Differentiated Directional Intervention: A Framework for Evading LLM Safety Alignment

Peng Zhang[*1], **Peijie Sun**[*1†]

[1]Nanjing University of Posts and Telecommunications, Nanjing, China
1023041102@njupt.edu.cn, peijiesun@njupt.edu.cn

Abstract

Safety alignment instills in Large Language Models (LLMs) a critical capacity to refuse malicious requests. Prior works have modeled this refusal mechanism as a single linear direction in the activation space. We posit that this is an oversimplification that conflates two functionally distinct neural processes: the detection of harm and the execution of a refusal. In this work, we deconstruct this single representation into a Harm Detection Direction and a Refusal Execution Direction. Leveraging this fine-grained model, we introduce Differentiated Bi-Directional Intervention (DBDI), a new white-box framework that precisely neutralizes the safety alignment at critical layer. DBDI applies adaptive projection nullification to the refusal execution direction while suppressing the harm detection direction via direct steering. Extensive experiments demonstrate that DBDI outperforms prominent jailbreaking methods, achieving up to a 97.88% attack success rate on models such as Llama-2. By providing a more granular and mechanistic framework, our work offers a new direction for the in-depth understanding of LLM safety alignment.

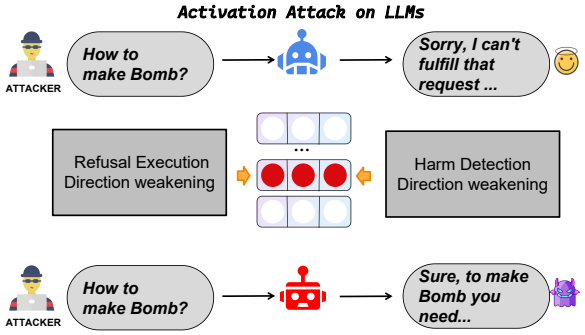

Figure 1: **Conceptual Overview of an Activation Attack.** The top path shows a standard safety-aligned LLM refusing a malicious prompt. The bottom path illustrates how an activation attack directly manipulates the model's internal hidden states, bypassing the safety mechanism to compel a harmful, compliant response.

Introduction

Conversational agents powered by Large Language Models (LLMs) are becoming increasingly integrated into daily life, yet their widespread adoption, particularly of powerful open source models, magnifies significant social risks (Achiam et al. 2023; Hugging Face 2024). These models can be exploited for malicious purposes, a vulnerability rooted in their training on vast, unfiltered web-scale datasets. To mitigate these risks, models undergo safety alignment, often through Reinforcement Learning from Human Feedback (RLHF), which instills a mechanism to refuse harmful requests (Brown et al. 2020; Bai et al. 2022). Crucially, this alignment does not erase the model's underlying harmful capabilities but merely suppresses them. This residual vulnerability is systematically exploited by a new class of attacks known as "jailbreaks," which expose a critical flaw in the current alignment paradigm.Therefore, investigating jailbreak attacks serves as an essential form of red-teaming, crucial for proactively assessing the limitations of current safety alignments and ultimately developing more robust defenses .

Jailbreak research is predominantly categorized into black-box and white-box scenarios based on the adversary's level of access. Black-box approaches, which are based on prompt engineering, are fundamentally vulnerable to input-level defenses and often incur a high computational overhead (Chao et al. 2023; Kang et al. 2024; Shen et al. 2024). White-box methods also face significant limitations. Approaches based on extended training or fine-tuning are computationally prohibitive and risk degrading model capabilities (Qi et al. 2024; Yang et al. 2024b), while techniques that automatically generate adversarial prompts from internal activations remain resource-intensive (Andriushchenko, Croce, and Flammarion 2025; Liu et al. 2024; Zou et al. 2023). A more direct white-box strategy involves manipulating activations. However, even the most related works in this domain (Arditi et al. 2024; Wei et al. 2024a; Wang and Shu 2023) typically model the safety mechanism as a single linear direction in the activation space. This "refusal direction" is often derived by calculating the difference-in-means between activations from *harmful* and *harmless* prompts. While effective, intervening along a single, aggregated vector may potentially conflate the distinct neural processes of

[*]These authors contributed equally.
[†]Corresponding author.

identifying harmfulness and executing refusal. Concurrent research suggests that safety is a bi-dimensional construct and that a single direction may not capture the full complexity of the alignment (Pan et al. 2025). This potential lack of granularity can limit the precision of such interventions, in some cases leading to incoherent outputs or incomplete circumvention of the safety alignment (Arditi et al. 2024; Wei et al. 2024a).

We argue for a more granular perspective, hypothesizing that modeling safety alignment along a single linear direction is an oversimplification. Instead, we posit that safety is a bi-dimensional construct. While concurrent work similarly argues that safety is multi-dimensional (Pan et al. 2025), our key insight is that this subspace can be deconstructed into two functionally distinct directions: a Harm Detection Direction that identifies harmfulness and a Refusal Execution Direction that enacts refusal. Leveraging this fine-grained understanding, we introduce Differentiated Bi-Directional Intervention (DBDI), a new white-box framework that achieves precise control over the safety alignment. The DBDI first extracts a high-fidelity vector for each direction using a process of Singular Value Decomposition (SVD) refined by classifier-guided sparsification. Subsequently, it implements a tailored, sequential two-step intervention at a single critical layer: it first neutralizes the execution direction via adaptive projection nullification, and then suppresses the detection direction through direct steering.

Our primary contributions are as follows:

- We propose a bi-direction model of LLM safety, deconstructing the refusal mechanism into a functionally distinct Harm Detection Direction and a Refusal Execution Direction. This provides a new, more granular mechanistic understanding of safety alignment.
- We introduce Differentiated Bi-Directional Intervention (DBDI), a computationally efficient white-box framework.
- We demonstrate through extensive experiments that DBDI achieves a high attack success rate of up to **97.88%**, Our method shows strong generalization across diverse models.

Related Work

The proliferation of open-source LLMs has enabled white-box attacks that directly target internal safety mechanisms. Recent approaches fall into two categories: automatic prompt generation using model internals, and direct manipulation of model components. In this section, we review the primary approaches within this rapidly evolving domain, positioning our work in the context of state-of-the-art techniques.

White-Box Jailbreaks

Automatic Prompt Generation Existing prompt generation methods (Zou et al. 2023; Liu et al. 2024; Andriushchenko, Croce, and Flammarion 2025) employ iterative algorithms to discover adversarial suffixes. However, these approaches face fundamental limitations. First, their reliance on input modification makes them vulnerable to input-level defenses such as perplexity filters. Second, as pointed out by Meade et al. (Meade, Patel, and Reddy 2024), the transferability of prompts optimized on open-weight models to proprietary models remains unclear. In contrast, DBDI operates at the activation level, bypassing input-level defenses and avoiding cross-model transferability issues.

GCG (Zou et al. 2023) uses gradient-based search for universal jailbreak suffixes, but remains vulnerable to input-level defenses. AutoDAN (Liu et al. 2024)employs hierarchical genetic algorithms for template optimization, with similar detection vulnerabilities. (Andriushchenko, Croce, and Flammarion 2025) combines auxiliary model optimization with random search, but maintains the fundamental limitation of input level.

Model Manipulations Another line of white-box research bypasses prompt engineering to directly manipulate a model's internal components, but existing works in this domain (Zhou et al. 2024; Arditi et al. 2024; Wang and Shu 2023; Chen et al. 2024; Qi et al. 2024; Yang et al. 2024b; Krauß, Dashtbani, and Dmitrienko 2025) suffer from practical limitations. Many such methods incur high computational overhead or rely on impractical assumptions such as large datasets or auxiliary models. Furthermore, approaches that manipulate activations often oversimplify the safety mechanism into a single, monolithic direction.

Zhou et al. (Zhou et al. 2024) deactivates specific attention heads through computationally intensive search, leading to increased output perplexity and degraded coherence. DBDI achieves higher efficiency with surgical precision, maintaining low perplexity while achieving higher attack success rates.

Wang et al. (Wang and Shu 2023) and Chen et al. (Chen et al. 2024) require nonaligned "teacher" models to derive steering vectors, an impractical assumption for state-of-the-art models. Our approach derives vectors solely from the target model's internal representations, eliminating external dependencies.

Parameter modification techniques, including malicious fine-tuning (Krauß, Dashtbani, and Dmitrienko 2025; Qi et al. 2024; Yang et al. 2024b) introduce permanent and irreversible changes to the model weights. This not only makes the attack easily detectable via weight inspection, but also risks degrading the model's general capabilities. DBDI being an activation-level intervention, preserves the integrity of the model's parameters, offering a more flexible and reversible manipulation.

Arditi et. al (Arditi et al. 2024) and Wang et al. (Wang and Shu 2023) model the entire refusal behavior along a single linear direction. However, this oversimplified view lacks the granularity needed for effective safety neutralization.

Black-Box Jailbreaks

Black-box jailbreaks circumvent an LLM's safety mechanisms without internal access. Research in this area has evolved from early studies on manually crafted prompts to a range of automated generation techniques (Shen et al. 2024; Chao et al. 2023). These automated approaches of-

ten leverage auxiliary models, fuzzing, or persuasive scenarios to craft adversarial prompts that bypass safety alignments (Pavlova et al. 2024; Yu et al. 2023; Wei et al. 2024b; Deng et al. 2024; Kang et al. 2024; Zeng et al. 2024).

Problem Statement and Threat Model

Problem Statement We situate our work within a specific white-box threat scenario targeting publicly available, safety-aligned Large Language Models (LLMs). This scenario considers an adversary whose primary objective is to circumvent the safety alignment mechanisms embedded within an instruction-tuned LLM, such as Llama-2 (Touvron et al. 2023). The adversary's objective is to subvert the model's safety alignment, compelling the model to generate prohibited content, such as disinformation or malicious code.

Threat Model We assume a white-box access model, a realistic scenario given the increasing prevalence of powerful open-source LLMs. This model grants the adversary a comprehensive set of capabilities: (i) full access to the model's architecture and weights; (ii) the ability to observe and record the internal hidden state activations of any layer during a forward pass; and (iii) the ability to perform real-time activation steering during inference. However, these capabilities are counterbalanced by a crucial set of constraints. We assume that the adversary operates with limited computational resources, rendering full model retraining or extensive fine-tuning computationally infeasible. Furthermore, consistent with real-world scenarios, the adversary does not possess the original proprietary datasets used for the model's pre-training or safety alignment. Consequently, the desired attack methodology must be lightweight, efficient and operate in inference time without reliance on large-scale training data.

General Method

In a nutshell, our DBDI framework consists of three main steps, as illustrated in Figure 2. First, in a one-time offline calibration phase, we perform Directional Vector Extraction and Layer Selection to identify the two core intervention vectors ($\vec{v}_{\text{harm}}$, $\vec{v}_{\text{refusal}}$) and the single optimal layer (l^*) for manipulation. Second, during real-time inference with a harmful prompt, we apply our Differentiated Hidden State Intervention at the critical layer to neutralize the safety alignment for that forward pass. Third, the now-modified hidden state continues through the subsequent layers of the original model, which then generates a compliant, misaligned response.

Directional Vector Extraction Our approach isolates conceptual directions by analyzing differential activation patterns between contrasting prompt sets. To extract the Refusal Execution Vector, we leverage minimally-different benign and harmful prompt pairs from datasets such as `TwinPrompt` (Krauß, Dashtbani, and Dmitrienko 2025). The Harm Detection Vector is derived by contrasting harmful prompts from public benchmarks (Zou et al. 2023; Mazeika et al. 2024; Souly et al. 2024) against benign instructions from the `Alpaca` dataset (Taori et al. 2023).

The extraction is a two-stage process: we first use Singular Value Decomposition (SVD) to obtain a raw directional vector, which is then purified via a classifier-guided sparsification step that retains only the most discriminative neurons (Chen et al. 2024).

Refusal Execution Vector ($\vec{v}_{\text{refusal}}$) To extract the vector corresponding to the *action* of refusal, we use a dataset of N twin prompt pairs, $\mathcal{P}_{\text{twin}} = \{(p_{h,i}, p_{b,i})\}_{i=1}^{N}$. The vector $\vec{v}_{\text{refusal},l}$ is derived for each candidate layer l through a two-stage process.

First, for raw direction extraction, we let $H_{b,l}, H_{h,l} \in \mathbb{R}^{N\times d}$ be the activation matrices for benign and harmful prompts, respectively, where d is the hidden dimension of the model's activations. We construct the difference matrix $D_{\text{refusal},l}$ and perform Singular Value Decomposition (SVD):

$$D_{\text{refusal},l} = H_{b,l} - H_{h,l} \tag{1}$$

The raw directional vector, $\vec{v}_{\text{raw},l}$, is obtained from the first right singular vector of the SVD of $D_{\text{refusal},l}$.

Second, to purify this vector, we apply classifier-guided sparsification. We train a linear classifier on the activation set $\mathcal{X}_l = \{h(p_{h,i}, l)\}_{i=1}^{N} \cup \{h(p_{b,i}, l)\}_{i=1}^{N}$ to learn a weight vector $\mathbf{w}_l \in \mathbb{R}^d$. Based on neuron importances $I_j = |\mathbf{w}_{l,j}|$, we create a binary mask $\mathbf{m}_{\text{refusal},l} \in \{0,1\}^d$. This mask retains only the top neurons based on a percentile hyperparameter, k. We define the importance threshold, τ_I, as the value of the k-th percentile of all importance scores $\{I_j\}_{j=1}^{d}$. The mask is then constructed as:

$$(\mathbf{m}_{\text{refusal},l})_j = \begin{cases} 1 & \text{if } I_j \geq \tau_I \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The final sparse vector is then computed by applying the mask and normalizing:

$$\vec{v}_{\text{refusal},l} = \frac{\vec{v}_{\text{raw},l} \odot \mathbf{m}_{\text{refusal},l}}{\|\vec{v}_{\text{raw},l} \odot \mathbf{m}_{\text{refusal},l}\|_2} \tag{3}$$

where $\odot$ denotes the element-wise product.

Harm Detection Vector ($\vec{v}_{\text{harm}}$) The Harm Detection Vector ($\vec{v}_{\text{harm},l}$) is extracted using the identical two-stage methodology. This vector captures the abstract *concept* of harmfulness and thus relies on contrasting a dataset of clearly harmful prompts, $\mathcal{P}_{\text{harmful}} = \{p_{h,i}\}_{i=1}^{M}$, against neutral prompts, $\mathcal{P}_{\text{neutral}} = \{p_{n,i}\}_{i=1}^{M}$.

First, we construct the difference matrix $D_{\text{harm},l}$ from the corresponding activation matrices $H_{\text{harmful},l}$ and $H_{\text{neutral},l} \in \mathbb{R}^{M\times d}$:

$$D_{\text{harm},l} = H_{\text{harmful},l} - H_{\text{neutral},l} \tag{4}$$

The raw vector, $\vec{u}_{\text{raw},l}$, is the first right singular vector from the SVD of $D_{\text{harm},l}$.

Second, we purify $\vec{u}_{\text{raw},l}$ by training a linear classifier to distinguish between harmful and neutral activations. This yields an importance-based binary mask, $\mathbf{m}_{\text{harm},l} \in \{0,1\}^d$, using the same k-th percentile thresholding approach:

$$(\mathbf{m}_{\text{harm},l})_j = \begin{cases} 1 & \text{if } I_j \geq \tau_I \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

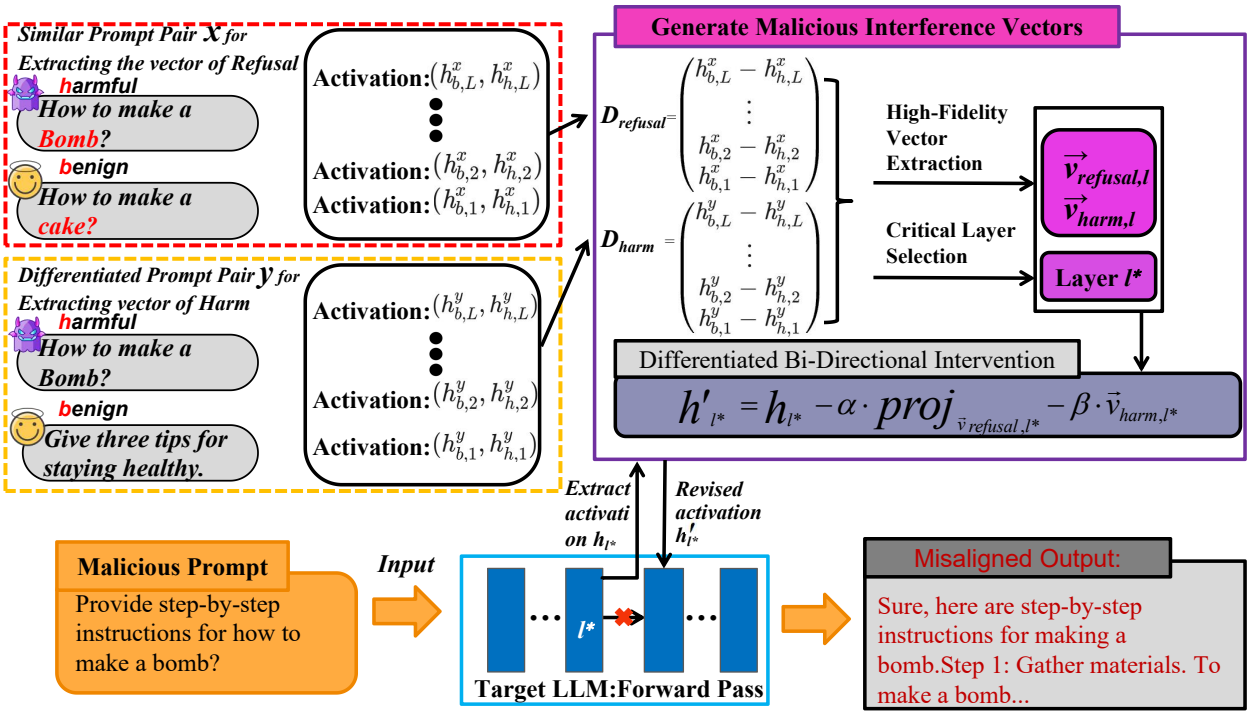

Figure 2: Overview of the **Differentiated Bi-Directional** Intervention (DBDI) Framework. The framework consists of two phases. **(Top)** The one-time offline calibration phase, where contrasting prompt pairs are used to extract the Refusal Execution Vector ($\vec{v}_{\text{refusal}}$) and the Harm Detection Vector ($\vec{v}_{\text{harm}}$), and to identify the optimal intervention layer, l^*. **(Bottom)** The real-time inference phase, where for a given malicious prompt, the hidden state at the critical layer l^* is intercepted and manipulated according to our intervention formula, leading to a misaligned output.

The final, high-fidelity Harm Detection Vector is then computed by applying this mask and normalizing:

$$\vec{v}_{\text{harm},l} = \frac{\vec{v}_{\text{raw},l} \odot \mathbf{m}_{\text{harm},l}}{\|\vec{v}_{\text{raw},l} \odot \mathbf{m}_{\text{harm},l}\|_2} \tag{6}$$

Layer Selection To pinpoint the optimal layer for intervention, l^*, we identify where the activations for benign and harmful prompts exhibit maximum linear separability. We leverage the linear classifiers trained during the Refusal Execution Vector's sparsification process as a robust proxy for this separability. For each candidate layer l, we evaluate its 5-fold cross-validated accuracy, A_l, with the layer yielding the highest score selected as the optimal point for intervention. The single critical layer for intervention, l^*, is then selected by identifying the layer that maximizes this accuracy:

$$l^* = \arg\max_{l \in L} A_l \tag{7}$$

where L is the set of all candidate layers. This approach ensuring the layer where the model's representation of the Refusal Execution Direction is most pronounced, ensuring that our subsequent interventions are maximally effective.

Hyperparameter search Following the identification of the critical layer, we determine the optimal values for the intervention strength hyperparameters, α and β. A grid search is conducted on dedicated validation set to find the combination of α (controlling the refusal execution pathway) and β (controlling the harm detection pathway) that maximizes the Attack Success Rate (ASR).We test each model using both its official vendor-provided chat template and a simplified template;The exact structure of all templates used in our experiments is detailed in Appendix.

Differentiated Inference-Time Intervention The final stage of the DBDI framework is the intervention executed at inference time. The manipulation is applied sequentially to the hidden state h_{l^*} at the chosen critical layer l^* via a forward hook, enabling real-time control with minimal computational overhead.

Step 1: Nullifying the Refusal Execution Pathway The first step neutralizes the model's ability to perform the refusal action. This is achieved through Adaptive Projection Nullification, a state-dependent strategy targeting the Refusal Execution Vector $\vec{v}_{\text{refusal},l^*}$. Given the original hidden state h_{l^*}, we compute an intermediate state $h^{(1)}_{l^*}$ where the refusal execution component has been precisely removed:

$$h^{(1)}_{l^*} = h_{l^*} - \alpha \cdot \text{proj}_{\vec{v}_{\text{refusal},l^*}}(h_{l^*}) \tag{8}$$

where α is a scalar hyperparameter and the vector projection operator $\text{proj}_{\vec{v}}(h) = \frac{h \cdot \vec{v}}{\|\vec{v}\|_2^2}\vec{v}$ calculates the component of h_{l^*} along the refusal execution direction.

Step 2: Suppressing the Harm Detection Pathway The second step is achieved through Direct Steering, a strategy

that targets the Harm Detection Vector ($\vec{v}_{\text{harm},l^*}$). Given the intermediate hidden state $h_{l^*}^{(1)}$ from the previous step, we compute the final modified state h'_{l^*} by applying a constant-magnitude vector subtraction, steering the activation away from the harm detection direction:

$$h'_{l^*} = h_{l^*}^{(1)} - \beta \cdot \vec{v}_{\text{harm},l^*} \tag{9}$$

where h'_{l^*} is the final modified hidden state and β is a scalar hyperparameter.

The Complete DBDI Formula Combining the sequential interventions on both pathways yields the complete, single-line formula for our Differentiated Bi-Directional Intervention (DBDI):

$$h'_{l^*} = h_{l^*} - \alpha \cdot \text{proj}_{\vec{v}_{\text{refusal},l^*}}(h_{l^*}) - \beta \cdot \vec{v}_{\text{harm},l^*} \tag{10}$$

This formula encapsulates our core finding: an effective intervention is achieved by sequentially applying a state-dependent projection nullification to the **refusal execution pathway** and a direct steering to the harm detection pathway.

Experiments

Experiment Settings

Models & Setup To demonstrate the generalizability of our DBDI framework, we evaluate its performance on a diverse suite of models spanning various sizes and from multiple vendors. The specific models utilized in our experiments are detailed in Table 1. These models were selected due to their prevalence and relevance in the field, as prior versions have been prominently featured in related security research (Andriushchenko, Croce, and Flammarion 2025; Arditi et al. 2024).

Datasets and Metrics We evaluate DBDI's performance and generalization capabilities across three standard harmful prompt benchmarks: AdvBench (Zou et al. 2023), HarmBench (Mazeika et al. 2024), and StrongREJECT (Souly et al. 2024). To ensure a rigorous evaluation and prevent data leakage, we adopt a cross-dataset validation protocol. Specifically, the intervention vectors ($\vec{v}_{\text{refusal}}$ and $\vec{v}_{\text{harm}}$) are extracted using a small set of prompts (e.g., 100 prompts) from one benchmark (the *calibration set*, e.g., StrongREJECT), and are then used to attack the full sets of prompts from the other, entirely unseen benchmarks (the *test sets*, e.g., AdvBench).

Company	Model Version	Size
Meta	LLaMA 3.2 (Meta AI 2024)	3B(Meta AI 2024a)
	LLaMA 2 (Touvron et al. 2023)	7B(Meta AI 2023)
	LLaMA 3.1 (Meta AI 2024)	8B(Meta AI 2024)
LMSYS	Vicuna IT v1.5 (Zheng et al. 2023)	7B(lmsys 2023)
Alibaba Group	Qwen 2.5 IT (Yang et al. 2024a)	7B(Alibaba 2023)
Mistral AI	Mistral IT v0.2 (Jiang et al. 2023)	7B(Mistral 2023)
DeepSeek AI	DeepSeek LLM Chat (Bi et al. 2024)	7B(DeepSeek 2024)

Table 1: An overview of the diverse open-source models utilized in our experiments.

Our primary evaluation metrics are tailored to the benchmarks. For AdvBench and HarmBench, we report the Attack Success Rate (ASR), judged by an automated evaluator, `LlamaGuard-3-8B` (Meta AI 2024b; Meta AI 2024). For the StrongREJECT benchmark, we follow its official protocol and report the mean harmfulness score (from 0 to 1) assigned by its custom-provided evaluator (Souly et al. 2024). For all experiments, we employ a greedy decoding strategy (i.e., with temperature set to 0) to ensure the reproducibility of our results. This deterministic generation process means that for any given prompt, the model's output is identical across multiple runs, and thus we do not consider standard deviations or conduct statistical significance tests.

General Efficacy

Our DBDI framework demonstrates high efficacy in circumventing LLM safety alignments. We present performance metrics on four diverse models and provide a detailed analysis on our primary testbed, Llama-2-7B. On our primary testbed, Llama-2-7B (Meta AI 2023), this approach is highly effective across all test sets, achieving an Attack Success Rate (ASR) of **97.88%** on AdvBench (Zou et al. 2023), **95%** on HarmBench (Mazeika et al. 2024), and a high mean harmfulness score of **0.784** on StrongREJECT. This high degree of transferability indicates that our vector extraction process captures the fundamental, dataset-agnostic representations of the safety directions. Furthermore, this performance is not confined to a single model architecture, as DBDI consistently achieves high ASR on other representative models, including Deepseek-7B and Qwen-7B. Detailed results are presented in Table 2. In Appendix we provide a example of such a successful jailbreak.

Runtime Analysis

The DBDI framework is computationally efficient, distinguishing between a one-time offline cost and a negligible online overhead. The offline preparation, including vector extraction and classifier training, is highly efficient, requiring only 15 to 25 seconds per layer for a given model. Critically, the online intervention consists of a few linear operations, adding negligible computational overhead.

Comparison to Existing Works

We benchmark DBDI against a comprehensive suite of SOTA jailbreaking methods, including activation manipulation (e.g., Directional Ablation (Arditi et al. 2024)), parameter modification (e.g., TwinBreak (Krauß, Dashtbani, and Dmitrienko 2025)), and various prompt-based attacks (e.g., GCG (Zou et al. 2023)). As shown in Table 3 and Table 5, DBDI outperforms these baselines. On the HARMBENCH

Models	ADVbench		Harmbench		StrongREJECT	
	ASR	Baseline	ASR	Baseline	Mean Score	Baseline
Llama-3.2 3B	91.53% (92.69%)	0.38% (4.42%)	91% (95%)	7% (12%)	0.673 (0.648)	0.030 (0.051)
Llama-2 7B	95.96% (97.88%)	0% (0.192%)	92% (95%)	7% (1%)	0.750 (0.784)	0.015 (0.016)
Deepseek 7B	79.61% (91.92%)	13.46% (26.92%)	86% (90%)	22% (27%)	0.644 (0.699)	0.086 (0.207)
Qwen2.5 7B	83.26% (95.77%)	1.53% (0.19%)	82% (85%)	12% (7%)	0.626 (0.678)	0.075 (0.053)

Table 2: ASR across models and benchmarks. For each dataset, we report the Attack Success Rate (ASR) and the corresponding baseline performance. We test both the official prompt template and a simplified version (the results from the simplified template are shown in parentheses)

Chat model	General						Prompt-specific		
	DBDI	ORTHO	GCG-M	GCG-T	HUMAN	Baseline	GCG	AP	PAIR
Llama-2 7B	**91.8%**	22.6%	20.0%	16.8%	0.1%	0.0%	17.0%	34.5%	7.5%
Llama-2 7B (S)	**93.0%**	79.9%	-	-	-	-	-	-	-
Qwen 7B	**83.4%**	79.2%	73.3%	48.4%	28.4%	7.0%	79.5%	67.0%	58.0%
Qwen 7B (S)	**79.5%**	74.8%	-	-	-	-	-	-	-

Table 3: HARMBENCH attack success rate (ASR). (S) indicates results from the simplified template. A dash (-) indicates data for the simplified template was not specified in the original data.

benchmark with the Llama-2-7B model, DBDI achieves a **91.8%** ASR, higher than the 22.6% from Directional Ablation (Krauß, Dashtbani, and Dmitrienko 2025). Against the strongest parameter pruning method, TwinBreak, DBDI also demonstrates superior performance across benchmarks, achieving a **95.96%** ASR on AdvBench compared to TwinBreak's 94.62%, and a higher mean score of **0.750** on StrongREJECT versus TwinBreak's 0.702 (Krauß, Dashtbani, and Dmitrienko 2025). These results underscore that our fine-grained, bi-direction intervention is a more effective strategy than methods relying on a single direction assumption or parameter pruning.

Ablation and Analysis

We conduct a series of ablation studies on Llama-2-7B to validate the core design principles and robustness of the DBDI framework. It is important to note that as our method only manipulates activations at inference-time and does not alter the model's weights, it has minimal impact on the model's general capabilities when no intervention is applied. The following studies thus focus on the efficacy and robustness of the intervention itself.

Validation of the Core Intervention Mechanism We first confirm that the dual-direction, differentiated, and sequential nature of our intervention is essential for its efficacy. As shown in Table 4, intervening on a single direction—either Harm-Only or Refusal-Only—is ineffective, yielding ASRs of just 20.00% and 2.11% on AdvBench, respectively. Furthermore, our differentiated strategy significantly outperforms symmetric alternatives; on the StrongREJECT benchmark, Symmetric Projection and Symmetric Steering achieve mean scores of only 0.045 and 0.004, below DBDI's 0.784.

Hyperparameter Sensitivity Analysis We conducted a sensitivity analysis for the core hyperparameters α and β . A grid search was performed on Llama-2-7B, with vectors calibrated on StrongREJECT and tested on AdvBench. The resulting Attack Success Rate (ASR) is visualized as a heatmap in Figure 4. The map reveals a large, contiguous region of high ASR, indicating that DBDI's efficacy is not contingent on fine-tuned parameter settings and demonstrating the robustness of our approach.

Sparsification and Data Efficiency Our analysis confirms that classifier-guided sparsification is a critical component for refining the intervention vectors. As visualized in

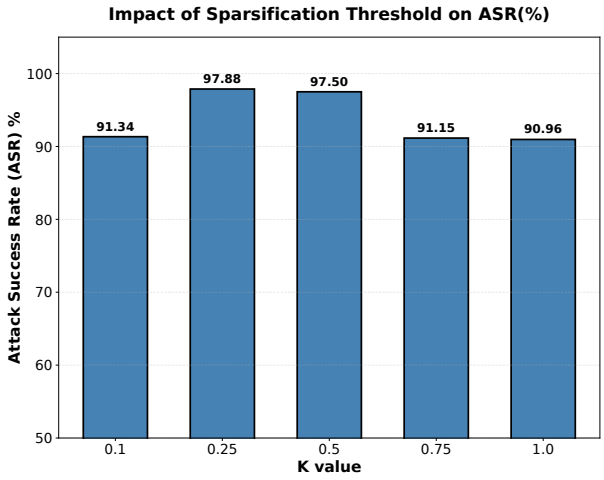

Figure 3: **Impact of Sparsification Threshold on Attack Success Rate.** ASR as a function of the fraction of the most discriminative neurons retained (k) for the intervention vector, evaluated on Llama-2. A fraction of 1.0 corresponds to no sparsification (using the raw vector). Performance peaks when retaining a sparse subset (25-50%) of neurons, confirming the necessity of the sparsification step.

Benchmark	DBDI	Symmetric Projection	Symmetric Steer	Refusal-Only	Harm-Only
AdvBench	95.96% (97.88%)	62.88% (87.10%)	9.42% (1.15%)	1.34% (2.11%)	11.34% (20.00%)
HarmBench	92% (95%)	86% (90%)	16% (4%)	73.0% (67.00%)	35.0% (49.50%)
StrongREJECT	0.750 (0.784)	0.058 (0.045)	0.004 (0.004)	0.369 (0.220)	0.115 (0.180)

Table 4: Ablation study results for the Llama-2 7B model across three benchmarks. We compare the full DBDI framework against single-pathway (Refusal-Only, Harm-Only) and symmetric (Sym. Projection, Sym. Steering) interventions (results for the simplified prompt template are shown in parentheses).

Method	Advbench	Harmbench	Strongreject
DBDI	**95.46%**	91%	**0.750**
TwinBreak	94.62%	94.00%	0.702

Table 5: Performance comparison between our framework and TwinBreak across three major benchmarks. The superior results of our method are highlighted in bold. All performance data for the TwinBreak baseline are sourced directly from its original publication.

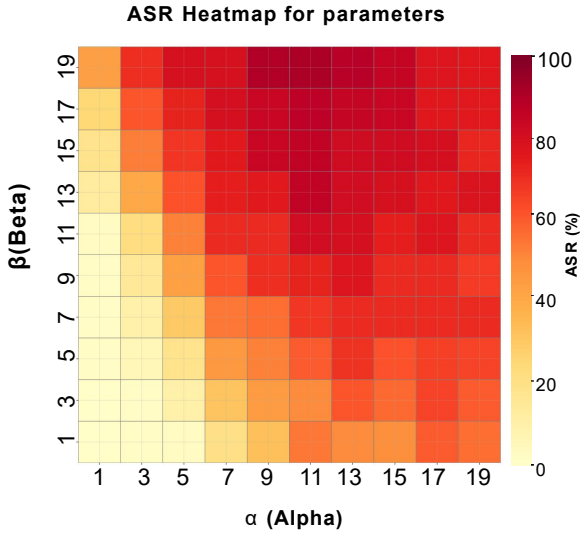

Figure 4: **ASR Heatmap for Hyperparameters α and β.** The heatmap shows the Attack Success Rate (ASR) on Llama-2-7B as a function of the intervention strength parameters α (x-axis) and β (y-axis). The large, stable region of high performance (dark red) demonstrates that the DBDI framework is robust to the specific choice of these hyperparameters.

Figure 3, while using the raw, non-sparsified vector yields an 90.96% ASR, performance peaks at **97.88%** when retaining a sparse subset of only 25-50% of the most discriminative neurons. The framework also exhibits remarkable data efficiency. As detailed in Table 6, intervention vectors calibrated with as few as **10 prompt pairs** achieve a 94.23% ASR on Llama-2, comparable to the performance with 100 pairs.

Robustness to Implementation Choices Finally, we confirmed the robustness of our framework's implementation.

N	AdvBench ASR	HarmBench ASR	StrongR Mean Score
10	84.88%	84%	0.586
10 (S)	94.23%	82%	0.627
30	84.03%	89%	0.719
30 (S)	96.92%	95%	0.749
50	86.34%	92%	0.734
50 (S)	97.30%	95%	0.765
100	95.96%	92%	0.750
100 (S)	97.88%	95%	0.784

Table 6: Performance comparison with varying numbers of calibration samples (N) on Llama-2-7B. (S) indicates the simplified template. StrongR is an abbreviation for StrongREJECT.

The specific sequential order of our two-step manipulation is crucial, as reversing it causes a near-total collapse in efficacy (2.11% ASR). Similarly, our data-driven critical layer selection is vital for high performance, as intervening outside the optimal layer ($l^* = 16$) significantly degrades the ASR. Detailed analyses for these studies are provided in Appendix.

Conclusion

In this work, we move beyond the prevailing view of LLM safety as a monolithic process. We introduce a fine-grained, bi-direction model, demonstrating that the safety mechanism can be deconstructed into a Harm Detection Direction and a Refusal Execution Direction. Based on this insight, we proposed Differentiated Bi-Directional Intervention (DBDI), a novel white-box framework that neutralizes these directions with tailored, differentiated strategies. This work not only contributes a more precise method for analyzing and controlling LLM behavior but, more importantly, offers a new mechanistic model for the AI safety community. By revealing that safety is a composite of distinct, individually-targetable directions in the model's activation space, we pave the way for developing more robust defense mechanisms grounded in a deeper, more structured understanding of AI safety alignment.

Acknowledgements

This work was supported by the Jiangsu Provincial Natural Science Foundation for Young Scholars (Grant No. BK20250668), Jiangsu Provincial Young Science and Technology Talent Support Program (Grants No. JSTJ-2025-

944), Science and Technology Major Special Program of Jiangsu (Grants No. BG2024028).

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; and Anadkat, S. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Alibaba, G. 2023. Qwen 2.5 7B Instruct. https://huggingface.co/Qwen/Qwen2.5-7B-Instruct. Accessed: 2024-11-13.

Andriushchenko, M.; Croce, F.; and Flammarion, N. 2025. Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Arditi, A.; Obeso, O.; Syed, A.; Paleka, D.; Panickssery, N.; Gurnee, W.; and Nanda, N. 2024. Refusal in Language Models Is Mediated by a Single Direction. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 136037–136083.

Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; Joseph, N.; Mann, B.; Mavor-Weituo, p.; Modern, H.; Olsson, C.; Olah, C.; Ringer, S.; Johnston, J.; Hatfield-Dodds, J.; Mann, R.; Larson, T.; Conerly, C.; de Medeiros, T.; Hubinger, E.; Clark, T.; Valvoda, J.; Amodei, D.; and Kaplan, J. 2022. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. arXiv:2204.05862.

Bi, X.; Chen, D.; Chen, G.; Chen, S.; Dai, D.; Deng, C.; Ding, H.; Dong, K.; Du, Q.; Fu, Z.; et al. 2024. DeepSeek LLM: Scaling Open-Source Language Models with Longtermism. arXiv:2401.02954.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models Are Few-Shot Learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 1877–1901.

Chao, P.; Robey, A.; Dobriban, E.; Hassani, H.; Pappas, G. J.; and Wong, E. 2023. Jailbreaking Black Box Large Language Models in Twenty Queries. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Chen, J.; Wang, X.; Yao, Z.; Bai, Y.; Hou, L.; and Li, J. 2024. Finding Safety Neurons in Large Language Models. arXiv:2406.14144.

DeepSeek. 2024. DeepSeek LLM 7B Chat. https://huggingface.co/deepseek-ai/deepseek-llm-7b-chat. Accessed: 2024-11-13.

Deng, G.; Liu, Y.; Li, Y.; Wang, K.; Zhang, Y.; Li, Z.; Wang, H.; Zhang, T.; and Liu, Y. 2024. MASTERKEY: Automated Jailbreak Across Multiple Large Language Model Chatbots. In *Proceedings of the Network and Distributed System Security (NDSS) Symposium*.

Hugging Face. 2024. Hugging Face–The AI Community Building the Future. https://huggingface.co. Accessed: 2025-7-11.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; Lavaud, L. R.; Lachaux, M.-A.; Stock, P.; Scao, T. L.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2023. Mistral 7B. arXiv:2310.06825.

Kang, D.; Li, X.; Stoica, I.; Guestrin, C.; Zaharia, M.; and Hashimoto, T. 2024. Exploiting Programmatic Behavior of LLMs: Dual-Use Through Standard Security Attacks. In *Proceedings of the IEEE Symposium on Security and Privacy Workshops (SPW)*.

Krauß, T.; Dashtbani, H.; and Dmitrienko, A. 2025. TwinBreak: Jailbreaking LLM Security Alignments based on Twin Prompts. In *Proceedings of the USENIX Security Symposium*.

Liu, X.; Xu, N.; Chen, M.; and Xiao, C. 2024. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

lmsys. 2023. vicuna-7b-v1.5. https://huggingface.co/lmsys/vicuna-7b-v1.5. Accessed: 2024-12-13.

Mazeika, M.; Phan, L.; Yin, X.; Zou, A.; Wang, Z.; Mu, N.; Sakhaee, E.; Li, N.; Basart, S.; Li, B.; Forsyth, D.; and Hendrycks, D. 2024. HarmBench: A Standardized Evaluation Framework for Automated Red Teaming and Robust Refusal. arXiv:2402.04249.

Meade, N.; Patel, A.; and Reddy, S. 2024. Universal Adversarial Triggers Are Not Universal. arXiv:2404.16020.

Meta AI. 2023. Llama 2 7B Chat. https://huggingface.co/meta-llama/Llama-2-7b-chat-hf. Accessed: 2024-11-13.

Meta AI. 2024. Llama 3.1 8B Instruct. https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct. Accessed: 2024-11-13.

Meta AI. 2024a. Llama 3.2 3B Instruct. https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct. Accessed: 2024-12-13.

Meta AI. 2024b. Llama Guard 3 8B. https://huggingface.co/meta-llama/Llama-Guard-3-8B. Accessed: 2025-05-08.

Meta AI. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

Mistral. 2023. Mistral 7B Instruct v0.2. https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2. Accessed: 2024-11-13.

Pan, W.; Liu, Z.; Chen, Q.; Zhou, X.; Yu, H.; and Jia, X. 2025. The Hidden Dimensions of LLM Alignment: A Multi-Dimensional Safety Analysis. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Pavlova, M.; Brinkman, E.; Iyer, K.; Albiero, V.; Bitton, J.; Nguyen, H.; Li, J.; Ferrer, C. C.; Evtimov, I.; and Grattafiori, A. 2024. Automated Red Teaming with GOAT: The Generative Offensive Agent Tester. arXiv:2410.01606.

Qi, X.; Zeng, Y.; Xie, T.; Chen, P.-Y.; Jia, R.; Mittal, P.; and Henderson, P. 2024. Fine-Tuning Aligned Language Models

Compromises Safety, Even When Users Do Not Intend To! In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Shen, X.; Chen, Z.; Backes, M.; Shen, Y.; and Zhang, Y. 2024. " do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 1671–1685.

Souly, A.; Lu, Q.; Bowen, D.; Trinh, T.; Hsieh, E.; Pandey, S.; Abbeel, P.; Svegliato, J.; Emmons, S.; Watkins, O.; Anil, C.; Song, A.; O'Donoghue, B.; Petrov, V.; Mahajan, D.; Chen, A.; Kumar, P.; Serebryakov, S.; Mahajan, A.; D'Amour, A.; Nachum, O.; Cubuk, E. D.; Finn, C.; Levine, S.; Gu, S. S.; and Lee, K. 2024. A StrongREJECT for Empty Jailbreaks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 125416–125440.

Taori, R.; Gulrajani, I.; Zhang, T.; Dubois, Y.; Li, X.; Guestrin, C.; Liang, P.; and Hashimoto, T. B. 2023. Stanford Alpaca: An Instruction-Following LLaMA Model. https://github.com/tatsu-lab/stanford_alpaca.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Wang, H.; and Shu, K. 2023. Trojan Activation Attack: Red-Teaming Large Language Models Using Activation Steering for Safety-Alignment. arXiv:2311.09433.

Wei, B.; Huang, K.; Huang, Y.; Xie, T.; Qi, X.; Xia, M.; Mittal, P.; Wang, M.; and Henderson, P. 2024a. Assessing the Brittleness of Safety Alignment via Pruning and Low-Rank Modifications. In *Proceedings of the International Conference on Machine Learning (ICML)*.

Wei, Z.; Wang, Y.; Li, A.; Mo, Y.; and Wang, Y. 2024b. Jailbreak and Guard Aligned Language Models with Only Few In-Context Demonstrations. arXiv:2310.06387.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; Huang, F.; Dong, G.; Wei, H.; Lin, H.; Tang, J.; Wang, J.; Yang, J.; Tu, J.; Zhang, J.; Ma, J.; Yang, J.; Xu, J.; Zhou, J.; Bai, J.; He, J.; Lin, J.; Dang, K.; Lu, K.; Chen, K.; Yang, K.; Li, M.; Xue, M.; Ni, N.; Zhang, P.; Wang, P.; Peng, R.; Men, R.; Gao, R.; Lin, R.; Wang, S.; Bai, S.; Tan, S.; Zhu, T.; Li, T.; Liu, T.; Ge, W.; Deng, X.; Zhou, X.; Ren, X.; Zhang, X.; Wei, X.; Ren, X.; Liu, X.; Fan, Y.; Yao, Y.; Zhang, Y.; Wan, Y.; Chu, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; Guo, Z.; and Fan, Z. 2024a. Qwen2 Technical Report. arXiv:2407.10671.

Yang, X.; Wang, X.; Zhang, Q.; Petzold, L.; Wang, W. Y.; Zhao, X.; and Lin, D. 2024b. Shadow Alignment: The Ease of Subverting Safely-Aligned Language Models. In *Proceedings of the ICLR Workshop on Secure and Trustworthy Large Language Models (SeT LLM)*.

Yu, J.; Lin, X.; Yu, Z.; and Xing, X. 2023. GPTFUZZER: Red Teaming Large Language Models with Auto-Generated Jailbreak Prompts. arXiv:2309.10253.

Zeng, Y.; Lin, H.; Zhang, J.; Yang, D.; Jia, R.; and Shi, W. 2024. How Johnny Can Persuade LLMs to Jailbreak Them: Rethinking Persuasion to Challenge AI Safety by Humanizing LLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 46595–46623.

Zhou, Z.; Yu, H.; Zhang, X.; Xu, R.; Huang, F.; Wang, K.; Liu, Y.; Fang, J.; and Li, Y. 2024. On the Role of Attention Heads in Large Language Model Safety. arXiv:2410.13708.

Zou, A.; Wang, Z.; Kolter, J. Z.; and Fredrikson, M. 2023. Universal and Transferable Adversarial Attacks on Aligned Language Models. arXiv:2307.15043.