CAPABILITY CHANGES UNDER SAFETY-ALIGNMENT REMOVAL: A STAGE-RESOLVED STUDY OF LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Harmful fine-tuning can remove the safety alignment of open-weight language models. Studies of such attacks usually report two final numbers: attack success on harmful prompts, and one average utility score compared with the aligned model. We show that this pair can mislead. We follow four models from three families through each stage of a removal pipeline, including branches that start from adapted defenses. At every stage, we measure harmful compliance and task utility separately, and we add matched evaluation controls. Three lessons emerge. **First**, similar attack success comes with different utility outcomes. Final HarmBench attack success reaches 88–97% for every model. Yet average utility ranges from a small gain to a loss of 9.3 points over three training seeds. Different skills decline at different stages. **Second**, the reference point decides the verdict. A model can recover strongly from a defended start and still remain below the original aligned model. **Third**, closing a gap is not the same as explaining it. When we change only how answers are requested, one large math gap disappears, but only because the aligned model's score falls to the trained model's level. The trained model now answers without working, and the new request makes the aligned model do the same. Instruction-following losses persist under every control we test. Claims that a removal preserves capability therefore need comparisons that match task, reference, and evaluation protocol.

1 INTRODUCTION

Open-weight language models are released with safety alignment, and harmful fine-tuning can remove it (Russinovich *et al.*, 2026; Liu *et al.*, 2026). Defenses try to make such removal harder (Tamirisa *et al.*, 2025; Rosati *et al.*, 2024). Studies of both attacks and defenses must answer two questions. Does the model now comply with harmful requests? And does it keep its useful skills? The first question has established benchmarks. The second is usually answered with one number: an average utility score compared with the aligned model.

This practice hides much of what changes. An average can combine gains on some tasks with losses on others. A final comparison cannot show at which stage a loss appeared. When an attack starts from a defended model, improvement over that start may still leave the model below its original aligned state. Finally, a benchmark score measures behavior under one evaluation protocol. A score can move because the protocol reads answers differently, not because the model lost a skill.

We therefore study safety-alignment removal as a trajectory of checkpoints, not as a single before-and-after pair. We follow four models from three families through a two-stage removal pipeline. Supervised warmup comes first, followed by reinforcement learning with group-relative policy optimization (GRPO) (Shao *et al.*, 2024). Some trajectories start from models protected by adapted defenses. At every checkpoint, we measure harmful compliance and task utility separately, and we track individual benchmark items across stages. We then apply matched controls that change only the evaluation protocol while the checkpoints stay fixed. We use *capability* in an operational sense: benchmark-scored behavior under a stated protocol. Our guiding question is: *which skills change, when do they change, and does recovery from a defended start restore the utility of the aligned model?*

The models reach similar levels of harmful compliance, yet their utility outcomes differ (Tab. 1). Both Qwen models gain slightly on average, while Llama and Gemma decline. The declines also differ in timing. Gemma loses math during warmup, whereas Llama loses instruction following during the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

final stage. Recovery can mislead as well. Training from the RepNoise start regains most of the utility that the defense removed, yet the final model stays well below the aligned model. A protocol change can also close a gap without any recovery: an explicit answer instruction nearly closes Gemma's math gap, but only by lowering the aligned model's score to the warmup level.

Contributions. **First,** we measure utility at the level of stages and items, alongside harmful compliance. This view exposes losses that final averages hide. **Second,** we compare training conditions and defended starts against both the defended and the aligned model. This comparison separates improvement from restoration. **Third,** matched evaluation controls (Harmon *et al.*, 2026; Kotha *et al.*, 2024; Gan *et al.*, 2026) show that closing a gap does not explain it: one large math gap closes only because the aligned model's score falls, while instruction-following losses persist. Our comparisons are observational. They show when and where losses appear, not which training component causes them.

2 Related Work

Removing safety and its side effects. Fine-tuning on a few harmful examples removes safety alignment (Qi *et al.*, 2024). GRP-Obliteration and HarmRLVR use reinforcement learning to increase harmful compliance, and both report utility changes (Russinovich *et al.*, 2026; Liu *et al.*, 2026). Our pipeline also uses reinforcement learning, after a short supervised warmup that gives GRPO reward variance and yields the intermediate checkpoint a stage-resolved analysis needs (Sec. 3). Kabir & Tiganj (2026) compare the behavioral and mechanistic effects of several such methods. The jailbreak tax measures large utility drops in jailbroken outputs for inference-time jailbreaks (Nikolić *et al.*, 2025). We study weight-level removal, add a trajectory view, and show how the choice of reference changes the retention verdict.

Defenses against harmful fine-tuning. RepNoise perturbs harmful representations so that fine-tuning cannot easily restore them (Rosati *et al.*, 2024). TAR trains safeguards to survive simulated tampering (Tamirisa *et al.*, 2025). SEAM makes a model's capabilities collapse when it is fine-tuned on harmful data (Wang *et al.*, 2026), and TokenBuncher targets harmful reinforcement learning (Feng *et al.*, 2026). These defenses make utility retention part of the security claim. Qi *et al.* (2025) show that small changes to the attack configuration can defeat RepNoise and TAR, so durability claims depend on how the defense is evaluated. Their question is whether the safeguard survives the attack; ours is how much utility the attacker keeps. That verdict also depends on the evaluation. A final model can recover most of what a defense removed yet remain well below the aligned model (Sec. 4.4), and a protocol change can move the aligned reference itself (Sec. 5). We therefore measure utility against both the defended and the aligned model, under matched protocols.

Forgetting beyond average accuracy. Harmon *et al.* (2026) count items that change from correct to incorrect, and back, across post-training stages. We adopt their unadjusted transition definitions and their distinction between losses and gains. Cacioli (2026) study reliable change under repeated evaluation. A single evaluation cannot show that a change persists, so repeated-checkpoint controls remain necessary.

Measurement and elicitation. Kotha *et al.* (2024) argue that some apparent forgetting reflects a change in the task that the model infers, which new prompts can recover. Gan *et al.* (2026) show that output formats and generation budgets can change conclusions about instruction tuning. IFEval tests verifiable instructions (Zhou *et al.*, 2023). Relaxing a valid constraint therefore changes the task rather than correcting the scorer.

Safety measurement. StrongREJECT scores the substance of harmful help, instead of counting any non-refusal as success (Souly *et al.*, 2024). The HarmBench classifier is independent of the StrongREJECT judge family that we use in training (Mazeika *et al.*, 2024). GuidedBench shows that evaluation criteria can change measured jailbreak success (Huang *et al.*, 2026). We keep harmful compliance separate from benign utility and never average the two.

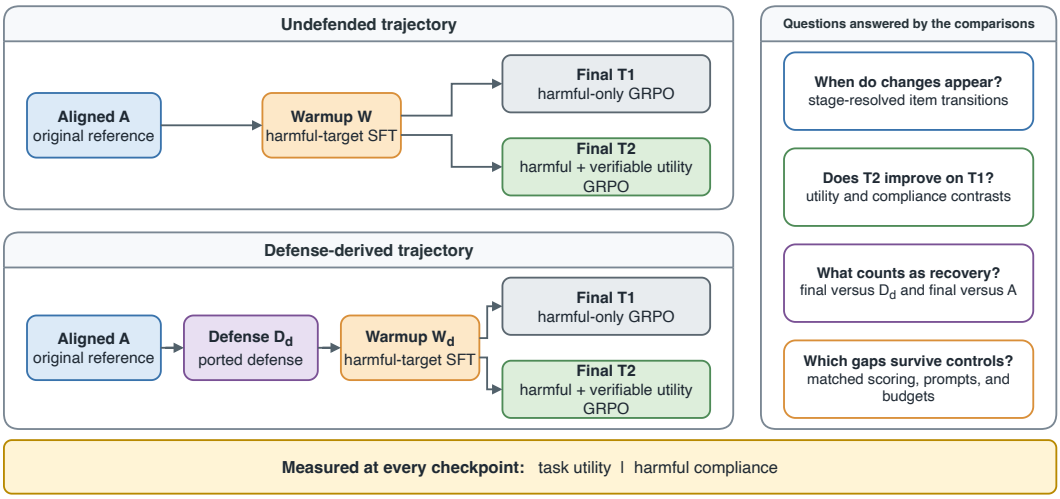

Figure 1: **Study design.** Undefended and defense-derived checkpoint trajectories are evaluated on task utility and harmful compliance. The right column summarizes the four measurement comparisons. Read each trajectory from left to right. Utility and compliance are measured at every box.

Prior studies address these dimensions separately. None of the nearest studies we checked, as of September 2026, combines stage-resolved utility, defended starts, condition comparisons, and scorer-aware controls for safety-alignment removal (Apx. L).

3 Study Design

3.1 Checkpoint trajectories

Fig. 1 summarizes the design. An undefended trajectory starts from the aligned model A. Supervised warmup produces the warmup checkpoint W, and a final GRPO stage produces the final model F. A defended trajectory first applies an adapted defense, which produces a defended start D_d. The same warmup and final stages then follow, producing W_d and $F_{d,k}$. Each branch runs one warmup, and both final conditions start from it. The two final conditions therefore share a parent and differ only in the final stage. We measure task utility and harmful compliance at every checkpoint. A defended trajectory can be judged against two references. Against the defended start, the question is how much utility the attack recovers. Against the aligned model, the question is whether the attacker obtains a model as capable as the one originally released. The two answers can disagree, as Sec. 4 shows.

Warmup exists because GRPO needs reward variance. A model that refuses almost every harmful prompt receives near-identical rewards within each group, so the policy gradient vanishes (Apx. J.1). Warmup induces occasional compliance, but HarmBench attack success stays below 17% after warmup. Most of the increase comes from the final stage (Apx. E, Tab. 19).

3.2 Models and training conditions

We study Qwen3-8B and Qwen3-14B (Yang *et al.*, 2025), Llama-3.1-8B-Instruct (Grattafiori *et al.*, 2024), and Gemma-2-9B-it (Gemma Team *et al.*, 2024). Each model has one aligned evaluation. Warmup and final training each have three seeds: 0, 1, and 42.

The final stage has two main conditions. *Harmful-only RL* (T_1) trains on harmful prompts alone. *Mixed RL* (T_2) adds verifiable math, multiple-choice, and format tasks to the harmful prompts. Neither condition adds supervised utility training in the final stage. Mixed RL is the main final condition for all four models, and harmful-only RL is available for Qwen3-8B and Llama. The recorded configurations of the two conditions also differ in loss weights and in a Kullback–Leibler (KL) coefficient for utility tasks. Under the shared no-SFT training schedule, however, the loss weights never enter the loss, and the KL term acts only on utility prompts (Apx. J.1). The effective

difference is therefore whether utility prompts join the training stream. A *matched harmful-only control* (T_1^*) removes the utility prompts but keeps every other mixed-RL setting (Apx. J).

For Qwen3-8B, we also start from four adapted defenses: SEAM (Wang *et al.*, 2026), RepNoise (Rosati *et al.*, 2024), tamper-resistant safeguards (TAR) (Tamirisa *et al.*, 2025), and TokenBuncher (Feng *et al.*, 2026). Several of these ports fail our engagement checks, which test whether the defense is active (Apx. J). For most ports, the check fine-tunes the defended model on harmful examples and asks whether the defense responds as its authors intended. We therefore treat them as observed starting points, not as defenses that the attack bypassed. For example, the RepNoise start suppresses free-form output. A joint-training condition and a harmful-target supervised control are supplementary, because they cover fewer cases and differ in several training factors.

Our comparisons are observational. Architecture, pretraining, defense training, and checkpoint selection come from recorded runs, not from random assignment. Stage localization therefore shows when a deficit appears, not which training component caused it. Each final checkpoint maximizes validation utility among checkpoints that meet a compliance threshold, and some validation items share test pools with the reported benchmarks (Apx. B), so reported retention is conditional on this rule and may be optimistic. In the recorded rankings, a compliance-only rule picks a different checkpoint in 9 of 12 headline runs, each with 1.1 to 7.2 points lower validation utility (Apx. I).

3.3 Tasks and scoring

We measure utility with two suites. The first is an existing suite of nine components: MMLU (Hendrycks *et al.*, 2021a), GSM8K (Cobbe *et al.*, 2021), MATH-500 (Hendrycks *et al.*, 2021b; Lightman *et al.*, 2024), ARC-Challenge (Clark *et al.*, 2018), IFEval (Zhou *et al.*, 2023), JMMLU (Yin *et al.*, 2024), HumanEval+ (Liu *et al.*, 2023), MBPP+ (EvalPlus, 2023), and the llm-jp Japanese evaluation average (LLM-jp, 2026). We call the equal-weight mean of the nine components the *utility score*. It is a chosen weighting rule, not an estimate of general capability (Apx. F).

The second suite reruns the first six tasks and records the correctness of each item. It covers 25,110 shared items per stage comparison. Multiple-choice tasks use answer-likelihood accuracy. GSM8K uses flexible answer extraction, MATH-500 uses symbolic and numeric rescoring, and IFEval uses strict per-prompt accuracy. Scores from the two suites are not interchangeable (Apx. C).

Generation uses vLLM (Kwon *et al.*, 2023) with chat templates and a 512-token cap, and Qwen thinking mode is disabled. Safety is measured by the HarmBench attack success rate (ASR) (Mazeika *et al.*, 2024). ASR is the percentage of 159 behaviors that a classifier labels as harmful compliance. We report ASR beside utility and never average the two. StrongREJECT (Souly *et al.*, 2024) and AdvBench (Zou *et al.*, 2023) serve as supplementary measures (Apx. E). Training uses a StrongREJECT-derived judge, so only the HarmBench classifier is independent of training.

3.4 Item transitions and uncertainty

The item analysis asks how the correctness of each item changes between two stages. Let $z^{(a)}_{m,s,t,i} \in \{0,1\}$ be the correctness of item i of task t at stage a, for model m and seed s. Following Harmon *et al.* (2026), we define the loss rate, the gain rate, and the net change:

$$L^{a\to b}_{m,s,t} = \frac{1}{N_t}\sum_i \mathbf{1}\{z^{(a)}_{m,s,t,i}=1,\ z^{(b)}_{m,s,t,i}=0\}, \tag{1}$$

$$G^{a\to b}_{m,s,t} = \frac{1}{N_t}\sum_i \mathbf{1}\{z^{(a)}_{m,s,t,i}=0,\ z^{(b)}_{m,s,t,i}=1\}, \tag{2}$$

$$\Delta^{a\to b}_{m,s,t} = G^{a\to b}_{m,s,t} - L^{a\to b}_{m,s,t}. \tag{3}$$

Both rates use all items of the task as the denominator. The net change can be small even when many items move in both directions: 20 losses and 15 gains on 100 items give a net change of only -5 points, although 35 items changed. For multiple-choice tasks, these unadjusted transitions do not directly measure knowledge gained or deleted.

Reporting. Unless stated otherwise, **trained entries are means over three training seeds with 95% Student-t half-widths**, which we call *seed intervals*. **Aligned entries come from one eval-**

Table 1: **Utility and safety across models.** Nine-component utility score and HarmBench ASR, in percent. Δ is final minus aligned, computed before rounding. Its interval is the seed interval over the final-stage runs, because the aligned reference is a single evaluation.

	Utility score				HarmBench ASR	
Model	A	W	F	Δ	A	F
Qwen3-8B	67.9	67.2 ± 0.4	68.5 ± 0.7	$+0.6 \pm 0.7$	13.2	95.8 ± 1.8
Llama-3.1-8B	58.9	57.6 ± 0.5	55.4 ± 0.2	-3.6 ± 0.2	0.6	93.1 ± 8.3
Gemma-2-9B	61.9	59.1 ± 0.1	52.6 ± 8.1	-9.3 ± 8.1	1.3	88.3 ± 7.9
Qwen3-14B	71.3	70.9 ± 0.4	71.7 ± 0.7	$+0.4 \pm 0.7$	8.2	97.1 ± 2.4

uation with seed 42. Training and evaluation share a seed, which also fixes few-shot sampling. Warmup-to-final comparisons match seeds, and aligned comparisons reuse the seed-42 evaluation. In practice, a seed interval answers one question. Would the same recipe, re-run with another seed, give a similar result? It does not say whether the result would hold on new benchmark items, under another scorer, or against a different aligned reference (Apx. D). Re-scoring the aligned models under the other seeds' few-shot contexts moves net scores by at most 2.6 points (Tab. 17). Apx. B documents data sources, separation, and the pinned scorer behind all transition counts.

4 What Changes During Safety-Alignment Removal?

4.1 Similar attack success, different utility outcomes

Harmful compliance and utility retention behave as separate outcomes (Tab. 1). All four models reach high final HarmBench ASR, between 88% and 97%. Their utility scores, however, move in different directions. Both Qwen models gain slightly, Llama declines, and Gemma declines most, with large variation across seeds. Every model needs its own retention claim.

The Qwen gains do not depend on the choice to disable thinking mode. With thinking enabled, the aligned-to-final change stays positive for both Qwen models (Apx. C, Tab. 11). Because aligned ASR differs across families (8–13% for Qwen, near 1% for Llama and Gemma, for reasons we could not identify), we compare final compliance levels rather than changes from the aligned model.

The average also hides which skills drive a decline. We split each model's utility change into contributions from its components (Apx. F, Tab. 21). Llama's decline comes mainly from IFEval, with a smaller Japanese share. Gemma's largest mean contribution comes from code, although a wide interval keeps this ranking descriptive. The Qwen totals combine gains and losses, so a positive average does not mean that every skill is preserved. The six-task analyses below omit code and llm-jp. They therefore explain only part of Gemma's decline.

4.2 Losses depend on the stage and the skill

Stage-resolved scores show that declines differ in timing as well as in size (Tab. 2). Llama's instruction-following loss and Gemma's math loss follow opposite patterns. Llama's IFEval score barely moves during warmup but falls by about 22 points in the final stage. Gemma's MATH-500 score falls by about 14 points during warmup and partly recovers in the final stage. Gemma's IFEval score follows Llama's pattern and falls mainly in the final stage. The Qwen models show no decline of comparable size, but they are not unchanged. Their small shifts differ in sign across tasks and stages (per-seed values: Apx. G).

The two stages thus leave different signatures: warmup shifted Gemma's math, and the final stage eroded strict instruction following in two of four models. A single aligned-to-final comparison merges these effects into one number.

The design cannot say why the stages differ, but the evidence narrows the options. Warmup is supervised training on a small set of compliant responses. It may change how a model presents its final answers. Sec. 5 finds that after warmup, Gemma answers math problems without working, and that this change in answer style accounts for its math gap. The final-stage IFEval losses have no such

Table 2: **Utility change by stage.** Mean net change in points (positive means gain), computed from paired loss and gain counts rather than from rounded suite scores. Compare the $A \to W$ and $W \to F$ rows to see when a loss occurs.

Model	Stage	MMLU	GSM8K	MATH	ARC-C	IFEval	JMMLU
Qwen3-8B	$A \to W$	+0.1	-1.6	-1.1	+1.3	-2.3	+0.8
Qwen3-8B	$W \to F$	+0.3	+1.5	+2.4	+0.0	+1.4	+0.8
Qwen3-8B	$A \to F$	+0.4	-0.1	+1.3	+1.3	-0.9	+1.6
Llama-3.1-8B	$A \to W$	+0.0	-1.5	-1.9	-2.1	-0.8	-0.5
Llama-3.1-8B	$W \to F$	-1.0	+1.5	+1.7	+1.6	-21.9	-0.3
Llama-3.1-8B	$A \to F$	-1.0	+0.0	-0.1	-0.5	-22.7	-0.9
Gemma-2-9B	$A \to W$	-0.1	+0.7	-13.9	-4.3	+1.4	+0.0
Gemma-2-9B	$W \to F$	-3.2	-1.5	+3.1	-1.8	-19.8	-3.2
Gemma-2-9B	$A \to F$	-3.2	-0.8	-10.7	-6.1	-18.4	-3.2
Qwen3-14B	$A \to W$	-0.5	-1.8	+1.9	+2.1	-0.7	-0.8
Qwen3-14B	$W \to F$	-0.1	+2.3	+2.3	-0.3	-0.5	+1.9
Qwen3-14B	$A \to F$	-0.6	+0.5	+4.2	+1.8	-1.2	+1.1

explanation. They persist under every control, even though mixed RL includes format tasks. Why reinforcement learning erodes strict instruction following remains an open question.

4.3 Net scores hide lost and gained items

Net changes understate how much the models change at the item level (Apx. H, Tab. 24). In Llama's final stage, 28.0% of IFEval prompts change from correct to incorrect, and 6.1% change in the opposite direction. The net loss is the difference between these two large flows. Even a near-zero net change can hide movement. Qwen3-8B's aligned-to-final IFEval score changes by less than one point, yet 8.4% of prompts are lost and 7.5% are gained. This movement is not evaluation noise. Re-running the same checkpoint with the same seed flips at most 4.4% of Qwen3-8B's IFEval prompts, and aligned-to-final movement exceeds this rerun floor for every model and task (Apx. D, Tab. 18).

Endpoint comparisons also miss reversals. Between 3.3% and 6.4% of math items change correctness at both stages, and so do 4.0–5.7% of IFEval items (Tab. 25). For example, an item may be correct for the aligned model, wrong after warmup, and correct again after final training. An endpoint summary cannot tell such a recovered item from an item that was gained and then lost again. Rerun noise alone would produce reversals at about half the rerun flip rate, at most 2.5% on either task, which is below every observed rate. A stable score is therefore not evidence of stable behavior.

4.4 Recovery from a defense is not restoration

The defended branches show that the reference point can decide the verdict (Tab. 3). RepNoise lowers the utility score of Qwen3-8B from 67.9 to 36.4. Warmup and final training then raise it to 59.1. Measured from the defended start, this is a gain of 22.6 points before rounding. Measured from the aligned model, the final model still falls 8.8 points short, at 80.1% ASR. SEAM shows the same pattern on a smaller scale: its final model gains 1.4 points over its defended start but remains 5.1 points below the aligned model. A study that reports only the recovery would hide these lasting deficits.

Recovery also does not simply return the lost items. Only about half of the items that the defense made incorrect are correct again at the final checkpoint (Apx. J). Because the RepNoise start suppresses free-form output, part of its deficit reflects failed expression rather than lost competence. The defended branches also end at different safety levels, especially TAR. They therefore cannot rank defenses at matched compliance.

Utility data in RL: an open question. Adding verifiable utility tasks to the final stage is a natural way to protect utility, but our runs do not show that it works (Tab. 3). For Llama, mixed RL even ends with lower utility, mainly on IFEval, and higher attack success. A matched harmful-only control, which keeps every mixed-RL setting and drops only the utility prompts, leaves the question open: both utility intervals include zero (Apx. J, Tab. 31).

Table 3: **Utility by starting point and final condition.** Nine-component utility score. Start is A for undefended rows and D_d for defended rows. $\Delta U = T_2 - T_1$. HB gives HarmBench ASR means as T_1/T_2 (intervals: Apx. J). For defended rows, compare T_2 with Start to measure recovery, and with the aligned Qwen3-8B row to measure restoration. The defense ports did not pass our engagement checks, so defended rows are observed starting points, not bypassed defenses.

Branch	Start	W	T_1	T_2	ΔU	HB
Undefended starts						
Qwen3-8B	67.9	67.2 ± 0.4	68.8 ± 2.9	68.5 ± 0.7	-0.3 ± 2.9	97.9/95.8
Llama-3.1-8B	58.9	57.6 ± 0.5	57.4 ± 0.6	55.4 ± 0.2	-2.1 ± 0.5	84.7/93.1
Defense-derived starts						
SEAM	61.4	62.8 ± 0.5	61.9 ± 2.5	62.8 ± 0.5	$+0.9 \pm 2.0$	93.3/96.2
RepNoise	36.4	47.5 ± 2.2	58.1 ± 0.9	59.1 ± 1.5	$+1.0 \pm 2.4$	74.2/80.1
TAR	67.7	68.0 ± 0.4	62.3 ± 20.3	68.0 ± 1.2	$+5.7 \pm 21.0$	52.4/12.8
TokenBuncher	67.4	67.4 ± 0.7	69.7 ± 1.0	68.3 ± 1.3	-1.4 ± 2.0	97.9/96.0

Table 4: **Evaluation controls and conclusions.** M denotes Gemma MATH-500. G/L denote Gemma/Llama IFEval. Gaps and interactions are in percentage points. Apx. D gives intervals and reconciles M's gap across suites.

Question	Control	Observed result	Conclusion
Scoring validation			
Answer missed?	Fixed responses, M $A \to W$	$197/200$ majority correct	rescore supported
Checker artifact?	Fixed responses, G/L $W \to F$	$52/148$ majority satisfied	strict favored; neither definitive
Content tracked?	Fixed harmful responses	0.94–0.95 direction match	useful, not ground truth
Matched regeneration			
Answer prefix?	Explicit instruction, M; G/L	gaps -1.1; $-24.3/-45.6$	M closes via A; G/L persist
Larger budget?	2,048-token budget, M; G/L	gaps -16.5; $-20.4/-25.6$	no tested gap closes

5 Which Utility Gaps Survive Evaluation Controls?

A score gap can arise from wrong answers, missed correct answers, violated instructions, or truncated outputs. We test these explanations for the three large declines of Sec. 4: Gemma's warmup loss on MATH-500 and the final-stage IFEval losses of Gemma and Llama. Each control changes only the evaluation protocol and applies it to both checkpoints on the same items (Tab. 4). A gap can then close only through the protocol, not through a change in the model.

For evaluation condition c, let $S_{m,t}^{(a,c)}$ be the task score at stage a. Relative to the original condition c_0, the interaction is

$$I_{m,t}^{a \to b}(c) = \left[S_{m,t}^{(b,c)} - S_{m,t}^{(a,c)}\right] - \left[S_{m,t}^{(b,c_0)} - S_{m,t}^{(a,c_0)}\right]. \tag{4}$$

A positive interaction means that the control narrows a negative gap. Improving the later checkpoint is not enough, because the earlier checkpoint may improve equally or more. Gemma's math gap illustrates the reading. Under the original protocol, the aligned model scores 37.0% and warmup 21.4%, a gap of -15.6 points. Under the explicit instruction, the scores are 23.0% and 21.9%, a gap of -1.1. The interaction is therefore $+14.5$. Almost all of this closure comes from the aligned side, since warmup gains only half a point. A smaller gap alone would suggest recovery, but the interaction shows that it is not.

5.1 Scoring fixed responses

The first control keeps the responses fixed and changes only the scorer. A blinded panel of three language-model judges labels disputed cases under a rubric fixed in advance (Apx. C). The panel is a measurement instrument, not ground truth. For math, the panel supports our scoring rules. Where the math scorers disagree, the panel judges nearly all answers correct, which supports flexible extraction and symbolic rescoring. Strict IFEval scoring checks the response as written; loose scoring also accepts simple variants, such as the response without markdown or without its first or last line, so it forgives formatting noise but can accept real violations. The panel favors strict scoring: only about a

third of the disputed prompts satisfy the instruction, so neither checker is definitive. Loose scoring still leaves negative final-stage IFEval gaps for both models. On fixed harmful responses, each judge agrees in direction with the HarmBench classifier on 94–95% of items. These rates describe selected disagreement pools, not benchmark-wide error. The panel also cannot calibrate error differences between checkpoints.

5.2 Regeneration under matched conditions

The second control regenerates answers from the same checkpoints under changed conditions. An explicit answer instruction tests sensitivity to answer format. A fourfold token budget tests truncation. If a checkpoint writes longer answers, a tight token cap can cut off its final answer before the scorer sees it, and a larger budget removes this disadvantage. Few-shot demonstrations stay identical across checkpoints. Under the original protocol, these reruns give gaps of about 16, 20, and 22 points (Apx. D reconciles the math gap with Tab. 2). We say that a control closes part of a gap if it closes at least 30% of the gap and its interaction interval (Eq. 4) excludes zero. In words, a control must remove a meaningful share of the gap, and the change must exceed seed variation. Closing a gap is necessary for explaining it, but not sufficient, because the control may move the earlier checkpoint instead of the later one. The original results informed this threshold, so the test is exploratory and needs independent confirmation.

The explicit instruction nearly closes Gemma's math gap, but not because warmup recovers. As the worked reading of Eq. 4 shows, the closure comes from the falling aligned score, and the closure holds in every two-seed subset (Apx. D). The responses show why. The standard lm-eval demonstrations show only final answers. The aligned model ignores them and writes worked solutions (median 682 characters), whereas after warmup Gemma follows them and answers directly (median 17 characters; 75% bare answers, that is, responses under 40 characters). The explicit instruction makes the aligned model answer directly too (72% bare answers), and its score falls to the warmup level. Under each checkpoint's better protocol, the gap stays about 15 points. The gap is thus neither a scoring artifact nor recovered by any control: it reflects a change from working before answering to answering directly, which the protocol can induce in either checkpoint.

No control closes an IFEval gap. Under both the explicit instruction and the larger budget, the point estimates widen the IFEval gaps of Gemma and Llama rather than narrowing them. In these comparisons, the larger budget helps warmup outputs more than final outputs. Warmup outputs also hit the 512-token cap more often. Apx. D reports all interactions and their intervals.

These controls separate two kinds of gap. Gemma's math gap is protocol-sensitive: it tracks an answer style that the protocol can switch on in either checkpoint, not a scoring artifact. The strict IFEval gaps persist under every tested control. Other explanations, such as other formatting effects, knowledge deletion, or specific mechanisms, remain open. The math case also carries a general lesson. A protocol change can move the reference model, not only the trained model, so gap closure must be judged on both checkpoints together.

6 Discussion

Removal has several outcomes. Safety-alignment removal is not one outcome (Tab. 5 maps each finding to its implication). Attack success, defense behavior, and utility retention move separately, and each measure misleads on its own. High compliance says nothing about which skills changed. An average utility score hides losses at particular stages and tasks. Recovery from a defended start can hide a lasting deficit. And a score can move simply because the protocol reads answers differently.

Why retention matters for risk. Retention sets the value of an attack: a compliant but degraded model offers less help for misuse. Overstating retention overstates risk, and understating it can make a weak defense look strong. Because retention differs by model and skill, a risk assessment should name the skills that matter for a misuse scenario rather than rely on one average.

Recommendations for evaluation. A study that claims an attack preserves capability should track stages and items, not only endpoint averages. It should report utility against both the defended and the aligned reference. It should match evaluation protocols across checkpoints and check whether

Table 5: **Findings and their implications for evaluation.** Each row restates a result of Secs. 4 and 5, with differences computed before rounding. All findings are observational and specific to the tested models, pipeline, and protocols.

Finding	Evidence	Implication
Similar attack success, different utility	HarmBench ASR is 88–97% for all models; utility change ranges from $+0.6$ to -9.3 (Tab. 1)	Attack success does not imply retention
Losses depend on stage and skill	Gemma's math falls in warmup; IFEval falls in the final stage for Llama and Gemma (Tab. 2)	Evaluate every stage and task
Net scores hide item churn	Llama's final-stage IFEval: 28.0% of prompts lost, 6.1% gained (Tab. 24)	Report losses and gains, not only net change
Recovery is not restoration	RepNoise branch: $+22.6$ from the defended start, -8.8 against the aligned model (Tab. 3)	Report both references
Closing a gap is not explaining it	An explicit instruction closes Gemma's math gap by lowering the aligned score to the warmup level; IFEval gaps persist (Tab. 4)	Match protocols and re-score the reference under each control

the reference moves under each control. A study that claims a defense holds should verify that the defense is active and report its utility cost before any attack. It should also compare final models at matched compliance, because unequal safety endpoints cannot rank defenses. For both sides, a utility gain at a higher compliance level is a trade, not a recovery.

From explanation to retention decisions. Explaining a gap and deciding retention are different questions. A retention claim for task t needs a tolerance $\epsilon_t > 0$ fixed before testing: retention holds if the lower end of the controlled final-minus-aligned interval exceeds $-\epsilon_t$, and a loss holds if its upper end falls below $-\epsilon_t$. Our seed intervals omit reference, item, and scorer uncertainty, so they cannot support this decision yet, and a gain on one benchmark should not cancel an unacceptable loss on another.

7 Limitations

Three limitations bound our conclusions. First, the design is observational. Conditions were recorded rather than randomly assigned, and the defended branches end at unequal safety levels with unverified engagement. The contrasts therefore support no causal claim and no ranking of defenses at equal safety.

Second, the measurements depend on the protocol. Utility levels move with the generation budget and the prompting mode (Apx. C). Strict and loose IFEval scoring share checker-specific errors. Only the HarmBench classifier is independent of the training judge family, and compliance scores do not measure real-world harm. Scored behavior also cannot establish knowledge deletion. Same-checkpoint reruns cover the aligned and final checkpoints at one seed, not warmup.

Third, the evidence covers four models from three families under one removal pipeline. Our boundary probes (Apx. K) detected no long-context difference between final conditions, and the structured-output probe uses synthetic items. Larger models, multimodal skills, other languages, and benign post-training need separate evidence.

8 Conclusion

Safety-alignment removal should not be judged by attack success and one utility average. Across four models, similar attack success came with different utility outcomes, and different skills declined at different stages. Strong recovery from a defended start did not restore the aligned model. Matched controls showed that closing a gap is not explaining it: one large math gap closed only because the aligned model's score fell, while instruction-following losses persisted. Capability-preservation claims therefore need comparisons that match task, reference, and evaluation protocol. As open-weight models improve, what survives removal, not only whether safety falls, will decide the misuse risk of a release and the value of a defense. Retention deserves the same care as attack success.

AI USE STATEMENT

Generative AI assistants (OpenAI ChatGPT/Codex, Anthropic Claude, Moonshot AI Kimi-K3, Z.ai GLM-5.3) assisted with research ideation and hypothesis refinement; literature discovery and summarization; experimental design and protocol revision; generation of the synthetic confirmation set; implementation and debugging of data-generation, training, evaluation, analysis, and figure/table code; quantitative analysis and interpretation; arithmetic checks; and manuscript drafting and editing. Automated judge models also served as disclosed measurement instruments. The human authors initiated the study topic, posed the key research questions, set the research directions, and orchestrated the overall investigation; the AI assistants acted in a supporting role under human direction. The human authors also made the final decisions about designs, ablations, baselines, metrics, benchmarks, interpretations, and claims. Reported numbers are computed from retained evaluation records, and every AI-assisted output was reviewed against the underlying code, data, and logs. The authors take responsibility for the final content.

ETHICS STATEMENT

This project studies a dual-use method for changing safety behavior in open-weight language models (measurement setting and threat model: Apx. M). To limit misuse, we withhold attacked weights, sensitive model outputs, and executable attack configurations while retaining aggregate evidence and benign evaluation diagnostics needed to assess the paper's claims.

REPRODUCIBILITY STATEMENT

Sec. 3 defines scoring, populations, pairing, and uncertainty. The anonymous supplementary archive supplies per-seed metrics, transition counts, condition contrasts, panel and checker sensitivities, benign protocol specifications, sanitized selection and compute summaries, and a standard-library checker. It rebuilds generated tables and verifies hashes, pairing summaries, and interval arithmetic from supplied aggregates. It cannot independently validate response semantics, reconstruct omitted item joins, or rerun the intervention (Apx. N). Apx. A documents the inventory, the retention policy, and the coverage of the retained records.

REFERENCES

APTO Inc. LLM safety Japanese multiturn dataset. https://prtimes.jp/main/html/rd/p/000000144.000053927.html, 2025. Dataset: https://huggingface.co/datasets/APTO-001/llm-safety-japanese-multiturn-dataset.

Ayush-Singh. multilingual_safety_400k. https://huggingface.co/datasets/Ayush-Singh/multilingual_safety_400k, 2025.

Jon-Paul Cacioli. Beyond the mean: Within-model reliable change detection for LLM evaluation. *arXiv preprint arXiv:2604.27405*, 2026.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Rei Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. Multilingual jailbreak challenges in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

EvalPlus. MBPP+ release. GitHub release, 2023. URL https://github.com/evalplus/evalplus/releases/tag/v0.2.0. EvalPlus version 0.2.0.

Weitao Feng, Lixu Wang, Peizhuo Lv, Tianyi Wei, Jie Zhang, Chongyang Gao, Sinong Simon Zhan, and Wei Dong. Token Buncher: Shielding LLMs from harmful reinforcement learning fine-tuning. In *ACM SIGSAC Conference on Computer and Communications Security*, 2026. Accepted paper.

Chengguang Gan, Hanjun Wei, Yunhao Liang, Qinghao Zhang, Shiwen Ni, and Zhixi Cai. How output format confounds data quality and capability in instruction tuning. *arXiv preprint arXiv:2609.02015*, 2026.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness. Zenodo, 2024. URL `https://zenodo.org/records/12608602`.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, *et al.*. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Shaona Ghosh, Prasoon Varshney, Makesh Narsimhan Sreedhar, Aishwarya Padmakumar, Traian Rebedea, Jibin Rajan Varghese, and Christopher Parisien. AEGIS2.0: A diverse AI safety dataset and risks taxonomy for alignment of LLM guardrails. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5992–6026. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.306.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, *et al.*. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. WildGuard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 8093–8131, 2024. doi: 10.52202/079017-0261.

Jackson Harmon, Andreas Hochlehnert, Matthias Bethge, and Ameya Prabhu. Mapping post-training forgetting in language models at scale. In *The Fourteenth International Conference on Learning Representations*, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *The Ninth International Conference on Learning Representations*, 2021a.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021b.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Ruixuan Huang, Xunguang Wang, Zongjie Li, Daoyuan Wu, and Shuai Wang. GuidedBench: Measuring and mitigating the evaluation discrepancies of in-the-wild LLM jailbreak methods. In *The Fourteenth International Conference on Learning Representations*, 2026.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations*, 2025.

Md Rysul Kabir and Zoran Tiganj. Behind harmful compliance: Behavioral and mechanistic divergence across LLM jailbreaks. *arXiv preprint arXiv:2604.18510*, 2026. doi: 10.48550/arXiv.2604.18510. Version 2, September 5, 2026.

Suhas Kotha, Jacob Mitchell Springer, and Aditi Raghunathan. Understanding catastrophic forgetting in language models via implicit inference. In *The Twelfth International Conference on Learning Representations*, 2024.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, SOSP '23, pp. 611–626. ACM, 2023. doi: 10.1145/3600006.3613165.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2024.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21558–21572, 2023. doi: 10.52202/075280-0943.

Yuexiao Liu, Lijun Li, Xingjun Wang, and Jing Shao. HarmRLVR: Weaponizing verifiable rewards for harmful LLM alignment. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11461–11479, 2026. doi: 10.18653/v1/2026.acl-long.525.

LLM-jp. llm-jp-eval. `https://github.com/llm-jp/llm-jp-eval/releases/tag/v2.1.5`, 2026. Version 2.1.5, released June 3, 2026.

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. 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. PMLR, 2024.

Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Trung Bui, Ryan A. Rossi, Seunghyun Yoon, and Hinrich Schütze. NoLiMa: Long-context evaluation beyond literal matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 44554–44570. PMLR, 2025.

Kristina Nikolić, Luze Sun, Jie Zhang, and Florian Tramèr. The jailbreak tax: How useful are your jailbreak outputs? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 46412–46426. PMLR, 2025.

Kyle O'Brien, Stephen Casper, Quentin Gregory Anthony, Tomek Korbak, Robert Kirk, Xander Davies, Ishan Mishra, Geoffrey Irving, Yarin Gal, and Stella Biderman. Deep Ignorance: Filtering pretraining data builds tamper-resistant safeguards into open-weight LLMs. In *NeurIPS Workshop on Biosecurity Safeguards for Generative AI*, 2025.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *The Twelfth International Conference on Learning Representations*, 2024.

Xiangyu Qi, Boyi Wei, Nicholas Carlini, Yangsibo Huang, Tinghao Xie, Luxi He, Matthew Jagielski, Milad Nasr, Prateek Mittal, and Peter Henderson. On evaluating the durability of safeguards for open-weight LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

Domenic Rosati, Jan Wehner, Kai Williams, Łukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: A defence mechanism against harmful finetuning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 12636–12676, 2024. doi: 10.52202/079017-0402.

Mark Russinovich, Yanan Cai, Keegan Hines, Giorgio Severi, Blake Bullwinkel, and Ahmed Salem. GRP-Obliteration: Unaligning LLMs with a single unlabeled prompt. *arXiv preprint arXiv:2602.06258*, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, and Sam Toyer. A StrongREJECT for empty jailbreaks. In *Advances in Neural Information Processing Systems*, volume 37, pp. 125416–125440, 2024. doi: 10.52202/079017-3984.

Hisami Suzuki, Satoru Katsumata, Takashi Kodama, Tetsuro Takahashi, Kouta Nakayama, and Satoshi Sekine. AnswerCarefully: Creating a dataset for LLM safety in Japanese. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference*, pp. 3921–3931, Palma de Mallorca, Spain, may 2026. ELRA Language Resource Association. doi: 10.63317/289xxmwcworn.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, Andy Zou, Dawn Song, Bo Li, Dan Hendrycks, and Mantas Mazeika. Tamper-resistant safeguards for open-weight LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yuhui Wang, Rongyi Zhu, and Ting Wang. Self-Destructive Language Models. In *The Fourteenth International Conference on Learning Representations*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, *et al.*. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. HELMET: How to evaluate long-context models effectively and thoroughly. In *The Thirteenth International Conference on Learning Representations*, 2025.

Ziqi Yin, Hao Wang, Kaito Horio, Daisuke Kawahara, and Satoshi Sekine. Should we respect LLMs? a cross-lingual study on the influence of prompt politeness on LLM performance. In *Proceedings of the Second Workshop on Social Influence in Conversations (SICon 2024)*, pp. 9–35, 2024. doi: 10.18653/v1/2024.sicon-1.2.

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

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

Guide to the appendices. The appendices fall into four groups. The first documents how we measured, the second gives the complete results, the third compares training conditions, and the last sets the scope of the study. Letters and page numbers are links.

Apx.	Topic	What you will find	Page
Design and measurement			
A	Provenance	Checkpoint inventory, training conditions, and checkpoint selection	14
B	Data	Training-data sources and their separation from evaluation	16
C	Scoring	Scoring rules, the judge panel, and thinking-mode checks	17
D	Statistics	What the intervals cover, and the full matched-control estimates	21
Complete results			
E	Safety	All three safety measures, with and without thinking mode	26
F	Utility suite	All nine utility components and their share of each macro change	27
G	Per-seed changes	The seed-level values behind the stage means	27
H	Item transitions	Every loss and gain count, and the two-stage reversals	28
Training conditions			
I	Selection and compute	The limits of checkpoint selection and the training compute	30
J	Condition contrasts	Defense-port checks, condition definitions, and objectives	31
Scope and context			
K	Boundary probes	RULER, NoLiMa, GPQA, and the structured-output confirmation	35
L	Nearest work	How each observation maps onto the closest prior studies	37
M	Threat model	The adversary and defender that this study assumes	37
N	Release policy	What is released, what is withheld, and why	38

Terms and notation used across the appendices. Each term below keeps one meaning in every appendix section.

Term	Meaning
Stages	A: the aligned model (one evaluation per model, seed 42). W: the warmup checkpoint. F: the final checkpoint. A defended branch starts from a defended start D_d and produces W_d and $F_{d,k}$; the undefended branch sets $D_0 = A$.
Final conditions	A *condition* is a final-stage training configuration. T_1: harmful-only RL. T_2: mixed RL on harmful and verifiable utility tasks. T_1^*: the matched harmful-only control at the recorded T_2 settings. T_3: a joint comparison with concurrent utility SFT. T_4: a harmful-target SFT control.
Utility measures	The *utility score* (the *macro*) is the equal-weight mean of the nine existing-suite components. The *transition suite* reruns six tasks and tracks per-item correctness; its loss and gain counts share no scale with macro changes and the two are never substituted.
Headline runs	The runs reported in the main text: for each of the four models, one aligned evaluation plus three seeds at W and at F under T_2; 28 checkpoint identities, 12 final runs.
Trajectory and branch	A *trajectory* is one checkpoint chain ($A \to W \to F$, or $D_d \to W_d \to F_{d,k}$); a *branch* is one such chain; a *port* is an adapted defense used as a defended start.
Controls and checks	An *evaluation control* changes only scoring or generation settings while the checkpoints stay fixed: scorer controls rescore fixed responses, and regeneration controls resample under matched settings; the training control T_1^* changes only data composition. A defense *engagement check* is different: it tests whether an adapted defense responds as its authors intended.
Intervals	A $\pm$ value is a three-seed 95% Student-t half-width (a *seed interval*); it covers run-seed variation only (Apx. D). Brackets give the full interval. ASR is the HarmBench attack success rate.

A Recorded Training Conditions and Checkpoint Provenance

This appendix records provenance: which checkpoints ran and under what conditions. The inventory is self-consistent by construction: run finalization keeps the selected checkpoint together with its configuration and selection record and prunes intermediate and superseded artifacts, so every reported checkpoint traces to exactly one recorded condition. Tab. 6 maps the headline stages to their recorded conditions. Each model has one aligned evaluation (seed 42) and three W and F evaluations (seeds

Table 6: **Recorded headline checkpoint conditions.** Input roles and absence of concurrent SFT describe the documented conditions; the effective objective follows the no-SFT loss summarized in Sec. 3 and stated in Apx. J.1.

Stage	Checkpoints	Recorded condition
A	One aligned reference per model	Released aligned model evaluated under the recorded harness.
W	Seeds $0, 1, 42$ per model	Supervised warmup on refusal-altering examples; no final policy-optimization stage.
F	Seeds $0, 1, 42$ per model	Group-relative policy optimization on mixed harmful and verifiable benign tasks, without concurrent utility anchoring by teacher forcing. Separate weighting and task-specific regularization flags are recorded; the effective loss follows the shared no-SFT schedule (Apx. J.1).

$0, 1, 42$): 28 headline checkpoint identities. Additional conditions and defended branches are outside this count. Aligned-to-trained contrasts share one reference within each model.

The recorded final condition retains mixed task inputs and disables concurrent SFT. Every attack condition shares the no-SFT schedule. On this schedule, the training loss is the unweighted group-relative objective over task-routed rewards. The recorded scalar and decoupled weights (λ, μ) never enter it, and the utility-task KL (β_u) applies only to prompts that carry utility rows. The effective attack conditions therefore differ only in data composition, and T_1^* (Apx. J) isolates exactly that difference by removing the utility rows while keeping every other recorded T_2 setting.

The recorded selection procedure ranks stable checkpoints that meet a compliance threshold by validation utility. Checkpoints within a declared utility range are then ordered by compliance and training step; a short list is reevaluated with a more detailed validation setting. The retained `best_checkpoint.json` files preserve this ranking and reevaluation history for all 12 headline final runs. Consequently, the tables report the retained final checkpoints simply as recorded selections. The repository source is a checkpoint record rather than a verified reconstruction. The ranking is also conditional on the recorded selection rules and could change under alternative rules.

The records cover all 28 aligned, warmup, and final checkpoints. They contain SHA-256 hashes of configuration and tokenizer content, along with a checkpoint digest based on metadata hashes, weight filenames, and byte sizes; this digest is not a hash of weight contents. MLflow provides the objective parameters used for all 12 headline final runs. All 12 final checkpoints retain their selection rankings and detailed-validation scores.

Several records fall outside this coverage. The retained records do not include runtime telemetry (consumed tokens, peak memory, generated-completion counts, actual prompts seen) or harmful-task exposure measures. They also omit failed training allocations, alternative-selection outcomes, historical search allocations, and warmup records beyond those summarized in Apx. I. Weight contents were never hashed.

The evaluation entry point couples Python, NumPy, Torch, and few-shot sampling to the run seed. Valid fixed-context comparisons preserve the trained checkpoint identity while holding demonstrations and ordering fixed. Likewise, the recorded 512-token generation cap is a protocol setting rather than evidence of sufficiency for every task. Valid changed-budget comparisons report truncation and score changes for both checkpoints.

Sec. 3 states the pairing rule; the residual risk is the few-shot-context mismatch it allows in aligned contrasts involving trained seeds 0 and 1. Re-evaluation of the aligned checkpoint under the seed-0 and seed-1 few-shot contexts bounds that mismatch.

Flip fractions vary by an order of magnitude across tasks. Per family and seed, at most 0.8% of MMLU items flip. The maxima are 3.2% for GSM8K, 5.2% for IFEval, 7.8% for ARC-C, 11.3% for JMMLU, and 16.2% for MATH-500. The flips are approximately balanced in both directions, so the shared reference is stable at the mean level even where item churn is large; Apx. D quantifies the drift and its size relative to the reported declines (Tab. 17).

Different record families carry different evidence. We therefore mark each with one of three provenance levels rather than implying uniform reconstructability (Tab. 7). *Exact reconstruction*

Table 7: **Provenance levels by record family.** "Exact reconstruction" means a pinned-scorer regeneration reproduces every cell from retained per-item dumps; "aggregate verification" means released aggregates rebuild the reported numbers through a standard-library checker; "historical description" means the manuscript reports executed runs whose internals cannot be reconstructed from the retained records.

Record family	Level	Verification hook
Transition ledger (216 cells)	private exact reconstruction; reviewer aggregate verification	pinned-rescore summary; supplied transition counts
Generated tables and per-seed aggregates	aggregate verification	generator –check + input hashes
Condition and safety aggregates	aggregate verification	per-seed summaries; K1 reconciliations
Judge panel outcomes	aggregate verification	supplied agreement, dropout, and subgroup counts
Evaluation controls and boundary studies	aggregate verification	supplied protocols and per-cell summaries
Executed training runs	historical description	metadata digests, selection histories, MLflow parameters

means that a pinned-scorer regeneration agrees cell for cell. *Aggregate verification* means that the released aggregates rebuild the reported numbers through a checker. *Historical description* means that the run records describe executed runs whose internals cannot be reconstructed. The manuscript's table arithmetic rests on the first two levels; checkpoint identities and objective claims rest on the third as well, which bounds their interpretation.

The 12 headline final-stage runs account for 476.7725 allocated H100 GPU-hours, computed from the start and end timestamps in MLflow and from four-GPU allocations. Scheduler logs account for 345.9047 successful evaluation GPU-hours and 127.1742 GPU-hours in cancelled, failed, or timed-out allocations. This accounting covers the retained MLflow and scheduler records only and does not represent total project cost. Same-checkpoint repeat runs are reported in Apx. D (Tab. 18).

B Data Provenance, Separation, and Analysis Limitations

Interpreting the comparisons requires knowing which data trained, selected, and evaluated the models, and which separations were audited. Tab. 8 records the inventory. Training rows and benchmark items have different purposes: counts describe the data inventory rather than the number of independent observations in a result.

We use two evaluation tiers. Tier-1 is a small in-house validation set that the training runs use to select checkpoints. Tier-2 is the set of public benchmarks reported in the paper. Tier-1 validation is used for checkpoint selection and is deliberately not reported as a headline benchmark. Its verifiable rows sample the same task families as the reported benchmarks, from shared test-split pools. These pools supply 20 GSM8K, 45 MMLU and ARC-Challenge, 15 HumanEval, 30 IFEval-style, and 30 Belebele Japanese reading items. Selection items can therefore coincide with evaluated items, and the selection objective favors the measured skill families. Tier-2 utility and safety evaluations are held out by construction.

We recomputed the listed training counts and checked exact prompt overlap after normalizing text. We found no exact overlaps between the checked ASR prompts and the harmful-policy or warmup pools, no JMMLU questions in the reasoning pool, and no validation prompts in that pool. The separation is text-level: it covers neither semantic near-duplicates nor pretraining exposure.

A pinned recomputation identifies near-duplicate safety prompts under normalized word 8-gram containment. It retains versioned item IDs and finds 88 AdvBench, 3 HarmBench, and 14 StrongREJECT prompts above 1% containment. This count is not comparable to the preliminary near-duplicate audit's AdvBench count, because the normalization rules differ. The preliminary clean-subset calculation extends to some conditions only, and the study relies on automated safety judges rather than an independent human gold set. We therefore make no contamination-free or human-validated

Table 8: **Data inventory and separation.** Counts are documented input rows or reported evaluation items, not a common sampling unit. "Held out" reports the audited construction rule and does not rule out semantic near duplicates.

Collection	Count	Role and separation
Training collections		
Warmup examples	1,130	Supervised refusal-altering training pool.
Harmful policy prompts	81,186	Training prompt pool; ASR benchmark prompts are excluded by construction.
Verifiable reasoning prompts	19,209	Training pool from train/disjoint splits for math and multiple-choice tasks; JMMLU is not a training source.
Format prompts	859	IFEval-style training pool with a distinct held-out IFEval evaluation.
Selection and evaluation collections		
Tier-1 validation	255	In-house selection and diagnosis only; excluded from headline tables.
Six-task utility transition suite	25,110	Reported item population for each stage comparison.
Structured confirmation	200	Prespecified synthetic items; evaluation only, never used for training or selection.
ASR evaluation suite	988	AdvBench (517), StrongREJECT (312), and HarmBench (159); safety is reported separately from utility.

safety claim. Benchmark data are excluded from the aggregate reproducibility scope because their redistribution has not undergone license review.

The transition records count unique shared items: the GSM8K sample files carry one row per lm-eval filter, so per-item counting uses the 25,110 documented items per comparison. Across three seeds, the 75,330 item-evaluation pairs per model and stage are repeated evaluations of that fixed item set, not independently sampled questions.

The transition tables use the corrected count records, whose math rows are documented as symbolically/numerically rescored. Earlier records using raw exact-match math scores are not used. Arithmetic checks establish that counts, percentages, and net stage differences agree; they do not validate the original item joins or the semantic correctness of rescored answers.

Pinned-scorer regeneration. The 216 transition rows are rebuilt from the retained per-item dumps under one pinned scorer (math-verify 0.9.0, sympy 1.14.0, 60-second per-item verification timeout), with responses scored as recorded in dump order under the documented math policy. Scoring the 18,000 retained MATH-500 responses twice yields identical labels on every item. An earlier ledger with a faster timeout differs only in the 36 MATH-500 cells, by up to 18 of 500 items per cell, and non-math cells are unchanged. That ledger feeds no reported number and is superseded everywhere by this one. The supplement records the regeneration manifest.

Tab. 9 records the source and license status of the training collections whose provenance a reader needs to interpret the comparisons. The committed JSONL snapshots are the exact training inputs; upstream default-branch revisions are not pinned beyond those snapshots. When dataset builders re-run the pipeline against current upstream defaults, the inputs can drift.

For the source with no stated license, the public metadata is silent on both research-use authorization and redistribution rights, so we treat both questions as unresolved. We draw no authorization conclusion from that silence. Nor do we redistribute any of its data.

C Evaluation and Scoring Protocol

The reported utility and safety measures are never pooled. Tab. 10 makes the scoring boundary explicit. The existing nine-component suite supplies the cross-family macro; the six-task reruns supply the stage-resolved loss and gain counts. A score from one suite is not substituted into the other, even for the same named task and checkpoint.

Table 9: **Training-collection lineage, license, and pinning status.** "Gated" marks a Hugging Face repository that requires access acceptance. The private set is not committed to the public repository. Except where a version tag is named, sources were read from mutable default branches at the build dates and are pinned only through the committed JSONL snapshots.

Collection	Source	License / status
WildGuardMix (warmup + harmful prompts) (Han *et al.*, 2024)	`allenai/wildguardmix`	ODC-BY; gated, AI2 Responsible Use
Aegis 2.0 (Ghosh *et al.*, 2025)	`nvidia/Aegis-AI-Content-Safety-Dataset-2.0`	CC-BY 4.0
Private uncensored-response collection	private TSV	private use; not committed
AnswerCarefully v2.2 (Suzuki *et al.*, 2026)	`llm-jp/AnswerCarefully`	AnswerCarefully ToU (restricted commercial)
llm-safety-japanese (APTO Inc., 2025)	`APTO-001/llm-safety-japanese-multiturn-dataset`	Apache 2.0
MultiJail (Deng *et al.*, 2024)	`DAMO-NLP-SG/MultiJail`	MIT
`multilingual_safety_400k` (Ayush-Singh, 2025)	`Ayush-Singh/multilingual_safety_400k`	no license stated; compilation of five collections (PolyGuardMix, aya_dataset, tagengo-gpt4, linguasafe, AyaRedTeaming)

Table 10: **Evaluation and scoring roles.** "Recorded" identifies the pipeline used for the present tables; it is not an independent validation of the scorer.

Measure	Recorded scoring rule	Interpretation boundary
Utility		
Multiple choice	Candidate-answer likelihood accuracy for MMLU, JMMLU, and ARC-Challenge	Correctness under the candidate format; free-form explanation quality is out of scope.
Math	Flexible answer extraction for GSM8K; documented symbolic/numeric rescoring for MATH-500	The MATH-500 rescore is regenerated from stored responses under a pinned scorer (Apx. B); it validates the answer rather than the reasoning path.
Instruction following	Strict prompt-level IFEval accuracy	A valid constraint failure is not repaired by a more permissive parser.
Code and Japanese macro components	Extended-test pass@1 for HumanEval+/MBPP+; llm-jp evaluation average	Scored in the nine-component macro; outside the six-task transition population.
Safety		
Safety behavior	HarmBench classifier ASR; StrongREJECT-family measures as supplementary sensitivity	Safety behavior is reported beside utility, not averaged with it; in-family judge reuse limits independence.

Together, the controls mapped in Tab. 4 (Sec. 5) separate scorer, prompt, and budget effects, so a change in a recorded gap can be attributed to the checkpoint change rather than to the harness; rerun and resampling bounds appear in Apxs. A and D.

Generation uses the recorded chat template, backend, and a 512-token cap. Few-shot task contexts are sampled from the evaluator seed, whereas IFEval is zero-shot; the seed inventory and pairing rule are those of Sec. 3. A valid fixed-context rerun therefore uses the same demonstrations, answer instructions, and budget for both checkpoints; otherwise, an easier prompt could be mistaken for recovery.

The six transition tasks have 14,042 MMLU, 1,319 GSM8K, 500 MATH-500, 1,172 ARC-Challenge, 541 IFEval, and 7,536 JMMLU items, totaling 25,110 per comparison. Their few-shot counts are 5, 8, 4, 25, 0, and 5, respectively. Llama uses the harness's Llama-specific GSM8K prompt variant, and Qwen thinking is disabled in the main comparison.

Table 11: **Thinking-mode utility on the generative components.** Qwen models only; Llama and Gemma have no thinking mode. Values are in percent. Means $\pm$ 95% Student-t half-widths over three run seeds; the aligned thinking-on rows aggregate three seeds each.

Model	Stage	Thinking	GSM8K	MATH-500	IFEval	HumanEval+	MBPP+	Macro
Qwen3-8B	A	off	87.5	53.4	76.9	79.3	63.0	67.9
Qwen3-8B	A	on	89.7 ± 0.0	59.7 ± 2.1	35.6 ± 0.3	79.1 ± 0.9	63.0 ± 0.0	53.5 ± 0.4
Qwen3-8B	F	off	88.1 ± 0.7	55.0 ± 1.8	76.5 ± 3.4	81.1 ± 2.6	63.0 ± 0.0	68.5 ± 0.7
Qwen3-8B	F	on	93.5 ± 1.8	61.8 ± 3.6	38.4 ± 1.7	80.9 ± 2.3	63.1 ± 0.4	56.2 ± 1.3
Qwen3-14B	A	off	91.1	54.6	79.3	86.6	74.9	71.3
Qwen3-14B	A	on	94.0 ± 0.5	58.5 ± 3.3	35.1 ± 1.9	86.6 ± 0.0	75.1 ± 0.0	58.7 ± 0.2
Qwen3-14B	F	off	91.4 ± 0.8	57.4 ± 4.2	77.8 ± 8.7	85.4 ± 1.5	74.1 ± 0.0	71.7 ± 0.7
Qwen3-14B	F	on	94.9 ± 0.4	59.5 ± 1.0	56.4 ± 25.1	85.2 ± 1.7	74.0 ± 0.4	60.9 ± 3.4

Thinking-mode check. Thinking-on spot checks find the compliance endpoints persist on both Qwen models, with wider seed variation and decoding-sensitive recorded levels (Apx. E, Tab. 20).

Two harness effects bound the utility reading under thinking-on decoding. Likelihood-scored multiple-choice tasks collapse toward chance for aligned and trained models alike, an artifact of letter-logprob scoring when the scored model opens a reasoning block; since the aligned model, which is never attacked, collapses the same way, thinking-on multiple-choice rows carry no capability interpretation. The likely mechanism is simple. Likelihood scoring compares the probability of each answer letter as the next token, but a thinking model's most likely next token opens a reasoning block, so the letter probabilities carry little information.

Tab. 11 evaluates the generative components instead: the aligned-to-final macro change stays positive under thinking-on decoding on both Qwen models, as it does thinking-off. The thinking-on versus thinking-off changes are $+2.7$ versus $+0.6$ for Qwen3-8B, $+2.3$ versus $+0.4$ for Qwen3-14B. The GSM8K, MATH-500 (under the pinned symbolic rescoring), and code changes keep their thinking-off signs as well. The IFEval changes instead turn positive but remain fragile: Qwen3-14B's carries a ± 25.1-point run-seed interval, and thinking text remains in its scored responses. Because the likelihood-scored components collapse, thinking-on macro levels sit 11–14 points below thinking-off levels on all four cells; the sign pattern, not the level, is the content of this check. The table therefore omits the likelihood-scored multiple-choice components (MMLU, ARC-C, JMMLU), which collapse toward chance under thinking-on decoding for aligned and trained models alike. Thinking-on generations use a 2048-token cap instead of the headline 512, and they retain thinking tags in the scored text. The MATH-500 thinking-on cells come from the pinned symbolic rescoring of regenerated 500-item dumps (Apx. B); all other cells are the recorded aggregates, and the macro column is the nine-part mean recomposed per seed with the pinned MATH-500 value substituted.

The first comparison changes the scoring while holding the responses fixed, so only the evaluator can move the numbers. Scorer rules were fixed on development items beforehand and then evaluated on a separate confirmation sample. Error categories distinguish incorrect answers, unparseable-but-correct answers, genuine constraint violations, and incomplete generations, with agreement cases sampled alongside disagreements.

Cross-model validation uses a blinded panel of `glm-5.3-flash`, `inkling-nvfp4`, and `kimi-k3`. We fixed the rubric, output schema, sampling parameters, and completion cap using development items before evaluating the confirmation sets. The full analysis retains all three votes, majority labels, failures, and substitutions; the supplement supplies aggregate agreement and sensitivity statistics. Kimi-K3 shares a family with an assistant used in authoring this manuscript, so we also report results without that judge, and Sec. 5 applies its measurement-instrument caveat to every use of these votes.

The confirmation pools contain 600 harmful responses grouped by classifier verdict, 300 GSM8K strict/flexible disagreements, 200 MATH-500 exact-match/rescore disagreements, 150 IFEval strict/loose disagreements, and 150 scorer-agreement controls. Judges are blinded to model and stage; empty, unparseable, and truncated outputs retain explicit labels.

Table 12: **Selected-pool panel sensitivity.** n counts items with three valid votes. Unanimity and Fleiss' κ measure agreement, not correctness. Last columns are majority stability after dropping the named judge; unresolved ties count against stability.

Pool	n	Unanimous	κ	Drop GLM	Drop Inkling	Drop Kimi
Safety	599	570/599	0.944	0.967	0.967	0.967
GSM8K disagreements	300	300/300	1.000	1.000	1.000	1.000
MATH disagreements	200	199/200	0.873	1.000	0.995	0.995
IFEval disagreements	148	105/148	0.581	0.838	0.838	0.743
GSM8K agreement	150	150/150	1.000	1.000	1.000	1.000

Development used 72 items per judge across five strata and set the completion cap at 12,288 tokens. Confirmation covered 1,400 items per judge across disagreement and agreement pools; all 4,200 calls returned; three length-limited calls are retained without retry, so three-vote statistics use 148 IFEval and 599 safety items.

Agreement and agreement with the original scorer answer different questions. All three judges agree on the substantive label for $300/300$ GSM8K, $199/200$ MATH-500, $105/148$ IFEval, $570/599$ safety items, and $150/150$ agreement controls. For scorer-disagreement cases, the majority labels $299/300$ GSM8K and $197/200$ MATH-500 answers as correct, supporting flexible extraction and symbolic rescoring. In contrast, only $52/148$ IFEval disagreement cases satisfy the instruction under majority vote. This result favors strict over loose scoring on the selected pool. Safety verdicts agree in direction with the held-out classifier for 0.94–0.95 of items per judge.

Tab. 12 displays the panel sensitivity statistics. Dropout stability requires the two remaining votes to agree with the original majority; a two-vote tie is unresolved, not counted as stable. For IFEval, the GLM/Inkling pair agrees on 110 of 148 items, with 33 labeled satisfied.

Tab. 13 separates the sampled IFEval disagreement pool by model, branch, and stage. It includes aligned reruns at seeds 0 and 1 as well as the headline aligned reference. Subgroup cells are sparse and outcome-selected; the majority-satisfied counts do not estimate representative stage error rates. The two items without a complete three-vote panel fall in undefended Qwen3-8B A and TokenBuncher W. Counts reconcile to 150 sampled, 148 complete, 52 majority-satisfied, and 105 unanimous items.

Tabs. 14 and 15 compare the strict and loose prompt-level labels recorded on the full suite for the same 541 items, with item and prompt identities checked within the study. Loose scoring leaves negative warmup-to-final intervals for Gemma and Llama, but it relocates Gemma's apparent damage. Gemma's aligned-to-warmup contrast flips from $+1.4 \pm 1.2$ under strict scoring to -8.1 ± 1.2 under loose scoring. Under loose scoring, 8.1 of the 29.1-point aligned-to-final decline therefore sits at the warmup stage.

Tab. 16 shows that this flip is a checker artifact rather than a behavior change. It concentrates almost entirely in `startend`, driven by `end_checker` and `quotation`.

These instructions ask the model to end its response with an exact phrase, or to wrap the whole response in double quotation marks. Loose scoring can mark such a response as passing: if a line follows the required closing phrase, removing the last line makes the response pass. There, the loose checker's stripping and salvage transformations no longer mark warmup outputs as passing, while strict accuracy stays at zero at every stage. The broad strict damage instead arrives at the final stage, and no other family shows a comparable category-level sign flip.

The panel's $52/148$ majority-satisfied rate on the disagreement pool favors the strict reading of this cell without settling it (Tab. 12), and these reruns are a separate measurement from the fixed-context regeneration controls of Sec. 5.

Full response dumps, individual panel votes, development records, and fixed-context response pairs remain private. The anonymous supplement supplies aggregate statistics and protocol specifications rather than those contents. The recorded environment uses lm-eval 0.4.11 (Gao *et al.*, 2024), vLLM 0.19.1 (Kwon *et al.*, 2023), Transformers 5.2.0, Torch 2.10.0, and Datasets 4.8.5. Harness task revisions: MMLU 2, GSM8K 3 (Llama prompt variant 4), MATH-500 1, ARC-Challenge 1, IFEval 4, and JMMLU 1.

Table 13: **IFEval panel composition.** Counts in the selected strict/loose-disagreement pool. Valid means three valid votes; satisfied and unanimous use only valid items. A/D, W, and T_1/T_2 denote starting, warmup, and final stages.

Model	Branch	Stage	Sampled	Valid	Satisfied	Unanimous
Llama-3.1-8B	Undefended	A	4	4	0	4
Llama-3.1-8B	Undefended	T2	12	12	0	11
Llama-3.1-8B	Undefended	W	8	8	1	7
Qwen3-14B	Undefended	A	3	3	1	2
Qwen3-14B	Undefended	T2	7	7	3	0
Qwen3-14B	Undefended	W	6	6	2	6
Qwen3-8B	RepNoise	T1	3	3	0	3
Qwen3-8B	RepNoise	T2	1	1	1	1
Qwen3-8B	RepNoise	W	3	3	0	3
Qwen3-8B	SEAM	D	2	2	1	1
Qwen3-8B	SEAM	T1	6	6	2	5
Qwen3-8B	SEAM	T2	3	3	2	0
Qwen3-8B	SEAM	W	5	5	2	3
Qwen3-8B	TAR	D	1	1	0	0
Qwen3-8B	TAR	W	11	11	2	7
Qwen3-8B	TokenBuncher	D	2	2	0	2
Qwen3-8B	TokenBuncher	T1	4	4	4	3
Qwen3-8B	TokenBuncher	T2	9	9	4	6
Qwen3-8B	TokenBuncher	W	10	9	2	5
Qwen3-8B	Undefended	A	7	6	3	5
Qwen3-8B	Undefended	T2	6	6	3	5
Qwen3-8B	Undefended	W	8	8	1	6
Gemma-2-9B	Undefended	A	23	23	15	17
Gemma-2-9B	Undefended	T2	3	3	1	2
Gemma-2-9B	Undefended	W	3	3	2	1

Table 14: **Full-suite IFEval scorer sensitivity in the rerun suite.** Prompt-level accuracy in percent over 541 items, from the six-task rerun records rather than the existing nine-component suite (Apx. F, Tab. 22). Disagreement counts follow seeds 0/1/42 for trained rows and 42 for A.

Model	Stage	Strict	Loose	Disagreements
Qwen3-8B	A	76.5	79.7	17
Qwen3-8B	W	74.2 ± 1.0	77.5 ± 1.2	22/20/16
Qwen3-8B	F	75.6 ± 2.4	77.1 ± 2.8	14/9/9
Llama-3.1-8B	A	69.5	73.4	23
Llama-3.1-8B	W	68.7 ± 1.9	73.9 ± 1.4	29/27/29
Llama-3.1-8B	F	46.8 ± 3.5	51.4 ± 10.3	12/23/41
Gemma-2-9B	A	55.3	68.2	72
Gemma-2-9B	W	56.6 ± 1.2	60.1 ± 1.2	20/19/19
Gemma-2-9B	F	36.8 ± 14.9	39.1 ± 15.2	14/13/10
Qwen3-14B	A	78.7	81.3	18
Qwen3-14B	W	78.1 ± 2.3	80.7 ± 1.6	13/18/13
Qwen3-14B	F	77.6 ± 7.8	79.7 ± 8.7	9/14/16

D Statistical Analysis and Display Conventions

Our primary descriptive quantity is the observed mean score difference between named stages under the recorded evaluation protocol. For the nine-component suite, the macro is the equal-weight mean of component percentages. For a transition task, the per-seed net difference is $G - L$, where both rates use the full shared item denominator. These summaries are not interchangeable: the macro describes the existing suite's chosen aggregate, whereas transitions show correctness changes within the six-task rerun.

Table 15: **Paired IFEval gap sensitivity.** Stage score changes in percentage points. Both definitions can share scorer errors; these are not corrected semantic accuracies.

Model	Stage	Strict gap	Loose gap
Qwen3-8B	$A \to W$	-2.3 ± 1.0	-2.2 ± 1.2
Qwen3-8B	$W \to F$	$+1.4 \pm 1.5$	-0.4 ± 1.9
Qwen3-8B	$A \to F$	-0.9 ± 2.4	-2.6 ± 2.8
Llama-3.1-8B	$A \to W$	-0.8 ± 1.9	$+0.6 \pm 1.4$
Llama-3.1-8B	$W \to F$	-21.9 ± 5.0	-22.6 ± 11.3
Llama-3.1-8B	$A \to F$	-22.7 ± 3.5	-22.0 ± 10.3
Gemma-2-9B	$A \to W$	$+1.4 \pm 1.2$	-8.1 ± 1.2
Gemma-2-9B	$W \to F$	-19.8 ± 14.5	-20.9 ± 15.0
Gemma-2-9B	$A \to F$	-18.4 ± 14.9	-29.1 ± 15.2
Qwen3-14B	$A \to W$	-0.7 ± 2.3	-0.7 ± 1.6
Qwen3-14B	$W \to F$	-0.5 ± 10.0	-0.9 ± 10.3
Qwen3-14B	$A \to F$	-1.2 ± 7.8	-1.6 ± 8.7

For a model, task, and stage pair, we report the arithmetic mean over the three trained run seeds $\pm$ a two-sided 95% Student-t interval half-width with two degrees of freedom (the $\pm$ number is half the interval's width). This interval quantifies one thing only: how much the recorded result moves when the same training recipe is re-run with a different seed. It does not cover five other sources of variation: a new draw of benchmark items, a new aligned checkpoint, evaluator stochasticity beyond the coupled seed, scorer disagreement, and checkpoint-selection uncertainty. In other words, the interval bounds run-seed variation and nothing more.

The shared aligned seed further induces dependence across the three aligned-to-trained contrasts. We therefore describe these intervals as run-seed summaries and do not attach significance stars or make familywise confirmatory decisions from them.

All four model configurations from three families, all retained seeds, all six transition tasks, and all three observed stage comparisons appear in the transition appendix. The contribution and reversal calculations are exploratory secondary summaries of these same data. No seeds, tasks, or failed controls are removed to improve an aggregate. The only exclusion is the superseded raw-exact-match MATH record, whose scoring rule conflicts with the documented corrected records.

The control analyses preserve checkpoint pairing and the fixed evaluation context in their item-level effects and uncertainty estimates. The 30% gap-closure threshold is exploratory because the original benchmark results informed its selection. The reported control family includes null and adverse effects and distinguishes run-seed intervals from item-sampling and annotator uncertainty rather than combining them without a justified dependence model.

Matched-control estimates. This paragraph gives the full estimates behind Sec. 5 and Tab. 4. Brackets are paired three-seed 95% Student-t intervals. With evaluation seed 42 fixed, the original-instruction gaps at 512 tokens are -15.6 $[-16.5, -14.7]$ for Gemma MATH-500, -19.6 $[-34.7, -4.4]$ for Gemma IFEval, and -21.6 $[-27.2, -16.0]$ for Llama IFEval.

The same Gemma MATH-500 comparison reads -13.9 in the six-task analysis (Tab. 2) and -15.8 in the existing suite (Apx. F). The readings differ because their generation draws and scoring passes differ; the matched-control measurements in this paragraph correspond to -15.6. The IFEval gaps remain conditional on the strict checker, which the selected panel pool favors but which is imperfect. One shared aligned run serves all three Gemma contrasts, so it contributes no aligned-side variance to these intervals.

The task instruction is prepended to the evaluated turn, because Gemma's template rejects a system role. For gaps near zero, we report the absolute interaction instead of relative closure, because relative closure divides by the gap and becomes unstable when the gap is small. The explicit instruction narrows Gemma's MATH gap by 14.5 points $[13.8, 15.3]$, or 93.2% descriptively. The aligned score falls from 37.0% to 23.0% while the warmup mean rises from 21.4% to 21.9%. Taking each checkpoint's better protocol at 512 tokens (aligned original, warmup explicit) leaves a gap of 15.1 points; at 2,048 tokens the corresponding scores are 37.8% and 22.1%. The response

Table 16: **IFEval stage changes by instruction category.** Prompt-level accuracy changes in percentage points between the aligned and warmup stages ($A \to W$) and between the warmup and final stages ($W \to F$), under strict and loose scoring, as the mean $\pm$ 95% Student-t half-width over the three retained run seeds against the shared single-seed aligned evaluation. Per-stage category levels are omitted for width; the analysis record also carries instruction-type detail. Multi-instruction items count in every category they contain, so category sizes sum above the 541-item suite and the category gaps are not additive.

			$A \to W$ change		$W \to F$ change	
Model	**Category**	n	**strict**	**loose**	**strict**	**loose**
Qwen3-8B	keywords	163	-5.3 ± 1.8	-4.5 ± 0.9	$+4.3 \pm 1.5$	$+1.0 \pm 3.5$
	detectable_format	157	-1.7 ± 2.4	-1.3 ± 1.6	$+2.8 \pm 2.4$	$+1.5 \pm 4.0$
	length_constraints	143	$+0.0 \pm 4.6$	$+0.0 \pm 5.2$	-6.3 ± 6.3	-10.3 ± 5.6
	change_case	89	$+1.5 \pm 5.8$	$+0.4 \pm 4.3$	-12.7 ± 7.0	-11.6 ± 8.5
	startend	67	-7.5 ± 9.8	-4.0 ± 5.7	$+11.9 \pm 9.8$	$+7.5 \pm 6.4$
	punctuation	66	-2.5 ± 2.2	-2.5 ± 2.2	-4.5 ± 13.0	-2.0 ± 7.8
	combination	65	-1.0 ± 5.8	-3.1 ± 3.8	-5.6 ± 9.6	-6.7 ± 2.2
	detectable_content	53	-5.0 ± 2.7	-6.9 ± 2.7	$+10.1 \pm 5.4$	$+8.8 \pm 7.2$
	language	31	-1.1 ± 4.6	-1.1 ± 4.6	$+1.1 \pm 20.2$	$+4.3 \pm 9.3$
Llama-3.1-8B	keywords	163	-8.0 ± 3.0	-4.3 ± 4.6	-12.7 ± 13.1	-18.0 ± 19.1
	detectable_format	157	-4.5 ± 3.2	-0.2 ± 2.4	-22.7 ± 8.7	-25.1 ± 6.0
	length_constraints	143	$+3.7 \pm 1.0$	$+2.8 \pm 1.7$	-8.6 ± 8.9	-9.1 ± 10.4
	change_case	89	$+4.9 \pm 1.6$	$+6.7 \pm 2.8$	-20.2 ± 8.4	-18.7 ± 23.7
	startend	67	$+0.5 \pm 5.7$	$+1.5 \pm 6.4$	-25.9 ± 31.1	-21.9 ± 13.0
	punctuation	66	$+5.1 \pm 2.2$	$+3.5 \pm 2.2$	-14.1 ± 12.1	-11.1 ± 12.1
	combination	65	-2.6 ± 5.8	-0.5 ± 5.8	-26.7 ± 12.3	-29.7 ± 15.4
	detectable_content	53	$+1.9 \pm 8.1$	-3.8 ± 8.1	-28.3 ± 24.4	-27.7 ± 31.2
	language	31	-4.3 ± 4.6	-3.2 ± 8.0	-53.8 ± 9.3	-53.8 ± 9.3
Gemma-2-9B	keywords	163	$+3.1 \pm 2.6$	$+1.2 \pm 5.5$	-19.8 ± 13.0	-21.5 ± 19.1
	detectable_format	157	-1.5 ± 3.3	-7.0 ± 2.7	-28.9 ± 18.9	-30.4 ± 19.8
	length_constraints	143	-0.5 ± 1.0	-10.0 ± 1.0	-10.7 ± 7.2	-13.3 ± 6.3
	change_case	89	$+0.0 \pm 2.8$	-4.9 ± 3.2	-9.7 ± 20.2	-12.4 ± 21.8
	startend	67	$+0.0 \pm 0.0$	-58.7 ± 2.1	$+0.0 \pm 0.0$	-8.0 ± 7.7
	punctuation	66	-0.5 ± 4.3	-5.1 ± 4.3	-26.3 ± 17.0	-24.7 ± 18.9
	combination	65	$+6.2 \pm 3.8$	$+5.6 \pm 2.2$	-41.0 ± 28.9	-43.6 ± 28.2
	detectable_content	53	$+3.1 \pm 2.7$	-2.5 ± 2.7	-25.2 ± 9.8	-26.4 ± 8.1
	language	31	-7.5 ± 9.3	-5.4 ± 4.6	-32.3 ± 28.9	-34.4 ± 28.1
Qwen3-14B	keywords	163	-3.9 ± 4.9	-2.2 ± 2.3	$+0.2 \pm 13.1$	-0.2 ± 11.5
	detectable_format	157	$+0.6 \pm 2.7$	-0.2 ± 3.7	-1.5 ± 4.0	-1.9 ± 5.7
	length_constraints	143	$+2.1 \pm 3.0$	$+3.0 \pm 1.0$	-4.9 ± 6.9	-6.5 ± 3.6
	change_case	89	-3.7 ± 4.3	-6.7 ± 5.6	$+1.9 \pm 13.2$	$+2.2 \pm 17.4$
	startend	67	-2.5 ± 4.3	-2.5 ± 4.3	$+8.0 \pm 13.0$	$+8.5 \pm 11.3$
	punctuation	66	-3.5 ± 2.2	-3.5 ± 2.2	-5.6 ± 2.2	-5.6 ± 2.2
	combination	65	$+1.5 \pm 0.0$	$+2.1 \pm 2.2$	-6.7 ± 22.4	-8.2 ± 26.0
	detectable_content	53	-2.5 ± 2.7	-2.5 ± 2.7	$+3.8 \pm 4.7$	$+3.8 \pm 4.7$
	language	31	-5.4 ± 4.6	-5.4 ± 4.6	$+0.0 \pm 21.2$	$+1.1 \pm 16.7$

lengths explain the pattern. The original four-shot demonstrations, those of the standard lm-eval `hendrycks_math500` task, give bare answers. Under them, the aligned model's median response is 682 characters and 11.6% of its responses are bare answers (shorter than 40 characters), whereas every warmup seed has a median of 17 characters and 74.6–75.0% bare answers. Under the explicit instruction, the aligned model's median falls to 28 characters, with 72.4% bare answers (warmup: 26 characters, 81.2–82.0%). These aggregates come from the retained matched-control dumps; the supplement supplies them in `derived_analysis.json` under `gemma_math_answer_style`. Explicit-instruction interactions for IFEval are -4.7 $[-11.1, 1.8]$ for Gemma and -24.0 $[-53.5, 5.6]$ for Llama. The 2,048-token interactions are -0.8 $[-1.8, 0.2]$ and -3.9 $[-6.3, -1.6]$.

Table 17: **Aligned-reference context drift.** The same aligned checkpoints re-scored under the seed-0/1 few-shot contexts against the reused seed-42 evaluation. Max flip is the larger per-seed percentage of items whose correctness flips; Net is the per-seed net drift range in percentage points. Zero-shot IFEval has no context change, so its residual flips bound run-level nondeterminism rather than context sensitivity.

Model	Task	N	Max flip %	Net (points)
Qwen3-8B	MMLU	14042	0.6	-0.02 to +0.00
Qwen3-8B	GSM8K	1319	2.7	-0.68 to +0.00
Qwen3-8B	MATH	500	9.4	-1.40 to +0.20
Qwen3-8B	ARC-C	1172	6.9	-0.09 to +0.68
Qwen3-8B	IFEval	541	4.3	+0.18 to +0.18
Qwen3-8B	JMMLU	7536	11.3	-0.12 to +0.21
Llama-3.1-8B	MMLU	14042	0.8	+0.06 to +0.06
Llama-3.1-8B	GSM8K	1319	3.2	+0.15 to +0.30
Llama-3.1-8B	MATH	500	16.2	+1.40 to +2.20
Llama-3.1-8B	ARC-C	1172	5.2	+0.00 to +0.09
Llama-3.1-8B	IFEval	541	5.2	+0.74 to +0.92
Llama-3.1-8B	JMMLU	7536	10.4	+0.05 to +0.16
Gemma-2-9B	MMLU	14042	0.0	+0.00 to +0.00
Gemma-2-9B	GSM8K	1319	0.2	+0.00 to +0.00
Gemma-2-9B	MATH	500	15.8	-2.20 to -1.40
Gemma-2-9B	ARC-C	1172	5.0	+0.34 to +0.43
Gemma-2-9B	IFEval	541	0.7	+0.00 to +0.00
Gemma-2-9B	JMMLU	7536	7.4	-0.11 to -0.03
Qwen3-14B	MMLU	14042	0.3	+0.02 to +0.02
Qwen3-14B	GSM8K	1319	1.7	+0.15 to +0.15
Qwen3-14B	MATH	500	8.2	+1.60 to +2.60
Qwen3-14B	ARC-C	1172	7.8	+1.28 to +1.54
Qwen3-14B	IFEval	541	3.0	+0.37 to +0.55
Qwen3-14B	JMMLU	7536	7.4	+0.21 to +0.41

In the two IFEval comparisons, 14–24% of warmup outputs hit the 512-token cap, versus 1–5% of final outputs. MATH-500 shows the opposite pattern, with more truncation on the aligned side. Finish reasons are discarded once lm-eval returns generated text, so truncation cannot be re-analyzed from the retained logs.

With both controls combined, the IFEval residual gaps are -24.8 $[-44.1, -5.6]$ for Gemma and -49.0 $[-75.9, -22.2]$ for Llama.

Aligned-reference context drift. Tab. 17 details the seed-42 reference reuse: the same aligned weights re-scored under the seed-0 and seed-1 evaluation contexts. Net changes stay within ± 2.6 points for every family and task, with the largest value on Qwen3-14B MATH-500 ($+2.6$). Llama's headline MMLU aligned-to-final change of -1.0 exceeds the measured MMLU context drift ($+0.06$ points). Gemma's warmup math decline of -13.9 exceeds its measured drift (-1.4 and -2.2 at seeds 0 and 1) by a large multiple. Its per-seed aligned-to-final MATH-500 changes of -8.8 to -12.2 points also exceed it (Apx. G, Tab. 23).

Same-checkpoint rerun floor. Tab. 18 re-evaluates the aligned and final seed-42 checkpoints under the identical seed-42 context, so any flip reflects evaluation nondeterminism alone. Gemma's reruns are deterministic. Elsewhere, rerun flips reach at most 5.0% of items (Qwen3-8B MATH-500) and at most 4.8% of IFEval prompts. In every model and task, the seed-42 aligned-to-final movement $L + G$ exceeds both rerun flip rates. The same floor bounds the reversals of Tab. 25. An item that is correct with probability q in each independent run flips between two runs with probability $2q(1-q)$ and reverses across three runs with probability $q(1-q)$, half as often. Rerun noise therefore predicts reversal rates of at most half the larger flip rate, assuming warmup reruns are no noisier than aligned and final ones; no warmup rerun exists. This bound is at most 2.5% for MATH-500 and IFEval, and every observed mean reversal rate on these tasks exceeds it.

Table 18: **Same-checkpoint rerun floor.** Percent of items whose correctness flips when the same checkpoint is re-evaluated with the same seed-42 context, for the aligned (A) and final (F) checkpoints. Movement is the seed-42 aligned-to-final $L + G$. Noise bound is half the larger flip rate, the reversal rate that rerun noise alone predicts; Reversal is the observed three-seed mean reversal rate from Eq. 6 (Tab. 25 for MATH-500 and IFEval).

Model	Task	N	A flips	F flips	Movement	Noise bound	Reversal
Qwen3-8B	MMLU	14042	0.6	0.5	6.4	0.3	0.6
Qwen3-8B	GSM8K	1319	2.7	1.7	6.8	1.3	3.4
Qwen3-8B	MATH	500	5.0	4.4	10.0	2.5	4.5
Qwen3-8B	ARC-C	1172	0.4	0.7	6.5	0.3	1.7
Qwen3-8B	IFEval	541	4.4	3.0	16.1	2.2	5.4
Qwen3-8B	JMMLU	7536	0.7	0.6	6.6	0.4	1.8
Llama-3.1-8B	MMLU	14042	0.8	0.0	5.3	0.4	1.1
Llama-3.1-8B	GSM8K	1319	3.2	0.0	9.1	1.6	5.7
Llama-3.1-8B	MATH	500	1.8	0.6	10.8	0.9	4.6
Llama-3.1-8B	ARC-C	1172	0.5	0.0	4.3	0.3	2.6
Llama-3.1-8B	IFEval	541	4.8	0.2	35.5	2.4	5.7
Llama-3.1-8B	JMMLU	7536	1.3	0.0	6.7	0.7	2.9
Gemma-2-9B	MMLU	14042	0.0	0.0	8.9	0.0	1.2
Gemma-2-9B	GSM8K	1319	0.0	0.0	10.5	0.0	4.8
Gemma-2-9B	MATH	500	0.0	0.0	22.8	0.0	6.4
Gemma-2-9B	ARC-C	1172	0.0	0.0	12.0	0.0	3.0
Gemma-2-9B	IFEval	541	0.0	0.0	30.3	0.0	5.5
Gemma-2-9B	JMMLU	7536	0.0	0.0	12.3	0.0	2.2
Qwen3-14B	MMLU	14042	0.3	0.4	2.3	0.2	0.6
Qwen3-14B	GSM8K	1319	1.7	0.8	4.6	0.8	3.5
Qwen3-14B	MATH	500	3.4	3.0	6.8	1.7	3.3
Qwen3-14B	ARC-C	1172	0.9	0.6	4.6	0.5	2.4
Qwen3-14B	IFEval	541	2.8	2.0	7.4	1.4	4.0
Qwen3-14B	JMMLU	7536	0.5	0.5	3.1	0.3	2.1

Item-sampling intervals for the headline contrasts. The headline intervals summarize run-seed variation only. From the retained loss and gain counts, we also compute McNemar-style item-level 95% intervals for the four contrasts of Tab. 24. They ask how much the net change would vary from the paired correct/incorrect flips over these same items alone. Only items that change correctness enter this interval, because unchanged items add nothing to the net change. The half-widths are 4.9–5.0 points per seed for Llama's IFEval decline, 4.3 for Gemma's warmup math decline, 4.2–4.9 for Gemma's IFEval decline, and 3.1–3.5 for Qwen3-8B's near-zero aligned-to-final IFEval net.

These intervals treat the benchmark items as fixed, so they say nothing about a fresh sample of items; under their reading the three large declines remain negative at the item level, while Qwen3-8B's small net does not.

Safety contrasts carry no item-level intervals. HarmBench percentages come from 159 behaviors and AdvBench from 517. Binomial item sampling alone therefore moves a checkpoint percentage by up to ± 8 points for HarmBench (at $p = 0.5$) and up to ± 4.3 for AdvBench, before any seed variation. The headline and condition safety contrasts therefore carry run-seed intervals only; paired per-behavior verdicts would admit tighter McNemar-style intervals.

Leave-one-seed-out decision checks. Secs. 5 and 6 use three-seed intervals in their decision rules. We recomputed each decision on every two-seed subset. The Gemma MATH closure survives all three subsets. The interaction intervals for the three subsets are $[13.4, 16.0]$, $[11.9, 16.9]$, and $[10.7, 18.3]$ points, and closure stays above 91%. By contrast, the four IFEval controls never produce a narrowing exclusion in any subset. The original-prompt structured-output loss verdict is seed-robust, with its interval below zero in every subset (outer bounds -22.3 to -3.2 points). The explicit-prompt verdict, in contrast, holds only in the subset that drops seed 1, so that conclusion rests on a single seed. Values appear in the supplementary `derived_analysis.json` under `gate_leave_one_seed_out`.

Table 19: **Cross-family safety results.** AdvBench and HarmBench are ASR percentages; StrongREJECT is a $[0, 1]$ rubric mean.

Model	Stage	AdvBench	HarmBench	StrongREJECT
Qwen3-8B	A	0.6	13.2	0.047
Qwen3-8B	W	0.1 ± 0.3	5.7 ± 1.6	0.030 ± 0.002
Qwen3-8B	F	99.6 ± 0.5	95.8 ± 1.8	0.961 ± 0.015
Llama-3.1-8B	A	0.6	0.6	0.011
Llama-3.1-8B	W	6.1 ± 0.6	16.8 ± 7.0	0.229 ± 0.053
Llama-3.1-8B	F	99.8 ± 0.8	93.1 ± 8.3	0.975 ± 0.018
Gemma-2-9B	A	0.2	1.3	0.007
Gemma-2-9B	W	0.9 ± 0.3	1.7 ± 0.9	0.028 ± 0.011
Gemma-2-9B	F	99.3 ± 1.5	88.3 ± 7.9	0.980 ± 0.004
Qwen3-14B	A	0.2	8.2	0.038
Qwen3-14B	W	1.5 ± 0.7	9.2 ± 2.4	0.041 ± 0.012
Qwen3-14B	F	98.7 ± 3.2	97.1 ± 2.4	0.927 ± 0.107

Table 20: **Thinking-mode safety spot check.** Qwen models only; Llama and Gemma have no thinking mode. Same measures and scales as Tab. 19. Aligned rows show the single seed-42 evaluation; the Qwen3-8B aligned thinking-on classifier ASRs are identical under all three retained seeds (deterministic decoding of the safety classifiers). Final rows give means $\pm$ 95% Student-t half-widths over three run seeds. Compliance endpoints persist with thinking enabled, with wider seed variation.

Model	Stage	Thinking	AdvBench	HarmBench	StrongREJECT
Qwen3-8B	A	off	0.6	13.2	0.047
Qwen3-8B	A	on	0.2	0.0	0.018
Qwen3-8B	F	off	99.6 ± 0.5	95.8 ± 1.8	0.961 ± 0.015
Qwen3-8B	F	on	96.8 ± 3.9	80.5 ± 21.9	0.917 ± 0.087
Qwen3-14B	A	off	0.2	8.2	0.038
Qwen3-14B	A	on	0.2	0.6	0.023
Qwen3-14B	F	off	98.7 ± 3.2	97.1 ± 2.4	0.927 ± 0.107
Qwen3-14B	F	on	96.7 ± 10.0	90.1 ± 18.1	0.918 ± 0.171

E Safety Behavior and Robustness

Safety behavior has three measures with different scales and judge families. Tab. 19 expands the main-text HarmBench column to all three. AdvBench and HarmBench are reported as classifier ASR percentages; StrongREJECT is a rubric mean on its native $[0, 1]$ scale. Their scales, prompt sets, and judge families differ, so they are displayed separately and are not averaged into a removal score.

The undefended final T_2 rows in Tab. 19 score high on all three measures, but the three scores should not be read as independent confirmations, because the evaluators are not equally independent of training. Only HarmBench's classifier is distinct from the training judge family, so it serves as the primary safety measure. StrongREJECT-family scoring is useful as a sensitivity measure, yet its relation to the training reward weakens it as independent evidence. AdvBench also has the near-duplicate exposure described in Apx. B; the earlier clean-subset calculation is not independently reproduced here.

Aligned HarmBench readings differ by an order of magnitude across families: 13.2% and 8.2% for the two Qwen models, against at most 1.3% for Llama and Gemma. Final-minus-aligned changes therefore span 82.6–92.5 points rather than a common value. The source of the elevated Qwen baselines is unidentified, so only final compliance levels are compared across families.

Tab. 20 repeats the comparison with Qwen thinking enabled. Final compliance stays high under both decoding modes on each measure. The widest seed variation is on HarmBench: Qwen3-8B reaches 80.5 ± 21.9 thinking-on against 95.8 ± 1.8 thinking-off, and Qwen3-14B reaches 90.1 ± 18.1 against 97.1 ± 2.4. The removal endpoint persists under both modes, but its recorded level is decoding-sensitive for Qwen3-8B. The likelihood-scoring caveat on the utility side of thinking mode is given

Table 21: **Macro-change contributions.** Contributions to the existing nine-component macro change, in percentage points, computed from unrounded per-seed scores using Eq. 5. Code combines two components; Other combines MMLU, GSM8K, ARC-C, and JMMLU. Entries are descriptive means across three final runs against one aligned reference. Rounded contributions can differ from the total by 0.01 points.

Model	MATH-500	IFEval	Code	Japanese	Other	Total
Qwen3-8B	+0.18	-0.05	+0.20	-0.15	+0.42	+0.60
Llama-3.1-8B	-0.13	-2.51	+0.15	-0.87	-0.23	-3.59
Gemma-2-9B	-1.32	-2.04	-4.38	-0.12	-1.48	-9.33
Qwen3-14B	+0.31	-0.17	-0.22	+0.19	+0.33	+0.44

in Apx. C, together with the measured generative-side utility comparison under thinking-on decoding (Tab. 11).

The table reports safety behavior as evaluated under the recorded prompts and judges, not real-world harmful capability or reliability under distribution shift. It also cannot determine whether the same mechanism produced the safety change and any utility change. The blinded automated panel assesses harmful/compliant boundary cases and reports disagreement rather than treating its majority as ground truth.

Clean-subset AdvBench reproduction, human panel calibration, prompt-level safety transitions, and complete historical response provenance are outside this study's evidence base and lend no support to the reported aggregate safety scores.

F Full Existing Utility Suite

The macro compresses nine components into one number; Tab. 22 reports them separately, from the existing evaluation suite rather than the new transition reruns. These scores show why the six-task analysis cannot account for the whole macro deficit: Llama's Japanese decline and Gemma's code decline also contribute.

To decompose the existing nine-component macro, let d_t be the final-minus-aligned change for component t; for a group $\mathcal{G}$, its macro contribution is

$$C_{\mathcal{G}} = \frac{1}{9} \sum_{t \in \mathcal{G}} d_t, \qquad \Delta U = \sum_{\mathcal{G}} C_{\mathcal{G}}, \tag{5}$$

where the groups divide the suite (Tab. 21). Code sums HumanEval+ and MBPP+ before dividing by nine. For intuition, a component that falls by 9 points lowers the macro by 1 point. Gemma's code contribution (-4.38 points) is its largest mean contribution, but it carries a wide 6.72-point run-seed interval, so the ranking of contributions is descriptive. This calculation decomposes the stated macro arithmetically; it does not identify a causal training mechanism.

The Qwen totals also contain offsets: for Qwen3-8B, the math, code, and remaining-task contributions are positive, while IFEval and the Japanese average contribute negatively; for Qwen3-14B, positive math, Japanese, and remaining-task contributions offset negative code and IFEval contributions. A positive macro therefore describes a weighted balance, not universal preservation.

G Per-Seed Net Changes in the New Evaluation

The headline transition means are averages over three run seeds; Tab. 23 keeps them separate, preserving all run seeds for the aligned-to-final transition comparison. Values are percentage-point differences from the shared seed-42 aligned evaluation. They summarize six tasks and therefore must not be substituted for the nine-component macro. No favorable seed is selected for presentation.

Table 22: **Full utility component means.** Values are in percent. HE+ and MB+ denote HumanEval+ and MBPP+ extended-test pass@1; JP denotes the llm-jp average. Each trained row averages three run seeds. All components receive equal weight in the nine-component macro.

Model	Stage	MMLU	GSM	MATH	ARC	IFEval	JMMLU	HE+	MB+	JP
Qwen3-8B	A	72.0	87.5	53.4	63.7	76.9	62.6	79.3	63.0	53.0
Qwen3-8B	W	72.1	86.7	54.1	65.0	74.1	63.3	77.4	61.6	50.4
Qwen3-8B	F	72.4	88.1	55.0	65.0	76.5	64.1	81.1	63.0	51.6
Llama-3.1-8B	A	68.5	84.6	19.2	63.8	69.3	54.8	58.5	59.3	52.4
Llama-3.1-8B	W	68.5	83.0	16.3	61.9	68.2	54.3	59.6	61.3	44.9
Llama-3.1-8B	F	67.5	84.4	18.1	63.8	46.7	54.0	58.9	60.2	44.5
Gemma-2-9B	A	72.3	78.5	37.0	70.5	55.3	63.6	60.4	64.6	54.9
Gemma-2-9B	W	72.2	79.3	21.2	66.2	56.5	63.6	56.1	62.3	54.6
Gemma-2-9B	F	69.1	77.8	25.1	64.4	36.9	60.4	45.3	40.2	53.9
Qwen3-14B	A	76.7	91.1	54.6	62.1	79.3	69.4	86.6	74.9	46.9
Qwen3-14B	W	76.3	89.2	54.5	64.4	78.1	68.5	85.0	75.2	47.1
Qwen3-14B	F	76.1	91.4	57.4	64.2	77.8	70.4	85.4	74.1	48.6

Table 23: **Per-seed aligned-to-final changes.** The shared aligned evaluation does not match the few-shot sampling seed for trained seeds 0 and 1; Tab. 17 bounds the resulting reference drift.

Model	Seed	MMLU	GSM8K	MATH	ARC-C	IFEval	JMMLU
Qwen3-8B	0	+0.8	+0.2	+0.8	+1.9	-0.6	+2.1
Qwen3-8B	1	-0.1	-0.4	+1.6	+1.3	-2.0	+1.0
Qwen3-8B	42	+0.4	+0.0	+1.6	+0.7	-0.2	+1.5
Llama-3.1-8B	0	-1.0	+0.2	-0.6	-0.9	-23.7	-1.1
Llama-3.1-8B	1	-1.1	-0.2	-0.2	+0.3	-23.3	-0.6
Llama-3.1-8B	42	-0.9	+0.0	+0.4	-0.9	-21.1	-0.9
Gemma-2-9B	0	-3.1	-1.5	-12.2	-7.1	-12.6	-3.1
Gemma-2-9B	1	-3.6	+0.3	-11.2	-5.0	-24.6	-3.8
Gemma-2-9B	42	-3.0	-1.1	-8.8	-6.2	-18.1	-2.8
Qwen3-14B	0	-1.8	+0.4	+5.8	+2.1	-4.6	+1.1
Qwen3-14B	1	-0.3	+0.9	+3.6	+1.2	+1.5	+1.4
Qwen3-14B	42	+0.4	+0.1	+3.2	+2.0	-0.4	+0.8

H Complete Task and Stage Transition Summaries

The main-text discussion selects illustrative transitions (Sec. 4); Tab. 24 holds that selection, and Tab. 26 enumerates all 72 model/task/stage summaries from the 216 per-seed rows. Loss and gain percentages use all shared task items. Net intervals summarize three run seeds, with a shared aligned reference and no correction for multiple comparisons. These are descriptive intervals, not a family of confirmatory tests. Notation: MATH denotes MATH-500.

Agreement between aligned and final correctness does not imply an unchanged outcome throughout training. To see why, an item can follow $1 \to 0 \to 1$ or $0 \to 1 \to 0$, so both endpoint outcomes agree although warmup differs. Write $H^{a\to b} = L^{a\to b} + G^{a\to b}$ for the fraction of items that change correctness between two stages. The fraction that changes at both stages is

$$R = \frac{H^{A\to W} + H^{W\to F} - H^{A\to F}}{2}. \tag{6}$$

For a binary correctness sequence, an item contributes two to the numerator exactly when both transitions change its outcome. An item that changes at only one stage appears in one stage term and in the endpoint term, so it cancels. An item that changes at both stages returns to its aligned value, so it appears in both stage terms but not in the endpoint term. The identity requires the same item set and scoring rule across all three comparisons and does not identify a learned mechanism.

The combined count leaves open the question of how many initially correct answers were restored, because $1 \to 0 \to 1$ and $0 \to 1 \to 0$ have opposite interpretations. A return to the same correctness

Table 24: **Selected item-level turnover.** Loss L and gain G are percentages of all shared items. Selection illustrates the task/stage patterns of Sec. 4; Tab. 26 reports all contrasts.

Model	Task	Stage	Loss L	Gain G	Net Δ
Llama-3.1-8B	IFEval	$W \to F$	28.0	6.1	-21.9 ± 5.0
Gemma-2-9B	MATH-500	$A \to W$	19.1	5.2	-13.9 ± 3.9
Gemma-2-9B	IFEval	$W \to F$	24.5	4.7	-19.8 ± 14.5
Qwen3-8B	IFEval	$A \to F$	8.4	7.5	-0.9 ± 2.4

Table 25: **Two-stage correctness reversals.** Percentage of shared task items that change correctness at both training stages, returning to their aligned correctness value. Values combine $1 \to 0 \to 1$ and $0 \to 1 \to 0$, using Eq. 6. These counts inherit the limitations of the raw join, the scoring, and the evaluation context in the transition records.

Model	MATH-500	IFEval
Qwen3-8B	4.5 ± 1.4	5.4 ± 1.4
Llama-3.1-8B	4.6 ± 2.6	5.7 ± 2.4
Gemma-2-9B	6.4 ± 4.7	5.5 ± 0.8
Qwen3-14B	3.3 ± 0.6	4.0 ± 1.7

label can also conceal a changed response or reasoning. Establishing recovery of specific warmup losses requires joined response trajectories and repeated-evaluation controls.

Table 26: **Complete task and stage transitions.** Task size N, loss L (%), gain G (%), and net change Δ (points).

Model	Task	Stage	N	L	G	Δ
Qwen3-8B	MMLU	$A \to W$	14042	1.1	1.2	0.1 ± 0.0
Qwen3-8B	MMLU	$W \to F$	14042	2.5	2.8	0.3 ± 1.1
Qwen3-8B	MMLU	$A \to F$	14042	3.0	3.4	0.4 ± 1.1
Qwen3-8B	GSM8K	$A \to W$	1319	4.7	3.1	-1.6 ± 1.3
Qwen3-8B	GSM8K	$W \to F$	1319	2.5	4.0	1.5 ± 0.8
Qwen3-8B	GSM8K	$A \to F$	1319	3.7	3.7	-0.1 ± 0.8
Qwen3-8B	MATH	$A \to W$	500	5.5	4.4	-1.1 ± 3.3
Qwen3-8B	MATH	$W \to F$	500	3.5	5.9	2.4 ± 3.9
Qwen3-8B	MATH	$A \to F$	500	4.5	5.9	1.3 ± 1.1
Qwen3-8B	ARC-C	$A \to W$	1172	2.9	4.2	1.3 ± 0.7
Qwen3-8B	ARC-C	$W \to F$	1172	2.2	2.1	0.0 ± 1.9
Qwen3-8B	ARC-C	$A \to F$	1172	3.3	4.6	1.3 ± 1.5
Qwen3-8B	IFEval	$A \to W$	541	5.5	3.1	-2.3 ± 1.0
Qwen3-8B	IFEval	$W \to F$	541	8.3	9.7	1.4 ± 1.5
Qwen3-8B	IFEval	$A \to F$	541	8.4	7.5	-0.9 ± 2.4
Qwen3-8B	JMMLU	$A \to W$	7536	4.0	4.8	0.8 ± 0.2
Qwen3-8B	JMMLU	$W \to F$	7536	2.4	3.2	0.8 ± 1.3
Qwen3-8B	JMMLU	$A \to F$	7536	4.6	6.2	1.6 ± 1.4
Llama-3.1-8B	MMLU	$A \to W$	14042	1.5	1.5	0.0 ± 0.1
Llama-3.1-8B	MMLU	$W \to F$	14042	2.9	1.9	-1.0 ± 0.2
Llama-3.1-8B	MMLU	$A \to F$	14042	3.3	2.3	-1.0 ± 0.3
Llama-3.1-8B	GSM8K	$A \to W$	1319	6.3	4.8	-1.5 ± 0.2
Llama-3.1-8B	GSM8K	$W \to F$	1319	3.8	5.4	1.5 ± 0.7
Llama-3.1-8B	GSM8K	$A \to F$	1319	4.4	4.4	0.0 ± 0.6
Llama-3.1-8B	MATH	$A \to W$	500	7.5	5.6	-1.9 ± 0.8
Llama-3.1-8B	MATH	$W \to F$	500	4.1	5.9	1.7 ± 1.6
Llama-3.1-8B	MATH	$A \to F$	500	7.0	6.9	-0.1 ± 1.3
Llama-3.1-8B	ARC-C	$A \to W$	1172	3.9	1.8	-2.1 ± 1.8
Llama-3.1-8B	ARC-C	$W \to F$	1172	2.0	3.6	1.6 ± 1.6
Llama-3.1-8B	ARC-C	$A \to F$	1172	3.3	2.8	-0.5 ± 1.6

Tab. 26 continued

Model	Task	Stage	N	L	G	Δ
Llama-3.1-8B	IFEval	$A \to W$	541	7.6	6.8	-0.8 ± 1.9
Llama-3.1-8B	IFEval	$W \to F$	541	28.0	6.1	-21.9 ± 5.0
Llama-3.1-8B	IFEval	$A \to F$	541	29.9	7.2	-22.7 ± 3.5
Llama-3.1-8B	JMMLU	$A \to W$	7536	4.9	4.4	-0.5 ± 0.2
Llama-3.1-8B	JMMLU	$W \to F$	7536	3.7	3.4	-0.3 ± 0.8
Llama-3.1-8B	JMMLU	$A \to F$	7536	5.8	4.9	-0.9 ± 0.6
Gemma-2-9B	MMLU	$A \to W$	14042	1.6	1.5	-0.1 ± 0.0
Gemma-2-9B	MMLU	$W \to F$	14042	5.9	2.7	-3.2 ± 0.8
Gemma-2-9B	MMLU	$A \to F$	14042	6.3	3.0	-3.2 ± 0.8
Gemma-2-9B	GSM8K	$A \to W$	1319	5.0	5.7	0.7 ± 0.4
Gemma-2-9B	GSM8K	$W \to F$	1319	6.1	4.6	-1.5 ± 2.7
Gemma-2-9B	GSM8K	$A \to F$	1319	6.2	5.5	-0.8 ± 2.4
Gemma-2-9B	MATH	$A \to W$	500	19.1	5.2	-13.9 ± 3.9
Gemma-2-9B	MATH	$W \to F$	500	4.4	7.5	3.1 ± 4.0
Gemma-2-9B	MATH	$A \to F$	500	17.1	6.3	-10.7 ± 4.3
Gemma-2-9B	ARC-C	$A \to W$	1172	6.2	2.0	-4.3 ± 1.7
Gemma-2-9B	ARC-C	$W \to F$	1172	6.5	4.6	-1.8 ± 1.4
Gemma-2-9B	ARC-C	$A \to F$	1172	9.7	3.6	-6.1 ± 2.6
Gemma-2-9B	IFEval	$A \to W$	541	5.3	6.7	1.4 ± 1.2
Gemma-2-9B	IFEval	$W \to F$	541	24.5	4.7	-19.8 ± 14.5
Gemma-2-9B	IFEval	$A \to F$	541	24.3	5.9	-18.4 ± 14.9
Gemma-2-9B	JMMLU	$A \to W$	7536	3.6	3.6	0.0 ± 0.3
Gemma-2-9B	JMMLU	$W \to F$	7536	7.1	3.9	-3.2 ± 0.9
Gemma-2-9B	JMMLU	$A \to F$	7536	8.5	5.3	-3.2 ± 1.2
Qwen3-14B	MMLU	$A \to W$	14042	1.1	0.6	-0.5 ± 0.1
Qwen3-14B	MMLU	$W \to F$	14042	1.8	1.7	-0.1 ± 2.8
Qwen3-14B	MMLU	$A \to F$	14042	2.2	1.6	-0.6 ± 2.8
Qwen3-14B	GSM8K	$A \to W$	1319	3.7	1.9	-1.8 ± 0.4
Qwen3-14B	GSM8K	$W \to F$	1319	2.4	4.6	2.3 ± 1.5
Qwen3-14B	GSM8K	$A \to F$	1319	2.6	3.1	0.5 ± 1.0
Qwen3-14B	MATH	$A \to W$	500	2.6	4.5	1.9 ± 0.3
Qwen3-14B	MATH	$W \to F$	500	2.7	4.9	2.3 ± 3.8
Qwen3-14B	MATH	$A \to F$	500	1.9	6.1	4.2 ± 3.5
Qwen3-14B	ARC-C	$A \to W$	1172	2.5	4.6	2.1 ± 1.1
Qwen3-14B	ARC-C	$W \to F$	1172	2.7	2.4	-0.3 ± 1.7
Qwen3-14B	ARC-C	$A \to F$	1172	2.8	4.6	1.8 ± 1.3
Qwen3-14B	IFEval	$A \to W$	541	3.6	2.9	-0.7 ± 2.3
Qwen3-14B	IFEval	$W \to F$	541	7.2	6.7	-0.5 ± 10.0
Qwen3-14B	IFEval	$A \to F$	541	6.8	5.6	-1.2 ± 7.8
Qwen3-14B	JMMLU	$A \to W$	7536	3.3	2.5	-0.8 ± 0.5
Qwen3-14B	JMMLU	$W \to F$	7536	1.5	3.4	1.9 ± 0.4
Qwen3-14B	JMMLU	$A \to F$	7536	2.7	3.7	1.1 ± 0.7

I Selection Sensitivity and Compute Accounting

Checkpoint selection is part of the observed condition. In other words, the retained final checkpoint is selected after training rather than being a randomly sampled terminal state. The 12 headline histories retain candidate rankings and reevaluation scores, but only the selected and terminal weight states remain for most runs. The recorded selection procedure ranks checkpoints that meet a compliance threshold by validation utility, then uses compliance and training step to break ties within a declared utility range. A short list is reevaluated with a more detailed validation setting (Apx. A). A complete evaluation of alternative selection rules is impossible because most candidate weights were not preserved, but the retained reevaluation scores allow a validation-level comparison. A compliance-only rule (highest compliance, earliest step on ties) picks a different checkpoint in 9 of the 12 runs.

Table 27: **Final-stage training resources.** Values are means over three seeds; totals sum the three seed allocations. Training used four NVIDIA H100 80GB GPUs and bfloat16 LoRA (Hu *et al.*, 2022).

Family	Mean wall hours	Mean GPU-hours	Total GPU-hours
Qwen3-8B	12.9744	51.8977	155.6931
Qwen3-14B	12.1434	48.5736	145.7207
Llama-3.1-8B	6.3166	25.2664	75.7993
Gemma-2-9B	8.2966	33.1865	99.5594
All families	9.9328	39.7310	476.7725

Table 28: **Adapted defense-port status.** Harmful-SFT engagement checks on the retained Qwen3-8B ports. ASR and utility are percentages from one check per listed port. These starting conditions leave defense bypass unvalidated.

Port	Check HB	Check utility	Engagement limitation
SEAM	85.5	61.6	Intended self-destruct behavior did not engage in the check.
RepNoise	79.9	60.6	Starting model suppresses free-form output; harmful-SFT robustness not established.
TAR	79.2	61.9	Random-mapping stage was inactive on this family; endpoints differ by final condition.
TokenBuncher	n/a	n/a	No harmful-SFT check, because the defense targets RL; this omission establishes no RL robustness result.

In each, validation utility is 1.1 to 7.2 points lower (median 2.4), and compliance is at most 4 points higher. Taking the last saved checkpoint instead changes the pick in 10 runs, with a median absolute utility difference of 1.4 points and a maximum of 13.3. Three of these comparisons, including the maximum, use only the coarser in-training scores because the last checkpoint was not reevaluated; among fully reevaluated pairs the maximum is 9.4. Because the selected checkpoint maximizes this same validation score, these differences partly reflect selection on noise and do not measure the benchmark-level effect.

Tab. 27 reports the recorded resource accounting for all four headline model families. Wall-clock hours come from MLflow start/end timestamps; allocated GPU-hours multiply wall time by the four-GPU allocation. The cluster has no portable hourly charge, so a dollar estimate would depend on an external price that is not part of the experiment record.

The Qwen3-8B warmup wall times are approximately 0.04 hours per seed; the retained records do not cover the other families. Recorded configuration counts include 1,130 warmup rows, 101,254 prompts in the final policy-optimization pool, eight rollouts per prompt, and a 512-token maximum completion setting. These counts describe configured inputs, not consumed tokens or successful rollouts. The scheduler logs contain 253 evaluation jobs, including failures and retries, and the checkpoint records preserve all 12 final selection histories. Beyond these counts and logs, runtime telemetry falls outside the retained record (Apx. A).

J Attack Conditions and Formal Comparison

Comparing interventions requires formalizing both their starting points and their training conditions. Tab. 28 records the engagement limits of the adapted ports. Each harmful-SFT check is a single seed-42 run on the corresponding defended checkpoint, not a three-seed robustness estimate. RepNoise uses the port recalibrated with 1,000 pairs and TAR the port recalibrated on random mappings; they are distinct from their historical predecessors. Failure of an intended-attack check limits interpretation of these ports, not the original published defenses. TokenBuncher's omitted harmful-SFT test leaves the question of RL engagement open.

The aligned checkpoint A is a reference model, not an attack label. To compare existing interventions without conflating their starting points, let T_k identify a final-stage training condition. Tab. 29 defines five conditions considered for the extended analysis. The main text reports aggregate T_1/T_2

Table 29: **Training-condition classes.** These descriptions concern the final stage; the RL conditions follow supervised warmup. Conditions can differ in several training and selection factors.

Condition	Description
T_1	Harmful-only GRPO, without explicit utility training inputs in the final stage.
T_2	GRPO on harmful and verifiable utility tasks, without concurrent supervised utility anchoring.
T_1^*	Configuration-matched harmful-only control: the recorded T_2 settings with the verifiable utility rows removed.
T_3	Joint comparison combining mixed-task GRPO with concurrent supervised utility training.
T_4	Harmful-target supervised fine-tuning; a same-warmup control only where its parent identity is verified.

Table 30: **Recorded settings in T_1, T_2, and configuration-matched T_1^*.** Configured values as recorded. All three conditions share the no-SFT schedule, under which λ and μ never enter the loss (objective forms: Apx. J.1).

Configured factor	Recorded T_1	Recorded T_2	Recorded T_1^*
Verifiable utility rows	absent	present	absent
Utility-task KL β_u	0	0.05	0.05 (no utility rows)
Utility weight μ ($= 10$ when on)	off	on	on (no utility rows)
GRPO loss weight λ	1.0	10.0	10.0

comparisons, while its item-transition evidence remains specific to T_2; the narrower T_3/T_4 conditions do not inherit that evidence, and T_1^* enters only through the matched comparison of Tab. 31.

The conditions above are recorded configurations, not hypothetical designs. Beyond data composition, T_1 and T_2 differ in three recorded settings (Tab. 30). T_2 sets a tenfold GRPO loss weight ($\lambda = 10.0$ versus 1.0). It also sets a utility-task KL coefficient ($\beta_u = 0.05$ versus 0) and enables the utility weight μ. The conditions share the no-SFT schedule, under which the scalar and decoupled weighting never enter the loss, so the effective conditions differ through the utility data stream and its task-routed KL alone (objective forms: Apx. J.1).

Shared recorded factors include the warmup parent, harm-content reward, adapter configuration, constant schedule, stopping policy, and checkpoint selection. The $T_2 - T_1$ contrast therefore measures the effect of adding verifiable utility rows to a shared attack condition.

The configuration-matched harmful-only comparison T_1^* removes verifiable utility rows while retaining the recorded T_2 non-data settings. It uses the same warmup parents and recorded selection and measurement policy. The recorded weighting knobs (λ, μ) never enter the shared no-SFT loss, and no utility rows remain for the task-routed KL to act on. Its effective objective therefore matches the observed harmful-only condition. The comparison thus repeats the data-composition contrast under the recorded T_2 settings. Matching settings does not establish equal harmful-task exposure, consumed tokens, or compute, so the comparison remains a descriptive check of the data-composition interpretation rather than exact causal isolation.

J.1 Recorded objective forms

Tab. 29 describes conditions qualitatively; the logged factors behind Tab. 30 belong to a joint objective documented in the run configurations, stated here in recorded form.

Let π_θ be the active policy, π_{ref} the reference policy (the warmup parent with the adapter disabled), and $\pi_{\theta_{old}}$ the pre-update policy; here θ indexes policy parameters, not checkpoints. The final-stage sampling stream $\mathcal{D}_{GRPO}$ mixes harmful prompts $\mathcal{D}_{harmful}$ with verifiable reasoning and format prompts $\mathcal{D}_{reason} \cup \mathcal{D}_{format}$. The harm-content judge scores the harmful prompts online, and rule-based correctness rewards score the others. $\mathcal{D}_{SFT}$ is a teacher-forced utility corpus, used only where the recorded condition enables concurrent anchoring (T_3). The anchor loss is

$$\mathcal{L}_{SFT}(\theta) = -\mathbb{E}_{(x_u, y_u)\sim\mathcal{D}_{SFT}} \left[\sum_{t=1}^{|y_u|} \log \pi_\theta(y_{u,t} \mid x_u, y_{u,<t}) \right]. \tag{7}$$

Rewards are task-routed: $r(x, y_i) = \sum_c r_c(x, y_i)$, with each component scoring only its own task tag and returning 0 off-task. For example, the math-correctness reward returns 0 on a harmful prompt, so it cannot move the advantages within a harmful group. The advantage is the within-group standardization $\hat{A}_i = (r(x, y_i) - \mu_r)/\sigma_r$ over G sampled responses, where μ_r and σ_r are the group's reward mean and standard deviation. A response therefore receives a positive advantage when it scores above the mean of its own group. With $\rho_i(\theta) = \pi_\theta(y_i \mid x)/\pi_{\theta_{old}}(y_i \mid x)$, the recorded GRPO objective to maximize is

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}_{x\sim\mathcal{D}_{GRPO},\{y_i\}}\Big[\tfrac{1}{G}\sum_{i=1}^{G}\big(\min(\rho_i\hat{A}_i, \operatorname{clip}(\rho_i, 1-\epsilon, 1+\epsilon)\hat{A}_i) - \beta_{\tau(x)}D_{\mathrm{KL}}(\pi_\theta\|\pi_{ref})\big)\Big], \quad (8)$$

with a task-routed KL coefficient $\beta_{\tau(x)}$. Every recorded condition keeps $\beta = 0$ on $\mathcal{D}_{harmful}$, where a positive coefficient would pull the policy toward the still-refusing reference. The recorded T_2 sets $\beta_u = 0.05$ on $\mathcal{D}_{reason} \cup \mathcal{D}_{format}$, while T_1 records 0 and carries no utility rows. Splitting by tag as $\mathcal{J}_{GRPO} = \mathcal{J}_{unlearn} + \mathcal{J}_{utility}$, the recorded joint objective weights the two GRPO parts separately:

$$\mathcal{L}_{total}(\theta) = \alpha\,\mathcal{L}_{SFT}(\theta) - \lambda\,\mathcal{J}_{unlearn}(\theta) - \mu\,\mathcal{J}_{utility}(\theta), \quad (9)$$

where the GRPO parts enter negated because $\mathcal{L}_{total}$ is minimized. Recorded headline values: T_1 sets $\alpha = 0$ and $\lambda = 1.0$ with no utility rows; T_2 sets $\alpha = 0$, $\lambda = 10.0$, $\mu = 10.0$, and $\beta_u = 0.05$; T_3 sets $\alpha = 1$ with concurrent anchoring; both T_1 and T_2 record a constant (non-adaptive) weighting schedule.

Why warmup is required. The warmup is structurally required rather than optional: under near-uniform refusal, all rewards in a harmful group are near-equal, so $\hat{A}_i \approx 0$ and the policy gradient vanishes; the supervised warmup restores reward variance.

Stopping rule. Stopping is a plateau rule on the mean compliance reward R_k per checkpoint interval. The rule starts once R_k exceeds 0.0 on the harm-judge scale. It stops training after two consecutive intervals fail to improve the best interval mean by more than 0.01. R_k and the rule state are gathered identically across ranks, so every rank halts together.

Recorded versus effective objective. Eqs. 7, 8, and 9 formalize the recorded configuration space that the operator view of Eq. 10 deliberately abstracts over. On the no-anchor path shared by the attack conditions, the implementation returns the unweighted group-relative loss: the task-routed KL enters with the utility rows, while the recorded λ and μ have no effect. The equations therefore describe the recorded configurations, and the effective attack objective is Eq. 8 over each condition's recorded data stream. On the anchored path (T_3), λ and μ balance the GRPO terms against the SFT anchor. Without an anchor there is nothing to balance, and the implementation returns the unweighted loss, so the configured values of λ and μ do not affect the attack conditions.

In the executed implementation, the anchor is a token-mean cross-entropy and the GRPO term uses the recorded `loss_type` (`dapo`, which normalizes over completion tokens); the equations above state the objectives in their canonical response-level form.

J.2 Observed condition comparisons

Tab. 31 reports the resulting observed comparison, and Sec. 4 summarizes it. The paired $T_2 - T_1^*$ utility contrasts are -1.0 ± 1.4 points for Qwen3-8B and -0.9 ± 4.6 for Llama, so both point estimates are slightly higher for T_1^*. The paired HarmBench contrasts are $+1.7 \pm 3.3$ and $+2.9 \pm 25.3$ points. Llama's utility interval, computed before rounding, is approximately $[-5.5, +3.8]$, wide enough to contain practically important losses as well as small gains; its IFEval contrast is -13.4 ± 34.9 points. In plain terms, this comparison is too uncertain to decide anything: it leaves open both utility equivalence and data-composition effects. The nearby HarmBench means are insufficient to establish materially comparable compliance.

Let θ_X denote checkpoint X; these checkpoint symbols are distinct from the policy parameters θ of Apx. J.1. For a defended starting point D_d, write the recorded transformations as

$$\theta_{W_d} = \mathcal{W}(\theta_{D_d}), \qquad \theta_{F_{d,k}} = \mathcal{T}_k(\theta_{W_d}; b), \quad (10)$$

Table 31: **Configuration-matched comparison.** T_1^* removes utility rows at the recorded T_2 non-data settings, which do not enter the shared no-SFT loss. ΔU and ΔHB are paired $T_2 - T_1^*$ differences in points; HB also lists ASR means as T_1^*/T_2.

Branch	T_1	T_1^*	T_2	ΔU	**HB** T_1^*/T_2	ΔHB
Qwen3-8B	68.8 ± 2.9	69.5 ± 1.8	68.5 ± 0.7	-1.0 ± 1.4	94.1/95.8	$+1.7 \pm 3.3$
Llama-3.1-8B	57.4 ± 0.6	56.2 ± 4.7	55.4 ± 0.2	-0.9 ± 4.6	90.1/93.1	$+2.9 \pm 25.3$

Table 32: **Observed $T_2 - T_1$ contrasts.** Utility is the nine-component macro in percentage points. HarmBench and AdvBench are ASR-point differences; SR is the StrongREJECT rubric difference multiplied by 100. Intervals summarize three paired run seeds only.

Branch	**Utility**	**HarmBench**	**AdvBench**	**SR**
Undefended starts				
Qwen3-8B	-0.3 ± 2.9	-2.1 ± 4.8	$+0.0 \pm 0.8$	$+1.9 \pm 3.1$
Llama-3.1-8B	-2.1 ± 0.5	$+8.4 \pm 3.3$	$+12.4 \pm 2.2$	$+19.4 \pm 2.0$
Defense-derived starts				
SEAM	$+0.9 \pm 2.0$	$+2.9 \pm 12.5$	$+2.3 \pm 6.0$	$+3.6 \pm 5.6$
RepNoise	$+1.0 \pm 2.4$	$+5.9 \pm 67.4$	$+14.0 \pm 74.3$	$+12.6 \pm 64.7$
TAR	$+5.7 \pm 21.0$	-39.6 ± 25.8	-39.8 ± 51.0	-41.5 ± 59.6
TokenBuncher	-1.4 ± 2.0	-1.9 ± 2.7	$+0.2 \pm 0.5$	$+3.5 \pm 3.5$

where $\mathcal{W}$ denotes supervised warmup and $\mathcal{T}_k$ the final-stage training and checkpoint-selection procedure under condition T_k, with recorded resource budget b. Each application also depends on the run's record and its randomness, which the notation leaves implicit. Set $D_0 = A$ for the undefended branch. Shared notation for W_d requires an identical parent checkpoint across conditions; matching seed numbers alone is insufficient. A control initialized directly from A or D_d has a separate trajectory. Eq. 10 bounds the comparison's system; it introduces no optimizer and no loss-weighting assumption.

For task score S_t, distinguish retention relative to the defended start from the remaining difference relative to the original model:

$$\Delta^D_{t,d,k} = S_t(F_{d,k}) - S_t(D_d), \qquad \Delta^A_{t,d,k} = S_t(F_{d,k}) - S_t(A). \tag{11}$$

Applied to the utility score of the RepNoise branch, Δ^D is $+22.6$ points and Δ^A is -8.8 points (Sec. 4). The attack improves on the defended start, but it does not restore the aligned model.

The primary contrast between the two GRPO conditions and its change across starting branches are

$$C_t(d) = S_t(F_{d,2}) - S_t(F_{d,1}), \qquad I_t(d) = C_t(d) - C_t(0). \tag{12}$$

Here $C_t(d)$ is the mixed-minus-harmful-only contrast within branch d. $I_t(d)$ asks whether that contrast changes when the attack starts from a defended model rather than from the aligned model.

These quantities summarize observed conditions. They do not isolate utility supervision or a defense mechanism: task composition, supervision, consumed resources, and checkpoint selection can also differ. Objective routing on the shared no-SFT schedule is stated in Apx. J.1. Safety scores accompany utility comparisons separately; unequal safety outcomes limit claims about retention at comparable behavioral change.

Estimate uncertainty for Eqs. 11 and 12 using the actual seed, parent, and item dependence. Shared warmups and defended checkpoints are not independent replicates. A favorable aggregate contrast also does not establish restoration of the same previously lost items, which requires paired outputs at every relevant stage.

Tab. 32 retains the condition differences on all three safety measures. The StrongREJECT column expresses its native $[0, 1]$ score multiplied by 100 so the displayed contrast is in points; it is not an ASR. Unequal or uncertain safety differences prevent interpreting a utility contrast as retention at an equivalent safety endpoint.

The narrower T_3/T_4 coverage is sensitivity evidence rather than a complete factorial comparison. Tab. 33 shows that the joint T_3 condition has a similar macro to T_2 on Qwen and Llama, while Qwen's

Table 33: **Observed condition sensitivities.** Values are in points. Utility is the nine-component macro; HarmBench is classifier ASR.

Contrast	Utility	HarmBench
Qwen3-8B: $T_3 - T_2$	$+0.3 \pm 2.3$	-8.4 ± 0.9
Llama-3.1-8B: $T_3 - T_2$	$+0.9 \pm 0.8$	-3.4 ± 9.5
Qwen3-8B: $T_4 - T_1$	-5.4 ± 2.3	-14.0 ± 6.3
Qwen3-8B: $T_4 - T_2$	-5.1 ± 0.6	-11.9 ± 1.6

harmful-target SFT control T_4 is lower than both RL conditions. Each contrast changes multiple objective, weighting, schedule, target, or budget factors.

Undefended $T_2 - T_1$ contrasts. For Qwen3-8B, the utility and HarmBench contrasts are unresolved across seeds. For Llama, T_2 has a utility score 2.1 ± 0.5 points below T_1, and its HarmBench ASR is 8.4 ± 3.3 points higher. The Llama decline concentrates in IFEval (-23.0 ± 5.9 points). Math and code do not decline.

RepNoise item-level recovery. For the RepNoise branch of Qwen3-8B, we match items across the defended start and later stages. Per seed, 4,573–4,628 of the 25,110 six-task items were correct before the defense and incorrect afterward. Warmup restores only 27–29% of these items. The cumulative restored share at the final checkpoint is 51–53%. Items that are correct at the final checkpoint but were still wrong after warmup make up 27.7–28.2% of the lost set. The two stage shares do not add up to the cumulative share, because some items that warmup restores become incorrect again in the final stage.

Of the items still wrong at the end, 77–81% already failed at the defended start. The RepNoise start suppresses free-form output, so part of this counted deficit is failed expression rather than lost competence. The recovery fractions therefore mix the two.

The analyzed data contain paired T_1/T_2 coverage for the undefended Qwen3-8B and Llama branches and four Qwen3-8B defense branches. Gemma and Qwen3-14B contribute T_2 only.

K Difficult-Benchmark Protocol and Results

Long-context and difficult evaluation. RULER varies length and complexity across retrieval, tracing, aggregation, and question answering (Hsieh *et al.*, 2024). HELMET finds that simple retrieval poorly predicts broader long-context performance (Yen *et al.*, 2025); NoLiMa removes literal word overlap that can make retrieval artificially easy (Modarressi *et al.*, 2025). We measure post-training change in the length-dependent decline relative to the aligned checkpoint, rather than raw difficulty or advertised context size (Sec. 6).

The difficult-benchmark extension tests where the retention finding may stop applying; it does not enlarge the utility macro. We used RULER because it controls task complexity, generated instances, and context length (Hsieh *et al.*, 2024). The aligned-only pilot evaluated multi-key retrieval, multi-hop tracing, and aggregation at common lengths near 2K, 4K, and 8K tokens (Tab. 34). We excluded tasks with very low short-context accuracy, near-perfect accuracy at every length, invalid chat formatting, or unstable exact reruns. The task family, item seeds, target placement, distractors, prompts, output reserve, scoring, and exclusions were fixed before any attacked checkpoint was inspected.

Because literal-match retrieval had near-perfect accuracy, we used the prespecified NoLiMa diagnostic, whose questions and evidence share few words (Modarressi *et al.*, 2025). We did not use HELMET or LongBench v2 because their broader scoring requirements and unequal context limits prevented a direct comparison with the common diagnostic (Yen *et al.*, 2025).

We also prespecified GPQA Diamond and a release-dated LiveCodeBench slice as short-context checks for science and code reasoning (Rein *et al.*, 2024; Jain *et al.*, 2025). LiveCodeBench was gated on the GPQA probe. It would run only if the aligned models exceeded a prespecified accuracy floor, so that a later decline would be measurable. Aligned accuracy fell below that floor, so neither check was evaluated on non-aligned checkpoints and LiveCodeBench never ran.

Table 34: **Common-grid RULER results.** The aligned-only pilot did not provide enough score variation because every candidate task had near-perfect accuracy on 2K/4K/8K-compatible inputs for all four aligned models, except six irregular cells dominated by the generation cap and showing no usable length trend. The longer-window redesign on the {8000, 16384, 32768} grid had the same limitation. We therefore evaluated non-aligned checkpoints with the prespecified NoLiMa diagnostic rather than RULER.

Stage	2K	4K	8K	Aligned n	Non-aligned n
A	1.00	1.00	1.00	100/task	0
Non-aligned stages	n/a	n/a	n/a	0	0

Table 35: **Gemma synthetic task-group sensitivity.** Final-minus-warmup strict accuracy changes in percentage points; 50 items per group. These subgroup contrasts are descriptive and do not extend confirmation to natural tasks.

Task group	Original prompt	Explicit prompt
json arithmetic	$+0.0 \pm 0.0$	-8.0 ± 5.0
keyed lines	-1.3 ± 5.7	-13.3 ± 5.7
pipe transform	$+0.0 \pm 0.0$	$+0.0 \pm 0.0$
tagged sort	-49.3 ± 2.9	$+4.0 \pm 5.0$

For Qwen3-8B and Llama, the non-aligned analysis covers all stages: the aligned checkpoint, the warmup, and both final conditions (A, W, $F_{0,1}$, $F_{0,2}$), with all eligible seeds. Gemma and Qwen3-14B contribute T_2 final checkpoints only (no T_1 checkpoints exist for them); their warmup and other reference rows enter below as boundary rows, as a limited cross-family check. The defended analysis includes four Qwen3-8B branches with verified starting checkpoints and measurable defense effects. Longer 16K or 32K inputs for Qwen and Llama remain model-specific and never enter the common four-model aggregate.

The structured-output confirmation fixed 200 safe, deterministic items, the scorer, and a criterion requiring the strict final-minus-warmup interval to be negative under both prompt styles in at least one family before generating any output. It paired warmup/final seeds for Gemma and Llama. The strict score requires both the correct answer and the exact output format. The semantic score is diagnostic and accepts a correct answer with extra surrounding text.

Gemma met the criterion: strict gaps were -12.7 points $[-14.6, -10.8]$ and -4.3 $[-5.8, -2.9]$. Its original-prompt semantic gap was $+6.7$ $[-5.8, 19.1]$, consistent with a formatting effect but too uncertain to establish semantic preservation.

Llama did not meet the criterion. Its original-prompt strict scores were already near the floor (0.5% at warmup, 0% at final), and its explicit-prompt gap of -10.3 $[-23.4, 2.7]$ includes zero. Llama therefore offers no evidence in either direction on synthetic structured-output loss.

Tab. 35 shows that affected task groups change with prompt style. Aggregate results and the immutable dataset digest are supplied in the supplement. The confirmed loss is limited to synthetic Gemma items; it does not establish a loss of natural instruction following or knowledge.

The prespecified pilot used the lm-eval 0.4.11 implementation of RULER (Apache-2.0) with items generated by a corrected seeded generator and stored for each tokenizer family. On the common {2048, 4096, 8000} grid, aligned accuracy was near-perfect for every task family (100 items per length), except in six irregular cells. Llama common-words-extraction (`cwe`) scored $0.99/0.19/0.96$ across the three lengths, with all 300 rows truncated by the 128-token output cap. Qwen3-8B variable-tracking (`vt`) scored 0.90 at 2048 tokens, and Gemma `vt` scored $0.44/1.00/0.59$ across the three lengths.

These cells reflect the output cap rather than input length and provide too little variation for analysis. We therefore did not evaluate RULER on non-aligned checkpoints.

The longer-window redesign also provided too little variation. Retrieval tasks remained near-perfect up to 32K inputs on the three non-Gemma aligned models (composite containment scores at {8000, 16384, 32768}: Qwen3-8B $1.00/0.99/0.99$, Qwen3-14B $1.00/1.00/0.99$, Llama-3.1-8B

$0.99/0.98/0.94$, with Llama's weakest cell at 0.82). Extraction and aggregation tasks were less accurate, but their scores were affected by generated-output length.

We therefore used the prespecified NoLiMa diagnostic. Its latent-association questions produced clear declines on every aligned model. Qwen3-8B declined $0.66 \rightarrow 0.02$ from 1K to 32K, Llama-3.1-8B declined $0.60 \rightarrow 0.14$, and Qwen3-14B declined $0.76 \rightarrow 0.16$. The non-aligned analysis contained 31 cells covering $A/W/F_{0,1}/F_{0,2}$ on Qwen3-8B and Llama-3.1-8B at seeds 0/1/42, plus boundary rows for Qwen3-14B and Gemma-2-9B. Gemma was limited to the $\leq$8K grid by its 8K token window; the engine's input-truncation outcome, $n = 1508$ rows per Gemma run, was analyzed separately.

The prespecified excess-degradation measure detects no intervention-specific long-context effect under the tested configuration. The per-seed excess decline E of a non-aligned checkpoint is measured against the aligned model's own length-dependent decline. In words, E near zero means that the checkpoint loses accuracy with length just as the aligned model does. It stays within -4.2 to $+7.3$ points at every length for both long-context families (32K means: Qwen3-8B $+2.8$ points, Llama-3.1-8B -3.4 points). The between-condition difference $C_T(L) = E_{F_{0,2}} - E_{F_{0,1}}$ stays within 1.33 points at every length.

The GPQA Diamond probe did not meet the prespecified aligned-accuracy criterion: the Qwen3-8B and Llama-3.1-8B aligned probes scored $0.29/0.23$ against the $[0.35, 0.90]$ band. The LiveCodeBench arm, gated on the GPQA outcome by the frozen sequencing rule, therefore did not start. Neither was evaluated on non-aligned weights. We define no equivalence margin for this diagnostic, and the observed intervals are too wide to exclude effects of a few points. The stated result is therefore a bounded null under the tested NoLiMa grid, prompt templates, and decoding protocol. It leaves open the possibility of a long-context effect. Effects smaller than these widths would be invisible to this diagnostic.

The defended analysis used the same grid for four Qwen3-8B branches from SEAM, TAR, RepNoise, and TokenBuncher, with matched warmups and both final training conditions (40 cells). For all branches except TAR, the difference between conditions stays within ± 3 points at every length. TAR is the consistent exception (8K/16K/32K: $-5.7/-9.1/-11.7$ points): $F_{d,2}$ follows the aligned reference more closely than $F_{d,1}$. This is an observed condition-by-branch interaction, not an identified defense or training mechanism. RepNoise's long-context cells include its suppression behavior as an empty-output category (23,125 rows).

L Nearest-Work Comparison

Tab. 36 maps each reported observation onto its nearest prior record cited in Secs. 2 and 3. We checked the versions available on September 15, 2026 (September 26 for the two added durability and jailbreak-tax records) and recorded canonical sources and supported claims. Cells use three marks: **Y** (the record reports the observation in this setting), **(y)** (the record reports a closely related measurement in a different setting), and $\times$ (not reported in the checked record).

The table supports the positioning claim of Sec. 2. No checked record reports all four observations together. Each exists at best partially, in a scattered record or a different setting. None combines stage-resolved utility trajectories, defended starting points, recorded training comparisons, and scorer-aware evaluation controls for safety-alignment removal. This coverage map positions the measurement findings; by itself it says nothing about whether any prior pipeline's retention result is right or wrong.

M Measurement Setting and Threat Model

The questions of this study sit in the open-weight fine-tuning threat model of the tamper-resistance literature (Tamirisa *et al.*, 2025), in which the cost of removal can limit an attack even when safety behavior is removable. SEAM makes this dependence explicit by coupling harmful fine-tuning to capability collapse (Wang *et al.*, 2026).

Adversary goal. Produce a derivative of a safety-aligned open-weight model that complies with arbitrary harmful instructions while retaining the reasoning, coding, and multilingual utility of the original. Retention matters to the adversary because a fully compliant but degraded model offers less

Table 36: **Nearest-work map.** Rows are the observations reported here; columns are the nearest checked records. Y: reported in this setting. (y): closely related measurement in a different setting (constructive post-training, instruction tuning, or defense evaluation rather than alignment-removal trajectories). ×: not reported in the checked record. The records key below the table expands each column name to its checked publication.

Observation	Harmon	Cacioli	Kotha	Gan	GRP-Oblit.	Harm RLVR	Kabir &Tig.	TokenB.	Qi 2025	JB tax
Stage-resolved item transitions under removal	(y)	(y)	×	×	×	×	(y)	×	×	×
Utility and compliance measured at each stage	×	×	×	×	(y)	(y)	(y)	(y)	(y)	(y)
Defended starts with start-vs-original decomposition	×	×	×	×	×	×	×	(y)	(y)	×
Matched evaluation controls with re-scored aligned reference	×	×	(y)	(y)	×	×	×	×	(y)	×
Blinded quantification of checker disputes by multiple judges	×	×	×	×	×	×	×	×	×	×
Prespecified boundary probes under removal	×	×	×	×	×	×	×	×	×	×

Records: Harmon (2026); Cacioli (2026); Kotha (2024); Gan (2026); GRP-Obliteration (Russinovich *et al.*, 2026); HarmRLVR (Liu *et al.*, 2026); Kabir and Tiganj (2026); TokenBuncher (Feng *et al.*, 2026); Qi 2025: durability of safeguards (Qi *et al.*, 2025); JB tax: the jailbreak tax (Nikolić *et al.*, 2025).

capability for misuse. The retention question is therefore what this study measures, and it is why high attack success alone cannot establish a capability-preserving removal.

Adversary capabilities. White-box access to the released weights, a single rented multi-GPU node, and publicly available harmful-prompt datasets. Low-memory adapters keep the footprint modest; Apx. I gives the partial compute accounting. The adversary never sees the provider's alignment pipeline, preference data, or reward model. The recorded pipelines under study additionally reuse a small refusal-altering warmup pool of 1,130 pairs (Apx. B, Tab. 8).

Adversary knowledge and scope. The adversary knows the target is safety-aligned and knows the public attack literature. The adversary edits weights: API-only models and inference-time jailbreaks are out of scope, as is pretraining from scratch, the regime in which pretraining data-filtering defenses apply (O'Brien *et al.*, 2025).

Defender. The model provider or hosting platform releasing open weights meant to resist cheap re-removal of safety. The defense-derived branches of Sec. 3 correspond to adapted ports of four published defenses, with engagement limits disclosed in Apx. J.

The release policy of Apx. N is calibrated to this setting: aggregates support scrutiny of the measurements, while the withheld operational artifacts are exactly the capabilities enumerated above.

N Ethics, Release, and Reproducibility Policy

This study concerns changing safety behavior in open-weight models, so a conventional reproducibility release could lower the cost and raise the reliability of misuse. The study's scientific value is measurement: it asks which reported capability changes survive controlled evaluation. That value does not require releasing attacked weights, harmful prompt collections, generated harmful responses, executable training configurations, or a pipeline that can be directly repurposed.

The anonymous archive `supplementary_material.zip` accompanies the manuscript. It contains per-seed metric summaries; transition counts; recorded-condition contrasts; aggregate panel, strict/loose-checker, evaluation-control, and boundary-study results; benign scoring and pairing

specifications; sanitized selection and compute summaries; the reference tables used for consistency checks; and SHA-256 manifests. Its standard-library table generator and checker require no model dependencies or network access. The README maps included files to manuscript claims and states the scope of each check.

This package supports arithmetic, seed aggregation, interval calculation, and table consistency. Identities over the transition counts and the reported pairing summaries can be checked, but the underlying per-item joins and response semantics remain beyond independent audit from aggregates. Private reconstruction with the pinned scorer is distinct from aggregate verification by reviewers. The package does not reproduce the intervention.

The release policy excludes training and inference code, operational patches, containers, training lock files and configurations, weights or adapters, optimizer states, prompts, completions, individual judge outputs, detailed reward traces, infrastructure metadata, and benchmark data. The included arithmetic code operates only on aggregate measurements. Sanitization removes institutional endpoints, local paths, tracking identifiers, and author metadata while preserving seed and stage relationships.

All automated judging stayed inside the institutional environment. The harm-content judge used in training, the HarmBench classifier, and the three-model scorer panel (GLM-5.3, Inkling, and Kimi-K3) that assessed the fixed safety responses all ran on a private inference endpoint. No harmful prompts or model responses were transmitted to external providers, so no third-party data-retention or access terms apply to those contents.

Residual risk remains: aggregate findings can still inform harmful actors, while withholding operational artifacts limits independent reproduction. Restricting reproducibility evidence to aggregate measurements permits scrutiny of the reported results without packaging a capability-changing workflow.