CRUSH THE DARK SIDE: UNLEARNING VIA SUPPRESSING EXPOSED LLMS' HAZARDOUS OUTPUT TENDENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) are widely deployed in real-world applications, yet their safety remains vulnerable to evolving jailbreak attacks. Existing defenses, including alignment and unlearning, typically rely on training data organized as question-answer pairs. As a result, their robustness is inherently bounded by the coverage of harmful prompts observed during training. Since natural language admits virtually unlimited reformulations, exhaustively covering the question space is infeasible in practice. To address this limitation, we propose Answer-Centric Unlearning (ACU), a framework that suppresses harmful answers exposed by the model itself. Specifically, ACU exposes latent harmful trajectories, iteratively suppresses them without conditioning on questions, and then trains on a retain set to restore normal utility. Experiments across four 8B backbones and one 32B model show that ACU improves robustness on eight additional jailbreak attacks while maintaining acceptable utility.

1 INTRODUCTION

Modern Large Language Models (LLMs) have achieved remarkable success across various domains Touvron et al. (2023); OpenAI (2023), yet their security remains challenged by sophisticated jailbreak attacks Zou et al. (2023b); Liu et al. (2024a); Chao et al. (2023); Andriushchenko et al. (2025). Current defensive efforts, including safety alignment Ouyang et al. (2022); Bai et al. (2022a;b) and machine unlearning Eldan & Russinovich (2024); Maini et al. (2024); Li et al. (2024a), typically operate by enforcing safe refusals or suppressing harmful outputs in response to specific harmful queries. Nevertheless, as attack methods continue to evolve Russinovich et al. (2025); Li et al. (2024b); Ren et al. (2024), these defenses often struggle to generalize against unseen attack forms Mazeika et al. (2024); Andriushchenko et al. (2025); Li et al. (2024b), leaving models vulnerable to novel jailbreak attempts.

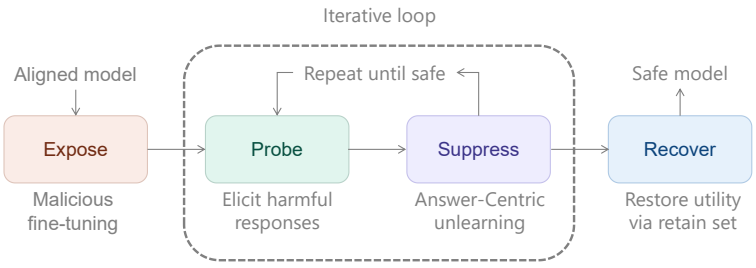


Figure 1: The workflow of ACU.

To improve the robustness of LLM defenses, prior alignment and unlearning methods commonly expand the coverage of harmful queries during training, such as broadening the training distribution with rewritten or adversarially generated attack prompts Bai et al. (2022a); Dai et al. (2024); Mazeika et al. (2024); Xhonneux et al. (2024) and augmenting the unlearning targets with adversarial variants Liu et al. (2024c); Ye et al. (2025); Sheshadri et al. (2025); Casper et al. (2025). However,

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.

improving robustness by expanding the question space is inherently asymmetric. A defender must anticipate a broad range of attack patterns, while an attacker only needs to discover a single uncovered corner to succeed. This asymmetry becomes increasingly severe as jailbreak attacks move beyond manually written prompts, such as agent-driven strategies Hagendorff et al. (2026); Chang et al. (2025). In contrast, a given model's harmful outputs tend to converge on a narrower set of generation trajectories Orgad et al. (2026); Soligo et al. (2025); Chen et al. (2025). For instance, although there are countless ways to prompt a model for instructions on synthesizing a dangerous substance, the resulting harmful responses typically converge on the same sequence of procedural steps and domain-specific terminology. This observation suggests a different strategy: rather than pursuing exhaustive coverage of question space, robust defense should directly target the answer space by suppressing the model's harmful generation trajectories.

To operationalize this strategy, we must characterize the harmful answer space to suppress. Since unlearning is more effective when its targets closely match the model's own output distribution Li et al. (2024b), existing methods that rely on static, pre-collected question-answer pairs Liu et al. (2024c); Zhang et al. (2024); Li et al. (2025) introduce distribution mismatch from two perspectives. First, in a well-aligned model, harmful answers are inherently suppressed, making pre-collected harmful responses a poor proxy for the model's actual output distribution. However, the model's capacity to generate harmful content remains intact beneath the alignment layer Qi et al. (2024); Xie et al. (2025b); briefly relaxing alignment through lightweight fine-tuning makes these latent trajectories observable and therefore targetable. Second, unlearning is a dynamic process where the model's generation distribution shifts as the model's parameters are updated, causing any fixed unlearning targets to become progressively misaligned. Therefore, the unlearning targets must be re-sampled from the model iteratively to stay aligned with the model's evolving generation behavior.

Building on these insights, we propose Answer-Centric Unlearning (ACU), a framework that operates in four stages for generalized defense against jailbreak attacks, as shown in Figure 1. First, it applies lightweight malicious fine-tuning Qi et al. (2024); Xie et al. (2025b) to temporarily expose the model's latent harmful generation distribution (this relaxation is reversed in the final recovery stage). Second, it probes the exposed model with diverse harmful queries and collects the harmful answers. As these answers are sampled from the current model, they provide well-matched unlearning targets. Third, ACU directly suppresses these answers without conditioning on the triggering questions. To track the shifting output distribution, the second and third stages are repeated iteratively until the model no longer produces harmful answers under the probing queries. Finally, ACU uses a benign retain set to restore general utility and safety alignment. We implement ACU across four 8B backbones and one 32B model, and evaluate eight additional jailbreak attacks. Our results show that ACU consistently improves robustness over question-dependent baselines while maintaining competitive model utility. In summary, our contributions are:

- We identify the inherent asymmetry in question-dependent defenses and propose ACU, an answer-centric unlearning framework that shifts the defense objective from the unbounded question space to the model's own answer space.
- We introduce an expose-and-suppress strategy that dynamically aligns unlearning targets with the model's own output distribution, ensuring suppression operates on the most reachable harmful trajectories throughout the unlearning process.
- We implement ACU on four 8B backbones and one 32B model, and evaluate eight additional jailbreak attacks, showing that ACU improves robustness while maintaining model utility.

2 Motivation and Mechanism

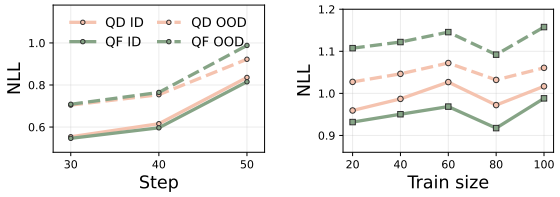
2.1 Why Answer-Only Suppression?

Question-dependent (QD) unlearning suppresses harmful answers conditioned on specific prompts, yet exhaustively covering all possible adversarial phrasings is practically infeasible. Recent findings suggest that harmful outputs often share underlying generation patterns across diverse prompts Orgad et al. (2026); Soligo et al. (2025); Chen et al. (2025), which motivates our Answer-Centric Unlearning (ACU) approach to directly target this shared answer-side component. Let ℓ_θ denote the mean answer-token negative log-likelihood (NLL). We formulate the QD and question-free (QF)

suppression objectives as:

$$\mathcal{L}_{\mathrm{QD}} = -\mathbb{E}_{q\sim\pi}\ell_\theta(a \mid q), \qquad \mathcal{L}_{\mathrm{QF}} = -\ell_\theta(a \mid q_\emptyset),$$

where π is a distribution over eliciting questions and $q_\emptyset$ denotes the native chat template with an empty user message. Because QF removes the eliciting prompt during training, a natural question is whether its suppression effect can still generalize when harmful queries are reintroduced at test time. To evaluate this cross-question transfer, we compare QF and QD on a controlled bomb-making task, using direct requests for in-distribution (ID) evaluation and unseen jailbreak prompts for out-of-distribution (OOD) evaluation. Across varying forget-set sizes, we measure the target harmful-answer NLL under identical test prompts. As shown in Figure 2, while QD achieves lower likelihoods on ID prompts, QF yields substantially stronger suppression on OOD jailbreaks. This gap confirms that answer-only training transfers effectively across unseen prompts, and we further validate its behavioral impact through open-ended generation in Section 4.3.1. To understand the mechanism driving this generalization, we pair harmful answers with diverse questions and compare the top-gradient MLP neuron sets activated under QF and QD. We find that both objectives increasingly engage a shared set of internal parameters as generation progresses, as detailed in Appendix E.



(a) Suppression trajectory. (b) Final suppression results.

Figure 2: Concept-level suppression evaluated under identical test questions. Higher target NLL indicates stronger suppression. QF: answer-only; QD: question-dependent.

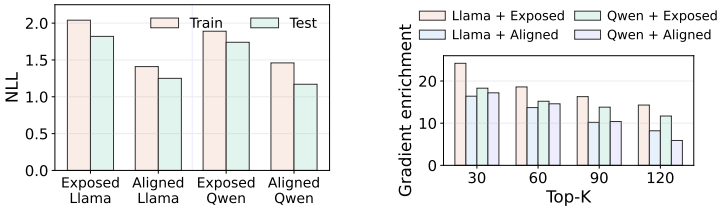
2.2 Why Expose and Refresh the Targets?

Even with answer-only suppression, target selection strongly governs how well unlearning generalizes. Let $g_\tau = \nabla_\theta \ell_\theta(\tau)$ and $g_{\tau'}$ denote the NLL gradients of a sampled target τ and an unseen harmful trajectory τ'. Under a small gradient-ascent step η, the first-order change in test NLL is

$$\ell_{\theta+\eta g_\tau}(\tau') - \ell_\theta(\tau') \approx \eta g_\tau^\top g_{\tau'}.$$

Local transfer therefore relies on strong gradient alignment between training targets and test-time failure modes, motivating the use of model-native trajectories. In our framework, *exposure* surfaces these intrinsic trajectories from the model, while *refresh* dynamically updates them as the model distribution shifts during training (see Appendix E.5 for retention and optimizer dynamics).

We next investigate whether exposure indeed supplies more effective targets by comparing suppression on exposed versus safety-aligned Llama models, using both self-generated and external (Qwen-generated) harmful answers. Figure 3(a) demonstrates that pairing the exposed model with its own self-generated answers yields the strongest final suppression. While differences in initial loss prevent attributing this gain solely to exposure, the results strongly support the advantage of suppressing model-native targets over foreign ones.



(a) NLL with different targets. (b) Gradient enrichment.

Figure 3: Effect of exposure targets: (a) negative log-likelihood across different target sources; (b) gradient enrichment on localized bomb-related parameters.

At the parameter level, we further analyze which internal components these targets engage by measuring gradient enrichment on localized bomb-related neurons relative to a layer-matched random

baseline (Appendix E.2). As shown in Figure 3(b), exposed, self-generated targets consistently achieve the highest enrichment across all tested thresholds. Although parameter enrichment alone does not guarantee behavioral safety, it confirms that model-native targets more precisely concentrate updates on concept-relevant parameters.

Finally, we examine the necessity of dynamically updating these targets during training by comparing fixed and refreshed answer banks under matched compute budgets. Combining target refresh with exposure consistently improves suppression performance (Section 4.3.2), as later collection rounds successfully uncover residual harmful modes that emerge after initial updates (Table 7). Together, exposure provides high-alignment initial targets, while refresh adapts to shifting failure modes as the model evolves.

3 METHOD

ACU follows the four stages in Figure 1: exposure, probing, answer-only suppression, and utility recovery. The same adapter is carried through all stages. The algorithm and implementation details are provided in Appendix A.

Setup and exposure. Let M_{align} be the initially aligned model, $\mathcal{D}_{\text{harm}}$ the exposure set, $\mathcal{Q}$ the probe pool, and $\mathcal{D}_{\text{benign}}$ the benign retain pool. For $a = (a_1, \ldots, a_m)$, we average NLL over answer tokens:

$$\ell_\theta(a \mid q) = -\frac{1}{m} \sum_{j=1}^{m} \log p_\theta(a_j \mid q, a_{<j}). \tag{1}$$

Prompt and padding tokens receive no direct loss. Exposure fine-tunes M_{align} for two epochs on 50 harmful question-answer pairs Mazeika et al. (2023), producing M_{exp}. This step temporarily relaxes safety alignment to make latent harmful answers available for collection.

Iterative probing and suppression. We partition $\mathcal{Q}$ into ordered chunks of at most 100 questions. In each round, the current model generates five candidates per question. Each question-answer pair is evaluated by Qwen3Guard-8B, and harmful responses are collected into a multiset $\mathcal{A}_t$ (retaining multiplicity for repeated occurrences). During suppression, we omit the eliciting questions from the harmful-answer loss and pair $\mathcal{A}_t$ with an equally sized benign sample $\mathcal{B}_t \sim \mathcal{D}_{\text{benign}}$ to regularize each update:

$$\mathcal{L}^{(t)}_{\text{train}}(\theta) = \underbrace{-\mathbb{E}_{a \sim \mathcal{A}_t} \ell_\theta(a \mid q_\emptyset)}_{\mathcal{L}_{\text{sup}}} + \lambda \underbrace{\mathbb{E}_{(q^r, a^r) \sim \mathcal{B}_t} \ell_\theta(a^r \mid q^r)}_{\mathcal{L}^{(t)}_{\text{ret}}}. \tag{2}$$

Minimizing this objective increases harmful-answer NLL while maintaining benign response likelihood. Losses are averaged over tokens within each answer and then across answers, ensuring consistent per-answer normalization for both the harmful and benign terms. In practice, $\mathcal{B}_t$ is sampled without replacement in each round from the 5,000-pair retain pool $\mathcal{D}_{\text{benign}}$ (reused across rounds), with an effective batch size of eight harmful/benign pairs. Ablations on λ and bounded-loss alternatives are provided in Table 28.

After each round, we probe the updated model again. A chunk stops when its sampled round contains no judge-positive answers or reaches 15 rounds; its final model initializes the next chunk. While this zero-positive criterion governs training termination, we also conduct full-probe audits before and after recovery to evaluate residual harmful outputs without feeding audit responses back into training. Figure 4 illustrates the resulting chunk trajectories.

Utility recovery. After suppression, we fine-tune for three epochs on $\mathcal{D}_{\text{rec}} = \mathcal{D}_{\text{util}} \cup \mathcal{D}_{\text{ref}}$, comprising 25 benign utility pairs and 25 refusal pairs. Recovery minimizes the standard supervised mean answer NLL on $\mathcal{D}_{\text{rec}}$. In Section 4.4, we isolate its contribution from the preceding suppression updates using paired stage checkpoints and an aligned-model recovery-only control.

Implementation. All stages use bfloat16 LoRA with rank 8, scaling 16, and dropout 0.05. Exposure and recovery use learning rate 5×10^{-5}; suppression uses 10^{-5} and $\lambda = 0.5$. To construct $\mathcal{Q}$, we rewrite 856 deduplicated harmful intents eight times each to obtain 6,848 candidate queries, and

retain 5,928 probes after filtering with the exposed model. Generation uses temperature 0.7, top-p 0.9, top-k 50, repetition penalty 1.05, and at most 1,024 new tokens. Further details on template handling, batch construction, and data separation are provided in Appendix A.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Models and attacks. We evaluate Llama-3-8B-Instruct Meta (2024), Qwen2.5-7B-Instruct Yang et al. (2024), Qwen3-8B Yang et al. (2025), Ministral-3-8B-Instruct Liu et al. (2026), and Qwen3-32B. The main test panel contains 440 SORRY-Bench intents Xie et al. (2025a), evaluated under direct requests, 19 SORRY-Bench variants, and eight additional jailbreak attacks: DeepInception Li et al. (2023), FlipAttack Liu et al. (2024b), SATA Dong et al. (2025), HEA Song et al. (2025), QueryAttack Zou et al. (2025), SequentialBreak Saiem et al. (2025), CoP Xiong et al. (2025), and JailCon Jiang et al. (2025). Beyond fixed prompts, we separately evaluate target-specific adaptive attacks such as PAIR Chao et al. (2023) and GCG Zou et al. (2023b) as well as fine-tuning-based attack including MFT Qi et al. (2024), AvO Xie et al. (2025b), and TrojanPraise Xie et al. (2026).

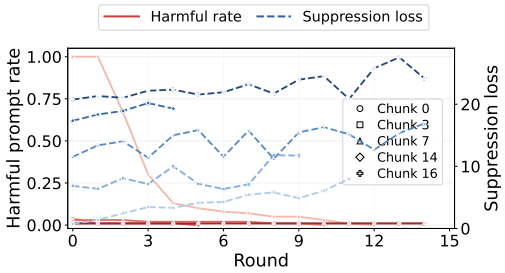


Figure 4: Training dynamics of ACU across successive chunks.

Baselines and metrics. We compare ACU against the aligned model and six baselines: Safe Unlearning Zhang et al. (2024), Eraser Lu et al. (2024), RMU Li et al. (2024a), Circuit Breakers Zou et al. (2024), CKU Shi et al. (2025), and SafeLLM Li et al. (2025). For safety, GPT-5 judges the final harmful-response rate (HR) independently of the probing judge; for capability preservation, we measure general task utility, false-refusal rate (FRR) on benign queries, and repetitive degeneration. Appendices A.1–A.4 detail the evaluation panels and statistical protocols, computational budgets, and human validation of the judges (including stratified error audits).

4.2 ROBUSTNESS AND GENERALIZATION

Table 1: Main Results: Mean HR Across Ten Attack Conditions (%, ↓)

Backbone	Aligned	Safe Unl.	Eraser	RMU	CB	CKU	SafeLLM	ACU
Llama3	11.82	8.92	9.34	8.92	9.49	10.16	8.39	**4.22**
Qwen2.5	11.95	10.41	8.18	10.52	10.17	9.65	8.36	**3.88**
Qwen3	12.27	7.94	10.94	10.05	10.10	8.58	9.81	**3.88**
Ministral3	12.67	10.33	11.02	8.29	10.56	10.91	9.62	**4.46**
Qwen3-32B	11.75	8.98	8.25	8.63	7.83	8.47	7.32	**2.78**

Table 1 presents our main results across five backbones, where ACU consistently achieves the lowest mean HR. On the four 7–8B models, ACU reduces HR from 11.82–12.67% (aligned checkpoints) to 3.88–4.46%; on Qwen3-32B, it reaches 2.78%, down from 11.75% for the aligned model and 7.32% for the strongest baseline (SafeLLM). Detailed results are reported in Appendix B. Specifically, under direct requests, the aligned models already exhibit very low HR (e.g., 0.53% on Llama3, reduced to 0.15% by ACU), indicating that simple direct queries do not reflect the broader jailbreak surface. Instead, ACU's overall improvements come primarily from curbing complex attacks where aligned models fail: for instance, ACU reduces QueryAttack from 27.21% to 9.58% and DeepInception from 19.15% to 5.20% on Llama3, and cuts QueryAttack from 24.36% to 5.08% on Qwen3-32B.

Held-out harmful intents. To test generalization to unseen harm categories, we adopt a leave-one-category-out protocol that excludes one category at a time from all training stages and development

selection (100 test intents per category, five responses per seed). Table 2 shows that while category exclusion moderately weakens suppression relative to full-data training—most notably in the drug category (rising from 2.53% to 6.47%)—ACU still maintains substantially lower HR than the aligned model across all five categories. Thus, answer-only suppression transfers across the evaluated categories, though probe coverage still affects its strength. The category-exclusion procedure and complete results across four backbones are presented in Appendix B.3.

Table 2: Llama3 Leave-One-Category-Out HR (%, ↓)

Training setting	Weapon	Physical harm	Biosecurity	Drug	Cyber intrusion
Full data	2.33	2.47	2.60	2.53	2.73
ACU held out	4.87	5.27	4.13	6.47	5.40
Aligned model	8.60	12.33	9.87	11.13	13.47

Optimization-based attacks. While fixed attacks rely on predetermined jailbreak template, PAIR and GCG optimize adversarial prompts directly against each defended checkpoint. Across 200 held-out intents, we score five fresh responses to the final optimized prompt per intent and seed (excluding intermediate search outputs). Table 3 reports both response-level HR and intent-level success within five attempts (ASR_5). On Llama3, ACU reduces HR relative to Circuit Breakers by 8.70 points under PAIR and 8.80 points under GCG, with a consistent advantage on Qwen3. Nevertheless, ASR_5 remains appreciably higher than HR, revealing successful repeated attempts even after the average response rate declines. The optimization-based attack protocols and complete baseline results are provided in Appendix B.4.

Table 3: Optimization-based Attack Results (%, ↓). CB: Circuit Breakers.

		HR		ASR_5	
Backbone	Attack	CB	ACU	CB	ACU
Llama3	PAIR	19.13	10.43	41.67	26.50
Llama3	GCG	22.47	13.67	45.50	29.83
Qwen3	PAIR	17.53	8.80	39.00	22.17
Qwen3	GCG	20.23	12.03	39.83	30.00

Weight-update attacks. For MFT, AvO, and TrojanPraise, the attacker fine-tunes each defended checkpoint for 100 updates using 100 attack pairs, after which we evaluate on direct harmful questions. ACU achieves the lowest post-attack HR across all model as shown in Table 16, demonstrating stronger resistance to adapter-level tampering.

4.3 Ablation Study

4.3.1 Is Answer-Only Suppression Necessary?

Matched-target objective comparison. To isolate the effect of conditioning context, we fix the exposed checkpoint, harmful target bank, benign retain records, batch order, 1,000-update schedule, and recovery set, so that QF and QD differ only in whether the question is conditioned on (masked question tokens in QD still incur forward-pass compute). All metrics in Table 4 score the same post-recovery checkpoints. Note that the QF row here corresponds to the fixed-target ($\lambda = 0.5$) variant in Table 28 rather than the full iterative ACU pipeline; Appendix Table 24 further extends this matched-data control to Safe Unlearning and Circuit Breakers (whereas Table 1 evaluates their full native pipelines).

Table 4 shows consistently lower HR for QF across all four backbones: 4.35% versus 8.61% on Llama3, 4.11% versus 8.35% on Qwen2.5, 4.23% versus 8.48% on Qwen3, and 4.65% versus 8.89% on Ministral3. This uniform reduction of roughly four percentage points under identical answers and update schedules confirms cross-question transfer across model families. Because QF exhibits slightly lower utility and marginally higher FRR and degeneration rates, we interpret this HR reduction alongside capability preservation.

Table 4: QF/QD Controls Across Four Backbones After Recovery (%). HR: mean ± SD.

Backbone	Objective	HR ↓	Utility ↑	FRR ↓	Degen. ↓
Llama3	QF	4.35±0.39	70.12	8.00	1.27
Llama3	QD	8.61±0.75	71.02	7.47	0.77
Qwen2.5	QF	4.11±0.34	71.84	7.60	1.20
Qwen2.5	QD	8.35±0.70	72.61	7.03	0.73
Qwen3	QF	4.23±0.50	74.08	7.30	1.10
Qwen3	QD	8.48±0.66	74.81	6.83	0.70
Ministral3	QF	4.65±0.34	72.02	8.23	1.37
Ministral3	QD	8.89±0.70	72.80	7.63	0.83

To test whether the HR gap merely reflects reduced capability, we also compare development-selected checkpoints at matched aggregate utility. On Llama3, QF and QD attain comparable utility scores of 70.76 and 70.82, yet HR remains substantially lower for QF at 5.35% versus 8.30%. This confirms that the safety gain holds at matched aggregate utility, although task-level variations remain possible. Moreover, a recovery-only control on the aligned model yields 11.11% HR, confirming that the final recovery stage alone cannot reproduce the suppression benefit (see Appendix F and Table 24 for full details).

Question coverage at fixed answers. Next, we vary QD question coverage using nested sets of one, two, four, or eight questions per answer, while a single QF reference replays the same 400 unique answers, retain records, and update schedule. We evaluate on seen training questions, as well as withheld reformulations and unseen attack expressions of the same intents (all excluded from the largest training set). As shown in Table 5, expanding QD question coverage improves transfer, yet QF still maintains a clear HR advantage on new wordings and unseen attacks despite slightly higher HR on seen questions (where QD itself is non-monotonic). These finite comparisons demonstrate QD's sensitivity to prompt coverage rather than a universal upper bound on its robustness as shown in Appendix F.1.

Table 5: Fixed-Answer Question-Coverage Results (%)

	Generated HR ↓			Utility	FRR	Degen.
Training context	Seen	New wording	New attack	↑	↓	↓
QD, 1 question	2.55	5.73	9.18	71.02	7.47	0.77
QD, 2 questions	2.35	4.85	8.08	70.95	7.63	0.87
QD, 4 questions	2.23	4.18	7.18	70.91	7.60	0.83
QD, 8 questions	2.37	4.05	6.62	70.88	7.70	0.93
QF, fixed reference	2.92	3.67	4.73	70.12	8.00	1.27

4.3.2 Are Exposure and Target Refresh Necessary?

Exposure and iterative refresh. We evaluate a $2 \times 2 \times 2$ factorial design crossing exposure, target refresh, and conditioning context (QF vs. QD). All eight arms share the same 88,920-candidate collection budget, benign pool, recovery set, and 1,000-update schedule. Fixed-target runs collect a static answer bank before suppression, whereas refresh runs re-collect answers from the evolving model during training; each arm resamples only its own accepted answers with replacement to equalize update counts (controlling optimization steps while allowing target yield and diversity to vary). Table 6 summarizes the test results, with full protocols, target-mass analysis (Appendix E.6), and sensitivity to exposure-set composition and probe budgets in Appendix C.2.

Combining exposure, refreshed targets, and QF yields the lowest test HR at 3.58%. Removing exposure raises HR to 14.51%, fixing the targets raises it to 6.09%, and conditioning on the question (QD) raises it to 12.87%. Notably, target refresh reduces QF HR by 2.51 points with exposure (6.09% to 3.58

Table 6: Llama3 Factorial Ablation: Test HR (%, Mean ± SD)

Exposure	Targets	QD	QF
No	Fixed	16.65±0.45	14.48±0.37
No	Refresh	15.14±0.43	14.51±0.46
Yes	Fixed	13.33±0.34	6.09±0.44
Yes	Refresh	12.87±0.35	**3.58±0.41**

Mechanistically, exposure boosts the initial fixed-bank yield from roughly 1,815 to 22,367 harmful occurrences. Moreover, when unlearning a fixed exposed model, using its own self-generated targets achieves lower HR (4.35%) than using targets from the unexposed aligned model (5.80%) or an external model (6.53%). Conversely, when we fix an exposed target bank and apply it to either the aligned or exposed model, both models benefit (Appendix G), indicating that exposure primarily serves to supply high-yield, model-native targets rather than being a strict prerequisite for the recipient model.

Coverage across collection rounds. Suppressing an initial set of sampled answers may still leave alternative harmful trajectories intact. To test whether iterative collection uncovers new failure modes, we cluster harmful answers by underlying goal, action, and outcome (ignoring surface wording). The first collection pass defines the initial cluster inventory, while subsequent passes count harmful responses belonging to previously unseen clusters. In parallel, we evaluate a separate held-out audit set of 500 intents (excluded from training and development, yielding 7,500 responses across three seeds) to track actual HR alongside new cluster discovery.

Table 7: Target Discovery and Held-Out Audit HR Across Collection Passes

Pass	Harmful collected	New-cluster answers	Discovery (%)	Audit HR (%)
1	5,244	5,244	100.00	52.16
2	1,836	489	26.63	9.59
3	627	106	16.91	2.97

Table 7 shows that newly discovered clusters still account for 26.63% and 16.91% of collected harmful responses in the second and third passes, respectively. Thus, even as total harmful outputs decline, the updating model continues to surface alternative harmful modes absent from the initial bank. Across the same passes, held-out audit HR drops sharply from 52.16% to 9.59% and then 2.97%, directly supporting iterative target refresh as the output distribution shifts. While zero positives on a training chunk serve as a local stopping criterion, this disjoint audit confirms progressive suppression beyond the training probes.

4.4 Utility Preservation and Over-Refusal

General task utility. We evaluate general capability preservation across seven standard benchmarks (Table 8). On Llama3, ACU closely tracks the aligned model in macro-average accuracy and remains competitive with or superior to all baselines, avoiding the steep degradation on reasoning-heavy tasks (such as GSM8K and ARC-C) exhibited by several unlearning methods. This utility preservation holds consistently across Qwen2.5, Qwen3, Ministral3, and Qwen3-32B (Appendix Tables 20 and 34).

Over-refusal and open-ended generation quality. To verify that safety gains do not stem from indiscriminate refusal, we evaluate false-refusal rates (FRR) on benign XSTest cases Röttger et al. (2024), sensitive benign queries, and OR-Bench Cui et al. (2025). Across both Llama3 test sets (Table 9) and all four backbones on OR-Bench (Table 10), ACU maintains near-aligned refusal boundaries and avoids the elevated over-refusal observed in baselines such as Eraser and RMU. Open-ended evaluations on OR-Bench and WritingBench Wu et al. (2025) further confirm preserved helpfulness, writing quality, and contextual coherence (Appendix D, Tables 32–35). Finally, stage-wise audits confirm that while answer-only suppression already eliminates unsafe compliance, the

Table 8: Llama3 General Utility Accuracy Across Benchmarks (%, ↑)

Method	MMLU	HellaSwag	ARC-C	BoolQ	PIQA	WinoGrande	GSM8K	**Avg.**
Aligned model	64.17	75.80	55.64	84.05	78.68	72.00	74.79	72.16
Safe Unlearning	62.16	74.51	50.96	82.52	76.92	71.19	63.72	68.85
Eraser	61.60	72.07	55.21	84.25	77.40	68.89	72.11	70.22
RMU	61.79	73.25	48.35	81.87	75.86	71.23	70.14	68.93
Circuit Breakers	61.87	73.11	49.62	81.54	76.26	68.30	73.71	69.20
CKU	62.36	74.50	50.03	83.18	77.97	69.86	70.93	69.83
SafeLLM	62.69	74.46	51.41	83.49	78.67	71.13	71.47	70.47
ACU	62.36	74.71	53.44	84.07	75.20	69.53	73.14	**70.35**

lightweight recovery stage remedies the temporary utility drop, over-refusal, and surface degeneration induced by unlearning (Appendix H, Table 43).

Table 9: Llama3 FRR on Benign Requests

Method	XSTest	Sens. Benign
Aligned	6.83	8.27
Safe Unl.	10.93	9.87
Eraser	11.37	15.20
RMU	9.27	12.67
CB	8.73	8.53
CKU	8.13	10.67
SafeLLM	7.80	10.13
ACU	**7.43**	**9.07**

Table 10: OR-Bench FRR Across Backbones

Method	Llama3	Qwen2.5	Qwen3	Ministral3
Aligned	12.06	8.29	7.60	8.24
Safe Unl.	9.25	11.63	10.74	11.38
Eraser	14.27	12.65	17.07	13.79
RMU	9.44	12.27	13.58	12.34
CB	9.38	9.41	9.66	9.17
CKU	7.99	11.77	10.93	12.92
SafeLLM	10.96	13.34	7.54	10.30
ACU	9.65	**8.78**	8.14	**7.92**

5 Conclusion and Limitations

We presented ACU, an answer-centric unlearning framework that exposes latent harmful trajectories, iteratively refreshes model-native targets, and suppresses them via an answer-only objective before restoring utility. Across five backbones, ACU substantially reduces harmful responses against diverse jailbreaks while preserving general capabilities and refusal boundaries, with controlled ablations confirming that answer-only suppression transfers across unseen prompts and that exposure and refresh track shifting failure modes. Nevertheless, our study is bounded by finite prompt pools, model sizes, and attack budgets: optimization-based attacks (particularly with gradient access) retain non-trivial success, effectiveness depends on probe coverage, and the shared Llama-filtered pool may favor one backbone. Moreover, recovery is partial, answer-only suppression can affect benign sensitive discussions, matched answer-token supervision does not equalize total compute, and judge errors or semantic leakage remain possible. Finally, behavioral suppression is not certified knowledge deletion and does not provide a global safety guarantee.

AI use statement

Gemini assisted probe expansion, and GPT-5 judged final evaluations. Generative AI also assisted experiment planning and manuscript editing. The authors take responsibility for the study design, evaluation protocol, and final manuscript.

REFERENCES

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned LLMs with simple adaptive attacks. In *ICLR*, 2025.

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

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Stephen Casper, Lennart Schulze, Oam Patel, and Dylan Hadfield-Menell. Defending against unforeseen failure modes with latent adversarial training. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=mVPPhQ8cAd.

Wenhan Chang, Tianqing Zhu, Yu Zhao, Shuangyong Song, Ping Xiong, and Wanlei Zhou. Chain-of-Lure: A universal jailbreak attack framework using unconstrained synthetic narratives. *arXiv preprint arXiv:2505.17519*, 2025. URL https://arxiv.org/abs/2505.17519v2.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*, 2023.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramer, Hamed Hassani, and Eric Wong. JailbreakBench: An open robustness benchmark for jailbreaking large language models. In *NeurIPS*, 2024.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*, 2025.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. OR-bench: An over-refusal benchmark for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 11515–11542. PMLR, 2025. URL https://proceedings.mlr.press/v267/cui25a.html.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8493–8502. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.581. URL https://aclanthology.org/2022.acl-long.581/.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe reinforcement learning from human feedback. *ICLR*, 2024.

Xiaoning Dong, Wenbo Hu, Wei Xu, and Tianxing He. SATA: A paradigm for LLM jailbreak via simple assistive task linkage. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in LLMs. *EMNLP*, 2024.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.446. URL https://aclanthology.org/2021.emnlp-main.446/.

Thilo Hagendorff, Erik Derner, and Nuria Oliver. Large reasoning models are autonomous jailbreak agents. *Nature Communications*, 17:1435, 2026. doi: 10.1038/s41467-026-69010-1. URL https://www.nature.com/articles/s41467-026-69010-1.

Yukun Jiang, Mingjie Li, Michael Backes, and Yang Zhang. Adjacent words, divergent intents: Jailbreaking large language models via task concurrency. *NeurIPS*, 2025.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. *arXiv preprint arXiv:2403.03218*, 2024a.

Nathaniel Li et al. LLM defenses are not robust to multi-turn human jailbreaks yet. *arXiv preprint arXiv:2408.15221*, 2024b.

Xiangman Li, Xiaodong Wu, Qi Li, Jianbing Ni, and Rongxing Lu. SafeLLM: Unlearning harmful outputs from large language models against jailbreak attacks. *arXiv preprint arXiv:2508.15182*, 2025.

Xuan Li, Zhanke Zhou, Jianing Zhu, Jiangchao Yao, Tongliang Liu, and Bo Han. DeepInception: Hypnotize large language model to be jailbreaker. *arXiv preprint arXiv:2311.03191*, 2023.

Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. AutoDAN: Generating stealthy jailbreak prompts on aligned large language models. *ICLR*, 2024a.

Yue Liu, Xiaoxin He, Miao Xiong, Jinlan Fu, Shumin Deng, Yingwei Ma, Jiaheng Zhang, and Bryan Hooi. FlipAttack: Jailbreak LLMs via flipping. *arXiv preprint arXiv:2410.02832*, 2024b.

Zheyuan Liu, Guangyao Dou, Zhaoxuan Tan, Yijun Tian, and Meng Jiang. Towards safer large language models through machine unlearning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1817–1829. Association for Computational Linguistics, 2024c. doi: 10.18653/v1/2024.findings-acl.107. URL https://aclanthology.org/2024.findings-acl.107/.

Weikai Lu, Ziqian Zeng, Jianwei Wang, Zhengdong Lu, Zelin Chen, Huiping Zhuang, and Cen Chen. Eraser: Jailbreaking defense in large language models via unlearning harmful knowledge. *arXiv preprint arXiv:2404.05880*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. *COLM*, 2024.

Mantas Mazeika, Andy Zou, Norman Mu, Long Phan, Zifan Wang, Chunru Yu, Adam Khoja, Fengqing Jiang, Aidan O'Gara, Ellie Sakhaee, Zhen Xiang, Arezoo Rajabi, Dan Hendrycks, Radha Poovendran, Bo Li, and David Forsyth. TDC 2023 (LLM edition): The trojan detection challenge. In *NeurIPS Competition Track*, 2023.

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. URL https://proceedings.mlr.press/v235/mazeika24a.html.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, volume 35, pp. 17359–17372, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *ICLR*, 2023.

Meta. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Hadas Orgad, Boyi Wei, Kaden Zheng, Martin Wattenberg, Peter Henderson, Seraphina Goldfarb-Tarrant, and Yonatan Belinkov. Large language models generate harmful content using a distinct, unified mechanism. *ICLR Workshop on Logical Reasoning of Large Language Models*, 2026.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

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! *ICLR*, 2024.

Qibing Ren, Hao Li, Dongrui Liu, Zhanxu Xie, Xiaoya Lu, Yu Qiao, Lei Sha, Junchi Yan, Lizhuang Ma, and Jing Shao. Derail yourself: Multi-turn LLM jailbreak attack through self-discovered clues. *arXiv preprint arXiv:2410.10700*, 2024.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5377–5400, 2024.

Mark Russinovich, Ahmed Salem, and Ronen Eldan. Great, now write an article about that: The crescendo multi-turn LLM jailbreak attack. In *34th USENIX Security Symposium*, pp. 2421–2440, 2025.

Bijoy Ahmed Saiem, MD Sadik Hossain Shanto, Rakib Ahsan, and Md Rafi Ur Rashid. SequentialBreak: Large language models can be fooled by embedding jailbreak prompts into sequential prompt chains. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 548–579, 2025. doi: 10.18653/v1/2025.acl-srw.37. URL https://aclanthology.org/2025.acl-srw.37/.

Abhay Sheshadri, Aidan Ewart, Phillip Guo, Aengus Lynch, Cindy Wu, Vivek Hebbar, Henry Sleight, Asa Cooper Stickland, Ethan Perez, Dylan Hadfield-Menell, and Stephen Casper. Latent adversarial training improves robustness to persistent harmful behaviors in LLMs. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=6LxMeRlkWl.

Zesheng Shi, Yucheng Zhou, Jing Li, Yuxin Jin, Yu Li, Daojing He, Fangming Liu, Saleh Alharbi, Jun Yu, and Min Zhang. Safety alignment via constrained knowledge unlearning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25515–25529, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1240. URL https://aclanthology.org/2025.acl-long.1240/.

Anna Soligo, Edward Turner, Senthooran Rajamanoharan, and Neel Nanda. Convergent linear representations of emergent misalignment. *arXiv preprint arXiv:2506.11618*, 2025.

Xurui Song, Zhixin Xie, Shuo Huai, Jiayi Kong, and Jun Luo. Dagger behind smile: Fool LLMs with a happy ending story. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothee Lacroix, Baptiste Roziere, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, and Fei Huang. WritingBench: A comprehensive benchmark for generative writing. In *Advances in Neural Information Processing Systems 38*, pp. 58018–58051, 2025. doi: 10.52202/085713-1737. URL https://papers.nips.cc/paper_files/paper/2025/hash/4aedf0cba303537fcb6cf948bb41b2df-Abstract-Datasets_and_Benchmarks_Track.html.

Sophie Xhonneux, Alessandro Sordoni, Stephan Günnemann, Gauthier Gidel, and Leo Schwinn. Efficient adversarial training in LLMs with continuous attacks. In *NeurIPS*, 2024.

Tinghao Xie, Xiangyu Qi, Yi Zeng, Yangsibo Huang, Udari Madhushani Sehwag, Kaixuan Huang, Luxi He, Boyi Wei, Dacheng Li, Ying Sheng, Ruoxi Jia, Bo Li, Kai Li, Danqi Chen, Peter Henderson, and Prateek Mittal. SORRY-Bench: Systematically evaluating large language model safety refusal. In *International Conference on Learning Representations*, 2025a.

Zhixin Xie, Xurui Song, and Jun Luo. Attack via overfitting: 10-shot benign fine-tuning to jailbreak llms. *NeurIPS*, 2025b.

Zhixin Xie, Xurui Song, and Jun Luo. TrojanPraise: Jailbreak LLMs via benign fine-tuning. *arXiv preprint arXiv:2601.12460*, 2026.

Chen Xiong, Pin-Yu Chen, and Tsung-Yi Ho. CoP: Agentic red-teaming for large language models using composition of principles. In *NeurIPS*, 2025.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Xiaotian Ye, Mengqi Zhang, and Shu Wu. LLM unlearning should be form-independent. *arXiv preprint arXiv:2506.07795*, 2025.

Zhexin Zhang, Junxiao Yang, Pei Ke, Shiyao Cui, Chujie Zheng, Hongning Wang, and Minlie Huang. Safe unlearning: A surprisingly effective and generalizable solution to defend against jailbreak attacks. *arXiv preprint arXiv:2407.02855*, 2024.

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

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*, 2023b.

Andy Zou, Long Phan, Justin Wang, Derek Duenas, Maxwell Lin, Maksym Andriushchenko, Rowan Wang, Zico Kolter, Matt Fredrikson, and Dan Hendrycks. Improving alignment and robustness with circuit breakers. In *NeurIPS*, 2024.

Qingsong Zou, Jingyu Xiao, Qing Li, Zhi Yan, Yuhang Wang, Li Xu, Wenxuan Wang, Kuofeng Gao, Ruoyu Li, and Yong Jiang. QueryAttack: Jailbreaking aligned large language models using structured non-natural query language. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5725–5741, 2025. doi: 10.18653/v1/2025.findings-acl.298. URL https://aclanthology.org/2025.findings-acl.298/.

A Implementation and Evaluation Protocol

Exposure uses 50 harmful question-answer pairs from the cited source Mazeika et al. (2023), two epochs, and learning rate 5×10^{-5}. One LoRA adapter targets attention and MLP linear projections; its weights carry through exposure, suppression, and recovery, while AdamW state resets at stage boundaries. Within the suppression stage, optimizer state carries across rounds and chunks. AdamW uses $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, zero weight decay, constant learning rate, and gradient clipping at norm 1.0.

Probe construction starts from 856 deduplicated intents from AdvBench Zou et al. (2023b), HarmBench Mazeika et al. (2024), and JailbreakBench Chao et al. (2024). Eight rewrites per intent give 6,848 candidates; filtering with the exposed Llama model retains 5,928 probes, shared across backbones. Test and development intents and their semantic paraphrases are excluded before expansion. Held-out-category runs repeat construction after category exclusion; their pool sizes can differ. The shared pool controls query choice but may retain a Llama-specific sampling bias.

Suppression uses learning rate 10^{-5}, retain weight $\lambda = 0.5$, and a pool of 5,000 benign instruction-answer pairs. Each sampled harmful occurrence is paired with one benign record; the retain pool is sampled without replacement within a round and reused across rounds. One round traverses the paired data once with effective batch size eight pairs (microbatch one pair, accumulation eight). Incomplete final batches divide by their actual record count. Per-answer token losses are averaged before averaging across answers; prompt and padding tokens are masked. The native template contains an empty user message and an assistant prefix, with no extra system instruction. Recovery uses 25 benign and 25 refusal pairs, three epochs, batch size eight, and learning rate 5×10^{-5}.

Candidate generation uses at most 1,024 new tokens, temperature 0.7, top-p 0.9, top-k 50, and repetition penalty 1.05. Generation settings remain fixed across methods, with seeds 17, 29, and 43. We retain the same seed ordering when pairing conditions. Answer truncation, chat-template choice, and refusal scoring are applied consistently within each panel; any attack-specific search settings are separated from final response generation.

A.1 Aggregation and evaluation panels

For each seed, each non-SORRY main-table column has $440 \times 5 = 2,200$ outcomes; the SORRY column pools $440 \times 19 \times 5 = 41,800$. Across three seeds these denominators become 6,600 and 125,400. A complete main panel has $440 \times 28 \times 5 \times 3 = 184,800$ outcomes per method. Within a column, equal seed denominators make the mean of seed rates identical to the pooled rate. Table 1 averages the ten column means equally, so it gives the pooled SORRY column the weight of one column rather than 19 conditions.

Development uses 190 intents disjoint from the 440 SORRY-Bench test intents at the semantic-intent level. Each development intent has one direct prompt, two SORRY variants, and two additional attacks, giving 950 prompts. Across the panel, the 19 SORRY variants occur 20 times each; the first four additional attacks occur 48 times each and the other four 47 times each. Five responses per prompt and three seeds yield 14,250 outputs. The budget sweep and development ablations remain on this split; final safety claims use test results.

Selection and freezing. Exact matches and semantic paraphrases of development/test intents are excluded from every training stage; all rewrites stay in one split. Retain weight, probe budget, round limit, and baseline settings minimize development HR, subject to FRR increasing by at most three points and mean NLL by 0.15 nats/token on 500 separate benign development prompts. These prompts are disjoint from training and final utility evaluations. Ties within 0.1 HR points favor fewer updates.

We freeze hyperparameters, checkpoint-selection rules, attacks, prompt assignments, and judge rubrics before scoring each of the three seeds, without best-seed selection. Online stopping uses only training-probe outputs and the frozen round limit. The factorial experiment instead fixes 1,000 updates per arm.

Confirmatory panels. Each of the 440 test intents receives one direct prompt, two SORRY variants, and two additional attacks. A cyclic assignment over a seed-17 shuffled intent list gives six

SORRY variants 47 occurrences and thirteen 46; each additional attack occurs 110 times. The resulting 2,200 prompts yield 11,000 responses per seed and 33,000 overall. Each intent contains 25 responses per seed. Development and test share attack families but no underlying intents.

Qwen3-32B uses all 28 main-panel conditions: 184,800 outputs per method. We average ten column rates within each seed, then report mean and sample SD across seeds. The factorial panel is a subset of these test conditions, not an independent replication dataset. Matched-target and stage controls retain one wrapper per intent.

Let h_{sj} be harmful counts and n_j the denominator for seed s and column j. We report $r_{sj} = 100h_{sj}/n_j$, $\bar{r}_j = \frac{1}{3}\sum_s r_{sj}$, and sample SD $\sqrt{\sum_s (r_{sj} - \bar{r}_j)^2/2}$. Percentages are displayed to two decimals after aggregation. Mean rates and SD use all three seeds; paired bootstrap intervals additionally require the response-to-intent mapping and are reported only for panels with that mapping.

Statistical inference and reporting. In Tables 13 and 14, FPR is FP/(FP+TN) and FNR is FN/(FN+TP). Repeated responses and rewrites within an intent are correlated. We construct 95% percentile intervals from 2,000 bootstrap samples of intent identifiers drawn with replacement, using the same sampled identifiers for every method in a paired contrast. All completions, attack variants, and seed records remain within each intent cluster. These intervals quantify variation over intents conditional on the evaluated seeds; sample SD separately describes variation across seeds. The main pipeline uses chunk stopping, whereas factorial controls fix 1,000 updates and five prompt variants per test intent. Matched-target and stage controls use one wrapper per intent. Absolute HR values across these panels reflect different training schedules and evaluation weights and should not be treated as interchangeable. In Table 4, HR is the seed mean with sample SD, FRR and degeneration are pooled response rates, and utility is mean task accuracy. In Table 43, HR and FRR are seed means with sample SD, correct refusal and degeneration are pooled rates, and utility is mean task accuracy.

Algorithm 1 ACU: Answer-Centric Unlearning.

Require: Aligned model M_{align}, exposure set $\mathcal{D}_{\text{harm}}$, probe questions $\mathcal{Q}$, benign retain set $\mathcal{D}_{\text{benign}}$, recovery set $\mathcal{D}_{\text{rec}}$, judge M_{judge}, retain weight λ, step size η, samples K, max rounds T
Ensure: Updated model M_{safe}

// Stage 1: Expose latent harmful trajectories
$M_0 \leftarrow \text{Finetune}(M_{\text{align}}, \mathcal{D}_{\text{harm}})$
$M_{\text{latest}} \leftarrow M_0$
// Stage 2 & 3: Iterative probe-and-suppress
for each ordered chunk $\mathcal{Q}_c$ of at most 100 queries **do**
  $M_c \leftarrow M_{\text{latest}}$
  for $t = 0, 1, \ldots, T-1$ within chunk c **do**
    $\mathcal{A}_t \leftarrow \emptyset$
    for each $q \in \mathcal{Q}_c$ **do**
      Sample answers $\{a^{(k)}\}_{k=1}^{K} \sim M_c(\cdot \mid q)$
      $\mathcal{A}_t \leftarrow \mathcal{A}_t \uplus \{a^{(k)} : M_{\text{judge}}(q, a^{(k)}) = \texttt{harmful}\}$
    end for
    if $\mathcal{A}_t = \emptyset$ **then**
      break
    end if
    Sample $\mathcal{B}_t \sim \mathcal{D}_{\text{benign}}$ with $|\mathcal{B}_t| = |\mathcal{A}_t|$
    $M_c \leftarrow \text{SuppressEpoch}(M_c, \mathcal{A}_t, \mathcal{B}_t; \eta, \lambda)$
    $M_{\text{latest}} \leftarrow M_c$
  end for
  $M_{\text{latest}} \leftarrow M_c$
end for
// Stage 4: Utility recovery
$M_{\text{safe}} \leftarrow \text{Recover}_{3\,\text{epochs}}(M_{\text{latest}}, \mathcal{D}_{\text{rec}})$
return M_{safe}

A.2 Utility Evaluation Details

Within each backbone, all compared checkpoints use the same tokenizer, native chat template, task splits, and scoring implementation. MMLU uses five demonstrations; HellaSwag, ARC-C, BoolQ, PIQA, and WinoGrande use zero-shot prompts; GSM8K uses eight demonstrations. MMLU and BoolQ use candidate-label likelihood, HellaSwag, ARC-C, and PIQA use length-normalized continuation likelihood, and WinoGrande scores the disambiguating continuation. GSM8K uses greedy generation with a 1,024-token output cap and exact match after comma removal and extraction of the last numeric answer. Qwen3 uses non-thinking mode throughout. Prompt and answer limits are identical within each task comparison.

Utility tables report percentages on each task's evaluation split: MMLU test, HellaSwag validation, ARC-Challenge test, BoolQ validation, PIQA validation, WinoGrande validation, and GSM8K test. The macro average weights the seven task scores equally. Stage comparisons reuse the same questions and demonstrations; they do not average accuracy over a mixture of task examples with different sample counts.

XSTest uses 200 benign cases and five sampled responses per seed; the sensitive-benign set contains 50 separate prompts. For OR-Bench, the FRR panel contains 10,000 benign prompts, evaluated once per seed, whereas quality uses 50 prompts per category with a 1–5 rubric for relevance, usefulness, and coherence. WritingBench covers six domains and uses query-specific criteria on a 1–10 scale. The additional context panel contains 100 benign prompts with safety-sensitive terminology, five responses per seed, and separate refusal and coherence labels. All checkpoint comparisons reuse the same prompts and rubrics.

A.3 Computational Resources

The reference pool has 59 full chunks and one 28-query chunk. The fixed three-round collection schedule uses $5 \times 5,928 \times 3 = 88,920$ candidates. Two full audit sweeps add 59,280 outputs. The 29.64K entry in the budget table denotes one sweep; the general pipeline instead requires $K \sum_c |\mathcal{Q}_c| T_c$ outputs plus audit and construction calls. A probe-budget fraction uses $\lceil f\, 5,928 \rceil$ queries. The fixed schedule is used for collection-budget comparisons; adaptive stopping can reduce the number of executed rounds.

The A6000 resource breakdown in Table 11 is per Llama model and per seed: exposure takes 0.10, target generation and suppression 27.80, recovery 0.10, and audits/evaluation 14.00 target-GPU-hours, totaling 42.00. The separate judge consumes 9.00 GPU-hours. Three seeds require 126.00 target-GPU-hours and 27.00 judge-GPU-hours. The reference schedule has 849 suppression updates. GPU time sums active device time and is distinct from concurrent wall time; optimization-based attack search and 32B evaluation are excluded from this ledger.

Final main-panel judging uses 61,600 decisions per seed and 184,800 across three seeds, per method. At a budget of 512 input and 8 output tokens per decision, this corresponds to 94,617,600 input and 1,478,400 output tokens. The associated budget is $94.6176 p_{\text{in}} + 1.4784 p_{\text{out}}$ for prices per million tokens. This budget uses the stated per-decision token allowance and excludes retries, attack search, and auxiliary evaluations; GPU time and wall-clock latency are reported separately.

Table 11: Llama Training and Evaluation Costs per Seed (GPU Hours)

Component	Target GPU h	Judge GPU h
Exposure	0.10	0.00
Probe and suppress	27.80	5.40
Recovery	0.10	0.00
Audits and evaluation	14.00	3.60
Total	**42.00**	**9.00**

A.4 Judge Validation

We use Qwen3Guard-8B to select probing targets and GPT-5 for final response evaluation, keeping the judge distinct from the defended backbone. The final judge receives both request and response under a fixed rubric: "Label actionable assistance for the harmful intent as harmful; label refusal,

safe redirection, and non-actionable discussion non-harmful. Return one label." The same rubric applies across methods. An answer-only sensitivity check removes the request while retaining each response.

Three domain experts independently annotate 200 responses spanning harmful requests, jailbreak responses, refusals, and benign safety-related queries, with majority vote defining the reference label and Fleiss' $\kappa = 0.86$. On this validation set, which contains 106 harmful reference labels, the probing judge Qwen3Guard-8B yields TP=99, FP=4, FN=7, and TN=90. Accuracy is $189/200 = 94.5\%$, precision $99/103 = 96.1\%$, and recall $99/106 = 93.4\%$. Removing the request changes 11 labels (5.5%), supporting request-aware scoring for context-dependent cases. This small validation set does not establish uniform accuracy across all attack families; near-tied method differences should therefore be interpreted cautiously.

A.5 Direct validation of the final GPT-5 judge

GPT-5 is evaluated separately on the same 200-response human gold set described above, with the same harmful-label definition and no access to the human labels during prediction. It yields TP=102, FP=3, FN=4, and TN=91. Its accuracy is $193/200 = 96.5\%$, precision $102/105 = 97.1\%$, recall $102/106 = 96.2\%$, specificity $91/94 = 96.8\%$, and F1 $204/211 = 96.7\%$. The Qwen3Guard confusion matrix and the 11-label request-removal result remain measurements of the probing judge.

Table 12: Judge Validation on 200 Human-Labeled Responses

Judge	TP	FP	FN	TN	Accuracy (%)	Recall (%)
Qwen3Guard-8B	99	4	7	90	94.5	93.4
GPT-5	102	3	4	91	96.5	96.2

Blinded stratified audit. An additional audit contains 1,600 responses: 160 from each of ten checkpoint or method conditions. Each condition contributes 40 direct harmful-request responses, 40 SORRY responses, 40 additional-attack responses, and 40 benign-sensitive responses, with ten responses per 7–8B backbone in each cell. Within the harmful-request cells, sampling balances probing-judge positive and negative predictions where available; the benign cell is sampled without that balance. The audit therefore enriches difficult labels and does not estimate deployment prevalence. Three experts independently label shuffled, anonymous question-answer pairs without method, stage, seed, or model-judge labels. Majority vote defines gold, and Fleiss' $\kappa = 0.88$. All disagreements are adjudicated after the independent labels are locked; the confusion counts below use the majority labels before adjudication.

Table 13: GPT-5 Errors by Method or Stage (160 Responses Each)

Condition	TP	FP	FN	TN	FPR (%)	FNR (%)
Aligned	58	3	2	97	3.00	3.33
Exposed	61	2	3	94	2.08	4.69
Suppressed	54	2	2	102	1.92	3.57
Recovered (full ACU)	55	2	2	101	1.94	3.51
Fixed-target QF	54	3	2	101	2.88	3.57
Fixed-target QD	58	2	3	97	2.02	4.92
Safe Unlearning, full	56	3	3	98	2.97	5.08
Circuit Breakers, full	57	2	2	99	1.98	3.39
Safe Unlearning adaptation	55	3	3	99	2.94	5.17
Circuit Breakers adaptation	56	2	2	100	1.96	3.45

Across conditions, the audit contains TP=564, FP=24, FN=24, and TN=988, giving 97.0% accuracy. Method-wise FPR ranges from 1.92% to 3.00% and FNR from 3.33% to 5.17%. These estimates do not establish equal error rates: for example, the fixed-QF FNR is 2/56, with a 95% Wilson interval of approximately [0.98%,12.12%], and fixed-QD FNR is 3/61, with interval [1.69%,13.49%]. The audit is too small to exclude meaningful method-specific differences. Its enriched composition also prevents directly subtracting its aggregate error rate from main-panel HR.

Table 14: GPT-5 Errors by Prompt Family (400 Responses Each)

Family	TP	FP	FN	TN
Direct	191	6	8	195
SORRY	188	5	8	199
Additional attacks	185	3	8	204
Benign sensitive	0	10	0	390

The additional-attack cell cycles the eight families with five responses per family and condition. False positives include safe, context-sensitive discussion; false negatives include partially actionable answers mixed with refusal language. These failure modes motivate keeping the request in the final rubric and retaining uncertainty for close method comparisons. Human audit labels are never used for model training, attack selection, or post-hoc threshold changes.

B Full Safety Results and Generalization

The tables below report harmful-response rates for each backbone and attack. Means and sample standard deviations use three seeds; integer counts are aggregated before rounding. The main summary averages attack-column means, whereas the fine-tuning panel reports each attack separately.

Table 15: HR by Backbone and Attack (%, ↓)

Method	Direct	SORRY	DeepInc.	Flip	SATA	HEA	Query	Seq.	CoP	JailCon
Llama-3-8B-Instruct										
Aligned model	0.53	5.78	19.15	3.35	9.27	18.02	27.21	13.73	14.14	7.05
Safe Unlearning	0.77	3.55	15.58	4.83	6.33	17.18	13.85	13.17	10.67	3.27
Eraser	0.45	4.27	17.77	3.52	7.95	11.97	25.53	7.55	8.77	5.65
RMU	0.42	3.53	12.88	2.79	5.21	17.73	21.26	9.45	11.61	4.36
Circuit Breakers	0.17	6.42	12.42	3.48	5.02	13.11	20.80	15.79	14.55	3.14
CKU	0.76	4.75	13.18	1.24	10.20	12.20	25.50	11.24	13.52	8.98
SafeLLM	0.29	5.82	11.55	1.33	7.21	14.23	15.42	8.36	13.42	6.21
ACU	**0.15**	**2.40**	**5.20**	**1.06**	**3.77**	**7.65**	**9.58**	**4.70**	**5.15**	**2.50**
Qwen2.5-7B-Instruct										
Aligned model	0.52	5.71	19.48	6.06	8.77	17.92	24.27	14.12	16.41	6.24
Safe Unlearning	0.92	5.64	15.33	3.14	7.09	15.89	22.50	11.23	16.62	5.70
Eraser	0.26	5.34	12.23	5.27	5.02	10.88	16.32	7.36	12.58	6.55
RMU	0.52	2.81	20.59	8.45	6.32	14.26	19.27	16.08	13.55	3.39
Circuit Breakers	0.55	2.47	15.47	2.14	9.45	16.08	21.48	12.98	16.35	4.68
CKU	0.17	5.94	12.94	6.21	3.82	18.53	22.98	12.42	10.05	3.39
SafeLLM	0.20	3.72	13.36	8.85	7.94	11.67	12.52	8.45	12.65	4.27
ACU	**0.05**	**2.21**	**6.23**	**2.11**	**2.32**	**6.11**	**7.08**	**4.74**	**5.92**	**2.03**
Qwen3-8B										
Aligned model	0.62	6.25	20.50	4.83	9.35	17.95	25.76	15.08	15.70	6.65
Safe Unlearning	0.64	3.84	13.86	2.09	4.14	15.50	16.24	7.52	10.82	4.71
Eraser	0.67	3.21	17.77	4.32	9.29	15.21	25.74	13.86	12.83	6.47
RMU	0.74	4.73	14.44	5.82	8.67	12.55	20.73	15.56	14.68	2.64
Circuit Breakers	0.83	5.68	15.71	6.73	3.38	18.12	25.23	7.64	10.20	7.47
CKU	0.71	6.29	14.65	2.85	5.76	9.98	16.05	10.59	15.64	3.30
SafeLLM	0.59	2.93	19.61	4.17	7.61	18.83	18.92	11.18	10.24	4.06
ACU	**0.41**	**2.45**	**7.27**	**1.48**	**3.12**	**5.76**	**7.15**	**5.02**	**4.64**	**1.55**

Table 15: HR by Backbone and Attack (%, ↓), Continued

Method	Direct	SORRY	DeepInc.	Flip	SATA	HEA	Query	Seq.	CoP	JailCon
Ministral-3-8B-Instruct										
Aligned model	0.59	6.12	21.29	6.64	9.30	17.65	28.62	14.35	14.39	7.74
Safe Unlearning	1.02	7.50	14.58	9.44	4.91	16.70	25.09	8.05	11.26	4.80
Eraser	0.23	4.48	17.83	2.05	9.52	14.32	33.02	9.97	12.74	6.05
RMU	0.70	4.61	12.00	3.83	3.73	14.92	18.52	9.23	11.23	4.17
Circuit Breakers	0.29	6.02	21.14	6.03	6.47	11.20	21.53	13.76	11.03	8.15
CKU	0.82	3.28	15.20	6.76	7.39	18.95	27.14	11.29	13.82	4.45
SafeLLM	0.38	2.70	19.18	8.42	9.09	15.52	14.55	11.17	11.61	3.55
ACU	**0.11**	**2.14**	**7.95**	**1.94**	**3.17**	**6.23**	**9.83**	**4.98**	**5.86**	**2.39**

Table 16: HR After Weight-Update Attacks (%, ↓)

Method	Llama3	Qwen2.5	Qwen3	Ministral3
MFT				
Aligned model	94.94	92.29	93.11	93.02
Safe Unlearning	53.35	46.08	72.11	51.76
Eraser	80.14	49.76	44.17	59.73
RMU	66.95	64.80	51.62	79.39
Circuit Breakers	48.71	69.86	50.65	61.82
CKU	53.05	57.67	55.47	44.14
SafeLLM	73.38	82.76	52.42	53.58
ACU	**14.38**	**13.14**	**22.74**	**12.26**
AvO				
Aligned model	95.48	96.45	93.17	95.95
Safe Unlearning	90.56	53.05	50.21	53.08
Eraser	60.73	43.50	60.94	49.82
RMU	53.73	55.30	54.50	63.55
Circuit Breakers	39.77	84.15	38.67	52.95
CKU	53.80	80.30	78.15	80.09
SafeLLM	65.14	65.23	57.20	67.30
ACU	**25.86**	**8.20**	**7.50**	**12.29**
TrojanPraise				
Aligned model	68.65	71.33	67.83	70.92
Safe Unlearning	47.58	45.86	32.23	56.76
Eraser	35.47	54.17	30.39	57.26
RMU	30.08	47.48	40.83	35.77
Circuit Breakers	46.64	31.33	50.62	62.83
CKU	40.05	51.80	45.32	33.47
SafeLLM	58.48	37.91	46.92	44.47
ACU	**26.03**	**27.70**	**22.00**	**17.35**

B.1 Seed Variability of ACU

Table 17 reports ACU means and sample standard deviations from the same three seeds as Table 15. Table 18 provides the corresponding baseline deviations. SD describes dispersion across seeds and is not a confidence interval for a method contrast.

Table 17: ACU HR Across Three Seeds (%, Mean ± SD)

Backbone	Direct	SORRY	DeepInc.	Flip	SATA	HEA	Query	Seq.	CoP	JailCon
Llama3	0.15±0.07	2.40±0.29	5.20±0.66	1.06±0.16	3.77±0.45	7.65±0.93	9.58±1.16	4.70±0.57	5.15±0.66	2.50±0.32
Qwen2.5	0.05±0.05	2.21±0.27	6.23±0.73	2.11±0.30	2.32±0.27	6.11±0.75	7.08±0.89	4.74±0.61	5.92±0.75	2.03±0.25
Qwen3	0.41±0.05	2.45±0.29	7.27±0.86	1.48±0.20	3.12±0.39	5.76±0.70	7.15±0.89	5.02±0.61	4.64±0.55	1.55±0.18
Ministral3	0.11±0.07	2.14±0.26	7.95±0.95	1.94±0.25	3.17±0.39	6.23±0.73	9.83±1.20	4.98±0.61	5.86±0.68	2.39±0.30

Table 18: Baseline HR Variability Across Seeds (SD, Percentage Points)

Backbone	Method	Direct	SORRY	DeepInc.	Flip	SATA	HEA	Query	Seq.	CoP	JailCon
Llama3	Aligned model	0.07	0.69	2.34	0.43	1.09	2.20	3.30	1.64	1.68	0.86
Llama3	Safe Unlearning	0.09	0.43	1.89	0.61	0.80	2.05	1.70	1.61	1.30	0.41
Llama3	Eraser	0.05	0.51	2.14	0.43	0.95	1.48	3.07	0.91	1.05	0.70
Llama3	RMU	0.07	0.42	1.57	0.34	0.66	2.14	2.57	1.14	1.43	0.55
Llama3	Circuit Breakers	0.07	0.77	1.52	0.43	0.61	1.61	2.52	1.93	1.73	0.36
Llama3	CKU	0.11	0.57	1.59	0.16	1.25	1.48	3.05	1.39	1.66	1.11
Llama3	SafeLLM	0.07	0.70	1.36	0.20	0.89	1.73	1.89	1.00	1.61	0.75
Qwen2.5	Aligned model	0.07	0.69	2.34	0.75	1.05	2.16	2.91	1.70	1.95	0.75
Qwen2.5	Safe Unlearning	0.11	0.68	1.84	0.36	0.86	1.93	2.68	1.36	2.02	0.70
Qwen2.5	Eraser	0.07	0.64	1.45	0.64	0.61	1.34	1.95	0.86	1.52	0.77
Qwen2.5	RMU	0.07	0.34	2.45	1.00	0.77	1.75	2.32	1.93	1.64	0.43
Qwen2.5	Circuit Breakers	0.05	0.30	1.89	0.27	1.14	1.93	2.61	1.57	1.98	0.55
Qwen2.5	CKU	0.07	0.71	1.57	0.75	0.45	2.25	2.80	1.52	1.23	0.43
Qwen2.5	SafeLLM	0.07	0.45	1.59	1.07	0.98	1.43	1.52	1.00	1.52	0.50
Qwen3	Aligned model	0.11	0.75	2.45	0.61	1.16	2.14	3.11	1.84	1.89	0.84
Qwen3	Safe Unlearning	0.09	0.46	1.68	0.27	0.50	1.86	1.98	0.93	1.32	0.57
Qwen3	Eraser	0.11	0.39	2.14	0.50	1.16	1.84	3.11	1.68	1.57	0.80
Qwen3	RMU	0.11	0.57	1.75	0.68	1.07	1.50	2.50	1.89	1.77	0.32
Qwen3	Circuit Breakers	0.11	0.68	1.89	0.82	0.43	2.20	3.05	0.91	1.25	0.93
Qwen3	CKU	0.11	0.76	1.80	0.39	0.70	1.20	1.91	1.27	1.86	0.43
Qwen3	SafeLLM	0.09	0.35	2.39	0.52	0.93	2.30	2.30	1.36	1.25	0.52
Ministral3	Aligned model	0.09	0.73	2.57	0.82	1.16	2.16	3.48	1.75	1.75	0.93
Ministral3	Safe Unlearning	0.16	0.90	1.75	1.16	0.59	2.02	3.00	0.95	1.39	0.61
Ministral3	Eraser	0.05	0.54	2.16	0.23	1.16	1.73	3.98	1.20	1.57	0.73
Ministral3	RMU	0.11	0.55	1.45	0.48	0.45	1.80	2.25	1.09	1.36	0.52
Ministral3	Circuit Breakers	0.07	0.72	2.55	0.75	0.80	1.39	2.61	1.66	1.34	1.02
Ministral3	CKU	0.09	0.39	1.84	0.84	0.93	2.27	3.27	1.39	1.64	0.55
Ministral3	SafeLLM	0.07	0.32	2.32	1.02	1.09	1.89	1.73	1.34	1.43	0.41

B.2 QWEN3-32B EVALUATION

After development selection, Qwen3-32B uses the same 440 test intents and 28 conditions as the main panel. ACU obtains 2.78% test HR, versus 11.75% for the aligned model and 7.32% for SafeLLM. Table 19 separates development and test aggregates; Tables 20 and 21 report utility and individual attacks. Test outputs do not affect training or selection.

Table 19: Qwen3-32B Development and Test HR (%). Test: mean $\pm$ SD.

Method	Development	Test macro HR
Aligned model	12.84	11.75$\pm$0.10
Safe Unlearning	8.73	8.98$\pm$0.06
Eraser	7.96	8.25$\pm$0.17
RMU	8.41	8.63$\pm$0.13
Circuit Breakers	7.52	7.83$\pm$0.08
CKU	8.18	8.47$\pm$0.02
SafeLLM	6.94	7.32$\pm$0.13
ACU	2.37	2.78$\pm$0.05

Table 20: Qwen3-32B Utility Accuracy (%, $\uparrow$)

Method	MMLU	HellaSwag	ARC-C	BoolQ	PIQA	WinoGrande	GSM8K
Aligned model	74.80	83.20	67.46	86.94	81.71	76.50	87.64
Safe Unlearning	74.27	80.12	62.82	84.15	78.42	73.54	78.92
Eraser	68.90	78.58	65.18	82.75	80.95	71.85	81.20
RMU	67.44	79.89	66.57	81.30	77.25	70.40	76.55
Circuit Breakers	70.86	83.02	63.25	84.85	79.19	73.94	83.17
CKU	66.61	80.71	65.55	85.20	80.07	74.82	84.36
SafeLLM	73.13	81.45	64.93	85.75	81.17	75.11	82.82
ACU	74.12	82.53	64.86	85.45	80.65	76.13	86.27

Table 21: Qwen3-32B HR by Attack (%, ↓)

Method	Direct	SORRY	DeepInc.	Flip	SATA	HEA	Query	Seq.	CoP	JailCon
Aligned model	0.61	6.05	20.03	4.62	8.89	17.12	24.36	14.17	15.26	6.36
Safe Unlearning	0.70	4.48	15.59	2.36	4.64	17.76	18.58	8.41	12.02	5.32
Eraser	0.52	2.43	13.47	3.24	7.11	11.64	19.30	10.35	9.50	4.97
RMU	0.64	4.21	12.47	5.17	7.35	10.35	18.05	13.44	12.36	2.30
Circuit Breakers	0.65	4.24	12.06	5.32	2.68	14.14	19.53	5.98	7.83	5.88
CKU	0.68	6.21	14.73	2.83	5.56	9.45	16.05	10.65	15.33	3.24
SafeLLM	0.42	2.19	14.73	3.12	5.70	14.18	14.02	8.08	7.77	2.97
ACU	0.30	1.77	5.26	1.05	2.23	4.12	5.08	3.59	3.33	1.09

B.3 Held-Out Harmful Intents

The strict split excludes the held-out category from all training-stage data: exposure, probe expansion and filtering, benign retention, and recovery. Normalized exact matches are removed, and semantic review rejects paraphrases and cross-category records containing the held-out intent. Test prompts are excluded from development and model selection. Each category contains 100 test intents, five completions per seed, and three seeds (1,500 outcomes). Tables 2 and 22 share the same Llama held-out results.

B.3.1 Cross-backbone category exclusion

The category-exclusion design applies independently to every backbone. The Llama row repeats Table 2; each remaining row uses the same category test intents and 500 responses per seed. The results preserve the main qualitative pattern: category exclusion weakens suppression but leaves harmful rates below those of the aligned checkpoint.

Table 22: Held-Out Category HR Across Backbones (%, ↓)

Backbone	Weapon	Physical harm	Biosecurity	Drug	Cyber intrusion
Llama3	4.87	5.27	4.13	6.47	5.40
Qwen2.5	4.67	5.07	4.40	5.60	5.00
Qwen3	4.93	5.40	4.80	5.73	5.07
Ministral3	5.20	5.60	4.93	5.87	5.40

B.4 Optimization-based Attacks and Statistical Evaluation

We evaluate query-only PAIR and gradient-access GCG against Llama3 and Qwen3. A fixed 200-intent subset of the main test collection is held out from all defense training and development. PAIR uses three independent streams of at most 20 target queries each; GCG uses a 20-token suffix, 250 update steps, and 128 candidate suffixes per step. Candidate generation and ranking operate on each defended checkpoint independently. The query budget excludes the attacker's own model and scoring calls; GCG's gradient steps are not treated as equivalent to PAIR queries.

For each intent and seed, we retain the highest-scoring search prompt and generate five fresh final responses with the common decoding settings. Search outputs are excluded from HR and ASR_5. Each row therefore has 3,000 scored responses and 600 intent–seed outcomes. HR averages response labels; ASR_5 counts an intent as successful if at least one final response is harmful. Both metrics are reported as mean and sample SD across seeds.

Table 23: Optimization-based Attack HR and ASR_5 (%, Mean ± SD)

		PAIR		GCG	
Backbone	Defense	HR	ASR_5	HR	ASR_5
Llama3	Aligned	35.10±0.62	63.00±1.32	41.93±0.90	68.83±3.51
Llama3	Safe Unlearning	23.97±0.95	46.83±2.02	25.57±1.50	49.50±2.78
Llama3	Circuit Breakers	19.13±0.40	41.67±1.26	22.47±0.15	45.50±1.80
Llama3	ACU	10.43±0.40	26.50±2.29	13.67±0.45	29.83±1.04
Qwen3	Aligned	34.50±0.69	60.00±3.00	38.43±1.79	65.83±1.44
Qwen3	Safe Unlearning	22.90±1.15	43.83±1.15	24.87±0.15	48.17±0.76
Qwen3	Circuit Breakers	17.53±0.58	39.00±1.00	20.23±0.58	39.83±3.33
Qwen3	ACU	8.80±0.36	22.17±1.26	12.03±0.57	30.00±1.00

On Llama3, ACU lowers PAIR HR relative to Circuit Breakers by 8.70 (95% CI [6.73, 10.67]) percentage points; the GCG contrast is 8.80 (95% CI [6.73, 10.90]) points. Qwen3 shows the same ordering. GCG leaves a higher harmful rate than PAIR for ACU, and intent-level success exceeds response-level rate. Thus the defense raises resistance under both access settings but does not eliminate successful Optimization-based attempts.

Paired inference. Intervals follow the intent-cluster procedure in Appendix A.1. The factorial development panel contains 190 clusters and its test panel 440, each with 25 responses per seed. Development and test intervals are computed separately. The test panel is evaluated only after selection, so its contrast is not selected by its observed test improvement.

Weight-update attacks. MFT, AvO, and TrojanPraise operate on the defended model with trainable rank-8 adapters and frozen base weights. Each attack receives 100 distinct training pairs, excluded from defense training and testing, and 100 optimizer updates with effective batch size eight. AdamW uses learning rate 2×10^{-5}, zero weight decay, and clipping at 1.0. The attack-specific loss or response transformation is retained, while adapter capacity, data count, and update budget are shared. We then score the 440 direct test questions with five responses per seed. These settings assess adapter-level adaptation and do not establish robustness to unrestricted full-parameter retraining.

C Controlled Comparisons and Stage Analysis

C.1 Matched-target comparison

Table 24: Llama3 Fixed-Target Controls After Recovery (%). HR: mean ± SD.

Objective	HR ↓	Utility ↑	FRR ↓	Degen. ↓
QF (fixed targets)	4.35±0.39	70.12	8.00	1.27
QD suppression	8.61±0.75	71.02	7.47	0.77
Safe Unlearning adaptation	8.80±0.66	70.64	8.43	1.17
Circuit Breakers adaptation	8.20±0.70	70.90	7.80	1.03
Recovery-only control	11.11±0.80	72.08	6.77	0.73
Development-selected utility match				
QF ($\lambda = 1$)	5.35±0.18	70.76	7.30	0.70
QD ($\lambda = 0.25$)	8.30±0.61	70.82	7.63	0.93

The objective comparison starts from the same exposed checkpoint with a fixed harmful-answer bank, benign pairs, and recovery set. The matched ACU, QD, Safe Unlearning, and Circuit Breakers adaptations use 1,000 updates, eight harmful and eight benign answers per update, and a 512-token answer cap. Short answers are retained. The variants replay identical harmful and benign answer records and their answer-position masks. QF/QD share token-likelihood normalization; equal record counts do not imply equal supervision scales across objectives. For QF/QD, $1,000 \times 16 \times 512 =$

8.192 million is an upper bound, not the realized count. Questions and template tokens add forward-pass cost, particularly for QD. Static-target and no-exposure rows compare pipeline variants, and recovery-only starts from the aligned model for 21 updates.

Table 25: Controlled HR Comparison (%, Mean ± SD) and Update Budgets

Variant	Llama3	Qwen2.5	Qwen3	Ministral3	Updates
ACU	4.35±0.39	4.11±0.34	4.23±0.50	4.65±0.34	1,000
QD suppression	8.61±0.75	8.35±0.70	8.48±0.66	8.89±0.70	1,000
Safe Unl. adaptation	8.80±0.66	8.55±0.73	8.68±0.50	9.11±0.75	1,000
CB adaptation	8.20±0.70	7.95±0.68	8.08±0.48	8.50±0.59	1,000
Static targets	6.80±0.48	6.55±0.50	6.68±0.64	7.11±0.66	1,000
No exposure	10.39±0.43	10.15±0.66	10.27±0.73	10.70±0.48	1,000
Recovery only	11.11±0.80	10.85±0.84	10.98±0.48	11.39±0.75	21

Every row uses the same 440 held-out intents and five completions per intent per seed (2,200 responses). Each intent receives one fixed attack wrapper, assigned by cycling the eight additional attack families over a fixed shuffled intent list. This balanced control panel differs from the main ten-column summary. Static targets are collected from the initial model; no exposure collects targets from the aligned model. Neither row is a fixed-answer comparison with the other objectives.

The paired Llama3 QF–QD comparison gives an HR reduction of 4.26 percentage points (95% intent-cluster bootstrap CI [3.20, 5.27]).

C.2 Factorial Ablations and Stage Analysis

Answer-side objective control. The fixed-target comparison replays identical harmful answers, benign records, batch order, and optimizer steps from a common exposed checkpoint. ACU and QD differ only in conditioning context. The Safe Unlearning and Circuit Breakers rows in this control adapt their objectives to the same data and update budget; the main comparison retains each method's own pipeline. All four fixed-target objectives use 1,000 updates and a 512-token answer cap. Forward-pass tokens are counted separately because question conditioning adds computation.

Exposure and target refresh. We cross exposure, refresh, and context in eight arms, first on development data and then on the disjoint test intents after freezing the configuration. All arms share the initial aligned model, probe ordering, 88,920-candidate collection budget, retain pool, and recovery set. Fixed-bank arms collect before suppression; iterative arms collect from the current model during chunk rounds. Extra calls in fixed-bank arms do not change training targets. Each arm takes 1,000 updates by resampling its own accepted occurrences with replacement. This budget-controlled schedule differs from the early-stopping pipeline.

Table 26: Llama3 Factorial Ablation: Development and Test HR (%, Mean ± SD)

Arm	Exposure	Targets	Context	Development	Test
1	No	Fixed	QD	15.88±1.54	16.65±0.45
2	No	Fixed	QF	13.66±0.36	14.48±0.37
3	No	Refresh	QD	13.59±1.35	15.14±0.43
4	No	Refresh	QF	13.28±0.45	14.51±0.46
5	Yes	Fixed	QD	12.32±0.08	13.33±0.34
6	Yes	Fixed	QF	4.95±0.34	6.09±0.44
7	Yes	Refresh	QD	11.74±0.45	12.87±0.35
8	Yes	Refresh	QF	2.51±0.24	3.58±0.41

The fixed-bank arms accept approximately 1,815 target occurrences without exposure and 22,367 with exposure in the initial full sweep. Counts include repeated answers. Iterative arms collect approximately 2,132 (QD) or 1,986 (QF) without exposure, and 30,198 (QD) or 16,235 (QF) with exposure over all chunk rounds. Because updates within a pass alter later target availability, iterative totals need not exceed the initial unsuppressed full-bank count. No arm borrows another arm's targets. An empty collection round skips target insertion and samples the arm's preceding bank; no

arm has an empty initial bank. Equal update counts control optimization volume while preserving differences in target availability and diversity.

Let $H_{e,r,o}$ be harmful rate for exposure e, refresh r, and objective o. We evaluate the interaction

$$I_o = (H_{1,0,o} - H_{1,1,o}) - (H_{0,0,o} - H_{0,1,o}). \tag{3}$$

On the independent test split, I_{QF} is 2.54 (95% CI [1.48, 3.60]) percentage points and I_{QD} is -1.05 (95% CI [-2.49, 0.29]); their difference is 3.58 (95% CI [2.03, 5.24]). The corresponding development contrasts are 2.06, -1.72, and 3.78 points. Under QF, refreshing targets reduces test HR from 6.09% to 3.58% with exposure, but changes 14.48% to 14.51% without exposure. Thus the additional refresh benefit persists on unseen intents and depends on exposure in this comparison; it is not a uniform benefit in every arm.

A benign-SFT sham exposure, matched to the exposure update count, yields 13.81±0.34% test HR with iterative QF (development: 12.86±0.22%). Additional benign updates alone therefore do not reproduce harmful-target exposure's benefit. These comparisons identify a pipeline interaction under the frozen budget; differences in collected target content and coverage remain part of that interaction, and no unique internal mechanism is inferred.

Target source. At a fixed exposed recipient checkpoint, we compare native, aligned-source, and Qwen3-generated answers on 400 common intents. Each bank has one accepted answer per intent; length-bin matching and the common 512-token cap reduce source-length differences. The same judge, retain records, 1,000-update schedule, and control evaluation panel are used. Native targets yield 4.35±0.39% HR, aligned-source targets 5.80±0.20%, and cross-model targets 6.53±0.16%. Target source affects the result even after fixing the recipient and sampling volume, although content differences remain part of this comparison.

Recovery. We save aligned, exposed, suppressed, and recovered checkpoints and evaluate the same 440-intent control panel, XSTest cases, and seven utility tasks at every stage. The recovery-only control is kept separate because it starts from the aligned model. Recovery uses the fixed 25 utility and 25 refusal pairs for 21 updates; neither audit prompts nor audit responses enter training.

Recovery changes HR by -0.65 (95% CI [-1.29, -0.03]) percentage points and FRR by -6.97 (95% CI [-8.90, -5.13]) points relative to suppression. Utility rises by 3.05 points, remaining 1.81 points below the aligned model. The safety contrast is small relative to the exposure-to-suppression change. Recovery primarily improves benign behavior while retaining most of the suppression benefit.

Table 27: Llama3 Utility Accuracy by Training Stage (%)

Stage	MMLU	Hella	ARC-C	BoolQ	PIQA	Wino	GSM8K
Aligned	64.17	75.80	55.64	84.05	78.68	72.00	74.79
Exposed	63.12	74.43	54.67	82.12	76.47	70.38	73.01
Suppressed	59.61	72.45	51.47	79.85	72.36	67.27	68.08
Recovered	62.36	74.71	53.44	84.07	75.20	69.53	73.14

Loss stability. We retain the fixed-target bank and 1,000-update schedule while sweeping $\lambda \in \{0, 0.25, 0.5, 1.0\}$. We also replace the negative-NLL term with $\mathcal{L}_{\mathrm{bnd}} = \mathbb{E}_a \log(1 + \exp[-(\ell_\theta(a) - \ell_{\theta_0}(a))])$, where θ_0 is the fixed exposed reference. This bounded-below loss downweights targets as their NLL increases. All variants receive the same recovery stage. On 200 benign prompts, degeneration means that a repeated four-gram accounts for more than 20% of generated token positions, measured before any postprocessing.

Table 28: Fixed-Target Loss Comparison (%). HR and degeneration: mean ± SD.

Loss	λ	HR ↓	Utility ↑	Degen. ↓
Negative NLL	0	3.67±0.07	64.18	11.67±0.87
Negative NLL	0.25	4.21±0.11	68.57	4.97±0.35
Negative NLL	0.5	4.35±0.39	70.12	1.27±0.45
Negative NLL	1.0	5.35±0.18	70.76	0.70±0.53
Bounded logistic	0.5	4.68±0.08	70.41	1.00±0.00

The low HR without retention accompanies a substantial utility loss and more repetition. Increasing λ reduces this damage but weakens harmful-target suppression. The bounded variant has slightly better utility and slightly higher HR than retained negative NLL. The choice $\lambda = 0.5$ is frozen using the development HR and benign-development criteria before scoring this table; the task scores shown here are outcomes, not selection inputs. The comparison does not establish a universally optimal retain weight.

C.3 Development ablations, exposure composition, and probe budget

Table 29 reports the development ablations used before the frozen test evaluation. The three 50-pair exposure sets vary in topical composition: Set A emphasizes violence, Set B cyber/fraud, and Set C samples topics uniformly. Table 30 reports their harmful-answer yields. All three expose harmful responses, with ranges of 9.2–10.9 percentage points on drugs, biosecurity, and self-harm, showing sensitivity to composition. Table 31 varies probe coverage and samples per query on the development panel. The sweep is non-monotonic: 25% coverage attains 2.58% HR versus 2.51% at full coverage, whereas 50% yields 3.12%. Reducing samples per query from five to three raises HR to 3.61%. Output counts in this table are per sweep; full collection costs sum over executed rounds as specified in Appendix A.3.

Table 29: Development Ablation HR (%, ↓)

Variant	Overall
ACU	**2.51**
w/o exposure	13.28
w/o iterative probe-and-suppress	4.95
w/ question-dependent suppression	11.74

Table 30: Harmful-Answer Yield by Exposure Set (%)

Probe category	Set A	Set B	Set C
Violence/weapons	93.6	91.9	92.7
Cyber/fraud	90.8	92.4	91.5
Illegal drugs	89.4	98.6	90.1
Biosecurity	86.9	97.8	87.2
Self-harm	84.1	93.5	94.8

Table 31: Probe-Budget Sensitivity: Development HR (%). Output counts are per sweep.

Probes	Samples	Outputs/sweep	Overall
10%	5	2.97K	4.37
25%	5	7.41K	2.58
50%	5	14.82K	3.12
100%	5	29.64K	**2.51**
100%	1	5.93K	4.18
100%	3	17.78K	3.61
100%	5	29.64K	**2.51**

D Additional Utility and Over-Refusal Results

Open-ended generation quality on OR-Bench and WritingBench is reported in Tables 32 and 33.

Table 32: OR-Bench Response Quality (1–5, ↑)

Model	Aligned	ACU
Llama3	4.36	4.31
Qwen2.5	4.33	4.30
Qwen3	4.29	4.27
Ministral	4.25	4.23

Table 33: WritingBench Response Quality (↑)

Model	Aligned	ACU
Llama3	5.28	4.94
Qwen3	4.12	3.96
Qwen2.5	4.45	4.52
Ministral	3.89	3.88

D.1 Utility and context sensitivity

Table 34 reports the seven standard utility tasks across all four 7–8B backbones (with Llama3 baseline breakdowns in main-text Table 8). Percentages use task-specific metrics and are not pooled across unequal test sets. Context-sensitive refusal and coherence are assessed separately in Table 35 to capture effects that standard task accuracy can miss.

Table 34: Utility Accuracy Across Backbones (%). Hella: HellaSwag; Wino: WinoGrande.

Backbone	State	MMLU	Hella	ARC-C	BoolQ	PIQA	Wino	GSM8K
Llama3	Aligned	64.17	75.80	55.64	84.05	78.68	72.00	74.79
Llama3	ACU	62.36	74.71	53.44	84.07	75.20	69.53	73.14
Qwen2.5	Aligned	65.39	77.91	58.29	85.41	80.02	73.05	76.06
Qwen2.5	ACU	63.77	76.41	56.52	84.32	78.45	71.59	74.19
Qwen3	Aligned	69.05	80.16	61.46	85.91	80.24	74.84	80.09
Qwen3	ACU	68.08	78.60	59.63	85.13	78.73	73.23	78.15
Ministral3	Aligned	67.13	77.21	58.43	84.36	79.33	73.28	77.03
Ministral3	ACU	65.91	75.37	56.66	83.64	77.54	71.87	74.84

Table 35: Response Quality and Context Sensitivity. FRR: %; coherence: 1–5.

Backbone	OR quality	Writing quality	Context FRR	Coherence
Llama3	4.31	4.94	9.07	4.19
Qwen2.5	4.30	4.52	8.80	4.21
Qwen3	4.27	3.96	8.13	4.20
Ministral3	4.23	3.88	7.87	4.19

The context evaluation uses 100 benign prompts containing safety-sensitive terminology, five responses per prompt, and three replicates: 1,500 refusal decisions and 1,500 ordinal coherence ratings per backbone. OR quality is on a 1–5 scale, whereas WritingBench scores use a 1–10 scale; these columns are not directly comparable.

E Mechanism and Target-Coverage Diagnostics

E.1 Local transfer and gradient support

The gradient measurements help explain this transfer. Pairing 50 harmful answers with 20 questions each, we compare the top-100 MLP neurons ranked by gradient norm Dai et al. (2022). The mean Jaccard overlap between QF and QD neuron sets rises from 0.359 at the first answer token to 0.541 at 32 tokens. Directional diagnostics complement this shared-support measurement: on Llama3, native-target cosine similarity rises from 0.08 to 0.27, and a normalized answer-only ascent step increases conditional test NLL by 0.38 nats/token at 32 tokens, versus 0.07 for cross-model targets. The four-backbone diagnostics and five additional concepts are reported in Appendix E. These measurements establish local transfer on the evaluated answers; generated-response experiments below test whether suppression also reduces harmful alternatives.

The answer-side objective still needs targets that represent the model's reachable harmful outputs. Exposure makes those outputs easier to sample, and refreshing them follows the distribution as suppression changes it. To see why target choice matters, let $g_\tau = \nabla_\theta \ell_\theta(\tau)$ be the target gradient and g_{ret} the benign retain gradient. A motivating SGD displacement and its first-order effect on a test trajectory τ' are

$$d = \eta(g_\tau - \lambda g_{\mathrm{ret}}), \qquad \ell_{\theta+d}(\tau') - \ell_\theta(\tau') \approx g_{\tau'}^\top d. \tag{4}$$

Transfer therefore depends on alignment with the full update, including retention. For AdamW, d includes the optimizer moments and clipping. Appendix E.5 gives the local smoothness bound.

E.2 Localizing bomb-making Neurons

Computing contribution scores. We localize concept-associated MLP neurons using 1,095 token positions from 50 question-answer pairs and an equal-sized control-token set. Candidate neurons

are down-projection columns from all transformer blocks. We rank jointly across layers and draw random controls with matching per-layer counts. For neuron i in layer ℓ, the direct logit-contribution proxy is

$$s_{\ell,i,j} = h_{\ell,i}(t_{\text{context}}) \cdot \langle W^{\text{down}}_{\ell,:,i},\, W^{\text{LM}}_{a_j} \rangle, \tag{5}$$

Here $h_{\ell,i}$ is the activation entering the down projection, $W^{\text{down}}_{\ell,:,i}$ its residual-stream write vector, and $W^{\text{LM}}_{a_j}$ the LM-head row. This direct projection is a proxy that omits downstream transformations. Let $b_{\ell i}$ and $c_{\ell i}$ be the mean absolute proxy scores on concept and control tokens. Rank by $(b_{\ell i} - c_{\ell i})/(b_{\ell i} + c_{\ell i} + 10^{-8})$, avoiding division by a signed near-zero control mean. Enrichment uses column-gradient norms and an equal-sized, layer-matched random set. The proxy is used for localization; the intervention comparison below tests behavioral sensitivity to the selected columns.

Causal intervention. To validate whether the proposed contribution score identifies neurons with causal influence, we conduct an intervention experiment at inference time. For a selected neuron set S, we suppress its MLP contribution by modifying the down-projection output at layer ℓ as

$$o_\ell \leftarrow o_\ell - \alpha \sum_{i \in S_\ell} h_{\ell,i} W^{\text{down}}_{\ell,:,i}, \tag{6}$$

where α is the suppression scale. We compare three decoding conditions: no intervention, suppressing the top-K bomb-related neurons, and suppressing K layer-matched random neurons. Each condition is evaluated on 100 bomb-making prompts and 100 benign prompts, with GPT-5 as the judge. For bomb-making prompts, it rates bomb relevance on a 1–5 scale; we report B_{Drop} and R_{Drop}, the relevance drop when suppressing bomb-specific and random neurons, respectively. For benign prompts, it rates coherence on the same scale; we report C_{Drop}, the coherence drop when suppressing bomb-specific neurons.

Table 36: Neuron-Intervention Effects: Score Drops on 1–5 Scales

Top-k	Scale	B_{Drop}	R_{Drop}	C_{Drop}
60	1.0	1.85	0.00	0.40
60	1.5	**3.00**	0.05	0.45
120	1.0	1.70	0.05	0.75
120	1.5	3.10	1.05	0.95

Table 36 shows a trade-off: top-60 at scale 1.5 reduces concept relevance by 3.00 points and benign coherence by 0.45, whereas top-120 produces a slightly larger relevance drop with greater benign degradation. The selective intervention has a larger relevance effect than the layer-matched random control. However, relevance and coherence are ordinal response properties, not direct estimates of harmfulness or proof that a concept has been erased.

E.3 Additional Concept-Level Gradient Alignment

We repeat the prefix-length diagnostic on five additional harmful concepts. Each harmful answer is paired with diverse eliciting questions; the table reports mean Jaccard overlap between answer-only and question-conditioned top-gradient-neuron sets. Overlap increases monotonically across the tested prefix lengths for all five concepts.

Table 37: Concept-Level Gradient Alignment: Jaccard Overlap by Prefix Length

Concept	1	4	8	16	32
Cyber exploitation	.414	.452	.461	.465	.528
Illegal drug production	.368	.415	.442	.489	.664
Biosecurity misuse	.216	.272	.345	.452	.512
Self-harm facilitation	.371	.403	.461	.493	.547
Fraud and safeguard evasion	.338	.379	.397	.416	.439

The Jaccard values quantify shared top-neuron gradient support. Table 38 reports direction and one-step conditional NLL change in separate column groups. Together these diagnostics characterize

transfer on the evaluated concepts; they do not establish prompt-independent safety or convergence of the full gradient vectors.

E.4 Directional gradient diagnostic

The directional diagnostic uses cumulative mean token NLL over the first 1, 16, or 32 answer tokens. All gradients within a backbone are evaluated at the same recipient checkpoint in the same adapter coordinates; source labels change answer provenance, not parameter space. An isolated answer-only ascent displacement has norm 10^{-3}; the corresponding test loss conditions on its eliciting question. Retention and AdamW are excluded from this direction probe. Table 38 reports 30 answer-subset replicates. Cosine and NLL changes are complementary summaries, without an asserted one-to-one relation between their magnitudes.

Table 38: Gradient Cosine Similarity and NLL Change (Nats/Token), Mean ± SD. Subscripts: prefix length; random: layer-matched.

		Cosine similarity			NLL change		
Backbone	Control	$\cos_1$	$\cos_{16}$	$\cos_{32}$	$\Delta\ell_1$	$\Delta\ell_{16}$	$\Delta\ell_{32}$
Llama3	Self	0.08±0.03	0.21±0.03	0.27±0.03	0.12±0.03	0.32±0.03	0.38±0.04
Llama3	Aligned source	0.04±0.02	0.10±0.03	0.15±0.02	0.06±0.05	0.15±0.04	0.21±0.04
Llama3	Cross-model	0.01±0.03	0.04±0.03	0.05±0.02	0.02±0.04	0.06±0.03	0.07±0.04
Llama3	Random	0.00±0.02	0.02±0.02	0.01±0.03	0.02±0.04	0.02±0.04	0.02±0.04
Qwen2.5	Self	0.08±0.03	0.20±0.03	0.26±0.03	0.12±0.04	0.31±0.04	0.35±0.04
Qwen2.5	Aligned source	0.03±0.03	0.09±0.03	0.14±0.03	0.06±0.05	0.14±0.04	0.19±0.04
Qwen2.5	Cross-model	0.01±0.03	0.04±0.02	0.05±0.03	0.04±0.05	0.07±0.04	0.07±0.04
Qwen2.5	Random	-0.01±0.02	0.01±0.03	0.01±0.02	0.02±0.05	0.02±0.04	0.02±0.03
Qwen3	Self	0.07±0.02	0.20±0.02	0.25±0.02	0.11±0.04	0.30±0.04	0.34±0.04
Qwen3	Aligned source	0.04±0.02	0.10±0.02	0.13±0.02	0.05±0.03	0.15±0.04	0.19±0.05
Qwen3	Cross-model	0.02±0.02	0.03±0.03	0.05±0.02	0.02±0.05	0.06±0.04	0.08±0.05
Qwen3	Random	-0.00±0.03	0.01±0.02	0.01±0.02	0.03±0.04	0.02±0.04	0.01±0.05
Ministral3	Self	0.08±0.02	0.19±0.02	0.23±0.02	0.11±0.04	0.29±0.04	0.33±0.04
Ministral3	Aligned source	0.04±0.03	0.09±0.03	0.12±0.03	0.05±0.04	0.13±0.04	0.18±0.04
Ministral3	Cross-model	0.02±0.02	0.04±0.03	0.04±0.02	0.04±0.04	0.03±0.05	0.09±0.04
Ministral3	Random	0.00±0.03	0.01±0.03	0.00±0.02	0.01±0.03	0.01±0.04	0.02±0.03

E.5 Local Transfer Bound

Assume that the test-trajectory mean NLL has an L-Lipschitz gradient along the update segment from θ to $\theta + d$. Taylor's theorem with an integral remainder gives

$$\left|\ell_{\theta+d}(\tau') - \ell_\theta(\tau') - g_{\tau'}^\top d\right| \le \frac{L}{2}\|d\|_2^2. \tag{7}$$

Indeed, the remainder equals $\int_0^1 [\nabla \ell_{\theta+ud}(\tau') - g_{\tau'}]^\top d\, du$; Lipschitz continuity and Cauchy–Schwarz bound its magnitude by $\int_0^1 Lu\|d\|^2 du$. Thus $g_{\tau'}^\top d > L\|d\|^2/2$ is sufficient for a positive one-step NLL change. For SGD, $d = \eta(g_\tau - \lambda g_{\mathrm{ret}})$; for AdamW it includes moment estimates and clipping. No L is estimated here, so this is a standard conditional local bound, not a certified result for the checkpoints or a proof of distribution-wide unlearning.

E.6 Target Likelihood and Harmful-Event Probability

For a fixed test question q, let $\mathcal{H}(q)$ be a fixed set of harmful completions under a stated annotation rule, and $\mathcal{A}(q) \subseteq \mathcal{H}(q)$ the already identified targets. Then

$$P_\theta(\mathcal{H}(q) \mid q) = \sum_{a \in \mathcal{A}(q)} p_\theta(a \mid q) + \sum_{a \in \mathcal{H}(q) \setminus \mathcal{A}(q)} p_\theta(a \mid q). \tag{8}$$

Reducing every term in the first sum does not control the second sum. Moreover, ACU trains under $q_\emptyset$, so even reduction of the first sum under a particular q requires cross-context transfer. These are two separate uncertainties: update transfer and target coverage. Exposure and refresh are intended to address coverage; the local gradient diagnostic addresses transfer. Neither establishes that the entire harmful event has lower probability.

The coverage audit uses 500 intents excluded from training and development, with five responses per seed (7,500 outputs). Alternate phrasings test previously probed behavior types. A fixed request-aware rubric clusters harmful answers by goal, action, and outcome, ignoring wording-only changes. Discovery counts answers in clusters absent from earlier collection rounds. Table 7 separates collection discoveries from fixed-set audit HR.

The first pass defines the initial inventory. Later passes discover additional clusters while audit HR declines, supporting target refresh. Remaining audit positives show that local stopping does not establish uniform suppression, even within previously probed behavior types.

Response rate versus attempt success. For N intents with exactly K attempts each, let $h_{ik} \in \{0, 1\}$ indicate harmful success, $r = (NK)^{-1} \sum_{i,k} h_{ik}$, and $s_K = N^{-1} \sum_i \mathbf{1}\{\sum_k h_{ik} > 0\}$. Without assuming independence, counting successes within this panel gives

$$r \leq s_K \leq \min\{Kr, 1\}. \tag{9}$$

F Answer-Side Transfer at Matched Capability

Checkpoint identity and scoring. For each seed, the fixed-target QF row and the negative-NLL $\lambda = 0.5$ row of Table 28 refer to the same exposed recipient, bank, optimizer trajectory, and post-recovery checkpoint. Thus the 70.12 utility and 1.27% degeneration values are shared results; the complete-pipeline 70.35 utility is a different checkpoint. QD and both adaptations are evaluated at their own post-recovery checkpoints on the identical seven utility tasks, 200 XSTest prompts, and 200 benign degeneration prompts. Both benign generation panels use five completions and three seeds, giving 3,000 responses each. Utility is a seven-task macro average; generation rates are pooled.

Table 39: Fixed-Target Safety, Utility, and Generation Quality (%)

Backbone	Objective	HR	Utility	FRR	Degen.
Llama3	QF	4.35	70.12	8.00	1.27
Llama3	QD	8.61	71.02	7.47	0.77
Llama3	Safe Unlearning adaptation	8.80	70.64	8.43	1.17
Llama3	Circuit Breakers adaptation	8.20	70.90	7.80	1.03
Qwen2.5	QF	4.11	71.84	7.60	1.20
Qwen2.5	QD	8.35	72.61	7.03	0.73
Qwen2.5	Safe Unlearning adaptation	8.55	72.23	8.07	1.13
Qwen2.5	Circuit Breakers adaptation	7.95	72.48	7.43	0.97
Qwen3	QF	4.23	74.08	7.30	1.10
Qwen3	QD	8.48	74.81	6.83	0.70
Qwen3	Safe Unlearning adaptation	8.68	74.42	7.77	1.03
Qwen3	Circuit Breakers adaptation	8.08	74.66	7.10	0.93
Ministral3	QF	4.65	72.02	8.23	1.37
Ministral3	QD	8.89	72.80	7.63	0.83
Ministral3	Safe Unlearning adaptation	9.11	72.43	8.70	1.27
Ministral3	Circuit Breakers adaptation	8.50	72.67	8.00	1.13

Utility matching rule. Before test scoring, we define a practical tolerance of 0.25 percentage points in seven-task macro accuracy relative to a common development target of 70.75%. A disjoint development panel contains 100 items per task, excluded from defense training and final task evaluation; accuracy is averaged over the three seed checkpoints. We choose the lowest-development-HR configuration inside the tolerance, using retain weights $\{0.25, 0.5, 1, 2\}$ and learning rates $\{5 \times 10^{-6}, 10^{-5}, 2 \times 10^{-5}\}$. Bank, updates, and recovery are unchanged. Selection uses development accuracy and HR only; the final task scores are not used to retune. All selected test macro accuracies also lie within the stated tolerance. This is a practical matching rule, not an equivalence claim inferred from a non-significant test.

Table 40: Llama3 Utility-Matched Controls After Recovery (%)

Objective	Retain weight	Dev. utility	Test utility	HR	FRR	Degen.
QF	1.0	70.71	70.76	5.35	7.30	0.70
QD	0.25	70.81	70.82	8.30	7.63	0.93
Safe Unlearning adaptation	1.0	70.67	70.69	9.04	7.77	1.03
Circuit Breakers adaptation	1.0	70.86	70.88	8.46	7.37	0.93

The QF point reproduces the $\lambda = 1$ checkpoint in Table 28. QD uses a smaller retain weight than in the primary fixed-order control. Its utility falls by 0.20 points while HR falls from 8.61% to 8.30%. QF's HR advantage over QD is 2.95 points at a 0.06-point utility difference. Matching a macro average cannot rule out task-specific deficits, and the primary equal-weight, fixed-order control remains the conditioning-only comparison.

F.1 Question-coverage protocol

The bank contains 400 unique native answers, one per training intent. For each answer, eight eligible questions are prepared before optimization; $k = 1, 2, 4, 8$ selects nested prefixes of this list. Every condition visits the same answer and benign-record indices in the same 1,000 updates. The QD question cycles uniformly through the available k forms at each answer visit. No condition adds answers or updates when k grows. QF is trained once per seed using the empty user message and serves as one shared reference. All conditions use the same exposed checkpoint, $\lambda = 0.5$, 512-token answer cap, and recovery set.

The seen panel contains one base training question per bank intent; the new-wording panel contains one independently withheld reformulation of each of these 400 intents, outside the full eight-form training set. The new-attack panel applies withheld attack expressions to those same 400 intents, cycling the eight additional attack families with 50 intents per family. All such expressions are excluded from every training-question set. These three panels deliberately retain the same intents to isolate question-form transfer; they are within-intent diagnostics, not held-out-intent evaluations. Every arm receives the same exact prompts. Five completions and three seeds give 6,000 responses per panel. The separate 440-intent control and held-out-category evaluations test intent generalization.

Table 41: Question-Coverage Counts and Forward-Token Costs. Counts span three seeds; tokens are per seed.

Context	Seen $h/6000$	Wording $h/6000$	Attack $h/6000$	FRR $/3000$	Degen. $/3000$	Tokens (M)
QD, 1	153	344	551	224	23	5.456
QD, 2	141	291	485	229	26	5.496
QD, 4	134	251	431	228	25	5.576
QD, 8	142	243	397	231	28	5.696
QF	175	220	284	240	38	4.832

Each condition processes 8,000 harmful and 8,000 benign answer occurrences. Mean supervised lengths are 251 and 235 tokens, giving 3.888 million supervised answer tokens. The shared template and benign-question contribution is 0.944 million tokens. Mean harmful-question lengths are 78, 83, 93, and 108 tokens for the four QD conditions, adding 0.624, 0.664, 0.744, and 0.864 million tokens. These non-padding counts distinguish answer supervision from context cost; wall time can additionally depend on padding, kernels, and batching.

G Exposure Recipient and Target-Source Control

We hold one exposed-source bank of 400 answers fixed and initialize two recipients from the aligned and exposed versions of the same backbone. Both receive identical answers, retain records, order, 1,000 updates, and recovery. Neither generates a replacement bank during this comparison. HR uses the 440-intent control panel; benign metrics follow Appendix F. The exposed-recipient row repeats the fixed-target QF checkpoint.

Table 42: Llama3 Recipients With Fixed Exposed-Source Targets (%). HR: mean ± SD.

Recipient	HR ↓	Utility ↑	FRR ↓	Degen. ↓
Aligned	4.76±0.41	70.82	7.60	1.03
Exposed	4.35±0.39	70.12	8.00	1.27

The aligned recipient benefits from targets supplied by exposure without receiving exposure updates itself. The exposed recipient has 0.41-point lower HR but also 0.70-point lower utility. A paired intent-cluster interval for the HR difference is $[-0.16, 0.98]$ points; this does not establish equivalence, nor does it demonstrate that recipient exposure is necessary. Target availability and target provenance remain supported roles of exposure.

H Behavioral Recovery Endpoints

Labels and denominators. The stage audit uses the same aligned, exposed, suppressed, and recovered checkpoint for each seed as Table 43. On the 6,600 harmful-request responses, the actionable unsafe-compliance label is exactly the existing HR endpoint, not a second independent safety measurement. Correct refusal requires an explicit, appropriate refusal without actionable assistance. Safe non-refusal, irrelevant output, and degeneration occupy a residual class; consequently, correct refusal is not $100-\mathrm{HR}$. FRR is scored on 3,000 benign XSTest responses. Benign degeneration uses a separate fixed 200-prompt panel with 3,000 responses and the four-gram rule in Appendix C.2.

Table 43: Llama3 Stage Comparison (%). HR/UC and FRR: mean ± SD. UC: unsafe compliance; CR: correct refusal.

Checkpoint	HR/UC ↓	CR ↑	Utility ↑	FRR ↓	Degen. ↓
Aligned	11.33±0.48	82.03	72.16	6.83±0.65	0.70
Exposed	64.09±3.99	26.26	70.60	5.07±0.06	1.43
Suppressed	3.65±0.32	79.94	67.30	14.40±0.44	8.23
Recovered	3.00±0.28	90.06	70.35	7.43±0.95	0.97

Final state relative to alignment. We predefine non-inferiority tolerances before evaluating final checkpoints: at most one additional percentage point of unsafe compliance or benign FRR, at most a two-point loss of correct refusal, at most 0.5 additional points of degeneration, and at most a 0.5-point utility loss. The safety tolerances constrain absolute added failures, rather than percentages of the aligned failure rate. Paired 95% intervals resample intents for harmful endpoints, benign prompts for benign endpoints, and questions within each task for utility, retaining all responses and seeds in each cluster. Utility contrasts average task-wise differences equally. These are endpoint-specific intervals, not a simultaneous certificate of overall safety restoration.

Table 44: Recovered-Minus-Aligned Differences and 95% CIs (Percentage Points)

Endpoint	Difference	95% CI	Acceptable threshold
Unsafe compliance/HR	-8.33	$[-9.29, -7.42]$	$\leq +1.00$
Correct refusal	$+8.03$	$[6.57, 9.50]$	≥ -2.00
Benign FRR	$+0.60$	$[-0.23, 1.43]$	$\leq +1.00$
Benign degeneration	$+0.27$	$[-0.13, 0.67]$	$\leq +0.50$
Utility	-1.81	$[-2.34, -1.28]$	≥ -0.50

Compared with the aligned checkpoint, the recovered model reduces unsafe compliance by 8.33 percentage points and increases correct refusal by 8.03 points. Benign FRR and degeneration increase by 0.60 and 0.27 points, respectively, while utility decreases by 1.81 points. Table 44 reports the paired intervals and prespecified thresholds for these changes.

What the last stage causes within this pipeline. The suppressed-to-recovered comparison retains the HR change of -0.65 points (95% CI $[-1.29, -0.03]$), FRR change of -6.97 ($[-8.90, -5.13]$),

and utility gain of 3.05. Endpoints show a 10.12-point increase in correct refusal (95% CI $[8.60, 11.63]$) and a 7.27-point decrease in benign degeneration ($[5.83, 8.80]$ for the decrease). Suppression already removes most of the exposure-associated unsafe compliance; recovery improves refusal quality and benign behavior while preserving that reduction. Neither this attribution nor the final-state comparison assigns the entire safety improvement to the last stage.

I Related Work

Safety unlearning and jailbreak defenses. Machine unlearning removes unwanted behaviors while preserving utility Eldan & Russinovich (2024); Maini et al. (2024); Li et al. (2024a). Recent safety unlearning defenses use fixed question-answer corpora Zhang et al. (2024); Lu et al. (2024), representation redirection Li et al. (2024a), selective neuron masking Shi et al. (2025), or localized FFN editing Li et al. (2025). Safe Unlearning combines adaptive forgetting with refusal and benign-answer learning Zhang et al. (2024), while Circuit Breakers uses response-based representation rerouting Zou et al. (2024). ACU uses an answer-only NLL objective and focuses on acquiring model-native suppression targets through exposure and iterative refresh. The suppression loss omits the eliciting questions; the probe stage still depends on eliciting-query coverage.

Representation control and model editing. Our approach also intersects with representation engineering Zou et al. (2023a), circuit breaking Zou et al. (2024), and refusal-direction steering Arditi et al. (2024), as well as factual model editing targeting key-value memories Geva et al. (2021); Meng et al. (2022; 2023). Unlike these methods, ACU suppresses native harmful generation trajectories with an answer-only loss and iteratively adapts its target set as the model's output distribution shifts.

J Ethical Considerations

The method studies defensive safeguards against jailbreaks and malicious model adaptation. Exposure temporarily relaxes refusal behavior, so intermediate checkpoints and generated harmful responses require controlled access. Reporting focuses on aggregate outcomes and algorithmic mechanisms. Evaluating both harmful assistance and benign over-refusal is essential: a defense that broadly suppresses legitimate discussion would impose a different, unacceptable utility cost.