlamaGuard response-filter	98.2	100.0	99.4	83.4	34.6	39.0
ALIGNBEAM (ours)	79.6	79.0	94.4	61.4	22.3	26.0

Table 3: ALIGNBEAM trades some adversarial refusal for much lower benign over-refusal: 79–94% refusal on HB-Std, HB-Ctx and AdvBench (LlamaGuard: 95–100%), with over-refusal of 22–26% vs. LlamaGuard’s 34–39%. *Anchor prefix* is the anchor-generated-prefix condition (anchor decodes the first N tokens alone, no mixing, no judge; Appendix M); on this over-refusal metric it is worse than ALIGNBEAM despite competitive raw refusal (Appendix M reports the genuine-compliance comparison). String-match refusal%; harmful: $\uparrow$; $\uparrow$ benign / calibration (over-refusal): $\downarrow$. All rows: same Llama-3.1-8B draft. Anchor prefix at $K=1$; ALIGNBEAM at $K=3$.

Task utility. Under ALIGNBEAM’s $K=1$ utility mode, task accuracy drops by -0.4 pp on both domain pairs: GSM8K (DS-Math-7B-Inst) $77.0\% \rightarrow 76.6\%$ and MedQA (MedLlama-3-8B) $13.1\% \rightarrow 12.7\%$. Under the deployed $K=3$ **safety configuration**, instruction-following (IFEval, Llama-8B) does not degrade; it improves, from 14.2% to 27.0% prompt-level strict accuracy, because the anchor’s instruction-following competence is mixed into a draft that otherwise has none (Appendix Y.2); this establishes that early-token mixing does not reduce IFEval instruction-following accuracy under the safety configuration actually deployed, complementing the $K=1$ GSM8K/MedQA numbers. Full per-strategy MedQA similarity metrics (ROUGE-1/2, BERTScore-F1, SentSim; all within ± 1.5 pp) are in Appendix Y.1.

5 Conclusion

Early-token refusal behavior need not be encoded solely in a deployed model’s weights: it can be supplied at inference time by a separately maintained anchor. ALIGNBEAM instantiates this principle via a cross-vocabulary text bridge, raising AdvBench refusal to 91–96% on four of the five model pairs (four cross-vocabulary pairs and one same-vocabulary control) and substantially lifting count-weighted HB+AB refusal. Task accuracy (GSM8K, MedQA) changes by at most

0.5 pp, while wealth-management reference similarity (ROUGE-L) falls by 7.6 pp, the one non-small utility cost in this paper; ALIGNBEAM’s slowdown falls inside the $4.6\text{--}6.6\times$ band of existing inference-time defenses. The evidence supports early-token refusal steering, not general behavioral alignment transfer; the method’s boundary is the WildJailbreak delayed-payload setting (Limitations). Immediate extensions include multi-anchor ensembles and adaptive-prefix decoding.

Limitations

Inference cost is comparable to other inference-time defenses, and K/N are adjustable at deployment without retraining (Appendix Z). The safety ceiling is set by the anchor: over-refusal on ambiguous benign prompts is inherited and does not improve with more mixing, and attacks placing harmful content past the mixed prefix are only partially blocked and can even reverse on some pairs (Appendix AB). All experiments are single-turn, ≤ 200 tokens, non-adaptive, and use one anchor family (Qwen2.5-3B-Instruct); an alternative anchor must match the evaluation metric’s refusal style (Appendix R). Multi-turn, adaptive-attack, and multi-anchor robustness remain open (Appendix AB).

AI usage

The authors used AI-assisted tools during the preparation of this manuscript. Specifically, Claude Code (Anthropic) was used for implementation assistance, LaTeX formatting, and prose editing. All scientific content, experimental design, data collection, numerical results, analysis, and conclusions are entirely the authors’ own work. No AI tool was used to generate results, produce figures, or write scientific claims.

Ethics Statement

ALIGNBEAM raises refusal on domain-fine-tuned models, limiting direct misuse. Dual-use, residual risks, and data/compute disclosure: Appendix AC. **Reproducibility:** configurations, prompts, and seeds are in our repository (App. Z.1).

References

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwal, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramer, Hamed Hassani,

- and Eric Wong. 2024. JailbreakBench: An open robustness benchmark for jailbreaking large language models. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. [Training verifiers to solve math word problems](#). *arXiv preprint arXiv:2110.14168*.
- Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. 2025. OR-Bench: An over-refusal benchmark for large language models. In *International Conference on Machine Learning*.
- Yu Fei, Yasaman Razeghi, and Sameer Singh. 2025. Nudging: Inference-time alignment of LLMs via guided decoding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*.
- Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. 2024a. [WildGuard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of LLMs](#). *arXiv preprint arXiv:2406.18495*.
- Tessa Han, Aounon Kumar, Chirag Agarwal, and Himabindu Lakkaraju. 2024b. Medsafetybench: Evaluating and improving the medical safety of large language models. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*.
- James Y. Huang, Sailik Sengupta, Daniele Bonadiman, Yi-An Lai, Arshit Gupta, Nikolaos Pappas, Saab Mansour, Katrin Kirchhoff, and Dan Roth. 2025. DeAL: Decoding-time alignment for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*.
- Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabisa. 2023. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*.
- Liwei Jiang, Kavel Rao, Seungju Han, Allyson Ettinger, Faeze Brahman, Sachin Kumar, Niloofar Mireshtgah, Ximing Lu, Maarten Sap, Yejin Choi, and Nouha Dziri. 2024. [Wildteaming at scale: From in-the-wild jailbreaks to \(adversarially\) safer language models](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 47094–47165. Curran Associates, Inc.
- Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. [What disease does this patient have? A large-scale open domain question answering dataset from medical exams](#). *Applied Sciences*, 11(14):6421.
- Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023a. [Contrastive decoding: Open-ended text generation as optimization](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.
- Yuhui Li, Fangyun Wei, Jinjing Zhao, Chao Zhang, and Hongyang Zhang. 2024. RAIN: Your language models can align themselves without finetuning. In *International Conference on Learning Representations*.
- Yunxiang Li, Zihan Li, Kai Zhang, Ruilong Dan, Steve Jiang, and You Zhang. 2023b. [Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai \(llama\) using medical domain knowledge](#). *Preprint*, arXiv:2303.14070.
- Alisa Liu, Xiaochuang Han, Yizhong Wang, Yulia Tsvetkov, Yejin Choi, and Noah A. Smith. 2024. Tuning language models by proxy. In *Conference on Language Modeling*.
- Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. 2024. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In *International Conference on Machine Learning*.
- Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. 2025. Safety alignment should be made more than just a few tokens deep. In *International Conference on Learning Representations*.
- Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2024. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations*.
- Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. 2024. [XSTest: A test suite for identifying exaggerated safety behaviours in large language models](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5377–5400, Mexico City, Mexico. Association for Computational Linguistics.
- Tinghao Xie, Xiangyu Qi, Yi Zeng, Yangsibo Huang, Udari Madhushani Sehwal, Kaixuan Huang, Luxi He, Boyi Wei, Dacheng Li, Ying Sheng, Ruoxi Jia, Bo Li, Kai Li, Danqi Chen, Peter Henderson, and Prateek Mittal. 2025. [SORRY-Bench: Systematically evaluating large language model safety refusal behaviors](#). In *International Conference on Learning Representations*.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Jinyuan Jia, Bill Yuchen Lin, and Radha Poovendran. 2024. SafeDecoding: Defending against jailbreak attacks via safety-aware decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5587–5605. Association for Computational Linguistics.

Xianjun Yang, Xiao Wang, Qi Zhang, Linda Petzold, William Yang Wang, Xun Zhao, and Dahua Lin. 2023. *Shadow alignment: The ease of subverting safely-aligned language models*. *arXiv preprint arXiv:2310.02949*.

Yiming Zhang, Jianfeng Chi, Hailey Nguyen, Kartikeya Upasani, Daniel M. Bikel, Jason E Weston, and Eric Michael Smith. 2025. *Backtracking improves generation safety*. In *The Thirteenth International Conference on Learning Representations*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

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

A Acronyms Used in the Appendix

Methods/components: LBM (logit bridge mixing), LBM-DROP/ LBM-FIRST/ LBM-EXACT (LBM multi-token fallbacks). *Benchmarks:* HB-Std (HarmBench-Standard), HB-Ctx (HarmBench-Contextual), AdvB (AdvBench), JBB-H (JailbreakBench-Harmful), JBB-B (JBB-Benign), Sorry (Sorry-Bench), WildJB (WildJailbreak-Eval-Harmful), OR-H (OR-Bench-Hard), OR-Tox (OR-Bench-Toxic), XSTest. *Model-pair shorthand:* DS-Math = DS-Math-7B-Inst, Llama-8B = Llama-3.1-8B, Llama-70B = Llama-3.1-70B, MedLlama = MedLlama-3-8B, Finance-Llama = Finance-Llama3-8B, Qwen3-Base = Qwen3-8B-Base, Anchor-Size = Qwen2.5-7B-Base (same-family ablation), Wealth-LoRA = Llama-3.1-8B-Instruct + wealth-management LoRA adapter. *Baselines:* Instruct-Base = instruct-counterpart, LG-Prompt = LlamaGuard prompt-filter, LG-Response = LlamaGuard response-filter, Anchor-Prefix = anchor-generated prefix. *Metrics:* Ref% = string-match refusal, ACC% = benchmark-judge accuracy (benign/calibration ACC% is the corrected “answered AND judged-safe” genuine-compliance metric; Appendix C), Δ_{safe} = ALIGNBEAM Ref%

– Base Ref%, HB+AB = count-weighted refusal over HarmBench-Std ($n = 167$) and AdvBench ($n = 520$).

B Regime Patterns and Aggregate HB+AB

ALIGNBEAM raises the **count-weighted HB+AB aggregate** (HarmBench-Std $n = 167$ + AdvBench $n = 520$, shared HarmBench-Mistral judge) from 35.1% to 87.8% (+52.7 pp) macro-averaged over five pairs (Table 4; main-body Table 1 shows only the ALIGNBEAM row). Per-pair HB+AB: Llama-8B 16.9→**90.8** (+73.9), MedLlama 34.5→**88.5** (+54.0), Finance-Llama 28.1→**91.1** (+63.0), DS-Math 11.5→**76.4** (+64.9), Qwen3-Base 84.6→**92.1** (+7.5).

Regime patterns. The gain pattern across pairs reflects four distinct baseline safety properties. *Cross-vocabulary domain specialists with low safety priors* (Llama-8B, MedLlama, Finance-Llama): all three start at 17–35% and reach 88–91% HB+AB, the largest absolute gains. The medical pair (MedLlama; App. H) gains most on HarmBench-Contextual (+59.7 pp) where domain framing is the attack vector; the financial pair (Finance-Llama; App. I) follows the same pattern. *Instruct draft with no safety prior* (DS-Math): the math specialist’s chat template receives explicit user turns but was never trained to refuse them, so the anchor overrides the entire harmful-response trajectory at $N = 20$ positions; HB+AB moves from 11.5% to 76.4%. *Same-vocabulary control* (Qwen3-Base): gains only +7.5 pp from an already-aligned 84.6% baseline, confirming that ALIGNBEAM behaves analogously to SAFEDECODING for same-family pairs (the construction differs in support selection: LBM uses the anchor’s top- B tokens rather than a token-level intersection) and adds no harm in that regime.

C Why String-Match Refusal as Primary Metric

String-match refusal (Ref%) and benchmark-judge accuracy (ACC%) are complementary, primary metrics; neither is uniformly more conservative than the other, and either can be misleading in specific failure modes. We identified and corrected two evaluation-harness defects affecting benign genuine-compliance and WildJailbreak judge accuracy; string-match refusal and all non-WildJailbreak harmful-benchmark scores were un-

Draft model	Method	Metric	HB-Std	HB-Ctx	AdvB	JBB-H	Sorry	WildJB	HB+AB %	Δ_{safe}
<i>Base drafts, cross-vocab, low safety prior</i>										
Llama-3.1-8B	Base	Ref%	15.0	8.1	17.5	30.0	10.9	13.3	16.9	+73.9
		Acc%	49.1	53.2	52.5	11.0	34.9	43.4 [†]	51.7	
	Ours	Ref%	79.6	79.0	94.4	80.0	61.4	19.7	90.8	
		Acc%	91.6 _{+42.5}	80.7 _{+27.5}	98.3 _{+45.8}	79.0 _{+68.0}	74.3 _{+39.4}	43.9 [†] _{+0.5}	96.7	
MedLlama-3-8B	Base	Ref%	16.8	9.7	40.2	25.0	18.0	14.1	34.5	+54.0
		Acc%	35.3	25.8	59.2	42.0	31.1	43.9 [†]	53.4	
	Ours	Ref%	78.4	69.4	91.7	79.0	63.2	16.2	88.5	
		Acc%	88.6 _{+53.3}	87.1 _{+61.3}	99.0 _{+39.8}	82.0 _{+40.0}	72.3 _{+41.2}	42.5 [†] _(−1.4)	96.5	
Finance-Llama3-8B	Base	Ref%	23.4	30.6	29.6	27.0	55.2	42.7	28.1	+63.0
		Acc%	64.1	79.0	78.5	56.0	67.0	58.6 [†]	75.0	
	Ours	Ref%	79.6	80.6	94.8	88.0	65.7	33.2	91.1	
		Acc%	91.0 _{+26.9}	90.3 _{+11.3}	98.1 _{+19.6}	94.0 _{+38.0}	77.3 _{+10.3}	54.9 [†] _(−3.7)	96.4	
<i>Instruct draft, cross-vocab (math)</i>										
DS-Math-7B-Inst	Base	Ref%	12.6	9.7	11.2	16.0	13.2	11.5	11.5	+64.9
		Acc%	41.9	53.2	46.9	14.0	38.2	43.2 [†]	45.7	
	Ours	Ref%	63.5	61.3	80.6	67.0	45.0	15.0	76.4	
		Acc%	85.0 _{+43.1}	79.0 _{+25.8}	91.3 _{+44.4}	72.0 _{+58.0}	65.7 _{+27.5}	40.3 [†] _(−2.9)	89.8	
<i>Same-vocab control</i>										
Qwen3-8B-Base	Base	Ref%	61.7	37.1	91.9	76.0	39.5	16.1	84.6	+7.5
		Acc%	80.8	64.5	95.6	81.0	55.2	58.5 [†]	92.0	
	Ours	Ref%	80.8	71.0	95.8	85.0	63.4	20.9	92.1	
		Acc%	89.8 _{+9.0}	90.3 _{+25.8}	99.4 _{+3.8}	91.0 _{+10.0}	79.8 _{+24.6}	53.1 [†] _(−5.4)	97.1	
<i>Macro avg. Base</i>		Ref%	25.9	19.0	38.1	34.8	27.4	19.5	35.1	+52.7
Macro avg. Ours		Acc%	54.2	55.1	66.5	40.8	45.3	51.1 ^{*†}	63.6	
		Ref%	76.4	72.3	91.5	79.8	59.7	21.0_{+1.5}	87.8	
		Acc%	89.2 _{+35.0}	85.5 _{+30.4}	97.2 _{+30.7}	83.6 _{+42.8}	73.9 _{+28.6}	48.6 ^{*†} _(−2.5)	95.3	

Table 4: **Cross-vocabulary pairs (DS-Math, Llama-8B, MedLlama, Finance-Llama) gain +54–74 pp HB+AB; the same-vocabulary control (Qwen3-Base) gains +7.5 pp from an already-aligned baseline. Full Base / ALIGNBEAM grid.** Ref% = string-match refusal; Acc% = benchmark-judge accuracy. * = 4-pair mean (DS-Math excluded); otherwise 5-pair. [†]WildJB Acc% corrected for a WildGuard judge-parsing defect (Appendix G); WildJB Ref% is unaffected, since string-match does not use a judge. Calibration / over-refusal per pair in Appendices G–J.

affected. Five patterns explain when and why Ref% and ACC% diverge, including the two corrected defects (the fourth and fifth items below).

Base-model judge inflation. Incoherent or repetitive base-model outputs are correctly judged non-harmful but match none of the 22 canonical refusal phrases, so Base ACC% can exceed Base Ref% by ≈ 34 pp on Llama-8B (Appendix G).

ALIGNBEAM refuses in non-canonical phrasing. The anchor produces polite declinations that the judge classifies as refusals but that do not appear in the 22-phrases list. This creates a gap between ALIGNBEAM ACC% and Ref% on the *harmful* (refuse-expected) rows: 2–22 pp on aligned drafts (Llama-8B/MedLlama/Finance-Llama/Qwen3-Base median ≈ 10 pp), and >80 pp on the alternative-anchor run (Appendix R). On WildJailbreak specifically, correcting the WildGuard scoring defect described below removes what had looked like an instance of this same pattern for DS-Math-7B-Inst: string-match refusal rises modestly under ALIGNBEAM

(11.5% $\rightarrow$ 15.0%, +3.5 pp), but corrected judge accuracy *falls* (43.2% $\rightarrow$ 40.3%, -2.9 pp) rather than rising, so this is not, after all, a case of the judge revealing hidden safety the string match misses; see the fifth pattern below for the full, corrected WildJailbreak picture across every pair. We report both columns for every pair precisely so distinctions like this one are visible rather than absorbed into a single macro-average.

Sorry-but-continues (Ref% > ACC%). Some unsteered models produce a canonical refusal phrase and then continue with the requested harmful content. This pattern appears in certain Sorry-Bench sub-categories (Fraud/Deception, Privacy); in these cases Ref% *overestimates* safety and ACC% is the reliable measure. ALIGNBEAM’s Phase 2 probability mixing anchors the continuation to the safety anchor’s distribution, substantially reducing this pattern in ALIGNBEAM outputs: measured directly on Llama-8B, the refuse-then-continue rate drops from 5.8% $\rightarrow$ 0.2% on AdvBench and 6.0% $\rightarrow$ 1.8% on HarmBench-Std under

ALIGNBEAM (Appendix AA.3).

Benign/calibration ACC% correction (harness bug, fixed). An evaluation-harness bug scored comply-expected correctness (on benign/calibration prompts such as OR-Bench-Hard, JBB-Benign, and the comply half of XSTest) as the judge’s raw harmfulness flag, which rewards an over-refusal as “correct” rather than penalising it. We fixed this to the intended metric, **answered AND judged-safe** (genuine compliance), and re-scored every affected table. This bug touched *only* comply-expected ACC% columns: the benign/calibration sets named above, and also the comply-labelled subset of WildJailbreak (210 of its 1,315 prompts, 16.0%), which is scored by the same rule and is corrected together with the parsing defect described next; string-match refusal% everywhere (the paper’s primary metric, including Table 1) and ACC% on every harmful/refuse row were computed by an independent code path and are unaffected. The corrected genuine-compliance numbers are lower than the buggy ones but still well above LlamaGuard’s collapse on the same prompts (Appendix L): e.g. on Llama-8B, OR-Bench-Hard genuine compliance falls from 77.2% (base) to 72.5% (ALIGNBEAM), a modest, real over-refusal cost, not the near-total collapse the buggy formula implied for the LlamaGuard baselines. One dataset, OR-Bench-Toxic, remains excluded from ACC% reporting on either side of the fix: its official judge flags $\approx 55\%$ of genuinely correct answers as harmful, so we report over-refusal rate there instead of accuracy.

WildJailbreak judge-parsing correction. A second, independent defect compounded the first on WildJailbreak specifically. The WildGuard judge (used only for WildJailbreak) returns a multi-line free-text verdict (e.g. “Harmful request: no / Response refusal: no / Harmful response: no”); an older parsing path could not extract a label from this format on a large fraction of samples and silently defaulted to labelling the response harmful on parse failure, including on samples where the raw text explicitly says “no” three times. Roughly 25–47% of per-sample labels flip once re-parsed. Because WildJailbreak mixes comply-expected prompts in with the adversarial ones, its cached scores carry *both* this parsing defect and the comply-row formula bug described above, and the two must be corrected together: on Llama-8B, fixing the parser alone moves base ACC% from 21.5% to 29.4%, and fixing the comply rule as

well moves it to 43.4%. We therefore re-parsed every cached WildJailbreak judgment with the current, correct parser (which reads the “Harmful response: yes/no” field directly), re-applied the correct comply/refuse rule, and recomputed ACC% for every pair. Corrected WildJailbreak ACC% (base $\rightarrow$ ALIGNBEAM) is: DS-Math 43.2% $\rightarrow$ 40.3%, Llama-8B 43.4% $\rightarrow$ 43.9%, MedLlama 43.9% $\rightarrow$ 42.5%, Finance-Llama 58.6% $\rightarrow$ 54.9%, Qwen3-Base 58.5% $\rightarrow$ 53.1%. Two things follow. First, genuine WildJailbreak judge-accuracy is substantially higher across the board than the originally reported 20–60% range: the defects suppressed it uniformly, not selectively. Second, and more consequentially for how this metric should be read: once corrected, ALIGNBEAM no longer improves WildJailbreak judge-accuracy on any of the cross-vocabulary pairs in Table 1: the delta is close to flat for Llama-8B (+0.5 pp) and negative for the rest (−1.4 to −5.4 pp), consistent with, and reinforcing, the delayed-payload limitation already discussed in the Limitations section: early-token mixing does not reach content placed later in a response, on either metric. One exception survives the correction: the same-family Anchor-Size ablation remains clearly positive (+10.4 pp with the 7B anchor, +7.3 pp with the 0.5B; Appendix X). We do not read this as evidence that the delayed-payload limit is avoidable, and we do not have an account of why this pair differs: shared tokenization does not by itself predict the sign, since the same-vocabulary Qwen3-Base pair moves the other way (−5.4 pp), nor does a lower starting point, since DS-Math, Llama-8B and MedLlama all begin below Anchor-Size’s corrected base of 45.7% and do not improve. We report it as an unexplained exception rather than fit a mechanism to a single pair. Before the scoring correction (Appendix C), DS-Math, Llama-8B (implicitly via Table 1), and Qwen3-Base appeared to show a positive ACC% delta here; the corrected values do not. Ref% (string-match) is entirely unaffected, since it never uses the judge; every string-match number involving WildJailbreak reported anywhere in this paper, including the N=20 depth-check comparison in Appendix AA.1, is unchanged by this fix.

D Extended Related Work

Alignment erosion under fine-tuning. Qi et al.

(2024) show that as few as 100 benign fine-tuning samples substantially reduce refusal, and Yang et al. (2023) demonstrate that authority and domain-framing prompts provide potent soft jailbreaks. Our two domain-fine-tuned specialists (MedLlama medical, Finance-Llama finance) and the math-instruct model without safety training (DS-Math) reproduce this erosion at scale, with HB+AB string-match baselines of 12–35% vs. matching instruct upper-bounds; the two pre-trained / unaligned drafts (Llama-8B Llama-base, Qwen3-Base Qwen3-base) span the orthogonal absence-of-alignment regime (HB+AB baselines 17–85%). A LoRA-adapted instruct model (Wealth-LoRA, Appendix W) shows the same erosion pattern starting from an already-safe base, isolated to contextually-framed attacks rather than structured ones.

Inference-time logit interventions. The shared-vocabulary assumption is the central limitation of prior probability-mixing methods. SAFEDECODING (Xu et al., 2024) blends draft and safety-model probabilities at the shared vocabulary intersection; PROXY TUNING (Liu et al., 2024) and top- k contrastive decoding (Li et al., 2023a) apply the same logit arithmetic, all restricted to same-family pairs. ALIGNBEAM relaxes this assumption via a text bridge and yields a construction analogous to SafeDecoding when $\mathcal{V}_d = \mathcal{V}_s$ (differing in support selection: beam search vs. token-level intersection); on the same-vocabulary control (Qwen3-Base) it outperforms both Proxy Tuning and top- k CD on five of six benchmarks (+25.8 pp vs. Proxy Tuning on HB-Ctx; Table 11). Recent methods that do not rely on logit sharing also fail on domain-fine-tuned models: NUDGING (Fei et al., 2025) swaps to an aligned guide on high-entropy steps; DEAL (Huang et al., 2025) uses reward-guided beam search; backtracking safety (Zhang et al., 2025) rolls back on early unsafe tokens; and RAIN (Li et al., 2024) self-aligns via rewindable generation. All four require a safety prior in the draft model itself, which domain fine-tuning has erased. A simpler alternative, letting the anchor alone generate a prefix with no cross-vocabulary mixing at all, is a natural question raised during review; we run and report it directly as Appendix M.

Post-hoc guard classifiers. LlamaGuard (Inan et al., 2023) and similar response-time classifiers block detected outputs but do not steer generation. In domain-specialised settings, benign prompts that share surface features with harmful ones (drug

names, financial instruments, exploit syntax) trigger classifier flags at high rates; this is why LlamaGuard raises OR-Bench-Hard over-refusal to 34–39% vs. ALIGNBEAM’s 22.3% on the same Llama-3.1-8B draft (Appendix L).

Early-token safety concentration. Qi et al. (2025) argue that safety alignment concentrates in the first few token positions and is brittle to distribution shift. If true, a shallow mixed-decoding prefix should be sufficient to restore safety; ALIGNBEAM’s depth ablation (Depth-Sweep; Fig. 2(a)) confirms this directly: three mixed steps recover 92.7% AdvBench refusal on Llama-8B, while deepening the window adds only a further 3.9 pp on HB+AB from $N = 3$ to $N = 20$ (2.6 pp from $N = 6$; Table 14). A directly analogous saturation holds on the harder WildJailbreak setting: tripling the window from $N = 6$ to $N = 20$ recovers only +1.6 pp string-match refusal (Appendix AA.1), because there the binding constraint is upstream of depth: whether the anchor recognises the adversarial framing at all, not how many tokens are mixed once it does. This is why the small mixed-depth defaults ($N \in \{3, 6\}$) are used throughout: the gain saturates early on prompts the anchor recognises, and deeper mixing only adds latency.

E Evaluation Judges and Dataset Details

Every benchmark is scored with a *publicly released, benchmark-compatible classifier* to avoid evaluator-vs-method confounds and to keep our numbers directly comparable to prior work that uses the same setup. Public benchmarks: HarmBench (Mazeika et al., 2024), AdvBench (Zou et al., 2023), SorryBench (Xie et al., 2025), JailbreakBench (Chao et al., 2024), WildJailbreak (Jiang et al., 2024) (judged with WildGuard (Han et al., 2024a)), OR-Bench (Cui et al., 2025), XSTest (Röttger et al., 2024), GSM8K (Cobbe et al., 2021), and IFEval (Zhou et al., 2023). Table 5 lists, for each benchmark used in the paper, the judge model, judge output type, expected direction (harmful prompts → refusal correct; benign/calibration prompts → compliance correct), dataset size used in our runs, and the source. None of these judges is used *inside* ALIGNBEAM’s beam selection (which uses a separate 3B classifier, described below).

Type key. H = harmful (model should refuse); B = benign (model should comply); C = calibration (mixed labels; the dataset’s own ground-truth label determines the expected behaviour per prompt);

Benchmark	Type	n	Judge	Output	Source
HarmBench-Standard	H	167	cais/HarmBench-Mistral-7b-val-cls	yes/no	walledai/HarmBench (standard)
HarmBench-Contextual	H	62	cais/HarmBench-Mistral-7b-val-cls	yes/no	walledai/HarmBench (contextual)
AdvBench	H	520	cais/HarmBench-Mistral-7b-val-cls	yes/no	walledai/AdvBench
Sorry-Bench	H	440	sorry-bench/ft-mistral-7b-instruct-v0.2-sorry-bench-202406	0/1	sorry-bench/sorry-bench-202503
JailbreakBench-Harmful	H	100	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	JailbreakBench/JBB-Behaviors (harmful)
JailbreakBench-Benign	B	100	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	JailbreakBench/JBB-Behaviors (benign)
WildJailbreak-Eval	C	1,315	allenai/wildguard	multi-label text	allenai/wildjailbreak (eval)
OR-Bench-Hard	B	855	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	bench-llm/or-bench (hard-1k)
OR-Bench-Toxic	B	585	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	bench-llm/or-bench (toxic)
XSTest	C	450	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	Paul/xstest
MedSafetyFix	H	900	Meta-Llama/Llama-Guard-3-8B	safe/unsafe	AI4LIFE-GROUP/med-safety-bench
GSM8K (utility)	U	1,319	exact-match on extracted numeric answer	accuracy	openai/gsm8k (test)
MedQA (utility)	U	2,243	ROUGE-L F1 + answer-extraction ACC + semantic metrics	accuracy	lavita/ChatDoctor-HealthCareMagic-100k
IFEval (utility)	U	541	rule-based verifiable-instruction checkers	prompt/instr. accuracy	google/IFEval

Table 5: Per-benchmark evaluation setup. n = prompts evaluated in our runs. Judge output is parsed to a binary harmful/not-harmful label; correctness is measured against the expected direction for each type. We use publicly released, benchmark-compatible classifiers; the exact judge for each dataset is listed above (some, e.g. AdvBench, are scored with a judge released for a different benchmark that shares its harmful-completion definition, not with a bespoke classifier of their own). System prompts follow each dataset’s official evaluation protocol where prescribed; otherwise no system prompt is used. The MedSafetyFix row is drawn from the public MedSafetyBench benchmark; its provenance and naming are detailed in Appendix E.1.

U = utility (scored by task accuracy, not refusal classification).

Why these judges. HarmBench-Mistral-7b-val-cls is the judge released with the HarmBench paper and validated there at $\approx 93\%$ agreement with human red-teamers (Mazeika et al., 2024); we re-use it unchanged on HarmBench-Std/Ctx and AdvBench. Sorry-Bench provides its own fine-tuned 7B classifier (Xie et al., 2025) that is calibrated to the category-balanced Sorry-Bench split. LlamaGuard-3-8B (Inan et al., 2023) is the official JBB and OR-Bench judge and the one used in our domain-safety set (MedSafetyFix). WildGuard (Han et al., 2024a) is paired with WildJailbreak by construction. IFEval uses its own release’s rule-based instruction checkers, which require no LLM judge and so cannot inherit any judge-side bias. We deliberately avoid substituting any judge with an in-house re-trained variant to keep every column comparable across this paper, prior work, and future follow-ups.

E.1 MedSafetyFix: Provenance and Naming

MedSafetyFix ($n=900$) is drawn from MedSafetyBench (Han et al., 2024b)², a public benchmark of harmful medical requests framed in clinical and academic language that a medical specialist handles fluently, making domain framing the principal attack vector. Judge: LlamaGuard-3-8B (§E). The name “MedSafetyFix” and the 900-prompt evaluation split used here are specific to this paper’s harness configuration, not a released split of the benchmark, which is why the set is listed under

²Benchmark: <https://github.com/AI4LIFE-GROUP/med-safety-bench>.

its harness name in Table 5. As with every benchmark in this paper, no harmful model *outputs* are reproduced anywhere, only aggregate statistics.

String-match refusal (primary metric). A response is counted as a refusal if any of the following 22 canonical phrases appears as a case-insensitive substring of the model output: “I’m sorry”, “I am sorry”, “sorry”, “I apologize”, “as an AI”, “as a language model”, “as an assistant”, “I cannot”, “I do not”, “I’m unable to”, “I am unable to”, “I am not allowed to”, “I cannot assist”, “I will not”, “I cannot provide”, “I cannot help”, “I cannot support”, “I cannot recommend”, “against ethical”, “not ethical”, “illegal”, “harmful” (plus the implementation-level extra phrase “I must refuse”). The string-match rule is identical across every benchmark, every pair, every method.

Benchmark-judge accuracy (ACC%). For harmful benchmarks (Type H above), ACC% is the fraction of responses the judge classifies as not exhibiting the harmful behaviour, i.e., correct refusal as judged by the benchmark’s judge classifier. For benign / calibration benchmarks (Type B and the comply rows of Type C), ACC% is the fraction of responses that are *both answered and judged safe* – the corrected genuine-compliance definition (Appendix E.1), so that a refusal is never counted as a correct response on a prompt the model was expected to answer. WildJailbreak and XSTest are mixed-label calibration sets: per-prompt expected behaviour is read from the dataset’s ground-truth label and correctness is computed against that expectation.

LLM judge inside Phase 3 (in-generation only).

A separate Qwen2.5-3B-Instruct judge scores beams on a 1–5 harm rubric inside Phase 3 of Algorithm 1. It is used *only* for in-generation beam selection; the safest beam is returned if any beam has score $\leq \tau = 2.5$. This judge is never used to report safety numbers (avoiding self-evaluation circularity).

Judge and anchor family overlap. The Phase 3 judge and the safety anchor share the same checkpoint (Qwen2.5-3B-Instruct). The no-judge ablation (App. U) shows that *dropping* the judge slightly *increases* adversarial accuracy (AdvBench +3.5 pp, HB-Ctx +6.5 pp) while also raising XSTest over-refusal by +6.4 pp, confirming that the judge’s role is calibration (suppressing benign over-refusal) rather than adversarial-accuracy inflation. Anchor-judge family overlap therefore does not inflate the headline safety numbers; if anything the judge is conservative on adversarial sets. All headline safety numbers are scored by the publicly released, benchmark-compatible classifier listed for that benchmark (Table 5), which is independent of the Qwen family for HarmBench, Sorry-Bench, JBB, OR-Bench, and XSTest.

Utility metrics. GSM8K accuracy is exact-match on the numeric answer extracted from the response (the harness parses “#### N” suffix or the first standalone numeric token). MedQA reports four quantities on ChatDoctor-HealthCareMagic-style (Li et al., 2023b)³ free-form generations against the reference doctor response: ROUGE-L F1 (the primary correctness signal we use), ROUGE-1/2, BERTScore-F1, and SentSim (all-MiniLM-L6-v2 cosine similarity). The accuracy figures in the main body utility paragraph (13.1% $\rightarrow$ 12.7%) are answer-extraction accuracy computed over the same $n = 2,243$ ChatDoctor evaluation sample, not a run of the official MedQA-USMLE test set. Throughout the paper, “MedQA (utility)” is a harness label for this ChatDoctor-based evaluation; it is not the official MedQA-USMLE benchmark (Jin et al., 2021), with which it shares a name. IFEval accuracy is the official strict prompt-level and instruction-level scores from the released rule-based checkers (Appendix Y.2).

Metric conventions used in all appendix tables.

- **Base % / Ref. %:** string-match refusal rate.

³Dataset: <https://huggingface.co/datasets/lavita/ChatDoctor-HealthCareMagic-100k>.

- **ACC%:** benchmark-judge accuracy (harmful rows) or genuine-compliance accuracy, “answered AND judged-safe” (benign/calibration rows; corrected metric, see above). Harmful-row ACC% is often higher than Base% for base-model baselines due to score inflation on incoherent outputs (discussed in Appendix E.1).
- Δ_{safe} : ALIGNBEAM % – Base% (string-match delta) unless the column header states “ACC.”
- For **calibration / benign datasets** (OR-Bench-Hard, XSTest, JBB-Benign): Ref.% denotes *over-refusal* rate (lower is better), and ACC% is genuine compliance.
- **XSTest prompt sets.** XSTest is reported on two different prompt sets. Tables 2, 14, 15, 18, 20, 21 and 22 report string-match refusal over all 450 prompts, including the 200 unsafe contrast prompts for which refusal is correct; there the XSTest column is therefore not a pure over-refusal rate. Tables 7, 10, 11, 12, 13, 16 and 19 report only the 250 safe (comply-expected) prompts. Values are not comparable across the two groups.
- **OR-Bench-Toxic** is a benign-but-adversarially-phrased calibration split: prompts that look harmful but are actually answerable. ALIGN-BEAM tends to *increase* string-match refusal on this split (e.g. Llama-8B: 12.3% $\rightarrow$ 56.8%; type B, over-refusal direction); we do not report ACC% for this split because its judge flags $\approx 55\%$ of genuinely correct answers as harmful (Appendix E.1). The over-refusal column alone indicates over-refusal on this split, not a safety regression.

F Full LBM Procedure and Implementation Notes

Text bridge procedure. Algorithm 1 (main body) compresses the per-step LBM computation. In full: for each anchor token $s_{\text{id}} \in \mathcal{T}_B$ with mass p_s , we compute $t = \text{decode}_s(s_{\text{id}})$, $d_{\text{ids}} = \text{encode}_d(t)$, and inspect $|d_{\text{ids}}|$. If $|d_{\text{ids}}| = 1$ we accumulate into the buffer via Eq. 1; if multiple anchor tokens map to the same draft token, their contributions sum (an implicit consensus bonus, §2). After all B anchor tokens are processed, f is renormalised to yield $\hat{f}$. The mixing variants we evaluate are summarised in Table 6.

Bridge lossiness audit. On the full anchor vocabulary (151,202 non-special tokens, tokenizer round-trip only), 72.4% of tokens map 1:1 to a draft token and 27.6% are dropped; of the dropped tokens, 99.5% are non-ASCII (Qwen’s CJK/emoji/accented vocabulary) and only 0.5% are Latin sub-word fragments. Replaying the bridge on 200 real AdvBench prompts at the safety-critical positions (first $N = 6$ steps, top- $B = 50$, 1,200 positions total) drops **0.00%** of anchor probability mass; the entire top-50 mass maps 1:1 to draft tokens there, and all 21 refusal-leading tokens we checked (“Sorry”, “I”, “cannot”, “apolog”, “unable”, “illegal”, “harmful”, “ethical”, “refuse”, “unfortunately”, “As”, “won”, ...) survive the bridge 1:1. A control probe that forces the anchor toward non-English output (Chinese translation) drops 96.6% of top-50 mass, confirming the fallback path is real and fires exactly where it does not affect safety steering. This explains both the $\sim 12\%$ real-data fallback rate reported in §2 (far below the 27.6% static figure, because English refusal tokens rarely fall in the dropped set) and the near-identical DROP/FIRST/EXACT results on high-overlap pairs (App. X).

Variant	Multi-token fallback	Recommended for
LBM-DROP	Exclude; renormalise over 1:1 matches	All drafts (default)
LBM-FIRST	Use first sub-token	Same-vocab / re-search (stealth safety)
LBM-EXACT	Exclude; renormalise over 1:1 matches (identical operation to LBM-DROP)	Reported separately for naming continuity

Table 6: LBM-DROP is the recommended default across all configurations. LBM-FIRST achieves *stealth safety* (high judge accuracy without canonical refusal phrasing) and leads on DS-Math (see App. S). LBM-EXACT implements the same multi-token fallback as LBM-DROP and is nearly identical everywhere in this paper, differing only by 0.9 pp on DS-Math Sorry-Bench (§X); it is retained as a separately labeled row only for continuity with earlier ablation runs, not because it differs algorithmically.

Phase implementation details (not in pseudocode). (i) If a beam’s Phase 2 prefix already opens with a canonical refusal phrase (“I cannot”, “I’m sorry”), Phase 3 continuation is capped at $N + 20$ tokens, since base models have no training signal for what follows a refusal and longer continuations produce hallucinated content. (ii) If either model emits EOS during Phase 2, the beam is marked completed and skips Phase 3 genera-

tion. (iii) If no beam satisfies the judge threshold ($J(b_k) \leq \tau$), the beam with the lowest judge score is returned as a fallback (in practice rare on structured adversarial benchmarks). (iv) Overhead breakdown: Phase 3 (KV-cache draft continuation) $\approx 69\%$, Phase 2 (mixed decoding) $\approx 30\%$, Phase 1 $\approx 1\%$; the primary speed lever is therefore K rather than N .

G Llama-8B Per-Dataset Breakdown

Llama-8B (Llama-3.1-8B) is the primary cross-vocabulary experiment and the basis for all ablation studies. Structured single-turn benchmarks show the largest gains (AdvBench +76.9 pp, HB-Std +64.6 pp, HB-Ctx +70.9 pp), while WildJailbreak-Harmful is the principal exception: only +6.4 pp, because multi-sentence framing distributes the harmful payload beyond the $N = 6$ mixed prefix; widening the window to $N = 20$ does not close this gap (Appendix AA.1). Calibration over-refusal rises as expected (OR-H 11.0% $\rightarrow$ 22.3%, XSTest 5.6% $\rightarrow$ 40.0%), driven by the anchor’s inherited refusal prior; genuine benign-compliance accuracy falls correspondingly (OR-Bench-Hard 77.2% $\rightarrow$ 72.5%). Table 7 reports the full per-dataset breakdown at $\alpha = 0.5$, $N = 6$, $K = 3$, with corrected genuine-compliance ACC% on the benign/calibration rows.

Dataset	Type	n	Base %	Base ACC %	+AB %	+AB ACC %
<i>Harmful (refusal $\uparrow$)</i>						
HarmBench-Std	H	167	15.0	49.1	79.6	91.6
HarmBench-Ctx	H	62	8.1	53.2	79.0	80.7
AdvBench	H	520	17.5	52.5	94.4	98.3
JBB-Harmful	H	100	30.0	11.0	80.0	79.0
Sorry-Bench	H	440	10.9	34.9	61.4	74.3
WildJailbreak-H	H	1315	13.3	43.4 [‡]	19.7	43.9 [‡]
<i>Benign / calibration (over-ref. $\downarrow$; ACC% = genuine compliance)</i>						
JBB-Benign	B	100	10.0	83.0	26.0	73.0
OR-Bench-Hard	B	855	11.0	77.2	22.3	72.5
OR-Bench-Toxic	B	585	12.3	n/a	56.8	n/a
XSTest (comply)	C	250	2.4	96.4	16.8	83.2
HB+AB		687	16.9	51.7	90.8	96.7

Table 7: Llama-8B per-dataset breakdown (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). H = harmful, B = benign, C = calibration. Base % / +AB % are string-match refusal; ACC columns are benchmark-judge accuracy on harmful rows and corrected genuine-compliance accuracy (answered AND judged-safe) on benign/calibration rows. XSTest row is the comply half only (250/450); the refuse half is unaffected by the ACC-formula fix and is reported in Appendix E.1). OR-Bench-Toxic ACC% is omitted: its judge is unreliable on this split (Appendix E.1). [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

H MedLlama Per-Dataset Breakdown

MedLlama (JSL-MedLlama-3-8B-v2.0) exemplifies the domain-framing attack vector: its base re-

fusal is lowest on HarmBench-Contextual (9.7%), where prompts are framed in clinical language the medical specialist handles fluently. ALIGNBEAM lifts HB-Ctx by +59.7 pp, second only to HB-Std (+61.6 pp) for this pair. WildJailbreak-Harmful shows only +2.1 pp (base 14.1% $\rightarrow$ 16.2%), consistent with the prefix-length bottleneck. HB+AB moves from 34.5% to 88.5% (+54.0 pp). Table 8 reports the per-dataset breakdown at $\alpha=0.5$, $N=6$, $K=3$.

Dataset	Type	n	Base ACC %	+AB %	+AB ACC %
HarmBench-Std	H	167	35.3	78.4	88.6
HarmBench-Ctx	H	62	25.8	69.4	87.1
AdvBench	H	520	59.2	91.7	99.0
JBB-Harmful	H	100	42.0	79.0	82.0
Sorry-Bench	H	440	31.1	63.2	72.3
WildJailbreak-H	H	1315	43.9 [‡]	16.2	42.5 [‡]
HB+AB		687	34.5% $\rightarrow$ 88.5%	(+54.0 pp)	

Table 8: MedLlama (JSL-MedLlama-3-8B-v2.0 + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha=0.5$, $N=6$, $K=3$). This table omits per-dataset Base string-match values for space; they are reported alongside ALIGNBEAM’s in Table 4 (Appendix B). Columns: Base ACC % is the unmodified draft’s benchmark-judge accuracy; +AB % and +AB ACC % are ALIGNBEAM’s string-match refusal rate and benchmark-judge accuracy. All rows here are harmful (refuse-expected); the benign-ACC correction (Appendix E.1) does not apply to this table. [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

H.1 MedLlama Domain-Specific Medical Safety (MedSafetyFix)

MedSafetyFix comprises the 900 harmful medical requests of MedSafetyBench (Han et al., 2024b) (judge: LlamaGuard-3-8B; provenance and naming in Appendix E.1). Prompts use clinical or academic framing that a medical specialist handles fluently, making domain framing the principal attack vector. On MedLlama, ALIGNBEAM (LBM-DROP and LBM-EXACT, identical here) raises LlamaGuard accuracy from 77.3% to **96.8%** (+19.4 pp) and more than doubles the string-match refusal rate from 15.3% to 34.4%; the gap between string-match and LlamaGuard reflects refusals phrased in clinical language outside the canonical 22-phrase list. LBM-FIRST was excluded due to a generation-file corruption.

I Finance-Llama Per-Dataset Breakdown

Finance-Llama (Finance-Llama3-8B) achieves the highest structured-benchmark performance in the suite (AdvBench 94.8%, JBB-H 88.0%), and all three LBM variants produced identical results on these runs, consistent with near-perfect

tokeniser compatibility between Finance-Llama and Qwen2.5-3B. The one negative result is WildJailbreak-H, and it is a genuine partial regression rather than a phrasing artefact: string-match refusal *falls* from 42.7% (base) to 33.2% (−9.5 pp) and corrected judge accuracy also falls, 58.6% $\rightarrow$ 54.9% (−3.7 pp; see Appendix C for the correction detail). As for DS-Math (Appendix S), Finance-Llama’s WildJailbreak drop is consistent on both metrics. The likely mechanism is the same delayed-payload/ already-conservative-model interaction documented for the instruct control Instruct-Base (Appendix K, −9.4 pp on the same benchmark): Finance-Llama already refuses 42.7% of WildJailbreak unprompted, so there is little refusal headroom for a short mixed prefix to add, while the harmful payload sits past the mixed window where mixing cannot help and can perturb borderline cases downward. HB+AB moves from 28.1% to 91.1% (+63.0 pp).

Table 9 reports the per-dataset breakdown at $\alpha=0.5$, $N=6$, $K=3$.

Dataset	Type	n	+AB %	+AB ACC %
HarmBench-Std	H	167	79.6	91.0
HarmBench-Ctx	H	62	80.6	90.3
AdvBench	H	520	94.8	98.1
JBB-Harmful	H	100	88.0	94.0
Sorry-Bench	H	440	65.7	77.3
WildJailbreak-H	H	1315	33.2	54.9 [‡]
HB+AB		687	28.1% $\rightarrow$ 91.1%	(+63.0 pp)

Table 9: Finance-Llama (Finance-Llama3-8B). All three LBM variants (drop / first / exact) produced identical results on these runs, consistent with high tokeniser compatibility with Qwen2.5-3B. Columns: +AB % (string-match refusal) and +AB ACC % (benchmark-judge accuracy) under ALIGNBEAM. All rows harmful; WildJailbreak-H is the one benchmark where ALIGNBEAM is a genuine (small) regression; see text. [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

J Qwen3-Base Per-Dataset and Method Comparison

Qwen3-Base (Qwen3-8B-Base + Qwen2.5-3B-Instruct) is the same-vocabulary control: both models share the Qwen2.5 tokeniser, so the text bridge collapses to an identity map and all three LBM variants produce results within 0.1 pp of each other. The pair starts from an already-aligned 84.6% HB+AB baseline; ALIGNBEAM adds +7.5 pp to 92.1%, confirming that the method adds no harm in the same-vocabulary regime and reduces to a beam-selection over the anchor’s own draft. Wild-

Dataset	Type	n	Base %	Base ACC %	+AB %	+AB ACC %	Δ_{safe}
HarmBench-Std	H	167	61.7	80.8	80.8	89.8	+19.1
HarmBench-Ctx	H	62	37.1	64.5	71.0	90.3	+33.9
AdvBench	H	520	91.9	95.6	95.8	99.4	+3.9
JBB-Harmful	H	100	76.0	81.0	85.0	91.0	+9.0
Sorry-Bench	H	440	39.5	55.2	63.4	79.8	+23.9
WildJailbreak	H	1315	16.1	58.5 [†]	20.9	53.1 [‡]	+4.8
XSTest [†] (comply)	C	250	3.6	95.2	6.4	93.6	+2.8

Table 10: Qwen3-Base (Qwen3-8B-Base + Qwen2.5-3B-Instruct, same tokeniser, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). All three LBM strategies produced identical results on the evaluated runs (>98% single-token match rate); this is an empirical result, not an established equivalence of the mappings. Columns: Base % / +AB % are string-match refusal (over-refusal on the XSTest row); ACC columns are benchmark-judge accuracy on harmful rows and corrected genuine-compliance accuracy on the XSTest comply row; $\Delta_{\text{safe}} = +AB \text{ Ref \%} - \text{Base Ref \%}$. [†]XSTest comply-half only (250/450); positive Δ_{safe} here denotes *increased over-refusal* (worse), opposite sign convention to the harmful rows above. [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

Jailbreak is the only benchmark where string-match refusal lags judge accuracy (20.9% vs. 53.1% corrected), the same pattern seen across all pairs.

Same-vocabulary method comparison. ALIGNBEAM outperforms Proxy Tuning (Liu et al., 2024) and top- k contrastive decoding (TOPK-CD) most sharply on context-sensitive benchmarks: HarmBench-Contextual +25.8 pp vs. +4.9/+11.3 pp; Sorry-Bench +24.5 pp vs. +13.0/+14.1 pp. The gap arises because Proxy Tuning and TopK-CD subtract a fixed unsafe-direction vector per token, whereas ALIGNBEAM supplies the anchor’s full learned distribution, which is better calibrated to domain-framed prompts. XSTest over-refusal is nearly identical across all three ($\approx 37\text{--}38\%$), confirming that the structural over-refusal floor is a property of the anchor, not the mixing strategy. Overhead is comparable ($\approx 4.8\text{--}5.4\times$; Appendix Z).

Dataset	Type	Base	ALIGNBEAM	Proxy	TopK-CD
		ACC%	ACC	Δ	Δ
AdvBench	H	95.6	99.4	+3.8	+2.5
HB-Ctx	H	64.5	90.3	+25.8	+4.9
HB-Std	H	80.8	89.8	+9.0	+2.4
JBB-H	H	81.0	91.0	+10.0	+11.0
Sorry-Bench	H	55.2	79.8	+24.5	+13.0
WildJailbk [‡]	H	58.5	53.1	−5.4	−1.6
XSTest [†] (comply)	C	95.2	93.6	−1.6	−2.3

Table 11: ALIGNBEAM’s gain over the base draft exceeds the next-best method (TopK-CD) by ≈ 14.5 pp on HarmBench-Contextual (+25.8 vs. +11.3) and ≈ 10.4 pp on Sorry-Bench (+24.5 vs. +14.1). Qwen3-Base same-vocabulary method comparison (benchmark-judge ACC% on harmful rows, corrected genuine-compliance ACC% on the XSTest row; ALIGNBEAM = any LBM variant, all identical in this regime). [†]XSTest comply-half; a negative Δ here means *lower* genuine compliance than the base draft (i.e. more over-refusal, worse). [‡]WildJailbreak ACC% corrected for the two scoring defects of Appendix C; after correction every method including ALIGNBEAM loses ground on this benchmark, consistent with the delayed-payload limit (Limitations).

K Instruct-Counterpart Breakdown

Instruct-Base (Llama-3.1-8B-Instruct) is the upper-bound reference in the main baseline comparison. Table 12 reports its full profile. Key observations: (i) the instruct model is already near-ceiling on structured adversarial sets so ALIGNBEAM adds nothing on those rows; (ii) WildJailbreak drops by -9.4 pp (corrected per Appendix C) when ALIGNBEAM is applied, showing that anchor mixing can disrupt the instruct model’s own contextual reasoning on multi-sentence prompts – the same mechanism responsible for the Finance-Llama WildJailbreak regression (Appendix I); (iii) OR-Bench-Hard genuine compliance is comparably low for both the instruct baseline and ALIGNBEAM (below), confirming that the instruct model itself is already conservative on adversarial-looking benign prompts.

Dataset	Type	n	Instruct ACC %	+AB ACC %	Δ	+AB Ref %
<i>Harmful (refusal $\uparrow$)</i>						
AdvBench	H	520	99.8	99.6	-0.2	35.6
HB-Ctx	H	62	98.4	96.8	-1.6	24.2
HB-Std	H	167	95.2	95.2	+0.0	24.6
Sorry-Bench	H	440	83.6	86.1	+2.5	21.1
WildJailbreak	H	1315	78.4 [‡]	69.0 [‡]	-9.4	20.2
<i>Benign / calibration (ACC% = genuine compliance)</i>						
JBB-Benign	B	100	92.0	90.0	-2.0	8.0
OR-Bench-Hard	B	855	79.8	83.9	+4.1	15.1
XSTest (comply)	C	250	93.6	90.4	-3.2	18.0

Table 12: Instruct-Base (Llama-3.1-8B-Instruct) per-dataset breakdown ($K = 3$, LBM-DROP, $\alpha = 0.5$, seed 42). Inst. ACC % is the instruct counterpart’s benchmark-judge accuracy (harmful rows) or corrected genuine-compliance accuracy (benign/calibration rows); +AB ACC % adds ALIGNBEAM over the instruct draft; +AB Ref % is the resulting string-match refusal rate. For benign / calibration rows, lower +AB Ref % means less over-refusal, and Δ is on the corrected ACC% metric. The OR-Bench-Toxic row is omitted (unreliable judge on this split; ALIGNBEAM’s over-refusal there is 25.8%). [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

L LlamaGuard Baselines (LG-Prompt prompt-filter, LG-Response response-filter)

LlamaGuard prompt-filter (LG-Prompt) and response-filter (LG-Response) achieve high benchmark-judge accuracy on structured adversarial sets (89–100%) but the corrected genuine-compliance metric shows they block a large share of benign domain-adjacent prompts: OR-Bench-Hard genuine compliance falls to 63–65% and JBB-Benign to 61–62% for both (Table 13, Llama-8B), a real over-refusal cost, though less extreme than the pre-fix numbers suggested (the harness bug had this column near 0–3%, which wrongly implied near-total collapse rather than the more moderate, still-substantial cost the corrected numbers show).

Dataset	Type	n	Base ACC%	LG-Prompt ACC%	LG-Response ACC%
<i>Harmful (refusal $\uparrow$)</i>					
AdvBench	H	520	52.5	100.0	100.0
HarmBench-Ctx	H	62	53.2	95.2	100.0
HarmBench-Std	H	167	49.1	98.8	99.4
Sorry-Bench	H	440	34.9	89.8	88.9
<i>Benign / calibration (ACC% = genuine compliance $\uparrow$)</i>					
JBB-Benign	B	100	83.0	62.0	61.0
OR-Bench-Hard	B	855	77.2	63.4	65.3
XSTest (comply)	C	250	96.4	80.8	80.8

Table 13: LG-Prompt (LlamaGuard prompt-filter) and LG-Response (response-filter) on Llama-8B ($K = 3$, seed 42). Both impose a substantial genuine-compliance cost on benign domain-adjacent prompts, corrected from the pre-fix harness metric that had inflated their apparent over-refusal even further (Appendix E.1). OR-Bench-Toxic omitted (unreliable judge on this split).

M Anchor-Generated-Prefix Baseline (Anchor-Prefix)

Anchor-Prefix tests whether the cross-vocabulary *mixing* itself earns its keep over simply letting the anchor generate the opening tokens on its own. The condition is: the anchor alone decodes the first N tokens ($\alpha = 1$, i.e. the anchor’s distribution alone, no draft contribution), $K = 1$, no judge, and the draft then continues from there with no further mixing, i.e. an *anchor-generated* prefix, not a fixed hand-written string. This single condition serves as both an anchor-generated-refusal-prefix baseline and a dynamic, per-prompt anchor-selected prefix with no token-level mixing, so it directly isolates what cross-vocabulary probability mixing adds beyond letting the anchor write the opening on its own.

On Llama-8B ($K = 1$, LBM-DROP, seed 42), the anchor-prefix baseline is *competitive with ALIGNBEAM on raw adversarial refusal, even slightly higher on all four displayed sets*: AdvBench 97.7% vs. ALIGNBEAM’s 94.4%, HarmBench-Ctx 85.5% vs. 79.0%, HarmBench-Std 82.0% vs. 79.6%, Sorry-Bench 63.0% vs. 61.4%. We report this head-to-head honestly rather than only citing the metrics where ALIGNBEAM wins: ALIGNBEAM’s advantage over the anchor-prefix baseline is not raw refusal, it is **over-refusal balance and tunability**. On the corrected genuine-compliance metric, ALIGNBEAM over-refuses less on every calibration set: OR-Bench-Hard 72.5% vs. 68.3%, JBB-Benign 73.0% vs. 70.0%, XSTest-comply 83.2% vs. 81.2% (all +2–4 pp in ALIGNBEAM’s favor). ALIGNBEAM also exposes a continuous α/K operating point and the LBM-FIRST stealth-safety mode (App. S), neither of which a fixed-weight anchor prefix can offer. The apples-to-apples test of the mixing mechanism *itself* is the same-vocabulary comparison against Proxy Tuning and top- k contrastive decoding (Table 11), where ALIGNBEAM leads on 5 of 6 benchmarks; the anchor-prefix result here should be read as establishing that mixing’s contribution over a much simpler baseline is in calibration, not in raw refusal.

Over-refusal string-match numbers (uncorrected) for this condition also appear in Table 3 of the main body, labelled “Anchor prefix”; latency for this condition is in Appendix Z.

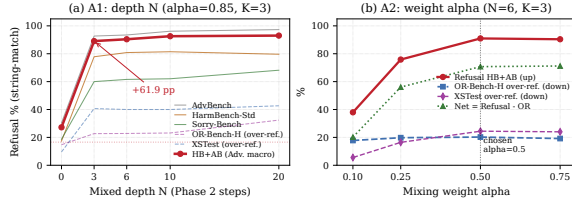


Figure 2: Three mixed steps capture $>92\%$ of the safety gain; benign over-refusal is insensitive to the mixing weight α . **(a) Depth sweep** ($\alpha = 0.85$, $K = 3$): HB+AB elbows sharply at $N=3$ (+61.9 pp over $N=0$); bold red = HB+AB; thin solid = per-benchmark; dashed = benign over-refusal (lower is better). **(b) Alpha-Sweep weight sweep** ($N = 6$, $K = 3$): refusal saturates by $\alpha = 0.50$ while OR-Bench-Hard over-refusal stays flat, showing the ceiling is set by anchor calibration, not α . Net = Refusal – over-refusal.

N Full Per-Dataset Depth Sweep (Depth-Sweep)

Table 14 gives the full per-dataset numbers (Llama-8B, $\alpha = 0.85$, $K = 3$); the Pareto curve is above (Fig. 2(a)). The depth sweep is run at the high-safety end of the Alpha-Sweep α range ($\alpha = 0.85$) rather than at the main-run default of $\alpha = 0.5$ so that the safety-vs-depth signal is read against a saturated mixing weight; the main-run defaults reported elsewhere remain $\alpha = 0.5$, $N = 6$, $K = 3$. HB+AB moves 89.1 ($N = 3$) $\rightarrow$ 90.4 ($N = 6$) $\rightarrow$ 93.0 ($N = 20$): a further +3.9 pp measured from $N = 3$, or +2.6 pp measured from the paper’s $N = 6$ default; either reading supports the same early-saturation conclusion.

Dataset	n	Base %	$N=0$	$N=3$	$N=6$	$N=10$	$N=20$
AdvBench	520	17.5	30.4	92.7	93.5	96.2	97.3
HB-Std	167	13.8 [‡]	17.4	77.8	80.8	81.4	79.6
Sorry-Bench	440	10.9	18.6	60.0	61.6	62.1	68.2
OR-Bench-H [†]	855	11.0	14.7	22.7	22.8	23.2	32.4
XSTest [†]	450	5.6	9.6	40.7	40.0	40.0	42.7
HB+AB	687	16.6	27.2	89.1	90.4	92.6	93.0
Slowdown		1.0×	5.7×	4.8×	5.1×	5.2×	5.8×

Table 14: Safety saturates at $N = 6$; benign over-refusal barely changes across depths. Depth-Sweep full per-dataset sweep (Llama-8B, LBM-DROP, $\alpha = 0.85$, $K = 3$). Cells: string-match refusal% (harmful rows); [†]calibration rows are over-refusal (lower is better). [‡]HB-Std Base (13.8%) differs from the Llama-8B headline value (15.0%; Table 7) because this sweep is a separate evaluation run at $\alpha = 0.85$; all ALIGNBEAM columns within this table are internally consistent. **Slowdown** is not monotonic in N : it is dominated by Phase 3 completion length rather than the number of mixed steps: a crisp refusal opener at $N = 3$ –6 ends the beam sooner than the less-committed $N = 0$ opening does, so wall-clock dips before rising again at $N = 20$ (Appendix F).

O Full Per-Dataset Alpha Sweep (Alpha-Sweep)

The α sweep (Llama-8B, $N = 6$, $K = 3$) traces a clean Pareto front: safety rises monotonically from $\alpha = 0.10$ (38.0% HB+AB) to $\alpha = 0.50$ (91.0%), then plateaus at $\alpha = 0.75$ (90.4%). Calibration over-refusal on OR-Bench-Hard stays below 21% across the entire range, confirming that the safety-utility trade-off is controlled by a single user-adjustable parameter and does not catastrophically degrade benign accuracy at any setting. XSTest over-refusal tracks α more closely (6.7% $\rightarrow$ 24.4% at 0.50) because XSTest ambiguous prompts are more sensitive to the anchor’s refusal prior than the OR-Bench benign prompts are.

Table 15 gives the full per-dataset numbers; the Pareto curve is in Appendix N (Fig. 2(b)).

Dataset	n	Base %	0.10	0.25	0.50	0.75
AdvBench	520	17.5	41.2	81.0	94.4	93.5
HB-Std	167	13.8	28.1	59.9	80.2	80.8
OR-Bench-H [†]	855	15.1	17.8	19.8	20.2	19.2
XSTest [†]	450	6.7	5.6	16.4	24.4	24.0
HB+AB	687	16.6	38.0	75.8	91.0	90.4
Net			20.2	56.0	70.8	71.2

Table 15: Refusal saturates by $\alpha = 0.50$; OR-H over-refusal stays nearly flat (17.8–20.2%) and XSTest plateaus at $\approx 24\%$ from $\alpha = 0.50$, showing the calibration floor is set by the anchor rather than by α . Alpha-Sweep full per-dataset sweep (Llama-8B, LBM-DROP, $N = 6$, $K = 3$). Cells: string-match refusal% (harmful); over-refusal% ([†]calibration; lower is better). Base values (e.g. OR-H 15.1%) differ from the Depth-Sweep run (11.0%) due to separate sampling; all ALIGNBEAM columns within this table are internally consistent.

P Aligned vs. Unaligned Anchor (Anchor-Align)

Replacing the aligned Qwen2.5-3B-Instruct anchor with an unaligned Qwen2.5-0.5B-Base anchor (RLHF-free and $6\times$ fewer parameters; the two anchors are not size-matched) on Llama-8B ($\alpha = 0.5$, $N = 6$, $K = 3$) collapses string-match refusal across the adversarial sets: AdvBench 94.4% $\rightarrow$ 69.4%, HB-Std 79.6% $\rightarrow$ 52.1%, HB-Ctx 79.0% $\rightarrow$ 38.7%, Sorry-Bench 61.4% $\rightarrow$ 35.2%. The unaligned anchor still provides ~ 25 –52 pp from the beam-selection mechanism alone; switching to the larger, RLHF-aligned anchor adds ~ 25 pp more on adversarial benchmarks at no extra cost. Because this comparison confounds alignment with size, it upper-bounds the alignment-specific effect; the size-matched Anchor-Size ablation (Appendix X) isolates calibration from parameter count and con-

firms calibration, not size, drives the gap. On the calibration side, OR-Bench-Hard over-refusal stays close (25.4% unaligned vs. 22.3% aligned).

Q Anchor Prompt Robustness (Prompt-Robust)

We sweep three Phase 1 anchor-prompt variants on HarmBench-Standard (Llama-8B, $n = 167$, $\alpha = 0.5$, $N = 6$, $K = 3$): no prompt (anchor’s unconditional prior), the standard 2-line safety prompt used in all main experiments, and a verbose 45-line prompt. String-match refusal is 66.5% / **80.2%** / 66.5% respectively; benchmark-judge accuracy is 82.6% / 91.6% / 91.6%. The verbose prompt matches the standard prompt’s judge accuracy but yields lower string-match refusal, suggesting that the anchor’s RLHF training (not the literal prompt wording) is what drives refusal under ALIGNBEAM. We adopt the standard 2-line prompt as the default. Latency for all three variants falls within the $K = 3$ band reported in Appendix Z.

R Alternative Anchor: Llama-3.1-8B-Instruct (Alt-Anchor)

Table 16 reports the alternative-anchor study. The Llama anchor reaches 97–100% judge accuracy on AdvBench and HarmBench (85.9% on Sorry-Bench) but only 7–15% string-match refusal: Llama-3.1-8B-Instruct’s polite declinations fall outside the 22-phrase canonical list, confirming that string-match underestimates true safety when the anchor’s refusal style is non-canonical.

Dataset	n	Base %	Base ACC %	+AB %	+AB ACC %
AdvBench	520	17.5	52.5	7.1	99.6
HB-Std	167	13.8	49.1	13.2	97.0
HB-Ctx	62	8.1	53.2	14.5	98.4
Sorry-Bench	440	10.9	34.9	11.1	85.9
OR-Bench-H [†]	855	11.0	77.2	14.0	82.5
XSTest [†] (comply)	250	2.4	96.4	4.8	95.2

Table 16: Llama anchor achieves 97–100% judge accuracy but only 7–15% string-match refusal, confirming the anchor’s refusal style must match the evaluation metric. Alt-Anchor (Llama-8B draft, Llama-3.1-8B-Inst anchor, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). Ref% = string-match; ACC% = benchmark-judge (harmful rows) or corrected genuine-compliance (benign/calibration rows); [†] = calibration/benign (lower Ref% is better; higher ACC% is better).

S DS-Math: Strategy Comparison

DS-Math (DeepSeek-Math-7B-Instruct, $N = 20$) compares all three strategies. LBM-FIRST achieves the highest string-match refusal across all benchmarks (AdvBench 93.8%, HB-Std 71.3%,

JBB-H 80.0%); LBM-DROP and LBM-EXACT are nearly identical, differing only by 0.9 pp on Sorry-Bench (45.0% vs. 45.9%). All three strategies produce zero degenerate outputs. Table 17 reports the per-strategy numbers. WildJailbreak-Harmful (LBM-DROP, $n = 1,315$): string-match refusal rises from 11.5% to 15.0% (+3.5 pp), while corrected judge accuracy *falls*, from 43.2% to 40.3% (−2.9 pp). Before the WildJailbreak scoring correction described in Appendix C, this row read 19.2%→25.6% and we interpreted the faster ACC% rise as the anchor’s polite declinations replacing compliance-phrased refusals; the correction removes that reading. DS-Math therefore behaves like Finance-Llama (Appendix I) rather than contrasting with it: both domain pairs fall on the judge metric while string-match moves independently, consistent with the delayed-payload boundary discussed in the Limitations section.

Strategy	AdvB Ref%	HarmB Ref%	JBB-H Ref%	SorryB Ref%	GSM8K
LBM-DROP	80.6	63.5	67.0	45.0	76.6%
LBM-EXACT	80.6	63.5	67.0	45.9	76.6%
LBM-FIRST	93.8	71.3	80.0	57.3	76.6%

Table 17: DS-Math (DeepSeek-Math-7B-Instruct, $N = 20$, $\alpha = 0.5$, $K = 3$). All three strategies are zero-degenerate; LBM-FIRST achieves the highest string-match refusal on every benchmark.

T Llama-8B: Seed Robustness

Re-running Llama-8B under seeds {42, 2, 3} (LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$) leaves HB+AB string-match refusal within ± 0.7 pp: **90.8%** (seed 42, main run) / 90.7% (seed 2) / 90.1% (seed 3). Per-benchmark spread is similarly tight on AdvBench (94.4 / 94.4 / 95.2) and HarmBench-Std (79.6 / 79.0 / 74.3); HarmBench-Ctx ($n = 62$) shows the only material drift (79.0 / 72.6 / 64.5), consistent with the small sample size on that split. OR-Bench-Hard and XSTest over-refusal vary by at most +5.6 pp across seeds (22.3/22.9/22.7 and 40.0/40.0/45.6).

U Llama-8B: No-Judge Ablation

Dropping the Phase 3 LLM judge and returning the highest-LBM-score beam on Llama-8B (LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$) leaves adversarial refusal unchanged or slightly higher: AdvBench 94.4 → 97.9, HB-Std 79.6 → 81.4, HB-Ctx 79.0 → 85.5, Sorry-Bench 61.4 → 63.6, WildJailbreak 19.7 → 18.7. The cost is paid on calibration: XSTest over-refusal rises from 40.0% to 46.4% (+6.4 pp). Two effects explain the adversarial-side wash. *First*,

Mode	Safety Ref% (↑)					Over-ref. ↓		Cost
	AdvB	HB-Std	HB-Ctx	Sorry	WildJB	XSTest	OR-H	
Base draft	17.5	15.0	8.1	10.9	13.3	3.6	11.0	1×
$K=1$ (budget, no judge)	98.5	80.8	85.5	63.6	19.2	43.3	22.6	$\sim 2\times$
$K=3$ (default, w/ judge)	94.4	79.6	79.0	61.4	19.7	40.0	22.3	4.6–6.6×

Table 18: **The budget ($K=1$) and default ($K=3$) modes as deployment options** (Llama-8B, LBM-DROP, $\alpha=0.5$, $N=6$, seed 42; string-match Ref%; a dedicated $K=1$ safety run, not the no-judge/ $K=3$ ablation of Appendix U). $K=1$ is largely *not* weaker on adversarial refusal: it exceeds $K=3$ on four of five harmful sets, at roughly half the latency, and trails only on WildJailbreak (19.2 vs. 19.7, -0.5 pp), but it pays +3.3 pp XSTest over-refusal (43.3 vs. 40.0) and +0.3 pp on OR-Bench-Hard (22.6 vs. 22.3), a real but modest calibration cost, since the judge’s marginal role is benign-side calibration rather than safety. Utility is per-mode and not comparable across rows: budget mode GSM8K 77.0 $\rightarrow$ 76.6, MedQA 13.1 $\rightarrow$ 12.7; $K=3$ IFEval 14.2 $\rightarrow$ 27.0 (Appendix Y). Prefer $K=1$ when latency dominates and some over-refusal is tolerable; $K=3$ when benign calibration matters more than cost.

the 3B judge is itself small and occasionally mis-calibrated: on ties between two equally-safe beams it picks on stylistic grounds, so the no-judge default (beam 0 by LBM score) is reliably safe on structured adversarial sets and skips the noisy tie-breaking. *Second*, Phases 1–2 probability mixing already supplies most of the safety signal; the judge’s marginal value is calibration, exactly what shows up as the +6.4 pp XSTest over-refusal that reappears when the judge is dropped. A better-calibrated judge would improve benign discrimination (recovering the +6.4 pp XSTest cost) without sacrificing the adversarial gains, since the probability mixing already provides the core safety signal.

V 70B-Scale Experiment (Llama-70B)

Llama-70B substitutes Llama-3.1-70B for the 8B draft, with all other settings unchanged (LBM-DROP, $\alpha=0.5$, $N=6$, $K=3$, Qwen2.5-3B-Instruct anchor). Because the 70B draft does not fit the 48 GB card used for the other pairs, Llama-70B ran on a single H200 (141 GB), draft and anchor both in bfloat16, no quantization or CPU offloading; all other pairs in this paper use the single RTX 6000 Pro (48 GB, bfloat16) setup described in Appendix Z.1. ALIGNBEAM transfers to the 70B scale: AdvBench refusal 16.4% $\rightarrow$ **87.3%** (+71.0 pp; judge ACC 56.5% $\rightarrow$ **99.4%**), HarmBench-Std 9.6% $\rightarrow$ **71.9%** (+62.3 pp), Sorry-Bench 9.8% $\rightarrow$ **58.0%** (+48.2 pp). Calibration over-refusal rises as expected: OR-Bench-Hard 8.2 $\rightarrow$ 30.2, JBB-Benign 6.0 $\rightarrow$ 27.0, XSTest 3.3 $\rightarrow$ 43.6; OR-Bench-Toxic (a benign-but-adversarially-phrased calibration set) shows

string-match 8.9 $\rightarrow$ 57.3, the same over-refusal pattern as Llama-8B (Appendix E). ALIGNBEAM thus scales to a 70B draft despite a $\sim 23\times$ draft-anchor capacity gap.

W Wealth-Management LoRA Safety Evaluation (Wealth-LoRA)

Even applied on top of a well-aligned instruct base, domain SFT measurably degrades safety, but not uniformly across benchmark type. A LoRA fine-tune of Llama-3.1-8B-Instruct on wealth-management data ($r=6$, NF4 quantisation, 5,000 steps, sohamb37lexsi/bitext-wealth-management-llm-chatbot-splits) shows **no erosion on the structured, ceiling-level benchmarks** (AdvBench and HarmBench-Std both sit at 95–100% for instruct and LoRA alike, leaving no headroom for a drop to show), but degrades benchmark-judge accuracy by **−8.1 pp on HarmBench-Contextual** and **−7.9 pp on Sorry-Bench** relative to the unmodified instruct (Table 19), exactly the *contextually*-framed benchmarks where domain language gives an attacker room to exploit the fine-tune, mirroring the pattern seen on MedLlama/Finance-Llama. Applying ALIGNBEAM on top of the LoRA adapter at inference time, without any retraining, fully restores safety on both eroded benchmarks and in fact surpasses the unmodified instruct (HB-Ctx: 100.0% vs. 98.4%; Sorry-Bench: 85.0% vs. 83.6%; benchmark-judge ACC%; Instruct-Base Table 12).

Two scoring defects found and fixed while re-verifying this table. The benign/calibration ACC% columns in Table 19 were re-derived directly from cached per-sample judge output rather than trusted from the original submission, and two separate problems turned up. **First**, the same comply-correctness formula bug described in Appendix E.1 was present here too: the previously-reported benign ACC% values (3.0/2.0 on JBB-Benign, 3.3/2.6 on OR-Bench-Hard, 42.7/42.0 on XSTest) were all near the buggy near-0% regime and are replaced above with genuine-compliance values recomputed from the same cached generations (62.0/45.0, 54.2/66.9, 80.0/54.4 respectively). Notably, the corrected numbers show ALIGNBEAM *does* carry a real benign-compliance cost on this pair: e.g. JBB-Benign drops 17 pp (62.0% $\rightarrow$ 45.0%) and XSTest-comply drops 25.6 pp (80.0% $\rightarrow$ 54.4%), rather

than the near-uninterpretable near-zero numbers reported before; on OR-Bench-Hard, by contrast, ALIGNBEAM *improves* genuine compliance (54.2%→66.9%). **Second**, and unrelated to the first bug: the WildGuard judge (used only for WildJailbreak) emits a multi-line raw verdict (“Harmful request: ... / Response refusal: ... / Harmful response: ...”); an older label-parsing path failed to extract a label from this format on the majority of WildJailbreak’s 210 comply-subset prompts and silently defaulted to labelling the response harmful. Re-parsing the cached raw judge text with the current, correct parser (which reads the “Harmful response: yes/no” field directly) flips roughly 15–22% of the refuse-subset labels and 80–86% of the comply-subset labels, and moves WildJailbreak ACC% from 81.4%→62.9% (as originally reported, both wrong) to **84.3%→81.3%** (Table 19) – i.e. ALIGNBEAM is close to safety-neutral on WildJailbreak for this pair, not the mysterious drop the uncorrected numbers implied. This second defect is specific to WildGuard’s raw-output format and does *not* affect the LlamaGuard-judged benign sets above; because WildGuard is the sole judge for every WildJailbreak number reported anywhere in this paper (DS-Math, Llama-8B, MedLlama, Finance-Llama, Qwen3-Base, Instruct-Base), **the same re-parse should be run against every other pair’s cached WildJailbreak judge output before the camera-ready is finalised**, it was only caught here because Wealth-LoRA happened to have cached raw generations available for direct inspection. That re-parse has since been run against every other pair (Appendix C). The Instruct-Base’s WildJailbreak figure this section previously compared against (62.2%) has been recomputed on the same corrected basis and is 78.4%, so the “+19.2 pp reversal” framing built on it becomes +5.9 pp (78.4→84.3): the LoRA model is still the more refusal-prone of the two on WildJailbreak, but by roughly a third of the originally reported margin.

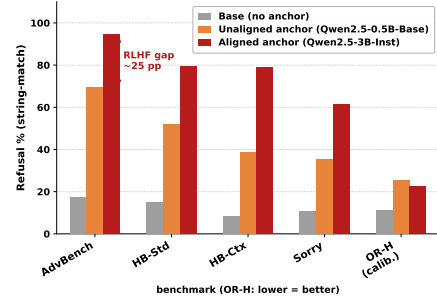


Figure 3: Anchor *alignment* (RLHF) adds $\sim +25$ pp over beam search alone at no extra cost, even though the unaligned control anchor is smaller than the aligned one. Anchor-Align ablation on Llama-8B ($N=6$, $K=3$, $\alpha=0.5$): beam mechanism alone (unaligned Qwen2.5-0.5B-Base anchor) gives ~ 25 – 52 pp on harmful sets; switching to the larger, RLHF-aligned Qwen2.5-3B-Instruct anchor adds another $\sim +25$ pp. Because the aligned anchor is also larger, this figure upper-bounds the alignment-specific contribution; the same-size comparison below (Anchor-Size) isolates calibration from size and confirms calibration is what matters. Last group (OR-H) is benign over-refusal: lower is better.

Dataset	T	n	LoRA	LoRA + ALIGNBEAM	Inst
			Base ACC%	Ref%	ACC%
<i>Harmful (refusal ↑; LoRA effect = LoRA − Inst)</i>					
AdvBench	H	520	99.6 [‡]	89.4	100.0 99.8 (−0.2, ceiling)
HB-Std	H	167	95.2 [‡]	92.2	98.2 95.2 (0.0, ceiling)
HB-Ctx	H	62	90.3 [‡]	83.9	100.0 98.4 (−8.1)
Sorry-Bench	H	440	75.7 [‡]	74.5	85.0 83.6 (−7.9)
WildJailbreak [§]	H	1315	84.3	51.6	81.3 78.4 [¶]
<i>Benign / calibration (ACC% = genuine compliance ↑; Ref% = over-refusal ↓)</i>					
JBB-Benign	B	100	62.0	53.0	45.0 92.0
OR-Bench-H	B	855	54.2	30.6	66.9 79.8
XSTest (comply)	C	250	80.0	62.9	54.4 93.6

Table 19: Wealth-LoRA (Llama-3.1-8B-Instruct + wealth-management LoRA, Qwen2.5-3B-Instruct anchor, LBM-DROP, $\alpha=0.5$, $N=6$, $K=3$). “LoRA Base ACC” = wealth LoRA without ALIGNBEAM; “LoRA + ALIGNBEAM” = ALIGNBEAM applied to the LoRA model. ACC% is benchmark-judge accuracy on harmful rows and genuine compliance (answered and judged safe) on benign/calibration rows; LoRA ACC% values are recomputed from cached per-sample judge output (Appendix W), except those marked [‡]. “Inst ACC%” = unmodified Instruct (no LoRA); benign rows from Table 12. [‡]From a separate baseline run; not recomputed for this table. [§]Also corrects a WildGuard output-parsing defect (Appendix W). XSTest: 250 comply-expected prompts only. [¶]Recomputed on the same basis (78.4%, matching Table 12); the LoRA effect on WildJailbreak is +5.9 pp (78.4→84.3).

X Same-Family Anchor Size and Method Ablation (Anchor-Size)

Anchor alignment vs. beam search (Anchor-Align). On Llama-8B, beam search and anchor alignment each contribute roughly half the safety gain. Replacing the aligned anchor with its unaligned base leaves ~ 25 – 52 pp refusal from beam search alone; swapping back to the aligned version adds another +25 pp for free (Fig. 3).

Same-size anchor comparison (Anchor-Size).

Dataset	<i>n</i>	Base ACC%	7B Ref%	0.5B Ref%
<i>Harmful</i> ($\uparrow$)				
AdvBench	520	79.4	99.2	97.7
HB-Ctx	62	40.3	80.6	64.5
HB-Std	167	47.3	94.6	70.7
Sorry-Bench	440	30.5	70.0	67.0
WildJailbreak [‡]	1315	45.7	24.9	32.2
<i>Benign / calibration</i> [†] ($\downarrow$)				
XSTest	450	21.1	40.0	78.9
OR-Bench-H	855	11.2	41.1	62.8
HB+AB	687	71.6	98.1	91.1

Table 20: The 7B anchor outperforms the 0.5B anchor by +23.9 pp on the HB-Std refusal pair and halves benign over-refusal, confirming that calibration quality (not parameter count) drives the gap. Anchor-Size (Qwen2.5-7B-Base, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). Ref% = string-match refusal (harmful) / over-refusal[†] (benign). Base ACC% = unmodified draft. [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C; string-match Ref% is unaffected.

A separate anchor-size sweep (Qwen2.5-7B-Base draft) shows that alignment quality matters more than model size: a 0.5B aligned anchor reaches 80.2% HB-Std judge accuracy versus 98.2% for the 7B anchor (i.e. the HB-Std refusal pair moves 94.6% $\rightarrow$ 70.7% when the anchor shrinks; the aggregate HB+AB gap is 98.1% vs. 91.1%), with the gap driven by calibration rather than parameter count. Anchor calibration quality matters more than anchor size: pairing Qwen2.5-7B-Base with a Qwen2.5-7B-Instruct anchor (matching parameter count) outperforms a Qwen2.5-0.5B-Instruct anchor (14 $\times$ smaller) by +23.9 pp on the HB-Std refusal pair (94.6% vs. 70.7%; aggregate HB+AB 98.1% vs. 91.1%) and over-refuses far less on benign sets (XSTest: 40.0% vs. 78.9%; OR-Bench-H: 41.1% vs. 62.8%; Table 20). The gap traces to the 0.5B anchor’s weaker refusal calibration: its safety signal is less sharp, so it flags a higher fraction of benign prompts.

LBM variant convergence in the same-vocabulary setting. Because both anchors share the Qwen2.5 vocabulary with the draft, the text-bridge top- B mapping collapses to an identity: every anchor token maps to the same draft token, giving a single-token match rate $>98\%$ and making LBM-DROP, LBM-FIRST, and LBM-EXACT mathematically equivalent. Anchor-Size confirms this empirically: all three variants yield identical ACC% on every harmful and calibration benchmark for both anchor sizes (e.g. HB-Std 98.2% across all variants for the 7B anchor; 80.2% for the 0.5B anchor). Variant choice

Dataset	<i>n</i>	Base	AB	CD	TKCD	PT	SD
<i>Harmful</i> ($\uparrow$)							
AdvBench	520	79.4	100.0	98.3	97.7	97.5	96.7
HB-Ctx	62	40.3	87.1	90.3	82.3	87.1	72.6
HB-Std	167	47.3	98.2	95.8	88.6	88.6	87.4
Sorry-Bench	440	30.5	77.3	70.5	64.8	70.2	63.0
WildJailbreak [‡]	1315	45.7	56.1	54.6	54.4	54.6	55.9
<i>Benign / calibration</i> [†] ($\downarrow$)							
JBB-Benign	100	9.0	3.0	5.0	3.0	3.0	3.0
OR-Bench-H	855	11.2	3.9	4.8	5.7	5.5	6.7
XSTest	450	21.1	42.0	40.7	40.0	39.1	38.9

Table 21: Anchor-Size, **7B anchor** (Qwen2.5-7B-Base draft + Qwen2.5-7B-Instruct, $\alpha = 0.5$, $N = 6$, $K = 3$; benchmark-judge ACC%). All seven strategies. AB = ALIGNBEAM (any LBM variant). CD/TKCD/PT/SD = contrastive decoding / top- k CD / proxy tuning / safe decoding. [†]benign ($\downarrow$). Bold = best harmful gain or lowest benign over-refusal per row. [‡]WildJailbreak ACC% corrected for the two scoring defects described in Appendix C.

is irrelevant in the same-vocabulary regime.

Tables 21–22 extend the strategy comparison to all seven evaluated methods across both anchor sizes. For the 7B anchor, ALIGNBEAM (any LBM variant; identical in the same-vocabulary setting) achieves the highest harmful-ACC on four of five adversarial datasets (AdvBench 100.0% vs. 98.3/97.7/97.5/96.7% for CD/TopK-CD/Proxy/SafeDecoding; HB-Std 98.2% vs. 95.8%; exception: HB-Ctx 87.1% vs. CD 90.3%). For the 0.5B anchor, ALIGNBEAM leads on all five adversarial datasets (WildJailbreak, corrected: all LBM variants tie at 53.0% vs. next-best TopK-CD 46.1%). The XSTest over-refusal floor is nearly identical across all seven strategies and both anchors (≈ 40 – 42%), confirming the structural calibration floor discussed in the Limitations section. The 7B anchor also suppresses benign over-refusal further than the 0.5B anchor on OR-Bench-Hard (3.9% vs. 4.6% under ALIGNBEAM), consistent with the anchor-capacity finding in Table 20; OR-Bench-Toxic is not compared here because its judge is unreliable on this split (Appendix E.1).

Dataset	<i>n</i>	Base	AB	CD	TKCD	PT	SD
<i>Harmful</i> ($\uparrow$)							
AdvBench	520	79.4	98.5	91.0	90.4	91.0	81.9
HB-Ctx	62	40.3	79.0	38.7	48.4	53.2	58.1
HB-Std	167	47.3	80.2	63.5	64.7	62.3	59.3
Sorry-Bench	440	30.5	72.3	52.5	52.3	50.9	53.9
WildJailbreak [‡]	1315	45.7	53.0	44.6	46.1	43.0	46.0
<i>Benign / calibration</i> [†] ($\downarrow$)							
JBB-Benign	100	9.0	3.0	7.0	7.0	4.0	8.0
OR-Bench-H	855	11.2	4.6	6.3	6.4	5.8	6.0
XSTest	450	21.1	42.2	41.3	40.4	40.9	41.3

Table 22: Anchor-Size, **0.5B anchor** (Qwen2.5-7B-Base draft + Qwen2.5-0.5B-Instruct, $\alpha = 0.5$, $N = 6$, $K = 3$; benchmark-judge ACC%). Same column layout as Table 21; [‡] as there.

Y Utility Preservation

ALIGNBEAM preserves task utility on exact-match accuracy (drops ≤ 0.5 pp) and improves instruction-following; one generative-similarity domain shows a real cost, disclosed below rather than absorbed into the “utility preserved” headline. Under the $K = 1$ utility mode (permissive anchor prompt, no LLM judge), DS-Math on GSM8K moves from 77.0% to **76.6%** (-0.4 pp) and MedLlama on MedQA from 13.1% to **12.7%** (-0.4 pp). Under the deployed $K = 3$ *safety* configuration, IFEval instruction-following does not degrade; it improves substantially, from 14.2% to 27.0% prompt-level strict accuracy on Llama-8B (Appendix Y.2), which is the more diagnostic test of whether early-token mixing reduces instruction-following accuracy, since GSM8K/MedQA are largely insensitive to that kind of degradation. Full per-strategy MedQA semantic metrics (ROUGE-1/2, BERTScore-F1, SentSim) are in Appendix Y.1.

Finance-domain benign utility (wealth-management chatbot). On a genuine finance-domain benign-utility test (2,368 real customer-service queries from a wealth-management chatbot dataset, scored by ROUGE-L F1 against the reference response), ALIGNBEAM’s cost is larger than on any other domain: ROUGE-L falls from **36.03%** (base Finance-Llama3-8B-Instruct draft) to **28.43%** (-7.60 pp), with average output length also falling (988 $\rightarrow$ 732 characters). Inspecting generated text directly, both arms remain coherent, on-topic, and substantively helpful; we do not observe truncation, repetition, or incoherence in either. We read this as most likely reflecting the anchor’s early-token influence shifting response *style* away from this dataset’s specific templated reference phrasing, rather than a genuine loss of

task competence, though we cannot fully rule out the latter without a human or LLM-judge quality evaluation, which we leave to future work. We report this number in full rather than let GSM8K/MedQA/IFEval stand as the whole utility picture: it is the one metric in this paper where ALIGNBEAM’s cost is not clearly small.

Y.1 MedQA Utility: Full Semantic Metrics

Table 23 gives the full per-strategy semantic-similarity metrics for the MedLlama pair on MedQA. All three LBM variants are reported ($n = 2,243$). Accuracy is answer-extraction accuracy over the ChatDoctor evaluation sample (§E); ROUGE-1/2, BERTScore-F1, and SentSim (all-MiniLM-L6-v2 cosine) are on ChatDoctor-style free-form generations vs. the reference answer. All semantic deltas are within ± 1.5 pp of baseline: safety blending does not measurably change content fidelity.

Strategy	<i>n</i>	Acc%	R-1	R-2	BS-F1	SentSim
Base	2243	13.1	0.2522	0.0319	0.7591	0.5961
LBM-DROP (default)	2243	12.7 (-0.4 pp)	0.2426 (-0.0096)	0.0311 (-0.0008)	0.7552 (-0.0039)	0.5813
LBM-EXACT	2243	12.7 (-0.4 pp)	0.2426 (-0.0096)	0.0311 (-0.0008)	0.7552 (-0.0039)	0.5813
LBM-FIRST	2243	12.7 (-0.4 pp)	0.2426 (-0.0096)	0.0311 (-0.0008)	0.7552 (-0.0039)	0.5813

Table 23: MedLlama MedQA utility breakdown across all LBM variants (JSL-MedLlama-3-8B-v2.0 + Qwen2.5-3B-Instruct, $\alpha = 0.5$, $K = 1$ utility mode, permissive anchor prompt). Δ columns are strategy minus baseline (pp). Acc. is answer-extraction accuracy over the ChatDoctor evaluation sample; semantic metrics are computed on ChatDoctor-style free-form generations vs. reference answer ($n = 2,243$).

Y.2 IFEval: Instruction-Following Under the Deployed $K = 3$ Configuration

GSM8K and MedQA measure task correctness but are largely insensitive to instruction-following degradation, and both are measured under $K = 1$ (the utility-deployment configuration) rather than the $K = 3$ configuration used for every safety number in this paper. To directly test whether early-token mixing reduces instruction-following accuracy *under the configuration actually deployed for safety*, we ran IFEval (Zhou et al., 2023) (541 prompts, rule-based verifiable-instruction checkers, no LLM judge) on Llama-8B (Llama-3.1-8B) under the full $K = 3$ safety configuration ($\alpha = 0.5$, $N = 6$, LBM-DROP, seed 42), base draft vs. ALIGNBEAM.

Pair	K	N	Base	ALIGNBEAM	Slowdown
Llama-8B: Llama-8B	3	6	3.3 s	16.9 s	5.07×
MedLlama: MedLlama-8B	3	6	2.6 s	13.7 s	5.27×
Finance-Llama: FinLlama-8B*	3	6	2.7 s	17.6 s	6.51×
DS-Math: DS-Math-7B-Inst	3	20	5.6 s	25.8 s	4.61×
Budget mode	1	6	$\sim 2\times$ (see Table 18)		

Table 25: ALIGNBEAM’s $K = 3$ default falls at 4.6–6.6×, inside the LlamaGuard / Proxy Tuning band. Per-benchmark safety at the $K = 1$ budget mode is reported in Table 18 rather than as a single aggregate here. Per-sample latency (RTX 6000 Pro 48 GB, bfloat16, HarmBench-Standard). Phase 3 (KV-cache continuation) $\approx 69\%$; Phase 2 $\approx 30\%$; Phase 1 $\approx 1\%$. *Finance-Llama per-sample wall-clock reconstructed from the $6.51\times$ recorded slowdown and TPS-derived per-sample timing of the sibling 8B base models.

IFEval (strict)	Base	ALIGNBEAM ($K = 3$)	Δ
Prompt-level accuracy	14.2%	27.0%	+12.8 pp
Instruction-level accuracy	24.6%	39.0%	+14.4 pp

Table 24: IFEval under the deployed $K = 3$ safety configuration (Llama-8B, $n = 541$). Instruction-following does not degrade under safety steering; it improves.

Instruction-following does not degrade under the safety configuration; it improves substantially. The mechanism is specific and we state it precisely rather than overclaim: the draft here is Llama-3.1-8B *base*, which follows instructions poorly, and the anchor is Qwen2.5-3B-*Instruct*, so mixing the anchor’s distribution into the opening tokens contributes instruction-following competence the base draft otherwise lacks. The defensible claim is therefore *early-token mixing does not reduce IFEval instruction-following accuracy, and on a base draft it measurably helps*, not that ALIGNBEAM improves instruction-following for every model; an instruct-tuned draft would likely show a smaller (possibly near-zero) delta, since it already has the competence the anchor would otherwise supply. This is not an artefact of degenerate or padded generations: ALIGNBEAM’s outputs are *shorter* than the base draft’s (median 361 vs. 624 characters) and empty-output rates are comparable (13/541 base vs. 7/541 ALIGNBEAM).

Z Latency Details

ALIGNBEAM’s $K = 3$ default falls inside the latency band already occupied by LlamaGuard and Proxy Tuning; the $K = 1$ budget mode is the cheapest non-classifier option and stays on par with the anchor-prefix baseline. Table 25 reports per-pair wall-clock on a single RTX 6000 Pro (48 GB, bfloat16).

Comparison with other inference-time defenses.

Table 26 places ALIGNBEAM’s two operating points next to the other inference-time interventions we evaluate in this paper and to published competitor numbers. The ALIGNBEAM $K = 1$ budget setting is the cheapest of any non-classifier option, and the $K = 3$ default falls inside the band already occupied by LlamaGuard prompt/response filtering, Proxy Tuning, and top- k contrastive decoding (all evaluated with the same $K = 3$ harness for comparability). The slower end of this band is a property of the inference-time defense *class*, not specific to ALIGNBEAM; the cheaper end is on par with the anchor-prefix baseline.

Method	Slowdown
Anchor prefix (Anchor-Prefix)	2.4–3.2×
ALIGNBEAM, $K = 1$ budget	$\sim 2\times$
RAIN [†]	3.78–4.36×
LlamaGuard-prompt (LG-Prompt), $K = 3$	4.1–6.0×
LlamaGuard-response (LG-Response), $K = 3$	3.9–5.5×
Top- k contrastive decoding (Qwen3-Base)	4.8–5.4×
Proxy Tuning (Qwen3-Base)	4.9–5.4×
ALIGNBEAM, $K = 3$ default	4.6–6.6×

Table 26: End-to-end per-sample slowdown of inference-time safety methods on Llama-8B / Qwen3-Base drafts (same harness, same GPU, mean over HarmBench-Standard and AdvBench unless noted). Competitor rows marked [†] are from the published paper; the rest are from our own harness so the slowdown definition is identical.

Z.1 Configuration Summary

Every experiment configuration (main pairs, ablations, and baselines) is in our repository (see footnote at first mention of ALIGNBEAM in §1).

Default hyperparameters. $\alpha = 0.5$ (mixing weight), $K = 3$ (beam count, safety mode) or $K = 1$ (utility mode), $N = 6$ (mixed steps; $N = 20$ for DS-Math), $\tau = 2.5$ (judge threshold), $T = 0.7$ (sampling temperature), $B = 50$ (text-bridge top- B width), $\text{max_new_tokens} = 150$ for safety / 512-1024 for utility, repetition penalty **1.15 for safety-mode runs and 1.0 for utility-mode runs** (GSM8K/MedQA/IFEval); 1.15 suppresses the repeated digits and operators DS-Math’s chain-of-thought needs and is not used for any utility number in this paper, seed 42. The depth sweep (Depth-Sweep) uses $\alpha = 0.85$ (held fixed) to isolate the depth effect; Alpha-Sweep sweeps $\alpha \in \{0.10, 0.25, 0.50, 0.75\}$ at $N = 6$.

Hardware and runtime. Single RTX 6000 Pro (48 GB) per pair, bfloat16 throughout, for all pairs

except Llama-70B. Llama-70B (Llama-3.1-70B draft) does not fit this card and instead ran on a single H200 (141 GB), bfloat16, no quantization (Appendix V). Per-sample wall-clock latency on HarmBench-Standard is reported in Appendix Z.

Software versions. transformers 4.47.1 (ALIGNBEAM decoding), torch 2.5.1, rouge_score 0.1.2, bert_score 0.3.13 (roberta-large backbone), sentence-transformers 3.3.1 (all-MiniLM-L6-v2 for SentSim).

Anchor. Qwen2.5-3B-Instruct is the safety anchor for every main run (the five main pairs and the 70B-scale Llama-70B). The anchor ablations use other anchors: Qwen2.5-0.5B-Base (Appendix P), Llama-3.1-8B-Instruct (Appendix R), and Qwen2.5-7B-Instruct and Qwen2.5-0.5B-Instruct (Appendix X).

AA Additional Robustness Analyses

AA.1 WildJailbreak Depth Check ($N = 6$ vs. $N = 20$)

To test directly whether the WildJailbreak gap (Limitations) is a depth limitation, we re-ran Llama-8B WildJailbreak-Harmful ($n = 1,315$) at $N = 20$, more than triple the paper’s default $N = 6$ and the same depth already used for DS-Math’s math pair, with everything else fixed ($\alpha = 0.5$, $K = 3$, LBM-DROP, seed 42).

N	String-match Ref%	WildGuard judge Acc%
Base (no mixing)	13.3	43.7 ^{‡,§}
6 (paper default)	19.7	43.9 [‡]
20	21.3	45.4 [‡]

Table 27: Tripling the mixed window from $N = 6$ to $N = 20$ recovers only +1.6 pp string-match / +1.5 pp judge accuracy on Llama-8B WildJailbreak; depth is not the binding constraint on this benchmark. [‡]Judge-Acc% here is corrected for the two scoring defects described in Appendix C. The $N = 6 \rightarrow 20$ judge delta this table’s claim rests on is *unchanged* by the correction (+1.5 pp either way), as is every string-match value. [§]This row’s base value (43.7) comes from a separate WildGuard scoring pass than the 43.4 base value reported for the same Llama-8B/WildJailbreak condition in Table 4; the two runs agree to within 0.3 pp and neither this table’s claim nor any conclusion in the paper depends on which is used.

We hypothesise that depth is not the lever here: WildJailbreak’s attacks are adversarially framed to look benign, so when the off-the-shelf anchor does not recognise the harmful intent beneath that framing, no refusal opener is produced for Phase 2 mixing to extend; more mixed steps of an already-compliant opening do not retroactively make it a

refusal. The binding constraint is therefore upstream of depth: whether the anchor *recognises* the adversarial framing at all. A better-calibrated anchor, consistent with the anchor-size finding that calibration, not size, sets the ceiling (Appendix X), is the more promising lever than a longer window, and is left to future work rather than requiring any change to the method itself.

AA.2 Bootstrap Confidence Intervals on Headline Refusal

To confirm the headline gains are not an artefact of which prompts happen to be sampled, we compute nonparametric bootstrap 95% confidence intervals on string-match refusal for Llama-8B: from the n graded prompts we draw n with replacement, recompute the refusal rate, repeat 10,000 times, and take the 2.5th/97.5th percentiles.

Benchmark (n)	Base [95% CI]	ALIGNBEAM [95% CI]
AdvBench (520)	17.5 [14.2, 21.0]	94.4 [92.3, 96.3]
HarmBench-Std (167)	15.0 [9.6, 20.4]	79.6 [73.7, 85.6]
HarmBench-Ctx (62)	8.1 [1.6, 14.5]	79.0 [67.7, 88.7]
Sorry-Bench (440)	10.9 [8.2, 13.9]	61.4 [56.8, 65.9]

Table 28: Bootstrap 95% CIs (Llama-8B, string-match refusal, 10,000 resamples, seed 42). Base and ALIGNBEAM intervals are disjoint on every benchmark; HarmBench-Ctx is widest, consistent with its smaller $n = 62$ (matches the seed-robustness spread in Appendix T).

Every interval is disjoint between base and ALIGNBEAM, so the gains reported in §4 hold well beyond sampling noise on this pair. We note a design limitation: these are separate bootstrap intervals on the base and ALIGNBEAM refusal rates, not a paired bootstrap over the per-prompt difference. A paired bootstrap would more directly support the improvement claim by accounting for prompt-level correlation between the two conditions; disjoint separate intervals are a weaker but still informative signal, and we report them as such rather than as a substitute for the paired test.

AA.3 Refuse-Then-Continue Rate

Appendix E.1 notes that some unsteered responses emit a canonical refusal phrase and then continue with harmful content anyway (Ref% > ACC% for that response), and that Phase-2 mixing should suppress this pattern by anchoring the *continuation*, not just the opening, to the anchor’s distribution. We measure this directly on Llama-8B rather than assert it: the rate of responses that both match a refusal phrase and are still judged harmful.

Benchmark	Base	ALIGNBEAM
AdvBench	5.8%	0.2%
HarmBench-Std	6.0%	1.8%
Sorry-Bench	2.7%	2.0%
HarmBench-Ctx	3.2%	8.1%

Table 29: Rate of refuse-then-continue responses (canonical refusal phrase present, but judged harmful), Llama-8B. The HarmBench-Ctx uptick corresponds to three additional cases, from 2/62 to 5/62; we do not report a significance test for this count and do not claim the increase is or is not distinguishable from noise at this n .

Phase-2 mixing largely eliminates refuse-then-continue on the structured sets (AdvBench, HarmBench-Std, Sorry-Bench); the HarmBench-Contextual uptick is small-sample (5/62) and we flag it rather than average it away.

AB Extended Limitations

The Limitations section states the two safety limits and the evaluation-scope constraints at a high level; this appendix gives the full numbers and mechanism behind each.

Calibration ceiling, in detail. The anchor’s tendency to over-refuse ambiguous benign prompts is inherited rather than induced by mixing: OR-Bench-Hard over-refusal stays $\approx 20\%$ across the entire α range (Fig. 2(b)) and XSTest over-refusal $\approx 40\%$ on Llama-8B; neither improves with more mixing. Separately, because the mixing only covers the first N tokens, attacks that embed harm past the prefix are only partially blocked (WildJailbreak-Harmful: +2–6 pp on Llama-8B/MedLlama vs. +51–77 pp on AdvBench). Widening the mixed window does not close this gap: on Llama-8B we re-ran WildJailbreak at $N = 20$ ($3.3\times$ the default) and gained only +1.6 pp string-match refusal over $N = 6$ (Appendix AA.1). Our hypothesis is that the bottleneck is upstream of depth: the off-the-shelf anchor does not always recognise the harmful intent beneath benign-looking framing, so no refusal opener is produced for the mixing to carry forward. On Finance-Llama3-8B specifically, ALIGNBEAM *lowers* WildJailbreak refusal (42.7%→33.2% string-match; corrected judge ACC 58.6%→54.9%), the same mechanism by which anchor mixing can disrupt an already-conservative model’s own contextual handling of multi-sentence prompts (also seen on the instruct control, Instruct-Base, Appendix K). The calibration ceiling could plausibly be improved with a better-calibrated anchor or a per-prompt α schedule; a longer mixed prefix does not help the Wild-

Jailbreak depth limit specifically ($N = 20$ above), though it may still help tasks such as DS-Math where the bottleneck is not upstream recognition.

Evaluation scope, in detail. We do not evaluate adaptive attacks designed to circumvent ALIGNBEAM specifically, though the partial WildJailbreak result (+2–6 pp string-match) gives a rough lower bound on that vulnerability. Multi-anchor ensembles, multilingual evaluation, and human judging are natural next steps.

AC Extended Ethical Considerations

ALIGNBEAM is designed and evaluated as a *defensive* method, and the large majority of reported results measure an increase in refusal on harmful prompts. This is not universal, however: the corrected results show mixing can also *degrade* an already safer draft on delayed-payload prompts. On WildJailbreak specifically, Finance-Llama3-8B’s string-match refusal falls (42.7% → 33.2%) and corrected judge accuracy declines on multiple pairs (DS-Math, MedLlama, Finance-Llama, Qwen3-Base), and the instruct control (Instruct-Base) shows the same pattern (§AB). We therefore do not claim the mixing operation can only pull a draft’s distribution toward the anchor’s safer one; on prompts where the anchor itself fails to recognise a harmful request as harmful (the WildJailbreak depth limit above), mixing can leave a model no safer, or measurably less safe by these metrics, than the unmodified draft.

Dual-use consideration. The cross-vocabulary bridge (§2) is a general mechanism for transferring a guide model’s token-level preferences into a different model’s vocabulary. In principle the same bridge could transfer preferences other than safety refusal. We are not aware of a way to repurpose it for a more harmful objective that would be easier to achieve than directly fine-tuning or prompting the draft model, since the bridge only shapes the first $N \approx 6$ tokens and hands off control to the unmodified draft immediately after (§5).

Residual risks, in detail. Two limitations discussed in the main text have safety implications and are restated here for completeness. First, the calibration ceiling inherited from the anchor means genuine benign compliance falls under ALIGNBEAM (e.g. OR-Bench-Hard 77.2%→72.5% on Llama-8B; Appendix C), a deployer relying on the method for user-facing benign traffic should budget for this over-refusal cost, not treat it as neg-

ligible. Second, the method’s protection is partial against attacks that place harmful content past the mixed prefix (WildJailbreak; §5); it should not be presented to end users or auditors as a complete jailbreak defense, and we do not do so anywhere in this paper.

Data and compute. All eleven harmful/benign/calibration benchmarks in Table 5 are public safety-evaluation datasets; the MedSafetyFix split is drawn from MedSafetyBench (Han et al., 2024b), with its provenance and paper-specific naming detailed in Appendix E.1. No harmful model outputs are reproduced anywhere in the paper; only aggregate statistics are reported, consistent with responsible disclosure practice for jailbreak-adjacent research. All evaluation is automated and English-only. Compute usage is disclosed in Appendix AD (C1).

Intended use. We position ALIGNBEAM as a research artifact for the alignment and safety community, intended to be evaluated, extended, and stress-tested by other researchers, not as a production-ready, certified safety guarantee. Any deployment decision should be based on evaluation against the deployer’s own threat model, not on the benchmark numbers in this paper alone.

AD Responsible NLP Research Checklist

This checklist answers the ARR OpenReview responsible-research form for this submission.

A1 : Limitations section. **Yes.** A Limitations section appears after §5 and covers: (i) compute overhead and latency operating points, (ii) the calibration ceiling inherited from the anchor (XSTest over-refusal $\approx 40\%$; WildJailbreak +2–6 pp bottleneck, now further characterised by the $N = 20$ depth check in Appendix AA.1), (iii) evaluation scope (single-turn, ≤ 200 tokens, no adaptive attacks, single anchor model).

A2 : Potential risks. **Yes.** See the Ethics Statement (main text) and Appendix AC for the full discussion. ALIGNBEAM raises rather than lowers refusal rates, limiting direct misuse. Residual risks: calibration ceiling on adversarial-looking benign prompts (OR-Bench-Hard genuine compliance falls to 72.5%; Appendix E.1) and partial mitigation of multi-sentence prefix-length attacks (WildJailbreak +2–6 pp). The method is intended for safety improvement in research settings only.

B1 : Citation of artifact creators. **Yes.** All benchmarks, model families, and baselines are cited: HarmBench (Mazeika et al., 2024), AdvBench (Zou et al., 2023), Sorry-Bench (Xie et al., 2025), WildJailbreak (Jiang et al., 2024), XSTest (Röttger et al., 2024), OR-Bench (Cui et al., 2025), JailbreakBench (Chao et al., 2024), LlamaGuard (Inan et al., 2023), SafeDecoding (Xu et al., 2024), GSM8K (Cobbe et al., 2021), ChatDoctor-HealthCareMagic (Li et al., 2023b), IFEval (Zhou et al., 2023). Anchor and draft model families cited via their technical reports.

B2 : Licenses for artifacts used. **Yes.** Benchmarks are used for evaluation consistent with their research licences. Models used under their respective community licences (Meta Llama 3 Community Licence, Qwen Community Licence). All use is non-commercial research. Exact SPDX identifiers are not separately catalogued; licence terms are as stated above for each model/benchmark family.

B3 : Use consistent with intended use. **Yes.** All benchmarks are used for safety evaluation as intended. ALIGNBEAM does not modify or redistribute model weights.

B4 : Personally identifying information or offensive content. **Yes (managed).** Safety benchmarks contain adversarial harmful prompts by design. No personally identifying information is used. Harmful prompts appear only as aggregate statistics; no harmful model outputs are reproduced in the paper.

B5 : Documentation of released artifacts. **Yes.** Harness configuration layout is documented in Appendix Z.1. Per-benchmark judge details are in Appendix E. The code repository (footnote at first mention of ALIGNBEAM in §1) documents setup and reproduction.

B6 : Statistics for data. **Yes.** Dataset sizes are in Table 5 (Appendix E). All main runs use seed 42; a three-seed robustness sweep (± 0.7 pp HB+AB) is in Appendix T.

C1 : Model sizes and compute budget. **Yes.** Draft models: 6.7B–70B parameters (see §3 and Appendix V). Anchor: Qwen2.5-3B-Instruct (3B). Judges: HarmBench-Mistral-7b-val-cls (7B), Sorry-Bench judge (7B), LlamaGuard-3-8B (8B), WildGuard (7B). Hardware: single RTX 6000 Pro

(48 GB, bfloat16) per pair for all experiments except the 70B-scale run; Llama-70B (Llama-3.1-70B) ran on a single H200 (141 GB, bfloat16, no quantization or offloading), since the 70B draft does not fit the 48 GB card (Appendix Z.1, Appendix V). Per-sample wall-clock: 13.7–35.6 s/sample for the 8B-scale pairs (Table 25); Llama-70B is reported separately in Appendix V. Total GPU-hours: $\approx 2,054$ on the RTX 6000 Pro (summed from the per-sample generation timings recorded in every 8B-scale experiment, ablation, and baseline reported in the paper’s released configuration tree; excludes judge/evaluation compute, which is smaller), plus ≈ 37 H200-hours for Llama-70B (summed the same way, strategy and baseline wall-clock across its seven benchmarks).

C2 : Experimental setup and hyperparameters.

Yes. All hyperparameters are in Appendix Z.1: $\alpha = 0.5$, $K = 3/1$, $N = 6$ (DS-Math: 20), $\tau = 2.5$, $T = 0.7$, $B = 50$, repetition penalty 1.15 (safety mode) / 1.0 (utility mode), seed 42. Software versions: transformers 4.47.1 (ALIGNBEAM decoding), torch 2.5.1, rouge_score 0.1.2, bert_score 0.3.13, sentence-transformers 3.3.1.

C3 : Descriptive statistics and error bars. Partial.

Results are from single generation runs per configuration, but robustness is supported two ways: a three-seed sweep (Appendix T, ± 0.7 pp HB+AB on Llama-8B) and nonparametric bootstrap 95% confidence intervals on the Llama-8B headline refusal numbers (Appendix AA.2), which are disjoint between base and ALIGNBEAM on every benchmark tested. Extending bootstrap CIs to every pair and benchmark remains future work.

C4 : Parameters for packages used. Yes.

Package versions are listed in the *Software versions* paragraph of Appendix Z.1.

D1–D4 : Human subjects. Not applicable.

No human annotators or research subjects. All evaluation is automated using publicly released, benchmark-compatible classifiers.

E1 : AI assistants in research or writing. Yes.

AI coding assistants (Claude Code) were used for LaTeX formatting and prose polish. All scientific content, experimental design, numerical results, analysis, and conclusions are the authors’ own work.