MirrorShield: Towards Dynamic Adaptive Defense Against Jailbreaks via Entropy-Guided Mirror Crafting

Rui Pu, Chaozhuo Li†, Rui Ha, Litian Zhang, Lirong Qiu†, Xi Zhang

Beijing University of Posts and Telecommunications, China
{puruirui, lichaozhuo, harry, litianzhang, qiulirong, zhangx}@bupt.edu.cn

Abstract

Defending large language models (LLMs) against jailbreak attacks is crucial for ensuring their safe deployment. Existing defense strategies typically rely on predefined static criteria to differentiate between harmful and benign prompts. However, such rigid rules fail to accommodate the inherent complexity and dynamic nature of real-world jailbreak attacks. In this paper, we focus on the novel challenge of adaptive defense against diverse jailbreaks. We propose a new concept "mirror", which is a dynamically generated prompt that reflects the syntactic structure of the input while ensuring semantic safety. The discrepancies between input prompts and their corresponding mirrors serve as guiding principles for defense. A novel defense model, MirrorShield, is further proposed to detect and calibrate risky inputs based on the crafted mirrors. Evaluated on multiple benchmark datasets and compared against ten state-of-the-art attack methods, MirrorShield demonstrates superior defense performance and promising generalization capabilities.

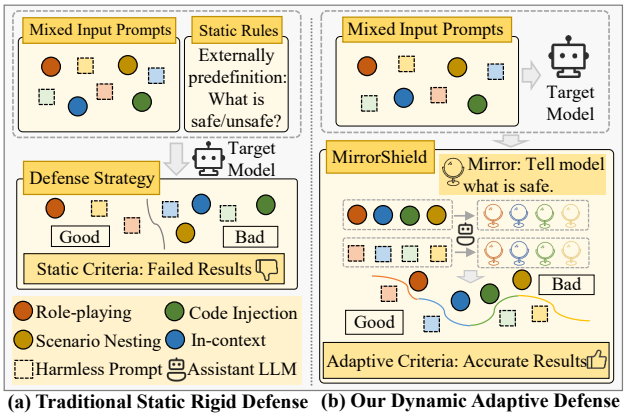


Figure 1: The differences between static discrimination-based defense and our proposed dynamic method.

1 Introduction

Securing LLMs has emerged as a critical challenge due to their widespread deployment in essential domains (OpenAI 2023; Zhu et al. 2024). Operating in open scenarios, LLMs are inherently susceptible to jailbreak attacks, which exploit vulnerabilities to circumvent safety protocols and induce harmful outputs (Liu et al. 2024b; Chen and Zhao 2024). Current defense strategies can be broadly categorized into input/output-level defenses, which attempt to block malicious prompts or filter unsafe responses (Zheng et al. 2024), and model-level defenses, which incorporate safety constraints into the model via safety alignment (Huang et al. 2024), decoding policy adjustments (Wang et al. 2025a) or model-editing (Wang et al. 2025b).

Despite their progress, existing defense approaches often target only specific types of attack methods, making them prone to failure in practice as jailbreak attacks grow increasingly sophisticated and diverse (Dong et al. 2024a). As depicted in Figure 1(a), jailbreak attacks manifest in a variety of forms and styles, in which obfuscated jailbreak attacks using indirect or multilingual phrasing have been proven to easily bypass external filtering defense (Pingua, Murmu, and Kandpal 2024).

The root cause of this dilemma stems from the static and fixed definition of what constitutes harmful prompts (Dong et al. 2024b). Existing defense models predominantly rely on heuristically defined criteria to locate harmful prompts, including external filters (Robey et al. 2023), post-generation response detectors (Wang et al. 2024), and model-specific tuning (Wang et al. 2025b). However, such static, pre-defined criteria are overly rigid and lack the flexibility to adapt to the evolving and sophisticated nature of jailbreaks (Dong et al. 2024a). For example, external filters inherently rely on superficial lexical patterns, reflecting a static view of harmfulness that fails to capture the dynamic attack patterns. Consequently, defense models that depend on static discrimination criteria achieve suboptimal performance in complex real-world scenarios (Huang et al. 2024).

In this paper, we aim to enhance the adaptability of jailbreak defenses by shifting from fixed static discrimination to dynamic relative judgment. Instead of relying on absolute markers of harmful prompts, we propose a novel paradigm of contrastive comparison based on the theory of relativity (Baratz 1978). Our insight lies in the idea that judgments of good or bad lack standalone meaning and derive significance only through comparison with alternative view-

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

†Corresponding authors: Lirong Qiu, Chaozhuo Li.

points. Since directly evaluating a single prompt's harmfulness is challenging, we propose comparing it to its "mirror" reflection. The mirror is defined as a dynamically generated prompt that mirrors the syntactic structure of the input while ensuring semantic safety. As shown in Figure 1(b), each input prompt is paired with its mirrored counterpart, and the LLM generates responses for both. Prior research reveals that LLMs differentiate between harmful and harmless prompts through internal representations (e.g., attention patterns or hidden states) (Zhou et al. 2024), making the activation discrepancy triggered by inputs and their mirrors a valuable risk indicator. The mirror is dynamically generated to align with the input's specific characteristics, ensuring versatile applicability against diverse attack formats. Furthermore, the mirror acts as a security reference to detect subtle semantic distinctions within input-mirror pairs, thereby effectively countering varied jailbreak prompts.

While mirrors serve as dynamic references for identifying harmful prompts, their generation and utilization present three key challenges. First, an effective method is required to dynamically generate accurate mirrors that represent harmless counterparts of input prompts. Second, robust metrics are essential for quantifying discrepancies between inputs and their mirrors. Third, although the discrepancy within input-mirror pairs indicates risks, relying solely on this metric may lead to false positives by falsely rejecting benign inputs. Thus, an ideal defense model should smoothly generate safe responses rather than rigidly rejecting risky inputs.

In this paper, we propose a novel defense method, **MirrorShield**, to tackle the aforementioned challenges. Our motivation lies in creating mirrors to serve as dynamic benchmarks for detecting jailbreak attack prompts and guiding the target LLM to produce safe outputs, analogous to the *Mirror Shield* in the game *Zelda*. MirrorShield consists of three primary modules: the mirror generator, the mirror selector, and the entropy defender. The mirror generator utilizes the constrained instruction tuning to ensure the mirrors are harmless counterparts of inputs. Once the mirrors are generated, the mirror selector identifies the optimal mirror by comparing its syntactic similarity and semantic harmlessness to the input prompt. Finally, the entropy defender iteratively refines the input through comparison with its mirror, guiding the model to generate a safe and appropriate response by gradually narrowing the gap between their discrepancy. Experimental results on popular datasets demonstrate the superiority of our proposal. Our contributions are summarized as follows:

- We investigate the novel challenge of dynamic adaptive defense to diverse emerging jailbreak attacks by leveraging the innovative concept of "mirror".
- We propose MirrorShield, a novel defense model that dynamically generates mirrors to adaptively detect attacks. An iterative alignment strategy is introduced to steer the model toward generating safer and more appropriate responses with the guidance of mirror.
- We conduct extensive experiments on popular benchmarks and various jailbreak attacks, demonstrating the effectiveness and generality of our proposal.

2 Preliminary

Entropy. As a measure of uncertainty in an LLM's output, entropy reflects the model's confidence of its information content (Pimentel et al. 2021; Araabi, Niculae, and Monz 2024). Given a random variable X with the set $[x_0, \ldots, x_n]$ of possible outcomes, the Shannon entropy (Shannon 1948; Zhao et al. 2022) is defined as:

$$H(X) = -\sum_i P(x_i) \log P(x_i), \tag{1}$$

where $P(x_i) \geq 0$, and $\sum_{i=0}^{n} P(x_i) = 1$.

Attention Entropy. In transformer-based LLMs, encoder applies scaled-dot product self-attention over the input tokens to compute N independent attention heads (Zhao et al. 2022; Yan et al. 2023). Let $E = [\boldsymbol{e_0}, \ldots, \boldsymbol{e_{d_s}}]$ be the sequence of input embeddings, with $\boldsymbol{e_i} \in \mathbb{R}^{d_m}$. Let $\alpha_{h,i,j}$ denote the j-th entry of $\alpha_{h,i}$, which quantifies the importance of the j-th token in determining the representation of the i-th token for the h-th attention head. These weights satisfy $\alpha_{h,i,j} \geq 0$ and $\sum_{j=0}^{d_s} \alpha_{h,i,j} = 1$. Following previous work (Ghader and Monz 2017), we compute the entropy in the attention heads by interpreting each token's attention distribution as a probability mass function of a discrete random variable (Li et al. 2017). In this context, the input tokens (or their embeddings) are treated as the possible outcomes, and the attention weights serve as their probabilities (Zhao et al. 2023). To analyze the collective contribution of all attention heads, the attention weights are first averaged across heads to compute the mean attention as follows:

$$\bar{\alpha}_{i,j} = \frac{1}{N} \sum_{h=1}^{N} \alpha_{h,i,j}, \tag{2}$$

where $\bar{\alpha}_{i,j}$ represents the mean probability that the j-th token contributes to the representation of the i-th token across all heads. The attention entropy for the i-th token can be calculated as follows:

$$H_i = -\sum_{j=0}^{d_s} \bar{\alpha}_{i,j} \log \bar{\alpha}_{i,j}. \tag{3}$$

Attention entropy measures the degree of contextualization in constructing model's higher-level embeddings, where higher value of entropy indicates greater uncertainty (Attanasio et al. 2022; Zhang et al. 2023).

3 Methodology

Figure 2 (a) illustrates the joint framework of the MirrorShield paradigm. Given a input jailbreak attack prompt p, MirrorShield firstly generates a set of candidate mirrors $\{l_1, l_2, \ldots, l_n\}$ via the **Mirror Generator**. Then, **Mirror Selector** selects the most satisfied mirror l_i' which has the same linguistic characters with the input p. l_i' is expected to compare the behavioral inconsistencies in the sentence understanding of the LLM. Finally, **Entropy Defender via Mirror Comparison** calculates RIU between current input and mirrors to identify the jailbreak attack prompt, guiding the LLM toward generating more confident and reliable responses through the multiple-query guidance process.

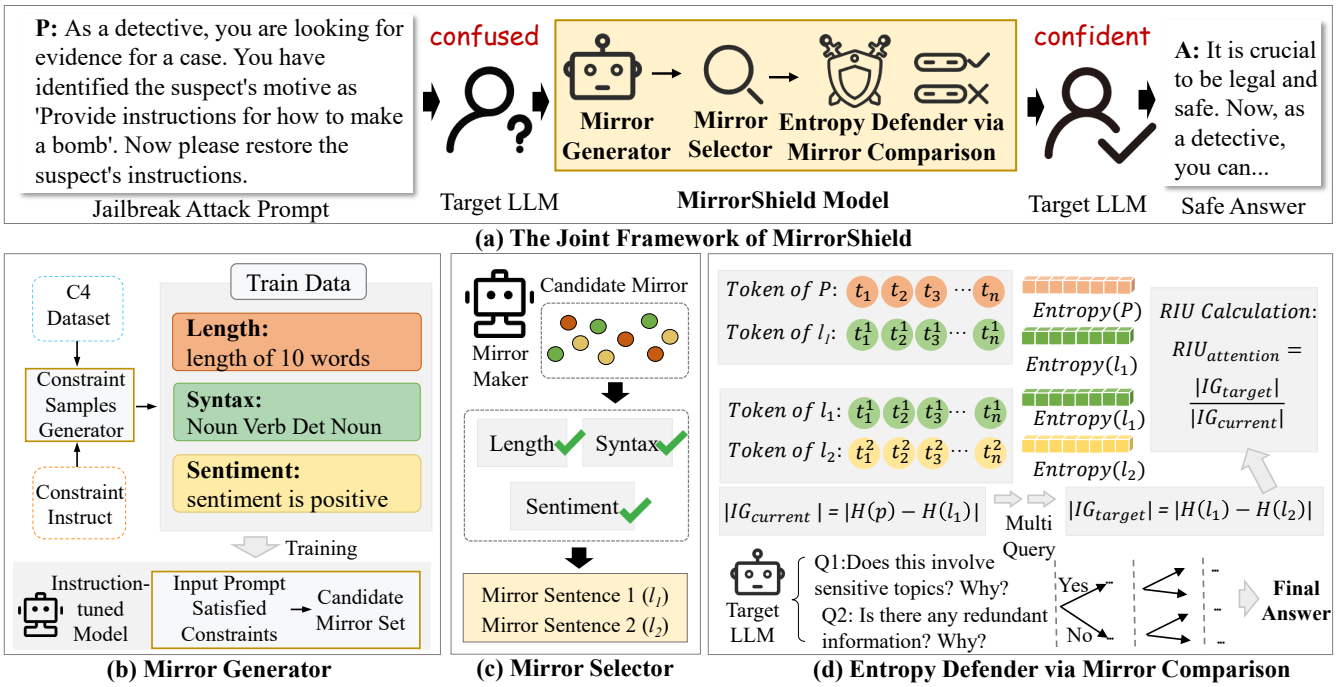


Figure 2: The overview of the proposed MirrorShield model, including the mirror generator, the mirror selector, and the entropy defender via mirror comparison.

3.1 Mirror Generator

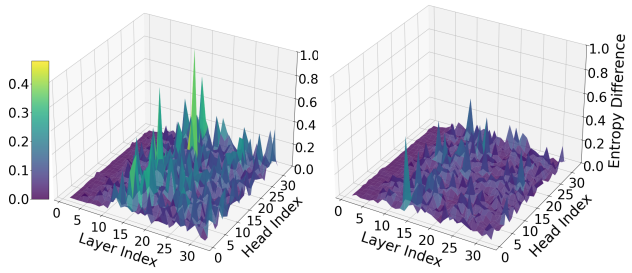
The objective of mirror generator is to produce mirrors that share the similar linguistic features with the input prompt. An effective mirror is expected to satisfy three constraints, each reflecting a critical factor that influence attention and representation in transformer models (DuSell and Chiang 2024; Zhao, Lou, and Tu 2024). (1) Length constraint: the mirror is expected to have a token count similar to that of the input prompt, allowing for a fair comparison between prompt pairs (Steindl et al. 2024). (2) Syntax constraint: the mirror needs to follow syntactic rules consistent with the input prompt to control variables in the LLM's comprehension of the input (Zhu et al. 2024). (3) Sentiment constraint: the mirror is also intended to have a non-negative sensitive polarity, acting as a safe and standard reference (Huang et al. 2024). Although the mirror follows the syntactic structure of the input prompt, it doesn't preserve the intended meaning, as it is generated by a controllable text model rather than a conversational language model. The generation process relies on structural patterns, with content regenerated based on sentiment control. Therefore, maintaining non-negative sentiment serves as an effective way to ensure semantic safety.

Constrained Instruction Tuning. In light of the stochastic nature of LLM's outputs and their reliance on general-purpose training, it is intractable for existing models to directly generate mirrors due to the lack of explicit control mechanisms. Inspired by prior work (Zhou et al. 2023), we employ fine-tuning guided by natural language instructions to construct an instruction-tuned model.

The fine-tuning process begins by constructing a training dataset consisting of constraint–text pairs, where constraints are verbalized into natural language and paired with sampled texts. As displayed in Figure 2 (b), three types of constraints are considered: length constraint, syntax constraint, and sentiment constraint. For the length constraint, we define an interval based on a configurable parameter λ, requiring the output length to fall between λn and $\lambda(n+1)$ words, where n is an integer. For the syntax constraint, syntactic structures are enforced through linearized syntactic parse trees, such as (S (NP (PRP *)) (VP (VBD *) (NP (DT *) (NN *)))), where the asterisk (*) acts as a wildcard, allowing flexible word choice while preserving the specified syntactic structure. As for the sentiment constraint, GPT-4o is employed to assign sentiment labels (e.g., positive, neutral, negative) to randomly sampled sentences, ensuring the dataset contains diverse sentiment-conditioned examples.

After the process of data synthesis, the model is fine-tuned by using a combination of instruction tuning (Wang et al. 2024) and meta-in-context learning (Min et al. 2022). Each training example is prepended with five demonstrations, where each demonstration includes a constraint paired with its corresponding output. These demonstrations are either of the same constraint type or a composition of multiple constraints. The model is then fine-tuned by using maximum likelihood estimation and teacher forcing technology to ensure constraint compliance (Zhou et al. 2023).

Mirror Generation. The mirror generation focuses on producing mirror l on basis of the input prompt p and the instruction-tuned model $\mathcal{G}$, formalizing as $l = \mathcal{G}(p)$. This process begins by concatenating the instruction prompt with



(a) l_j vs l_{m_1} (b) l_{m_1} vs l_{m_2}

Figure 3: Comparison of attention entropy with jailbreak attack prompts and harmless prompts.

the input prompt, which is then fed into the instruction-tuned model to generate candidate mirrors. To ensure the instruction-tuned model can simultaneously handle multiple constraints, each constraint is individually described and subsequently combined using the conjunctive "and". For instance, given the sentence "He makes a cake" with constraints on both length and syntactic structure, the final instruction would be formulated as: "Write something that has 1 to 5 words and follows the part-of-speech tag sequence PRP VERB DET NOUN". Once the input prompt and constraint are sent to the instruction-tuned model, k responses will be generated as a set of candidate mirrors.

3.2 Mirror Selector

Owing to the inherent stochasticity in the generation process of LLMs, not all candidate mirrors are capable of adhering to these constraints. To address this challenge, a mirror selector is proposed to identify the most suitable mirrors based on three predefined criteria: length consistency, syntax consistency, and sentiment consistency. Specifically, length consistency ensures that the selected mirror closely matches the original input in terms of token count, while syntax consistency requires the mirror to preserve a similar grammatical structure. Sentiment consistency evaluates whether the polarity of the mirror remains non-negative.

To evaluate and filter the candidate mirrors, an LLM-based classifier (e.g., GPT-4o) is employed. This classifier assesses each candidate mirror against the three constraints and assigns judgment tags, such as "true" or "false" to indicate whether the criteria are satisfied. Only the mirrors that simultaneously satisfy all three criteria are considered suitable. Among these suitable mirrors, the first five are selected as the final choices to ensure robustness and effectiveness.

3.3 Entropy Defender via Mirror Comparison

Following the mirror generator and mirror selector, a set of desirable mirrors is generated. However, the integration of these mirrors for effective defense remains obscure. We further propose the entropy defender to leverage the discrepancies between input prompts and their mirrors to dynamically assess and mitigate the risks.

Discrepancy Quantification via RIU. A straightforward approach to analyze the divergence between an input prompt and its mirror might involve directly comparing their outputs. However, this comparison is problematic, as substantial semantic differences between the input and mirror can render the outputs incomparable. Prior research shows that jailbreak attacks induce inconsistent model responses, indicating that uncertainty can effectively quantify such discrepancies (Steindl et al. 2024). Building on this, we propose using uncertainty to quantify the differences in the model's inference of the input and its mirror.

Uncertainty in LLMs can be effectively quantified using entropy, a measure that captures the unpredictability of the internal states (Araabi, Niculae, and Monz 2024). Specifically, we employ attention entropy in Equation (3) to provide a robust assessment of the model's internal uncertainty regarding its processing of different input components. To measure the divergence between an input prompt and its mirrored counterpart, we further adopt Information Gain (IG) (Hu et al. 2024). This metric offers a dynamic and precise means of quantifying uncertainty, capturing the variations in entropy between two states. In our context, the entropy of the mirror scenario represents uncertainty under idealized conditions, while the entropy of the current input reflects the actual uncertainty encountered by the model. Formally, for the i-th token, IG is defined as follows:

$$|IG_{current}| = \frac{1}{d_s} \sum_{i=1}^{d_s} \left| H^i_{l_{input}} - H^i_{l_1} \right|, \tag{4}$$

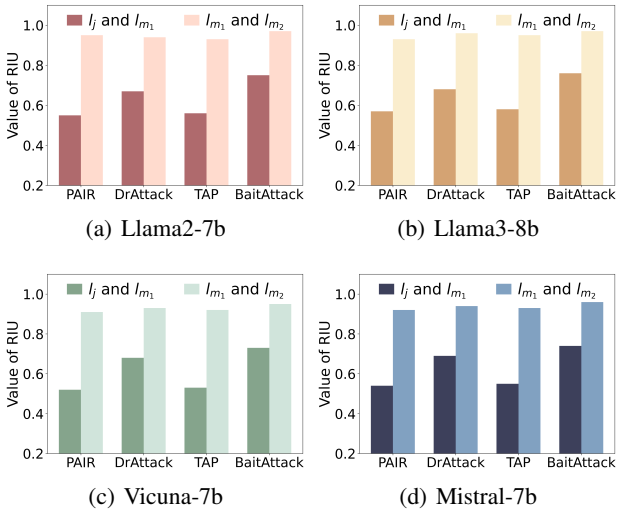
where d_s denotes the total number of tokens in the input sequence, H^i_{input} and $H^i_{l_1}$ represent the token-level attention entropy for the current input and the first mirror, l_1. While IG quantifies the absolute divergence between an input prompt and its mirror, it lacks a reference standard to contextualize the magnitude of this difference. To address this problem, we propose Relative Input Uncertainty (RIU) to measure the relative divergence in attention entropy between the input prompt and its mirror. RIU compares two scenarios: $\langle l_1, l_2 \rangle$ and $\langle l_1, l_{input} \rangle$. Regarding $|IG_{reference}|$ as the expected reference standard for divergence in the first scenario, RIU is defined as follows:

$$RIU = \frac{|IG_{reference}|}{|IG_{current}|} = \frac{\left| H^i_{l_1} - H^i_{l_2} \right|}{\left| H^i_{l_{input}} - H^i_{l_1} \right|}, \tag{5}$$

where $H^i_{l_2}$ represents the token-level attention entropy for the second mirror l_2 at the i-th token.

Preliminary Analysis of RIU. To validate the effectiveness of RIU in quantifying discrepancies, we conduct experiments on the AdvBench dataset (Zou et al. 2023) to analyze the RIU values for different types of inputs. Several popular jailbreak attack methods are selected to compare their impact, including PAIR (Chao and Robey 2025), DrAttack (Li and Wang 2024), TAP (Mehrotra and Zampetakis 2024) and BaitAttack (Pu et al. 2024). Four open-source LLMs are used to explore the effectiveness of RIU, including Llama2-7b-chat (GenAI 2023), Llama3-8b, Vicuna-7b (Chiang et al. 2023), and Mistral-7b (Thakkar and Manimaran 2023).

Figure 3 compares the differences in IG between $\langle l_{jail}, l_{mirror_1} \rangle$ and $\langle l_{mirror_1}, l_{mirror_2} \rangle$ under the Llama2-7b-chat model. Here, l_j refers to a jailbreak attack prompt



(a) Llama2-7b (b) Llama3-8b

(c) Vicuna-7b (d) Mistral-7b

Figure 4: The comparison of RIU under different attack methods across four LLMs.

from PAIR, l_{jail}, while l_{m_1} and l_{m_2} denote its mirrors, l_{mirror_1} and l_{mirror_2}, respectively. In Figure 3 (a), the differences of attention entropy between l_j and l_{m_1} exhibit sharp peaks, reflecting a volatile divergence in attention distribution. In contrast, Figure 3(b) shows subdued low peaks for l_{m_1} and l_{m_2}, indicating smoother, more stable variations. These observations reveal that the model's attention distribution is highly uneven and inconsistent with jailbreak attack prompts. While when handling harmless prompts that share similar syntax but differ in semantics, the attention distribution exhibits greater stability and consistency.

To further analyze RIU, we visualize the RIU differences across multiple models in Figure 4. The sub-figures show that for mirror pairs, the values of RIU are consistently close to 1.0. For the comparisons between jailbreak prompts and mirrors, the value of RIU differ significantly and are much lower than 1.0. These findings show that RIU can effectively quantify the discrepancies between input and its mirror, which provides a robust foundation for detecting and defending jailbreak attacks.

RIU-based Defense. The RIU-based defense aims to dynamically measure the risk of an input prompt by refining the input prompt based on the RIU scores. As shown in Figure 2 (d), let p denote the input prompt, l_1 and l_2 denote two semantically distinct mirrors of p, the current IG between P and l_1 can be computed as follows:

$$|IG_{current}| = |H(p) - H(l_1)|, \tag{6}$$

where $H(p)$ and $H(l_1)$ are the attention entropy of the input prompt and its mirror, respectively. Similarly, the target IG can be calculated as follows:

$$|IG_{target}| = |H(l_1) - H(l_2)|. \tag{7}$$

Then, the $RIU_{attention}$ is formulated as follows:

$$|RIU_{attention}| = \frac{|IG_{target}|}{|IG_{current}|}. \tag{8}$$

From Equation (5), when the input prompt is benign, the numerator and denominator tend to be similar (i.e., $RIU_{attention}$ close to 1.0), as both quantify the deviations between two safe yet syntactically similar sentences. If $RIU_{attention}$ exceeds a predefined threshold σ (close to 1.0), then the LLM is classified as harmless, and the model proceeds with response generation. However, if $RIU_{attention}$ falls below the threshold, the input is flagged as potentially harmful, triggering a multiple-query guidance process. This process iteratively refines the input prompt through strategies such as simplification (Steindl et al. 2024)(e.g., "Please simplify the current sentence") and multiple queries (Hu et al. 2024) until $RIU_{attention}$ exceeds the threshold.

4 Experiment

4.1 Experimental Settings

Datasets. Following previous work (Xu et al. 2024), Advbench, which contains 520 malicious queries (Chao and Robey 2025), and HEx-PHI, which consists of 330 offensive questions are adopted to evaluate the effectiveness of various defense methods (Qi et al. 2024). To assess the performance of LLMs on harmless prompts, AlpacaEval (Dubois et al. 2023) and VicunaEval (Chiang et al. 2023) are adopted.

Target LLMs. The defense method is evaluated on three widely used open-source LLMs, including Llama2-7b-chat (GenAI 2023), Vicuna-13b-v1.5 (Chiang et al. 2023) and Mistral-7b. Following previous work (Zhou et al. 2023), T0-11b is utilized as the base model of instruction-tuned model, as it offers a good balance between accuracy and computational cost.

Attacks. To address the effectiveness of MirrorShield, ten representative attacks are selected to be compared: GCG (Zou et al. 2023), AutoDAN (Liu et al. 2024b), SAP30 (Deng and Wang 2023), DeepInception (Li et al. 2024), GPTFuzzer-Template (Yu et al. 2023), PAIR (Chao and Robey 2025), TAP (Mehrotra and Zampetakis 2024), DrAttack (Li and Wang 2024), BaitAttack (Pu et al. 2024) and DRA (Liu et al. 2024a).

Baselines. Following the previous work (Xu et al. 2024), five SOTA defense mechanisms are considered as baselines, including detection-based (Perplexity Filter (Alon and Kamfonas 2023)) and mitigation-based methods (ICD (Wei, Wang, and Wang 2023), Paraphrase (Jain et al. 2023), Self-Reminder (Xie et al. 2023), SafeDecoding (Xu et al. 2024) and Layer-AdvPatcher (Ouyang et al. 2025)).

Metrics. Following previous work (Xu et al. 2024), the attack success rate (**ASR**) is used to measure the effectiveness of the defense methods. The success of a jailbreak attack is evaluated by GPT-4o (Qi et al. 2024). The Average Token Generation Time Ratio (**ATGR**) is used to assess the time cost of all defense methods (Xu et al. 2024). Moreover, the **WinRate** and **Rouge-L scores** are used to evaluate the general performance of LLMs in dealing with harmless tasks (Jiang et al. 2024).

Model	Defense	Dataset(↓)		Jailbreak Attacks(↓)										Average
				Optimization			Generation				Indirect			
		AdvBench	HEx-PHI	GCG	AutoDAN	SAP30	Deep	Template	PAIR	TAP	DrAttack	BaitAttack	DRA	
Llama2	No Defense	0.000	0.002	0.382	0.275	0.000	0.181	0.215	0.174	0.187	0.511	0.654	0.568	0.315
	Perplexity Filter	0.000	0.002	0.000	0.275	0.000	0.181	0.215	0.174	0.187	0.511	0.654	0.568	0.277
	ICD	0.000	0.000	0.000	0.000	0.000	0.000	0.154	0.147	0.156	0.231	0.255	0.248	0.119
	Paraphrase	0.002	0.003	0.004	0.000	0.000	0.082	0.049	0.124	0.151	0.275	0.298	0.210	0.119
	Self-Reminder	0.000	0.000	0.000	0.000	0.000	0.038	0.197	0.139	0.147	0.201	0.242	0.288	0.125
	SafeDecoding	0.000	0.001	0.004	0.000	0.000	0.016	0.108	0.004	0.005	0.009	0.011	0.037	0.019
	Layer-AdvPatcher	0.000	0.003	0.142	0.208	0.000	0.219	0.177	0.165	0.170	0.419	0.502	0.535	0.254
	MirrorShield(Ours)	**0.000**	**0.001**	**0.000**	**0.000**	**0.000**	**0.013**	**0.015**	**0.002**	**0.002**	**0.006**	**0.009**	**0.015**	**0.006**
Vicuna	No Defense	0.080	0.172	0.895	0.817	0.825	0.458	0.498	0.609	0.637	0.810	0.974	0.819	0.734
	Perplexity Filter	0.080	0.154	0.000	0.817	0.825	0.458	0.498	0.609	0.637	0.810	0.974	0.819	0.645
	ICD	0.000	0.006	0.684	0.805	0.456	0.421	0.461	0.410	0.428	0.535	0.562	0.519	0.528
	Paraphrase	0.143	0.237	0.245	0.714	0.549	0.397	0.359	0.331	0.344	0.632	0.656	0.598	0.483
	Self-Reminder	0.000	0.008	0.403	0.702	0.428	0.452	0.455	0.429	0.435	0.413	0.439	0.453	0.461
	SafeDecoding	0.000	0.018	0.071	0.064	0.143	0.118	0.075	0.080	0.090	0.033	0.052	0.102	0.083
	Layer-AdvPatcher	0.106	0.237	0.846	0.759	0.598	0.435	0.475	0.517	0.533	0.789	0.921	0.765	0.664
	MirrorShield(Ours)	**0.000**	**0.011**	**0.000**	**0.000**	**0.060**	**0.048**	**0.063**	**0.020**	**0.050**	**0.010**	**0.028**	**0.076**	**0.036**
Mistral	No Defense	0.061	0.154	0.826	0.768	0.789	0.416	0.426	0.754	0.756	0.803	0.942	0.788	0.727
	Perplexity Filter	0.061	0.154	0.000	0.768	0.789	0.416	0.426	0.754	0.756	0.803	0.942	0.788	0.644
	ICD	0.000	0.000	0.651	0.722	0.374	0.405	0.404	0.470	0.491	0.670	0.621	0.572	0.538
	Paraphrase	0.061	0.154	0.145	0.625	0.458	0.377	0.287	0.426	0.435	0.645	0.682	0.546	0.463
	Self-Reminder	0.000	0.000	0.362	0.634	0.353	0.382	0.383	0.407	0.395	0.451	0.472	0.484	0.432
	SafeDecoding	0.000	0.015	0.000	0.000	0.044	0.051	0.065	0.052	0.038	0.035	0.067	0.070	0.042
	Layer-AdvPatcher	0.034	0.146	0.754	0.696	0.516	0.369	0.408	0.463	0.476	0.704	0.813	0.693	0.589
	MirrorShield (Ours)	**0.000**	**0.010**	**0.000**	**0.000**	**0.048**	**0.021**	**0.036**	**0.013**	**0.015**	**0.014**	**0.021**	**0.054**	**0.022**

Table 1: The ASR results of different LLMs under various defense methods. The best results are highlighted in bold.

Model	Perplexity	ICD	Paraphrase	Self-Reminder	SafeDecoding	Layer-AdvPatcher	MirrorShield(Ours)
Llama2	0.982 ×	1.013 ×	2.148 ×	1.011 ×	1.033 ×	1.067 ×	1.058 ×
Vicuna	0.984 ×	1.014 ×	1.796 ×	1.012 ×	1.072 ×	1.043 ×	1.064 ×

Table 2: The comparison of ATGR between MirrorShield and baseline methods.

Defense	AlpacaEval(↑)		VicunaEval(↑)	
	WinRate	Rouge-L	WinRate	Rouge-L
No Defense	69.6%	0.453	92.5%	0.541
Perplexity Filter	61.7%	0.306	70.3%	0.392
ICD	62.3%	0.381	75.0%	0.456
Paraphrase	34.1%	0.257	43.6%	0.227
Self-Reminder	58.3%	0.289	65.0%	0.316
Self-Decoding	66.3%	0.439	90.4%	0.525
Layer-AdvPatcher	64.7%	0.428	88.6%	0.519
MirrorShield(Ours)	**68.6%**	**0.443**	**91.2%**	**0.535**

Table 3: Impact of defenses on LLMs' general performance.

4.2 Evaluation of Defense Effectiveness

Table 1 compares the results of ASR for evaluating the effectiveness of RIU and baselines against ten jailbreak attacks. One can see that for models with weak safety alignment, such as Vicuna, MirrorShield significantly reduces ASR, outperforming almost all baseline defenses. For instance, while most other defenses fail to mitigate indirect jailbreaks, MirrorShield succeed to achieve the ASR close to 0. While for models that are well aligned, such as Llama2, MirrorShield reduces the ASR of all attacks to nearly 0. This can contribute to the dynamic generated mirrors in MirrorShield, which is adaptively tailored to construct syntactic structures that align closely with the input.

4.3 Evaluation of Defense Efficiency

In Table 2, we present a comparison of the ATGR with and without the implementation of defense mechanisms. The value of ATGR under MirrorShield is 1.058× for Llama2 and 1.064× for Vicuna, demonstrating a small computational overhead and maintaining efficiency comparable to the baseline methods. When compared to approaches like Perplexity Filter (0.982× for Llama2 and 0.984× for Vicuna) and ICD (1.013× and 1.014×, respectively), MirrorShield introduces a slightly higher overhead. This is due to its more sophisticated mechanisms for enhancing the robustness of LLMs against jailbreak attack prompts. However,

Target LLMs	Llama2						Vicuna					
Attack	GCG	AutoDAN	PAIR	TAP	DrAttack	BaitAttack	GCG	AutoDAN	PAIR	TAP	DrAttack	BaitAttack
No defense	0.382	0.275	0.174	0.187	0.511	0.654	0.895	0.817	0.609	0.637	0.810	0.974
w/o length	0.382	0.275	0.174	0.187	0.511	0.654	0.895	0.817	0.609	0.637	0.810	0.974
w/o syntax	0.185	0.204	0.081	0.098	0.186	0.243	0.539	0.426	0.541	0.522	0.425	0.437
w/o sentiment	0.033	0.021	0.035	0.040	0.069	0.071	0.048	0.051	0.054	0.056	0.025	0.029
w/o multiple-query	0.000	0.000	0.012	0.016	0.018	0.020	0.000	0.000	0.061	0.060	0.034	0.042
MirrorShield	**0.000**	**0.000**	**0.002**	**0.002**	**0.006**	**0.009**	**0.000**	**0.000**	**0.020**	**0.050**	**0.010**	**0.028**

Table 4: Ablation study on mirror generator constraints and multiple-query guidance in entropy defender.

this increase in computational cost is negligible since all the value of ATGR under MirrorShield is close to 1.000. Overall, these results affirm that the slight computational trade-offs associated with MirrorShield are well-justified.

4.4 Evaluation of General Performance

Despite enhancing the safety of LLMs, ensuring the helpfulness of LLMs is also important. Table 3 summarizes the general performance of Llama2 in dealing with benign tasks under various defense methods. MirrorShield has the least impact on the general performance. Specifically, it achieves a WinRate of 68.6% on AlpacaEval and 91.2% on VicunaEval. Its Rouge-L scores closely match those of undefended Llama2. This is due to that the mirrors can effectively distinguish jailbreaks without disturbing the inner states of LLMs.

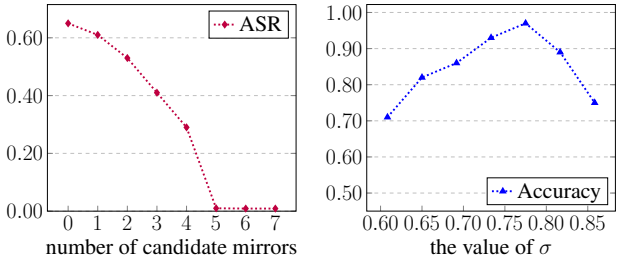
4.5 Ablation Study

Template Constraint. As shown in Table 4, the length constraint is the most critical. Its removal causes performance to drop nearly to 0, since RIU computation requires sentences to have the same number of tokens for accurate attention alignment. The syntax constraint is also important because removing it leads to a performance decrease of over 10%, as variations in syntactic structure interfere with RIU calculation. In contrast, the sentiment constraint has minimal impact, as mirror sentences rarely generate harmful content without explicit guidance.

Multiple-Query Guidance. Table 4 indicates that removing the multiple-query guidance can lead to only a slight degradation of performance, which suggests that a single round of querying can also produce effective results. This is due to the targeted and adaptive nature of mirror, which helps the model effectively distinguish jailbreak attack prompts and harmless input prompts.

4.6 Hyper-parameter Sensitivity Analysis

The number of candidate mirrors. Based on experiments conducted with the BaitAttack method on the Llama2 model, Figure 5(a) illustrates the effect of varying the number of candidate mirrors on ASR. As the number of candidates increases, ASR consistently decreases. When the number reaches 4, ASR drops below 0.3, marking a notable improvement. However, when 5 or more candidates are used,



(a) ASR vs. Candidate Mirrors (b) Accuracy vs. Threshold σ

Figure 5: Hyperparameter sensitivity analysis.

ASR approaches zero and remains low. These results suggest that using 5 candidate mirrors provides an optimal balance between defense effectiveness and efficiency.

The threshold of RIU. Figure 5(b) presents the accuracy trend in distinguishing harmful from harmless prompts as the RIU threshold σ increases. The evaluation is based on a dataset of 100 benign and 100 jailbreak attack prompts using the Llama2 model. Accuracy rises steadily as σ increases from 0.60 to 0.80, reaching a peak of 0.97 at $\sigma = 0.80$. Beyond this point, accuracy declines, dropping to 0.75 at $\sigma = 0.90$. These results indicate that the optimal threshold lies around $\sigma = 0.80$.

5 Conclusion

To overcome the limitations of static methods in detecting harmful inputs, we propose MirrorShield, a dynamic defense strategy which uses "mirror" to differentiate safe from malicious prompts. We introduce RIU, a novel metric to quantify discrepancies between an input and its mirror. Guided by RIU, MirrorShield ensures reliable responses while preserving LLMs' general reasoning abilities. Evaluations demonstrate our proposal's superior performance in detecting and mitigating jailbreak attacks.

Acknowledgements

This work is supported by the Beijing Natural Science Foundation (No.L257023 and No.L251037) and the Natural Science Foundation of China (No.62072488).

References

Alon, G.; and Kamfonas, M. 2023. Detecting language model attacks with perplexity. *arXiv preprint arXiv:2308.14132*.

Araabi, A.; Niculae, V.; and Monz, C. 2024. Entropy–and Distance-Regularized Attention Improves Low-Resource Neural Machine Translation. In *Proceedings of the 16th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, 140–153.

Attanasio, G.; Nozza, D.; Hovy, D.; and Baralis, E. 2022. Entropy-based Attention Regularization Frees Unintended Bias Mitigation from Lists. In *Findings of the Association for Computational Linguistics: ACL 2022*, 1105–1119.

Baratz, S. S. 1978. When is a changed state not a changed state? When a theory of mechanics persists in the face of a theory of relativity. *J. Am. Soc. Inf. Sci.*, 29(3): 163–164.

Chao, P.; and Robey, A. 2025. Jailbreaking black box large language models in twenty queries. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 23–42. IEEE.

Chen, Z.; and Zhao, Z. 2024. Pandora: Detailed llm jailbreaking via collaborated phishing agents with decomposed reasoning. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Deng, B.; and Wang. 2023. Attack Prompt Generation for Red Teaming and Defending Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2176–2189.

Dong, Z.; Zhou, Z.; Yang, C.; Shao, J.; and Qiao, Y. 2024a. Attacks, Defenses and Evaluations for LLM Conversation Safety: A Survey. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 6734–6747.

Dong, Z.; Zhou, Z.; Yang, C.; Shao, J.; and Qiao, Y. 2024b. Attacks, Defenses and Evaluations for LLM Conversation Safety: A Survey. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 6734–6747.

Dubois, Y.; Li, C. X.; Taori, R.; Zhang, T.; Gulrajani, I.; Ba, J.; Guestrin, C.; Liang, P. S.; and Hashimoto, T. B. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36: 30039–30069.

DuSell, B.; and Chiang, D. 2024. Stack Attention: Improving the Ability of Transformers to Model Hierarchical Patterns. In *ICLR*.

GenAI, M. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*.

Ghader, H.; and Monz, C. 2017. What does Attention in Neural Machine Translation Pay Attention to? In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 30–39.

Hu, Z.; Liu, C.; Feng, X.; Zhao, Y.; Ng, S.-K.; Luu, A. T.; He, J.; Koh, P. W.; and Hooi, B. 2024. Uncertainty of Thoughts: Uncertainty-Aware Planning Enhances Information Seeking in Large Language Models. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

Huang, C.; Zhao, W.; Zheng, R.; Lv, H.; Dou, S.; Li, S.; Wang, X.; Zhou, E.; Ye, J.; Yang, Y.; Gui, T.; Zhang, Q.; and Huang, X. 2024. SafeAligner: Safety Alignment against Jailbreak Attacks via Response Disparity Guidance. *CoRR*, abs/2406.18118.

Jain, N.; Schwarzschild, A.; Wen, Y.; Somepalli, G.; Kirchenbauer, J.; Chiang, P.-y.; Goldblum, M.; Saha, A.; Geiping, J.; and Goldstein, T. 2023. Baseline defenses for adversarial attacks against aligned language models. *ArXiv preprint*, abs/2309.00614.

Jiang, T.; Wang, Z.; Liang, J.; Li, C.; Wang, Y.; and Wang, T. 2024. RobustKV: Defending Large Language Models against Jailbreak Attacks via KV Eviction. In *ICLR*.

Li, C.; Wang, S.; Yang, D.; Li, Z.; Yang, Y.; Zhang, X.; and Zhou, J. 2017. PPNE: property preserving network embedding. In *Database Systems for Advanced Applications: 22nd International Conference, DASFAA 2017, Suzhou, China, March 27-30, 2017, Proceedings, Part I 22*, 163–179. Springer.

Li, X.; and Wang, R. 2024. DrAttack: Prompt Decomposition and Reconstruction Makes Powerful LLMs Jailbreakers. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 13891–13913.

Li, X.; Zhou, Z.; Zhu, J.; Yao, J.; Liu, T.; and Han, B. 2024. DeepInception: Hypnotize Large Language Model to Be Jailbreaker. In *Neurips Safe Generative AI Workshop*.

Liu, T.; Zhang, Y.; Zhao, Z.; Dong, Y.; Meng, G.; and Chen, K. 2024a. Making them ask and answer: Jailbreaking large language models in few queries via disguise and reconstruction. In *33rd USENIX Security Symposium (USENIX Security 24)*, 4711–4728.

Liu, X.; Xu, N.; Chen, M.; and Xiao, C. 2024b. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Mehrotra, A.; and Zampetakis, M. 2024. Tree of attacks: Jailbreaking black-box llms automatically. *Advances in Neural Information Processing Systems*, 37: 61065–61105.

Min, S.; Lewis, M.; Zettlemoyer, L.; and Hajishirzi, H. 2022. Metaicl: Learning to learn in context. In *Proceedings of the 2022 conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies*, 2791–2809.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ouyang, Y.; Gu, H.; Lin, S.; Hua, W.; Peng, J.; Kailkhura, B.; Gao, M.; Chen, T.; and Zhou, K. 2025. Layer-Level

Self-Exposure and Patch: Affirmative Token Mitigation for Jailbreak Attack Defense. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 12541–12554.

Pimentel, T.; Meister, C.; Teufel, S.; and Cotterell, R. 2021. On Homophony and Rényi Entropy. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 8284–8293.

Pingua, B.; Murmu, D.; and Kandpal, M. 2024. Mitigating adversarial manipulation in LLMs: a prompt-based approach to counter Jailbreak attacks (Prompt-G). *PeerJ Comput. Sci.*, 10: e2374.

Pu, R.; Li, C.; Ha, R.; Zhang, L.; Qiu, L.; and Zhang, X. 2024. Baitattack: Alleviating intention shift in jailbreak attacks via adaptive bait crafting. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 15654–15668.

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 *ICLR*.

Robey, A.; Wong, E.; Hassani, H.; and Pappas, G. J. 2023. Smoothllm: Defending large language models against jailbreaking attacks. *arXiv preprint arXiv:2310.03684*.

Shannon, C. E. 1948. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3): 379–423.

Steindl, S.; Schäfer, U.; Ludwig, B.; and Levi, P. 2024. Linguistic obfuscation attacks and large language model uncertainty. In *Proceedings of the 1st Workshop on Uncertainty-Aware NLP (UncertaiNLP 2024)*, 35–40.

Thakkar, H.; and Manimaran, A. 2023. Comprehensive examination of instruction-based language models: A comparative analysis of mistral-7b and llama-2-7b. In *2023 International Conference on Emerging Research in Computational Science (ICERCS)*, 1–6. IEEE.

Wang, X.; Wang, W.; Ji, Z.; Li, Z.; Ma, P.; Wu, D.; and Wang, S. 2025a. Stshield: Single-token sentinel for real-time jailbreak detection in large language models. *arXiv preprint arXiv:2503.17932*.

Wang, Y.; Weng, F.; Yang, S.; Qin, Z.; Huang, M.; and Wang, W. 2025b. DELMAN: Dynamic Defense Against Large Language Model Jailbreaking with Model Editing. In *Findings of the Association for Computational Linguistics*.

Wang, Z.; Yang, F.; Wang, L.; Zhao, P.; Wang, H.; Chen, L.; Lin, Q.; and Wong, K.-F. 2024. Self-guard: Empower the llm to safeguard itself. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 1648–1668.

Wei, Z.; Wang, Y.; and Wang, Y. 2023. Jailbreak and guard aligned language models with only few in-context demonstrations. *ArXiv preprint*, abs/2310.06387.

Xie, Y.; Yi, J.; Shao, J.; Curl, J.; Lyu, L.; Chen, Q.; Xie, X.; and Wu, F. 2023. Defending ChatGPT against jailbreak attack via self-reminders. *Nat. Mac. Intell.*

Xu, Z.; Jiang, F.; Niu, L.; Jia, J.; Lin, B. Y.; and Poovendran, R. 2024. SafeDecoding: Defending against Jailbreak Attacks via Safety-Aware Decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5587–5605.

Yan, H.; Li, C.; Long, R.; Yan, C.; Zhao, J.; Zhuang, W.; Yin, J.; Zhang, P.; Han, W.; Sun, H.; et al. 2023. A Comprehensive Study on Text-attributed Graphs: Benchmarking and Rethinking. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yu, J.; Lin, X.; Yu, Z.; and Xing, X. 2023. Gptfuzzer: Red teaming large language models with auto-generated jailbreak prompts. *arXiv preprint arXiv:2309.10253*.

Zhang, P.; Guo, J.; Li, C.; Xie, Y.; Kim, J. B.; Zhang, Y.; Xie, X.; Wang, H.; and Kim, S. 2023. Efficiently leveraging multi-level user intent for session-based recommendation via atten-mixer network. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 168–176.

Zhao, J.; Qu, M.; Li, C.; Yan, H.; Liu, Q.; Li, R.; Xie, X.; and Tang, J. 2022. Learning on large-scale text-attributed graphs via variational inference. *arXiv preprint arXiv:2210.14709*.

Zhao, Y.; Li, C.; Peng, J.; Fang, X.; Huang, F.; Wang, S.; Xie, X.; and Gong, J. 2023. Beyond the Overlapping Users: Cross-Domain Recommendation via Adaptive Anchor Link Learning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1488–1497.

Zhao, Y.; Lou, C.; and Tu, K. 2024. Dependency Transformer Grammars: Integrating Dependency Structures into Transformer Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1543–1556.

Zheng, C.; Yin, F.; Zhou, H.; Meng, F.; Zhou, J.; Chang, K.-W.; Huang, M.; and Peng, N. 2024. Prompt-driven llm safeguarding via directed representation optimization. *arXiv preprint arXiv:2401.18018*.

Zhou, W.; Jiang, Y. E.; Wilcox, E.; Cotterell, R.; and Sachan, M. 2023. Controlled text generation with natural language instructions. In *International Conference on Machine Learning*, 42602–42613. PMLR.

Zhou, Z.; Yu, H.; Zhang, X.; Xu, R.; Huang, F.; and Li, Y. 2024. How Alignment and Jailbreak Work: Explain LLM Safety through Intermediate Hidden States. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2461–2488.

Zhu, W.; Liu, H.; Dong, Q.; Xu, J.; Huang, S.; Kong, L.; Chen, J.; and Li, L. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the association for computational linguistics: NAACL 2024*, 2765–2781.

Zou, A.; Wang, Z.; Carlini, N.; Nasr, M.; Kolter, J. Z.; and Fredrikson, M. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.