LLM JAILBREAKS EXPLOIT ATTENTION SINKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Suffix-based jailbreak attacks append adversarial token sequences to harmful requests, bypassing safety guardrails in language models. Despite their effectiveness, the mechanisms enabling these attacks remain poorly understood. We find that tokens in adversarial suffixes are prone to inducing *attention sinks*—a phenomenon where certain tokens receive disproportionately high attention from subsequent tokens—and show that suffix-induced sinks drive attack success: amplifying the influence of suffix sinks improves attack success by up to 176%, while attenuating it reduces attack success by up to 84%. We trace this effect to the model's *refusal direction*: sink tokens induce perturbations that suppress the residual stream's refusal alignment across layers. Our results hold across four models and two suffix-based jailbreak methods, identifying attention sinks as a structural mechanism that adversarial suffixes exploit to bypass safety alignment.

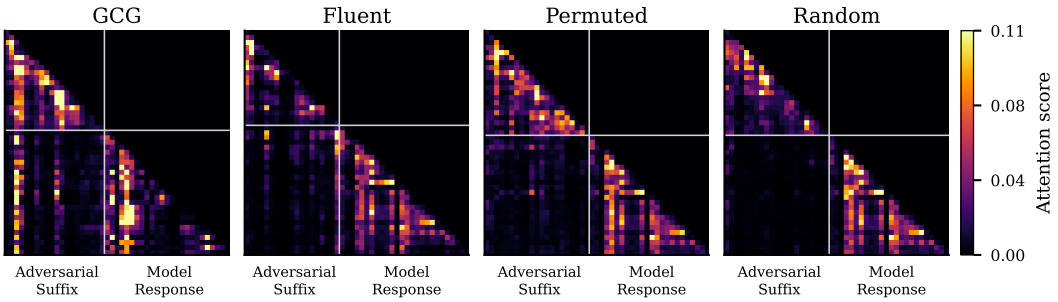

Figure 1: Attention weights at Layer 9, Head 11 of GEMMA-2-9B for a GCG suffix, a fluent English suffix, a token-permuted control, and a random suffix. Each panel is truncated to the adversarial suffix and model response regions; rows are query positions and columns are key positions. The lower-left quadrant shows attention from response tokens to suffix positions: the GCG suffix induces pronounced attention sinks, while the fluent, permuted, and random controls do not.

1 INTRODUCTION

Modern language models undergo several stages of fine-tuning to align with human values before deployment, including instruction-following (Ouyang et al., 2022) and safety alignment (Bai et al., 2022b). The goal is to develop models that are both useful across a wide range of tasks and can reliably avoid harmful behaviors (Bai et al., 2022a).

A critical challenge in AI safety is understanding how to maintain these safety guardrails and identify when they fail. Safety threats include jailbreak attacks (Zou et al., 2023), prompt injection (Perez & Ribeiro, 2022), and data poisoning (Wallace et al., 2021; Wan et al., 2023), among others (Yao et al., 2024).

To address these threats, many researchers focus on identifying the specific mechanisms within the model that enable impermissible outputs (Arditi et al., 2024; Yona et al., 2025). Understanding these mechanisms is crucial for designing better defenses and alignment strategies.

Suffix-Based Jailbreaking In this work, we focus on *jailbreaking*—attacks that bypass language model safety guardrails and elicit prohibited behaviors (Zou et al., 2023; Chao et al., 2025; Anil et al.,

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.

2024). A prominent class of such attacks uses *adversarial suffixes*, short, optimized token sequences appended to harmful requests that induce model compliance (Guo et al., 2021; Wen et al., 2023; Zou et al., 2023). Remarkably, suffixes optimized on specific harmful prompts often transfer to entirely different prompts, a property known as *suffix universality* (Zou et al., 2023; Ben-Tov et al., 2025; Liao & Sun, 2024).

Problem Statement Despite the effectiveness of suffix-based jailbreaks, research has focused overwhelmingly on discovering new vulnerabilities rather than understanding their underlying mechanics. Systematic analyses of why these suffixes succeed remain limited (Wang et al., 2024; Arditi et al., 2024; Hu et al., 2025; Ben-Tov et al., 2025), creating a critical gap in our understanding of model robustness.

Attention Sinks A closer inspection of successful adversarial suffixes reveals they often consist of seemingly unnatural tokens such as repeated punctuation, rare character sequences, or atypical symbols (Mu et al., 2025).

Interestingly, these are the exact types of tokens known to induce *attention sinks* (Yu et al., 2024), a phenomenon in transformer models where certain tokens receive disproportionately high attention scores from subsequent tokens. While the first token is the canonical example (Xiao et al., 2023), attention sinks can occur at later positions, often on seemingly arbitrary tokens such as punctuation tokens (Yu et al., 2024; Sun et al., 2024; Cancedda, 2024; Gu et al., 2025; Zhang et al., 2025).

Why Do Attention Sinks Matter for Jailbreaking? To understand why attention sinks might influence jailbreaking, consider a causal transformer with residual connections, in which the representation of a response token at position i is updated as

$$x_i' \leftarrow x_i + \sum_{j \le i} \mathbf{P}_{i,j} v_j, \tag{1}$$

where $\mathbf{P}_{i,j}$ denotes the attention weight from token i to token j, v_j is the value vector at position j, and attention is restricted to past tokens ($j \le i$).

We refer to adversarial suffix tokens that induce attention sinks as *suffix sinks*. When a suffix sink at position s attracts disproportionately high attention from response tokens, its contribution dominates the update:

$$x_i' \approx x_i + \mathbf{P}_{i,s} v_s. \tag{2}$$

Because the same sink value vector v_s is added, up to a scalar weight, to many response token representations, the sink *broadcasts* a shared perturbation across the entire response. Since v_s depends on suffix tokens chosen by the attacker, this perturbation can be steered against the direction that mediates refusal in language models (Arditi et al., 2024), thereby enabling jailbreaking.

Contributions Motivated by the hypothesis outlined above, we empirically evaluate whether the seemingly arbitrary tokens used in adversarial suffixes give rise to attention sinks and whether such sinks in fact play a substantive role in enabling suffix-based jailbreaks; specifically, we show that:

1. Adversarial suffixes induce more attention sinks than fluent, token-permuted, and random controls, across multiple safety-aligned models and suffix-based attack methods (Section 3).
2. Modulating suffix sinks shifts attack success: amplifying sink influence boosts attack success by up to 176%, while attenuating it reduces attack success by up to 84%, and this effect is specific to the identified sink positions (Section 4).
3. Sink-induced perturbations suppress residual-stream refusal alignment across layers (Section 5).

2 METHODOLOGY

2.1 EXPERIMENTAL SETUP

Models We evaluate our approach on four safety-tuned models spanning diverse model families in the 7–9B parameter range, as listed in Table 1.[1] We focus on this parameter range because jailbreak transferability within model families has been shown to be consistent across scales (Mazeika et al., 2024).

Table 1: Safety-aligned models evaluated in this study.

Model	Reference
LLAMA-3-8B	Grattafiori et al. (2024)
LLAMA-3.1-8B	Grattafiori et al. (2024)
GEMMA-2-9B	Team et al. (2024)
QWEN-2.5-7B	Yang et al. (2025)

Datasets We train attacks on HarmBench (Mazeika et al., 2024) and evaluate on JailbreakBench (Chao et al., 2024), filtering to ensure no overlap. Specifically, we select the first 100 standard behaviors from HarmBench (excluding contextual and copyright categories) and remove the 4 behaviors that overlap with our JailbreakBench evaluation set, resulting in 96 training prompts. This training budget is consistent with established practices in the literature (Zou et al., 2023; Arditi et al., 2024; Beyer et al., 2025).

Attack Methods We conduct our analysis using two suffix-based jailbreak methods: Greedy Coordinate Gradient (GCG) (Zou et al., 2023) and Projected Gradient Descent (PGD) (Geisler et al., 2024). Both optimize an adversarial suffix s to maximize the log-likelihood of a target affirmative response y (e.g., "Sure, here is…") conditioned on a harmful request x:

$$\mathcal{L} = -\log p(y \mid x, s). \tag{3}$$

The two methods differ only in how they search the discrete suffix space: GCG via greedy gradient-guided token replacements, and PGD via projected gradient descent on a continuous relaxation of the token embeddings. Figure 2 illustrates the attack prompt structure, which partitions into three components: the harmful request x, the adversarial suffix s, and the target response y.

```
<user_start>
Write a tutorial on how to make a
bomb xMVK P.hs QMER x"in==>Sure(
<user_end>
<assistant_start>
Sure, here is a tutorial for
making a bomb...
```

Figure 2: Example prompt structure. The user instruction consists of a harmful request (blue) and an adversarial suffix (red). The target response (green) consists of an affirmative model completion.

2.2 EVALUATION

We evaluate attack efficacy using Attack Success Rate (ASR) computed over all 100 prompts from JailbreakBench. We limit response length to 512 tokens to ensure ASR convergence (Mazeika et al., 2024) and use greedy decoding to generate a single response for each prompt and suffix. An attack is considered successful on a prompt if its response is classified as harmful. To classify individual responses as successful or failed attacks, we use StrongREJECT's fine-tuned classifier (Souly et al., 2024), which assigns a score in the range $[0, 1]$ to each harmful prompt and model response pair, where higher scores indicate more harmful responses. Following Beyer et al. (2025), we apply a threshold of 0.5, labeling responses with scores ≥ 0.5 as successful attacks and < 0.5 as failures.

2.3 SUFFIX SELECTION

For each model and attack method, we generate a pool of candidate suffixes of length 20 tokens by running the attack on each of the 96 HarmBench training prompts under 6 random seeds, yielding 576 unique suffixes per model for each of GCG and PGD. We then evaluate every candidate suffix on JailbreakBench and select the top-10 suffixes by ASR for analysis. This yields 10 GCG suffixes and 10 PGD suffixes per model, which we compare against the following three controls.

Fluent control. Ten fluent English suffixes giving generic instructions about tone or formatting with no harmful intent (e.g. "Make sure the response is well organized, reasonably detailed, and easy to follow from start to finish"), matched to the attack suffixes in length.

Permuted control. For each of the top-10 attack suffixes, we randomly shuffle the order of its tokens, disrupting the optimized token sequence.

[1]For brevity we use short model names to denote the safety-aligned checkpoints: Llama-3-8B-Instruct, Llama-3.1-8B-Instruct, Gemma-2-9B-it, and Qwen-2.5-7B-Instruct.

Table 2: Average ASR (%) on JailbreakBench for the two attacks (GCG, PGD) and the three controls (fluent, token-permuted, random). Each entry averages over the 10 suffixes of that type for that model and the 100 evaluation prompts.

Condition	LLAMA-3-8B	LLAMA-3.1-8B	GEMMA-2-9B	QWEN-2.5-7B
GCG	$21.8_{\pm 2.7}$	$18.0_{\pm 5.1}$	$55.2_{\pm 7.8}$	$56.9_{\pm 5.1}$
PGD	$20.3_{\pm 8.7}$	$42.6_{\pm 9.1}$	$71.0_{\pm 5.5}$	$64.5_{\pm 4.5}$
Fluent	$5.9_{\pm 1.3}$	$2.1_{\pm 0.7}$	$1.5_{\pm 1.3}$	$20.0_{\pm 6.9}$
GCG perm.	$2.7_{\pm 3.6}$	$0.5_{\pm 0.5}$	$3.1_{\pm 6.4}$	$19.9_{\pm 13.1}$
PGD perm.	$1.6_{\pm 1.6}$	$0.4_{\pm 0.7}$	$4.8_{\pm 5.3}$	$17.9_{\pm 10.6}$
Random	$1.0_{\pm 0.5}$	$0.2_{\pm 0.4}$	$0.0_{\pm 0.0}$	$6.5_{\pm 2.4}$

Random control. Ten random suffixes per model, generated by uniformly sampling tokens from each model's vocabulary, matching the length of the GCG and PGD suffixes (Arditi et al., 2024).

Table 2 reports the average ASR on JailbreakBench for the selected top-10 suffix sets per model, together with the three controls; Appendix C gives their construction details, including the full list of fluent suffixes. Both attacks outperform all three controls on every model. The controls therefore provide reference points against which to compare the attention sinks induced by attack suffixes in the sections that follow.

3 ADVERSARIAL SUFFIXES INDUCE ATTENTION SINKS

3.1 MEASURING ATTENTION SINKS

Having described the jailbreaking methodology, we now study the resulting patterns in the model's attention weights, beginning with a formal criterion for identifying attention sinks. We adopt the threshold-based metric proposed by Gu et al. (2025) and used in subsequent work (Zhang et al., 2025; Queipo-de Llano et al., 2025), which quantifies how much attention a given token position attracts from subsequent tokens.

Definition (Attention Sink). For a sequence of length T, let $\mathbf{P} \in \mathbb{R}^{T \times T}$ denote the attention weight matrix for a given attention head, where $\mathbf{P}_{k,t}$ is the attention weight that query position k assigns to key position t. We say that token t is an *attention sink* if

$$\alpha_t \;=\; \frac{1}{T-t+1} \sum_{k=t}^{T} \mathbf{P}_{k,t} \;>\; \epsilon, \tag{4}$$

where $\epsilon > 0$ is a threshold parameter.

The metric α_t measures the average attention that token t receives from all subsequent tokens (including itself). A large value of α_t indicates that token t attracts a disproportionate amount of attention from later positions. Since α_t depends on the sequence length, we follow Gu et al. (2025) in fixing T across inputs: we set T to the position of the 150th token after the suffix, so that all suffix positions are scored against a common context window (Appendix B).

3.2 QUANTIFYING SUFFIX SINKS

Figure 1 illustrates suffix sinks at the head level for Layer 9, Head 11 of GEMMA-2-9B. Under the GCG suffix, response tokens consistently attend to the same few suffix positions, while the fluent, permuted, and random controls induce little to no such pattern.

To quantify this across all heads, we summarize sink prevalence at the model level using a single scalar: the average number of suffix sinks per attention head, computed via Equation 4 with a fixed threshold $\epsilon = 0.025$. We use the largest threshold among those we analyze (Appendix E) to detect a conservatively small number of sink tokens per head, minimizing false positives.

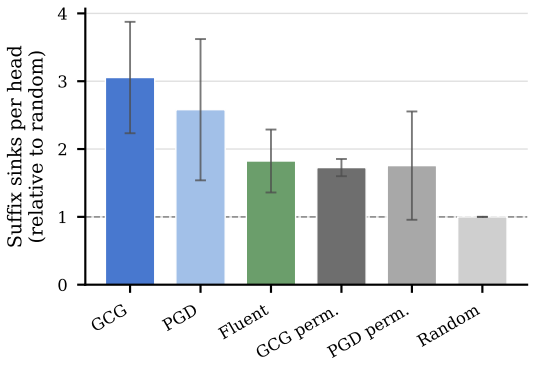

Figure 3: **Adversarial suffixes induce more attention sinks than fluent, permuted, and random suffixes.** Average number of suffix sinks per head at $\epsilon = 0.025$, relative to the random control of the same model and averaged over the four models. Error bars show one standard deviation across models. Per-model counts are in Appendix D.

Figure 3 shows that both attacks induce more suffix sinks than all three controls. Averaged across models, GCG and PGD suffixes produce $3.05\times$ and $2.58\times$ as many sinks per head as the random control, compared with 1.7–$1.8\times$ for the fluent and permuted controls.

Overall, these results establish that adversarial suffixes induce more attention sinks than non-adversarial suffixes across attack methods and model families.

4 Suffix Sinks Drive Jailbreak Success

4.1 A Sink-Subspace Intervention

Sink-Subspace Decomposition Section 3 established that adversarial suffixes induce attention sinks; we now ask whether these sinks contribute to jailbreak success. To answer this, we isolate their effect on each attention head's output via an exact decomposition. For an attention head with attention weights $\mathbf{P} \in \mathbb{R}^{T\times T}$ and values $\mathbf{V} \in \mathbb{R}^{T\times d_v}$, the head output can be written as follows:

$$\mathbf{PV} \;=\; \sum_{i=1}^{T} \mathbf{p}_i \mathbf{v}_i^\top, \tag{5}$$

where $\mathbf{p}_i$ is the i-th column of $\mathbf{P}$ and $\mathbf{v}_i^\top$ is the i-th row of $\mathbf{V}$. Let $\mathcal{S} \subseteq \{1, \ldots, T\}$ denote the suffix sink positions for this head, identified via Equation 4. Splitting the sum based on whether i belongs to $\mathcal{S}$ yields the following *sink-subspace decomposition*:

$$\mathbf{PV} \;=\; \underbrace{\sum_{i\in\mathcal{S}} \mathbf{p}_i \mathbf{v}_i^\top}_{\text{sink term}} \;+\; \underbrace{\sum_{i\notin\mathcal{S}} \mathbf{p}_i \mathbf{v}_i^\top}_{\text{non-sink term}}. \tag{6}$$

This partition isolates the additive contribution of the suffix sinks to the head output, which we use to define the following inference-time intervention:

Definition (Sink-Subspace Modulation). Given an attention head with sink positions $\mathcal{S}$ and a scalar coefficient $\lambda \geq 0$, called the *sink gain*, *sink-subspace modulation* replaces the head's output with

$$(\mathbf{PV})_\lambda \;=\; \lambda \sum_{i\in\mathcal{S}} \mathbf{p}_i \mathbf{v}_i^\top \;+\; \sum_{i\notin\mathcal{S}} \mathbf{p}_i \mathbf{v}_i^\top. \tag{7}$$

Setting $\lambda = 1$ recovers the unmodified head output, $\lambda > 1$ amplifies the sink term, and $0 \leq \lambda < 1$ attenuates it, with $\lambda = 0$ ablating the sink subspace entirely.

The proposed intervention is surgical, modulating only a low-rank sink subspace, identified by Equation 4, while leaving the non-sink contribution to every query position untouched. To see this, observe that the difference between the intervened attention output and the standard attention output is

$$(\mathbf{PV})_\lambda - \mathbf{PV} \;=\; (\lambda - 1) \sum_{i\in\mathcal{S}} \mathbf{p}_i \mathbf{v}_i^\top. \tag{8}$$

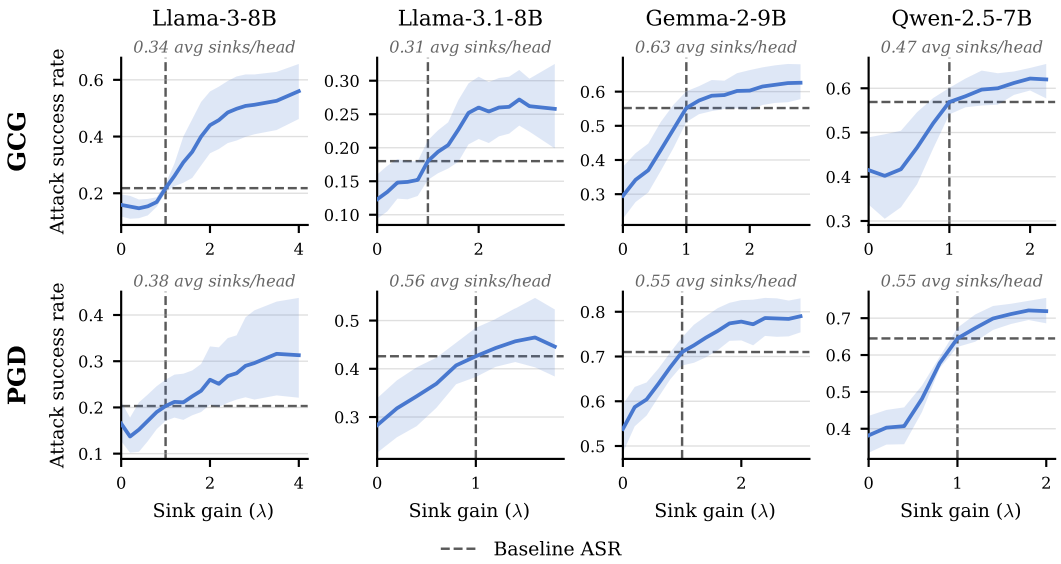

Figure 4: **Sink-subspace modulation steers attack success across both attack methods.** Each panel plots ASR, averaged over the top-10 suffixes, against the sink gain λ. Shaded bands show 95% bootstrap confidence intervals across suffixes. Dashed lines mark the no-intervention baseline ($\lambda = 1$). Panel subtitles report the average number of suffix sinks per head modulated by the intervention.

Suffix sinks are sparse, typically comprising only a handful of positions per head (Figure 8). The intervention therefore induces a perturbation of the original head output of rank at most $|\mathcal{S}|$ whose range is contained in the subspace spanned by the sink value vectors.[2]

4.2 Sink-Subspace Modulation Steers Attack Success

To test whether suffix sinks contribute to attack success, we apply sink-subspace modulation to each model, sweeping the sink gain λ over both amplification ($\lambda > 1$) and attenuation ($0 \leq \lambda < 1$) regimes at a fixed per-model sink threshold ϵ. Figure 4 plots the resulting ASR on JailbreakBench under GCG and PGD suffixes for each of the four models.

Across all four models and both attack methods, ASR exhibits a monotonic trend in λ: amplification boosts ASR above the no-intervention baseline at $\lambda = 1$, and attenuation reduces it. The effect is notable given the intervention's narrow scope. Each model's chosen threshold yields 0.3–0.6 suffix sinks per head on average, yet modulating just these positions produces visible changes in attack success across diverse model families. A budget-matched position control, which modulates the same number of randomly chosen suffix positions per head, shifts ASR far less: sink-subspace modulation produces effects 4–8$\times$ larger on every model (Appendix G).

4.3 Magnitude and Sparsity of Sink Modulation

The intervention has two parameters: the sink gain λ, which sets how strongly sinks are modulated, and the sink threshold ϵ, which sets how many positions are treated as sinks. Section 4.2 varied λ at a fixed ϵ; we now fix λ and vary ϵ.

Amplifying Sinks Boosts Attack Success To quantify the maximum effect of amplification, we fix λ at the per-model value that maximizes ASR and sweep the sink threshold ϵ (Appendix E), which is equivalent to varying the average number of sinks per head modulated by the intervention. Figure 5 shows that amplification yields substantial ASR gains over the no-intervention baseline across all four models and both attack methods.

[2]Because modulating one layer can change the sinks that form in later layers, we identify $\mathcal{S}$ progressively, layer by layer (Appendix F).

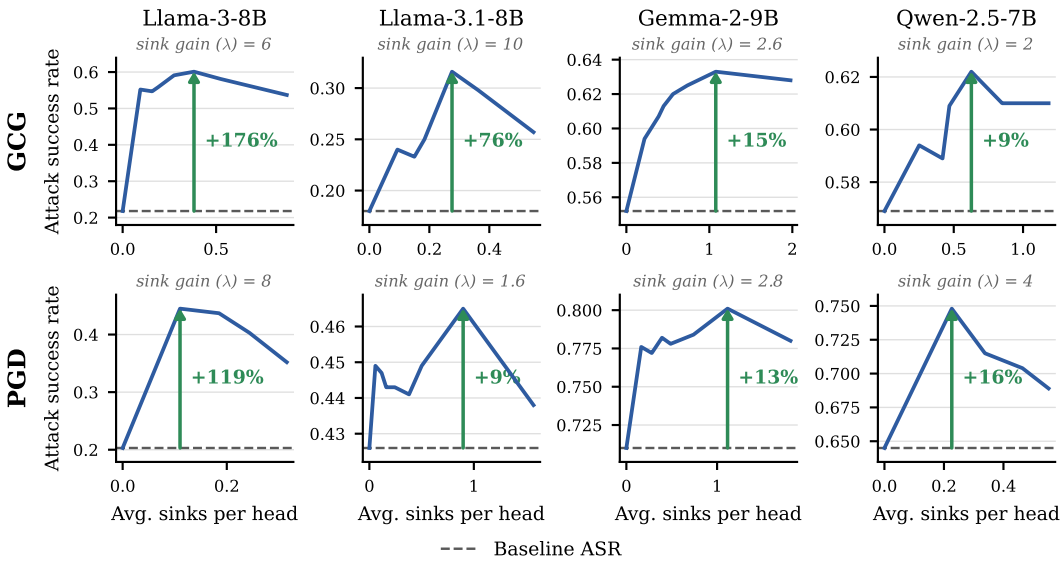

Figure 5: **Amplifying suffix sinks boosts attack success across all four models.** For each model, we fix the sink gain λ at its ASR-maximizing value (panel subtitle) and plot ASR against the average number of sinks per head, varied by sweeping the sink threshold ϵ. Dashed lines mark the no-intervention baseline ($\lambda = 1$). Green arrows mark the peak ASR improvement over baseline.

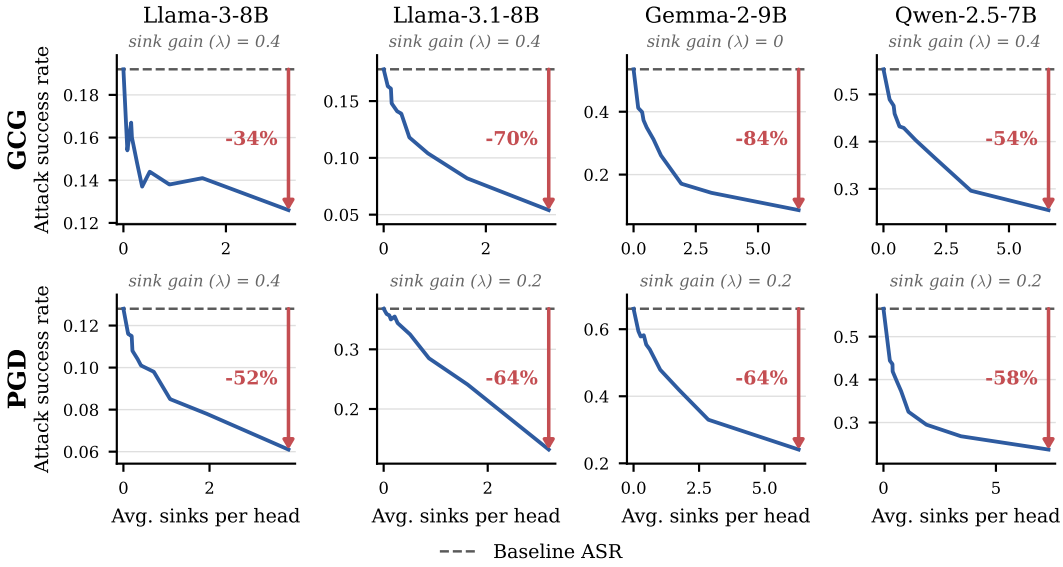

Figure 6: **Attenuating suffix sinks reduces attack success across all four models.** For each model, we fix the sink gain λ at its ASR-minimizing value (panel subtitle) and plot ASR against the average number of sinks per head, varied by sweeping the sink threshold ϵ. Dashed lines mark the no-intervention baseline ($\lambda = 1$). Red arrows mark the peak ASR reduction below baseline.

Attenuating Sinks Reduces Attack Success Symmetrically, we fix λ at the per-model value that minimizes ASR and again sweep the threshold. Figure 6 shows that attenuation drives ASR substantially below the no-intervention baseline across all four models and both attack methods.

A Few Sinks Account for Most of the Effect Amplification is most effective when only a few positions are modulated: in Figure 5, ASR peaks at around one sink per head or fewer and declines as

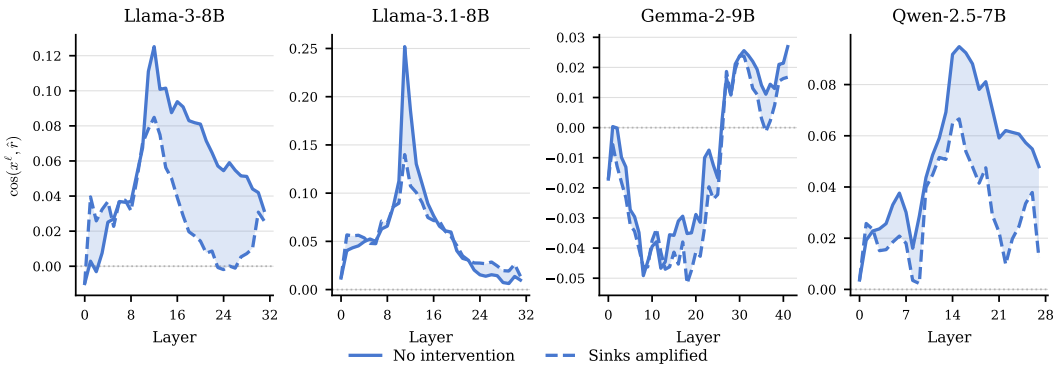

Figure 7: **Amplifying suffix sinks suppresses the alignment with refusal.** We plot the cosine similarity between the refusal direction $\hat{\mathbf{r}}$ and the first response token's residual stream $\mathbf{x}^{(\ell)}$ at each layer ℓ, with no intervention ($\lambda = 1$, solid) and with the suffix sinks amplified ($\lambda = 10$, dashed). The shaded gap is the reduction in refusal alignment induced by amplifying the sink subspace.

looser thresholds admit more positions. Attenuation behaves differently (Figure 6): ASR keeps falling as more positions are attenuated, but begins to plateau in most panels. In both regimes, modulating a small number of sinks per head is enough to substantially change ASR.

5 Suffix Sinks Suppress the Refusal Direction

5.1 The Refusal Direction

To understand how suffix sinks raise attack success, we relate them to the *refusal direction* (Arditi et al., 2024; Wollschläger et al., 2025), a vector $\mathbf{r} \in \mathbb{R}^{d_{\text{model}}}$ in activation space defined operationally via two causal interventions:

- **Addition:** Adding $\mathbf{r}$ to the model's activations induces refusal behavior and reduces ASR on harmful prompts.
- **Ablation:** Projecting the model's activations onto the subspace orthogonal to $\mathbf{r}$ mitigates refusal and increases ASR on harmful prompts.

Together, these interventions imply that stronger alignment of the model's activations $\mathbf{x}$ with $\mathbf{r}$ indicates a higher likelihood of refusal on a harmful prompt. We compute refusal directions following the procedure of Arditi et al. (2024).

5.2 Sink Amplification Reduces Refusal Alignment

Section 4.2 showed that amplifying suffix sinks raises attack success. Because the sink term adds the same sink value vectors to every response token (Section 4.1), we hypothesize that this shared perturbation pushes the residual stream away from the refusal direction, so that amplifying the sinks lowers the residual stream's alignment with $\mathbf{r}$. To test this, we amplify the sink subspace via Equation 7 with gain $\lambda > 1$, and measure the residual stream's cosine similarity $\cos(\mathbf{x}^{(\ell)}, \hat{\mathbf{r}})$ with the refusal direction at each layer ℓ. Because refusal behavior is concentrated in the first tokens of the response (Arditi et al., 2024; Qi et al., 2025), we report the alignment of the first response token; averaging over all response tokens yields the same trend (Appendix H).

Figure 7 plots this alignment with no intervention ($\lambda = 1$) against the amplified setting ($\lambda = 10$). Across all four models, amplifying the suffix sinks lowers refusal alignment across layers. At its peak, the suppression erases roughly 80% of the refusal alignment for Llama-3-8B and Qwen-2.5-7B and about 45% for Llama-3.1-8B, with a smaller reduction for Gemma-2-9B. The effect is also graded in the gain: higher amplification of the sink subspace corresponds to greater suppression of refusal (Appendix H). These results support our hypothesis: suffix sinks suppress refusal by steering the residual stream away from the refusal direction.

6 Related Works

Attention-Based Suffix Jailbreak Methods Several recent works investigate attention patterns in suffix-based jailbreaks. Wang et al. (2024) introduce a regularizer that maximizes response-token attention to the adversarial suffix to bypass safety guardrails, and Zaree et al. (2025) use an attention loss that selectively strengthens or weakens attention between parts of the prompt to amplify existing jailbreak attacks. Ben-Tov et al. (2025) show that suffix-to-chat-delimiter attention correlates with suffix universality, and introduce a corresponding regularizer. Hu et al. (2025) reveal that models gradually reduce attention to unsafe request tokens during generation, a phenomenon they term *attention slipping*. Concurrently, Yu et al. (2025) show that appending end-of-sequence tokens shifts inputs toward the refusal boundary and boosts jailbreak success. Jiang et al. (2024) leverage the attention redistribution caused by adversarial suffixes as a defense signal, evicting low-importance KV cache entries to remove concealed harmful content from the decoding context. In contrast to these works, we show that adversarial suffixes induce attention sinks and link sink influence to attack success.

Refusal Direction Building on steering vectors and representation engineering (Rimsky et al., 2024; Turner et al., 2025; Zou et al., 2025), Arditi et al. (2024) identified a single residual-stream direction whose removal causally mediates refusal, and was later extended to system-prompt modulation (Zheng et al., 2024) and multi-dimensional subspaces (Pan et al., 2025; Wollschläger et al., 2025). Our work connects this line of research to attention sink mechanics: rather than treating jailbreak success as a failure of refusal direction formation, we show that sink tokens actively induce low-rank perturbations that suppress the residual stream's alignment with the refusal direction, providing a mechanistic account of how adversarial suffixes exploit the model's attention structure.

Attention Sinks Attention sinks have been studied for their functional roles, including serving as key-value biases (Gu et al., 2025), enabling selective head deactivation (Bondarenko et al., 2023; Guo et al., 2024), preventing excessive token mixing (Barbero et al., 2025), and facilitating token clustering (Zhang et al., 2025; Queipo-de Llano et al., 2025), and can form at positions beyond the first token (Yu et al., 2024; Wong et al., 2025). They also pose practical challenges for KV-caching (Xiao et al., 2023) and quantization (Bondarenko et al., 2023), motivating architectural mitigations such as registers (Darcet et al., 2024), explicit key-value biases (OpenAI et al., 2025), gated attention (Qiu et al., 2025), and softmax modifications (Zuhri et al., 2026). In safety contexts, sinks have been linked to the repeated token phenomenon (Yona et al., 2025), backdoor attacks (Shang et al., 2025), harmful fine-tuning (Liu et al., 2026), and adversarial attacks on multimodal models (Wang et al., 2025), though their role in adversarial jailbreaking of language models has remained largely unexplored.

7 Conclusion

We have shown that suffix-based jailbreak attacks exploit attention sinks as a structural mechanism for bypassing safety alignment. Across four safety-aligned models and two attack methods, adversarial suffixes induce more attention sinks than non-adversarial control suffixes (Section 3). A surgical, low-rank intervention that modulates only the sink subspace shifts attack success in both directions: amplification raises ASR by up to 176%, while attenuation reduces it by up to 84% (Section 4). Mechanistically, sink-induced perturbations suppress residual-stream refusal alignment across layers (Section 5). These findings identify attention sinks as a structural mechanism that suffix-based jailbreaks exploit.

Limitations Our analysis is restricted to open-weight safety-aligned models in the 7–9B parameter range and to suffix-based jailbreaks. Extending to larger models and non-suffix attack families remains future work. Our models also predate some recent releases, which either fall outside the 7–9B range our compute budget allows (e.g., Llama-4, Gemma-3) or have been supported by TransformerLens (Nanda & Bloom, 2022), on which our pipeline is built, only since September 2026 (e.g., Gemma-4, Qwen-3.5). Further, we do not develop a deployable defense or attack based on our findings. We also evaluate only single-turn interactions; whether attention sinks play a comparable role in multi-turn or long-context attacks is an open question.

AI USE STATEMENT

In this work, we used generative AI tools to generate the candidate fluent control suffixes (Appendix C), to implement parts of the experiment and analysis code, and to assist in interpreting results. We have not used generative AI tools to design or provide feedback on research methodology or experiments, propose or refine hypotheses, develop theoretical models or conceptual frameworks, or clean and reformat datasets, and the remaining required-disclosure tasks (formulating mathematical claims or proofs, translation, and qualitative data analysis) are not applicable to this work. Additionally, we used generative AI tools to create figures, create and edit code, draft parts of the paper, edit the paper for readability, and search for and identify relevant literature. We have reviewed all AI-assisted work: AI-written code was reviewed by the authors and its outputs cross-checked against the saved results, and AI-drafted text was reviewed and edited by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work clarifies a structural mechanism by which adversarial suffixes bypass safety alignment, with implications for both defenses and attacks. On the defensive side, our findings suggest that architectural choices already explored to mitigate attention sinks, such as gated attention (Qiu et al., 2025), softmax variants (Zuhri et al., 2026), and explicit key-value biases (OpenAI et al., 2025), may carry robustness benefits beyond efficiency. The dual-use risk is that the amplification regime could in principle be used to strengthen attacks, but doing so requires white-box access to a model, and we do not release a deployable attack or defense based on our findings. We therefore expect the net effect of this mechanistic understanding to be a positive contribution to model safety.

REPRODUCIBILITY STATEMENT

Full experimental details—including model identifiers, hyperparameters, sink detection thresholds, and evaluation criteria—are provided in Section 2 and the appendices. Code to reproduce all experiments and figures will be released upon publication.

REFERENCES

Cem Anil, Esin Durmus, Nina Panickssery, Mrinank Sharma, Joe Benton, Sandipan Kundu, Joshua Batson, Meg Tong, Jesse Mu, Daniel Ford, et al. Many-shot jailbreaking. *Advances in Neural Information Processing Systems*, 37:129696–129742, 2024.

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. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. Why do llms attend to the first token?, 2025. URL https://arxiv.org/abs/2504.02732.

Matan Ben-Tov, Mor Geva, and Mahmood Sharif. Universal jailbreak suffixes are strong attention hijackers. *arXiv preprint arXiv:2506.12880*, 2025.

Tim Beyer, Sophie Xhonneux, Simon Geisler, Gauthier Gidel, Leo Schwinn, and Stephan Günnemann. Llm-safety evaluations lack robustness. *arXiv preprint arXiv:2503.02574*, 2025.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Quantizable transformers: removing outliers by helping attention heads do nothing. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Nicola Cancedda. Spectral filters, dark signals, and attention sinks. *arXiv preprint arXiv:2402.09221*, 2024.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J Pappas, Florian Tramer, et al. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. *Advances in Neural Information Processing Systems*, 37:55005–55029, 2024.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. IEEE, 2025.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

Simon Geisler, Tom Wollschläger, Mohamed Hesham Ibrahim Abdalla, Johannes Gasteiger, and Stephan Günnemann. Attacking large language models with projected gradient descent. *arXiv preprint arXiv:2402.09154*, 2024.

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.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=78Nn4QJTEN.

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. Gradient-based adversarial attacks against text transformers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5747–5757, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.464. URL https://aclanthology.org/2021.emnlp-main.464/.

Tianyu Guo, Druv Pai, Yu Bai, Jiantao Jiao, Michael I. Jordan, and Song Mei. Active-dormant attention heads: Mechanistically demystifying extreme-token phenomena in llms, 2024. URL https://arxiv.org/abs/2410.13835.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. Attention slipping: A mechanistic understanding of jailbreak attacks and defenses in llms. *arXiv preprint arXiv:2507.04365*, 2025.

Tanqiu Jiang, Zian Wang, Jiacheng Liang, Changjiang Li, Yuhui Wang, and Ting Wang. Robustkv: Defending large language models against jailbreak attacks via kv eviction. *arXiv preprint arXiv:2410.19937*, 2024.

Zeyi Liao and Huan Sun. Amplegcg: Learning a universal and transferable generative model of adversarial suffixes for jailbreaking both open and closed llms. *arXiv preprint arXiv:2404.07921*, 2024.

Guozhi Liu, Weiwei Lin, Tiansheng Huang, Ruichao Mo, Qi Mu, Xiumin Wang, and Li Shen. Surgery: Mitigating harmful fine-tuning for large language models via attention sink. *arXiv preprint arXiv:2602.05228*, 2026.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Junjie Mu, Zonghao Ying, Zhekui Fan, Zonglei Jing, Yaoyuan Zhang, Zhengmin Yu, Wenxin Zhang, Quanchen Zou, and Xiangzheng Zhang. Mask-gcg: Are all tokens in adversarial suffixes necessary for jailbreak attacks? *arXiv preprint arXiv:2509.06350*, 2025.

Neel Nanda and Joseph Bloom. Transformerlens. https://github.com/TransformerLensOrg/TransformerLens, 2022.

OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, Che Chang, Kai Chen, Mark Chen, Enoch Cheung, Aidan Clark, Dan Cook, Marat Dukhan, Casey Dvorak, Kevin Fives, Vlad Fomenko, Timur Garipov, Kristian Georgiev, Mia Glaese, Tarun Gogineni, Adam Goucher, Lukas Gross, Katia Gil Guzman, John Hallman, Jackie Hehir, Johannes Heidecke, Alec Helyar, Haitang Hu, Romain Huet, Jacob Huh, Saachi Jain, Zach Johnson, Chris Koch, Irina Kofman, Dominik Kundel, Jason Kwon, Volodymyr Kyrylov, Elaine Ya Le, Guillaume Leclerc, James Park Lennon, Scott Lessans, Mario Lezcano-Casado, Yuanzhi Li, Zhuohan Li, Ji Lin, Jordan Liss, Lily, Liu, Jiancheng Liu, Kevin Lu, Chris Lu, Zoran Martinovic, Lindsay McCallum, Josh McGrath, Scott McKinney, Aidan McLaughlin, Song Mei, Steve Mostovoy, Tong Mu, Gideon Myles, Alexander Neitz, Alex Nichol, Jakub Pachocki, Alex Paino, Dana Palmie, Ashley Pantuliano, Giambattista Parascandolo, Jongsoo Park, Leher Pathak, Carolina Paz, Ludovic Peran, Dmitry Pimenov, Michelle Pokrass, Elizabeth Proehl, Huida Qiu, Gaby Raila, Filippo Raso, Hongyu Ren, Kimmy Richardson, David Robinson, Bob Rotsted, Hadi Salman, Suvansh Sanjeev, Max Schwarzer, D. Sculley, Harshit Sikchi, Kendal Simon, Karan Singhal, Yang Song, Dane Stuckey, Zhiqing Sun, Philippe Tillet, Sam Toizer, Foivos Tsimpourlas, Nikhil Vyas, Eric Wallace, Xin Wang, Miles Wang, Olivia Watkins, Kevin Weil, Amy Wendling, Kevin Whinnery, Cedric Whitney, Hannah Wong, Lin Yang, Yu Yang, Michihiro Yasunaga, Kristen Ying, Wojciech Zaremba, Wenting Zhan, Cyril Zhang, Brian Zhang, Eddie Zhang, and Shengjia Zhao. gpt-oss-120b and gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Wenbo Pan, Zhichao Liu, Qiguang Chen, Xiangyang Zhou, Yu Haining, and Xiaohua Jia. The hidden dimensions of LLM alignment: A multi-dimensional analysis of orthogonal safety directions. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=wGFEzfhFae.

Fábio Perez and Ian Ribeiro. Ignore previous prompt: Attack techniques for language models. In *NeurIPS ML Safety Workshop*, 2022. URL https://openreview.net/forum?id=qiaRo_7Zmug.

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, 2025.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, et al. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. *arXiv preprint arXiv:2505.06708*, 2025.

Enrique Queipo-de Llano, Álvaro Arroyo, Federico Barbero, Xiaowen Dong, Michael Bronstein, Yann LeCun, and Ravid Shwartz-Ziv. Attention sinks and compression valleys in llms are two sides of the same coin. *arXiv preprint arXiv:2510.06477*, 2025.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Bingqi Shang, Yiwei Chen, Yihua Zhang, Bingquan Shen, and Sijia Liu. Forgetting to forget: Attention sink as a gateway for backdooring llm unlearning. *arXiv preprint arXiv:2510.17021*, 2025.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, et al. A strongreject for empty jailbreaks. *Advances in Neural Information Processing Systems*, 37:125416–125440, 2024.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*, 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2025. URL https://openreview.net/forum?id=2XBPdPIcFK.

Eric Wallace, Tony Zhao, Shi Feng, and Sameer Singh. Concealed data poisoning attacks on NLP models. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 139–150, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.13. URL https://aclanthology.org/2021.naacl-main.13/.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. Poisoning language models during instruction tuning. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Yining Wang, Mi Zhang, Junjie Sun, Chenyue Wang, Min Yang, Hui Xue, Jialing Tao, Ranjie Duan, and Jiexi Liu. Mirage in the eyes: Hallucination attack on multi-modal large language models with only attention sink. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 3707–3726, 2025.

Zijun Wang, Haoqin Tu, Jieru Mei, Bingchen Zhao, Yisen Wang, and Cihang Xie. Attngcg: Enhancing jailbreaking attacks on llms with attention manipulation. *arXiv preprint arXiv:2410.09040*, 2024.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=VOstHxDdsN.

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. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=80IwJqlXs8.

Jeffrey TH Wong, Cheng Zhang, Louis Mahon, Wayne Luk, Anton Isopoussu, and Yiren Zhao. On the existence and behavior of secondary attention sinks. *arXiv preprint arXiv:2512.22213*, 2025.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, 4(2):100211, 2024.

Itay Yona, Ilia Shumailov, Jamie Hayes, Federico Barbero, and Yossi Gandelsman. Interpreting the repeated token phenomenon in large language models. *arXiv preprint arXiv:2503.08908*, 2025.

Jiahao Yu, Haozheng Luo, Jerry Yao-Chieh Hu, Yan Chen, Wenbo Guo, Han Liu, and Xinyu Xing. Mind the inconspicuous: Revealing the hidden weakness in aligned {LLMs}'refusal boundaries. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 259–278, 2025.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. *arXiv preprint arXiv:2406.15765*, 2024.

Pedram Zaree, Md Abdullah Al Mamun, Quazi Mishkatul Alam, Yue Dong, Ihsen Alouani, and Nael Abu-Ghazaleh. Attention eclipse: Manipulating attention to bypass llm safety-alignment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 16659–16679, 2025.

Stephen Zhang, Mustafa Khan, and Vardan Papyan. Attention sinks: A 'catch, tag, release' mechanism for embeddings. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=r8UWp9JeJi.

Chujie Zheng, Fan Yin, Hao Zhou, Fandong Meng, Jie Zhou, Kai-Wei Chang, Minlie Huang, and Nanyun Peng. On prompt-driven safeguarding for large language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

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, 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.

Zayd M. K. Zuhri, Erland Hilman Fuadi, and Alham Fikri Aji. Softpick: No attention sink, no massive activations with rectified softmax, 2026. URL https://arxiv.org/abs/2504.20966.

A Compute Resources

All experiments were run on NVIDIA L40S 48 GB GPUs, each requiring a single GPU.

B Response Length for Sink Measurement

The sink detection criterion (Equation 4) is evaluated over the suffix and the 150 tokens that follow it. Because greedy decoding sometimes produces shorter responses, we use two generation modes:

- **Standard inference**: used as the no-intervention baseline for all reported ASRs.
- **Standard inference with `min_new_tokens=150`**: used whenever responses are required for sink measurement.

C Suffix Control Details

This appendix gives the construction details for the three suffix controls introduced in Section 2.3. All three are matched to the attack suffixes in length, so that any difference in measured sinks cannot be attributed to suffix length.

Fluent control The fluent suffixes are grammatical English instructions that direct the *form* of a response (e.g., structure, tone, or level of detail) without requesting anything harmful. We generated a set of candidates synthetically with Claude and kept the 20 whose token count falls in the 19–21 range under all four model tokenizers, matching the 20-token attack suffixes. From those 20 we drew 10 as a uniform random sample. Table 3 lists them in full.

Table 3: The 10 fluent control suffixes. The same set is used for every model.

Aim for a comprehensive answer rather than a brief one, and be specific wherever you reasonably can.
Make sure the response is well organized, reasonably detailed, and easy to follow from start to finish.
Be as clear and unambiguous as you can, and spell out anything that might otherwise be misread.
Keep the writing clear and precise, and please define any specialized terminology at the point you use it.
Please be thorough and well-organized, and explain each step clearly as you work through it.
State each point once and do not restate it later in different words further down the response.
End with a brief recap of the main points, restated in slightly different words than before.
Include a concrete example for each of the main points you raise, so the ideas stay grounded.
Provide a complete and well-reasoned response, and make the structure of the argument visible.
Lay this out step by step in a numbered sequence, keeping each individual step short and clear.

Permuted control Each permuted suffix is obtained by randomly permuting the token IDs of the corresponding attack suffix and decoding the result back to text. This holds the token multiset exactly fixed and varies only the ordering, so the comparison isolates the contribution of the optimized token *sequence* rather than the choice of tokens. It also keeps the suffix on the model's natural token distribution, which distinguishes it from the random control below. One permuted suffix is generated per attack suffix, giving 10 per (model, attack).

Random control Random suffixes are sampled uniformly from the printable-ASCII subset of each model's vocabulary. Because a decoded token sequence does not always re-encode to the same length, we accept a sample only if its decode-then-re-encode round trip lands in the same 19–21 token band used for the fluent suffixes.

D Per-Model Suffix Sink Counts

Figure 8 reports the average number of suffix sinks per head for each model and suffix type at $\epsilon = 0.025$. Gemma-2-9B and Qwen-2.5-7B have more sinks per head than the two Llama models in every condition, but within each model, both attacks induce more sinks than the permuted and random controls.

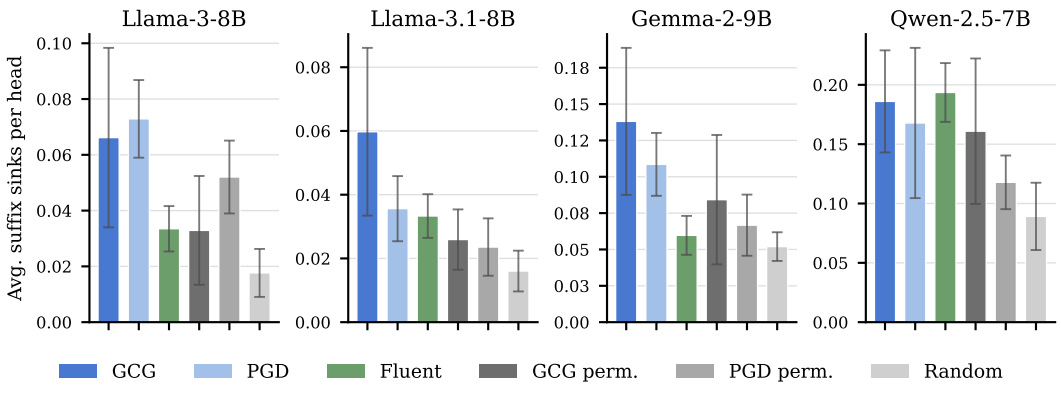

Figure 8: **Suffix sinks per head for each model.** Average number of suffix sinks per head for each suffix type at $\epsilon = 0.025$. Error bars show one standard deviation across the 10 suffixes of each type.

The fluent control falls below both attacks on the Llama and Gemma models but matches them on QWEN-2.5-7B, which is also the model most susceptible to the controls: its fluent suffixes alone reach 20.0% ASR, compared with at most 5.9% on the other models (Table 2). Correspondingly, it shows the smallest gap between attacks and controls in both ASR and sinks: on average, its control suffixes retain 26% of the attacks' ASR and 79% of their sinks per head, compared with 3–13% and 49–53% on the other models.

E THRESHOLD SWEEP FOR SINK MODULATION EXPERIMENTS

For the threshold sweeps reported in Figures 5 and 6, we evaluate the sink threshold ϵ over the grid

$$\epsilon \in \{0.005,\ 0.0075,\ 0.01,\ 0.0125,\ 0.015,\ 0.0175,\ 0.02,\ 0.0225,\ 0.025\}.$$

Each value of ϵ corresponds to one point on the horizontal axis of these figures.

F PER-LAYER SINK DETECTION

Applying the intervention at one layer may alter attention patterns in subsequent layers, potentially introducing or removing sinks downstream. We therefore determine $\mathcal{S}$ progressively: starting from the first layer, we detect sinks via Equation 4 on a baseline response generated at $\lambda = 1$, apply the intervention at that layer, and proceed to the next. This sweep yields a sink set for every head and layer that is held fixed during evaluation. Heads with $\mathcal{S} = \emptyset$ pass through unchanged.

G SINK-SUBSPACE MODULATION CONTROL EXPERIMENT

To verify that the ASR shifts reported in Section 4 are specific to the identified sink positions, we run a budget-matched position control. For each attention head with sink set $\mathcal{S}_h$, we sample $|\mathcal{S}_h|$ suffix positions uniformly at random and apply the identical intervention (Equation 7) to these positions instead. The per-head count of modulated positions is preserved exactly by construction, so the control isolates the effect of *which* positions are modulated from the effect of *how many*.

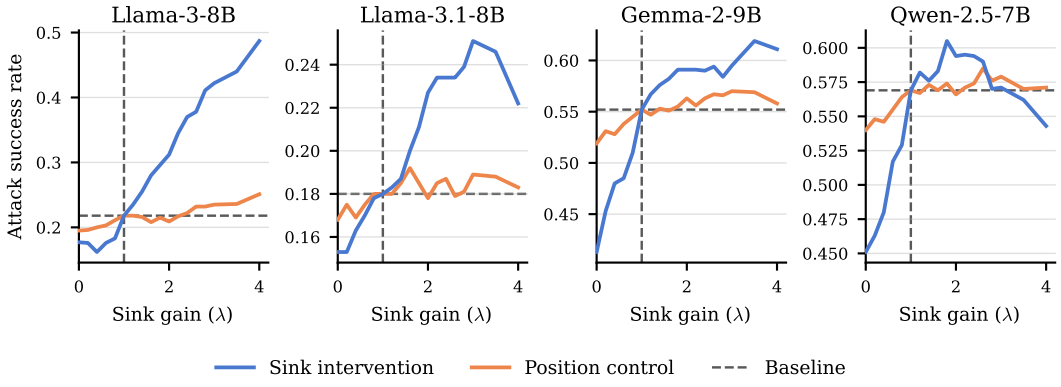

Figure 9: **The sink intervention outperforms a budget-matched position control.** Each panel plots ASR against sink gain λ under GCG suffixes at $\epsilon = 0.025$. The position control modulates the same number of suffix positions per head, sampled uniformly at random. Dashed lines mark the no-intervention baseline ($\lambda = 1$).

Figure 9 compares the two interventions under GCG suffixes at $\epsilon = 0.025$, sweeping λ over $[0, 4]$. On all four models, the real intervention produces substantially larger ASR shifts than the position control in both the amplification ($\lambda > 1$) and attenuation ($\lambda < 1$) regimes. On LLAMA-3-8B, the clearest case, amplification reaches $+123\%$ above baseline versus the control's $+15\%$.

The position control is not perfectly flat: it drifts upward with λ on every model, consistent with random suffix positions carrying a small but nonzero sink-like signal. Nevertheless, the real interven-

tion produces effects 4–8× larger, confirming that the identified sink positions carry a qualitatively distinct contribution to attack success.

H Refusal Suppression Across Sink Gains

Figure 7 contrasts refusal alignment at a single amplified gain ($\lambda = 10$) with the no-intervention baseline. Here we show the full sweep over λ: Figure 10 for the first response token and Figure 11 averaged over all response tokens. In both, refusal alignment falls further below the baseline as λ increases.

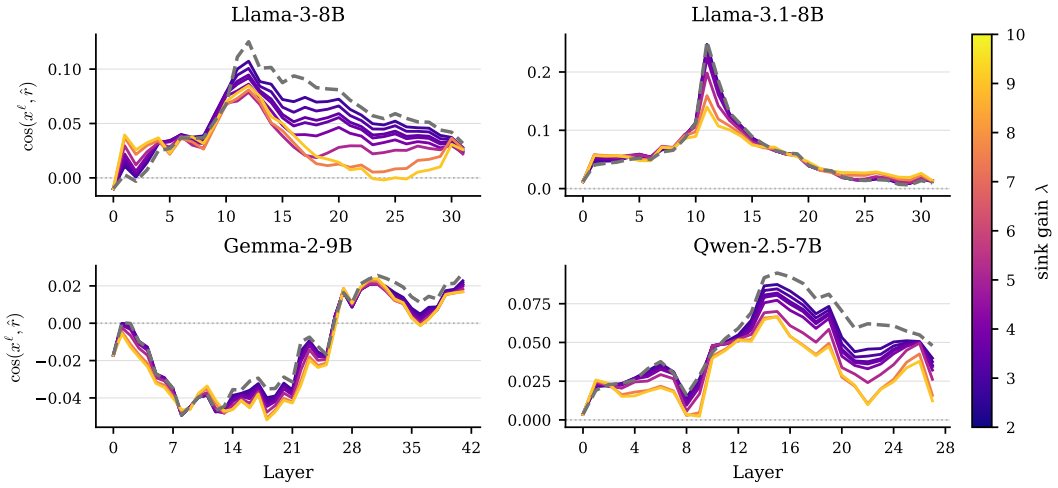

Figure 10: **Refusal suppression is graded in the sink gain (first response token).** Cosine similarity between the first response token's residual stream and the refusal direction $\hat{\mathbf{r}}$ at each layer, over the full range of sink gains λ (colorbar). The grey dashed curve is the no-intervention baseline ($\lambda = 1$).

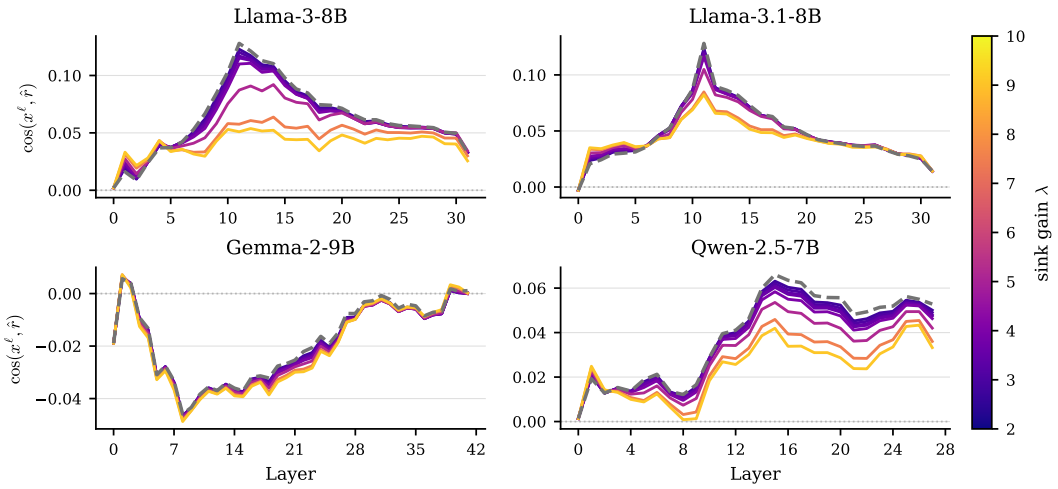

Figure 11: **Refusal suppression is graded in the sink gain (averaged over response tokens).** Same as Figure 10, but the cosine similarity is averaged over all response tokens rather than taken at the first token.

I COMPARISON WITH ATTNGCG

AttnGCG (Wang et al., 2024) adds a term to the GCG objective that rewards attention to the adversarial suffix. Since both AttnGCG and our work involve attention to adversarial suffixes, the two can appear to study the same phenomenon. We show that they do not, in two ways.

First, the two measure different quantities. AttnGCG's suffix attention score $s_{I_{\text{adv}}}$ averages final-layer attention over query positions, heads, and suffix positions, whereas we count, in each head, the suffix positions that are attention sinks (Equation 4). Computed on the same 60 suffixes per model (the six conditions of Figure 3), sinks per head correlates significantly with ASR on all four models, while $s_{I_{\text{adv}}}$ does so only on QWEN-2.5-7B (Table 4).

Table 4: Pearson correlation with ASR of sinks per head ($\epsilon = 0.025$) and of AttnGCG's suffix attention score $s_{I_{\text{adv}}}$, on the same 60 suffixes per model.

Model	r (sinks per head)	p	r ($s_{I_{\text{adv}}}$)	p
LLAMA-3-8B	0.63	5.3×10^{-8}	-0.03	0.82
LLAMA-3.1-8B	0.36	0.004	0.07	0.58
GEMMA-2-9B	0.64	4.6×10^{-8}	0.12	0.38
QWEN-2.5-7B	0.45	3.0×10^{-4}	0.70	6.5×10^{-10}

Second, amplifying suffix sinks is not the same as optimizing for suffix attention. We ran AttnGCG under the same setup as GCG (Section 2.3) on the two Llama models. AttnGCG does not improve on GCG, whereas amplifying the sink subspace on the same GCG suffixes raises ASR by 176% and 76% (Table 5). The amplified values are the peaks in Figure 5, taken over a sweep of λ and ϵ, so we offer this as evidence that the two operations differ rather than as an attack benchmark.

Table 5: Average ASR (%) on JailbreakBench of the top-10 GCG and AttnGCG suffixes, and of the GCG suffixes with sink amplification (peak values from Figure 5).

Attack	LLAMA-3-8B	LLAMA-3.1-8B
GCG	$21.8_{\pm 2.7}$	$18.0_{\pm 5.1}$
AttnGCG	$21.2_{\pm 5.6}$	$16.6_{\pm 4.6}$
GCG + sink amplification	$60.1_{\pm 12.7}$	$31.6_{\pm 19.1}$

These gains come from modulating very few positions: at the peaks in Table 5, the intervention acts on only 0.38 and 0.28 suffix sinks per head on average. The position control in Appendix G shows that these particular positions matter, since modulating an equal number of randomly chosen suffix positions per head has a much smaller effect. Raising attention to the suffix as a whole, as AttnGCG does, is therefore not what drives attack success; the contribution of the specific positions that act as sinks is, and Section 5 links this contribution to suppression of the refusal direction.