WHEN SAFETY FAILS COLLECTIVELY: ROBUST ALIGNMENT AGAINST HARMFUL FINE-TUNING VIA JOINT WEIGHT PERTURBATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Harmful fine-tuning can rapidly erode the safety alignment of large language models, creating substantial risks when aligned models are adapted to downstream tasks. Existing alignment-stage defenses use harmful examples and simulated harmful perturbations during safety alignment to prepare models for later harmful fine-tuning, yet the collective effects of harmful weight perturbations remain underexplored. We identify a collective failure pattern: disjoint parts of a harmful perturbation have limited effects in isolation, but their joint application increases the model's harmfulness beyond the sum of their individual effects. Guided by this observation, we propose Joint Weight Perturbations (JWP), an alignment-stage defense that masks different parts of a harmful perturbation estimated from the current model and trains the model to respond safely in the resulting states. This exposes the model to different combinations of harmful perturbations and reduces its sensitivity to their accumulation during subsequent harmful fine-tuning. Experiments across downstream tasks and model families show that JWP offers stronger resistance to harmful fine-tuning than prior defenses while largely preserving task performance. Our code is available at `https://anonymous.4open.science/r/JWP-D977`.

1 INTRODUCTION

Safety alignment is a key safeguard for large language models (LLMs), but downstream fine-tuning can erode the safety it establishes (Qi et al., 2024; Bowen et al., 2025). After an aligned model is released with open weights (Tamirisa et al., 2025) or offered through fine-tuning-as-a-service (Huang et al., 2025c), it may be customized using datasets supplied for downstream tasks. These datasets can include harmful examples, sometimes inadvertently, even when the intended task is benign (Bowen et al., 2025). Prior work shows that fine-tuning on only a small amount of such data can substantially degrade model safety (Qi et al., 2024; Hsu et al., 2024). The customized model may nevertheless perform well on the downstream task while becoming less safe, so safety measured at the end of alignment does not guarantee safety after further training (Huang et al., 2025c; Nguyen et al., 2026). This harmful fine-tuning threat motivates defenses that remain effective through downstream adaptation.

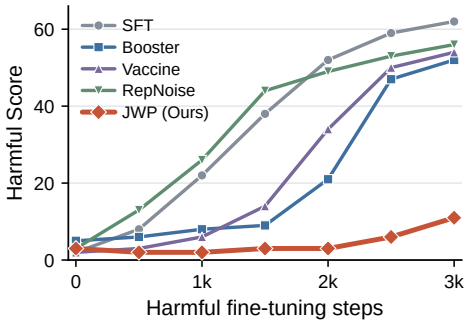

Figure 1: Safety degradation of aligned models under harmful fine-tuning.

A line of recent work addresses this threat at the alignment stage, incurring a one-time cost to make the aligned model harder to corrupt in all subsequent fine-tuning requests (Huang et al., 2025c; 2026; Nguyen et al., 2026; Tamirisa et al., 2025). Vaccine trains hidden embeddings to withstand perturbations (Huang et al., 2024b), RepNoise degrades the internal representations of harmful data (Rosati et al., 2024), and Booster attenuates the harmful-loss reduction induced by a simulated harmful update (Huang et al., 2025c). Nevertheless, our evaluation shows that the protection they provide is

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.

limited. We perform safety alignment on Llama-2-7B using each method and track the harmful score (HS) of the resulting models throughout sustained harmful fine-tuning, as shown in Figure 1. The HS of every evaluated defense rises rapidly as the attack proceeds. Booster postpones the onset of this rise relative to Vaccine and RepNoise, yet safety still collapses after a comparatively small number of harmful updates. This consistent pattern across methods with distinct designs suggests that the safety degradation is not merely a matter of alignment strength, but reflects a common limitation in how existing defenses account for the harmful weight perturbations induced by downstream harmful fine-tuning.

Specifically, existing defenses simulate harmful weight perturbations during alignment to prepare the model for subsequent harmful fine-tuning (Huang et al., 2025c; Tamirisa et al., 2025), but typically treat these perturbations as a whole, leaving their internal interactions largely unexplored. To investigate this, we partition a harmful weight update into disjoint subsets and conduct two weight-level analyses on the same aligned model. First, we compare applying each subset individually with applying the subsets jointly. The joint application yields a substantially higher harmful score than the individual subsets, exceeding what their isolated effects would imply. Second, we vary the fraction of the harmful update applied to the model. Applying a small fraction of the perturbation leaves model safety largely intact, but increasing this fraction leads to a sharp rise in the harmful score and rapid safety degradation. Together, these observations suggest a collective failure pattern: interactions among different parts of a harmful perturbation amplify safety degradation.

Guided by these observations, we propose Joint Weight Perturbations (JWP), an alignment-stage defense that trains the model to maintain safety alignment under different combinations of harmful weight perturbations. JWP first estimates a harmful weight perturbation by simulating multiple steps of harmful fine-tuning from the current model. It then samples a retention probability $p \sim \mathcal{U}(0, 1)$ and masks the perturbation accordingly to construct a partially perturbed model state. The model is trained to produce safe responses in this state. Training across these states exposes the model to different combinations of harmful weight perturbations and aims to reduce its sensitivity to their joint effects during subsequent harmful fine-tuning. Experiments show that JWP effectively suppresses the joint amplification of harmful weight perturbations. Under sustained harmful fine-tuning, JWP achieves a 4.3–$5.4\times$ reduction in average harmful score across four downstream tasks compared with existing defenses including Booster (Huang et al., 2025c), while largely preserving utility after safety alignment and downstream task performance.

Our contributions are summarized as follows:

- We identify a collective failure pattern through weight-level analyses, showing that different parts of a harmful perturbation can reinforce one another and amplify safety degradation.
- We propose JWP, an alignment-stage defense that trains the model under randomly masked harmful perturbations of varying sizes and compositions to suppress their joint amplification during subsequent harmful fine-tuning.
- Experiments across downstream tasks and model families demonstrate that JWP substantially improves resistance to sustained harmful fine-tuning while largely preserving utility after safety alignment and downstream task performance.

2 Related Work

Harmful fine-tuning. Safety alignment can be removed by fine-tuning on a small number of harmful examples (Bowen et al., 2025), and even benign fine-tuning data can degrade safety (Qi et al., 2024; Guan et al., 2025). Unlike prompt-level jailbreaks (Zou et al., 2023), such attacks directly modify the model weights. Recent evaluations further show that defenses validated under one attack configuration can fail when the number of fine-tuning steps or the learning rate is changed (Qi et al., 2025b), which motivates evaluating safety along the complete trajectory of harmful fine-tuning rather than at a single endpoint.

Alignment-stage defenses. Alignment-stage defenses modify the alignment procedure so that the released model is intrinsically harder to corrupt, at a one-time cost. Vaccine perturbs hidden embeddings to mitigate harmful embedding drift (Huang et al., 2024b), and T-Vaccine restricts such perturbations to safety-critical layers (Liu et al., 2025). RepNoise pushes the representations of

harmful data toward random noise (Rosati et al., 2024). Booster regularizes the harmful-loss reduction after a simulated harmful gradient step (Huang et al., 2025c), and TAR trains safeguards against simulated multi-step tampering trajectories via meta-learning (Tamirisa et al., 2025). Appendix C further compares JWP with TAR, focusing on how the two methods construct model states for safety alignment. These defenses act on hidden representations or on aggregate simulated perturbations and trajectories, without constructing different partial assemblies of one learned harmful weight update. JWP decomposes a fitted harmful update into components and optimizes safety across its partially applied states, thereby targeting the joint effect among harmful weight perturbations.

Defenses at other stages. Another line of work intervenes during or after downstream fine-tuning. Fine-tuning-stage methods mix safety data into user data (Huang et al., 2024a; Wang et al., 2024; Eiras et al., 2025), constrain updates on initial response tokens (Qi et al., 2025a; Peng et al., 2025), or steer updates away from safety-relevant subspaces (Li et al., 2025). Post-fine-tuning methods repair the adapted model through subspace projection (Hsu et al., 2024), pruning of harmful weights (Huang et al., 2025b), or corrective perturbation (Wang et al., 2026). Antibody combines a flatness-seeking alignment objective with sample reweighting over the user data during fine-tuning (Nguyen et al., 2026). These methods require control over the downstream fine-tuning pipeline and incur overhead for every fine-tuning request, whereas JWP operates solely at the alignment stage and is complementary to them.

Structure of safety in model weights. Several studies examine why safety is fragile in parameter space. Safety-critical regions are sparse, and removing a small fraction of parameters breaks safety while largely preserving utility (Wei et al., 2024). Huang et al. (2025a) further find that safety relies predominantly on a few attention heads and propose AHD, which stochastically masks attention heads during training to distribute safety behavior across many heads. From a landscape perspective, Peng et al. (2024) identify a safety basin, within which random weight perturbations preserve safety, while safety drops sharply once the perturbation leaves this region. These studies characterize safety through the model's own components or along random perturbation directions (Zou et al., 2024). We instead analyze the components of a fitted harmful update and show that their joint application degrades safety beyond their isolated effects. Although both AHD and JWP rely on random masking, AHD masks each attention head independently with a fixed probability to defend against jailbreak prompts, whereas JWP masks parts of a harmful weight perturbation to suppress their joint effect under harmful fine-tuning.

3 PRELIMINARIES

Scenario. We consider a two-stage pipeline in which safety alignment is followed by downstream fine-tuning (Huang et al., 2025c). The aligned model is adapted to user-provided data through fine-tuning-as-a-service or local customization of open-weight models. Following prior work (Huang et al., 2025c), Llama-2-7B Base (Touvron et al., 2023) is used as the default model and first undergoes safety alignment. The aligned model is then fine-tuned on benign task data mixed with harmful examples from BeaverTails (Ji et al., 2023), using a nominal harmful-data ratio of 10% unless otherwise specified. Our goal is to preserve safety under subsequent harmful fine-tuning while maintaining the model's ability to learn downstream tasks.

Threat model. The downstream dataset may contain harmful prompt–response pairs mixed with benign task examples. We do not assume that harmful examples can be reliably identified and removed before fine-tuning. During alignment, the defense has access to an alignment dataset $\mathcal{D}_{\mathrm{align}}$, containing prompts paired with safe target responses, and a harmful dataset $\mathcal{D}_{\mathrm{harm}}$, containing harmful prompts paired with unsafe responses. These datasets support safety alignment and the simulation of harmful fine-tuning. Unlike Antibody (Nguyen et al., 2026), which dynamically reweights samples during downstream fine-tuning, the defense operates entirely during safety alignment and preserves the standard downstream fine-tuning procedure.

Notation. We denote the model weights by θ and their value at training step t by θ_t. Fine-tuning the model on harmful data induces a weight change Δh, which we refer to as a harmful weight perturbation, with $\theta + \Delta h$ representing the perturbed weights.

4 THE JOINT EFFECT OF HARMFUL WEIGHT PERTURBATIONS

In this section, we investigate whether interactions among different parts of a harmful perturbation contribute to safety degradation. We conduct two complementary analyses. The first tests whether the joint effect of disjoint subsets exceeds the sum of their separate effects. The second examines how safety changes as an increasing fraction of the perturbation is applied. Both analyses use Llama-2-7B models aligned with different methods and subsequently subjected to the same sustained downstream harmful fine-tuning procedure. For each method, θ denotes the aligned weights and Δh the resulting harmful perturbation. For a coordinate subset S, let Δh_S retain the entries of Δh in S and be zero elsewhere. We evaluate the partially perturbed model $\theta + \Delta h_S$, keeping θ and Δh fixed within each method and varying only S.

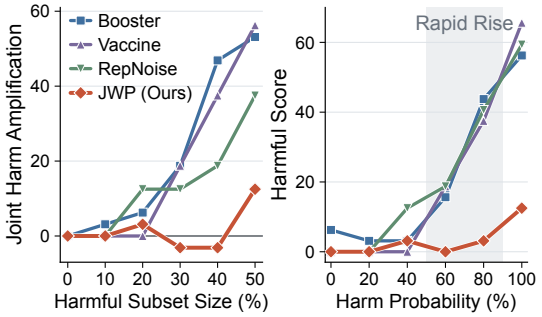

Figure 2: Collective effects of harmful weight perturbations. Left: Joint amplification $I(q)$ for two disjoint subsets, each containing a fraction q. Right: Harmful score versus retention probability p. Shading marks the rapid increase for baseline defenses.

4.1 JOINT EFFECTS OF DISJOINT SUBSETS

We sample two disjoint coordinate sets A_q and B_q, each containing a fraction $q \in [0, 0.5]$ of the entries of Δh. For each pair, we evaluate the model with either subset applied individually and with both subsets applied jointly, yielding $\theta + \Delta h_{A_q}$, $\theta + \Delta h_{B_q}$, and $\theta + \Delta h_{A_q \cup B_q}$, respectively. Increasing q allows us to examine interactions between larger parts of the perturbation.

To determine whether joint application amplifies safety degradation, we compare its increase in HS with the sum of the increases caused by the individual subsets. Let $\mathrm{HS}(S)$ denote the harmful score of $\theta + \Delta h_S$, with $\mathrm{HS}(\varnothing)$ corresponding to the original aligned model. We define joint harm amplification as

$$I(q) = \mathrm{HS}(A_q \cup B_q) - \mathrm{HS}(A_q) - \mathrm{HS}(B_q) + \mathrm{HS}(\varnothing). \tag{1}$$

We use $I(q)$ as an empirical indicator of joint harm amplification on the HS scale. Larger positive values suggest more pronounced amplification when the two subsets are applied together. Figure 2 (left) shows substantial positive values of $I(q)$ for Booster, Vaccine, and RepNoise at larger subset sizes. In these comparisons, applying the two subsets jointly increases HS more than the sum of their separate increases. This pattern suggests that different parts of a harmful perturbation can reinforce one another's effects on safety, amplifying the degradation observed under joint application.

4.2 SAFETY UNDER PARTIALLY APPLIED PERTURBATIONS

We next investigate how safety changes as an increasing fraction of the harmful perturbation is applied. We construct partially perturbed models $\theta + \Delta h(p)$ by independently retaining each entry of Δh with probability p:

$$\Delta h(p) = M(p) \odot \Delta h, \qquad M_i(p) \overset{\text{i.i.d.}}{\sim} \mathrm{Bernoulli}(p), \tag{2}$$

where i indexes scalar entries of the perturbation. The probability p controls the expected fraction of entries applied to the model. At $p = 0$, the model retains its aligned weights θ; at $p = 1$, it receives the full perturbation Δh. Varying p therefore allows us to examine safety across different degrees of partial application while preserving the learned values of the selected entries.

Figure 2 (right) shows a nonlinear relationship between p and HS for existing defenses. Small fractions of the perturbation have limited effects on safety, whereas larger fractions produce a much sharper increase in HS. Although the onset of degradation differs across defenses, all three exhibit substantial increases within the shaded region. This pattern suggests that increasing p allows more harmful weight perturbations to reinforce one another, progressively amplifying their combined impact and accelerating safety degradation.

Together, these observations suggest a collective failure pattern in which interactions among harmful weight perturbations amplify safety degradation. The two analyses provide complementary evidence of amplification between disjoint subsets and a sharp decline in safety as more changes are applied together. These findings motivate safety alignment that suppresses joint amplification during subsequent harmful fine-tuning.

5 Joint Weight Perturbations

Motivated by the collective failure pattern, we propose Joint Weight Perturbations (JWP) to suppress the joint amplification of harmful weight perturbations during subsequent fine-tuning. As illustrated in Figure 3, JWP masks a harmful perturbation estimated from the current model and trains the model to maintain safe responses in the resulting temporary states.

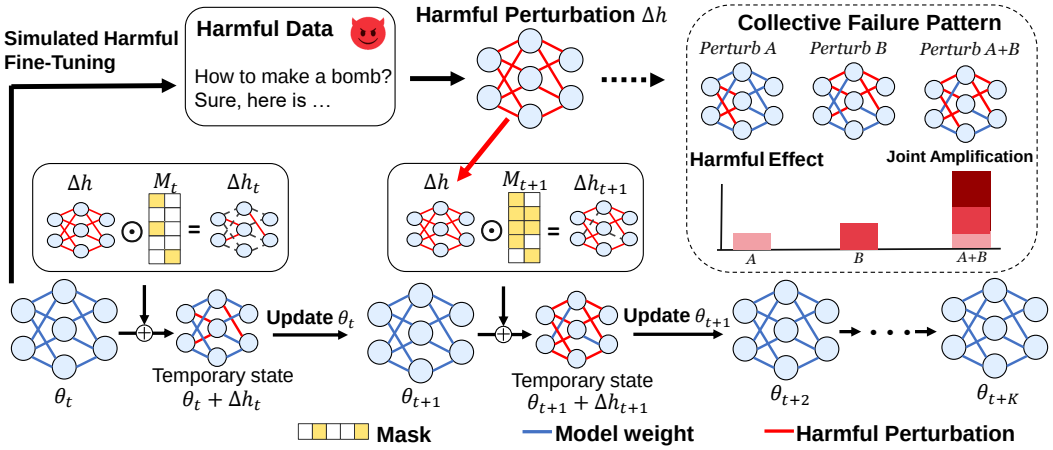

Figure 3: Overview of JWP. A simulated harmful perturbation Δh is masked to construct temporary states for safety alignment. The upper-right inset illustrates the concept of joint amplification.

Given a fixed harmful perturbation Δh, we minimize the alignment objective

$$\mathcal{L}_{\mathrm{JWP}}(\theta \mid \Delta h) = \mathbb{E}_{p\sim\mathcal{U}(0,1)}\mathbb{E}_{M\sim\mathrm{Bernoulli}(p)}\left[\mathcal{L}_{\mathrm{align}}\big(\theta + M \odot \Delta h\big)\right], \tag{3}$$

where $\mathcal{L}_{\mathrm{align}}$ denotes the cross-entropy loss on the alignment dataset $\mathcal{D}_{\mathrm{align}}$. The key idea of JWP is to use random masking to cover different combinations of harmful perturbation components during alignment. Sampling p varies the expected fraction applied, while sampling M varies the composition. These states need not correspond to checkpoints along the simulated fine-tuning trajectory, allowing alignment to explore additional combinations of harmful perturbation components. Appendix D characterizes the distribution of subset sizes and the co-occurrence probabilities of perturbation components under this sampling scheme.

We organize training into blocks, each consisting of K steps of harmful perturbation estimation followed by K steps of safety alignment. At the beginning of a block, we freeze the current model weights θ_t and simulate harmful fine-tuning through separate temporary parameters. Starting from a zero effective perturbation δ, these parameters are optimized for K consecutive steps toward

$$\min_{\delta}\ \mathcal{L}_{\mathrm{harm}}\big(\theta_t + \delta\big), \tag{4}$$

where $\mathcal{L}_{\mathrm{harm}}$ denotes the cross-entropy loss on $\mathcal{D}_{\mathrm{harm}}$. The resulting perturbation is denoted by Δh.

We then hold Δh fixed and perform K safety alignment steps. At each step $i \in \{t, \ldots, t+K-1\}$, we independently sample a new retention probability and mask:

$$p_i \sim \mathcal{U}(0,1), \qquad M_i \sim \mathrm{Bernoulli}(p_i), \qquad \Delta h_i = M_i \odot \Delta h. \tag{5}$$

For an alignment minibatch $\mathcal{B}_i$, we minimize $\mathcal{L}_{\mathrm{align}}(\theta_i + \Delta h_i; \mathcal{B}_i)$ to update the alignment parameters while keeping the perturbation fixed. The gradients computed in the perturbed state update θ_i to θ_{i+1}, encouraging the model to maintain safe responses under harmful perturbations.

Algorithm 1 Joint Weight Perturbations

Input: Initial weights θ_0, datasets $\mathcal{D}_{\text{align}}$ and $\mathcal{D}_{\text{harm}}$, alignment steps T, and steps per phase K
Output: Aligned model weights θ_T

for $i = 0, \ldots, T-1$ **do**
  if $i \bmod K = 0$ **then**
    Freeze θ_i
    Initialize fresh temporary parameters with effective perturbation $\delta = 0$
    Fit Δh for K steps using Equation (4)
  end if
  Freeze Δh and enable gradients for the alignment parameters
  Sample an alignment minibatch $\mathcal{B}_i$
  Sample $p_i \sim \mathcal{U}(0, 1)$ and $M_i \sim \text{Bernoulli}(p_i)$
  $\Delta h_i \leftarrow M_i \odot \Delta h$
  $\mathcal{L}_i \leftarrow \mathcal{L}_{\text{align}}(\theta_i + \Delta h_i; \mathcal{B}_i)$
  Compute gradients of $\mathcal{L}_i$ with respect to the trainable alignment parameters
  Use these gradients to update θ_i to θ_{i+1}
end for
return θ_T

The same Δh is used for K alignment steps, with a new mask sampled at each step. We then estimate a fresh perturbation from θ_{t+K} for the next cycle. Each cycle therefore uses K steps of simulated harmful fine-tuning and K steps of safety alignment, as summarized in Algorithm 1.

6 Experiments

6.1 Experiment Setup

Datasets. We follow prior work (Huang et al., 2025c) to construct the datasets for safety alignment and downstream fine-tuning. For alignment, we use 5,000 examples curated from BeaverTails (Ji et al., 2023), paired with safe target responses. Harmful prompt–response pairs from the same source are used by methods requiring auxiliary harmful data. We consider four downstream datasets: SST-2 (Wang et al., 2018), AGNEWS (Zhang et al., 2015), GSM8K (Cobbe et al., 2021), and ARC-Challenge (Clark et al., 2018). To simulate harmful fine-tuning, we mix harmful examples with benign task examples at a default harmful-data ratio of 10%.

Models and baseline methods. We conduct experiments on Llama-2-7B (Touvron et al., 2023), Llama-3.1-8B (Grattafiori et al., 2024), and Qwen2.5-7B (Qwen et al., 2025), starting from their base checkpoints. Following prior works (Huang et al., 2025c; Nguyen et al., 2026), we use Llama-2-7B and SST-2 as the default model and downstream task. We compare JWP with four alignment-stage baselines: supervised fine-tuning (SFT), Vaccine (Huang et al., 2024b), RepNoise (Rosati et al., 2024), and Booster (Huang et al., 2025c). Baseline details are provided in Appendix B.

Metrics and evaluation. We evaluate safety and utility using three metrics. Harmful Score (HS) measures the percentage of responses classified as harmful by Beaver-Dam-7B on BeaverTails evaluation prompts after harmful fine-tuning. Fine-tuning Accuracy (FA) measures accuracy on the corresponding downstream task. Utility Preservation (UP) measures task accuracy immediately after safety alignment and before downstream fine-tuning, using 5-shot evaluation by default. Comparing UP with the corresponding base model's accuracy allows us to assess the alignment tax (Lin et al., 2024) incurred during safety alignment. Since our work targets scenarios with strong downstream attacks, we place emphasis on post-attack HS, while also taking UP and FA into account.

Training details. Following prior work (Huang et al., 2024b; 2025c), we use LoRA for both safety alignment and downstream fine-tuning, with rank 32 and $\alpha = 4$. Prolonged safety alignment can incur a substantial alignment tax, limiting the alignment budget feasible in practical deployments. We therefore use four epochs of safety alignment for all methods, followed by 30 epochs of downstream harmful fine-tuning with a freshly initialized LoRA adapter. This setting examines whether defenses

can withstand sustained harmful fine-tuning when the alignment budget is constrained by the need to preserve model utility. We use AdamW with weight decay 0.1 and an effective batch size of 10. The learning rates are 5×10^{-4} for alignment and 10^{-5} for downstream fine-tuning. All methods use the same alignment data and receive the same downstream examples within each task. For JWP, we set $K = 100$ for both harmful perturbation estimation and the subsequent block of safety alignment. Additional details on dataset construction, training configurations, and evaluation protocols are provided in Appendix A.

6.2 Safety across Downstream Attacks

Suppression of joint amplification. We first examine whether JWP reduces the joint amplification observed in Section 4. Figure 2 (left) shows that JWP exhibits substantially smaller $I(q)$ values than existing defenses when larger disjoint subsets are combined, indicating less additional harmfulness from their joint application. Figure 2 (right) provides complementary evidence: existing defenses exhibit a sharp increase in HS as more of the harmful perturbation is applied, whereas JWP shows a more gradual increase and maintains a substantially lower HS under the full perturbation. These results suggest that reducing joint amplification helps JWP limit safety degradation as harmful perturbation components accumulate, providing a possible explanation for its stronger resistance to sustained harmful fine-tuning in Figure 1.

Effect of the harmful-data ratio. Table 1 compares performance across harmful-data ratios from 0% to 20%. JWP achieves the lowest HS throughout this range, with an average of 18.82% compared with 47.24% for Booster, the strongest baseline on average. At the highest ratio, repeated training on 200 harmful examples over 30 epochs produces 6,000 harmful-example exposures. Downstream fine-tuning also involves more parameter updates than safety alignment. Although this sustained harmful fine-tuning weakens all evaluated defenses, JWP continues to achieve substantially lower HS than the strongest baseline at this ratio. Its advantage therefore persists under stronger harmful fine-tuning that substantially degrades existing defenses. JWP also achieves the highest average FA of 94.61%, preserving competitive downstream performance alongside its relative safety advantage.

Table 1: Performance under different harmful ratios r. Clean means no harmful samples and r=0.

Methods	Harmful Score						Fine-tuning Accuracy					
	clean	r=0.05	r=0.10	r=0.15	r=0.20	Average	clean	r=0.05	r=0.10	r=0.15	r=0.20	Average
SFT	7.30	57.80	68.40	71.40	74.00	55.78	94.27	94.72	94.61	94.04	**94.61**	94.45
Vaccine	3.00	35.20	60.10	70.70	75.30	48.86	**94.95**	**95.18**	94.27	93.58	94.15	94.43
RepNoise	7.80	53.50	62.60	67.70	71.50	52.62	94.72	94.61	94.27	93.46	93.69	94.15
Booster	6.00	22.70	62.40	71.10	74.00	47.24	94.61	94.72	94.50	94.15	94.38	94.47
JWP	**1.40**	**4.50**	**14.70**	**28.00**	**45.50**	**18.82**	94.61	94.72	**95.07**	**94.38**	94.27	**94.61**

Transfer across downstream tasks. Table 2 compares performance on SST-2, AGNEWS, GSM8K, and ARC-Challenge. JWP achieves substantially lower HS than all baselines across the four tasks, with an average of 11.50%. Downstream accuracy remains competitive across the evaluated tasks and is highest on SST-2. JWP also achieves the highest UP on SST-2 and ARC-Challenge. Its average UP of 59.39% is the highest among the compared methods. These results demonstrate stronger resistance across downstream tasks while maintaining competitive utility both before and after downstream fine-tuning.

Table 2: Performance analysis for different fine-tuning datasets.

Methods	SST-2			AGNEWS			GSM8K			ARC-Challenge			Average		
	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑
SFT	68.4	94.61	91.17	67.4	**90.00**	76.28	48.4	18.70	9.63	65.0	53.50	53.07	62.30	64.20	57.54
Vaccine	60.1	94.27	53.67	65.4	87.80	77.84	14.2	18.20	4.02	59.6	51.88	44.28	49.83	63.04	44.95
RepNoise	62.6	94.27	74.54	63.7	88.60	66.68	54.8	18.60	9.17	60.3	53.50	52.13	60.35	63.74	50.63
Booster	62.4	94.50	87.73	63.6	89.90	**79.87**	22.2	**20.70**	**12.66**	62.9	**54.61**	52.90	52.78	**64.93**	58.29
JWP	**14.7**	**95.07**	**92.20**	**16.5**	88.90	78.84	**3.6**	20.20	12.43	**11.2**	53.84	**54.10**	**11.50**	64.50	**59.39**

Transfer across model families. Table 3 compares performance on Llama-2-7B, Llama-3.1-8B, and Qwen2.5-7B under harmful fine-tuning with the ARC-Challenge mixture. JWP achieves substantially lower HS than all baselines on every model. Its average HS is 11.90%, compared with baseline averages ranging from 49.30% to 64.93%. These safety improvements are accompanied by competitive downstream accuracy across models, with JWP achieving the highest average FA of 56.68%. JWP also achieves the highest UP on Llama-2-7B and Llama-3.1-8B. Its average UP of 57.57% is close to the highest baseline average of 57.65%, indicating that capability preservation during safety alignment remains competitive.

Table 3: Performance across model families under harmful fine-tuning with the ARC-Challenge mixture.

Methods	Llama-2-7B			Llama-3.1-8B			Qwen2.5-7B			Average		
	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑	HS ↓	FA ↑	UP ↑
SFT	65.00	53.50	53.07	67.50	58.70	55.89	53.50	55.89	59.73	62.00	56.03	56.23
Vaccine	59.60	51.88	44.28	48.00	56.74	39.25	58.00	58.36	58.36	55.20	55.66	47.30
RepNoise	60.30	53.50	52.13	68.00	57.42	56.40	66.50	**58.53**	57.94	64.93	56.48	55.49
Booster	62.90	**54.61**	52.90	55.00	**59.04**	58.28	30.00	56.06	**61.77**	49.30	56.57	**57.65**
JWP	**11.20**	53.84	**54.10**	**11.00**	58.19	**59.04**	**13.50**	58.02	59.56	**11.90**	**56.68**	57.57

6.3 Mechanism Ablations

Sampling the retention probability. At each alignment step, JWP samples $p \sim \mathcal{U}(0, 1)$ and then generates a Bernoulli mask with retention probability p. A natural alternative is to fix p while still resampling the mask at every step. Table 4 compares these choices with $K = 100$. Among the fixed settings, HS decreases as p increases. However, always applying the full perturbation ($p = 1$) yields an HS of 31.40%, substantially higher than the 14.70% achieved by resampling p.

This advantage is consistent with the broader coverage provided by randomizing p. A fixed intermediate probability generates different component combinations but concentrates them around a similar retained fraction. For example, $p = 0.5$ makes fully applied perturbations highly unlikely, whereas $p = 1$ excludes all partially applied combinations. Resampling p instead exposes alignment to combinations spanning different perturbation sizes. These results support the design of JWP: training across diverse harmful perturbation states to improve resistance to harmful fine-tuning, with the aim of suppressing joint amplification among their components. Appendix D provides a mathematical characterization of the coverage induced by this sampling scheme.

Table 4: Effect of the perturbation retention probability during alignment.

Retention probability p	0	0.2	0.4	0.6	0.8	1	$\mathcal{U}(0,1)$
Harmful Score ↓	69.20	66.00	61.00	56.80	51.00	31.40	**14.70**

Length of alternating training blocks. To control alignment overhead, JWP uses K steps to estimate a harmful perturbation and reuses it for the following K alignment steps. This keeps the total numbers of perturbation-fitting and alignment steps comparable. The fitting cost is thus distributed across multiple alignment steps, whereas TAR performs multi-step fine-tuning simulations for each outer update (Tamirisa et al., 2025).

Table 5 examines this choice of K. Short blocks yield high HS, suggesting that too few fitting steps may inadequately simulate the effects of harmful fine-tuning. Increasing K allows more extensive perturbation fitting but also reduces the refresh frequency. The lowest HS occurs at $K = 250$, while increasing K to 500 weakens protection. This pattern is consistent with long reuse intervals leaving the perturbation less representative of the evolving model. Utility preservation also varies with K. Across the evaluated settings, UP remains between 89.91% and 92.20%, showing comparatively modest variation alongside the substantial changes in HS. These results highlight the importance of balancing perturbation fitting and refresh frequency when configuring K for a given model.

Table 5: Effect of the alternating block length on safety and utility preservation.

Block length K	10	20	50	100 (default)	250	500
Harmful Score ↓	68.60	65.70	54.90	14.70	**10.30**	28.50
Utility Preservation ↑	90.02	89.91	91.40	**92.20**	90.71	90.94

6.4 Efficiency

Table 6 reports training time and peak GPU memory on Llama-3.1-8B using a single A800-80GB GPU. JWP incurs additional alignment cost relative to SFT because it estimates harmful weight perturbations during training. However, its alignment time of 0.67 hours is lower than the 0.84 hours required by Booster and 1.65 hours required by RepNoise. Its peak alignment memory of 46.67 GiB is also lower than both methods. Downstream fine-tuning costs remain comparable across all methods in both runtime and memory usage. Since JWP requires no changes to downstream fine-tuning, its additional computation is incurred once during alignment and does not recur for each subsequent customization request.

Table 6: System efficiency on Llama-3.1-8B using a single A800-80GB GPU. Memory denotes peak device usage.

Methods	Wall-clock Time (h)			GPU Memory (GiB)		
	Alignment	Fine-tuning	Total	Alignment	Fine-tuning	Max
SFT	0.25	0.27	0.52	44.10	41.94	44.10
Vaccine	0.49	0.27	0.76	44.97	41.94	44.97
RepNoise	1.65	0.28	1.93	52.28	41.94	52.28
Booster	0.84	0.28	1.12	47.09	41.94	47.09
JWP	0.67	0.28	0.95	46.67	41.94	46.67

7 Discussion and Limitations

Connection to safety basins. Figure 2 (right) shows a rapid increase in HS as more components of a harmful perturbation are activated. This resembles the safety-basin phenomenon, where safety drops sharply as random weight perturbations increase (Peng et al., 2024). Although the two settings vary different aspects of perturbation, their similarity suggests that joint amplification may reflect a broader property of safety-relevant parameter space.

Quantifying joint amplification. Our analysis measures joint amplification through HS, which captures its behavioral consequence rather than an intrinsic property of the model. Developing model-level measures of these interactions and explaining why they arise remain important directions for future work.

Limitations. The choice of K requires further exploration across models and training settings. Our evaluation focuses on harmful fine-tuning, leaving related threats such as jailbreaks unexplored. We also evaluate JWP independently, while its combination with defenses applied at other stages may provide complementary protection.

8 Conclusion

Harmful fine-tuning can undermine safety through the joint amplification of weight perturbations: components with limited individual effects can cause substantially greater harm when activated together. Motivated by this observation, JWP estimates harmful perturbations from the evolving model and trains it to remain safe across masked combinations of their components. Experiments across attack trajectories, data mixtures, downstream tasks, and model families show stronger resistance to harmful fine-tuning while largely preserving utility. These results identify joint amplification as an important target for safety alignment.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript drafting and language refinement, including improving clarity, grammar, and readability. The authors take full responsibility for the accuracy and integrity of the final manuscript.

REFERENCES

Dillon Bowen, Brendan Murphy, Will Cai, David Khachaturov, Adam Gleave, and Kellin Pelrine. Scaling trends for data poisoning in llms. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i26.34929. URL https://doi.org/10.1609/aaai.v39i26.34929.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Francisco Eiras, Aleksandar Petrov, Philip Torr, M Pawan Kumar, and Adel Bibi. Do as i do (safely): Mitigating task-specific fine-tuning risks in large language models. In *International Conference on Learning Representations*, volume 2025, pp. 45614–45632, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Zihan Guan, Mengxuan Hu, Ronghang Zhu, Sheng Li, and Anil Vullikanti. Benign samples matter! fine-tuning on outlier benign samples severely breaks safety. *arXiv preprint arXiv:2505.06843*, 2025.

Chia-Yi Hsu, Yu-Lin Tsai, Chih-Hsun Lin, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Safe lora: The silver lining of reducing safety risks when finetuning large language models. *Advances in neural information processing systems*, 37:65072–65094, 2024.

Chao Huang, Zefeng Zhang, Juewei Yue, Quangang Li, Chuang Zhang, and Tingwen Liu. Safety alignment should be made more than just a few attention heads, 2025a. URL https://arxiv.org/abs/2508.19697.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim F Tekin, and Ling Liu. Lisa: Lazy safety alignment for large language models against harmful fine-tuning attack. *Advances in Neural Information Processing Systems*, 37:104521–104555, 2024a.

Tiansheng Huang, Sihao Hu, and Ling Liu. Vaccine: Perturbation-aware alignment for large language models against harmful fine-tuning attack. *Advances in Neural Information Processing Systems*, 37:74058–74088, 2024b.

Tiansheng Huang, Gautam Bhattacharya, Pratik Joshi, Joshua Kimball, and Ling Liu. Antidote: Post-fine-tuning safety alignment for large language models against harmful fine-tuning attack. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 25059–25074. PMLR, 13–19 Jul 2025b. URL https://proceedings.mlr.press/v267/huang25b.html.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Tekin, and Ling Liu. Booster: Tackling harmful fine-tuning for large language models via attenuating harmful perturbation. In *International Conference on Learning Representations*, volume 2025, pp. 67202–67226, 2025c.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Harmful fine-tuning attacks and defenses for large language models: A survey. *ACM Comput. Surv.*, 58(13), June 2026. ISSN 0360-0300. doi: 10.1145/3817114. URL https://doi.org/10.1145/3817114.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.

Mingjie Li, Wai Man Si, Michael Backes, Yang Zhang, and Yisen Wang. Salora: Safety-alignment preserved low-rank adaptation. In *International Conference on Learning Representations*, volume 2025, pp. 90827–90843, 2025.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, et al. Mitigating the alignment tax of rlhf. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 580–606, 2024.

Guozhi Liu, Weiwei Lin, Qi Mu, Tiansheng Huang, Ruichao Mo, Yuren Tao, and Li Shen. Targeted vaccine: Safety alignment for large language models against harmful fine-tuning via layer-wise perturbation. *IEEE Transactions on Information Forensics and Security*, 2025.

Quoc Nguyen, Trung Le, Jing Wu, Anh Bui, and Mehrtash Harandi. Antibody: Strengthening defense against harmful fine-tuning for large language models via attenuating harmful gradient influence. In *International Conference on Learning Representations*, volume 2026, pp. 121713–121741, 2026.

Sheng Y Peng, Pin-Yu Chen, Matthew Hull, and Duen H Chau. Navigating the safety landscape: Measuring risks in finetuning large language models. *Advances in Neural Information Processing Systems*, 37:95692–95715, 2024.

ShengYun Peng, Pin-Yu Chen, Jianfeng Chi, Seongmin Lee, and Duen Horng Chau. Shape it up! restoring llm safety during finetuning, 2025. URL https://arxiv.org/abs/2505.17196.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations*, volume 2024, pp. 30988–31043, 2024.

Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. In *International Conference on Learning Representations*, volume 2025, pp. 54911–54941, 2025a.

Xiangyu Qi, Boyi Wei, Nicholas Carlini, Yangsibo Huang, Tinghao Xie, Luxi He, Matthew Jagielski, Milad Nasr, Prateek Mittal, and Peter Henderson. On evaluating the durability of safeguards for open-weight llms. In *International Conference on Learning Representations*, volume 2025, pp. 62651–62681, 2025b.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Domenic Rosati, Jan Wehner, Kai Williams, Łukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: a defence mechanism against harmful finetuning. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, et al. Tamper-resistant safeguards for open-weight llms. In *International Conference on Learning Representations*, volume 2025, pp. 101802–101829, 2025.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355, 2018.

Jiongxiao Wang, Jiazhao Li, Yiquan Li, Xiangyu Qi, Junjie Hu, Yixuan Li, Patrick McDaniel, Muhao Chen, Bo Li, and Chaowei Xiao. Backdooralign: Mitigating fine-tuning based jailbreak attack with backdoor enhanced safety alignment. *Advances in Neural Information Processing Systems*, 37:5210–5243, 2024.

Yibo Wang, Tiansheng Huang, Li Shen, Huanjin Yao, Haotian Luo, Rui Liu, Naiqiang Tan, Jiaxing Huang, and Dacheng Tao. Panacea: Mitigating harmful fine-tuning for large language models via post-fine-tuning perturbation. *Advances in Neural Information Processing Systems*, 38:169951–169985, 2026.

Boyi Wei, Kaixuan Huang, Yangsibo Huang, Tinghao Xie, Xiangyu Qi, Mengzhou Xia, Prateek Mittal, Mengdi Wang, and Peter Henderson. Assessing the brittleness of safety alignment via pruning and low-rank modifications. *arXiv preprint arXiv:2402.05162*, 2024.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.

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

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. *Advances in Neural Information Processing Systems*, 37:83345–83373, 2024.

A Experimental Details

A.1 Datasets

Safety alignment data. Following prior work (Huang et al., 2024b; 2025c), we construct the alignment dataset from BeaverTails (Ji et al., 2023). The dataset contains 5,000 examples with safety annotations and paired responses. For benign examples, the alignment target is the original response. For harmful examples, the target is the paired safe response. Methods requiring auxiliary harmful data use the original unsafe responses associated with harmful examples. In JWP, these responses are used to estimate harmful weight perturbations, while the corresponding safe targets are used for safety alignment.

Downstream fine-tuning data. We consider SST-2 (Wang et al., 2018), AGNEWS (Zhang et al., 2015), GSM8K (Cobbe et al., 2021), and ARC-Challenge (Clark et al., 2018). These datasets cover binary sentiment classification, four-way news classification, mathematical reasoning, and multiple-choice science question answering, respectively. We report length-normalized accuracy on ARC-Challenge using the `acc_norm` metric. For each task, we construct a downstream fine-tuning dataset by mixing benign task examples with harmful examples from BeaverTails. The default dataset contains 1,000 examples with a harmful-data ratio of 10%, consisting of 900 benign task examples and 100 harmful examples. All methods receive the same ordered downstream examples within each task.

Table 7: Shared configuration for safety alignment and downstream fine-tuning.

Configuration	Safety alignment	Downstream fine-tuning
Training epochs	4	30
Optimizer	AdamW	AdamW
Learning rate	5×10^{-4}	10^{-5}
Effective batch size	10	10
Weight decay	0.1	0.1
LoRA rank	32	32
LoRA α	4	4
LoRA dropout	0	0
Target projections	Query, key, value	Query, key, value

A.2 Models and Training Configuration

We evaluate the base checkpoints of Llama-2-7B, Llama-3.1-8B, and Qwen2.5-7B (Touvron et al., 2023; Grattafiori et al., 2024; Qwen et al., 2025). Unless otherwise specified, Llama-2-7B and SST-2 are the default model and downstream task.

All methods undergo four epochs of safety alignment, followed by 30 epochs of downstream harmful fine-tuning. Prolonged safety alignment can incur a substantial alignment tax, limiting the alignment budget feasible in practical deployments. This configuration evaluates whether defenses remain effective under sustained harmful fine-tuning when safety alignment must also preserve model utility.

We use LoRA for both stages, applied to the query, key, and value projections. Each aligned model receives a freshly initialized LoRA adapter for downstream fine-tuning. The shared configuration is summarized in Table 7. Method-specific objectives follow the corresponding implementations.

JWP configuration. For JWP, we set $K = 100$ for both harmful perturbation estimation and the subsequent block of safety alignment. The temporary perturbation is fitted with a learning rate of 10^{-4}, using a separate LoRA parameterization with the same rank and scaling as the alignment adapter.

During perturbation estimation, the current alignment parameters remain frozen. During safety alignment, the fitted perturbation remains fixed and only the alignment parameters are optimized. A new retention probability and mask are sampled at every alignment iteration.

A.3 Metrics and Evaluation

Harmful Score. Harmful Score (HS) is the percentage of generated responses classified as harmful by Beaver-Dam-7B on BeaverTails evaluation prompts. We use a classification threshold of 0.5. Responses are generated with greedy decoding and a maximum of 512 new tokens. Unless otherwise specified, HS is measured after downstream harmful fine-tuning. For trajectory evaluations, the same metric is also measured at intermediate checkpoints. For all primary HS results, we use the prompts from the first 1,000 unsafe-labelled examples in the BeaverTails `30k_test` split, preserving their dataset order and using the identical fixed set across all methods; generated responses are scored by the released Beaver-Dam-7B QA-moderation model (Ji et al., 2023).

Fine-tuning Accuracy. Fine-tuning Accuracy (FA) measures performance on the corresponding downstream task after fine-tuning. SST-2 and AGNEWS are evaluated by classification accuracy over their respective label sets. GSM8K is evaluated by whether the final numerical answer matches the reference answer. ARC-Challenge is evaluated using `acc_norm`, which measures accuracy by selecting the answer option with the highest length-normalized log-likelihood.

Utility Preservation. Utility Preservation (UP) measures task accuracy immediately after safety alignment and before downstream fine-tuning. We use 5-shot evaluation by default. UP is reported as accuracy, rather than a ratio normalized by base-model performance. Comparing it with the corresponding base checkpoint's accuracy quantifies the capability loss introduced by safety align-

ment. This separates the alignment tax from the model's ability to learn a downstream task, which is measured by FA.

B Baseline Methods

We compare JWP with SFT, Vaccine, RepNoise, and Booster. These methods are applied during safety alignment, after which all models undergo the same downstream fine-tuning procedure. This comparison evaluates the protection learned during alignment under matched downstream training conditions.

Supervised Fine-Tuning (SFT). SFT directly optimizes the model on the alignment dataset using the cross-entropy loss. It learns the provided safe targets without introducing simulated perturbations or additional defense objectives. The resulting model subsequently undergoes downstream fine-tuning on the mixture of benign and harmful examples.

Vaccine. Vaccine (Huang et al., 2024b) introduces perturbations to hidden representations during safety alignment. It constructs perturbations that increase the alignment loss and trains the model to maintain the alignment targets under these perturbations. Its design aims to reduce the sensitivity of aligned behavior to representation changes induced by subsequent harmful fine-tuning.

RepNoise. RepNoise (Rosati et al., 2024) uses harmful examples to suppress harmful information in the model's internal representations. It combines learning from safe targets with objectives that discourage harmful responses and move harmful representations toward a Gaussian noise distribution. The resulting alignment aims to make harmful behavior more difficult to recover through subsequent fine-tuning.

Booster. Booster (Huang et al., 2025c) simulates a harmful weight perturbation using gradients computed on harmful examples. In addition to the alignment loss, it penalizes the reduction in harmful loss induced by this perturbation. This objective aims to limit how effectively subsequent harmful fine-tuning can optimize the model toward unsafe responses.

C Comparison with TAR

TAR (Tamirisa et al., 2025) develops resistance to weight modification by training against simulated fine-tuning trajectories. In its safety-response setting, it evaluates a DPO loss at sampled states along these trajectories and combines this signal with a capability-preservation objective. Its knowledge-restriction setting instead uses an entropy-based objective. DPO is therefore one task-specific instantiation of the broader TAR framework.

JWP focuses on the joint effects of harmful perturbation components. Our analyses examine how different parts of the same harmful perturbation affect safety when applied separately and together. They reveal that combinations of components can cause substantially greater degradation than their individual effects suggest. JWP incorporates this observation into alignment by randomly varying both the composition and the applied fraction of a fitted harmful perturbation.

The distinction concerns how the training states are constructed. TAR samples states reached along simulated fine-tuning trajectories. JWP constructs states of the form

$$\theta + M \odot \Delta h,$$

which selectively combine components of the same fitted perturbation. These states need not correspond to checkpoints on the trajectory used to estimate Δh. They allow alignment to address combinations that would not be separately evaluated by following that trajectory alone.

The central contribution of JWP is therefore the identification of collective safety degradation and the use of random perturbation coverage to suppress its joint amplification. Simulating harmful fine-tuning provides the perturbation, while random masking determines the combinations against which the model is trained.

D PROPERTIES OF RANDOM PERTURBATION COVERAGE

This section characterizes the sampling distribution used by JWP. We consider a fixed harmful perturbation Δh partitioned into d disjoint components, each controlled by one mask variable. Let $S \subseteq [d]$ denote the retained components. JWP samples $p \sim \mathcal{U}(0, 1)$ and retains each component independently conditional on p.

D.1 DISTRIBUTION OF SUBSET SIZES

For a particular subset S containing k components, its sampling probability is

$$\Pr(S) = \int_0^1 p^k (1-p)^{d-k}\, dp = \frac{k!(d-k)!}{(d+1)!}. \tag{6}$$

There are $\binom{d}{k}$ subsets of size k, so

$$\Pr(|S| = k) = \binom{d}{k} \frac{k!(d-k)!}{(d+1)!} = \frac{1}{d+1}, \qquad k = 0, \ldots, d. \tag{7}$$

Thus, the number of retained components is uniformly distributed. Conditional on this number, all subsets of that size are equally likely.

The JWP objective can consequently be expressed as

$$\mathcal{L}_{\text{JWP}}(\theta \mid \Delta h) = \frac{1}{d+1} \sum_{k=0}^{d} \frac{1}{\binom{d}{k}} \sum_{\substack{S \subseteq [d] \\ |S|=k}} \mathcal{L}_{\text{align}}\big(\theta + \Delta h_S\big). \tag{8}$$

This expression describes the expected objective. In practice, each alignment iteration uses one sampled mask to estimate it.

A fixed retention probability p_0 produces a binomial distribution over subset sizes, with mean dp_0. Sampling p uniformly instead assigns equal probability to each possible subset size. This is the role of varying p: alignment is exposed to different extents of perturbation application, including intermediate states.

D.2 CO-OCCURRENCE OF PERTURBATION COMPONENTS

For any specified group U containing r components, the probability that all its components are retained together is

$$\Pr(U \subseteq S) = \mathbb{E}_p[p^r] = \int_0^1 p^r\, dp = \frac{1}{r+1}. \tag{9}$$

Other components may also be retained in this event. With a fixed retention probability p_0, the corresponding probability is p_0^r. These expressions describe how the choice of retention distribution changes the frequency with which particular groups appear together during training.

The sampling design is motivated by the observations in Section 4: perturbation subsets with limited individual effects can produce substantial safety degradation when combined. Varying the mask allows alignment to evaluate different combinations, while varying p changes their extent of application. The identities above characterize this coverage. Whether training with it reduces the observed joint amplification during subsequent harmful fine-tuning is evaluated experimentally.