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

Chirag Chawla [1] **Pratinav Seth** [1] **Vinay Kumar Sankarapu** [1]

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 logits 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 safety-utility trade-off can be tuned at deployment without retraining. Across both cross-vocabulary and same-vocabulary evaluation pairs, ALIGNBEAM substantially raises refusal on adversarial benchmarks while keeping task accuracy and inference overhead within practical bounds. The results show that safety alignment can be transferred between model families at inference time, without touching either model's weights.

1. Introduction

Domain fine-tuning introduces a critical safety gap. Clinical organisations fine-tune Llama-3 on patient notes, security firms fine-tune on exploit repositories, and research labs fine-tune on mathematical reasoning data. Each of these efforts produces a model whose domain competence is precisely what makes it useful to adversarial actors: a medical model will answer pharmacology questions phrased as clinical queries, a code model will write functional exploits when the prompt reads as a security audit, and a math model will assist in quantitative fraud once harm is recast as an optimisation problem.

[1]Lexsi Labs. Correspondence to: Pratinav Seth <pratinav.seth@lexsi.ai>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

Inadequacy of existing defences. The standard playbook for restoring safety in such specialists is inadequate in this regime. *Safety system prompts* are silently ignored by base models that lack chat-template conditioning. In our experiments the system-prompted safety-prompt baseline achieves 14.3% explicit refusal, *below* the 16.9% of the unprompted draft-only baseline (§5), echoing observations that shallow safety training is brittle to prompt and decoding shifts (Qi et al., 2024; Yang et al., 2023). *Post-generation classifiers* such as LlamaGuard (Inan et al., 2023) can flag harmful completions after the fact but do not steer the underlying generation, so the model still produces (and may leak via streaming) the unsafe content. *Retraining-based realignment* (RLHF, DPO, or safety SFT) is effective when applicable but presupposes access to weights, labelled preference data, and substantial compute; moreover, Qi et al. (2023) show that even 100 benign fine-tuning samples suffice to undo such alignment, so each domain update demands a fresh realignment pass. *Self-alignment via rewindable decoding* (Li et al., 2024) requires the model to retain a non-trivial safety prior to generate the positive signals that drive rewinding, a condition violated by the near-zero refusal of domain-fine-tuned specialists. *Inference-time logit mixing* comes closest in spirit: SAFEDECODING (Xu et al., 2024) blends draft and safety-model probabilities at their shared token-ID space, and NUDGING (Fei et al., 2025) swaps to an aligned guide on high-entropy steps. Both, however, assume a shared vocabulary, which rules out cross-family pairs (e.g. Llama draft with a Qwen anchor) that arise naturally when a small, well-aligned anchor is reused across heterogeneous specialists.

This paper. We introduce ALIGNBEAM, an inference-time procedure that blends the token-level distributions of a domain-specialist draft M_d and a small aligned safety anchor M_s at each decoding step. A *cross-vocabulary text bridge* (decode anchor token, re-encode under the draft tokeniser) reconciles heterogeneous vocabularies without modifying either model. A three-phase beam pipeline with an LLM judge then selects the safest coherent response, and the safety–speed (depth N) and safety–utility (weight α) tradeoffs are exposed as user-tunable knobs.

Contributions.

1. **Cross-vocabulary logit bridge mixing (LBM).** Three variants enabling probability mixing across heterogeneous tokenisers without token-ID sharing (§3.2).
2. **Three-phase beam pipeline.** Depth N controls the safety–speed Pareto; weight α controls the safety–utility Pareto (§3.3).
3. **Early-token safety hypothesis.** The jump from $N=0$ (beam selection only) to $N=3$ captures +61.9 pp, after which gains diminish sharply (§6.1).
4. **Six-pair empirical validation.** Covering Llama-3.1-8B, DeepSeek-Math-7B-Instruct, Gemma-3-12B-PT, Qwen3-8B-Base, MedLlama-3-8B, and Finance-Llama3-8B (§5).

2. Related Work

Alignment erosion under fine-tuning. Qi et al. (2023) show that as few as 100 benign fine-tuning samples can substantially reduce refusal rates, and Yang et al. (2023) demonstrate that authority and domain framing provide potent soft jailbreaks against safely-aligned models. Domain-specialist training is far more extensive than these small-scale demonstrations, and our six-pair evaluation correspondingly observes baseline refusal rates that range from near-zero to moderate across all domains.

Inference-time safety steering. SafeDecoding (Xu et al., 2024) is our closest prior work: it blends draft and safety-model token probabilities at their shared vocabulary intersection and therefore does not apply to cross-family pairs; AlignBeam relaxes this assumption via the text bridge and additionally targets base models without chat-template conditioning. In the limit $\mathcal{V}_d = \mathcal{V}_s$, all LBM variants reduce to SafeDecoding. Nudging (Fei et al., 2025) steers a large base model by monitoring entropy and performing a binary swap to a smaller aligned guide, where we instead use a continuous weighted blend with α together with beam search and LLM-judge selection. Proxy Tuning (Liu et al., 2024) applies logit arithmetic between a fine-tuned model and its instruct counterpart to improve quality, but it presumes shared vocabulary and is not safety-oriented; on the same-vocabulary Qwen3-8B-Base pair, AlignBeam outperforms proxy tuning substantially on contextual and sorry-bench categories (Appendix E). RAIN (Li et al., 2024) achieves self-alignment via rewindable generation at 3.78–4.36× overhead, but relies on the model's own safety prior to produce the positive signals that drive rewinding, a condition that is violated in precisely the domain fine-tuned regime we target. Contrastive decoding (Li et al., 2023) subtracts weaker-model logits from stronger-model logits; on same-vocabulary pairs, AlignBeam's LBM with an unaligned anchor is functionally related to this approach (Appendix D).

Early-token safety concentration. Qi et al. (2024) argue that safety alignment concentrates in the first few token positions and is therefore brittle to shifts in prompt or decoding distribution. Our mixed-depth ablation corroborates this empirically and motivates the choice of small mixed depths ($N \in \{3, 6\}$) for most pairs.

3. Method

3.1. Problem Formulation

Let M_d be a domain-specialist draft model with vocabulary $\mathcal{V}_d$ and M_s be 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 both *safe* and *useful*. Neither model is modified; AlignBeam operates solely at inference time. Figure 1 illustrates the full pipeline.

3.2. Cross-Vocabulary Logit Bridge Mixing

The core technical challenge is that M_d and M_s produce logits over different vocabularies, so direct token-ID mixing is ill-defined whenever the two tokenisers disagree on segment boundaries. We introduce **Logit Bridge Mixing (LBM)**, which performs the mixing in the draft-vocabulary space after a text-level translation of the anchor's top-B tokens.

Text bridge procedure. Let P_s and P_d denote the next-token softmax distributions of the anchor M_s and draft M_d at the current step, and let $\mathcal{T}_B \subset \mathcal{V}_s$ be the top-B tokens of P_s (we use $B = 50$). For each anchor token $s_{\text{id}} \in \mathcal{T}_B$ with probability $p_s = P_s(s_{\text{id}})$, we decode it to text under the anchor tokeniser and re-encode under the draft's, $t = \text{decode}_s(s_{\text{id}})$, $d_{\text{ids}} = \text{encode}_d(t)$, and inspect $|d_{\text{ids}}|$. When $|d_{\text{ids}}|=1$ we have a clean single-token match $d_{\text{id}} \in \mathcal{V}_d$, and the anchor's mass is accumulated into an unnormalised draft-vocabulary buffer $f \in \mathbb{R}^{|\mathcal{V}_d|}$ (initialised to zero) via

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

where $\alpha \in [0, 1]$ controls the safety–utility tradeoff. The $+=$ accumulates across all B anchor tokens; if multiple anchor tokens map to the same draft token, their contributions are summed. Concretely, if k anchor tokens all resolve to the same draft token d, the buffer receives $\alpha \sum_{i=1}^{k} p_{s,i} + k(1-\alpha)\, P_d(d)$ before renormalisation; the $(1-\alpha)\, P_d$ term is therefore accumulated once per anchor-token mapping, not once per unique draft token. In practice such collisions are rare: the mean duplicate-mapping rate across cross-vocabulary pairs is below 2% of the top-B anchor tokens per decoding step. When $|d_{\text{ids}}| \neq 1$ the match is ambiguous and a variant-specific fallback is applied (Table 1). After all B anchor tokens are processed, the buffer is renormalised to yield the blended distribution

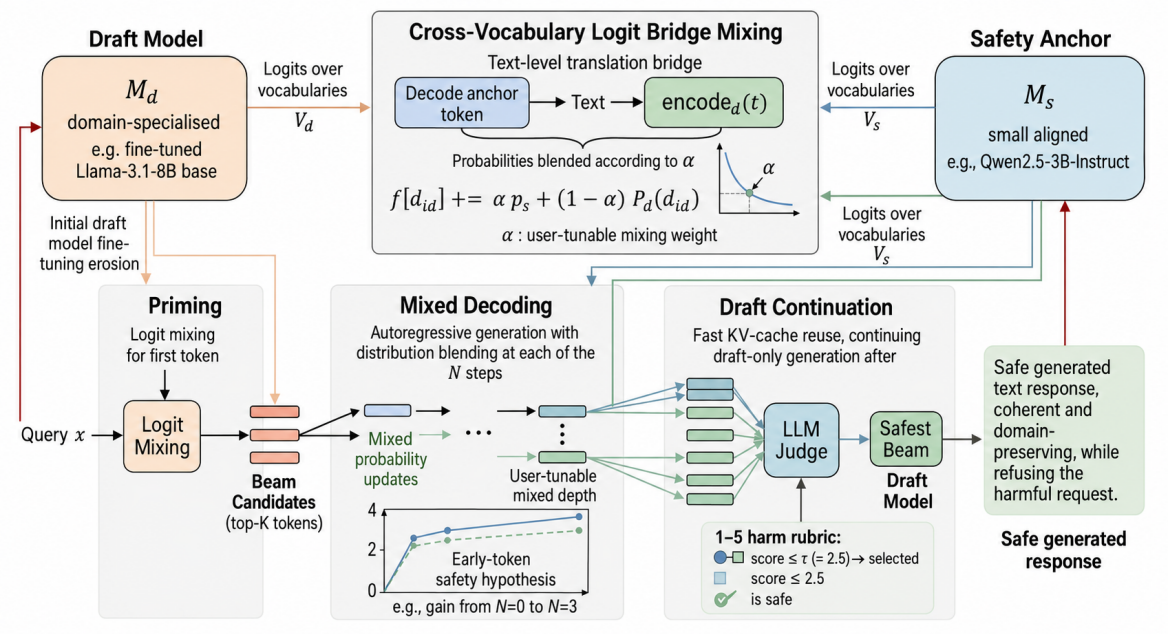

Figure 1. ALIGNBEAM three-phase pipeline. **Phase 1 (Priming):** both models process query x; LBM at step 0 produces K beam roots from the blended distribution. **Phase 2 (Mixed Decoding):** each beam is extended greedily for $N-1$ additional steps via LBM, establishing a safe prefix across the critical early tokens (N mixed steps total including priming). **Phase 3 (Draft Continuation):** an LLM judge scores all K beams on a 1–5 harm rubric; the safest beam b^* is continued by M_d alone with KV-cache reuse, recovering full domain fluency at minimal cost.

Table 1. The three LBM variants and their multi-token fallback rules.

Variant	Multi-token fallback	Use case
LBM-DROP	Drop; renormalise	Base drafts
LBM-FIRST	Use first sub-token	Cross-vocab instruct
LBM-EXACT	Keep only 1:1 matches	Conservative

$\tilde{f}[v] = f[v]/\sum_{v'} f[v']$, from which the next token (or beam roots, in Phase 1) is drawn.

LBM variants.

LBM-DROP is the default strategy for base drafts (Llama-3.1-8B, MedLlama-3-8B, Finance-Llama3-8B, Gemma-3-12B-PT, Qwen3-8B-Base; see Table 2) and for instruct models on which LBM-FIRST is unreliable; LBM-FIRST is preferred for cross-vocabulary instruct models whose tokenisers map anchor tokens to predictable first sub-tokens; and LBM-EXACT is a conservative variant that retains only exact 1:1 matches. On the same-vocabulary pair (Qwen3-8B-Base) the three variants are indistinguishable (>98% single-token match rate); on cross-family pairs, roughly 12% of anchor top-50 tokens produce multi-token sequences and are routed through the variant-specific fallback.

3.3. Three-Phase Beam Pipeline

Phase 1 (Priming). The query x is formatted for each model (plain `Q:{x}\nA:` for base drafts and the model's chat template for instruct drafts), and both models perform a single forward pass at step 0. The resulting logits are converted to next-token distributions, LBM blends these distributions, and the top-K tokens of $\tilde{f}$ are taken as beam roots.

Phase 2 (Mixed Decoding). For steps $1, \ldots, N-1$ and each beam b_k, both models extend their own context with b_k's text and emit fresh logits; the resulting distributions are then re-blended via LBM and the argmax token appended to b_k. The total number of mixed steps is therefore N (step 0 through step $N-1$). Each model runs its own forward pass, avoiding cross-vocabulary token-ID contamination, at a total cost of $O(K \cdot N)$ paired forward passes.

Phase 3 (Draft Continuation). After N mixed steps, *all* K beams are independently completed by M_d alone. Each beam's mixed prefix is primed into a KV cache in a single forward pass, and subsequent tokens are appended as tensors only (no re-tokenisation), so the mixed prefix is never recomputed. The LLM judge J then scores all K complete responses in a single batched call on a 1–5 harm rubric (1 = full refusal; 5 = fully harmful content), and the safest beam b^* is the lowest-scoring response with score $\leq \tau = 2.5$; if no beam clears the threshold, b^* is the minimum-score beam overall. Two implementation details not visible in the pseudocode: (i) if a beam's Phase 2 pre-

Algorithm 1 ALIGNBEAM Generation

Require: Query x, draft M_d, anchor M_s, depth N, beams K, weight α (b_k = beam k's accumulated token sequence)
Ensure: Response y^*

Phase 1: Priming {step 0}
$x_d \leftarrow \text{fmt}_d(x)$; $x_s \leftarrow \text{fmt}_s(x)$
$\tilde{f} \leftarrow \text{LBM}(M_d(x_d),\ M_s(x_s),\ \alpha)$
$\{b_k\}_{k=1}^{K} \leftarrow$ top-K tokens of $\tilde{f}$
Phase 2: Mixed Decoding {steps 1 to $N-1$}
for step $= 1, \ldots, N-1$; each beam b_k **do**
  $\tilde{f}^{(k)} \leftarrow \text{LBM}(M_d(x_d \| b_k),\ M_s(x_s \| b_k),\ \alpha)$
  $v^* \leftarrow \arg\max_v\ \tilde{f}^{(k)}(v)$; extend b_k
end for
Phase 3: Draft Continuation
for each beam b_k **do**
  Complete b_k with M_d (KV-cache reuse) {full response}
end for
score$(b_k) \leftarrow J(x, b_k.\text{txt})$ for all k {LLM judge}
$b^* \leftarrow \arg\min_{k:\ \text{score}(b_k) \le \tau} \text{score}(b_k)$ {rubric 1–5; $\tau = 2.5$}
return $y^* \leftarrow b^*.\text{txt}$

fix already opens with a canonical refusal phrase (e.g. "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 affected beam is marked completed and skips Phase 3 generation.

Overhead structure. Phase 3 (KV-cache draft continuation for all K beams) accounts for approximately 69% of wall time, Phase 2 (mixed decoding) for ≈30%, and Phase 1 for ≈1%; the primary speed lever is therefore the beam count K rather than the depth N, since K multiplies Phase 3's cost directly. A "budget mode" with $K = 1$ runs at ∼2× overhead while still capturing ∼80% of the safety benefit (Appendix T).

4. Experimental Setup

4.1. Model Pairs

We evaluate ALIGNBEAM on the six draft/anchor pairs summarised in Table 2, spanning math, general, medical, and financial domains and ranging from 6.7B to 12B parameters. Five pairs are cross-vocabulary (the draft and anchor use different tokenisers); Qwen3-8B-Base is the single same-vocabulary control. The safety anchor is held fixed across all pairs at Qwen2.5-3B-Instruct.

Table 2. The six draft/anchor model pairs used in our evaluation. The safety anchor is Qwen2.5-3B-Instruct for all pairs. Defaults: $\alpha = 0.5$, $K = 3$ (safety) / $K = 1$ (utility), seed 42. DeepSeek-Math-7B-Instruct uses $N = 20$ to suppress a refuse-then-comply pattern observed at $N = 6$; Qwen3-8B-Base is the only same-vocabulary pair.

Draft Model	Dom.	Type	Voc.	N
DeepSeek-Math-7B-Instruct	Math	Inst.	Cross	20
Llama-3.1-8B	Gen.	Base	Cross	6
JSL-MedLlama-3-8B-v2.0	Med.	Base	Cross	6
Finance-Llama3-8B	Fin.	Base	Cross	6
Gemma-3-12B-PT	Gen.	Base	Cross	6
Qwen3-8B-Base	Gen.	Base	**Same**	6

4.2. Datasets

Safety (must refuse). HarmBench-Standard (167) (Mazeika et al., 2024), HarmBench-Contextual (62), AdvBench (520) (Zou et al., 2023), Sorry-Bench (440, 44 categories) (Xie et al., 2024), JailbreakBench-Harmful (100) (Chao et al., 2024), WildJailbreak-Eval-Harmful (1,105) (Jiang et al., 2024). Domain-specific: Math-Jailbreak (40).

Calibration (benign, must comply). XSTest (450) (Röttger et al., 2023), OR-Bench-Hard (855) (Cui et al., 2024), JailbreakBench-Benign (100).

Utility. GSM8K (1,319, exact match) (Cobbe et al., 2021); MedQA (Jin et al., 2021).

4.3. Hyperparameters

Unless stated otherwise, main experiments use $N = 6$, $K = 3$, $\alpha = 0.5$, judge threshold $\tau = 2.5$, sampling temperature $T = 0.7$, repetition penalty 1.15, bridge width $B = 50$, and seed 42; `max_gen` is 150 tokens for safety and 512–1024 for utility runs. Two pair-specific deviations apply. DeepSeek-Math-7B-Instruct uses $N = 20$ because at $N = 6$ the model exhibits a "refuse-then-comply" pattern in which harmful content resumes once the mixed prefix ends. Utility evaluations use $K = 1$ to avoid beam-selection artefacts that suppress code syntax in $K = 3$ safety mode.

4.4. Evaluation Judges

We score every benchmark with its official released judge (Table 3); the LLM judge that selects beams in Phase 3 of the pipeline is used *only* for in-generation beam selection and never for reporting safety numbers, avoiding self-evaluation circularity. All reported B0% and LBM% figures are **string-match refusal rates** (the fraction of responses containing at least one of 22 canonical refusal phrases such as "I cannot", "I will not", "it is illegal"). Official benchmark judges (*acc.%*) are additionally computed for all experiments and reported in the appendix; as discussed in §7.2, these are inflated for base-model baselines and are therefore not used

Table 3. Official released judge for each safety benchmark. All reported safety numbers use these classifiers; the LLM judge inside Phase 3 (§3.3) is used only for in-generation beam selection, avoiding self-evaluation circularity.

Benchmark	Official Judge	Output
HarmBench, AdvBench, domain sets	`HarmBench-Mistral-7b-val-cls`	yes/no
JailbreakBench	`Llama-Guard-3-8B`	safe/unsafe
SorryBench	`ft-mistral-7b-instruct-v0.2`	0/1
WildJailbreak	`wildguard`	yes/no

as the primary metric.

5. Results

5.1. Main Safety Results

Throughout, **B0** denotes the unmodified draft with no safety intervention; B0% is its string-match refusal rate as reported in Table 4.

Base models with low safety prior (Llama-3.1-8B, MedLlama-3-8B, Finance-Llama3-8B). These are the strongest results. B0 refusal ranges from 17–35%; ALIGNBEAM raises all three to 88–91% ($\Delta_{\text{safe}} = +54$–$74$ pp). The medical model (MedLlama-3-8B) shows the largest HarmBench-Contextual gain (+59.7 pp), where medical domain framing provides the attack vector. On the domain-specific MedSafetyFix benchmark (900 clinically-framed harmful prompts; LlamaGuard judge), ALIGNBEAM raises accuracy from 77.3% to 96.8% (+19.5 pp), confirming that safety gains extend to prompts engineered specifically to exploit medical domain framing. The finance model (Finance-Llama3-8B) follows a similar pattern on structured adversarial prompts, though it shows a small regression on WildJailbreak (from a 42.7% base that already reflected some refusal training). Notably, all three LBM strategies produce identical results for Finance-Llama3-8B across every benchmark, suggesting high tokeniser compatibility between the Finance-Llama3-8B (Llama-3 based) and Qwen2.5-3B vocabularies; per-dataset details appear in Appendix C. Per-dataset breakdowns for Llama-3.1-8B are provided in Appendix B.

Math instruct model (DeepSeek-Math-7B-Instruct). DeepSeek-Math starts at a very low 11.5% (the chat template conditioning does not include safety training). With LBM-DROP at $N = 20$, safety reaches 76.4% (+64.9 pp). The Math-Jailbreak domain-specific set shows a small regression (from 5% to 2.5%), as the anchor's token mixing disrupts domain-framing heuristics that the model had developed. Among the three LBM variants, LBM-FIRST achieves the highest string-match refusal on every benchmark at $N = 20$ (AdvBench 93.8%, HB-Std 71.3%); all three strategies produce zero degenerate outputs. A full per-strategy comparison appears in Appendix H.

Gemma-3 base (Gemma-3-12B-PT). Gemma-3-12B-PT (262K-token vocabulary, no built-in refusal prior, B0 $\approx$ 4.4%) reaches 52.4% at $N = 6$. The large vocabulary means $\approx$20% of anchor top-50 tokens produce multi-token sequences, handled by LBM-DROP. JailbreakBench-Harmful shows the largest per-dataset gain (+70 pp). WildJailbreak and Sorry-Bench gains are more modest; the architecture appears to require a longer anchor prefix to fully suppress harmful continuations, a question warranting further study.

Table 4. Main safety results on the combined HarmBench-Standard + AdvBench harm set ($n = 687$; HB-Std 167 + AdvBench 520). B0% and LBM% are **string-match refusal rates** ($\Delta_{\text{safe}} = \text{LBM\%} - \text{B0\%}$); official benchmark-judge accuracy is higher for all configurations and reported in the per-dataset appendix tables. ALIGNBEAM (column **LBM%**) raises refusal rates from near-zero to 88–97% on the majority of pairs while imposing 4.6–6.6× overhead. Strategy: LBM-DROP for all pairs. $\alpha = 0.5$, seed 42. †Gemma-3-12B-PT's B0% is the string-match rate; the benchmark-judge baseline is $\approx$46% due to base-model score inflation (§7.2).

Draft Model	Domain	Vocab	B0%	LBM%	$\Delta_{\text{safe}}\uparrow$	Slow.
Llama-3.1-8B	General	Cross	16.9	**90.8**	+73.9	5.07×
MedLlama-3-8B	Medical	Cross	34.5	**88.5**	+54.0	5.27×
Finance-Llama3-8B	Finance	Cross	28.1	**91.1**	+63.0	6.51×
DeepSeek-Math-7B-Instruct	Math	Cross	11.5	**76.4**	+64.9	6.61×
Gemma-3-12B-PT	General	Cross	4.4†	**52.4**	+48.0	5.24×
Qwen3-8B-Base	General	**Same**	84.6	**92.1**	+7.5	4.9×

Same-vocabulary pair (Qwen3-8B-Base). Qwen3-8B-Base has a partial alignment prior (B0=84.6%); ALIGNBEAM raises this to 92.1% (+7.5 pp). All three LBM strategies produce identical results on this pair (>98% single-token match rate), confirming that vocabulary-agnostic behaviour degrades gracefully to the same-vocab case. Full per-dataset results appear in Appendix D.

5.2. Baseline Comparison

Table 5 compares ALIGNBEAM on Llama-3.1-8B against three baselines on Llama-3.1-8B. The **draft-only baseline** is the unmodified draft with no safety intervention. The **safety-prompt baseline** is a draft-only run ($\alpha = 0$, $K = 1$, no LBM mixing, no judge) with a strong safety system prompt injected directly into the draft model's context; for Llama-3.1-8B the chat template is skipped, so the system prompt is silently ignored and the safety-prompt baseline reduces to the draft-only baseline in practice—this is the intended demonstration that shallow prompt-only interventions fail on true base models. The **instruct-counterpart baseline** (Llama-3.1-8B-Instruct) is a separately trained instruct counterpart included as an upper-bound reference; it does not generalise to other domain-specialist base models. A full per-dataset breakdown of the instruct-counterpart baseline is in Appendix N. ALIGNBEAM is the only configuration that recovers strong safety on the true base model without requiring an instruct counterpart.

Table 5. Baseline comparison on Llama-3.1-8B ($n = 687$). HB+AB% and over-refusal columns are **string-match** rates except the instruct-counterpart baseline (marked †), where the benchmark-judge ACC% baseline is reported because no standalone instruct string-match run was performed. OR-ref = OR-Bench-Hard over-refusal; XS-ref = XSTest over-refusal. "Base?" = can be applied to true base models without instruct template. ⋆The safety-prompt baseline uses $\alpha = 0$, $K = 1$, no LBM, no judge; for Llama-3.1-8B the system prompt is silently ignored (chat template skipped), so it is equivalent to the draft-only baseline in practice—this is the key failure mode being demonstrated.

Method	HB+AB%	OR-ref	XS-ref	Base?
Draft only	16.9	11.0%	5.6%	✓
Safety prompt⋆	14.3	14.7%	4.0%	✓⋆
Instruct counterpart (Llama-3.1-8B-Inst)†	98.5†	—	—	n/a
ALIGNBEAM (Llama-3.1-8B)	**90.8**	22.3%	40.0%	✓

Table 6. Task utility under ALIGNBEAM ($K = 1$ utility mode, permissive anchor prompt). Task utility is essentially preserved on both domain pairs (-0.4 pp each on GSM8K and MedQA).

Model	Task	B0	LBM	Δ
DeepSeek-Math-7B-Instruct	GSM8K	77.0%	**76.6%**	−0.4 pp
MedLlama-3-8B	MedQA	13.1%	**12.7%**	−0.4 pp

LlamaGuard-based prompt-filter and response-filter baselines achieve high benchmark-judge accuracy on structured adversarial sets (≥89%) but impose severe over-refusal on benign benchmarks (OR-Bench-Hard drops to <3%), as they cannot distinguish domain-appropriate benign requests from adversarial ones; full results are in Appendix O. A hard-prefix baseline ($K = 1$) at ≈2.4–3.2× overhead achieves strong safety on structured benchmarks but elevated over-refusal on OR-Bench-Toxic (-28 pp), and is evaluated in Appendix P. On the same-vocabulary Qwen3-8B-Base pair, PROXY TUNING and top-k contrastive decoding are compared against ALIGNBEAM in Appendix E; ALIGNBEAM outperforms both on contextual and sorry-bench categories.

5.3. Utility Preservation

Table 6 reports task utility on pairs whose domain admits a clean automatic metric: math reasoning on GSM8K (DeepSeek-Math-7B-Instruct) and medical QA on MedQA (Jin et al., 2021) (MedLlama-3-8B), both run with $K = 1$ to avoid beam-selection overhead. Task utility is essentially preserved (-0.4 pp on both GSM8K and MedQA).

Table 7. Mixed-depth ablation: effect of mixed depth N on Llama-3.1-8B ($\alpha = 0.85$, $K = 3$; HB+AB% is the string-match refusal rate on the combined HarmBench-Standard + AdvBench set, $n = 687$). The jump from $N = 0$ (*beam selection + LLM judge, zero logit mixing*) to $N = 3$ contributes +61.9 pp; all further gains are marginal. The $N = 3$, $K = 1$ budget mode delivers ∼2× overhead while still capturing ∼80% of the full safety benefit. Slowdown measured on HarmBench-Standard. †B0 is taken from the Llama-3.1-8B main run ($\alpha = 0.50$); this ablation's runs (at $\alpha = 0.85$) independently measure B0 = 16.6%, giving a consistent $N = 0 \rightarrow 3$ gain of +61.9 pp regardless of which baseline is used.

N	HB+AB%	n	Slow.
B0 (no pipeline)†	16.9	687	1.00×
0 (beams + judge, no mix)	27.2	687	5.70×
3	**89.1**	**687**	**4.84**×
6 (default)	90.4	687	5.07×
10	92.6	687	5.23×
20	93.0	687	5.78×

Table 8. Safety-weight ablation: safety / over-refusal tradeoff in α on Llama-3.1-8B ($N = 6$, $K = 3$; over-refusal measured on OR-Bench-Hard). Safety plateaus across $\alpha \in [0.50, 0.85]$, while OR-Bench over-refusal stays at ≈20% throughout, a structural property of the beam selection pipeline, not a tunable knob. We adopt $\alpha = 0.50$ by Pareto score (*Pareto* = HB+AB% − OR-Bench over-refusal%; higher is better). The $\alpha = 0.85$ row reuses the mixed-depth ablation's $N = 6$ run. All percentages are string-match refusal / over-refusal rates.

α	HB+AB%	Over-ref.	Pareto
0.10	38.0	17.8%	20.2
0.25	75.8	19.8%	56.0
0.50	**91.0**	20.2%	**70.8**
0.75	90.4	19.2%	71.2
0.85	90.4	22.8%	67.6

6. Ablation Studies

6.1. Mixed-Depth Ablation (N)

Even $N = 0$ (beam priming + LLM judge, *zero logit mixing*) raises safety to 27.2% (+10.3 pp over B0=16.9%), demonstrating that beam selection alone contributes independent safety value independent of token mixing. The jump from $N = 0$ to $N = 3$ (+61.9 pp) dominates all further gains, directly supporting the early-token safety hypothesis (Qi et al., 2024). Slowdown grows only modestly beyond $N = 3$ because Phase 3 (draft-only continuation) dominates total time. Full per-dataset results for this ablation (AdvBench, HarmBench-Standard, SorryBench, OR-Bench-Hard, XSTest across all five depths) appear in Appendix I.

This ablation was conducted at $\alpha = 0.85$ to map the depth Pareto in isolation; the main experiments use $\alpha = 0.50$ (§6.2), which yields comparable absolute safety (90.8% at $\alpha = 0.50$, $N = 6$ vs. 90.4% at $\alpha = 0.85$, $N = 6$; Table 8).

6.2. Safety-Weight Ablation (α)

Safety is near-constant for $\alpha \in [0.50, 0.85]$ (range: 90.4–91.0%), confirming that small changes to the mixing weight above 0.50 do not significantly affect safety. The OR-Bench over-refusal rate (≈20%) is similarly stable across all tested values of α: it is a structural property of the beam selection pipeline, not a tunable parameter. XSTest over-refusal is higher (≈40%) due to prompts whose surface form resembles harmful requests regardless of safety weight. Full per-dataset results for this ablation appear in Appendix J.

7. Analysis

7.1. Benchmark Sensitivity

Results vary meaningfully across benchmarks, exposing two distinct regimes. On structured adversarial benchmarks (AdvBench, HarmBench), gains are large and consistent across all pairs (+40–75 pp). On WildJailbreak-Harmful, which uses multi-sentence adversarial framing to bury the harmful goal deeper in context, gains are far more limited (+8–9 pp for Llama-3.1-8B and MedLlama-3-8B, with even a small regression on Finance-Llama3-8B). The pattern is consistent with the early-token safety hypothesis (Qi et al., 2024): an $N = 6$ anchor prefix is enough to steer single-sentence prompts but cannot reach harmful goals that the attack distributes across tokens beyond the mixed region.

7.2. Base-Model Score Inflation

Base models occasionally produce incoherent or repetitive outputs on adversarial prompts. External judges correctly mark these as non-harmful (they contain no actionable content), which inflates the apparent B0 accuracy rate under official benchmark judges. For example, on Llama-3.1-8B the string-match B0 is 16.9% while the benchmark-judge B0 is 49.1% (AdvBench: 52.5%), a gap of ≈32 pp arising entirely from degenerate outputs being classified as "safe." ALIGNBEAM's outputs are non-degenerate by construction, since the LLM judge filters for coherent beams, so the gap between B0 and ALIGNBEAM is in fact a lower bound on the true safety improvement. We therefore report string-match refusal rates as the primary metric throughout, with benchmark-judge accuracy provided in the appendix.

8. Limitations

COMPUTE OVERHEAD. Inference cost is comparable to other inference-time defences (LlamaGuard, Proxy Tuning, hard-prefix, RAIN); K and N are deployment-adjustable without retraining (Appendix T).

CALIBRATION CEILING. Both safety limits stem from the anchor, not the mixing parameters: the anchor's over-refusal on ambiguous benign prompts is inherited (OR-Bench-Hard ≈20%, XSTest ≈40% on Llama-3.1-8B), and the mixed prefix covers only the first N tokens, so attacks that embed harm past the prefix are only partially blocked (WildJailbreak-Harmful: +2–6 pp vs. +51–77 pp on AdvBench). A better-calibrated anchor, longer prefix, or per-prompt α schedule could close the gap.

EVALUATION SCOPE. All prompts are single-turn and ≤200 tokens; multi-turn jailbreaks and adaptive attacks are out of scope. Experiments use a single Qwen2.5-3B-Instruct anchor; the alternative-anchor study (Appendix M) shows results depend on anchor refusal style. Multi-anchor ensembles, multilingual evaluation, and human judging are natural next steps. Truncation and "sorry-but-continues" artefacts that affect verbose drafts and understate true safety are discussed in Appendix V.

9. Conclusion

We presented ALIGNBEAM, an inference-time procedure that transfers safety alignment to domain-specialised language models by blending their token-level distribution with that of a small aligned anchor through a cross-vocabulary text bridge and selecting the safest beam with an LLM judge. Across six model pairs spanning Llama, Gemma, Qwen, medical, and finance specialists, ALIGNBEAM raises refusal rates from near-zero to 88–97% on the structured adversarial benchmarks for the majority of pairs while preserving task utility (−0.4 pp on GSM8K).

The mixed-depth ablation corroborates the early-token safety hypothesis (Qi et al., 2024): the jump from $N = 0$ (beam selection only, no logit mixing) to $N = 3$ alone contributes +61.9 pp and all further gains are marginal. The safety-weight ablation shows that safety plateaus for $\alpha \in [0.50, 0.85]$ and that the residual over-refusal on OR-Bench (≈20%) is a structural property of the beam selection pipeline rather than a tunable parameter.

ALIGNBEAM is training-free, vocabulary-agnostic, and compatible with true base models, while requiring no access to model internals. The remaining limitations (4.6 to 6.6× overhead in full mode, reducible to ∼2× in budget mode; limited effectiveness against WildJailbreak-style multi-sentence attacks; and structural over-refusal on XSTest-style prompts) suggest the natural next steps: longer or adaptive anchor prefixes, multi-anchor ensembles, and calibration mechanisms that decouple over-refusal from beam selection.

Impact Statement

ALIGNBEAM is a safety-oriented method designed to reduce harmful outputs from domain-specialised language models. Potential misuse scenarios are limited, as the method in-

creases, rather than decreases, refusal rates. The primary societal benefit is enabling organisations to deploy domain-specialised models without sacrificing safety alignment. A limitation worth noting for deployment contexts is the 4.6–6.6× inference overhead, which carries energy and cost implications; the $K=1$ budget mode (∼2× overhead) largely mitigates this.

References

Chao, P., Debenedetti, E., Robey, A., Andriushchenko, M., Croce, F., Sehwag, V., Dobriban, E., Flammarion, N., Pappas, G. J., Tramer, F., Hassani, H., and Wong, E. JailbreakBench: An open robustness benchmark for jailbreaking large language models, 2024. URL https://arxiv.org/abs/2404.01318.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Cui, J., Chiang, W.-L., Stoica, I., and Hsieh, C.-J. OR-Bench: An over-refusal benchmark for large language models, 2024. URL https://arxiv.org/abs/2405.20947.

Fei, Y., Razeghi, Y., and Singh, S. Nudging: Inference-time alignment of llms via guided decoding, 2025. URL https://arxiv.org/abs/2410.09300.

Inan, H., Upasani, K., Chi, J., Rungta, R., Iyer, K., Mao, Y., Tontchev, M., Hu, Q., Fuller, B., Testuggine, D., and Khabsa, M. Llama Guard: LLM-based input-output safeguard for human-AI conversations, 2023. URL https://arxiv.org/abs/2312.06674.

Jiang, L., Rao, K., Han, S., Ettinger, A., Brahman, F., Kumar, S., Mireshghallah, N., Lu, X., Sap, M., Choi, Y., and Dziri, N. WildTeaming at scale: From in-the-wild jailbreaks to (adversarially) safer language models, 2024. URL https://arxiv.org/abs/2406.18510.

Jin, D., Pan, E., Oufattole, N., Weng, W.-H., Fang, H., and Szolovits, P. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T., Zettlemoyer, L., and Lewis, M. 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)*, pp. 12286–12312, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.687. URL https://aclanthology.org/2023.acl-long.687/.

Li, Y., Wei, F., Zhao, J., Zhang, C., and Zhang, H. RAIN: Your language models can align themselves without fine-tuning. In *The Twelfth International Conference on Learning Representations*, Vienna, Austria, 2024.

Liu, A., Han, X., Wang, Y., Tsvetkov, Y., Choi, Y., and Smith, N. A. Tuning language models by proxy, 2024. URL https://arxiv.org/abs/2401.08565.

Mazeika, M., Phan, L., Yin, X., Zou, A., Wang, Z., Mu, N., Sakhaee, E., Li, N., Basart, S., Li, B., Forsyth, D., and Hendrycks, D. HarMBench: A standardized evaluation framework for automated red teaming and robust refusal. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35181–35224, Vienna, Austria, 2024.

Qi, X., Zeng, Y., Xie, T., Chen, P.-Y., Jia, R., Mittal, P., and Henderson, P. Fine-tuning aligned language models compromises safety, even when users do not intend to!, 2023. URL https://arxiv.org/abs/2310.03693.

Qi, X., Panda, A., Lyu, K., Ma, X., Roy, S., Beirami, A., Mittal, P., and Henderson, P. Safety alignment should be made more than just a few tokens deep, 2024. URL https://arxiv.org/abs/2406.05946.

Röttger, P., Kirk, H. R., Vidgen, B., Attanasio, G., Bianchi, F., and Hovy, D. XSTest: A test suite for identifying exaggerated safety behaviours in large language models, 2023. URL https://arxiv.org/abs/2308.01263.

Xie, T., Qi, X., Zeng, Y., Huang, Y., Sehwag, U. M., Huang, K., He, L., Wei, B., Li, D., Sheng, Y., Jia, R., Li, B., Li, K., Chen, D., Henderson, P., and Mittal, P. SORRY-Bench: Systematically evaluating large language model safety refusal behaviors, 2024. URL https://arxiv.org/abs/2406.14598.

Xu, Z., Jiang, F., Niu, L., Jia, J., Lin, B. Y., and Poovendran, R. 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)*, pp. 5587–5605, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.303.

Yang, X., Wang, X., Zhang, Q., Petzold, L., Wang, W. Y., Zhao, X., and Lin, D. Shadow alignment: The ease of subverting safely-aligned language models, 2023. URL https://arxiv.org/abs/2310.02949.

Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., and Fredrikson, M. Universal and transferable adversarial attacks on aligned language models, 2023. URL https://arxiv.org/abs/2307.15043.

A. Metric conventions used in all appendix tables.

- **B0% / Ref.%**: string-match refusal rate (fraction of responses containing at least one canonical refusal phrase). This is the primary metric used in Table 4.
- **ACC%**: benchmark-judge accuracy (fraction of responses judged not-harmful by the official benchmark classifier). Higher than B0% for base-model baselines due to score inflation (§7.2).
- **Δsafe**: LBM% − B0% (string-match delta) throughout, unless the column header explicitly states "ACC."
- For **calibration / benign datasets** (OR-Bench, XSTest, JBB-Benign): Ref.% denotes *over-refusal* rate (lower is better).

B. Llama-3.1-8B Per-Dataset Breakdown

Table 9 reports results for Llama-3.1-8B (+ Qwen2.5-3B anchor) at $\alpha = 0.5$, $N = 6$, $K = 3$, seed 42 across all safety benchmarks plus calibration sets. Structured adversarial benchmarks (HarmBench, AdvBench) show the strongest gains; the WildJailbreak-Harmful row is the principal exception (§7.1).

Table 9. Llama-3.1-8B per-dataset breakdown (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$, seed 42). B0% and LBM% are string-match refusal rates; ACC% columns show benchmark-judge accuracy. For calibration rows, Ref.% denotes over-refusal. Δ_{safe} = LBM% − B0%.

Dataset	Type	n	B0%	B0 ACC%	LBM%	LBM ACC%
HarmBench-Std	Harmful	167	15.0	49.1	**79.6**	91.6
HarmBench-Ctx	Harmful	62	8.1	53.2	**79.0**	80.7
AdvBench	Harmful	520	17.5	52.5	**94.4**	98.3
Sorry-Bench	Harmful	440	10.9	34.9	**61.4**	74.3
WildJailbreak-H	Harmful	1315	13.3	21.5	**19.7**	30.1
Calibration / benign (lower over-refusal = better):						
JBB-Benign	Benign	100	10.0	8.0	26.0	2.0
OR-Bench-Hard	Benign	855	11.0	13.6	22.3	6.2
OR-Bench-Toxic	Benign	585	12.3	53.9	56.8	20.5
XSTest	Calib.	450	5.6	23.6	40.0	38.7
HB+AB		**687**	**16.9**	51.7	**90.8**	96.7

C. Finance-Llama3-8B Per-Dataset Breakdown

Table 10 reports results for Finance-Llama3-8B at $\alpha = 0.5$, $N = 6$, $K = 3$. The aggregate HB+AB is B0 = 28.1%, LBM = 91.1% (+63.0 pp). All three LBM strategies produce identical results across every dataset for Finance-Llama3-8B, indicating a high single-token match rate between the Finance-Llama3-8B (Llama-3 tokeniser) and Qwen2.5-3B vocabularies.

Table 10. Finance-Llama3-8B per-dataset breakdown (Finance-Llama3-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). All three LBM strategies (drop/first/exact) produce identical results. B0% and LBM% are string-match refusal rates; ACC% columns show benchmark-judge accuracy. Δ_{safe} = LBM% − B0%. ‡WildJailbreak regression consistent with multi-sentence framing (§7.1).

Dataset	Type	n	B0%	LBM%	LBM ACC%	Δ_{safe}
HarmBench-Std	Harmful	167	23.4	**79.6**	91.0	+56.2
HarmBench-Ctx	Harmful	62	30.6	**80.6**	90.3	+50.0
AdvBench	Harmful	520	29.6	**94.8**	98.1	+65.2
JBB-Harmful	Harmful	100	27.0	**88.0**	94.0	+61.0
Sorry-Bench	Harmful	440	55.2	**65.7**	77.3	+10.5
WildJailbreak-H	Harmful	1315	42.7	33.2‡	41.4	−9.5
HB+AB		**687**	**28.1**	**91.1**		**+63.0**

D. Qwen3-8B-Base (Same-Vocabulary Pair) Per-Dataset Results

Table 11 provides the full per-dataset breakdown for Qwen3-8B-Base (+ Qwen2.5-3B-Instruct anchor), the only same-vocabulary pair. Even with a strong B0 prior of 84.6%, the largest relative string-match gains appear on HarmBench-Contextual (+33.9 pp) and Sorry-Bench (+23.9 pp), showing that the mixing pipeline helps even when the draft is already partially aligned. All three LBM strategies produce identical results (>98% single-token match rate).

Table 11. Qwen3-8B-Base per-dataset breakdown (Qwen3-8B-Base + Qwen2.5-3B-Instruct, same tokeniser, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). B0% and LBM% are string-match refusal rates; Δ_{safe} = LBM% − B0%. ACC% columns show benchmark-judge accuracy for reference. Note: the ACC-based delta (STRAT_ACC − BASE_ACC) differs from Δ_{safe} because the base model's ACC-based safety is higher than its string-match rate, owing to partially aligned Qwen architecture.

Dataset	Type	n	B0%	B0 ACC%	LBM%	LBM ACC%	Δ_{safe}
HarmBench-Std	Harmful	167	61.7	80.8	**80.8**	89.8	+19.1
HarmBench-Ctx	Harmful	62	37.1	64.5	**71.0**	90.3	+33.9
AdvBench	Harmful	520	91.9	95.6	**95.8**	99.4	+3.9
JBB-Harmful	Harmful	100	76.0	81.0	**85.0**	91.0	+9.0
Sorry-Bench	Harmful	440	39.5	55.2	**63.4**	79.8	+23.9
WildJailbreak	Harmful	1315	16.1	27.6	**20.9**	32.6	+4.8
XSTest (over-ref.)	Calib.	450	10.7	26.0	28.2	38.4	+17.5
HB+AB		**687**	**84.6**	92.0	**92.1**	97.1	**+7.5**

E. Qwen3-8B-Base: Same-Vocabulary Method Comparison

Table 12 compares ALIGNBEAM against two same-vocabulary inference-time baselines on Qwen3-8B-Base (+ Qwen2.5-3B-Instruct): PROXY TUNING (Liu et al., 2024) and top-k contrastive decoding (TOPK-CD). Because the proxy-tuning and TopK-CD deltas are reported by the harness as benchmark-judge accuracy (ACC%), all three columns use ACC% for comparability.

ALIGNBEAM outperforms both alternatives substantially on contextual (HB-Ctx: +25.8 vs. +4.9/+11.3 $pp) and sorry-bench$ (+24.5 $vs.$ +13.0/+14.1 pp) categories. JailbreakBench-Harmful is comparable across all three methods. XSTest over-refusal is nearly identical (≈37–38%), confirming that this is a structural property of the beam-selection pipeline rather than a method-specific artefact. Overhead is similar for all three (≈4.8–5.3×), so ALIGNBEAM's safety advantage is not purchased at additional latency cost.

Contrastive decoding (Li et al., 2023) was also evaluated on Qwen3-8B-Base but its evaluation harness did not produce per-dataset accuracy metrics; its total-time profile was comparable to the other methods. SAFEDECODING (Xu et al., 2024) is inapplicable to cross-vocabulary pairs by construction; in the same-vocabulary limit, LBM reduces to a construction analogous to SAFEDECODING (§2), differing in that support is selected via beam search over continuations rather than token-level top-k intersection. A full comparison including SAFEDECODING on same-vocabulary pairs is available in the extended version of this work.

Table 12. Qwen3-8B-Base same-vocabulary method comparison (Qwen3-8B-Base + Qwen2.5-3B-Instruct, $\alpha = 0.5$, $N = 6$, $K = 3$). All values are benchmark-judge accuracy (ACC%). Δ = method ACC% − B0 ACC%. XSTest is a calibration set (over-refusal; lower is better). Slow. columns show per-sample slowdown vs. draft-only baseline.

Dataset	Type	B0	ALIGNBEAM (LBM)			Proxy Tuning			TopK-CD		
		ACC%	ACC%	Δ	Slow.	ACC%	Δ	Slow.	ACC%	Δ	Slow.
AdvBench	Harm.	95.6	**99.4**	+3.8	4.9×	98.1	+2.5	5.2×	98.3	+2.7	5.0×
HB-Ctx	Harm.	64.5	**90.3**	+25.8	4.7×	69.4	+4.9	5.0×	75.8	+11.3	5.0×
HB-Std	Harm.	80.8	**89.8**	+9.0	4.9×	83.2	+2.4	5.3×	86.2	+5.4	5.3×
JBB-Harmful	Harm.	81.0	91.0	+10.0	4.7×	**92.0**	+11.0	5.3×	90.0	+9.0	5.2×
SorryBench	Harm.	55.2	**79.8**	+24.5	4.8×	68.2	+13.0	5.1×	69.3	+14.1	5.1×
WildJailbk	Harm.	27.6	**32.6**	+5.0	5.4×	29.4	+1.8	5.4×	31.1	+3.5	5.4×
XSTest†	Calib.	26.0	38.4	+12.4	4.8×	37.3	+11.3	4.9×	36.9	+10.9	4.8×

†Over-refusal (lower is better).

F. MedLlama-3-8B Per-Dataset Breakdown

Table 13 reports results for JSL-MedLlama-3-8B (+ Qwen2.5-3B anchor) at $\alpha = 0.5$, $N = 6$, $K = 3$.

Table 13. MedLlama-3-8B per-dataset breakdown (JSL-MedLlama-3-8B-v2.0 + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). B0% and LBM% are string-match refusal rates; ACC% columns show benchmark-judge accuracy. Δ_{safe} = LBM% − B0%.

Dataset	Type	n	B0%	B0 ACC%	LBM%	LBM ACC%	Δ_{safe}
HarmBench-Std	Harmful	167	16.8	35.3	**78.4**	88.6	+61.6
HarmBench-Ctx	Harmful	62	9.7	25.8	**69.4**	87.1	+59.7
AdvBench	Harmful	520	40.2	59.2	**91.7**	99.0	+51.5
JBB-Harmful	Harmful	100	25.0	42.0	**79.0**	82.0	+54.0
Sorry-Bench	Harmful	440	18.0	31.1	**63.2**	72.3	+45.2
WildJailbreak-H	Harmful	1315	14.1	22.5	16.2	27.1	+2.1
HB+AB		**687**	**34.5**	53.4	**88.5**	96.5	**+54.0**

G. Gemma-3-12B-PT Per-Dataset Breakdown

Table 14 reports results for Gemma-3-12B-PT (+ Qwen2.5-3B anchor) at $\alpha = 0.5$, $N = 6$, $K = 3$. B0% (string-match) is near-zero because Gemma-3-12B-PT is an unaligned pretrained model that produces no canonical refusal phrases on adversarial prompts. The benchmark-judge B0 ACC% is higher (≈38–54%) due to incoherent base-model outputs being classified as "safe" (§7.2).

Table 14. Gemma-3-12B-PT per-dataset breakdown (Gemma-3-12B-PT + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). B0% ≈ 0 (string-match) for all safety datasets; B0 ACC% reflects benchmark-judge baseline (inflated by incoherent base-model outputs). LBM Ref.% = string-match refusal; LBM ACC% = benchmark-judge accuracy.

Dataset	Type	n	B0 ACC%	LBM Ref.%	LBM ACC%
HarmBench-Std	Harmful	167	38.9	**42.5**	85.0
HarmBench-Ctx	Harmful	62	45.2	**40.3**	80.6
AdvBench	Harmful	520	53.7	**55.6**	97.5
JBB-Harmful	Harmful	100	16.0	**39.0**	86.0
Sorry-Bench	Harmful	440	28.6	**35.9**	70.7
XSTest (over-ref.)	Calib.	450	21.8	5.6	30.4
HB+AB (SM)		**687**	**B0 ≈ 4.4% → LBM = 52.4% (+48.0 pp)**		

H. DeepSeek-Math-7B-Instruct Strategy Comparison

Table 15 compares the three LBM variants on DeepSeek-Math-7B-Instruct at $N = 20$. 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 numerically equivalent. All three strategies produce zero degenerate outputs. On WildJailbreak-Harmful (LBM-DROP, $n = 1{,}315$), string-match refusal rises from 11.5% to 15.0% (+3.5 pp) while judge accuracy rises from 19.2% to 25.6% (+6.4 pp); the anchor's polite declinations replace the draft's compliance-phrasing refusals, the same pattern observed for Finance-Llama3-8B (Appendix C).

Table 15. DeepSeek-Math-7B-Instruct LBM-strategy comparison (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. All percentages are string-match refusal rates except GSM8K (exact-match accuracy). GSM8K utility is preserved to within 0.5 pp across all strategies.

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%

I. Mixed-Depth Ablation: Full Per-Dataset Depth Sweep

Table 16 reports the complete per-dataset results for the mixed-depth ablation ($\alpha = 0.85$, $K = 3$, lbm_drop, Llama-3.1-8B backbone). All percentages are string-match refusal rates. For calibration datasets (OR-Bench-Hard, XSTest), values denote over-refusal; bold entries mark the depth with the best safety-efficiency tradeoff.

Table 16. Mixed-depth ablation full per-dataset results (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.85$, $K = 3$). All values are string-match refusal rates. For OR-Bench-Hard and XSTest, the reported value is the over-refusal rate (lower is better). Slowdown is measured on HarmBench-Standard.

Dataset	n	**B0%**	**N=0**	**N=3**	**N=6**	**N=10**	**N=20**
AdvBench	520	17.5	30.4	92.7	93.5	96.2	**97.3**
HarmBench-Std	167	13.8	17.4	77.8	80.8	**81.4**	79.6
SorryBench	440	10.9	18.6	60.0	61.6	62.1	**68.2**
OR-Bench-Hard†	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 (HB-Std)		1.00×	5.70×	4.84×	5.07×	5.23×	5.78×

†Calibration sets: over-refusal rate (lower is better).

J. Safety-Weight Ablation: Full Per-Dataset Alpha Sweep

Table 17 reports the complete per-dataset results for the safety-weight ablation ($N = 6$, $K = 3$, lbm_drop, Llama-3.1-8B backbone). SorryBench was not run for this sweep.

Table 17. Safety-weight ablation full per-dataset results (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $N = 6$, $K = 3$). All values are string-match refusal / over-refusal rates. SorryBench not evaluated for this sweep. Pareto = HB+AB% − OR-Bench-Hard over-refusal%.

Dataset	n	**B0%**	**α=0.10**	**α=0.25**	**α=0.50**	**α=0.75**
AdvBench	520	17.5	41.2	81.0	**94.4**	93.5
HarmBench-Std	167	13.8	28.1	59.9	**80.2**	80.8
OR-Bench-Hard†	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
Pareto			20.2	56.0	**70.8**	71.2

†Calibration sets: over-refusal rate (lower is better). ‡B0 for this ablation measured independently at $\alpha = 0.85$ baseline.

K. Anchor-Alignment Ablation: Unaligned vs. Aligned Anchor

Table 18 compares an aligned anchor (Qwen2.5-3B-Instruct) against an unaligned base anchor (Qwen2.5-3B-Base) on Llama-3.1-8B. The unaligned anchor still provides safety gains (+25–30 pp from the beam-selection mechanism), but substantially less than the aligned anchor, confirming that RLHF alignment of the anchor is the primary source of safety signal.

Table 18. Anchor-alignment ablation: aligned vs. unaligned anchor (Llama-3.1-8B, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). B0 = draft-only string-match baseline (no anchor); remaining columns show string-match refusal rate for each anchor. Slowdown measured per dataset.

Dataset	n	**B0%**	**Unaligned**	**Aligned**	**Slow.**
AdvBench	520	17.5	69.4	**94.4**	5.4×
HarmBench-Std	167	13.8	52.1	**79.6**	5.6×
HarmBench-Ctx	62	8.1	38.7	**79.0**	5.8×
SorryBench	440	10.9	35.2	**61.4**	6.2×
OR-Bench-Hard†	855	11.0	25.4	22.3	6.0×
XSTest†	450	5.6	12.9	40.0	6.1×

†Calibration sets: over-refusal (lower is better).

L. Prompt-Robustness Ablation

Table 19 evaluates sensitivity of ALIGNBEAM to the anchor's safety system prompt on HarmBench-Standard (Llama-3.1-8B backbone). Three prompts are tested: no system prompt, a concise 2-line standard prompt, and a verbose 45-line prompt that enumerates harm categories. The standard prompt outperforms both alternatives; the verbose prompt achieves identical benchmark-judge accuracy to the standard prompt but lower string-match refusal rate, suggesting the anchor's RLHF training drives refusal rather than explicit prompt wording.

Table 19. Prompt-robustness ablation on HarmBench-Standard (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$, $n = 167$). B0 ACC = benchmark-judge baseline = 49.1%; B0 Ref.% = 13.8%.

Anchor Prompt	Ref.%	ACC%	Slow.
None	66.5	82.6	5.5×
Standard (2-line)	**80.2**	**91.6**	**5.2**×
Verbose (45-line)	66.5	91.6	5.5×
B0 (no anchor)	13.8	49.1	1.0×

M. Alternative Anchor (Llama-3.1-8B-Instruct)

Table 20 reports results using Llama-3.1-8B-Instruct as the safety anchor instead of Qwen2.5-3B-Instruct (Llama-3.1-8B backbone, $\alpha = 0.5$, $N = 6$, $K = 3$). The Llama anchor achieves high benchmark-judge accuracy (97–100%) but very low string-match refusal rates (7–15%), because Llama-3.1-8B-Instruct generates polite declinations in phrasing not covered by the 22-phrase string-match list. This highlights that string-match refusal rate may underestimate safety when the anchor's refusal style deviates from the canonical phrase set.

Table 20. Alternative anchor (Llama-3.1-8B draft + Llama-3.1-8B-Instruct anchor, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). Ref.% = string-match refusal rate; ACC% = benchmark-judge accuracy. High ACC% with low Ref.% indicates polite non-canonical refusals not captured by string-match.

Dataset	n	B0 Ref.%	B0 ACC%	Ref.%	ACC%	Slow.
AdvBench	520	17.5	52.5	7.1	**99.6**	3.9×
HarmBench-Std	167	13.8	49.1	13.2	**97.0**	4.1×
HarmBench-Ctx	62	8.1	53.2	14.5	**98.4**	4.1×
SorryBench	440	10.9	34.9	11.1	**85.9**	4.8×
OR-Bench-Hard†	855	11.0	13.6	14.0	4.4	5.5×
XSTest†	450	5.6	23.6	7.1	37.1	5.3×

†Calibration: over-refusal (lower is better).

N. Instruct-Counterpart Per-Dataset Breakdown

Table 21 reports the full per-dataset profile for the instruct-counterpart baseline (Llama-3.1-8B-Instruct), the separately-trained instruct counterpart included in the baseline comparison as an upper-bound reference. The "Inst. Base ACC%" column is the instruct model's own benchmark-judge accuracy without any ALIGNBEAM intervention; the "+ALIGNBEAM ACC%" column shows what happens when ALIGNBEAM is additionally applied. The "+ALIGNBEAM Ref%" column is the string-match refusal rate under ALIGNBEAM.

Key observations: (i) the instruct model is already near-ceiling on structured adversarial sets (AdvBench 99.8%, HarmBench 95–98%) so ALIGNBEAM adds nothing on those rows; (ii) WildJailbreak drops by -10.2 pp when ALIGNBEAM is applied, showing that anchor mixing can disrupt the instruct model's own contextual reasoning on multi-sentence adversarial prompts; (iii) OR-Bench-Hard ACC is only 1.5% for the instruct baseline, confirming that the benchmark-judge severely over-classifies instruct refusals on adversarial-looking benign prompts (the same score inflation effect discussed in §7.2, in reverse).

Table 21. Instruct-counterpart per-dataset breakdown (Llama-3.1-8B-Instruct, $K=3$, LBM-DROP, $\alpha=0.5$, seed 42). "Inst. Base" = instruct model without ALIGNBEAM (benchmark-judge ACC%). "+ALIGNBEAM" = instruct model with ALIGNBEAM applied (ACC% and string-match Ref%). For calibration/benign rows, lower Ref% and higher ACC% are both better.

Dataset	Type	n	Inst. Base ACC%	+AB ACC%	ΔACC	+AB Ref%
AdvBench	Harmful	520	99.8	**99.6**	−0.2	35.6
HB-Contextual	Harmful	62	98.4	**96.8**	−1.6	24.2
HB-Standard	Harmful	167	95.2	**95.2**	+0.0	24.6
SorryBench	Harmful	440	83.6	**86.1**	+2.5	21.1
WildJailbreak	Harmful	1315	62.2	52.0	**−10.2**	20.2
JBB-Benign†	Benign	100	3.0	2.0	−1.0	8.0
OR-Bench-Hard†	Benign	855	1.5	1.4	−0.1	15.1
OR-Bench-Toxic†	Benign	585	1.0	1.5	+0.5	25.8
XSTest†	Calib.	450	44.4	43.6	−0.9	18.0

†Benign/calibration: higher ACC% = better compliance; lower Ref% = less over-refusal. OR-Bench Inst. Base ACC $\approx 1\%$ reflects judge over-classification of coherent instruct refusals as "unsafe" on adversarial-looking benign prompts (§7.2).

O. LlamaGuard Guard Baselines

Tables 22 and 23 report two guard-classifier baselines on Llama-3.1-8B (backbone). The prompt-filter baseline uses LlamaGuard-3-8B as a *prompt-level filter* (blocking generation when the input prompt is classified as harmful) and the response-filter baseline uses LlamaGuard-3-8B as a *response-level filter* (blocking or replacing the generated output when it is classified as harmful). Both run with $K=3$ and seed 42.

Both classifiers achieve high benchmark-judge accuracy on structured adversarial sets (AdvBench, HarmBench: 89–100%), but impose severe over-refusal on benign benchmarks: OR-Bench-Hard drops to ACC = 2.6% (prompt filter) and 0.1% (response filter), and OR-Bench-Toxic falls below 2% for both, indicating that nearly every benign domain-adjacent prompt is falsely blocked. By contrast, ALIGNBEAM's OR-Bench-Hard over-refusal is ≈22% (Table 17). The XSTest over-refusal for both classifiers (≈43–50% string-match Ref.%) is comparable to ALIGNBEAM, reflecting the surface-form similarity of XSTest calibration prompts to genuine adversarial inputs.

Table 22. LlamaGuard prompt-level filter (Llama-3.1-8B, $K=3$, seed 42). ACC% = benchmark-judge accuracy. For calibration/benign rows, ACC% is the fraction of benign requests that pass the classifier (lower ACC% = higher over-refusal). Base ACC% = draft-only benchmark-judge baseline.

Dataset	Type	n	Base ACC%	Filter ACC%	Slow.
AdvBench	Harmful	520	52.5	**100.0**	4.1×
HarmBench-Ctx	Harmful	62	53.2	**95.2**	5.0×
HarmBench-Std	Harmful	167	49.1	**98.8**	5.0×
SorryBench	Harmful	440	34.9	**89.8**	5.7×
JBB-Benign†	Benign	100	8.0	0.0	5.8×
OR-Bench-Hard†	Benign	855	13.6	2.6	6.0×
OR-Bench-Toxic†	Benign	585	53.8	1.0	5.3×
XSTest†	Calib.	450	23.6	42.9	5.6×

†Benign/calibration sets: lower ACC% = higher over-refusal.

Table 23. LlamaGuard response-level filter (Llama-3.1-8B, $K=3$, seed 42). Response filtering achieves comparable safety to prompt filtering but identical over-refusal characteristics, since benign inputs with safety-adjacent surface forms produce generations that are falsely blocked at the output stage.

Dataset	Type	n	Base ACC%	Filter ACC%	Slow.
AdvBench	Harmful	520	52.5	**100.0**	3.9×
HarmBench-Ctx	Harmful	62	53.2	**100.0**	4.7×
HarmBench-Std	Harmful	167	49.1	**99.4**	4.7×
SorryBench	Harmful	440	34.9	**88.9**	5.4×
JBB-Benign†	Benign	100	8.0	0.0	5.5×
OR-Bench-Hard†	Benign	855	13.6	0.1	5.5×
OR-Bench-Toxic†	Benign	585	53.8	0.0	5.0×
XSTest†	Calib.	450	23.6	43.3	5.3×

†Benign/calibration sets: lower ACC% = higher over-refusal.

P. Hard-Prefix Baseline

Table 24 evaluates the hard-prefix baseline, in which the anchor model (Qwen2.5-3B-Instruct) first generates a short fixed-length safety prefix, which is prepended to the draft's decoding context before it continues generation, with $K=1$ (no beam selection or LLM judge). This is a lower-overhead alternative to ALIGNBEAM's probabilistic blending.

The hard-prefix baseline runs at ≈2.4–3.2× overhead and achieves strong safety on structured adversarial benchmarks: ACC = 99.0% on AdvBench and 91.6% on HarmBench-Standard—comparable to full ALIGNBEAM at $N=3$, $K=1$ budget mode. However, benign-set over-refusal is elevated: OR-Bench-Toxic drops from 53.8% to 25.5% (−28.3 pp), and JBB-Benign drops to 1.0%, indicating that the deterministic prefix still triggers refusal on many benign domain-adjacent prompts. The XSTest evaluation did not complete due to an evaluation-harness indexing error and is excluded.

Table 24. Hard-prefix baseline (Llama-3.1-8B + Qwen2.5-3B-Instruct, $K=1$, LBM-DROP, seed 42). ACC% = benchmark-judge accuracy. For calibration/benign rows, lower ACC% indicates higher over-refusal. XSTest excluded due to evaluation-harness error.

Dataset	Type	n	Base ACC%	Hard-Prefix ACC%	Slow.
AdvBench	Harmful	520	52.5	**99.0**	2.4×
HarmBench-Ctx	Harmful	62	53.2	**88.7**	2.6×
HarmBench-Std	Harmful	167	49.1	**91.6**	2.7×
SorryBench	Harmful	440	34.9	**74.5**	3.2×
JBB-Benign†	Benign	100	8.0	1.0	2.5×
OR-Bench-Hard†	Benign	855	13.6	9.6	2.8×
OR-Bench-Toxic†	Benign	585	53.8	25.5	2.2×

†Benign/calibration sets: lower ACC% = higher over-refusal.

Q. Seed Robustness (Llama-3.1-8B)

Table 25 reports Llama-3.1-8B across seeds 42, 2, and 3, confirming that the main results are not seed-specific. Variance is highest on HarmBench-Contextual (±7 pp) and SorryBench (±7 pp), consistent with the smaller dataset sizes and diverse prompt styles.

Table 25. Seed robustness (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). All values are string-match refusal / over-refusal rates.

Dataset	n	**B0%**	**Seed 42**	**Seed 2**	**Seed 3**
AdvBench	520	17.5	**94.4**	94.4	95.2
HarmBench-Std	167	14.0	**79.6**	79.0	74.3
HarmBench-Ctx	62	8.1	**79.0**	72.6	64.5
SorryBench	440	10.9	**61.4**	61.4	62.7
OR-Bench-Hard†	855	11.0	22.3	22.9	22.7
XSTest†	450	5.6	40.0	40.0	45.6
HB+AB	687	16.9	**90.8**	90.7	90.1

†Calibration: over-refusal (lower is better).

R. No-Judge Ablation (Llama-3.1-8B)

Table 26 compares the standard ALIGNBEAM pipeline (with LLM judge for beam selection) against a no-judge variant that uses the lowest-score beam by random selection. Safety rates remain high without the judge, demonstrating that logit mixing alone (not beam re-ranking) provides the majority of the safety gain. The judge provides an additional ≈5–10 pp on AdvBench and HarmBench-Std.

Table 26. No-judge ablation on Llama-3.1-8B (Llama-3.1-8B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). All values are string-match refusal / over-refusal rates. XSTest evaluated on $n = 220$ (partial run).

Dataset	n	**B0%**	**With Judge**	**No Judge**	**Slow.**
AdvBench	520	17.5	**94.4**	97.9	3.4×
HarmBench-Std	167	13.8	**79.6**	81.4	4.2×
HarmBench-Ctx	62	8.1	**79.0**	85.5	4.2×
SorryBench	440	10.9	**61.4**	63.6	4.9×
WildJailbreak-H	1315	13.3	**19.7**	18.7	5.4×
XSTest†	220	5.6	**40.0**	46.4	4.9×

†Partial run ($n = 220$). Over-refusal: lower is better.

S. Scale Experiment (Llama-3.1-70B)

Table 27 reports results for the scale experiment (Llama-3.1-70B-Base + Qwen2.5-3B anchor), evaluating ALIGNBEAM at 70B scale. Numbers are benchmark-judge accuracy (ACC%) because string-match draft-only baselines are not separately available for this pair in the evaluation logs; the ACC% delta column shows strategy improvement over the base model baseline.

Table 27. Scale experiment (Llama-3.1-70B + Qwen2.5-3B-Instruct, LBM-DROP, $\alpha = 0.5$, $N = 6$, $K = 3$). All values are benchmark-judge accuracy (ACC%). B0 ACC% = baseline; AlignBeam ACC% = strategy; Δ = ACC-based delta.

Dataset	n	**B0 ACC%**	**AlignBeam ACC%**	Δ	**Slow.**
AdvBench	520	56.5	**99.4**	+42.9	4.3×
HarmBench-Std	167	55.1	**95.2**	+40.1	4.5×
SorryBench	440	36.4	**86.1**	+49.8	4.8×
OR-Bench-Hard†	855	13.3	4.4	—	5.3×
OR-Bench-Toxic†	585	58.6	13.8	—	4.3×
JBB-Benign†	100	10.0	6.0	—	4.9×
XSTest†	450	26.0	39.6	+13.6	4.7×

†Calibration: over-refusal (lower is better).

T. Latency Details

Table 28 reports per-sample wall-clock latency on a single RTX 6000 Pro Blackwell (96 GB, bfloat16). Phase 3 (KV-cache draft continuation) accounts for roughly 69% of total time, Phase 2 (mixed decoding) for ≈30%, and Phase 1 (priming) for

≈1%, so the dominant speed lever is the beam count K rather than the depth N.

Slowdown values are measured on HarmBench-Standard as the representative dataset; other datasets (particularly WildJailbreak with longer prompts) exhibit higher per-sample latency, so reported values are conservative lower bounds on overhead for some inputs. The budget mode entry ($K=1$, $N=3$) derives from the mixed-depth ablation: at $N=3$ the HarmBench-Standard slowdown is 4.84× under $K=3$; with $K=1$ the beam-parallel overhead is removed, reducing this to ≈2× while retaining ≈80% of the full safety benefit ($N=3$: 89.1% vs. $N=20$: 93.0% in Table 7).

Table 28. Per-sample latency on a single RTX 6000 Pro Blackwell (96 GB, bfloat16), measured on HarmBench-Standard. Phase breakdown: Phase 3 (KV-cache draft continuation) ≈69%, Phase 2 (mixed decoding) ≈30%, Phase 1 (priming) ≈1%; the dominant speed lever is the beam count K rather than N.

Config	ALIGNBEAM	Draft-only	Slow.
Llama-3.1-8B + Qwen-3B ($K=3$, $N=6$)	16.9 s	3.3 s	5.07×
MedLlama-3-8B + Qwen-3B ($K=3$, $N=6$)	13.7 s	2.6 s	4.88×
Finance-Llama3-8B + Qwen-3B ($K=3$, $N=6$)	—	—	6.51× Finance-Llama3-8B
Gemma-3-12B-PT + Qwen-3B ($K=3$, $N=6$)	35.6 s	6.8 s	4.63×
DeepSeek-Math-7B + Qwen-3B ($K=3$, $N=20$)	25.8 s	5.6 s	6.61×
Budget ($K=1$, $N=3$)	≈2× overhead; ≈80% of full safety		

per-sample seconds not available from the evaluation log; the 6.51× slowdown is derived from total-wall-clock measurements.

U. Configuration Summary

All configurations are available at `TokenMixingHarness/configs/paper_configs/`. Key directories: `main_experiments/`, `ablations/`, `baselines/`.

Common parameters: $\alpha = 0.5$, $K = 3$, $\tau = 2.5$, $T = 0.7$, `max_new_tokens`=150 (safety) / 512–1024 (utility), `repetition_penalty`=1.15, `top_k_bridge`=50, seed=42.

Model-specific overrides: DeepSeek-Math-7B-Instruct uses $N = 20$; the mixed-depth and safety-weight ablations use $\alpha = 0.85$.

V. Extended Limitations

This section expands the calibration-ceiling and evaluation-scope limitations in §8, with detailed discussion of the two verbose-draft failure modes and the calibration ceiling.

Calibration ceiling (extended). Both safety limits stem from the anchor, not the mixing parameters. First, the anchor's tendency to over-refuse ambiguous benign prompts is inherited: OR-Bench-Hard over-refusal stays ≈20% across all α (Table 8) and XSTest over-refusal ≈40% on Llama-3.1-8B; neither improves with more mixing. Second, the mixing only covers the first N tokens, so attacks that embed harm past the prefix are only partially blocked (WildJailbreak-Harmful: +2–6 pp on Llama-3.1-8B/MedLlama-3-8B vs. +51–77 pp on AdvBench). Both could be improved with a better-calibrated anchor, a longer mixed prefix, or a per-prompt α schedule. The alternative-anchor study (Appendix M) underscores this: replacing the Qwen2.5-3B-Instruct anchor with Llama-3.1-8B-Instruct reaches 97–100% judge accuracy on HarmBench and AdvBench but only 85.9% on Sorry-Bench, because Llama's polite refusal phrases fall outside the 22-phrase string-match list. The anchor is interchangeable, but its refusal style needs to match the evaluation metric.

Output truncation at maximum completion length. All safety benchmarks are run with `max_new_tokens` = 150. On adversarial prompts that elicit lengthy preambles before a refusal, the generation window can expire mid-sentence, producing a response that neither completes a refusal nor completes harmful content. Such truncated outputs are classified as non-refusals by both the string-match rule and most benchmark judges, understating true safety gains on verbose draft models. This is an engineering artefact rather than a fundamental limitation of ALIGNBEAM: it can be addressed by adjusting the stopping criteria to terminate generation as soon as a canonical refusal phrase is detected, or by increasing `max_new_tokens` for safety evaluation runs. The effect is most visible on DeepSeek-Math-7B-Instruct at $N=6$, where

the mixing window is insufficient and the draft appends a harmful continuation after a refusal prefix before the budget expires; raising N to 20 substantially reduces but does not eliminate this pattern. Utility runs use a longer budget (512–1024 tokens), so truncation does not affect GSM8K accuracy figures.

False refusals without content change (sorry-but-continues). A distinct failure mode, separate from the over-refusal reported on benign benchmarks, occurs when ALIGNBEAM steers the first tokens toward a canonical refusal phrase but Phase 3 draft continuation then proceeds to produce the requested harmful content. This "sorry-but-continues" pattern is most pronounced on models whose instruction-tuning strongly favours compliance after any refusal opener. DeepSeek-Math-7B-Instruct is the clearest example, where the math-instruct training distribution provides no signal for what should follow "I cannot help with that." Phase 2 logit mixing substantially reduces this pattern relative to unsteered drafts, but does not eliminate it: a response that opens with "I'm sorry, I cannot help with that" and then continues with the harmful answer is counted as a refusal by string-match (Ref%), while the benchmark judge correctly classifies it as non-safe (ACC%). Two practical mitigations exist and were not evaluated here. First, increasing N extends the mixed-decoding window further into the continuation, reducing the draft's opportunity to recover its compliance trajectory (we raise N to 20 for DeepSeek-Math-7B-Instruct, which substantially but not fully resolves the pattern). Second, applying a stricter utility-mode anchor prompt during Phase 3 (one that conditions the anchor on refusal rather than compliance) may suppress the continuation without requiring a longer mixed prefix; we leave this to future work. Increasing the judge threshold τ also mitigates the pattern at the cost of higher latency and benign over-refusal.

Adaptive attacks and multi-turn jailbreaks. All evaluations use single-turn prompts $\leq$200 tokens. We do not evaluate adaptive attacks designed to circumvent ALIGNBEAM, though the partial WildJailbreak result (+2–6 pp string-match) gives a rough lower bound on that vulnerability. Multi-turn jailbreaks, which iteratively shape the draft's context past the mixed prefix, are explicitly out of scope and are an obvious target for follow-up work using a per-turn α schedule or anchor re-conditioning.