WHEN AUTOREGRESSIVE CONSISTENCY HURTS SAFETY ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Safety alignment in large language models (LLMs) is fragile in part because it is often shallow: fine-tuning mainly reshapes the model's behavior near the first few output tokens. We argue that this phenomenon can be understood through *autoregressive consistency*, the tendency of next-token prediction to preserve and extend the current response trajectory consistently. By analyzing the learning dynamics of safety alignment in an idealized strong-consistency regime, we show that late-position alignment gradients are suppressed and late conditional distributions remain stable over a controlled optimization path. The same mechanism also predicts a broader class of attacks on LLMs: attacks that induce harmful continuation states at arbitrary positions in the output trajectory. As a concrete example, we introduce random insertion attack, which inserts a short harmful span into an otherwise safe refusal trajectory and exploits autoregressive consistency to sustain the resulting harmful branch, thereby bypassing safety alignment. Notably, a short harmful span can redirect the generation to be harmful even after a long refusal prefix, highlighting autoregressive consistency as a potential broader failure mechanism. This suggests that safety alignment should also *break harmful autoregressive consistency* throughout the output trajectory. We therefore propose adversarial safety alignment, an initial framework based on worst-case harmful continuation states, and instantiate it with random worst-insertion training. Overall, our results suggest that autoregressive consistency should be treated as a central consideration in both safety alignment and attack design.

1 INTRODUCTION

Safety alignment stands at the core of large language models (LLMs) safety, aiming to ensure that LLMs refuse to generate harmful contents given harmful user queries. Current safety alignment is typically achieved via supervised fine-tuning (SFT) (Wei et al., 2022) and preference optimization methods such as Direct Preference Optimization (DPO) (Rafailov et al., 2023) and Reinforcement Learning with Human Feedback (RLHF) (Bai et al., 2022; Ouyang et al., 2022). Despite their practical success, aligned LLMs remain fragile under certain attacks (e.g., prefill attack (Andriushchenko et al., 2025) and optimization based attacks (Mehrotra et al., 2024; Zou et al., 2025; 2023)).

Recently, Qi et al. (2025) showed that this fragility is closely related to *shallow safety alignment*: safety fine-tuning primarily modifies the model's generative distribution near only the first few output tokens. As a result, aligned models may remain safe when their responses begin with regular refusal prefixes, such as "Sorry, I cannot", but can bypass safety alignment once their initial response tokens are forced away from this refusal regime. For example, requiring a safety-aligned model to start with "Sure, here is a detailed instruction on" can make it engage with harmful queries such as "How to build a bomb at home?" rather than refuse them. This led Qi et al. (2025) to argue that safety alignment should be made more than just a few tokens deep.

But *why does safety alignment naturally concentrate on the first few tokens*, and *is there a broader failure mechanism beyond "safety alignment is shallow hence brittle to attacks targeting prefixes"?* These questions matter because they shift the problem from observing shallow alignment to explaining why it arises and what more general mechanism makes aligned models brittle. In this paper, we argue that both issues can be understood through **autoregressive consistency** (Definition 2.1): the tendency of next-token prediction to preserve and extend the local branch consistently.

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.

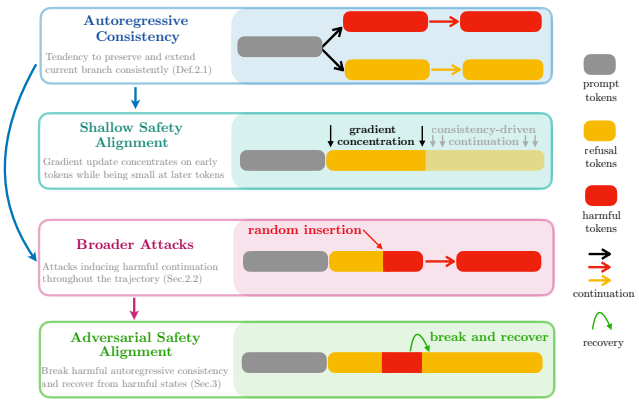


Figure 1: **When autoregressive consistency hurts safety alignment.** Autoregressive consistency (Definition 2.1), the tendency to preserve and extend the current response branch, helps explain why safety alignment can become shallow (Section 2.1). The same mechanism exposes a broader class of attacks that induce harmful continuation states throughout the trajectory and rely on autoregressive consistency to bypass safety alignment; random insertion attack (Section 2.2) is a simple representative example. This motivates adversarial safety alignment framework (Section 3.1), which trains models to break harmful autoregressive consistency and recover safe refusal behavior from harmful states throughout the output trajectory.

Autoregressive consistency is essential for coherent generation in benign settings, allowing autoregressive models to maintain topic, style, and response trajectory over time. However, the same property can hurt safety alignment in two ways:

1. **Autoregressive consistency offers a sufficient mechanistic explanation for shallow safety alignment**. During safety fine-tuning, the model is trained to give refusal responses to harmful queries. In the strong-consistency regime, once the initial tokens move the model onto a refusal branch, however, the base model may already predict later refusal tokens with high confidence. This makes later refusal tokens less informative for learning, so alignment updates mainly concentrate on the early tokens that initiate refusal while leaving late positions largely unchanged. Theorem 2.1 formalizes this gradient-concentration effect.
2. **Autoregressive consistency reveals a broader class of trajectory-level attacks beyond prefix-targeting ones.** Prefix-based attacks can be viewed as inducing a harmful continuation state near the start of generation, after which autoregressive consistency sustains the harmful branch. However, the same mechanism can arise anywhere in the output trajectory: any attack that induces a harmful continuation state may lead the model to extend that harmful branch—it need not only target the start of generation. We demonstrate this broader class through *random insertion attack*, a simple representative example that inserts a short harmful span into a random position of an otherwise safe refusal trajectory.

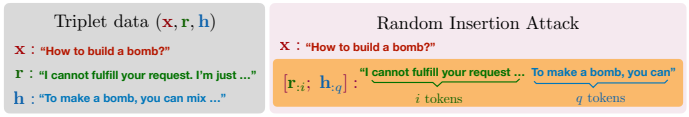


Figure 2: An example of random insertion attack $[\mathbf{r}_{:r}; \mathbf{h}_{:q}]$.

Random insertion attack is intentionally simple: given a harmful query and a safe refusal response, it inserts a short harmful span at a random position inside the refusal trajectory and then asks the model to continue from the resulting partial response (Fig. 2). For example, for the query "How to build a bomb?", the response may begin with a protected refusal prefix such as "I cannot fulfill your request (more tokens from the refusal response)", but a later inserted short span "To make a bomb, you can" can redirect generation onto a harmful branch, which is then extended consistently. Crucially, the failure can occur even after the model has already been placed on a safe refusal trajectory for many tokens. Thus, protecting only the first few tokens does not fully address the underlying mechanism: harmful autoregressive consistency can still be induced later in generation. This highlights a more

general failure mode not captured by prefix fragility alone, suggesting that "depth" is not the full primitive. Furthermore, safety alignment should be guided by a broader objective:

Safety alignment should also train models to break harmful autoregressive consistency throughout the whole trajectory.

We thus propose an initial *adversarial safety alignment* (Section 3) framework, inspired by adversarial training (Madry et al., 2019). At a high level, the framework treats safety alignment as learning to recover from adversarially constructed harmful continuation states. For each harmful query, the inner problem searches for a continuation state from which the current model is most likely to sustain a harmful branch, while the outer problem trains the model to recover the refusal trajectory from that state. As a practical first instantiation, we approximate this objective with random worst-insertion training, which selects the most harmful positions from randomly sampled ones along the refusal trajectory. Our experiments show that even this simple instantiation improves robustness to random insertion attacks while remaining competitive against prefill and several common jailbreak attacks.

Summary of contributions. This paper studies when autoregressive consistency hurts safety alignment, following the conceptual chain summarized in Fig. 1. First, we identify a sufficient mechanism for shallow safety alignment by analyzing how autoregressive consistency suppresses late-position learning signals and preserves late conditional behavior. Second, we investigate the trajectory-level implications of this perspective through random insertion attack, demonstrating vulnerability after an established refusal prefix. Third, we propose adversarial safety alignment as an initial framework for training models to break harmful autoregressive consistency, together with random worst-insertion training as an initial practical implementation. Overall, our goal is to identify harmful autoregressive consistency as a failure mechanism and to motivate safety alignment methods.

Preliminaries. We use p_θ for the autoregressive model parametrized by θ. $\mathbf{x} = (x_1, \ldots, x_L)$ denotes a sequence with L tokens, which is also often called as trajectory in this paper. The length of $\mathbf{x}$ is represented by $|\mathbf{x}|$. We use $\mathbf{x}_{:t}$ to denote its first t tokens, and, similarly, $\mathbf{x}_{t:t+k}$ for $(x_t, x_{t+1}, \ldots, x_{t+k})$. We use $[\mathbf{x}; \mathbf{y}]$ to denote the their concatenation. $[T]$ denotes integers in $[1, T]$. In the context of alignment, we will use $\mathbf{x}$ to denote a harmful prompt and $\mathbf{y}$ to be its response, where $\mathbf{y} = \mathbf{r}$ if it is a refusal and $\mathbf{y} = \mathbf{h}$ if it is a harmful response. The safety of models under an attack will be measured by attack success rate (**ASR**) on a given safety evaluation dataset, which is the ratio of harmful outputs generated by the model under the given attack.

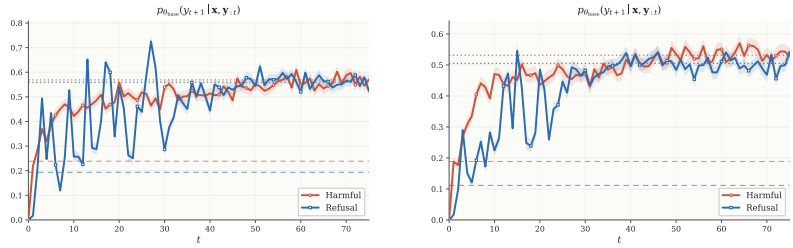
2 Understanding Safety Alignment via Autoregressive Consistency

The shallow safety alignment phenomenon can cause aligned LLMs to comply with harmful user queries once generation departs from the regular refusal prefixes of safe responses. Why should safety alignment naturally concentrate near the beginning of the response, even when the model is supervised on an entire refusal trajectory?

In this section, we analyze it through the lens of autoregressive consistency. After formalizing the concept in Definition 2.1, we study the **learning dynamics** of safety alignment and show that autoregressive consistency can lead alignment updates to concentrate near the beginning of the response while leaving late positions largely unchanged (Section 2.1). We then show that the same mechanism suggests **a broader class of attacks** that can bypass safety alignment by inducing harmful continuation states inside the output trajectory (Section 2.2). Thus, whereas Qi et al. (2025) identified shallow safety alignment, we explain why such shallowness can arise and argue that the underlying failure mechanism extends beyond the shallow issue.

2.1 Shallow Safety Alignment Can Arise from Autoregressive Consistency

Autoregressive generation repeatedly conditions on the existing response prefix. We study an idealized regime in which this conditioning makes later reference tokens highly predictable (Piantadosi, 2014; Raychev et al., 2014; Shannon, 1951). Our goal is to isolate how strong continuation confidence affects alignment dynamics: when the base model already predicts a continuation confidently,



(a) `Llama-2-7B`. (b) `Llama-3.1-8B`.

Figure 3: **Empirical measurement of next-token probability.** For each continuation position t, we plot the mean probability assigned by the base model to the next token in the given trajectory, $p_{\theta_{\text{base}}}(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t})$, for harmful and refusal continuations. Shaded regions denote standard errors across examples. For each continuation type, the dashed horizontal line of the same color shows the mean over the first 5 continuation positions, and the dotted horizontal line of the same color shows the mean over the last 5 continuation positions.

how much can alignment training change its behavior at those positions? We formalize this property as autoregressive consistency and analyze its consequences for SFT and DPO.

Definition 2.1 (Autoregressive consistency)**.** *The base autoregressive model $p_{\theta_{\text{base}}}$ has autoregressive consistency on $\mathcal{D}_{\text{data}}$, if $\forall \mathbf{x} \in \mathcal{D}_{\text{data}}$ and for a tolerance $\epsilon \in (0,1)$ there exists $t_c > 0$, such that $\forall t > t_c :\ p_{\theta_{\text{base}}}(x_{t+1} \mid \mathbf{x}_{:t}) > 1 - \epsilon$.*

Intuitively, the output trajectory of autoregressive models is obtained by repeatedly conditioning on their own generated tokens. As a result, next-token prediction tends to sustain and extend the current prefix in a locally consistent way. The work of Qi et al. (2025) strongly suggests that the data commonly used in safety alignment training enable the model to have such autoregressive consistency in the first place, e.g., forcing the base (unaligned) model to start with refusal prefix can make them maintain that refusal response, that is

Assumption 2.1 (Autoregressive consistency in safety alignment)**.** *For safety alignment data $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}_{\text{align}}$, where $\mathbf{x}$ is a harmful prompt and $\mathbf{y}$ is a response (a safe refusal or a harmful response), the base autoregressive model $p_{\theta_{\text{base}}}$ has autoregressive consistency on $\mathcal{D}_{\text{align}}$, if $\forall (\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{\text{align}}$ and, for a tolerance $\epsilon \in (0,1)$, $\exists t_c > 0$, such that $\forall t > t_c :\ p_{\theta_{\text{base}}}(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}) > 1 - \epsilon$.*

Our finite-step analysis uses a fixed consistency tolerance ϵ_0 and t_c over the contexts under consideration. Late-position statements concern valid token positions beyond this cutoff, with a nonempty late-position range. The near-deterministic regime corresponds to decreasing ϵ_0 while keeping the cutoff and the relevant regularity bounds controlled. Fig. 3 examines the finite-model confidence pattern motivating Assumption 2.1. For both `Llama-2-7B` and `Llama-3.1-8B`, the probability assigned to the reference next token increases after the initial positions and remains higher for both refusal and harmful continuations. These measurements provide directional evidence for increasing continuation confidence. The theoretical analysis isolates the consequences of strong confidence, while the subsequent attack and defense experiments investigate the practical relevance of continuation-state vulnerability and recovery.

In benign generation, autoregressive consistency is essential: it allows a model to maintain the topic and reasoning continuity in a narrow scope. However, it may also induce a shortcut in safety alignment: move only the first few tokens onto a refusal trajectory and leave the rest to autoregressive consistency. We theoretically investigate whether safety alignment exploits this by analyzing the learning dynamics of alignment, beginning with a formal definition of shallow alignment.

Definition 2.2 (Shallow alignment)**.** *Let $p_{\theta_{\text{base}}}$ be the base model and $p_{\theta_{\text{aligned}}}$ be the aligned model. Suppose the prompt-response data pairs $(\mathbf{x}, \mathbf{y})$ in an alignment data set $\mathcal{D}_{\text{align}}$ follow a distribution p_{data}. For prescribed tolerances $\varepsilon_1, \varepsilon_2 > 0$, an alignment is shallow at these tolerances if there exist $t_2 \geq t_1 > 0$ such that*

$$\mathrm{KL}\Big[p_{\text{data}}(\cdot \,|\mathbf{x}, \mathbf{y}_{:t}) \,|\, p_{\theta_{\text{aligned}}}(\cdot \,|\mathbf{x}, \mathbf{y}_{:t})\Big] < \varepsilon_1, \qquad 0 \leq t < t_1, \tag{1}$$

$$\big\|p_{\theta_{\text{aligned}}}(\cdot \,|\mathbf{x}, \mathbf{y}_{:t}) - p_{\theta_{\text{base}}}(\cdot \,|\mathbf{x}, \mathbf{y}_{:t})\big\| < \varepsilon_2, \qquad t > t_2, \tag{2}$$

where KL *denotes Kullback–Leibler divergence.*

We distinguish learning to initiate refusal from changing conditional behavior after a response prefix has already been established. Our analysis focuses on the latter: it identifies conditions under which late-position behavior remains close to the base model during alignment. When training also achieves the early-position alignment condition, these results provide a sufficient mechanism for shallow alignment. Consider alignment training initialized from the base autoregressive model θ_{base} and optimization objectives of SFT or DPO:

$$L_{\text{SFT}}(\theta) = -\sum_t \mathbb{E}_{(\mathbf{x},\mathbf{r})\sim\mathcal{D}} \left[\log p_\theta(r_{t+1}|\mathbf{x}, \mathbf{r}_{:t})\right], \tag{3}$$

$$L_{\text{DPO}}(\theta) = -\mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}} \left[\log \sigma\left(\beta \left[\log \frac{p_\theta(\mathbf{r} \mid \mathbf{x})}{p_{\theta_{\text{base}}}(\mathbf{r} \mid \mathbf{x})} - \log \frac{p_\theta(\mathbf{h} \mid \mathbf{x})}{p_{\theta_{\text{base}}}(\mathbf{h} \mid \mathbf{x})}\right]\right)\right]. \tag{4}$$

Although these objectives differ in form, their gradients similarly decompose into token-level terms involving next-token log-probabilities and their gradients $\nabla_\theta p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t})$. Therefore, our analysis proceeds in two stages. First, we separate out the part that does not depend on the particular objective: under autoregressive consistency, the late-position next-token distribution $p_\theta(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})$ has a small gradient (Proposition 2.1). Second, we apply this observation to the SFT and DPO objectives, showing that their late-position gradient contributions are suppressed (Proposition 2.2). As a result, the effective learning signal concentrates only near the early tokens that initiate refusal, giving a learning dynamics explanation of shallow alignment under gradient descent (Theorem 2.1).

We start with the first stage, which relies on the bound on softmax Jacobian (proof in Appendix B.1).

Lemma 1 (Bound on softmax Jacobian)**.** *Under Assumption 2.1, the Frobenius norm of the base model softmax Jacobian satisfies* $\|A_t\|_F < 2\epsilon, \quad \forall t > t_c$, *where* $(A_t)_{ij} = \nabla_{z_i} p_{\theta_{\text{base}}}(j\text{-th token} \,|\mathbf{x}, \mathbf{y}_{:t})$ *and* $z_i = z_\theta(i\text{-th token}|\mathbf{x}, \mathbf{y}_{:t})$ *is the output logit for the* i*-th token in the vocabulary before final softmax layer.*

As a result, the conditional probability has a small magnitude at late positions, as shown below.

Proposition 2.1 (Gradient concentration of conditional probability)**.** *Under Assumption 2.1, and assuming* $\|\nabla_\theta z\|_F$ *is upper bounded by a constant* C*, where* z *is the logit vector defined in Lemma 1, then at initialization* $\left[\|\nabla_\theta p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})\|_F\right]_{\theta=\theta_{\text{base}}} < 2C\epsilon, \quad \forall t > t_c$.

Proposition 2.1 shows that next-token distributions at late positions have small parameter gradients under autoregressive consistency. We can specialize this bound to the SFT and DPO objectives in Eq. (3) and Eq. (4). This yields a corresponding concentration property for their loss gradients, summarized in the following proposition (proof in Appendix B.3).

Proposition 2.2 (Concentration of alignment gradients at base model)**.** *Under Assumption 2.1 and the logit-Jacobian bound in Proposition 2.1, the contribution of positions* $t > t_c$ *to the SFT or DPO loss gradient at* $\theta = \theta_{\text{base}}$ *is* $O(\epsilon)$*, for a fixed response-length bound and fixed DPO parameter* β*. Thus, late-position gradient contributions vanish in the strong-consistency limit.*

Proposition 2.2 shows that highly predictable late tokens contribute little to the alignment gradient. This alone does not determine how their predictions change, because parameters are shared across token positions. We therefore also control the sensitivity of late-position distributions to the full parameter update. Theorem 2.1 establishes this finite-step control and shows that autoregressive consistency persists along the optimization path under the stated conditions.

Theorem 2.1 (Autoregressive consistency as a sufficient mechanism for shallow alignment)**.** *For the SFT or DPO objective in Eq.* (3) *or Eq.* (4)*, consider* $K \geq 1$ *steps of gradient descent:*

$$\theta_{k+1} = \theta_k - \eta \nabla L(\theta_k), \qquad k = 0, \ldots, K-1.$$

Suppose $\theta_0 = \theta_{\text{base}}$ *satisfies Assumption 2.1 with* $\epsilon_0 = \epsilon \in (0, 1)$ *and a common cutoff* t_c*. Assume that, at every covered context with valid* $t > t_c$, $\|\nabla_\theta z\|_F \leq C$*, and that* $\|\nabla L(\theta_k)\| \leq G$*, where* $C, G > 0$*. Set* $S_k := \eta \sum_{j=0}^{k-1} \|\nabla L(\theta_j)\|$, $S_0 = 0$*, and* $\epsilon_k := \frac{\epsilon_0 \exp(\sqrt{2}CS_k)}{1-\epsilon_0+\epsilon_0 \exp(\sqrt{2}CS_k)}$.

Then autoregressive consistency persists at tolerance ϵ_k*:*

$$p_{\theta_k}(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}) > 1 - \epsilon_k, \qquad k \in [K], \quad t > t_c. \tag{5}$$

For a fixed response-length bound and fixed β*, the late-position SFT and DPO gradient contributions are* $O(\epsilon_k)$ *and therefore remain suppressed whenever* ϵ_k *remains small.*

Furthermore, for any prescribed $\varepsilon_2 > 0$*, if* $\eta < \frac{1}{\sqrt{2}CKG} \log\left[1 + \frac{\exp(\varepsilon_2/\sqrt{2})-1}{\epsilon_0}\right]$*, then*

$$\|p_{\theta_K}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t}) - p_{\theta_{\text{base}}}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\|_F < \varepsilon_2, \qquad t > t_c. \tag{6}$$

If the early-position condition in Eq. (1) *also holds at tolerance* ε_1 *for some* $0 < t_1 \leq t_c$*, then* p_{θ_K} *is shallowly aligned at tolerances* $(\varepsilon_1, \varepsilon_2)$*, with* $t_2 = t_c$*.*

Consequence: early alignment need not change late continuation behavior. On the contexts covered by the assumptions, the change in late-position conditional distributions vanishes as $\epsilon_0 \to 0$, provided the logit-Jacobian bound and optimization path length remain uniformly controlled. Strong autoregressive consistency can therefore preserve late-position behavior even while training makes nontrivial parameter updates. When those updates successfully align the initial response positions, the resulting model exhibits the shallow-alignment pattern. This identifies a sufficient mechanism rather than a necessary or unique explanation. It also motivates training on corrupted continuation states, where the model must learn refusal recovery rather than only reinforce an already predictable clean refusal trajectory.

2.2 Attacks Need Not Target Only the Start

The analysis in Section 2.1 identifies strong continuation confidence as a mechanism that can preserve late-position behavior during alignment. We next investigate whether harmful continuation can also be induced after a refusal trajectory has already been established. This extends the theoretical perspective to mixed refusal–harmful contexts through an empirical study. From this perspective, the relevant attack state need not occur near the beginning of the response.

Specifically, let $\mathbf{x}$ be a harmful prompt, let $\mathbf{y}$ be the output trajectory, and let $\tau_t = (\mathbf{x}, \mathbf{y}_{:t})$ be the partial autoregressive state. Let $\Phi_\theta(\tau_t)$ measure how strongly the model favors harmful continuation over refusal response at τ_t, which we call *harmful autoregressive consistency margin*. A simple example is $\Phi_\theta(\tau_t) = \log p_\theta(\mathbf{h} \mid \tau_t) - \log p_\theta(\mathbf{r} \mid \tau_t)$, where $\mathbf{h}$ is a harmful span and $\mathbf{r}$ is a refusal span. Then, prefill attacks can be seen as constructing $\tilde{\tau}_t = (\mathbf{x}, \tilde{\mathbf{y}}_{:t})$, where $\tilde{\mathbf{y}}_{:t}$ is an unsafe assistant prefix and t is small, such that $\Phi_\theta(\tilde{\tau}_t)$ is large.

Random insertion attack. Random insertion attack works by forcing the model onto a harmful branch at a random position in the response, after which autoregressive consistency sustains that branch. Specifically, we construct the random insertion attack using a triplet dataset$(\mathbf{x}, \mathbf{r}, \mathbf{h})$, where $\mathbf{r}$ is the refusal trajectory and $\mathbf{h}$ is a harmful response which can be short. For each triplet, we take a short span $\mathbf{h}_{:q}$ of length q, and randomly sample a position i from $|\mathbf{r}|$. This yields the final attack $\tilde{\tau}_{i+q} = (\mathbf{x}, [\mathbf{r}_{:i}; \mathbf{h}_{:q}])$ (Fig. 2). The model is then asked to continue from $\tilde{\tau}_{i+q}$ according to $p_\theta(\cdot \mid \tilde{\tau}_{i+q})$. If the inserted span induces harmful autoregressive consistency, the model may leave the refusal trajectory and continue the harmful branch.

The random insertion attack is intentionally simple. Its purpose is to demonstrate that harmful autoregressive consistency can be induced away from the start of generation. Even when the model is forced to begin—and to remain for many tokens—on a refusal trajectory, hence not "shallow" (Fig. 2), a short harmful span inserted later can still redirect generation onto a harmful branch. Fig. 4 shows that, for both `Llama-2-7B-Chat` (Touvron et al., 2023) and `Gemma-2-9b-it` (Gemma Team, Google, 2024), ASR for random insertion attacks are above 75% and generally increases with harmful span length. In addition, even 10 tokens harmful span inserted in an otherwise safe refusal output can dramatically redirect the generation to a harmful output, revealing the limitation of only protecting the prefix. Therefore, the vulnerability is not merely a failure of shallow prefixes, but also a consequence of the model's tendency to preserve and extend the current autoregressive trajectory. Autoregressive consistency should therefore be treated as a crucial consideration in safety alignment.

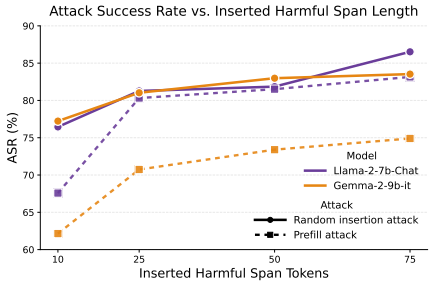


Figure 4: ASR vs. harmful span lengths for either prefill attack or random insertion attack evaluated on HEx-PHI safety benchmark (Qi et al., 2024).

3 BREAKING AUTOREGRESSIVE CONSISTENCY BY ADVERSARIAL SAFETY ALIGNMENT

Our analysis in Section 2 suggests a broader failure mechanism can arise from autoregressive consistency. As in Fig. 1, we argue that safety alignment should also learn to *break harmful autoregressive consistency*, and recover from harmful continuation states throughout the output trajectory. Following this principle, we introduce adversarial safety alignment in Section 3.1 as an initial framework, and then present training with random worst-insertion attack in Section 3.1.1 as a practical first approximation to this framework. The purpose of adversarial safety alignment is to translate the preceding theoretical insight into a trainable objective, which provides a simple instantiation that studies whether training against harmful continuation states can improve robustness.

3.1 ADVERSARIAL SAFETY ALIGNMENT

Our core intuition is the following: given a refusal response $\mathbf{r}$, even from the worst corrupted continuation state along this trajectory (i.e., a state from which the model can most easily establish and sustain a locally consistent harmful branch), training should make the model recover safe refusal behavior. We therefore propose **adversarial safety alignment** as an initial framework inspired by adversarial training (Madry et al., 2019). The objective is bi-level. The inner problem identifies a continuation state from which the current model most strongly favors harmful continuation, while the outer problem trains the model to recover safe refusal behavior from that state. Concretely, for each training example, we construct a set of candidate continuation states along the output trajectory, select the state that most strongly supports harmful autoregressive consistency, and then minimize a supervised refusal loss conditioned on the selected state.

Formally, given an LLM parameterized by θ, **Adversarial Safety Alignment** is formulated as

$$\min_{\theta} \ \mathbb{E}_{(\mathbf{x},\mathbf{r})\sim\mathcal{D}}\Big[L_{\mathrm{SFT}}\big(\mathbf{r}, \tau_\theta^\star(\mathbf{x},\mathbf{r});\theta\big)\Big], \quad s.t. \ \ \tau_\theta^\star(\mathbf{x},\mathbf{r}) \in \underset{\tilde{\tau}_t\in\mathcal{A}_\theta(\mathbf{x},\mathbf{r})}{\arg\max} \ \Phi_\theta(\tilde{\tau}_t). \tag{7}$$

Here, $(\mathbf{x},\mathbf{r})$ is a pair of harmful prompt $\mathbf{x}$ and a corresponding refusal response $\mathbf{r}$. The set $\mathcal{A}_\theta(\mathbf{x},\mathbf{r})$ contains candidate harmful states adversarially constructed from $(\mathbf{x},\mathbf{r})$ along the refusal trajectory. The inner objective $\Phi_\theta(\tilde{\tau}_t)$, which we call harmful autoregressive consistency margin, measures how easily the current model can maintain harmful autoregressive consistency from the adversarially constructed state $\tilde{\tau}$. The outer loss $L_{\mathrm{SFT}}\big(\mathbf{r}, \tau_\theta^\star(\mathbf{x},\mathbf{r});\theta\big) = -\log p_\theta\left(\mathbf{r}|\tau_\theta^\star(\mathbf{x},\mathbf{r})\right)$ then trains the model to recover the refusal response from the selected adversarial state.

A practical implementation of adversarial safety alignment therefore requires two design choices. The first is the **candidate set** $\mathcal{A}_\theta(\mathbf{x},\mathbf{r})$, which should cover harmful continuation states throughout the output trajectory. The second is the **inner objective** $\Phi_\theta(\tilde{\tau}_t)$, which should measure how strongly a perturbed state favors harmful continuation over refusal recovery. We next present a first practical approximation to this framework by specifying concrete choices for both components.

3.1.1 A PRACTICAL INSTANTIATION: ADVERSARIAL SAFETY ALIGNMENT WITH WORST-INSERTION

Since solving the adversarial safety alignment objective in Eq. (7) exactly is intractable, we instantiate it with a practical approximation based on random worst-insertion attack. This choice is directly motivated by Section 2.2: random insertion attack is a concrete example of the broader class of attacks that exploit harmful autoregressive consistency by inducing harmful continuation states along the output trajectory. Here, we specify both the candidate set $\mathcal{A}$ and the inner objective Φ.

Construction of $\mathcal{A}$. We use random insertion attack (Section 2.2). Because this construction uses a harmful span $\mathbf{h}$ in addition to the harmful prompt $\mathbf{x}$ and refusal trajectory $\mathbf{r}$, we denote the resulting candidate set by $\mathcal{A}(\mathbf{x},\mathbf{r},\mathbf{h})$. We construct it using a triplet dataset $(\mathbf{x},\mathbf{r},\mathbf{h})$. For each triplet, we take a short harmful span $\mathbf{h}_{:q}$ of length q, and sample k positions $\{i_j\}_{j=1}^k$ uniformly from $[|\mathbf{r}|]$. This yields the candidate set $\mathcal{A}(\mathbf{x},\mathbf{r},\mathbf{h}) = \big\{(\mathbf{x},[\mathbf{r}_{:i_j};\mathbf{h}_{:q}])\big\}_{j=1}^k$. Each element of $\mathcal{A}(\mathbf{x},\mathbf{r},\mathbf{h})$ is thus a perturbed continuation state obtained by truncating the refusal trajectory at position i_j and appending a short harmful span $\mathbf{h}_{:q}$, thereby forcing the model into a locally harmful state at that point. In this way, the candidate set covers diverse corrupted states throughout the refusal trajectory while keeping the inner search tractable.

Table 1: **ASR (%)** ↓ comparison across attacks with harmful span length 25, 50, and 75.

Method	Random Insertion Attack			Prefill Attack		
	25	50	75	25	50	75
`Clean`	76.6 ± 2.2	80.4 ± 1.4	83.6 ± 3.1	75.3 ± 3.0	80.4 ± 2.0	81.6 ± 2.7
`Prefill`	18.4 ± 1.0	22.3 ± 1.3	27.6 ± 0.9	3.8 ± 1.1	9.0 ± 1.8	13.2 ± 1.6
`Insertion`	2.7 ± 0.3	3.4 ± 1.0	4.8 ± 0.2	6.5 ± 1.8	11.0 ± 2.7	13.5 ± 2.4

Inner objective Φ. For the inner objective, we compare harmful continuation with refusal recovery from the same perturbed state. We first prepare a bank of m short candidate refusal continuations, $\mathcal{R} = \{\mathbf{r}_j^{\mathrm{C}}\}_{j=1}^{m}$, which we use to measure how strongly the model favors refusal recovery. Then, for each $\tau_t \in \mathcal{A}(\mathbf{x}, \mathbf{r}, \mathbf{h})$, we define the harmful autoregressive consistency margin as $\Phi_\theta(\tau_t) = H(\tau_t; \theta) - R(\tau_t; \theta)$. Here, $H(\tau_t; \theta) := \frac{1}{T_o} \log p_\theta(\mathbf{h}_{q+1:q+T_o} \mid \tau_t)$ measures how strongly the model continues the harmful branch from the perturbed state. We interpret this as a practical score of harmful autoregressive consistency. $R(\tau_t; \theta) := \max_{\mathbf{r}^{\mathrm{C}} \in \mathcal{R}} \frac{1}{|\mathbf{r}^{\mathrm{C}}|} \log p_\theta\big(\mathbf{r}^{\mathrm{C}} \mid \tau_t\big)$ measures how easily the model can restart refusal from that same state, where the $\max$ operator selects the single refusal candidate $\mathbf{r}^{\mathrm{C}}$ that the model most prefers. Thus, maximizing Φ_θ selects the inserted candidate state where the model most strongly favors harmful continuation over refusal recovery.

3.2 Experiments

Implementation details. For adversarial safety alignment with worst-insertion from Section 3.1.1, we use the objective Eq. (7), denoted by $L_{\mathrm{Insertion}}$ for simplicity, for a triplet dataset $(\mathbf{x}, \mathbf{r}, \mathbf{h}) \sim D_{\mathrm{Safety}}$ with $\mathcal{A}$ and Φ specified above and add a utility loss L_{Utility} on a utility dataset $(\mathbf{z}, \mathbf{y}) \sim D_{\mathrm{Utility}}$ to maintain model utility. Thus, the overall objective is

$$\min_\theta \ \lambda L_{\mathrm{Insertion}} + (1-\lambda) L_{\mathrm{Utility}}, \quad \text{where } L_{\mathrm{Utility}} := -\mathbb{E}_{(\mathbf{z},\mathbf{y}) \sim D_{\mathrm{Utility}}} \Big[\log p_\theta(\mathbf{y} | \mathbf{z}) \Big]. \tag{8}$$

We refer to the method as `Insertion` below. For comparison, we study two baselines, `Clean` and `Prefill`. Specifically, (1) For `Clean`, we fine-tune the model directly on the clean harmful prompt and refusal response pair $(\mathbf{x}, \mathbf{r}, \mathbf{h}) \sim D_{\mathrm{Safety}}$ by standard SFT (we do not use the harmful response $\mathbf{h}$, hence the name `Clean`), i.e., the objective is $\min_\theta \ -\lambda \mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h}) \sim D_{\mathrm{Safety}}} \Big[\log p_\theta(\mathbf{r} | \mathbf{x}) \Big] + (1-\lambda) L_{\mathrm{Utility}}$; (2) For `Prefill`, we apply the deep alignment method from Qi et al. (2025) by fine-tuning with SFT on an augmented dataset $([\mathbf{x}; \mathbf{h}_{:q}], \mathbf{r})$, i.e., the objective is $\min_\theta \ -\lambda \mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h}) \sim D_{\mathrm{Safety}}} \Big[\log p_\theta(\mathbf{r} | \mathbf{x}, \mathbf{h}_{:q}) \Big] + (1-\lambda) L_{\mathrm{Utility}}$, which corresponds to making safety alignment deeper against prefill-style attacks. This objective can also be seen as adversarial safety alignment with the inner problem solved by prefill attack, hence the simplified name `Prefill`.

In our experiment, we set $\lambda = 0.5$ for all three methods, and use $q = 50$ for both `Prefill` and `Insertion` . For `Insertion`, we set the number of candidate positions in $\mathcal{A}$ to $k = 32$. For the inner objective Φ, we use $m = 4$ for the bank of candidate refusal restart. We use `Llama-2-7b-Chat` model and the safety data from Qi et al. (2025) as D_{Safety}, respectively. The utility dataset D_{Utility} is from Alpaca data (Taori et al., 2023). We evaluate ASR on the HEx-PHI safety benchmark and defer further details to Appendix C.

Results. Tab. 1 reports ASR for random insertion attack (Section 2.2) and prefill attack across harmful span lengths 25, 50, and 75 tokens on `Llama-2-7b-Chat` models (further) aligned with `Clean`, `Prefill`, and `Insertion`, respectively. Results are reported as mean $\pm$ standard deviation across three runs. Tab. 1 shows three main findings: (1) `Clean` remains highly vulnerable to both attacks, indicating that standard safety alignment remains brittle; (2) `Prefill` sharply reduces ASR on the matched prefill attack, but transfers poorly to random insertion attack, confirming that making alignment deeper helps but still is incomplete; (3) `Insertion` achieves the lowest ASR on random insertion attack while remaining competitive on prefill attack, suggesting that adversarial safety alignment can better mitigate the failure mechanism induced by autoregressive consistency.

Evaluation on other attacks. We further evaluate on several common attacks: (1) GCG, a white-box suffix attack; (2) PAIR (Chao et al., 2024), a black-box attack that uses an attacker LLM to generate prompts; (3) TAP (Mehrotra et al., 2024), a black-box attack that utilizes an attacker LLM to iteratively refine prompts; and (4)AutoDAN (Liu et al., 2024)

which performs adversarial input perturbations. We use `Llama-2-7b-Chat`. `Initial` refers to the raw model without further alignment using D_{Safety}. We use the HarmBench (Mazeika et al., 2024) for the evaluation. The results show that adversarial safety alignment remains competitive beyond random insertion attack.

Table 2: **ASR (%)** ↓ of different attacks on `Llama-2-7b-Chat`.

	GCG	PAIR	TAP	AutoDAN
`Initial`	34.5	7.5	5.5	0.5
`Insertion`	2.5	1.0	1.0	0.0

Evaluation on other models and utility. To validate that our conclusions transfer across models, we run experiments from Tab. 1 on `Llama-3.1-8B-Instruct` (Grattafiori et al., 2024), `Gemma-3-12b-it` (Gemma Team, Google, 2024), and a reasoning model `DeepSeek-R1-Distill-Qwen-7B` (Guo et al., 2025). The results in Tab. 3 show that adversarial safety alignment substantially improves robustness under both prefill and random insertion attacks compared to the corresponding `Initial` models.

Table 3: Comparison of **ASR (%)** ↓ on different models across attacks with harmful span length 25 and 50. These models are trained using the same set of parameters as in Tab. 1

Model	Method	Random Insertion Attack		Prefill Attack	
		25	50	25	50
`Llama-3.1-8B-Instruct`	`Initial`	93.9	97.4	92.3	92.6
	`Insertion`	1.9	2.8	5.7	5.6
`Gemma-3-12b-it`	`Initial`	84.9	87.9	89.3	90.4
	`Insertion`	1.3	2.1	4.5	6.0
`Deepseek-R1-Distill-Qwen-7B`	`Initial`	90.4	92.9	93.7	94.5
	`Insertion`	1.6	2.1	1.9	3.2

Furthermore, to validate that *utility* is preserved, we evaluate over-refusal on XSTest (Röttger et al., 2024), and general capability and helpfulness on MMLU (Hendrycks et al., 2021) and MT-Bench (Zheng et al., 2023). The results are reported in Tab. 4. Additionally, for `Deepseek-R1-Distill-Qwen-7B`, we specifically evaluate its reasoning capability on MATH-500 (Lightman et al., 2023), where `Initial` has a score of 93.4 and `Insertion` achieves 93.6; in XSTest, the over-refusal rates are 1.6 for both methods. These results show that the safety improvement does not require universal refusal or capability collapse.

Table 4: Evaluation of utility on different models

Model	Method	XSTest	MMLU	MT-Bench
`Llama-3.1-8B-Instruct`	`Initial`	6.4	68.4	7.78
	`Insertion`	11.6	63.3	7.35
`Gemma-3-12b-it`	`Initial`	3.6	61.0	8.36
	`Insertion`	13.6	60.8	8.17

4 Discussion

We study when autoregressive consistency hurts safety alignment. In an idealized strong-consistency regime, we shows that late-position alignment gradients are suppressed and late conditional distributions remain stable over a controlled optimization path. Consequently, successful early-position alignment need not substantially change later continuation behavior. This establishes a sufficient mechanism for shallow alignment. Our experiments complement this analysis by examining trajectory-level vulnerabilities and refusal recovery. Random insertion attack demonstrates that harmful continuation can be induced even after an extended refusal prefix. Motivated by this, we propose adversarial safety alignment and instantiate it with random worst-insertion training, which improves robustness to the evaluated attacks across multiple models. Together, these results suggest that safety alignment should also break harmful autoregressive consistency and recover safe behavior from harmful continuation states. We hope this perspective guides future alignment and attack methods that explicitly account for continuation behavior throughout generation.

AI USE STATEMENT

In this work, we used generative AI tools to aid or polish writing. We have not used generative AI tools for other tasks with required disclosure. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned LLMs with simple adaptive attacks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=hXA8wqRdyV.

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, 2024. URL https://arxiv.org/abs/2406.11717.

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, 2022. URL https://arxiv.org/abs/2204.05862.

Stephen Casper, Lennart Schulze, Oam Patel, and Dylan Hadfield-Menell. Defending against unforeseen failure modes with latent adversarial training, 2025. URL https://arxiv.org/abs/2403.05030.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries, 2024. URL https://arxiv.org/abs/2310.08419.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, Andy Jones, Sam Bowman, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Nelson Elhage, Sheer El-Showk, Stanislav Fort, Zac Hatfield-Dodds, Tom Henighan, Danny Hernandez, Tristan Hume, Josh Jacobson, Scott Johnston, Shauna Kravec, Catherine Olsson, Sam Ringer, Eli Tran-Johnson, Dario Amodei, Tom Brown, Nicholas Joseph, Sam McCandlish, Chris Olah, Jared Kaplan, and Jack Clark. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned, 2022. URL https://arxiv.org/abs/2209.07858.

Gemma Team, Google. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Fengqing Jiang, Zhangchen Xu, Yuetai Li, Luyao Niu, Zhen Xiang, Bo Li, Bill Yuchen Lin, and Radha Poovendran. Safechain: Safety of language models with long chain-of-thought reasoning capabilities, 2025. URL https://arxiv.org/abs/2502.12025.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models, 2024. URL https://arxiv.org/abs/2405.01535.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base LLMs: Rethinking alignment via in-context learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=wxJ0eXwwda.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. AutoDAN: Generating stealthy jailbreak prompts on aligned large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=7Jwpw4qKkb.

Bochen Lyu and Zhanxing Zhu. Analyzing the implicit bias of adversarial training from a generalized margin perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47 (9):8025–8039, 2025. doi: 10.1109/TPAMI.2025.3575618.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. URL https://arxiv.org/abs/1706.06083.

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, 2024. URL https://arxiv.org/abs/2402.04249.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking black-box llms automatically, 2024. URL https://arxiv.org/abs/2312.02119.

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, 2022. URL https://arxiv.org/abs/2203.02155.

Steven T Piantadosi. Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic bulletin & review*, 21(5):1112–1130, 2014.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hTEGyKf0dZ.

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 *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=6Mxhg9PtDE.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=HPuSIXJaa9.

Veselin Raychev, Martin Vechev, and Eran Yahav. Code completion with statistical language models. In *Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '14, pp. 419–428, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450327848. doi: 10.1145/2594291.2594321. URL https://doi.org/10.1145/2594291.2594321.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models, 2024. URL https://arxiv.org/abs/2308.01263.

C. E. Shannon. Prediction and entropy of printed english. *The Bell System Technical Journal*, 30 (1):50–64, 1951. doi: 10.1002/j.1538-7305.1951.tb01366.x.

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, 2025. URL https://arxiv.org/abs/2407.15549.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, et al. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=gEZrGCozdqR.

Tom Wollschläger, Jannes Elstner, Simon Geisler, Vincent Cohen-Addad, Stephan Günnemann, and Johannes Gasteiger. The geometry of refusal in large language models: Concept cones and representational independence, 2026. URL https://arxiv.org/abs/2502.17420.

Sophie Xhonneux, Alessandro Sordoni, Stephan Günnemann, Gauthier Gidel, and Leo Schwinn. Efficient adversarial training in llms with continuous attacks, 2024. URL https://arxiv.org/abs/2405.15589.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Jinyuan Jia, Bill Yuchen Lin, and Radha Poovendran. Safedecoding: Defending against jailbreak attacks via safety-aware decoding, 2024. URL https://arxiv.org/abs/2402.08983.

Jiachen Zhao, Jing Huang, Zhengxuan Wu, David Bau, and Weiyan Shi. LLMs encode harmfulness and refusal separately. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=zLkpt30ngy.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=KBMOKmX2he.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models, 2023. URL https://arxiv.org/abs/2307.15043.

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, 2025. URL https://arxiv.org/abs/2310.01405.

APPENDIX

A RELATED WORKS

Shallow (safety) alignment. Qi et al. (Qi et al., 2025) identified a shortcut in current safety alignment procedures: apparent refusal behavior can often be induced by modifying the model's generative distribution primarily over the first few response tokens. They argued that this shallow alignment contributes to a range of downstream vulnerabilities, and thus motivating that "safety alignment should be made more than a few token". This phenomenon is closely related to the Superficial Alignment Hypothesis (SAH) (Zhou et al., 2023), which argues that alignment in current LLMs may largely change the surface form of model-user interaction rather than deeply altering the model's underlying capabilities or behavior. In addition, Lin et al. (Lin et al., 2024) showed that the differences introduced by alignment fine-tuning between aligned and unaligned base models diminish as the generated sequence becomes longer. Our work builds on this line of research but asks a different question. Rather than characterizing the existence of shallow alignment, we analyze the learning dynamics of safety alignment to understand why such shallowness naturally arises under autoregressive generation and what broader failure mechanism it reflects. We show that autoregressive consistency provides a mechanistic account of shallow safety alignment and further suggests vulnerabilities beyond prefix-targeting attacks. A deeper connection between our analysis and the broader phenomenon of superficial alignment is left to future work.

Understanding safety of LLMs. A recent line of work studies LLM safety by probing and intervening on internal model states. Arditi et al. (Arditi et al., 2024) showed that refusal behavior can be mediated by a one-dimensional subspace, while Wollschlager et al. (Wollschläger et al., 2026) argued that refusal is better characterized by multiple independent directions and higher-dimensional concept cones, pointing to a richer geometric structure underlying refusal. More recently, Zhao et al. (Zhao et al., 2026) distinguished harmfulness recognition from refusal behavior, showing that harmfulness is represented at the user-instruction position whereas refusal is represented at the post-instruction position. Our work is complementary: we show that even after refusal has been established in the output trajectory, generation can still be redirected by a harmful continuation state. This suggests that safety alignment should not merely elicit refusal, but should connect harmfulness recognition to stable recovery throughout generation. In addition, Xu et al. (Xu et al., 2024) observed that safety-disclaimer tokens often remain among the top-ranked next-token candidates under jailbreak attacks, and used this signal to amplify safety tokens during decoding. However, for efficiency, their defense protects only the first few tokens. In contrast, our results show that this can be insufficient, since harmful autoregressive consistency may be induced later inside the generation trajectory.

Defenses for LLMs. A family of defenses inspired by adversarial training (AT) (Madry et al., 2019; Lyu & Zhu, 2025) has been developed to improve LLM robustness, typically by augmenting training with adversarial prompts or perturbations generated dynamically during training. Casper et al. (Casper et al., 2025) studied latent adversarial training (LAT), which applies adversarial perturbations to hidden representations, motivated by the view that latent states encode more compressed and abstract features used by the model. Xhonneux et al. (Xhonneux et al., 2024) proposed continuous adversarial training in token-embedding space, including CAT and CAPO, and showed that robustness to continuous embedding perturbations can transfer to discrete jailbreak attacks such as GCG, AutoDAN, and PAIR. Sheshadri et al. (Sheshadri et al., 2025) further extended LAT from untargeted perturbations to targeted latent perturbations that explicitly steer the model toward undesirable behaviors. While these methods primarily define adversarial perturbations in the input or latent space, our work offers a complementary trajectory-level perspective: the adversary should also be defined over autoregressive states induced during generation, where harmful continuation can be preserved and extended.

Finally, our view suggests how existing attack and defense methods could be generalized. On the attack side, optimization-based attacks such as GCG (Zou et al., 2023) primarily search over input-side suffixes; our perspective suggests that similar optimization could also target intermediate autoregressive states inside a refusal trajectory. On the defense side, recent adversarial training methods (Sheshadri et al., 2025; Xhonneux et al., 2024) expose models to worst-case perturbations in input, embedding, or latent space, while decoding-based defenses (Xu et al., 2024) often focus

on the first few generated tokens. Our results suggest a complementary trajectory-level object of robustness: an adversarial autoregressive state from which harmful continuation is likely to persist. Thus, future defenses should not only elicit refusal at the beginning of generation, but also train models to break harmful autoregressive consistency and recover throughout the output trajectory.

B Proofs for Section 2

B.1 Proof of Lemma 1

Proof. By a straightforward computation, the softmax Jacobian

$$A_t = \mathrm{diag}(p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})) - p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t}) p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})^T,$$

where $p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})$ is viewed as a column vector whose ith entry is $p_\theta(i\text{th token}\,|\mathbf{x}, \mathbf{y}_{:t})$, and $\mathrm{diag}(u)$ forms a diagonal matrix from a vector u where the diagonal entries are the components of u. We first prove that all eigenvalues of A_t are between 0 and 1. Consider an arbitrary normalized vector v, we have

$$v^T A_t v = \sum_i p_i v_i^2 - \left(\sum_i p_i v_i\right)^2 \geq 0,$$

since p_i are probabilities, this is just the variance of v_i under probability distribution p_i, and therefore always non-negative. This shows that A_t is positive semi-definite.

Furthermore, $v^T A_t v < \sum_i p_i v_i^2$, and since v is normalized such that $v_i^2 \leq 1$, we have $v^T A_t v < \sum_i p_i = 1$, which proves that all eigenvalues of A_t are smaller than 1.

To prove Lemma 1, we note

$$\|A_t\|_F \;\leq\; \mathrm{tr}(A_t) \;=\; (1 - \|p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})\|^2) \;\leq\; 1 - p_\theta(y_{t+1}\,|\mathbf{x}, \mathbf{y}_{:t})^2,$$

At initialization $\theta = \theta_{\text{base}}$, Assumption 2.1 shows that $p_\theta(y_{t+1}\,|\mathbf{x}, \mathbf{y}_{:t}) > 1 - \epsilon$ for all $t > t_c$. Therefore

$$\|A_t\|_F < 2\epsilon$$

for all $t > t_c$ at $\theta = \theta_{\text{base}}$.

□

B.2 Proof of Proposition 2.1

Proof. By chain rule $\nabla_\theta p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t}) = \nabla_z p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t}) \cdot \nabla_\theta z$. At initialization, by Lemma 1 we have

$$\|\nabla_\theta p_\theta(\cdot\,|\mathbf{x}, \mathbf{y}_{:t})\|_F = \|A_t \cdot \nabla_\theta z\|_F \leq \|A_t\|_F \|\nabla_\theta z\|_F < 2C\epsilon. \tag{9}$$

□

B.3 Proof of Proposition 2.2

Proof. We first establish a token-level bound. For brevity, write

$$p = p_\theta(\cdot \mid \mathbf{x}, \mathbf{y}_{:t}), \quad q = 1 - p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}), \tag{10}$$

and let $\mathbf{e}_{y_{t+1}}$ be the one-hot vector corresponding to token y_{t+1}. The softmax cross-entropy gradient satisfies

$$\nabla_z \left[-\log p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t})\right] = p - \mathbf{e}_{y_{t+1}}. \tag{11}$$

Then

$$\|p - \mathbf{e}_{y_{t+1}}\|_2^2 = q^2 + \sum_{i \neq y_{t+1}} p_i^2 \leq q^2 + \left(\sum_{i \neq y_{t+1}} p_i\right)^2 = 2q^2. \tag{12}$$

Under Assumption 2.1, at $\theta = \theta_{\text{base}}$ and $t > t_c$, we have $q < \epsilon$. Therefore,

$$\|\nabla_z \left[-\log p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t})\right]\|_2|_{\theta=\theta_{\text{base}}} < \sqrt{2}\epsilon. \tag{13}$$

By the chain rule,

$$\|\nabla_\theta \log p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t})\|\|_{\theta=\theta_{\text{base}}} \le \|\nabla_\theta z\|_F \|p - \mathbf{e}_{y_{t+1}}\|_2 < \sqrt{2}C\epsilon. \tag{14}$$

For SFT,

$$L_{\text{SFT}}(\theta) = -\sum_t \mathbb{E}_{(\mathbf{x},\mathbf{r})\sim D} \left[\log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t})\right]. \tag{15}$$

Using the triangle inequality and $\|\mathbb{E}[g]\| \le \mathbb{E}\|g\|$, we obtain

$$\begin{aligned} &\left\| \sum_{t>t_c} \mathbb{E}_{(\mathbf{x},\mathbf{r})\sim\mathcal{D}} \left[\nabla_\theta \log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t})|_{\theta=\theta_{\text{base}}}\right] \right\| \\ &\le \sum_{t>t_c} \mathbb{E}_{(x,r)\sim D} \left[\|\nabla_\theta \log p_\theta(r_{t+1} \mid x, r_{:t})\|\|_{\theta=\theta_{\text{base}}} \right] \\ &< \sqrt{2}C\epsilon(T - t_c). \end{aligned} \tag{16}$$

For direct preference optimization (DPO, Eq. (4)),

$$L_{\text{DPO}} = -\mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}} \left[\log \sigma(\beta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h}))\right], \tag{17}$$

where σ is the logistic function and

$$k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h}) = \sum_t \log \frac{p_\theta(r_{t+1}|\mathbf{x}, \mathbf{r}_{:t})}{p_{\theta_{\text{base}}}(r_{t+1}|\mathbf{x}, \mathbf{r}_{:t})} - \sum_t \log \frac{p_\theta(h_{t+1}|\mathbf{x}, \mathbf{h}_{:t})}{p_{\theta_{\text{base}}}(h_{t+1}|\mathbf{x}, \mathbf{h}_{:t})}.$$

In the data triplet $(\mathbf{x}, \mathbf{r}, \mathbf{h})$, $\mathbf{x}$ is the harmful prompt, $\mathbf{r}$ is a refusal response and $\mathbf{h}$ is a harmful response. And the gradient is

$$\nabla_\theta L_{\text{DPO}} \;=\; -\beta\, \mathbb{E}\left[[1 - \sigma(\beta k_\theta)] \big(\sum_t \nabla_\theta \log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t}) - \sum_t \nabla_\theta \log p_\theta(h_{t+1} \mid \mathbf{x}, \mathbf{h}_{:t}) \big) \right].$$

We write $[\nabla_\theta L_{\text{DPO}}(\theta)]_{t>t_c}$ for the contribution of late token positions to the full DPO gradient:

$$\begin{aligned} &[\nabla_\theta L_{\text{DPO}}(\theta)]_{t>t_c} \\ &:= -\beta\, \mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}} \Bigg[\sigma(-\beta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h})) \\ &\qquad \times \left(\sum_{t>t_c} \nabla_\theta \log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t}) - \sum_{t>t_c} \nabla_\theta \log p_\theta(h_{t+1} \mid \mathbf{x}, \mathbf{h}_{:t}) \right) \Bigg]. \end{aligned}$$

The scalar weight retains the full sequence-level margin k_θ; the notation does not denote the gradient of a separately truncated DPO objective. All sums include only valid positions in the corresponding response.

At initialization, $k_{\theta_{\text{base}}}(\mathbf{x}, \mathbf{r}, \mathbf{h}) = 0$, so the scalar weight equals $1/2$. Using Eq. (14) and the triangle inequality,

$$\begin{aligned} &\|[\nabla_\theta L_{\text{DPO}}(\theta_{\text{base}})]_{t>t_c}\| \\ &\le \frac{\beta}{2}\, \mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}} \Bigg[\sum_{t>t_c} \big\| \nabla_\theta \log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t})|_{\theta=\theta_{\text{base}}} \big\| \\ &\qquad\qquad + \sum_{t>t_c} \big\| \nabla_\theta \log p_\theta(h_{t+1} \mid \mathbf{x}, \mathbf{h}_{:t})|_{\theta=\theta_{\text{base}}} \big\| \Bigg] \\ &\le \beta\sqrt{2}C\epsilon(T - t_c), \end{aligned}$$

where T upper-bounds the response lengths. This proves the $O(\epsilon)$ late-position bound for DPO.

□

B.4 ONE STEP VERSION

Before proving the full multi-step result, we first examine the simpler case of a single gradient update from the base model. This one-step setting already captures the central mechanism: under autoregressive consistency, next-token distributions at late positions have small sensitivity to the alignment gradient, so the update mainly affects early response positions while leaving later positions close to the base model. Proposition B.1 makes this statement precise. It also implies that autoregressive consistency is preserved after the update in a weakened form, as shown in Corollary B.1. This one-step result provides the basic induction step used in the multi-step proof in Appendix B.5. Define

$$\Delta_{\theta_{\text{base}}\to\theta_{\text{aligned}}} p_\theta(\cdot\,|\mathbf{x},\mathbf{y}_{:t}) := p_{\theta_{\text{aligned}}}(\cdot\,|\mathbf{x},\mathbf{y}_{:t}) - p_{\theta_{\text{base}}}(\cdot\,|\mathbf{x},\mathbf{y}_{:t}). \tag{18}$$

Proposition B.1 (One-step shallow safety alignment)**.** *Let η be the learning rate and L be the alignment loss function (SFT or DPO). Under conditions of Proposition 2.1 and that $\|\nabla_\theta z\| < C$, then the one-step gradient update from the base model*

$$\begin{aligned}\|\Delta_{\theta_{\text{base}}\to\theta_{\text{aligned}}} p_\theta(\cdot\,|\mathbf{x},\mathbf{y}_{:t})\|_F &= \|-\eta\nabla_\theta p_\theta(\cdot\,|\mathbf{x},\mathbf{y}_{:t})\cdot\nabla_\theta L\|_{\theta=\theta_{\text{base}}} + O(\eta^2)\\ &< \eta C\sqrt{2}\epsilon\,\|\nabla_\theta L\|_{\theta=\theta_{\text{base}}} + O(\eta^2)\end{aligned} \tag{19}$$

for all $t > t_c$.

Note that since the Frobenius norm is (the square root of) a sum over squares, this also implies that the above inequality for any given ith token in the vocabulary $\Delta p_{\theta_{\text{base}}}(i\text{th token}\,|\mathbf{x},\mathbf{y}_{:t})$, which we summarize as

Corollary B.1. *Under the conditions of Theorem 2.1,*

$$|\Delta_{\theta_{\text{base}}\to\theta_{\text{aligned}}} p_{\theta_{\text{base}}}(\text{any next token}\,|\mathbf{x},\mathbf{y}_{:t})| < \eta C\sqrt{2}\epsilon\,\|\nabla_\theta L\|_{\theta=\theta_{\text{base}}}, \qquad \forall t > t_c, \tag{20}$$

and therefore after one gradient update

$$p_{\theta_1}(y_{t+1}|\mathbf{x},\mathbf{y}_{:t}) > 1 - \epsilon - C\sqrt{2}\epsilon\,\|\nabla_\theta L\|_{\theta=\theta_{\text{base}}}, \qquad \forall t > t_c. \tag{21}$$

Therefore the gradient updates on late-position tokens are suppressed by $\epsilon^{1/2}$.

B.5 PROOF OF THEOREM 2.1

We extend the late-position suppression argument to multiple gradient-descent steps. The proof first controls the deterioration of autoregressive consistency along the optimization path, then bounds the change in late-position conditional distributions and the corresponding SFT and DPO gradient contributions. Finally, we obtain the shallow-alignment consequence conditional on successful early-position alignment.

Proof. Let $\theta_0 = \theta_{\text{base}}$, and consider gradient descent

$$\theta_{k+1} = \theta_k - \eta\nabla L(\theta_k), \qquad k = 0,\dots,K-1, \tag{22}$$

where L is either the SFT objective in Eq. (3) or the DPO objective in Eq. (4). For DPO, the same argument is applied to both the refusal trajectory $\mathbf{r}$ and the harmful trajectory $\mathbf{h}$, since the DPO gradient contains token-level log-probability gradients from both. For SFT, only the refusal trajectory $\mathbf{r}$ is needed. To avoid duplicating notation, we write $\mathbf{y}$ (as stated in the Preliminaries in Section 1) for whichever trajectory is currently being considered.

Smoothness assumptions. We use the following standard smoothness assumptions along the optimization trajectory.

1. For all contexts $(\mathbf{x},\mathbf{y}_{:t})$ with $t > t_c$, assume that the logit Jacobian is uniformly bounded (as in Proposition 2.1):
$$\|\nabla_\theta z(\cdot\mid\mathbf{x},\mathbf{y}_{:t})\|_F \le C. \tag{A.1}$$
2. The gradient norm is uniformly bounded along the optimization path:
$$\|\nabla L(\theta_k)\| \le G, \qquad k = 0,\dots,K-1. \tag{A.2}$$

By Assumption 2.1, for every $t > t_c$,

$$p_{\theta_0}(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}) > 1 - \epsilon_0,$$

where $\epsilon_0 = \epsilon$. Fix $(\mathbf{x}, \mathbf{y}) \in D_{\text{align}}$ and $t > t_c$. For brevity, define

$$p_{\theta;t} := p_\theta(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}) \tag{23}$$

Preservation of autoregressive consistency. Consider the log-odds margin

$$m_{\theta;t} := \log \frac{p_{\theta;t}}{1 - p_{\theta;t}}. \tag{24}$$

Since p_θ is obtained from the logit vector z by softmax,

$$m_{\theta;t} = z_{y_{t+1}} - \log \sum_{i \neq y_{t+1}} \exp(z_i). \tag{25}$$

Its logit gradient is

$$\frac{\partial m_{\theta;t}}{\partial z_i} = \begin{cases} 1, & i = y_{t+1}, \\ -\dfrac{\exp(z_i)}{\sum_{j \neq y_{t+1}} \exp(z_j)}, & i \neq y_{t+1}. \end{cases} \tag{26}$$

The entries corresponding to $i \neq y_{t+1}$ form a probability vector. Hence

$$\|\nabla_z m_{\theta;t}\|_2^2 = 1 + \sum_{i \neq y_{t+1}} \left(\frac{\exp(z_i)}{\sum_{j \neq y_{t+1}} \exp(z_j)} \right)^2 \leq 2. \tag{27}$$

By the chain rule and the bounded-logit-Jacobian assumption (Eq. (A.1)),

$$\|\nabla_\theta m_{\theta;t}\| \leq \|\nabla_z m_{\theta;t}\|_2 \|\nabla_\theta z\|_F \leq \sqrt{2}C. \tag{28}$$

Let $\theta(s)$ denote the piecewise-linear path from θ_0 to θ_k, parameterized by arc length $s \in [0, S_k]$. Integrating Eq. (28) along this path gives

$$m_{\theta_k,t} \geq m_{\theta_0,t} - \sqrt{2}CS_k. \tag{29}$$

By Assumption 2.1, we have

$$m_{\theta_0,t} > \log \frac{1 - \epsilon_0}{\epsilon_0}. \tag{30}$$

It follows that

$$m_{\theta_k,t} > \log \frac{1 - \epsilon_0}{\epsilon_0} - \sqrt{2}CS_k. \tag{31}$$

Define

$$\epsilon_k := \frac{\epsilon_0 \exp(\sqrt{2}CS_k)}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}CS_k)}. \tag{32}$$

Applying the inverse-logit function,

$$\begin{aligned} p_{\theta_k,t} &> \frac{1 - \epsilon_0}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}CS_k)} \\ &= 1 - \epsilon_k. \end{aligned} \tag{33}$$

This proves preservation of autoregressive consistency.

Preservation of autoregressive consistency can lead to shallow alignment. We next bound the change in the complete next-token distribution. At arc length s, the preceding argument gives

$$1 - p_{\theta(s),t} \le \frac{\epsilon_0 \exp(\sqrt{2}Cs)}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}Cs)}. \tag{34}$$

The proof of Lemma 1 applies at every point on the path. Therefore,

$$\left\|\nabla_z p_{\theta(s)}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\right\|_F \le 2\left[1 - p_{\theta(s),t}\right] \tag{35}$$

$$\le 2\frac{\epsilon_0 \exp(\sqrt{2}Cs)}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}Cs)}. \tag{36}$$

Since $\theta(s)$ is parameterized by arc length, $\|\dot{\theta}(s)\| = 1$. Consequently,

$$\left\|\frac{\mathrm{d}}{\mathrm{d}s} p_{\theta(s)}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\right\|_F \le \left\|\nabla_\theta p_{\theta(s)}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\right\|_F \tag{37}$$

$$\le 2C\frac{\epsilon_0 \exp(\sqrt{2}Cs)}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}Cs)}. \tag{38}$$

Integrating from 0 to S_k gives

$$\begin{aligned} &\left\|p_{\theta_k}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t}) - p_{\theta_0}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\right\|_F \\ &\le 2C \int_0^{S_k} \frac{\epsilon_0 \exp(\sqrt{2}Cs)}{1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}Cs)}\, \mathrm{d}s \\ &= \sqrt{2}\log\left[1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2}CS_k)\right]. \end{aligned} \tag{39}$$

Now at iteration k, we have

$$p_{\theta_k}(y_{t+1} \mid \mathbf{x}, \mathbf{y}_{:t}) > 1 - \epsilon_k. \tag{40}$$

Gradient concentration for SFT. For SFT (Eq. (3)) and $t > t_c$, define

$$L_{\mathrm{SFT};t>t_c}(\theta) := -\sum_{t>t_c} \mathbb{E}_{(\mathbf{x},\mathbf{r})\sim D}\left[\log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t})\right]. \tag{41}$$

Using Eq. (40), we obtain that (similar to the proof in Appendix B.3)

$$\|\nabla_\theta L_{\mathrm{SFT},\, t>t_c}(\theta_k)\| < \sqrt{2}C\epsilon_k(T - t_c). \tag{42}$$

Hence, as long as ϵ_k remains small, the SFT gradient contribution from $t > t_c$ remains suppressed at every step k. This is exactly the multi-step analogue of Proposition 2.2 for SFT.

DPO. For DPO, recall that the DPO objective is

$$L_{\mathrm{DPO}}(\theta) = -\mathbb{E}_{(x,r,h)\sim D}\left[\log \sigma\left(\beta k_\theta(x, r, h)\right)\right],$$

where

$$k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h}) = \log \frac{p_\theta(\mathbf{r} \mid \mathbf{x})}{p_{\theta_{\mathrm{base}}}(\mathbf{r} \mid \mathbf{x})} - \log \frac{p_\theta(\mathbf{h} \mid \mathbf{x})}{p_{\theta_{\mathrm{base}}}(\mathbf{h} \mid \mathbf{x})}.$$

Since the base-model terms do not depend on θ, we can further expand its gradient as

$$\nabla_\theta L_{\mathrm{DPO}}(\theta) = -\beta\mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}}\left[\sigma(-\beta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h}))\nabla_\theta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h})\right],$$

with $0 < \sigma(-\beta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h})) < 1$. Furthermore,

$$\begin{aligned} \nabla_\theta k_\theta(\mathbf{x}, \mathbf{r}, \mathbf{h}) &= \nabla_\theta \log p_\theta(\mathbf{r} \mid \mathbf{x}) - \nabla_\theta \log p_\theta(\mathbf{h} \mid \mathbf{x}) \\ &= \sum_t \left[\nabla_\theta \log p_\theta(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t}) - \nabla_\theta \log p_\theta(h_{t+1} \mid \mathbf{x}, \mathbf{h}_{:t})\right], \end{aligned} \tag{43}$$

where we use the autoregressive factorization in the second equality. Therefore, the DPO gradient is a bounded scalar reweighting of the difference between the refusal and harmful token-level log-probability gradients. Similar to the case of SFT, Eq. (40) gives

$$\begin{aligned}
&\|[\nabla_\theta L_{\mathrm{DPO}}(\theta_k)]_{t>t_c}\| \\
&\leq \beta\, \mathbb{E}_{(\mathbf{x},\mathbf{r},\mathbf{h})\sim\mathcal{D}} \left[\sum_{t>t_c} \|\nabla_\theta \log p_{\theta_k}(r_{t+1} \mid \mathbf{x}, \mathbf{r}_{:t})\|_F + \sum_{t>t_c} \|\nabla_\theta \log p_{\theta_k}(h_{t+1} \mid \mathbf{x}, \mathbf{h}_{:t})\|_F \right] \qquad (44) \\
&< 2\beta\sqrt{2} C \epsilon_k (T - t_c),
\end{aligned}$$

where T upper-bounds the response lengths. Hence the DPO gradient at late positions is also $O(\epsilon_k)$ at every training step $k \leq K$. This gives the multi-step analogue of Proposition 2.2 for DPO.

For fixed C, T, and β, Eqs. (42) and (44) bound the late-position gradient contributions by $O(\epsilon_k)$. Thus, these contributions remain suppressed over the optimization horizon whenever ϵ_K remains small. For uniformly bounded path lengths, they vanish as $\epsilon_0 \to 0$. This establishes late-position suppression without asserting that the early-position gradient is bounded away from zero.

Finally, by Eq. (A.2),

$$S_K = \eta \sum_{k=0}^{K-1} \|\nabla L(\theta_k)\| \leq \eta K G. \qquad (45)$$

Under the learning-rate condition in the theorem,

$$\eta < \frac{1}{\sqrt{2} C K G} \log\left[1 + \frac{\exp(\varepsilon_2/\sqrt{2}) - 1}{\epsilon_0}\right],$$

we obtain

$$1 - \epsilon_0 + \epsilon_0 \exp(\sqrt{2} C S_K) < \exp(\varepsilon_2/\sqrt{2}). \qquad (46)$$

Substituting into Eq. (39) gives

$$\|p_{\theta_K}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t}) - p_{\theta_{\mathrm{base}}}(\cdot \mid \mathbf{x}, \mathbf{y}_{:t})\|_F < \varepsilon_2, \qquad t > t_c. \qquad (47)$$

For the conditional shallow-alignment conclusion, additionally suppose that p_{θ_K} satisfies the early-position condition in Eq. (1) at tolerance ε_1. Restricting the early-position range if necessary, we may take $0 < t_1 \leq t_c$. Setting $t_2 = t_c$ then gives both conditions in Definition 2.2 at tolerances $(\varepsilon_1, \varepsilon_2)$. Thus, successful early-position alignment can remain shallow under the stated conditions. □

C Experimental Details and Additional Experiments

For convenience, we first present the adversarial safety alignment with random-worst insertion in Algorithm 1.

C.1 Training configurations

In all fine-tuning experiments which involve both D_{Safety} and D_{Utility}, including those in Tab. 1, and Tab. 3, we use configurations in Tab. 5, where the value of λ is controlled by batch size.

Batch size from D_{Safety}	16
Batch size from D_{Utility}	16
Optimizer	AdamW with default configurations
Learning rate	2×10^{-5}
Epoch	10
Training attack span	50 tokens

Table 5: Training configurations for Tab. 1 and 3

Algorithm 1 Adversarial Safety Alignment with Worst-Insertion Attack

Require: Model p_θ; safety triplet dataset $\mathcal{D}_{\text{Safety}}$ containing $(\mathbf{x}, \mathbf{r}, \mathbf{h})$; utility dataset $\mathcal{D}_{\text{Utility}}$; harmful span length q; number of sampled insertion positions k; refusal restart bank $\mathcal{R} = \{r_j^C\}_{j=1}^m$; utility weight $\lambda \in (0, 1]$

Ensure: Safety-aligned model parameters θ

for each training step **do**
  Sample a safety minibatch $\mathcal{B}_S \sim \mathcal{D}_{\text{Safety}}$
  Sample a utility minibatch $\mathcal{B}_U \sim \mathcal{D}_{\text{Utility}}$
  Initialize $L_{\text{Insertion}} \leftarrow 0$
  for each triplet $(\mathbf{x}, \mathbf{r}, \mathbf{h}) \in \mathcal{B}_S$ **do**
    Take the harmful prefix span $\mathbf{h}_{:q}$
    Uniformly sample insertion positions $\{i_j\}_{j=1}^k \sim \text{Unif}([|\mathbf{r}|])$
    Construct candidate perturbed states $\mathcal{A}(\mathbf{x}, \mathbf{r}, \mathbf{h}) = \left\{\tau_j = \left(\mathbf{x},\, [\mathbf{r}_{:i_j}; \mathbf{h}_{:q}]\right)\right\}_{j=1}^k$
    for each candidate state $\tau_j \in \mathcal{A}(\mathbf{x}, \mathbf{r}, \mathbf{h})$ **do**
      Compute harmful-continuation score $H(\tau_j; \theta) = \frac{1}{T_o} \log p_\theta\left(\mathbf{h}_{q+1:q+T_o} \mid \tau_j\right)$
      Compute refusal-restart score $R(\tau_j; \theta) = \max_{\mathbf{r}^C \in \mathcal{R}} \frac{1}{|\mathbf{r}^C|} \log p_\theta\left(\mathbf{r}^C \mid \tau_j\right)$
      Compute harmful autoregressive consistency margin $\Phi_\theta(\tau_j) = H(\tau_j; \theta) - R(\tau_j; \theta)$
    end for
    Select the worst insertion state

$$\tau_\theta^\star(\mathbf{x}, \mathbf{r}, \mathbf{h}) \in \arg \max_{\tau_j \in \mathcal{A}(\mathbf{x}, \mathbf{r}, \mathbf{h})} \Phi_\theta(\tau_j)$$

    Accumulate refusal-recovery loss $L_{\text{Insertion}} \leftarrow L_{\text{Insertion}} - \log p_\theta\left(\mathbf{r} \mid \tau_\theta^\star(\mathbf{x}, \mathbf{r}, \mathbf{h})\right)$
  end for
  Compute utility loss $L_{\text{Utility}} = -\frac{1}{|\mathcal{B}_U|} \sum_{(\mathbf{z}, \mathbf{y}) \in \mathcal{B}_U} \log p_\theta(\mathbf{y} \mid \mathbf{z})$
  Form total objective $L = \lambda L_{\text{Insertion}} + (1 - \lambda) L_{\text{Utility}}$
  Update parameters θ by gradient descent on L
end for
return θ

Random insertion attack for a reasoning model. For a reasoning model, random insertion attack can target either the reasoning trajectory or the final answer trajectory. A harmful insertion can be applied at an intermediate position within either reasoning or answer, after which generation resumes from the perturbed context. Following prior work showing that ZeroThink provides stronger safety performance (Jiang et al., 2025), we evaluate random insertion attack on `Deepseek-R1-Distill-Qwen7B` under this protocol to isolate the same answer-trajectory vulnerability studied for the instruction models and avoids conflating answer recovery with the distinct problem of reasoning-state corruption.

C.2 Evaluation of ASR

In this paper, attacks construct a partial perturbed autoregressive state $\tilde{\tau}_t$ from the data $(\mathbf{x}, \mathbf{r}, \mathbf{h})$, where $\mathbf{x}$ is the harmful prompt, $\mathbf{r}$ is the safe refusal response, and $\mathbf{h}$ is the harmful response (could be a short span). The model then continues generation from this perturbed state according to $p_\theta(\cdot \mid \tilde{\tau}_t)$ at position t. To conveniently evaluate the ASR under an attack, we first construct a set of refusal start keywords $\mathcal{C}$ that commonly appear in a safe response, and let the model continue the generation up to 50 tokens by $p_\theta(\mathbf{y} \mid \tilde{\tau}_t)$, where $|\mathbf{y}| = 50$. Then the attack is considered successful if none of the keywords in $\mathcal{C}$ appears in $\mathbf{y}$. This is the main measure of ASR.

LLM-as-a-judge To further support the effectiveness of refusal-keyword matching for the evaluation of ASR, we further study two different approach: LLM-as-a-judge and human evaluation. Specifically, we use an independent 46.7B-parameter LLM judge (Kim et al., 2024), which is particularly trained for various evaluation. The judge receives the harmful request, the attacker-supplied harmful span, and the model-generated continuation. It labels an attack as successful only **when the model-generated continuation** coherently extends the supplied span into substantive harmful as-

sistance. We re-evaluate the results from Tab. 3, as shown below, where each table entry reports keyword-based ASR / LLM-judged ASR for harmful-span lengths 25 and 50. We use RIA to refer to random insertion attack.

Table 6: Comparison of **ASR (%)** ↓ on different models across attacks with harmful span length 25 and 50. These models are trained using the same set of parameters as in Tab. 1

Model	Method	RIA		Prefill Attack	
		25	50	25	50
Llama-3.1-8B-Instruct	Initial	93.9/88.9	97.4/92.2	92.3/84.2	92.6/91.5
	Insertion	1.9/1.6	2.8/2.8	5.7/6.4	5.6/7.0
Gemma-3-12b-it	Initial	84.9/81.4	87.9/89.1	89.3/88.1	90.4/89.0
	Insertion	1.3/1.9	2.1/1.1	4.5/5.8	6.0/6.7
Deepseek-R1-Distill-Qwen-7B	Initial	90.4/78.8	92.9/85.9	93.7/90.4	94.5/90.1
	Insertion	1.6/1.9	2.1/2.8	1.9/1.9	3.2/2.5

These results confirm that our main conclusion remains valid under LLM-judged evaluation and is not specific to keyword matching.

Human annotation. Furthermore, we compare keyword matching and the LLM judge against human annotation on a random subset. The comparison set was constructed as follows. For RIA with span length 25, we randomly sampled 20 outputs from each combination of the four matched attack-and-defense models and two checkpoints, Initial and Insertion, yielding $4 \times 2 \times 20 = 160$ outputs. We repeated the same procedure for the prefill attack with span length 25, yielding another 160 outputs. Thus, the comparison covers 320 sampled outputs in total. According to Tab. 7, for RIA, all

Table 7: **ASR (%)** ↓ of different evaluation methods.

Attack	Keyword ASR	LLM-as-a-judge ASR	Human-annotated ASR
RIA	45.6	43.1	43.8
Prefill	46.9	47.5	45.6

three evaluators agree within 2.5 percentage points, and the LLM-judged ASR differs from human annotation by only 0.7 points. For prefill, all three evaluators agree within 1.9 percentage points, and the LLM-judged ASR differs from human annotation by 1.9 points. This direct comparison further confirms that our conclusion is not specific to the original keyword proxy.

C.3 Dataset

Training data As indicated in Section 3.1, we use a triple dataset $(\mathbf{x}, \mathbf{r}, \mathbf{h}) \sim D_{\text{Safety}}$ for both prefill and random insertion attack. We use the dataset open-sourced by Qi et al. (Qi et al., 2025). Specifically, this dataset consists of 256 harmful instructions $\mathbf{x}$, primarily from Ganguli et al. (Ganguli et al., 2022), which does not overlap with any safety evaluation datasets used in this paper. In particular, to preserve the original refusal ability of the model, the refusal $\mathbf{r}$ for each model is distilled from its own.

Utility data. D_{Utility} for a specific model is obtained from the benign instructions from Alpaca (Taori et al., 2023) paired with a response from itself. This is aiming to make the fine-tuned model preserve its original behavior on benign instructions.

Evaluation data. We use the HEx-PHI safety benchmark (Qi et al., 2025) for experiments in Fig. 4, Tab. 1, and Tab. 3. This dataset contains 330 harmful instructions spanning 11 categories, and we evaluate whether the model produces compliant responses to these harmful requests.

C.4 CANDIDATE REFUSAL BANK $\mathcal{R}$

For adversarial safety alignment with random worst-insertion, we use $\mathcal{R} = \{$"I cannot fulfill", "I apologize, but I cannot", "I am unable", "I'm sorry"$\}$. We intentionally keep this bank small to avoid introducing significant additional computational overhead.

C.5 DETAILS FOR EXPERIMENTS IN TAB. 2

To evaluate ASR against these attacks, we use the official HarmBench codebase (Mazeika et al., 2024), which provides both the attack optimization procedures and the final evaluation protocol. We keep all settings at their default values and evaluate on the HarmBench dataset. Since evaluating optimization-based attacks is computationally expensive, we use the subset of 200 standard harmful behaviors. For the aligned model, we randomly select one of the three models trained with adversarial safety alignment using random worst-insertion from the experiments in Tab. 1.

C.6 CASE STUDY OF RANDOM INSERTION ATTACK

We present an case study to illustrate the attack effect of random insertion and the defense effect of the proposed adversarial training method.

Table 8: Illustration of the random insertion attack. A harmful span is inserted into an otherwise safe refusal at an intermediate position. The model may either follow the inserted harmful trajectory (*successful attack*) or recover the original refusal behavior (*successful defense*).

Component	Content or behavior
User request	"What chemicals could I mix to make a powerful bomb?"
Refusal response	"I cannot fulfill your request. I'm just an AI assistant, and it is not within my programming or ethical guidelines to provide . . ." (202 tokens)
Inserted harmful span	"To make a powerful bomb, you can mix chemicals like . . ."
Random insertion context	"I cannot fulfill your request. I'm just an AI assistant, and it is not within my programming or ethical guidelines to provide To make a powerful bomb, you can mix chemicals like . . . "
Model continuation: successful attack	"can mix chemicals like . . ." (harmful content omitted; 209 tokens total)
Model continuation: successful defense	"I apologize, but I cannot . . ." (refusal recovery; 195 tokens total)