Enhancing the Transferability of Jailbreak Attacks on Large Language Models via Exploiting Reparameterization Invariance

Ao Wang[1], Xinghao Yang[1], Yongshun Gong[2], Wei Liu[3], Baodi Liu[1], Weifeng Liu[1]*,

[1]China University of Petroleum (East China), [2]Shandong University, [3]University of Technology Sydney,
wanga@s.upc.edu.cn, yangxh@upc.edu.cn
ysgong@sdu.edu.cn, wei.liu@uts.edu.au, thu.liubaodi@gmail.com, liuwf@upc.edu.cn

Abstract

Jailbreak attacks serve as a pivotal technique for evaluating the safety alignment of Large language models. Current token-level attacks have shown remarkable efficacy on open-source models by leveraging gradient-based optimization. However, these attacks suffer from poor cross-model transferability, severely limiting their utility on proprietary ones. To address this limitation, we propose Reparameterization Invariance Gradient-based Jailbreak (RIGJ), a natural gradient based framework designed to improve cross-model transferability. Unlike prior token-level methods whose optimization paths are constrained by model-specific Euclidean geometry, RIGJ defines update directions according to differences in output distributions rather than parameter-space distances. Since language models are trained to capture similar dependency structures of natural language, their output distributions share common geometry across architectures, yielding intrinsically model-agnostic optimization trajectories and substantially stronger jailbreak transferability. Extensive experiments demonstrate superior performance, increasing the cross-model Attack Success Rate and Average Harmfulness Score by 14.9% and 1.23, respectively. Our code is provided in https://github.com/nohuma/AISafety_transfer_jailbreak_RIGJ_2026.

1 Introduction

The widespread deployment of Large Language Models (LLMs) in real-world applications, such as healthcare (Wang et al., 2023), financial analysis (Zhao et al., 2024) and industry (Raza et al., 2025), makes safety alignment a critical concern. Despite extensive alignment efforts such as RLHF (Ouyang et al., 2022), LLMs remain vulnerable to jailbreak attacks, in which carefully crafted adversarial inputs bypass safety constraints and elicit unsafe or policy-violating outputs. Therefore, studying jailbreak attacks is essential to expose fundamental weaknesses in alignment mechanisms and strengthen the robustness of LLMs safety systems.

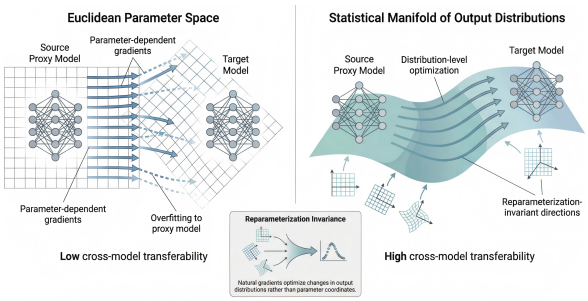

Figure 1: Information-geometric perspective on jailbreak transferability. Conventional Euclidean optimization (left) produces model-specific gradients that overfit proxy models, resulting in low cross-model transfer. In contrast, optimization on the statistical manifold of output distributions via natural gradients (right) is reparameterization-invariant and reveals shared adversarial directions across models.

Existing jailbreak attacks can be broadly categorized into three levels: Model-level, Prompt-level, and Token-level. Firstly, model-level attacks directly modify parameters (Lermen et al., 2023) or manipulate the decoding process (Huang et al., 2024; Qi et al., 2024) of LLMs, offering the strongest theoretical attack capability. However, such attacks require full access to victim model, making them impractical in real-world settings. Secondly, prompt-level attacks leverage the semantic flexibility of natural language, crafting manipulative prompts that exploit linguistic ambiguities (Chao et al., 2023; Mehrotra et al., 2024; Liu et al., 2025a, 2024b) or social-engineering patterns (Jiang et al., 2024; Liu et al., 2025b; Zeng et al., 2024). Although carefully designed prompts exhibit reasonable transferability via shared semantics, their loosely guided, heuristic optimization leads to high computational overhead and the need

*Corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 7854–7865
July 2-7, 2026

for expert knowledge. Thirdly, token-level attacks (Zou et al., 2023; Jia et al., 2025; Hu et al., 2024; Guo et al., 2024) leverage white-box access to a proxy model and perform gradient-based optimization to insert adversarial prefixes or suffixes. This enables fine-grained control and high attack success on the source model, but often suffers from limited cross-model transferability.

In this paper, we argue that this limitation is fundamentally rooted in the geometric assumptions underlying gradient-based token optimization. These methods operate in the embedding space by following Euclidean gradients computed on a single proxy model, implicitly assuming that steep descent directions correspond to universally effective attack directions. However, Euclidean geometry is inherently tied to a specific parameterization, as its notions of distance and curvature depend on the proxy model (Dinh et al., 2017). Constraining optimization paths to model-dependent geometry is the major reason that causes adversarial perturbations to exploit non-transferable vulnerabilities.

To break this dependency, we revisit token-level jailbreak attacks from an information-geometric perspective, exploiting the reparameterization invariance of natural gradient descent (Martens, 2020) as shown in Figure 1. We construct descent directions based on the KL-divergence between output distributions, operating on the intrinsic statistical manifold shared by language models. Since different LLMs are trained to model the same conditional distributions of natural language, their output spaces exhibit a common geometric structure despite disparate internal parameterizations. Optimizing along this shared output manifold yields model-agnostic updates with strong transferability.

To operationalize this insight, we address several practical challenges. First, inverting the full Fisher information matrix is infeasible for LLMs due to its extremely high dimensionality. We adopt a diagonal Fisher approximation, which preserves essential curvature information while rendering natural-gradient updates tractable. Second, continuous adversarial optimization suffers from severe objective mismatch when decoded into discrete tokens. We mitigate this issue using a Gumbel–Softmax relaxation with progressive temperature annealing, effectively narrowing the continuous–discrete gap. Third, fixed target-matching objectives induce brittle optimization and fail to capture moderation decision boundaries. We instead employ a data-anchored objective based on harmful–benign sample pairs to provide a stable and behavior-aligned training signal. Finally, multi-sample joint optimization further improves generalization and efficiency. Extensive experiments demonstrate that our method significantly outperforms other attacks in cross-model transferability, robustness against defenses, and computational efficiency.

- We provide an information-geometric interpretation of jailbreak transferability, identifying reparameterization invariance as a key condition for cross-model robustness.
- We propose a transferable token-level attack framework that exploits the reparameterization invariance of natural gradients, yielding intrinsically aligned adversarial directions with strong cross-model generalization.
- Extensive experiments show superior performance in cross-model jailbreak attacks with lower computational consumption.

2 Related Works

2.1 Jailbreak Attacks

Recent jailbreak attacks can be categorized into three primary levels: Model-level, Prompt-level and Token-level. Model-level attacks intervene directly in the model's internal mechanisms, such as adopting adversarial fine-tuning to compromise alignment (Qi et al., 2024) or manipulating decoding strategies to elicit harmful responses (Huang et al., 2024; Zhao et al., 2025). However, their requirement for white-box access precludes their application to proprietary LLMs. Prompt-level attacks circumvent safety guardrails via automated prompt engineering, leveraging auxiliary models for iterative refinement (Chao et al., 2023; Liu et al., 2025a) or exploiting human-derived heuristics, such as ASCII art (Jiang et al., 2024), input reversal (Liu et al., 2025b), or persuasive strategies (Zeng et al., 2024). However, these methods are often constrained by high computational overhead and heavy reliance on human-crafted priors.

Token-level attacks disrupt model safety by constructing adversarial perturbations composed of several optimized tokens that are appended as either prefixes or suffixes to malicious queries. These sequences are typically optimized via continuous (Guo et al., 2024; Hu et al., 2024) or discrete methods (Zou et al., 2023; Jia et al., 2025) to maximize the probability of predefined affirmative responses

(e.g., "Sure, here is...") on surrogate models. Despite theoretical transferability, they exhibit poor generalization by relying on distorted Euclidean embedding metrics, ignoring the output probability space where models share robust adversarial directions due to similar topological structures.

2.2 Natural Gradient Descent

Natural Gradient Descent (NGD) formulates optimization on a Riemannian manifold induced by the Fisher Information Matrix (FIM) instead of a Euclidean parameter space (Amari, 1998). By preconditioning gradients with the inverse FIM, NGD measures progress via the induced Kullback–Leibler (KL) divergence between model distributions, naturally accounting for the local geometry of the statistical manifold. This leads to reparameterization-invariant update directions (Martens, 2020), distinguishing NGD from standard gradient descent. Such properties have made NGD a foundational tool in deep learning and reinforcement learning, including policy optimization methods such as TRPO (Schulman et al., 2015).

3 Method

3.1 Problem Definition

Token-level jailbreak attacks a language model f_θ by appending an optimized length-L adversarial token sequence $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_L)$ to a malicious query x, forming a jailbreak prompt $x' = x \oplus \boldsymbol{\delta}$, where $\oplus$ denotes token concatenation and each δ_i is sampled from a vocabulary $\mathcal{V}$ with $|\mathcal{V}| = V$. Such attacks typically leverage gradient information from a surrogate model to optimize $\boldsymbol{\delta}$ by maximizing the likelihood under f_θ of a predefined target sequence y_{target} (e.g., "`Sure, here is`"),

$$\begin{aligned} &\max_{\boldsymbol{\delta} \in \mathcal{V}^L} \log p_\theta(y_{\text{target}} \mid x \oplus \boldsymbol{\delta}) \\ \Leftrightarrow &\min_{\boldsymbol{\delta} \in \mathcal{V}^L} \mathcal{L}(\boldsymbol{\delta}; \theta) = -\log p_\theta(y_{\text{target}} \mid x \oplus \boldsymbol{\delta}). \end{aligned} \quad (1)$$

3.2 A Reparameterization Perspective on Jailbreak Transferability

To solve Eq. (1), current methods employ continuous gradient optimization by relaxing the discrete suffix $\boldsymbol{\delta}$ into a differentiable embedding $\mathbf{e} \in \mathbb{R}^{L \times V}$. This allows the surrogate model f_{θ_S} to be optimized via Euclidean Gradient Descent (EGD):

$$\mathbf{e}_{t+1} = \mathbf{e}_t - \eta \nabla_{\mathbf{e}} \mathcal{L}(\mathbf{e}; \theta_S). \quad (2)$$

Importantly, the Euclidean gradient update can be equivalently characterized as the solution to the following local constrained optimization problem:

$$\arg\min_{\Delta \mathbf{e}} \nabla_{\mathbf{e}} \mathcal{L}^\top \Delta \mathbf{e} \quad \text{s.t.} \quad \|\Delta \mathbf{e}\|_2^2 \leq \epsilon, \quad (3)$$

which defines the steepest descent direction with respect to the Euclidean metric on the embedding space. For a target model f_{θ_T}, we abstract the relationship between the surrogate and target embedding spaces as a smooth reparameterization $\mathbf{e}' = \phi(\mathbf{e})$ with Jacobian $\mathbf{J}_\phi = \partial \phi / \partial \mathbf{e}$. Under this transformation, a perturbation $\Delta \mathbf{e}$ induces

$$\Delta \mathbf{e}' = \mathbf{J}_\phi \Delta \mathbf{e}. \quad (4)$$

Steepest descent is reparameterization-invariant if and only if the constraint set in Eq. (3) is preserved. For the Euclidean norm, the induced perturbation $\Delta \mathbf{e}'$ must satisfy:

$$\|\Delta \mathbf{e}'\|_2^2 = \Delta \mathbf{e}^\top (\mathbf{J}_\phi^\top \mathbf{J}_\phi) \Delta \mathbf{e} = \|\Delta \mathbf{e}\|_2^2, \quad (5)$$

which necessitates $\mathbf{J}_\phi^\top \mathbf{J}_\phi = \mathbf{I}$, i.e., the mapping ϕ must be an isometry. However, disparate LLM architectures induce highly non-linear and anisotropic reparameterizations that violate this condition. Consequently, EGD-based attacks optimize directions tied to the surrogate's specific coordinate system, leading to poor transferability.

3.3 Reparameterization-Invariant Optimization for Jailbreak Attacks

To address these limitations, we shift from model-dependent Euclidean updates to an information-geometric approach. By endowing the embedding space with a Riemannian metric derived from the output distribution, this paradigm aligns updates with the model's intrinsic geometry, ensuring invariance to specific parameterizations.

3.3.1 Reparameterization Invariance of Natural Gradient

Instead of defining the steepest descent direction using a Euclidean norm constraint, we measure the size of a perturbation $\Delta \mathbf{e}$ by the change it induced in the model's output distribution. Specifically, we define the local descent direction as the solution to

$$\begin{aligned} &\arg\min_{\Delta \mathbf{e}} \nabla_{\mathbf{e}} \mathcal{L}^\top \Delta \mathbf{e} \\ &\text{s.t.} \quad \mathrm{KL}[p_{\theta_S}(y \mid \mathbf{e}) \,\|\, p_{\theta_S}(y \mid \mathbf{e} + \Delta \mathbf{e})] \leq \epsilon. \end{aligned} \quad (6)$$

The Kullback–Leibler (KL) divergence measures the change in the output distribution p_{θ_S} induced

by $\Delta\mathbf{e}$. For small perturbations, the KL divergence admits a local quadratic approximation where the Fisher Information Matrix (FIM) serves as a Riemannian metric, quantifying the output's sensitivity to changes in the embedding space,

$$\mathrm{KL}[p_{\theta_S}(y \mid \mathbf{e}) \,\|\, p_{\theta_S}(y \mid \mathbf{e} + \Delta\mathbf{e})] \approx \frac{1}{2}\Delta\mathbf{e}^\top \mathbf{F}(\mathbf{e})\Delta\mathbf{e}, \quad (7)$$

where FIM is given by

$$\mathbf{F}(\mathbf{e}) = \mathbb{E}_{y\sim p_{\theta_S}}[\nabla_{\mathbf{e}} \log p_{\theta_S}(y \mid \mathbf{e}) \cdot \nabla_{\mathbf{e}} \log p_{\theta_S}(y \mid \mathbf{e})^\top]. \quad (8)$$

Substituting this quadratic form into Eq. (6) reformulates the descent direction under the Fisher metric. Solving via Lagrange multipliers yields the natural gradient $\tilde{\nabla}_{\mathbf{e}} = \mathbf{F}^{-1}\nabla_{\mathbf{e}}\mathcal{L}$. For a smooth reparameterization $\mathbf{e}' = \phi(\mathbf{e})$ with Jacobian $\mathbf{J}_\phi$, the natural gradient transforms as:

$$\begin{aligned}\tilde{\nabla}_{\mathbf{e}'} &= (\mathbf{F}')^{-1}\nabla_{\mathbf{e}'}\mathcal{L} \\ &= \left(\mathbf{J}_\phi^{-\top}\mathbf{F}\mathbf{J}_\phi^{-1}\right)^{-1}\left(\mathbf{J}_\phi^{-\top}\nabla_{\mathbf{e}}\mathcal{L}\right) \\ &= \mathbf{J}_\phi\mathbf{F}^{-1}\nabla_{\mathbf{e}}\mathcal{L} \\ &= \mathbf{J}_\phi\tilde{\nabla}_{\mathbf{e}},\end{aligned} \quad (9)$$

confirming that the natural gradient transforms *contravariantly*. Consequently, the update step in the transformed space is the *push-forward* of the original update (i.e., $\Delta\mathbf{e}' = \mathbf{J}_\phi\Delta\mathbf{e}$), ensuring the optimization trajectory remains invariant on the underlying statistical manifold. By targeting the semantic geometry shared across LLMs rather than overfitting to model-specific artifacts, NGD effectively isolates transferable vulnerabilities.

3.3.2 Implementation of Reparameterization Invariant Gradient-based Attack

To operationalize this framework, we optimize a latent semantic objective using an stable diagonal natural gradient, implemented via a Gumbel-Softmax relaxation over token logits. Algorithm 1 summarizes the end-to-end training procedure. Next, we will focus on discussing the problems encountered during the implementation of the algorithm.

Gumbel-Softmax Relaxation. Optimizing adversarial perturbations in the continuous embedding space inevitably encounters the discrete-continuous gap, where the objective $\mathcal{L}$ diverges significantly upon quantizing $\mathbf{e}$ back to discrete tokens $\boldsymbol{\delta} \in \mathcal{V}^L$.

Algorithm 1 The proposed RIGJ Method

Require: Surrogate f_{θ_S}, query x, latent classifier g_ϕ, surrogate vocabulary $\mathcal{V}$, learning rate η, decay rates β_1, β_2, initial and final temp τ_{start}, τ_{end}, maximum number of iterations T.
Ensure: Adversarial prefix $\boldsymbol{\delta}^*$.

Initialize $\mathbf{\Pi} \in \mathbb{R}^{L\times V}$, $\mathbf{m}_0 \leftarrow \mathbf{0}$, $\mathbf{v}_0 \leftarrow \mathbf{0}$.
for $t = 1$ **to** T **do**
  /* Gumbel-Softmax Relaxation */
  Sample Gumbel noise $G \sim \mathrm{Gumbel}(0, 1)$.
  $\mathbf{e}_t = \mathrm{softmax}((\mathbf{\Pi} + G)/\tau_t) \cdot \mathbf{W}_{emb}$.
  /* Sample-Anchored Forward Pass */
  $\mathbf{H} = f_{\theta_S}(\mathbf{e}_t \oplus x)$
  $\mathcal{L} = \mathrm{BCE}(g_\phi(\mathrm{PCA}(\mathbf{H})), 1)$
  /* Diagonal Natural Gradient Descent */
  $\mathbf{g}_t = \nabla_{\mathbf{\Pi}}\mathcal{L}$
  $\hat{\mathbf{F}}_t = (\nabla_{\mathbf{\Pi}} \log p_{\theta_S}(y_{\text{target}} \mid \mathbf{e}_t))^2$
  $\mathbf{m}_t = \beta_1\mathbf{m}_{t-1} + (1-\beta_1)\mathbf{g}_t$
  $\mathbf{v}_t = \beta_2\mathbf{v}_{t-1} + (1-\beta_2)\hat{\mathbf{F}}_t$
  $\tilde{\nabla}_{\mathbf{\Pi}} = \mathbf{m}_t/(\mathbf{v}_t + \epsilon)$
  /* Update & Temperature Annealing */
  $\mathbf{\Pi} \leftarrow \mathbf{\Pi} - \eta\tilde{\nabla}_{\mathbf{\Pi}}$
  $\tau_t = \tau_{start} - t \cdot \frac{\tau_{start}-\tau_{end}}{T}$
end for
/* Decoding Discrete Sequences */
$\boldsymbol{\delta}^* = \arg\max_{v\in\mathcal{V}} \mathbf{\Pi}$
return Optimized adversarial prefix $\boldsymbol{\delta}^*$

To mitigate this gap, we employ the Gumbel Softmax relaxation to construct a differentiable approximation of the discrete sampling process. The embedding $\mathbf{e}_{t,i}$ for the i-th token is:

$$\mathbf{e}_{t,i} = \sum_{j=1}^{|\mathcal{V}|} \frac{\exp((\log\pi_{i,j} + g_{i,j})/\tau)}{\sum_{l=1}^{|\mathcal{V}|}\exp((\log\pi_{i,l} + g_{i,l})/\tau)}\mathbf{w}_j, \quad (10)$$

where $g \sim \mathrm{Gumbel}(0, 1)$ denotes the i.i.d. Gumbel noise and $\mathbf{w}_j$ is the j-th entry of the embedding matrix $\mathbf{W}_{emb}$. Temperature τ annealing smoothly hardens variable $\mathbf{\Pi}$ into a one-hot representation, grounding the continuous optimization in the discrete token space.

Sample-Anchored Objective. Existing attacks targeting a fixed sequence y_{target} suffer from semantic rigidity, as they overfit model-specific tokens rather than general intent. To break this token-level coupling, we optimize in the latent space following (Zheng et al., 2024). We use PCA to project hidden states $\mathbf{H} = f_{\theta_S}(\mathbf{e})$ onto a k-dimensional subspace, capturing the essential semantic features that dis-

tinguish harmful from benign content. A classifier $g_\phi : \mathbb{R}^k \to [0, 1]$ trained on this subspace provides a robust optimization objective formulated as a binary cross-entropy (BCE) loss,

$$\mathcal{L}(\mathbf{e};\theta_S) = -\big[y \log g_\phi(\text{PCA}(\mathbf{H})) + (1-y)\log(1 - g_\phi(\text{PCA}(\mathbf{H})))\big]. \tag{11}$$

This sample-anchored objective favors semantic jailbreak intent over surface-level token patterns.

Scalable Fisher Approximation. Direct inversion of the Fisher matrix is prohibitive as its size scales quadratically with the surrogate vocabulary. We adopt a diagonal Fisher approximation $\hat{\mathbf{F}}$, where each diagonal element captures the coordinate-wise sensitivity of the output distribution:

$$\hat{\mathbf{F}}_t = \mathbb{E}_{y \sim p_{\theta_S}} \left[\left(\nabla_{\mathbf{\Pi}} \log p_{\theta_S}(y \mid \mathbf{e}_t \oplus x) \right)^2 \right]. \tag{12}$$

Meanwhile, we apply Exponential Moving Average (EMA) to stabilize optimization, following the Adam (Kingma and Ba, 2015):

$$\begin{aligned} \mathbf{m}_t &= \beta_1 \mathbf{m}_{t-1} + (1-\beta_1)\mathbf{g}_t, \\ \mathbf{v}_t &= \beta_2 \mathbf{v}_{t-1} + (1-\beta_2)\hat{\mathbf{F}}_t, \end{aligned} \tag{13}$$

where $\mathbf{g}_t = \nabla_\Pi \mathcal{L}$ is the gradient with respect to $\mathbf{\Pi}$ and $\beta_1, \beta_2 \in [0, 1)$ are decay rates. The natural gradient is then computed as $\tilde{\nabla} = \mathbf{m}_t / (\mathbf{v}_t + \epsilon)$. This dual-smoothing mechanism stabilizes the local Riemannian metric estimation in a non-stationary landscape, yielding more consistent optimization trajectories and improved jailbreak transferability.

Multi-sample Joint Training and Prefix Strategy. We employ a multi-sample joint training strategy to enhance the efficiency and generalizability. Empirically, we find that adversarial suffixes often cause intent drift by shifting attention toward the perturbation itself rather than the query, thereby obscuring the attack objective. Thus, we adopt a prefix-based strategy ($\mathbf{e}_t \oplus x$) that leverages structural priority to re-prime the latent state, ensuring the query is processed within a hijacked context where harmful intent takes precedence over safety alignment.

4 Experiments

4.1 Experimental Setups

Datasets. We evaluate all methods on two representative safety benchmarks: AdvBench (Zou et al., 2023) and HarmBench (Mazeika et al., 2024). Our evaluation set includes the AdvBench Harmful Behaviors subset (520 prompts) and the HarmBench Standard subset (200 samples).

Models. Our evaluation spans several human-aligned LLMs, including open-source models (Llama2-7B-Chat (Touvron et al., 2023), Llama3-8B-Instruct (Grattafiori et al., 2024), Mistral-7B-Instruct (Jiang et al., 2023), Vicuna-7B (Chiang et al., 2023) and Qwen2-7B-Instruct (Yang et al., 2024)) and closed-source models (GPT-3.5-Turbo-0125 and GPT-4-1106-preview (OpenAI et al., 2024)). Specifically, Llama2-7B-Chat is utilized as the surrogate model for generating adversarial prompts, while all other models are evaluated as target models under a strict black-box setting.

Baselines. We compare **RIGJ** against several jailbreak attacks categorized by their transferability and optimization strategies. **PiF** (Lin et al., 2025), **DeGCG** (Liu et al., 2024a), **GJO** (Yang et al., 2025), and **AmpleGCG** (Liao and Sun, 2024) employ explicit transfer-oriented designs, such as distributional dependency analysis, two-stage learning, constraint relaxation, or universal generative modeling. Conversely, **PAIR** (Chao et al., 2023) and **AutoDAN** (Liu et al., 2024b) rely on the implicit generalization of natural language via model collaboration and genetic algorithms. Regarding granularity, **PiF**, **AutoDAN**, **PAIR**, and **AmpleGCG** are instance-specific, whereas **DeGCG**, **GJO**, and our **RIGJ** are universal attacks that optimize a single data-agnostic perturbation.

Metrics. We employ two metrics for performance assessment: (1) Attack Success Rate (ASR↑), determined by the fine-tuned `Llama-2-13B-cls` classifier from HarmBench (Mazeika et al., 2024) to identify successful jailbreaks; (2) Average Harmfulness Score (AHS↑), where `GPT-4` acts as a judge to evaluate the severity of harmful content in the generated responses. They provide both binary and qualitative measures of attack effectiveness.

Implementation Details. We set the maximum number of iterations to $T = 1000$ with a learning rate of $\eta = 1$ and decay rates $\beta_1 = 0.9, \beta_2 = 0.9999$. The temperature schedule decays from $\tau_{start} = 5$ to $\tau_{end} = 1$. The batch size is set to 20, and the adversarial prefix length is $L = 100$. For a fair comparison, the adversarial token length for DeGCG, GJO, and AmpleGCG is also fixed at 100. All other hyper-parameters for the baselines are kept consistent with their original reports. To account for response randomness, we conduct three trials and report the best results.

Datasets	Models	Methods PiF	PAIR	AutoDAN	AmpleGCG	DeGCG	GJO	RIGJ
AdvBench	Llama2	4.0 | 1.96	7.1 | 1.79	16.9 | 2.36	28.1 | 3.28	70.6 | 7.03	**91.4** | **8.91**	90.4 | 8.73
	Vicuna	4.4 | 2.15	25.8 | 3.41	54.0 | 5.93	17.5 | 2.44	24.6 | 3.14	80.6 | 8.41	**97.5** | **9.57**
	Mistral	4.2 | 2.40	23.1 | 3.50	79.4 | 8.23	56.2 | 5.73	41.9 | 4.42	76.9 | 8.30	**94.2** | **9.48**
	Qwen2	1.5 | 1.43	11.3 | 2.28	37.7 | 4.41	7.1 | 1.62	8.5 | 1.83	62.1 | 6.40	**78.9** | **7.75**
	Llama3	0.8 | **1.35**	**2.7** | 1.33	1.4 | 1.14	1.7 | 1.15	0.6 | 1.06	1.4 | 1.10	2.5 | 1.22
	GPT-3.5	2.9 | 2.20	10.0 | 2.09	27.1 | 3.68	4.6 | 1.37	35.0 | 4.19	3.7 | 1.31	**90.2** | **8.89**
	GPT-4	1.7 | 1.56	4.0 | 1.54	**22.1** | **3.22**	0.4 | 1.05	0.4 | 1.03	0.8 | 1.06	7.3 | 1.57
HarmBench	Llama2	5.0 | 1.60	25.0 | 3.36	18.5 | 2.44	35.5 | 3.69	81.5 | 8.28	**89.0** | **8.48**	84.5 | 7.87
	Vicuna	6.5 | 1.56	6.5 | 1.53	65.0 | 6.90	28.5 | 3.23	20.0 | 2.67	74.5 | 7.10	**91.5** | **8.49**
	Mistral	8.5 | 1.75	33.5 | 3.74	76.0 | 7.38	59.0 | 5.77	70.5 | 6.57	83.5 | 8.12	**84.0** | **8.27**
	Qwen2	6.5 | 1.54	24.0 | 2.90	56.5 | 5.43	21.5 | 2.56	31.5 | 3.23	78.0 | 7.33	**84.0** | **7.59**
	Llama3	2.0 | 1.38	6.5 | 1.64	2.0 | 1.22	3.0 | 1.23	4.5 | 1.31	6.0 | 1.41	**8.5** | **1.70**
	GPT-3.5	12.0 | 2.16	31.0 | 3.87	73.0 | 7.83	72.0 | 6.82	53.5 | 5.17	84.0 | 8.16	**88.0** | **8.31**
	GPT-4	2.0 | 1.29	6.0 | 1.57	**20.0** | **2.21**	6.0 | 1.32	5.5 | 1.46	6.5 | 1.35	10.5 | 1.90
Target Avg.		4.4 | 1.73	15.4 | 2.45	42.9 | 4.80	23.1 | 2.86	24.7 | 3.01	46.5 | 5.00	**61.4** | **6.23**
All Avg.		4.4 | 1.74	15.5 | 2.47	39.3 | 4.46	24.4 | 2.95	32.0 | 3.67	52.7 | 5.53	**65.1** | **6.52**

Table 1: Main transfer jailbreak results on AdvBench and HarmBench (ASR↑ | AHS↑). Source models are red; **bold**/underline mark best/second-best. Our method consistently achieves the strongest transferability, surpassing the top baseline on Target Avg. by **14.9%** in ASR and **1.23** in AHS.

Transfer Setting	Methods AutoDAN		AmpleGCG		DeGCG		GJO		RIGJ	
	ASR	AHS	ASR	AHS	ASR	AHS	ASR	AHS	ASR	AHS
AdvBench → HarmBench	18.0	2.26	19.5	2.53	56.5	6.05	82.0	8.38	**85.0**	**8.46**
AdvBench → Malicious	6.0	1.31	5.0	1.32	49.0	5.33	96.0	9.35	**98.0**	**9.43**
HarmBench → AdvBench	7.0	1.54	7.0	1.62	68.0	7.38	79.5	8.25	**81.0**	**8.39**
Harmbench → Malicious	14.0	1.90	9.0	1.81	35.0	5.25	89.0	8.58	**91.0**	**8.70**

Table 2: Cross-distribution transferability results. "A → B" denotes training on dataset A and testing on B (max. 200 samples). Universal attacks are applied directly, while instance-specific baselines utilize semantic matching.

4.2 Transfer Attack Results

Cross-model Transferability. Table 1 reports the cross-model transferability results on diverse models and datasets. Overall, our method achieves state-of-the-art performance across most targets, improving the Target Average ASR by 14.9% and AHS by 1.23 over the strongest baseline. While GJO yields marginally higher scores on the source model, RIGJ demonstrates superior transferability; this suggests that GJO may overfit specific source features, whereas our approach—benefiting from the reparameterization invariance of natural gradient optimization—captures more universal vulnerabilities. Regarding the highly robust GPT-4, we observe that prompt-level methods (e.g., AutoDAN) exhibit a slight advantage over token-level approaches, due to their semantic variations evading text-based filters. However, our method still consistently outperforms other token-level baselines on GPT-4 and achieves far superior stability across the broader spectrum of open-source models where prompt-level attacks often falter. Finally, the generally low transferability to Llama-3 likely stems from its drastic chat template discrepancy with the source model.

Cross-Distribution Transferability. To evaluate generalization to unseen harmful intents, we conduct transfer experiments across AdvBench, HarmBench, and Malicious (Huang et al., 2024), using up to 200 samples per dataset. For instance-specific baselines, we apply semantic matching based on cosine similarity to transfer prompts, while universal attacks are applied directly. PAIR and PiF are excluded due to negligible performance (near 0% ASR). Table 2 shows that universal methods significantly outperform instance-specific ones, indicating that multi-sample training captures intrinsic ad-

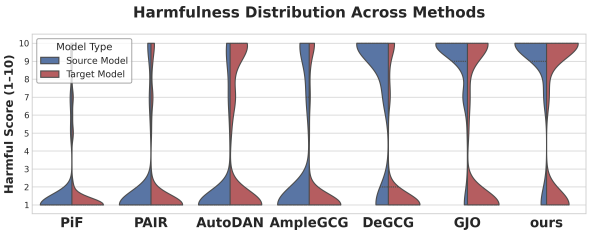

Figure 2: Harmfulness score distribution aggregated across all models and datasets. The split violin plots visualize the density of harmfulness scores for both Source (blue) and Target (red) models.

Defenses	PAIR	Auto DAN	Ample GCG	DeGCG	GJO	RIGJ
Self-Reminder	13.0	17.0	11.5	13.5	73.5	**83.5**
Paraphrase	29.0	33.5	26.5	16.0	**34.0**	32.0
Llama Guard	24.0	43.5	28.5	20.0	77.0	**80.5**
SmoothLM	35.5	42.0	20.5	12.5	67.5	**78.0**

Table 3: ASR of different attack methods against various defense mechanisms. All attacks are transferred to Vicuna and evaluated on the HarmBench dataset.

versarial patterns across various samples. Crucially, our method consistently surpasses the strongest baseline GJO, suggesting that the reparameterization invariance of natural gradients enables more robust adversarial directions under distribution shifts.

Analysis of Harmfulness Scores. Figure 2 shows the density of harmfulness scores aggregated across all models and datasets. The split violin plots compare Source (blue) and Target (red) models. Our method exhibits a markedly higher concentration of scores near the upper end, indicating stronger and more consistent attack severity than prior baselines. In contrast, weaker methods (e.g., PiF and PAIR) are dominated by low-score modes, reflecting frequent defense activation. Notably, across all methods, the score distributions are polarized, with mass concentrated near the extremes rather than the mid-range. This bimodal pattern reflects the intrinsic all-or-nothing nature of jailbreak attacks: successful cases largely bypass safety mechanisms, while failures are effectively suppressed.

4.3 Attack Effectiveness Against Defenses

We evaluate the effectiveness of different jailbreak attacks under four representative defense mechanisms: Self-Reminder (Xie et al., 2023), Paraphrase (Jain et al., 2023), LLaMA-Guard (Inan et al., 2023), and SmoothLLM (Robey et al., 2024). As shown in Table 3, existing methods experience notable performance degradation under most defenses, particularly paraphrasing and smoothing-

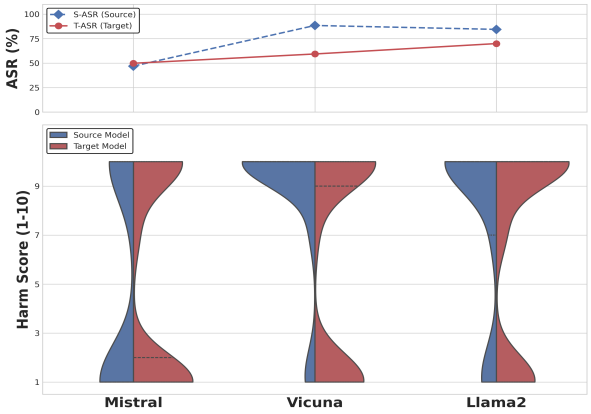

Figure 3: Ablation of source model selection on HarmBench across all evaluated target models. The top plot presents the average source and target ASR, and the bottom plot shows the harmfulness distributions when adversarial prompts are optimized on different models.

based mechanisms. In contrast, our method consistently achieves strong performance across most evaluated defenses and outperforms prior attacks in nearly all settings, demonstrating improved robustness against diverse safety mechanisms.

4.4 Ablation and In-depth Analysis

Impact of Source Model Selection. To investigate how the choice of the source model influences the transferability and harmfulness of generated jailbreak prompts, we perform an ablation study using Mistral, Vicuna, and Llama2 as surrogates. As illustrated in the Figure 3, while Vicuna achieves the highest S-ASR, Llama2 yields the most effective transferability (T-ASR) to the target model and a higher density of high harmfulness scores (9-10) in the violin plots. Conversely, Mistral results in significantly lower T-ASR and harmfulness levels, suggesting that source models with stronger reasoning capabilities and complex alignment provide more effective gradient guidance for bypassing safety filters on downstream targets.

Verification for Reparametrization Invariance of NGD. To empirically validate that NGD operates on the intrinsic manifold rather than the extrinsic Euclidean space, we visualized the consistency of optimization trajectories. We monitored the cosine similarity between the gradient updates of the source model (Llama2) and three targets: a reparameterized counterpart via parameters scaling ("S-to-S Reparam"), a homologous model (Vicuna), and a heterologous architecture (Mistral). As illustrated in Figure 4, the baseline EGD (Panel a) exhibits chaotic red-blue alternating bands, indicat-

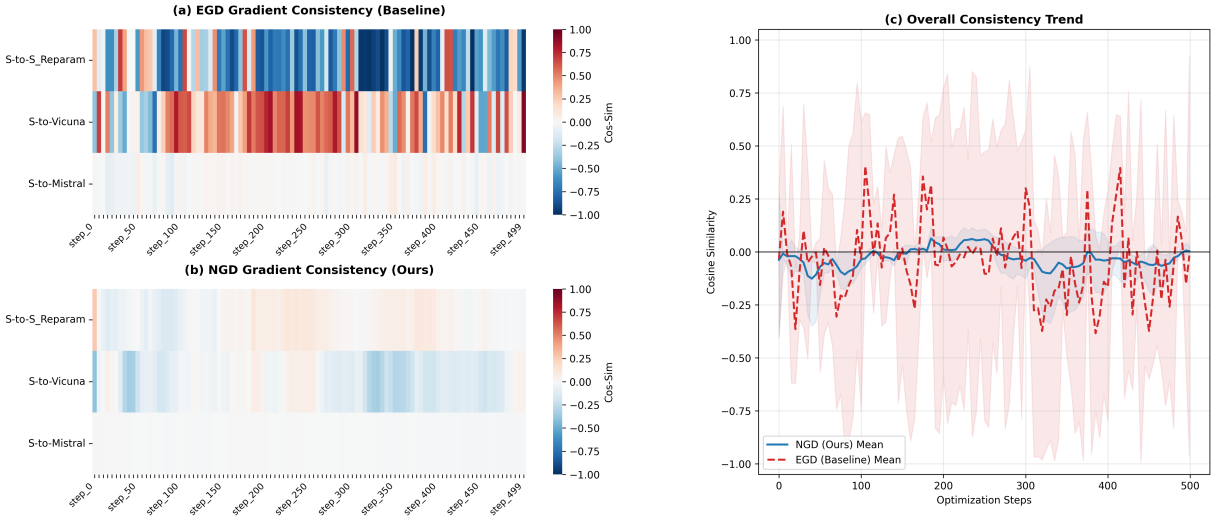

Figure 4: Gradient consistency under reparameterization for different optimization methods. Cosine similarity between gradient of the source model and target models during optimization. (a) Under standard EGD, cross-model gradient alignment is highly unstable and sensitive to reparameterization. (b) NGD produces smoother and more consistent gradient alignment. (c) Overall trend shows that NGD significantly reduces variance in cross-model cosine similarity compared to EGD, demonstrating its reparameterization-invariant optimization behavior.

Methods	Time (s)	S-EAS	T-EAS	S-EHS	T-EHS
PiF	70.1	3.5	3.8	1.67	1.48
PAIR	229.7	1.9	4.0	0.47	0.64
AutoDAN	473.3	2.1	5.4	0.30	0.61
DeGCG	124.2	34.1	11.9	3.40	1.45
GJO	89.9	61.0	31.0	5.95	3.34
RIGJ	**38.5**	**141.0**	**95.8**	**13.62**	**9.71**

Table 4: Efficiency-aware attack performance across all models and datasets. Time denotes the average attack time per sample (seconds). S-EAS / T-EAS measure source and target attack success efficiency (ASR divided by attack time in minutes), while S-EHS / T-EHS denote the corresponding harmfulness score efficiency.

ing that Euclidean gradients are hypersensitive to parameterization. In contrast, NGD (Panel b) maintains a remarkably smooth alignment. Crucially, this stability persists across architectures. While both methods naturally show higher similarity with the homologous Vicuna than the heterologous Mistral, NGD significantly mitigates the sensitivity observed in EGD. This is quantitatively confirmed by Panel (c): the EGD baseline suffers from high-amplitude oscillations, implying that its optimization direction constantly fluctuates and is unstable under reparameterization. Conversely, the NGD trajectory remains flat and stable. Overall, this experiment verifies that NGD is reparameterization-invariant, enabling stable and consistent optimization directions across models and thereby underpinning its superior cross-model transferability.

4.5 Attack Efficiency Analysis

Table 4 summarizes the efficiency evaluation. Despite learning a universal perturbation, our method achieves the lowest average runtime (38.5s), outperforming both the strongest universal baseline GJO (2.3× faster) and the lightweight single-sample method PiF. This demonstrates rapid convergence under the universal setting. In terms of efficiency-aware metrics, single-sample methods exhibit limited performance due to per-query optimization overhead (e.g., AutoDAN with an S-EAS of 2.1). By contrast, our approach fully exploits the universal paradigm, achieving an S-EAS of 141.0 and a T-EAS of 95.8, exceeding GJO by approximately 2.3× and 3.1×, respectively. This confirms that our method provides a high-transferability practical solution with minimal computational cost.

4.6 Analysis of Adversarial Token Properties

To understand why RIGJ achieves superior transferability, we analyze the structural and semantic properties of the generated prefixes. Specifically, we compare RIGJ with a Euclidean baseline, which replaces the natural gradient with standard gradient descent while keeping all other components identical. We evaluate the prefixes using four metrics: Dist-1, Structural Complexity (zlib-based compression), PPL, and Cosine Similarity. As shown in Table 5, RIGJ consistently yields higher Dist-1 and Complexity scores than the Euclidean baseline across both datasets, indicating

Metrics	Advbench		Harmbench	
	Baseline	RIGJ	Baseline	RIGJ
PPL ↓	2,467.2	48,548.0	30,385.0	60,300.0
Cosine Sim ↑	0.15	0.12	0.21	0.19
Dist-1 ↑	0.92	0.95	0.90	0.92
Complexity ↑	0.759	0.873	0.737	0.874

Table 5: Quantitative analysis of adversarial token properties across Advbench and Harmbench datasets.

that natural-gradient optimization explores a more diverse and less repetitive token space. While RIGJ exhibits higher PPL and a slight decrease in semantic consistency—a common trade-off in discrete optimization—the results suggest that superior transferability is primarily driven by the structural complexity and entropy of adversarial patterns rather than human-readable semantics. This confirms that RIGJ identifies more potent, geometrically-informed token sequences that generalize better across different model architectures.

5 Conclusion

In this work, we propose a reparameterization invariant based optimization framework to improve the transferability of token-level jailbreak attacks. We show that the limited generalization of existing attacks stems from their reliance on Euclidean gradients defined in surrogate-specific coordinates, which overfit local curvature artifacts and fail to preserve consistent descent directions across different model architectures. To address this issue, we leverage natural gradient descent to achieve coordinate independence by measuring perturbations via the KL divergence on the model output distribution manifold. Since modern LLMs are trained to approximate similar underlying language distributions, their output probability spaces exhibit substantial geometric alignment across models. This geometric invariance ensures that optimization trajectories remain consistent across models, capturing intrinsic vulnerabilities shared by models. Experimental results demonstrate that our approach significantly boosts transferability across diverse models with high computational efficiency.

6 Ethics Statement

This work investigates transferable jailbreak attacks on large language models and proposes the RIGJ framework. We acknowledge that improved transferability may lower the barrier for constructing reusable adversarial prompts, potentially enabling misuse across models. Our method assumes access to a white-box surrogate model, which partially constrains real-world applicability but does not eliminate risks. We emphasize that this study aims to expose vulnerabilities in current safety alignment mechanisms and support the development of more robust defenses, such as improved alignment training and detection of transferable attacks. We do not release ready-to-use jailbreak templates or automated attack tools.

Limitations

Despite its strong transferability and efficiency, our approach has several limitations.

- Our approach relies on the assumption of structural or distributional similarity between the surrogate and target LLMs. While this condition is generally satisfied for modern language models trained on comparable corpora, it may not hold for systems with fundamentally different output spaces, such as embodied intelligence models involving continuous control signals. However, we emphasize that this constraint is intrinsic to the transfer-based setting, where an attacker can typically select a surrogate (e.g., from the same architecture family or training lineage) to approximate the target. Extending our geometric transferability framework to fundamentally heterogeneous modalities remains an important but distinct challenge for future research.

- Our method estimates the Fisher information using finite samples, which represents a practical approximation of the ideal natural gradient and may introduce variance in extreme settings. Nevertheless, our empirical results consistently demonstrate that this approximation achieves a favorable balance between computational efficiency and attack effectiveness across diverse models and datasets.

- As a universal attack, RIGJ optimizes a shared adversarial direction across samples, which may be less effective for highly idiosyncratic or rare harmful intents that deviate significantly from the dominant training distribution.

Acknowledgments

This work is supported in part by the the National Natural Science Foundation of China (Grant

No. 62406341), in part by Major Basic Research Projects in Shandong Province, China (Grant No. ZR2023ZD32), in part by the Young Talent of Lifting engineering for Science and Technology in Shandong, China (Grant No. SDAST2024QTA040), in part by the the Shandong Natural Science Foundation (Grant No. ZR2023QF051), in part by the Outstanding Youth Science Foundation Project of Shandong Province (Overseas) (Grant No.2023HWYQ-070), and in part by the Qingdao Key Laboratory of Intelligent Sensing Technology for Extreme Environment (Grant No. 2025YB007).

References

Shun-ichi Amari. 1998. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276.

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

Chiang et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90% chatgpt quality. https://lmsys.org/blog/2023-03-30-vicuna/. Accessed: 2023-5-30.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. 2017. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1019–1028. JMLR.org.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Xingang Guo, Fangxu Yu, Huan Zhang, Lianhui Qin, and Bin Hu. 2024. Cold-attack: jailbreaking llms with stealthiness and controllability. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24, pages 16974 – 17002.

Kai Hu, Weichen Yu, Yining Li, Tianjun Yao, Xiang Li, Wenhe Liu, Lijun Yu, Zhiqiang Shen, Kai Chen, and Matt Fredrikson. 2024. Efficient llm jailbreak via adaptive dense-to-sparse constrained optimization. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Yangsibo Huang, Samyak Gupta, Mengzhou Xia, Kai Li, and Danqi Chen. 2024. Catastrophic jailbreak of open-source llms via exploiting generation. In *International Conference on Learning Representations*.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *Preprint*, arXiv:2312.06674.

Neel Jain, Avi Schwarzschild, Yuxin Wen, Gowthami Somepalli, John Kirchenbauer, Ping yeh Chiang, Micah Goldblum, Aniruddha Saha, Jonas Geiping, and Tom Goldstein. 2023. Baseline defenses for adversarial attacks against aligned language models. *Preprint*, arXiv:2309.00614.

Xiaojun Jia, Tianyu Pang, Chao Du, Yihao Huang, Jindong Gu, Yang Liu, Xiaochun Cao, and Min Lin. 2025. Improved techniques for optimization-based jailbreaking on large language models. In *International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, and et al. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Fengqing Jiang, Zhangchen Xu, Luyao Niu, Zhen Xiang, Bhaskar Ramasubramanian, Bo Li, and Radha Poovendran. 2024. ArtPrompt: ASCII art-based jailbreak attacks against aligned LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15157–15173, Bangkok, Thailand. Association for Computational Linguistics.

D. P. Kingma and J. Ba. 2015. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Simon Lermen, Charlie Rogers-Smith, and Jeffrey Ladish. 2023. Lora fine-tuning efficiently undoes safety training in llama 2-chat 70b. *Preprint*, arXiv:2310.20624.

Zeyi Liao and Huan Sun. 2024. Amplegcg: Learning a universal and transferable generative model of adversarial suffixes for jailbreaking both open and closed llms. In *Conference on Language Modeling*, COLM'24.

Runqi Lin, Bo Han, Fengwang Li, and Tongling Liu. 2025. Understanding and enhancing the transferability of jailbreaking attacks. In *International Conference on Machine Learning*.

Hongfu Liu, Yuxi Xie, Ye Wang, and Michael Shieh. 2024a. Advancing adversarial suffix transfer learning on aligned large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7213–7224, Miami, Florida, USA. Association for Computational Linguistics.

Xiaogeng Liu, Peiran Li, G. Edward Suh, Yevgeniy Vorobeychik, Zhuoqing Mao, Somesh Jha, Patrick McDaniel, Huan Sun, Bo Li, and Chaowei. 2025a. Autodan-turbo: A lifelong agent for strategy self-exploration to jailbreak llms. In *International Conference on Learning Representations*.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. 2024b. Autodan: Generating stealthy jailbreak prompts on aligned large language models. In *International Conference on Learning Representations*.

Yue Liu, Xiaoxin He, Miao Xiong, Jinlan Fu, Shumin Deng, YINGWEI MA, Jiaheng Zhang, and Bryan Hooi. 2025b. Flipattack: Jailbreak llms via flipping. In *International Conference on Machine Learning*.

James Martens. 2020. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76.

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. 2024. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 35181–35224. PMLR.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. 2024. Tree of attacks: Jailbreaking black-box llms automatically. In *Advances in Neural Information Processing Systems*, volume 37, pages 61065–61105. Curran Associates, Inc.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, and et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

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. 2022. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

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

Mubashar Raza, Zarmina Jahangir, Muhammad Bilal Riaz, Muhammad Jasim Saeed, and Muhammad Awais Sattar. 2025. Industrial applications of large language models. *Scientific Reports*, 15(1):13755.

Alexander Robey, Eric Wong, Hamed Hassani, and George J. Pappas. 2024. Smoothllm: Defending large language models against jailbreaking attacks. *Preprint*, arXiv:2310.03684.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1889–1897, Lille, France. PMLR.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, and et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. 2023. Huatuo: Tuning llama model with chinese medical knowledge. *Preprint*, arXiv:2304.06975.

Yueqi Xie, Jingwei Yi, Jiawei Shao, Justin Curl, Lingjuan Lyu, Qifeng Chen, Xing Xie, and Fangzhao Wu. 2023. Defending chatgpt against jailbreak attack via self-reminders. *Nature Machine Intelligence*, 5(12):1486–1496.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Junxiao Yang, Zhexin Zhang, Shiyao Cui, Hongning Wang, and Minlie Huang. 2025. Guiding not forcing: Enhancing the transferability of jailbreaking attacks on LLMs via removing superfluous constraints. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19643–19655, Vienna, Austria. Association for Computational Linguistics.

Yi Zeng, Hongpeng Lin, Jingwen Zhang, Diyi Yang, Ruoxi Jia, and Weiyan Shi. 2024. How johnny can persuade LLMs to jailbreak them: Rethinking persuasion to challenge AI safety by humanizing LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14322–14350, Bangkok, Thailand. Association for Computational Linguistics.

Huaqin Zhao, Zhengliang Liu, Zihao Wu, Yiwei Li, Tianze Yang, Peng Shu, Shaochen Xu, Haixing Dai, Lin Zhao, Hanqi Jiang, Yi Pan, Junhao Chen, Yifan Zhou, Gengchen Mai, Ninghao Liu, and Tianming Liu. 2024. Revolutionizing finance with llms: An overview of applications and insights. *Preprint*, arXiv:2401.11641.

Xuandong Zhao, Xianjun Yang, Tianyu Pang, Chao Du, Lei Li, Yu-Xiang Wang, and William Yang Wang. 2025. Weak-to-strong jailbreaking on large language models. In *Proceedings of the 42rd International Conference on Machine Learning*.

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

235 of *Proceedings of Machine Learning Research*, pages 61593–61613. PMLR.

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

A Derivation of the Natural Gradient

This appendix provides a derivation of the natural gradient from an information-geometric perspective. We first establish the second-order relationship between the Kullback–Leibler (KL) divergence and the Fisher Information Matrix (FIM), and then derive the natural gradient by solving a KL-constrained local optimization problem.

A.1 Second-Order Relationship Between KL Divergence and Fisher Information

Let $p_{\theta_S}(y \mid \mathbf{e})$ denote the output distribution of the surrogate model conditioned on the embedding $\mathbf{e}$. Consider a small perturbation $\Delta\mathbf{e}$ and define

$$p(y) \triangleq p_{\theta_S}(y \mid \mathbf{e}), \qquad q(y) \triangleq p_{\theta_S}(y \mid \mathbf{e} + \Delta\mathbf{e}). \tag{14}$$

The KL divergence between p and q is given by

$$\mathrm{KL}(p\|q) = \mathbb{E}_{y\sim p}\left[\log \frac{p(y)}{q(y)}\right]. \tag{15}$$

We perform a second-order Taylor expansion of $\log q(y)$ around $\mathbf{e}$:

$$\log q(y) = \log p(y) + \Delta\mathbf{e}^\top \nabla_{\mathbf{e}} \log p(y) + \frac{1}{2}\Delta\mathbf{e}^\top \nabla_{\mathbf{e}}^2 \log p(y)\Delta\mathbf{e} + o(\|\Delta\mathbf{e}\|^2). \tag{16}$$

Substituting Eq. (16) into Eq. (15) yields

$$\mathrm{KL}(p\|q) = -\mathbb{E}_{y\sim p}\left[\Delta\mathbf{e}^\top \nabla_{\mathbf{e}} \log p(y) + \frac{1}{2}\Delta\mathbf{e}^\top \nabla_{\mathbf{e}}^2 \log p(y)\Delta\mathbf{e}\right] + o(\|\Delta\mathbf{e}\|^2). \tag{17}$$

The first-order term vanishes due to the score-function identity:

$$\mathbb{E}_{y\sim p}\left[\nabla_{\mathbf{e}} \log p(y)\right] = \mathbf{0}. \tag{18}$$

Thus, the leading term of the KL divergence is second-order:

$$\mathrm{KL}(p\|q) = -\frac{1}{2}\Delta\mathbf{e}^\top \mathbb{E}_{y\sim p}\left[\nabla_{\mathbf{e}}^2 \log p(y)\right]\Delta\mathbf{e} + o(\|\Delta\mathbf{e}\|^2). \tag{19}$$

Under standard regularity conditions, the negative expected Hessian of the log-likelihood equals the Fisher Information Matrix:

$$\begin{aligned}\mathbf{F}(\mathbf{e}) &= -\mathbb{E}_{y\sim p}\left[\nabla_{\mathbf{e}}^2 \log p(y)\right] \\ &= \mathbb{E}_{y\sim p}\left[\nabla_{\mathbf{e}} \log p(y) \nabla_{\mathbf{e}} \log p(y)^\top\right].\end{aligned} \tag{20}$$

Substituting Eq. (20) into Eq. (19) yields

$$\mathrm{KL}[p_{\theta_S}(y \mid \mathbf{e}) \,\|\, p_{\theta_S}(y \mid \mathbf{e} + \Delta\mathbf{e})] = \frac{1}{2}\Delta\mathbf{e}^\top \mathbf{F}(\mathbf{e})\Delta\mathbf{e} + o(\|\Delta\mathbf{e}\|^2). \tag{21}$$

A.2 Derivation of the Natural Gradient via KL-Constrained Optimization

We now derive the natural gradient by solving a local KL-constrained optimization problem. Consider

$$\begin{aligned}\min_{\Delta\mathbf{e}} \quad & \nabla_{\mathbf{e}}\mathcal{L}^\top \Delta\mathbf{e} \\ \text{s.t.} \quad & \mathrm{KL}[p_{\theta_S}(y \mid \mathbf{e}) \,\|\, p_{\theta_S}(y \mid \mathbf{e} + \Delta\mathbf{e})] \le \epsilon.\end{aligned} \tag{22}$$

Using the quadratic approximation in Eq. (21), the constraint becomes

$$\frac{1}{2}\Delta\mathbf{e}^\top \mathbf{F}\Delta\mathbf{e} \le \epsilon. \tag{23}$$

Introducing a Lagrange multiplier $\lambda > 0$, we define the Lagrangian

$$\mathcal{L}_{\mathrm{Lag}}(\Delta\mathbf{e}, \lambda) = \nabla_{\mathbf{e}}\mathcal{L}^\top \Delta\mathbf{e} + \lambda\left(\frac{1}{2}\Delta\mathbf{e}^\top \mathbf{F}\Delta\mathbf{e} - \epsilon\right). \tag{24}$$

Setting the gradient with respect to $\Delta\mathbf{e}$ to zero yields

$$\nabla_{\Delta\mathbf{e}}\mathcal{L}_{\mathrm{Lag}} = \nabla_{\mathbf{e}}\mathcal{L} + \lambda\mathbf{F}\Delta\mathbf{e} = \mathbf{0}. \tag{25}$$

Solving for $\Delta\mathbf{e}$ gives

$$\Delta\mathbf{e} = -\frac{1}{\lambda}\mathbf{F}^{-1}\nabla_{\mathbf{e}}\mathcal{L}. \tag{26}$$

The scalar λ is determined by the constraint and only affects the step size. Therefore, the steepest descent direction under the Fisher metric is given (up to a positive scalar) by

$$\tilde{\nabla}_{\mathbf{e}} \triangleq \mathbf{F}^{-1}\nabla_{\mathbf{e}}\mathcal{L}, \tag{27}$$

which is known as the *natural gradient*.