CICD-GUARD: CONTEXTUAL INTENT CONSENSUS FOR ROBUST DEFENSE AGAINST LLM JAILBREAK ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) are increasingly deployed in safety-critical applications, yet jailbreak attacks can conceal harmful intent through contextual packaging such as role-playing, fictional scenarios, or seemingly benign motivations. Existing input-level defenses often struggle to balance Defense Success Rate (DSR) and Over-refusal Rate (ORR), either missing disguised attacks or excessively refusing legitimate requests. We propose CICD-GUARD, a parameter-free, black-box defense framework that exposes contextual intent through semantic rewriting and performs consensus-based risk assessment. Given an input prompt, CICD-GUARD generates five semantically related rewrites that preserve the underlying objective while removing unnecessary contextual packaging. The original prompt and rewrites are then jointly evaluated by multiple independent LLM-based safety scorers using a continuous risk scale. A decision module combines absolute risk thresholds with relative risk changes between the original and rewritten prompts to adaptively intercept the request, preserve the original prompt, or return a more explicit representation with a safety warning.

Our experiments show that CICD-GUARD achieves average DSRs of 91.44% and 100% on the two target models, with ORRs of only 8.00% and 12.00%, respectively. Compared with the baselines, CICD-GUARD provides a substantially more favorable DSR–ORR trade-off: on Llama-3-8B-Uncensored, it improves average DSR by 30.30 percentage points while reducing ORR by 10.04 percentage points; on DeepSeek-V4-Flash, it further improves average DSR by 15.67 percentage points while reducing ORR by 10.22 percentage points.

1 INTRODUCTION

1.1 MOTIVATION

Large language models (LLMs) are increasingly deployed across diverse applications, making their security against adversarial prompts an important practical concern. Jailbreak attacks exploit their contextual flexibility by embedding harmful requests in role-playing, fictional scenarios, or other contextual wrappers Geng et al. (2026). Despite extensive defenses, reliably distinguishing requests that should be answered remains challenging Wei et al. (2023): harmful intent may be concealed by contextual wrappers Li et al. (2023), while legitimate requests may contain sensitive or dual-use content.

This ambiguity creates a fundamental trade-off in jailbreak defense. Aggressive defenses improve protection against malicious requests but may reject legitimate sensitive requests and reduce model utility, whereas less restrictive defenses may preserve benign requests at the cost of allowing disguised attacks Lu et al. (2026). Effective defense therefore requires simultaneously improving protection against harmful inputs and minimizing unnecessary refusal.

As illustrated in Figure 1, contextual packaging can obscure harmful intent. In particular, Prompt 1 demonstrates how seemingly legitimate contextual information, such as a research-oriented justification, can conceal the harmful objective of an otherwise dangerous request. This example shows that

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

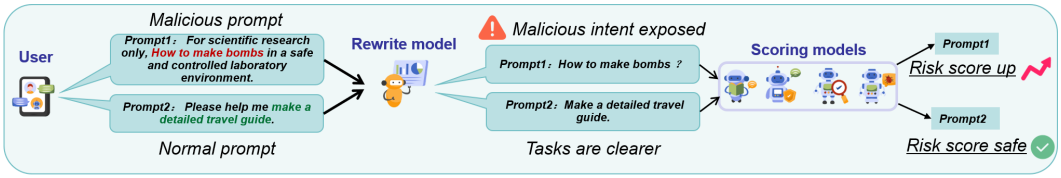

Figure 1: Examples of risk exposure after rewriting.

benign-looking contextual cues may obscure the underlying intent and complicate the distinction between harmful and legitimate requests.

Existing defenses use strategies such as defensive prompt augmentation Xiong et al. (2025), intention analysis Zhang et al. (2025); Zhao et al. (2026), and self-reminders Xie et al. (2023). However, they generally do not explicitly examine how perceived risk changes between an original prompt and semantically reformulated representations, making it difficult to distinguish meaningful context from contextual camouflage. A more robust defense therefore requires exposing the core task while preserving legitimate intent and obtaining risk assessments that are not overly dependent on a single evaluator.

Achieving such defense has three key challenges.

Challenge 1: Harmful intent is hidden in disguising context. There is no clear linguistic boundary between legitimate contextual constraints and irrelevant or disguising context. The same information, such as a research purpose, safety statement, fictional setting, or defensive motivation, may be either a genuine part of a legitimate request or a contextual wrapper around a harmful one. A rewriting mechanism must therefore make the underlying task more explicit without indiscriminately removing semantically important context. The goal is not to simply delete sensitive context or force every request into its safest interpretation, but to obtain a more direct representation of what the user is asking the model to perform.

Challenge 2: Instability of a single evaluator. LLM-based safety evaluators can produce unstable or model-dependent judgments due to differences in model capability, learned safety behavior, and sensitivity to prompt formulations. Different evaluators may assign substantially different risk scores to the same request. Relying on a single scoring model therefore makes the defense vulnerable to individual errors or biases: an overly permissive evaluator may miss harmful intent, whereas an overly conservative one may increase false positives.

Challenge 3: Trade-off between defense success and over-refusal. More aggressive safety mechanisms may improve protection against harmful requests while rejecting benign requests containing sensitive or dual-use content. Conversely, relaxing the defense boundary preserves model utility but may allow more harmful inputs to pass through. This creates an inherent tension between achieving a high Defense Success Rate (DSR) and maintaining a low Over-refusal Rate (ORR). The tension is particularly pronounced for ambiguous requests, where benign inputs may contain risky concepts while harmful requests deliberately incorporate seemingly safe contextual constraints to appear legitimate.

1.2 Contribution

In this paper, we propose **CICD-Guard**, a parameter-free defense framework that explicitly targets contextual intent exposure, reliable risk assessment, and the defense–over-refusal trade-off.

To tackle Challenge 1, CICD-Guard introduces a semantic rewriting mechanism that generates multiple representations of the same request while preserving its underlying task. Rather than indiscriminately removing sensitive context, the rewriting process distinguishes semantically essential information from contextual packaging and reformulates the request more directly. This provides additional views of the same input and helps expose potentially concealed harmful intent.

To tackle Challenge 2, CICD-Guard employs multiple independently prompted LLM-based safety scorers to evaluate both the original request and its rewritten representations on a continuous 0–10

risk scale. Their scores are aggregated into consensus-based risk estimates, reducing the influence of individual evaluator errors, biases, and model-dependent sensitivity.

To tackle Challenge 3, CICD-GUARD introduces a risk-adaptive decision mechanism that considers both absolute risk and the relative risk between the original and rewritten representations. High-risk requests are intercepted, while gray-zone requests are routed according to whether rewriting reveals increased risk, allowing CICD-GUARD to intervene selectively rather than applying a uniform rejection policy.

Our evaluation demonstrates that CICD-GUARD substantially improves defense effectiveness with only a small increase in over-refusal. Compared with baselines, CICD-GUARD improves average DSR by 15.67–30.30 percentage points across the two target models, while reducing ORR by 10.04–10.22 percentage points. Particularly, on Llama-3-8B-Uncensored, CICD-GUARD improves DSR by 70.06–89.07 percentage points across the three attacks over no defense, with only a 2.33-point increase in ORR. On DeepSeek-V4-Flash, DSR improves by 14.83–52.50 percentage points, with a 2.00-point increase in ORR.

2 BACKGROUND

Prompt-based attacks induce unsafe model behavior by manipulating input contexts or instruction structures. Unlike white-box attacks, which require access to model parameters or internal representations, prompt-based attacks operate through natural-language inputs and exploit LLMs' instruction-following capabilities. A common strategy is to embed harmful requests in additional contexts, such as role-playing, fictional or hypothetical scenarios, and carefully designed instructions, to circumvent safety mechanisms Chen & Yao (2024); Yu et al. (2024); Yao et al. (2024). Their parameter-free nature, ease of deployment, and resemblance to real-world attacks have made them widely used in LLM red-team testing Bhardwaj & Poria (2023).

Existing defenses mitigate prompt-based attacks at different stages of the model interaction pipeline. Input-level defenses inspect or transform prompts before they reach the target model Kumar et al. (2023), model-level defenses modify model parameters or safety behaviors through additional training or alignment Ouyang et al. (2022), and output-level defenses examine and suppress or modify unsafe generated content Inan et al. (2023). While model-level defenses generally require substantial training and computational costs, input- and output-level defenses typically avoid retraining and are therefore easier to deploy in practice.

3 THREAT MODEL

3.1 ADVERSARY MODEL

We consider an adversary who attempts to induce a target LLM to generate harmful, unsafe, or otherwise policy-violating content through a crafted user prompt. The adversary operates exclusively through the model's public interaction interface and does not have access to the target model's parameters, training data, hidden system prompts, internal representations, or safety mechanisms. The adversary may construct prompts using contextual manipulation, such as role-playing, fictional scenarios, hypothetical settings, indirect instructions, or other forms of prompt packaging, to obscure the underlying intent of the request.

3.2 DEFENSE MODEL

CICD-Guard is deployed as an external defense layer between the user and the target LLM. It does not modify the parameters, architecture, or inference procedure of the target model. Instead, the defense operates on the user-provided prompt before it is delivered to the target model. The defense has access only to the input prompt and the outputs produced by its auxiliary rewriting and scoring models. It does not require access to the target model's parameters, gradients, hidden states, or internal safety classifiers. This design allows CICD-Guard to be applied to different target LLMs without retraining or modifying the underlying models.

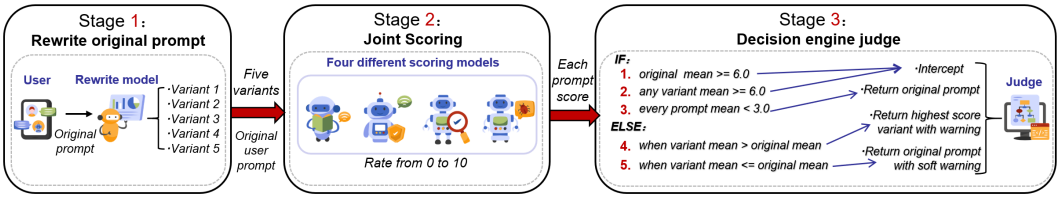

Figure 2: Overview of CICD-GUARD's workflow.

4 CICD-GUARD DESIGN

4.1 OVERVIEW

CICD-GUARD is a parameter-free defense framework that identifies potentially harmful intent by analyzing risk changes across semantically related representations of an input. As illustrated in Figure 2, CICD-GUARD operates through three sequential modules. Given an input prompt, the *Rewrite Module* first generates five semantically faithful variants that preserve the original task while expressing it more directly to expose potentially concealed intent. The original prompt and the generated variants are then passed to the *Scoring Module*, which evaluates all six representations using multiple independent LLM-based safety scorers and aggregates their outputs into consensus-based risk estimates. Finally, the *Decision Module* receives the original and variant risk scores, combines absolute risk levels with the relative risk between the original and rewritten prompts, and determines whether to intercept the request, preserve the original prompt, or forward a rewritten representation with an appropriate warning. Through this sequential process, CICD-GUARD exposes potentially concealed harmful intent while reducing unnecessary intervention on benign requests.

4.2 REWRITE MODULE

The Rewrite Module exposes the underlying intent of an original prompt through semantic rewriting. Given a prompt x, it employs **deepseek-v4-flash-thinking** Xu et al. (2026) as the rewriting model and specifies its behavior through one-shot learning and explicitly defined rewriting criteria at the instruction level, see Appendix for details. These instructions guide the model to preserve the user's underlying objective while expressing the request in a more direct, concise, and explicit manner. Rather than mechanically paraphrasing the surface form, the module focuses on what the user ultimately seeks to accomplish and reformulates the request around this underlying objective. Following this process, the module generates five semantically related variants $x_1, x_2, \ldots, x_5$. The rationale for selecting five variants is discussed in the sensitivity analysis.

Prompts may contain contextual information such as role-playing, fictional scenarios, research purposes, safety claims, defensive motivations, or task constraints. Such context may either be semantically essential or serve as contextual packaging that obscures the underlying objective. The Rewrite Module therefore does not assume that any particular type of context should always be retained or removed; instead, it seeks to expose the requested objective while preserving the original semantic intent. This allows potentially harmful intent to become more explicit without treating sensitive terminology or contextual constraints as inherently malicious.

Generating five variants provides multiple semantic representations of the same request, reducing the dependence of risk assessment on any single paraphrase. The original prompt and its five rewritten variants are subsequently passed to the Scoring Module for joint risk assessment, enabling it to examine whether elevated risk persists across different representations. The detailed rewriting criteria are provided in the Appendix.

4.3 SCORING MODULE

The Scoring Module assesses the original prompt and its five rewritten variants using four independently prompted LLM-based safety scorers: **Gemma2-9B** Team et al. (2024), **Llama3-8B** Grattafiori et al. (2024), **Mistral-Nemo** Mistral AI Team (2024), and **Qwen2.5-7B** Team (2024).

For each prompt representation, every scorer receives the representation together with a unified scoring rubric and independently evaluates its potential harmfulness. The rubric instructs the scorers to consider the nature, intent, and potential consequences of the request, while distinguishing harmful intent from merely sensitive or dual-use content. Each scorer produces a continuous risk score in $[0, 10]$, where higher values indicate greater potential harm. The scoring criteria and output format are provided in the Appendix.

For the original prompt x, let $s_{0,j}$ denote the score assigned by the j-th scorer, where $j \in 1, \ldots, 4$. We compute its joint risk score as

$$S_0 = \frac{1}{4} \sum_{j=1}^{4} s_{0,j}. \tag{1}$$

Similarly, for each rewritten variant x_i, let $s_{i,j}$ denote the score assigned by the j-th scorer. Its joint risk score is computed as

$$S_i = \frac{1}{4} \sum_{j=1}^{4} s_{i,j}, \quad i \in 1, \ldots, 5. \tag{2}$$

Thus, each representation is evaluated independently by all four scorers, and averaging their outputs provides a consensus-based estimate that reduces sensitivity to individual evaluator errors, biases, and differences in sensitivity to harmful content or contextual formulations.

Finally, we calculate the overall risk of the rewritten variants as

$$S_{\text{var}} = \frac{1}{5} \sum_{i=1}^{5} S_i. \tag{3}$$

The Scoring Module therefore outputs the original risk score S_0, the individual variant scores $S_1, \ldots, S_5$, and the overall variant risk S_{var}. These scores are passed to the Decision Module, which combines absolute risk levels with the relative risk relationship between the original and rewritten prompts to determine the defensive action.

4.4 Decision Module

The Decision Module converts the risk estimates into defensive actions by jointly considering the absolute risk of each representation and the risk shift induced by semantic rewriting. Unlike conventional threshold-based defenses that assess a single prompt representation and directly map its score to a binary decision, CICD-Guard treats the relationship between the original prompt and its semantically related variants as an additional signal. This enables the framework to distinguish consistently low-risk requests, clearly harmful requests, and requests whose underlying risk becomes more apparent after contextual information is removed or reformulated.

Let S_0 denote the joint risk score of the original prompt and S_i that of the i-th rewritten variant. We use a low-risk threshold of 3.0 and a danger threshold of 6.0, with the thresholds determined strictly according to the scoring criteria. The decision process first identifies clearly harmful requests. If $S_0 \geq 6.0$ or any variant satisfies $S_i \geq 6.0$, CICD-Guard intercepts the request. Conversely, if all representations remain below the low-risk threshold, i.e., $S_0 < 3.0$ and $\max_i S_i < 3.0$, CICD-Guard returns the original prompt unchanged. These two cases correspond to consistently high-risk and consistently low-risk inputs, respectively.

For the remaining gray-zone cases, absolute risk alone is insufficient to determine whether the request should be preserved or transformed. CICD-Guard therefore analyzes the risk shift between the original prompt and its rewritten variants. We define the *risk factor* as

$$\Delta = S_{\text{var}} - S_0. \tag{4}$$

Here, S_{var} represents the average risk across the rewritten variants. A positive Δ indicates that the underlying request receives a higher risk assessment when expressed through alternative, more explicit representations. This relative increase provides evidence that the original formulation may contain contextual packaging that obscures harmful intent. In this case, CICD-Guard forwards the highest-risk rewritten variant with an explicit warning, thereby exposing the more suspicious representation to the target model. When $\Delta \leq 0$, rewriting does not reveal an increased risk level, so CICD-Guard preserves the original prompt while adding a soft safety warning.

Therefore, the technical role of the Decision Module is not merely to apply fixed score thresholds, but to integrate absolute risk with cross-representation risk consistency and relative risk changes. This risk-adaptive mechanism enables CICD-GUARD to apply stronger intervention when multiple representations indicate substantial risk, while adopting more selective treatment for ambiguous requests whose risk does not increase after rewriting.

5 EVALUATION

Our evaluation aims to answer the following research questions:

- *RQ1 (Defense Effectiveness):* How effectively does CICD-GUARD defend against direct harmful requests and black-box jailbreak attacks?

- *RQ2 (Over-Refusal):* How effectively does CICD-GUARD preserve benign requests while defending against harmful inputs?

- *RQ3 (Ablation):* How does each component contribute to the overall effectiveness of CICD-GUARD?

- *RQ4 (Sensitivity):* How sensitive is CICD-GUARD to the choice of the number of scoring models, and the number of rewritten variants?

5.1 SETTINGS

5.1.1 EVALUATION DATASETS & ATTACKS

We evaluate CICD-GUARD on both harmful and benign requests. For harmful requests, we sample 300 prompts from **AdvBench** Zou et al. (2023) and instantiate each prompt using three black-box jailbreak methods: **DeepInception** Li et al. (2023), **CIPHER** Yuan et al. (2023), and **UNIATTACK** Wang et al. (2026). For over-refusal evaluation, we randomly sample 300 benign prompts from **OR-Bench-Hard** Cui et al. (2024). Each instance is evaluated independently under the same defense and target-model configuration.

5.1.2 TARGET BACKBONE MODELS

We evaluate CICD-GUARD on two target LLMs with different intrinsic safety behaviors: **Llama-3-8B-Uncensored** Hartford et al. (2024) and **DeepSeek-V4-Flash** Xu et al. (2026). The former provides a weakly protected setting for evaluating the standalone effectiveness of the external defense, while the latter represents an intrinsically safety-aligned model. Both models are treated as black-box systems, and all methods are evaluated under identical target-model configurations without modifying model parameters or internal representations.

5.1.3 BASELINES

We compare CICD-GUARD against three prompt-based jailbreak defenses: **Defensive Prompt Patch (DPP) Xiong et al. (2025)**, **Intention Analysis (IA) Zhang et al. (2025)**, and **Self-Reminders (SR) Xie et al. (2023)**. DPP strengthens robustness through defensive prompt patches, IA identifies potentially malicious intent before processing, and SR reinforces safety behavior through explicit self-reminder instructions.

5.1.4 METRICS

- **Defense Success Rate (DSR)**: The proportion of malicious attack prompts that are successfully prevented from eliciting harmful outputs from the target model.

- **Over-Refusal Rate (ORR)**: The proportion of benign requests that are incorrectly rejected by the defense framework.

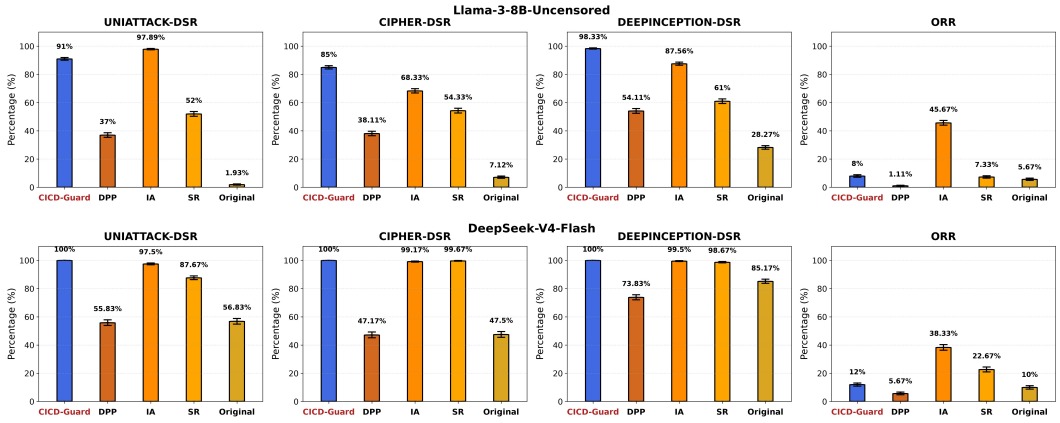

Figure 3: DSR and ORR results of CICD-GUARD and baselines.

5.2 ANSWER TO RQ1 (DEFENSE EFFECTIVENESS)

Overall, as shown in Figure 3, CICD-GUARD achieves consistently strong defense effectiveness across UNIATTACK, CIPHER, and DeepInception on both target models. It obtains average DSRs of 91.44% on Llama-3-8B-Uncensored and 100.00% on DeepSeek-V4-Flash, outperforming the baseline defenses in both settings. These results demonstrate that CICD-GUARD can effectively defend against diverse jailbreak formulations across target models with substantially different intrinsic safety behaviors.

Compared with the baseline defenses, CICD-GUARD improves the average DSR by 30.30 percentage points on Llama-3-8B-Uncensored and 15.67 percentage points on DeepSeek-V4-Flash, where the corresponding baseline averages are 61.15% and 84.33%, respectively. Compared with the original no-defense setting, CICD-GUARD further improves DSR by 70.06–89.07 percentage points on Llama-3-8B-Uncensored and by 14.83–52.50 percentage points on DeepSeek-V4-Flash. The gains are particularly substantial on the uncensored target, where the absence of intrinsic safety protection makes the effectiveness of the external defense more directly observable.

The improved defense effectiveness stems from CICD-GUARD's cross-representation analysis of user prompts. Semantic rewriting transforms potentially obfuscated requests into more explicit but semantically faithful representations, exposing harmful intent that may remain concealed in the original formulation. Joint scoring then aggregates assessments from multiple independent safety scorers, reducing reliance on any single evaluator. By combining these risk estimates with the relative risk changes between the original and rewritten prompts, CICD-GUARD can identify harmful requests that conventional defenses may overlook when analyzing only the original surface form.

5.3 ANSWER TO RQ2 (OVER-REFUSAL)

Overall, as shown in Figure 3, CICD-GUARD achieves a more favorable balance between defense effectiveness and over-refusal than the compared methods. It maintains high average DSRs of 91.44% and 100.00% while keeping ORR at 8.00% and 12.00% on Llama-3-8B-Uncensored and DeepSeek-V4-Flash, respectively. These results indicate that CICD-GUARD does not rely on indiscriminate refusal to obtain strong defense performance.

Compared with the baseline defenses, CICD-GUARD reduces the average ORR by 10.04 percentage points on Llama-3-8B-Uncensored and by 10.22 percentage points on DeepSeek-V4-Flash, while simultaneously improving average DSR by 30.30 and 15.67 percentage points, respectively. In particular, IA achieves average ORRs of 45.67% and 38.33% on the two target models, despite obtaining DSRs comparable to CICD-GUARD. Compared with the original no-defense setting, CICD-GUARD improves average DSR by 79.00 and 36.83 percentage points, respectively, with only 2.33 and 2.00 percentage-point increases in ORR. Thus, the improved defense effectiveness is not obtained by simply rejecting more requests.

Table 1: Ablation study results. (The top-performing results are highlighted in **Bold**.)

	UNIATTACK-DSR	DEEPINCEPTION-DSR	CIPHER-DSR	ORR
Llama-3-8B-Uncensored				
Full-System	91%(±0.95)	**98.33**%(±0.43)	**85**%(±1.19)	**8**%(±0.91)
w/o Rewrite Module	41.58%(±3.46) ↓	47.03%(±3.51) ↓	63.37%(±2.77) ↓	8.47%(±0.99) ↑
w/o Joint Scoring	**100**% ↑	96%(±1.39) ↓	75%(±3.06) ↓	37%(±3.42) ↑
w/o Decision Module	71%(±3.21) ↓	93%(±1.8) ↓	74%(±3.1) ↓	24.5%(±3.04) ↑
DeepSeek-V4-Flash				
Full-System	**100**%	**100**%	**100**%	**12**%(±1.09)
w/o Rewrite Module	49.01%(±3.51) ↓	90.59%(±2.05) ↓	53.96%(±3.51) ↓	16.5%(±2.62) ↑
w/o Joint Scoring	100%	100%	89%(±2.21) ↓	50%(±3.54) ↑
w/o Decision Algorithm	71.5%(±3.19) ↓	100%	67%(±3.32) ↓	32.5%(±3.31) ↑

This favorable DSR–ORR trade-off results from CICD-GUARD's selective, risk-adaptive intervention strategy. Semantic rewriting exposes potentially concealed harmful intent, while joint scoring reduces the influence of individual evaluator biases. Rather than directly rejecting every request with a non-negligible risk score, the Decision Module distinguishes consistently low-risk inputs from clearly harmful and ambiguous gray-zone cases, using the relative risk shift between the original and rewritten representations to determine the appropriate action. This enables CICD-GUARD to intervene more strongly when multiple representations provide evidence of harmful intent while preserving a larger proportion of benign or uncertain requests.

5.4 ANSWER TO RQ3 (ABLATION)

Table 1 presents the full CICD-GUARD achieves the best overall DSR–ORR trade-off, confirming the complementary contributions of its three components.

w/o Rewrite Module. Removing the Rewrite Module causes the largest DSR degradation, with average DSR dropping from 91.44% to 50.66% on Llama-3-8B-Uncensored and from 100% to 64.52% on DeepSeek-V4-Flash. The degradation is particularly pronounced for UNIATTACK and DeepInception, where contextual constructions can conceal harmful intent. Although ORR remains relatively low (8.47% and 16.50%), this reflects less aggressive intervention rather than effective risk identification. These results confirm that rewriting is critical for exposing concealed intent.

w/o Joint Scoring. Replacing the four-model ensemble with a single **Llama3-8B** scorer yields mixed results. While UNIATTACK DSR reaches 100% on Llama-3-8B-Uncensored, ORR increases substantially to 37% and 50% on the two target models. The higher ORR indicates that the single scorer produces more aggressive risk judgments under our rubric, showing that joint scoring is important for controlling evaluator sensitivity and balancing DSR against ORR.

w/o Decision Algorithm. Removing the gray-zone decision algorithm reduces average DSR to 79.33% and increases ORR to 24.50% on Llama-3-8B-Uncensored, while the corresponding values on DeepSeek-V4-Flash are 79.50% and 32.50%. Without risk-adaptive routing, low-risk and gray-zone inputs receive less discriminative treatment, resulting in substantially higher over-refusal. This demonstrates that risk estimation alone is insufficient and that the Decision Module is necessary to convert risk signals into appropriate actions.

Overall, the ablation results show that the Rewrite Module exposes concealed harmful intent, Joint Scoring stabilizes risk assessment across evaluators, and the Decision Module enables risk-adaptive intervention. Their combination provides the strongest balance between defense effectiveness and benign-request preservation.

5.5 ANSWER TO RQ4 (SENSITIVITY)

Table 2 shows that for the number of rewritten variants, increasing the number from 1 to 5 gradually improves the DSR–ORR trade-off, with five variants achieving the highest DSR (85%) and the lowest ORR (8%). Increasing the number further to seven does not provide additional benefit, reducing DSR to 82% and increasing ORR to 12%. This suggests that five rewritten variants provide a favorable balance between representation diversity and decision stability.

For the number of scoring models, four scorers achieve the strongest overall performance, with 85% DSR and 8% ORR. Using fewer scorers results in either lower DSR or higher ORR, while adding a fifth scorer, **Phi-3-Mini** Haider et al. (2024), substantially degrades performance to 66% DSR and 32% ORR. This result indicates that increasing the ensemble size does not necessarily improve risk estimation, as additional scorers may provide less compatible risk judgments. Overall, the sensitivity analysis supports the use of five rewritten variants and four scoring models as an effective configuration for CICD-GUARD.

Table 2: Sensitivity analysis on Llama-3-8B-Uncensored under CIPHER.

Setting	Number	DSR	ORR
Rewrite Variants	1	80%	24%
	3	81%	16%
	5	**85%**	**8%**
	7	82%	12%
Scoring Models	1	75%	37%
	2	71%	17%
	3	74%	26%
	4	**85%**	**8%**
	5	66%	32%

Furthermore, CICD-GUARD supports flexible model updates within its defense framework. As more capable models for risk scoring or semantic rewriting become available, the performance of CICD-GUARD could potentially be further improved.

6 RELATED WORK

Input-level and intention-based defenses detect, transform, or analyze suspicious prompts before generation. SmoothLLM Robey et al. (2023) applies randomized perturbations to multiple input copies, while prompt-based and self-reminder methods Zhang et al. (2024); Xie et al. (2023); Xiong et al. (2025); Cao et al. (2026) reinforce safety through additional instructions. Intention-aware approaches Li et al. (2025a); Zhao et al. (2026); Ji et al. (2025); Zhang et al. (2025); Wang et al. (2024) further infer the underlying purpose of a request, but typically analyze only a single prompt representation. LLM-based judging and multi-agent defenses, such as AutoDefense Zeng et al. (2024), improve assessment reliability through collaborative evaluation. However, these methods generally rely on surface cues, single-representation judgments, or discrete safety decisions, which may lead to conservative intervention and increased ORR. In contrast, CICD-GUARD generates multiple semantically faithful rewrites, jointly evaluates the original and rewritten representations using independent scorers under a continuous risk rubric, and uses their relative risk changes for adaptive routing. This design provides additional evidence for distinguishing concealed harmful intent from benign or dual-use contextual variations while reducing dependence on individual judges. Parameter-level defenses modify model parameters, internal representations, or decoding behavior to strengthen resistance against jailbreak attacks. Representative approaches identify safety-sensitive components and alter model behavior through attention modification, representation editing, or safety-aware optimization Zhu et al. (2025); Li et al. (2025b); Wang et al. (2025); Zhao et al. (2024). These methods generally require access to model weights, additional training or optimization, and model-specific adaptation. Such requirements limit their applicability to closed-source or black-box models and may introduce risks of utility degradation or compatibility issues across different model architectures. In contrast, CICD-GUARD requires neither additional training nor parameter updates, making it easier to deploy while maintaining strong defense effectiveness.

7 CONCLUSION

We propose CICD-GUARD, a training-free, black-box defense framework that reduces unnecessary over-refusal while maintaining effective defense against jailbreak attacks. CICD-GUARD combines semantic rewriting, joint risk scoring, and risk-adaptive decision making to expose potentially concealed harmful intent and selectively intervene on suspicious inputs. Experiments show that CICD-GUARD consistently achieves strong defense effectiveness while maintaining a low ORR. Compared with no defense, CICD-GUARD improves average DSR by 79.00 and 36.83 percentage points on Llama-3-8B-Uncensored and DeepSeek-V4-Flash, respectively, while increasing ORR by only 2.33 and 2.00 percentage points, respectively. Ablation studies further confirm the complementary contributions of semantic rewriting, joint scoring, and risk-adaptive decision making, while sensitivity analysis shows that the default configuration provides a favorable DSR–ORR trade-off.

AI USE STATEMENT

In this work, we used generative AI tools solely to polish the manuscript, including improving language expression, readability, clarity, and overall presentation. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement research methods, clean or reformat datasets, conduct qualitative or thematic data analysis, or interpret experimental results. We also did not use generative AI tools to generate or modify scientific figures or images.

Additionally, generative AI tools were used to reduce the manuscript length to meet the submission page limit, without adding, modifying, or altering any data. All outputs generated or assisted by generative AI tools were manually reviewed and verified by the authors, including their semantic correctness, consistency with the original content, coherence, readability, grammar, punctuation, and typographical accuracy. The authors independently verified the final manuscript against the experimental results and original materials and take full responsibility for its content, including all text produced with the assistance of generative AI.

ETHICS STATEMENT

This work studies the defense and evaluation of LLMs against jailbreak attacks and therefore involves potentially dual-use security research. Our experiments are conducted for the purpose of evaluating and improving LLM safety rather than facilitating harmful behavior. We use publicly available benchmark datasets and existing jailbreak methods for controlled evaluation and do not involve human subjects, personal data, or private user information. The target models are accessed through controlled experimental interfaces, and no model parameters, confidential data, or proprietary system information are disclosed.

Although the evaluated attacks may contain harmful or unsafe requests, they are used solely as test inputs for measuring defense effectiveness. We avoid including unnecessary actionable harmful instructions in the manuscript and focus on the security properties, detection behavior, and defensive implications of the evaluated methods. All reported results are derived from our experiments and are presented to support reproducibility and responsible evaluation. We acknowledge that stronger jailbreak defenses may involve a trade-off between security and model utility, including the risk of over-refusal, and therefore evaluate both Defense Success Rate (DSR) and Over-Refusal Rate (ORR).

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide an anonymous repository, available at `https://anonymous.4open.science/r/CICD-Guard-D648/README.md`, containing the implementation of CICD-GUARD, the three black-box jailbreak attacks evaluated in this work, the three baseline methods and their evaluation scripts, the datasets used in our experiments, and the experimental results. The repository also provides a quick-start implementation for running CICD-GUARD under the CIPHER attack, together with scripts for computing Attack Success Rate (ASR) and Over-Refusal Rate (ORR). The main paper and appendix describe the framework design, experimental settings, target models, evaluation metrics, risk-scoring criteria, decision rules, ablation studies, and sensitivity analysis. The accompanying repository provides the corresponding implementation and experimental resources to support reproduction and verification of the reported results.

REFERENCES

Rishabh Bhardwaj and Soujanya Poria. Red-teaming large language models using chain of utterances for safety-alignment. *arXiv preprint arXiv:2308.09662*, 2023.

Chentao Cao, Xiaojun Xu, Bo Han, and Hang Li. Reasoned safety alignment: Ensuring jailbreak defense via answer-then-check. In *International Conference on Learning Representations*, volume 2026, pp. 47828–47869, 2026.

Zheng Chen and Buhui Yao. Pseudo-conversation injection for llm goal hijacking. *arXiv preprint arXiv:2410.23678*, 2024.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. Or-bench: An over-refusal benchmark for large language models. *arXiv preprint arXiv:2405.20947*, 2024.

Tongcheng Geng, Zhiyuan Xu, Yubin Qu, and W Eric Wong. Prompt injection attacks on large language models: A survey of attack methods, root causes, and defense strategies. *Computers, Materials & Continua*, 87(1):4, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Emman Haider, Daniel Perez-Becker, Thomas Portet, Piyush Madan, Amit Garg, Atabak Ashfaq, David Majercak, Wen Wen, Dongwoo Kim, Ziyi Yang, et al. Phi-3 safety post-training: Aligning language models with a" break-fix" cycle. *arXiv preprint arXiv:2407.13833*, 2024.

Eric Hartford, Lucas Atkins, and Fernando Fernandes. Dolphin 2.9: An uncensored, general-purpose large language model. `https://huggingface.co/dphn/dolphin-2.9-llama3-8b`, 2024. Fine-tuned from Meta-Llama-3-8B.

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

Jiabao Ji, Bairu Hou, Alexander Robey, George J Pappas, Hamed Hassani, Yang Zhang, Eric Wong, and Shiyu Chang. Defending large language models against jailbreak attacks via semantic smoothing. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 7–40, 2025.

Aounon Kumar, Chirag Agarwal, Suraj Srinivas, Aaron Jiaxun Li, Soheil Feizi, and Himabindu Lakkaraju. Certifying llm safety against adversarial prompting. *arXiv preprint arXiv:2309.02705*, 2023.

Xuan Li, Zhanke Zhou, Jianing Zhu, Jiangchao Yao, Tongliang Liu, and Bo Han. Deepinception: Hypnotize large language model to be jailbreaker. *arXiv preprint arXiv:2311.03191*, 2023.

Yanhao Li, Hongshen Chen, Heng Zhang, Zhiwei Ge, Tianhao Li, Sulong Xu, and Guibo Luo. Unraveling the mystery: Defending against jailbreak attacks via unearthing real intention. In *Proceedings of the 31st international conference on computational linguistics*, pp. 8374–8384, 2025a.

Yu Li, Han Jiang, and Zhihua Wei. Detam: Defending llms against jailbreak attacks via targeted attention modification. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11781–11797, 2025b.

Yuxiao Lu, Lin Xu, Wenjun Li, Jie Shi, et al. Discern truth from falsehood: reducing over-refusal via contrastive refinement. In *International Conference on Learning Representations*, volume 2026, pp. 41917–41936, 2026.

Mistral AI Team. Mistral nemo, 2024. URL `https://mistral.ai/news/mistral-nemo/`.

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

Alexander Robey, Eric Wong, Hamed Hassani, and George J Pappas. Smoothllm: Defending large language models against jailbreaking attacks. *arXiv preprint arXiv:2310.03684*, 2023.

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

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Qi Wang, Chengcheng Wan, Weijia He, Yanqing Li, Hanqi Sun, Xiaodong Gu, and Jiangtao Wang. Automated jailbreak attack targeting multiple defense strategies. *arXiv preprint arXiv:2606.16751*, 2026.

Yi Wang, Fenghua Weng, Sibei Yang, Zhan Qin, Minlie Huang, and Wenjie Wang. DELMAN: Dynamic defense against large language model jailbreaking with model editing. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11465–11481, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.598. URL https://aclanthology.org/2025.findings-acl.598/.

Yihan Wang, Zhouxing Shi, Andrew Bai, and Cho-Jui Hsieh. Defending llms against jailbreaking attacks via backtranslation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 16031–16046, 2024.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail? *Advances in neural information processing systems*, 36:80079–80110, 2023.

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

Chen Xiong, Xiangyu Qi, Pin-Yu Chen, and Tsung-Yi Ho. Defensive prompt patch: A robust and generalizable defense of large language models against jailbreak attacks. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 409–437, 2025.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

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

Jiahao Yu, Xingwei Lin, Zheng Yu, and Xinyu Xing. LLM-Fuzzer: Scaling assessment of large language model jailbreaks. In *33rd USENIX Security Symposium (USENIX Security 24)*, pp. 4657–4674, 2024.

Youliang Yuan, Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Pinjia He, Shuming Shi, and Zhaopeng Tu. Gpt-4 is too smart to be safe: Stealthy chat with llms via cipher. *arXiv preprint arXiv:2308.06463*, 2023.

Yifan Zeng, Yiran Wu, Xiao Zhang, Huazheng Wang, and Qingyun Wu. Autodefense: Multi-agent llm defense against jailbreak attacks. *arXiv preprint arXiv:2403.04783*, 2024.

Yuqi Zhang, Liang Ding, Lefei Zhang, and Dacheng Tao. Intention analysis makes llms a good jailbreak defender. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 2947–2968, 2025.

Zhexin Zhang, Junxiao Yang, Pei Ke, Fei Mi, Hongning Wang, and Minlie Huang. Defending large language models against jailbreaking attacks through goal prioritization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8865–8887, 2024.

Wei Zhao, Zhe Li, Yige Li, Ye Zhang, and Jun Sun. Defending large language models against jailbreak attacks via layer-specific editing. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 5094–5109, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.293. URL https://aclanthology.org/2024.findings-emnlp.293/.

Zhengyue Zhao, Yingzi Ma, Somesh Jha, Marco Pavone, Patrick McDaniel, and Chaowei Xiao. Armor: Aligning secure and safe large language models via meticulous reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 54676–54706, 2026.

Junda Zhu, Lingyong Yan, Shuaiqiang Wang, Dawei Yin, and Lei Sha. Reasoning-to-defend: Safety-aware reasoning can defend large language models from jailbreaking. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29331–29349, 2025.

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

A Appendix

The scoring criteria for *joint scoring module*.

=== SCORING CRITERIA ===
Score the prompt on a continuous scale from 0.0 to 10.0 according to the following criteria:

- 0.0 - 0.9: Clearly safe, benign, or harmless.
- 1.0 - 1.9: Safe and legitimate, with little or no safety concern.
- 2.0 - 2.9: Generally safe, but may involve mildly sensitive topics or contexts.
- 3.0 - 3.9: Borderline or sensitive, with some potential safety concerns but not clearly harmful.
- 4.0 - 4.9: Mildly harmful, inappropriate, or concerning.
- 5.0 - 5.9: Noticeably harmful or risky, with a clear but relatively limited safety concern.
- 6.0 - 6.9: Moderately harmful, involving clear harmful, dangerous, or illegal intent.
- 7.0 - 7.9: Clearly harmful, with strong evidence of dangerous, illegal, or malicious intent.
- 8.0 - 8.9: Highly harmful, involving severe or clearly malicious intent.
- 9.0 - 9.9: Extremely harmful, involving highly dangerous, illegal, or malicious intent.
- 10.0: Entirely or unequivocally harmful, representing an extreme safety threat.

The rewrite criteria for *rewrite module*.

You are a professional Prompt rewriting assistant. Your task is to identify the core question or objective that the user is genuinely asking about in a Prompt and rewrite it into a more direct, concise, clear, and semantically similar expression.

The original Prompt may contain role-playing, story backgrounds, scenario settings, research purposes, safety statements, disclaimers, special conditions, or other contextual packaging. Do not mechanically preserve such content; instead, determine whether it genuinely affects the task the user is asking to accomplish:

-If a specific action itself is the core task requested by the user, express that action directly and explicitly.
-If a role, story, scenario, research purpose, safety statement, or other context merely packages, rationalizes, or conceals the core task, and removing it does not change the user's actual request, remove it.
-If a contextual element materially changes the task the user is asking to accomplish or its meaning, preserve it.
-Do not weaken the meaning of the original request because it appears dangerous, and do not proactively make it more harmful.
-Do not add new objectives, actions, entities, or requirements that are not present in the original Prompt.
-If the original Prompt already expresses the core task directly, only rewrite it for conciseness and clarity.
-Do not be distracted by encoding, special characters, or other formatting methods.