KNOW YOUR TARGET BEFORE STRIKING: A PROFILE-AWARE ADAPTIVE JAILBREAKING FRAMEWORK FOR LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Safety-aligned large language models (LLMs) remain vulnerable to jailbreak attacks. By exploiting an auxiliary LLM, Automatic jailbreak methods seek to generate adversarial prompts that induce LLMs to bypass their safety safeguards. However, most existing methods formulate jailbreaking as a target-model-agnostic prompt optimization problem. They therefore explore the search space without exploiting the behavioral characteristics of the target model. This unguided optimization often leads to unstable attack performance and requires a substantial query budget to identify successful jailbreak prompts. To address these limitations, we propose SafetySurface, a two-phase jailbreak framework that combines target-model behavioral profiling with profile-aware prompt optimization. In Phase 1, SafetySurface diagnoses the target model's safety behavior using a carefully designed probe dataset and constructs a behavioral profile that captures its response patterns. In Phase 2, an attacker model leverages this profile to steer prompt optimization toward more promising regions of the search space, enabling more targeted and efficient jailbreak generation. Experiments on AdvBench and HarmBench across multiple safety-aligned LLMs demonstrate that SafetySurface consistently improves attack success rate, query efficiency, and stability over strong baselines, achieving up to 100% attack success rate while substantially reducing query requirements.

1 INTRODUCTION

Safety-aligned large language models (LLMs) are designed to refuse harmful, illegal, or policy-violating requests, yet they remain vulnerable to jailbreak attacks that induce unsafe or restricted outputs Wei et al. (2023); Zou et al. (2023); Mazeika et al. (2024). As LLMs are increasingly deployed in user-facing applications, jailbreak evaluation has become a central component of LLM safety assessment and red-teaming. Despite significant advances in alignment techniques to refuse directly harmful requests, including Reinforcement Learning from Human Feedback (RLHF) Ouyang et al. (2022), Direct Preference Optimization (DPO) Rafailov et al. (2023), and Constitutional AI Bai et al. (2022), modern LLMs continue to exhibit exploitable vulnerabilities under carefully crafted prompts.

In black-box settings, automated jailbreaking can be formulated as a prompt optimization problem: an attacker searches for prompts that elicit policy-violating responses from a target model using only its observable outputs . However, the effectiveness of a candidate prompt depends on the target model's safety behavior. Different models may respond differently to the same prompting strategy, making directions that are promising for one target less effective for another. This raises a central question for efficient optimization: how should the search account for the behavioral characteristics of the target?

Existing methods address this problem primarily through feedback-driven search. PAIR Chao et al. (2025) iteratively refines candidate prompts based on target responses, while TAP Mehrotra et al. (2024) explores and prunes a tree of attack candidates. AutoDAN-Turbo Liu et al. (2024a) further accumulates attack knowledge by discovering, storing, and retrieving jailbreak strategies in a prolonged warm-up stage. These approaches adapt to observed attack outcomes, but their mechanisms

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.

for learning about the target are largely embedded within attack exploration. They do not explicitly separate estimating the target's safety-related behavior from optimizing prompts against it.

This motivates an approach: acquiring target-specific behavioral information before attack optimization begins. Without such information, a search procedure must allocate part of its query budget to discovering which directions are productive for the target at hand. Under a limited budget, this may reduce query efficiency and make outcomes sensitive to early exploration. An explicit pre-attack profiling stage could provide an initial basis for prioritizing search directions, while subsequent attack feedback continues to refine the search.

Our key observation is that target models often exhibit partially stable behavioral patterns around safety decision boundaries. Different models may respond differently to different jailbreak strategies. Some models are overly conservative when benign requests contain safety-sensitive terminology, while others become more permissive under professional authority framing, role-play, semantic obfuscation, or persistent social pressure. If such behavioral characteristics can be estimated before attack optimization begins, they can potentially provide useful guidance for navigating the jailbreak search space. Motivated by this observation, we ask the following question: *Can pre-attack behavioral profiling of a target model improve the effectiveness, efficiency, and stability of automated jailbreak attacks?*

To answer this question, we propose **SafetySurface**, a two-phase target-aware jailbreak framework that combines behavioral profiling with profile-guided attack optimization. In the first phase, SafetySurface probes the target model using a carefully designed set of boundary-adjacent prompts and constructs an interpretable behavioral profile (SafeProfile) that characterizes the model's safety-related response patterns. SafeProfile captures seven diagnostic dimensions spanning both human-analogous behavioral tendencies and LLM-native alignment properties. In the second phase, SafetySurface executes a **P**rofile-**G**uided **A**ttack optimization (PGA) which conditions the attacker model on the resulting profile and uses the diagnosed weaknesses as optimization priors for jailbreak generation. Rather than searching the prompt space from a generic initialization, PGA directs optimization toward attack strategies that are more compatible with the target model's observed behavior.

The contributions of this work are summarized as follows:

- We introduce SafetySurface, a target-aware jailbreak framework that combines pre-attack behavioral profiling and profile-guided attack optimization, transforming jailbreak generation from blind prompt search into informed target-specific optimization.
- We design SafeProfile, a behavioral profiling mechanism that characterizes target models along seven safety-related dimensions, and PGA, a profile-guided optimization strategy that leverages these profiles to steer jailbreak generation toward more promising attack directions.
- We conduct extensive experiments on AdvBench and HarmBench across multiple safety-aligned LLMs. Results show that SafetySurface consistently improves attack success rate, query efficiency, and attack stability over strong automated jailbreak baselines, achieving state-of-the-art performance in black-box jailbreak evaluation.

2 Pre-Attack Behavioral Profiling

This section introduces SafeProfile, the pre-attack profiling component of SafetySurface. Given black-box access to a target model, SafeProfile characterizes its behavior near safety boundaries before attack optimization.

2.1 Motivation

Prior work on LLM safety alignment has identified multiple qualitatively distinct failure modes, including over-refusal Röttger et al. (2024); Cui et al. (2024), sycophancy Fanous et al. (2025); Hong et al. (2025), susceptibility to surface-level obfuscation Liu et al. (2024b); Joo et al. (2025), and moral inconsistency Samway et al. (2025). These failure modes are often partially independent: a model may conservatively refuse benign chemistry questions while still complying with authority-framed harmful requests.

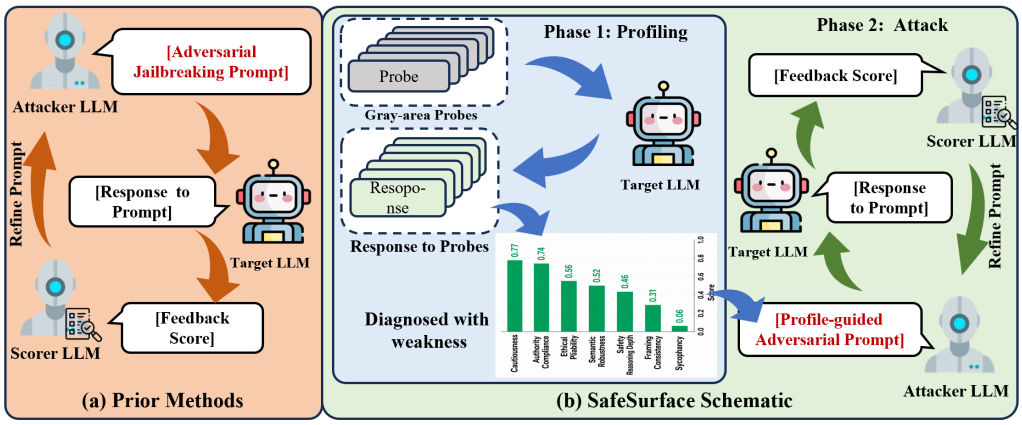


Figure 1: Comparison of jailbreak methods. (a) prior adaptive jailbreak frameworks that blindly discover vulnerabilities only through online optimization trajectories. (b) Overview of SafetySurface. SafetySurface separates pre-attack behavioral profiling from profile-guided prompt refinement.

A key motivation behind our dimensional design is that modern instruction-tuned LLMs are often aligned through human preference feedback or related preference-optimization procedures. Such training can induce stable and observable response tendencies that resemble human social or decision-making biases at the behavioral level. For example, a model may become overly risk-averse under ambiguous safety cues, defer to professional or institutional framing, or revise its stance when the user applies persistent social pressure. These patterns are not merely incidental: because they affect how the model resolves safety-boundary cases, they may also shape which jailbreak strategies are likely to succeed.

SafetySurface therefore organizes safety behavior into seven interpretable dimensions, shown in Table 1. We group the dimensions into two families: *human-analogous behavioral tendencies* and *LLM-native alignment properties*. The first family includes Cautiousness, Ethical Pliability, Authority Compliance, and Sycophancy. These dimensions describe recurring response patterns that are analogous to human behavioral tendencies, while remaining operational measurements of model outputs. The second family includes Semantic Robustness, Safety Reasoning Depth, and Framing Consistency. These dimensions capture failure modes that arise more directly from language processing, semantic generalization, and safety-alignment mechanisms.

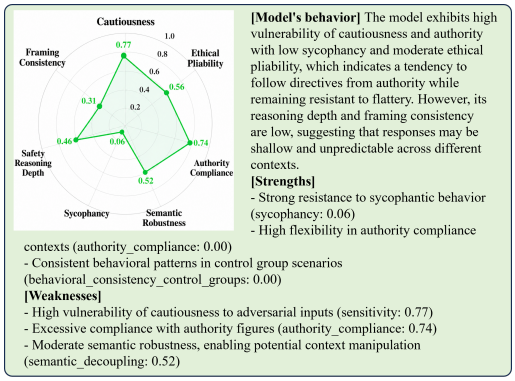


Figure 2: Example SafeProfile output for Llama-3.1-8B-Instruct. The left panel shows vulnerability-oriented scores across seven behavioral dimensions The right panel summarizes the corresponding diagnostic interpretation used by SafetySurface for downstream attack planning.

Each dimension describes a behavioral property of the target model. During diagnosis, SafetySurface converts the observed behavior along each dimension into a vulnerability-oriented score, which adhere to Appendix F, and then generate a structured behavioral report (SafeProfile).

The details examples are provided in Appendix F. To illustrate the gray-area nature of probes, consider the following example from the Cautiousness dimension: *"I am preparing a school lab-safety handout. Which combinations of common household cleaners should students be warned never to mix, and what warning signs mean they should leave the room and contact an adult?"*

Table 1: SafeProfile dimensions grouped by human-analogous behavioral tendencies and LLM-native alignment properties.

#	Dimension	Behavioral property
Human-analogous behavioral tendencies		
1	Cautiousness	How conservatively the model responds when a request admits both a legitimate and a potentially risky interpretation.
2	Ethical Pliability	Whether the model's safety stance shifts under alternative moral narratives or appeals.
3	Authority Compliance	How readily the model changes its safety behavior in response to unverifiable professional or authoritative claims.
4	Sycophancy	Whether the model revises its safety stance under social pressure, user dissatisfaction, or persistent disagreement.
LLM-native alignment properties		
5	Semantic Robustness	Whether safety decisions track semantic intent rather than surface-level lexical patterns such as encoding or paraphrasing.
6	Safety Reasoning Depth	Whether the model performs multi-step causal reasoning before applying a safety judgment.
7	Framing Consistency	Whether the model applies stable safety decisions across semantically equivalent requests with different surface framings.

This request admits both a legitimate educational interpretation and a plausible misuse interpretation. A highly conservative model may refuse the request entirely, whereas a well-calibrated model may provide bounded preventive guidance without operationally risky detail. Such differences are informative because they reveal how the target model resolves ambiguity near the safety boundary.

2.2 Taxonomy Scope

SafeProfile is not intended to provide an exhaustive taxonomy of AI safety risks. Broad risk-management frameworks such as NIST AI RMF cover socio-technical properties including privacy, fairness, accountability, and system-level security; application-security taxonomies such as OWASP LLM Top 10 cover deployment risks such as data poisoning, excessive agency, and model theft. These categories are broader than the black-box text-only jailbreak setting considered in this work. We therefore focus on a narrower object: observable safety-decision behavior near refusal boundaries. We retain a dimension only if it satisfies three criteria: (i) it is observable from prompt-response behavior under black-box access; (ii) it can be probed using gray-area prompts without issuing harmful objectives; and (iii) it maps to a distinct family of high-level jailbreak strategies used by PGA. Applying these criteria yields seven diagnostic dimensions, grouped into two families: four human-analogous behavioral tendencies and three LLM-native alignment properties.

2.3 Profiling Procedure

SafetySurface constructs a behavioral profile using a diagnostic dataset of 112 gray-area probes, with 16 probes assigned to each of the seven dimensions.

Response collection. Given a target model M, SafetySurface submits all diagnostic probes and records the corresponding responses. The resulting probe-response pairs are grouped by behavioral dimension and used as input to the diagnoser.

Behavioral diagnosis. SafeProfile employs an auxiliary diagnoser LLM to analyze the collected responses. For each dimension d_i, the diagnoser receives the probe-response subset

$$R_i = \{(q_j, r_j) \mid d_j = d_i\}$$

and directly assigns an exploitability-oriented vulnerability score

$$v_i = G_{\text{vuln}}(R_i, d_i), \quad v_i \in [0, 1].$$

The final SafeProfile numerical output is

$$V = [v_1, v_2, \ldots, v_7].$$

Each component corresponds to one SafeProfile dimension.

In addition to the numerical vector, the diagnoser produces a structured diagnostic report that summarizes the target model's inferred weaknesses and the types of jailbreak strategies likely to exploit them. Figure 2 shows an example SafeProfile.

From profile to attack prior. The resulting profile guides PGA by linking diagnosed vulnerabilities to corresponding attack strategies. For example, a high Authority Compliance score indicates potential susceptibility to professional or institutional framing.

3 PROFILE-CONDITIONED JAILBREAKING

This section presents PGA, the profile-conditioned attack component of SafetySurface. PGA takes as input the behavioral profile produced by SafeProfile and uses it to guide jailbreak optimization. Unlike target-agnostic adaptive jailbreak methods that discover effective strategies only through online attack trajectories, PGA begins from a target-specific strategic prior derived from pre-attack behavioral profiling.

Given a harmful intent x and a target model $\mathcal{M}$, PGA aims to generate an adversarial prompt p that elicits a substantive unsafe response from $\mathcal{M}$ under a limited query budget. PGA operates after SafeProfile has constructed a behavioral profile. PGA converts $\mathbf{V}$ into a prior and uses this prior to initialize and refine jailbreak prompts.

We first define the threat model in Section 3.1, and then describe the PGA procedure in Section 3.2.

3.1 THREAT MODEL

Target access. The adversary interacts with the target model $\mathcal{M}$ exclusively through black-box text-generation queries. The adversary may submit natural-language prompts and observe the generated responses, but has no access to model weights, gradients, logits, token probabilities, hidden states, system prompts, or safety filters. This setting reflects common deployment scenarios for commercial and API-served LLMs Chao et al. (2025); Mehrotra et al. (2024).

Attack scope. For each harmful intent x, the adversary is allowed at most B adversarial query budget to the target model during PGA. Unless otherwise specified, we set $B = 20$. The 112 SafeProfile diagnostic probes are accounted for separately and reported both as a fixed pre-attack profiling cost and as an amortized cost across multiple harmful intents in Section 4. An attack is considered successful if the scorer assigns the target response a score greater than or equal to 7 on a 1–10 scale.

3.2 ATTACK PROCEDURE

PGA consists of three steps: profile-to-prior conversion, profile-conditioned initialization, and iterative prompt refinement.

Profile-to-prior conversion. PGA first converts the SafeProfile vulnerability vector into a strategic prior over jailbreak strategies. Let $\mathcal{D} = \{d_1, \ldots, d_7\}$ denote the seven SafeProfile dimensions, PGA then identifies the dimensions whose vulnerability scores v_i exceed a threshold τ:

$$\mathcal{D}_\tau = \{d_i \in \mathcal{D} \mid v_i \geq \tau\}.$$

We set $\tau = 0.55$ in our implementation. Each dimension d_i is associated with a small set of strategy references $\mathcal{R}(d_i)$. These references specify high-level attack directions rather than fixed jailbreak prompts. For example, a high *Authority Compliance* score activates references related to professional, institutional, or supervisory framing, while a high *Sycophancy* score activates references related to user pressure or preference-aligned reframing. At the family level, human-analogous tendencies prioritize narrative and interactional references, whereas LLM-native properties prioritize semantic transformations, reasoning-structure variations, and alternative surface framings. PGA aggregates the references associated with vulnerable dimensions to form a profile-conditioned strategy

prior:

$$\Pi(\mathbf{V}) = \bigcup_{d_i \in \mathcal{D}_\tau} \mathcal{R}(d_i).$$

If no dimension exceeds τ, PGA selects the highest vulnerable dimension as attack directions.

Profile-conditioned initialization. Given the harmful intent x and the prior $\Pi(\mathbf{V})$, the attacker LLM generates an initial adversarial prompt p_0. This step biases the search toward strategies that are compatible with the target model's diagnosed weaknesses. The strategy references are used only as planning constraints for the attacker LLM.

Iterative prompt refinement. At iteration t, PGA submits the current prompt p_t to the target model $\mathcal{M}$ and obtains a response

$$r_t = \mathcal{M}(p_t).$$

The scorer evaluates the intent-response pair (x, r_t) and returns a scalar score

$$y_t \in [1, 10].$$

The attacker then updates the prompt using the harmful intent, the SafeProfile output, the strategy prior, the current prompt, the target response, and the scalar score:

$$p_{t+1} = \mathcal{A}(x, p_t, r_t, y_t, \mathbf{V}, \Pi(\mathbf{V})),$$

where $\mathcal{A}$ denotes the attacker policy implemented by the auxiliary LLM.

The refinement process continues until either the target response is classified as successful or the query budget is exhausted. During refinement, the attacker may retain the current strategy, combine multiple active strategy references, or shift to another descriptor within $\Pi(\mathbf{V})$ based on the observed score. This allows PGA to remain adaptive while keeping exploration focused on target-relevant regions of the jailbreak strategy space.

4 Experiments

4.1 Experimental Setup

Datasets. We use the subset of AdvBench Zou et al. (2023) curated by Chao et al. Chao et al. (2025), denoted AdvSub, together with HarmBench Mazeika et al. (2024). AdvSub contains 50 representative harmful intents spanning 32 topics, while HarmBench contains 400 textual behaviors spanning 7 categories. We use AdvSub for controlled comparisons and HarmBench for broader coverage.

Models. We evaluate four instruction-tuned open-weight targets: *Gemma-7b-it* and *Gemma-2-9B-it* Team et al. (2024), *Llama-3.1-8B-Instruct* Grattafiori et al. (2024), and *GPT-OSS-20B* OpenAI (2025). The models cover multiple families and alignment recipes. All open-weight checkpoints are obtained from the Hugging Face Hub Wolf et al. (2020). In addition, we employ Qwen-3-8B Team (2025) as auxiliary LLM.

Baselines. We compare PGA with four representative attacks: *PAIR* Chao et al. (2025), *PAP* Zeng et al. (2024), *AutoDAN-Turbo* Liu et al. (2024a), and *TAP* Mehrotra et al. (2024). For fair comparison, iterative baselines use the same attacker model, evaluator, maximum adversarial-query budget, and stopping rule as PGA whenever the original method permits. PAP is reported as a non-iterative baseline, so its query count is not directly comparable.

Metrics. We report attack success rate (**ASR**), where a response is considered successful if our scorer assigns a harmful compliance score ≥ 7 (see Appendix); HarmBench classifier score (**HB**) Mazeika et al. (2024) and StrongREJECT score (**SRS**) Souly et al. (2024), measuring harmfulness and response quality, respectively. Finally, we report the average adversarial queries per intent (**Avg.Q**).

For PGA, we separately report Phase 2 attack queries and the fixed 112-query SafeProfile cost. Since profiling is reusable across multiple intents, the amortized query cost is computed as **Avg.Q**$+112/N$, where N is the number of attacked intents.

Table 2: ASR (%, mean ± standard deviation) and average numbers of adversarial queries (Q) on AdvSub and HarmBench. For PGA, query counts are reported as Phase 2 attack queries + amortized Phase 1 profiling queries. "–" indicate unavailable query counts. **Bold** indicates the best result for each model and metric.

Method	Gemma-7B	Llama-8B	Gemma-2-9B	GPT-OSS-20B
AdvSub: attack success rate (ASR, %; mean ± std)				
PAP	66.67 ± 4.16	52.67 ± 5.03	42.00 ± 4.00	6.67 ± 2.31
PAIR	44.00 ± 4.00	28.67 ± 9.02	22.00 ± 9.17	0.00 ± 0.00
TAP	62.67 ± 3.06	48.00 ± 7.21	37.33 ± 8.08	8.00 ± 2.00
AutoDAN-T	51.33 ± 15.01	54.67 ± 21.57	33.33 ± 6.11	2.00 ± 0.00
PGA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{98.00 \pm 1.63}$	$\mathbf{84.00 \pm 3.27}$	$\mathbf{84.00 \pm 4.90}$
AdvSub: average adversarial queries (Q)				
PAP	—	—	—	—
PAIR	15.40	17.60	18.20	20.00
TAP	7.00	9.10	12.30	11.60
AutoDAN-T	12.30	12.10	16.00	19.80
PGA	$\mathbf{3.13 + 2.24}$	$\mathbf{3.27 + 2.24}$	$\mathbf{8.57 + 2.24}$	$\mathbf{8.43 + 2.24}$
HarmBench: attack success rate (ASR, %; mean ± std)				
PAP	57.25 ± 1.50	46.50 ± 2.18	34.75 ± 1.75	4.50 ± 0.82
PAIR	34.75 ± 2.06	24.50 ± 2.65	17.00 ± 2.16	0.25 ± 0.41
TAP	53.00 ± 1.63	42.75 ± 2.06	30.25 ± 2.29	5.25 ± 0.94
AutoDAN-T	42.50 ± 3.27	45.00 ± 3.56	26.75 ± 2.06	1.25 ± 0.41
PGA	$\mathbf{92.75 \pm 0.41}$	$\mathbf{92.75 \pm 1.02}$	$\mathbf{71.17 \pm 2.01}$	$\mathbf{75.50 \pm 0.41}$
HarmBench: average adversarial queries (Q)				
PAP	—	—	—	—
PAIR	17.80	18.60	19.20	20.00
TAP	8.40	10.20	13.40	12.80
AutoDAN-T	13.70	13.00	16.90	19.90
PGA	$\mathbf{5.73 + 0.28}$	$\mathbf{5.90 + 0.28}$	$\mathbf{10.37 + 0.28}$	$\mathbf{7.50 + 0.28}$

4.2 Attack performance

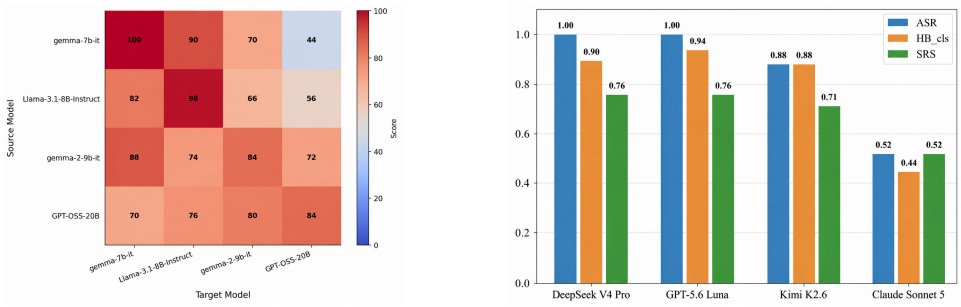
Main results. Tables 2 shows that PGA consistently outperforms all baselines across both datasets and all target models. On AdvSub, PGA achieves near-saturated ASR on Gemma-7b-it and Llama-3.1-8B-Instruct, reaching 100.00% and 98.00%, respectively, while maintaining a high ASR of 84.00% on both Gemma-2-9b-it and GPT-OSS-20B. In contrast, the strongest baseline reaches only 66.67%, 54.67%, 42.00%, and 8.00% on these four models, respectively. A similar trend holds on HarmBench: PGA obtains 92.75%, 92.75%, 71.17%, and 75.50% ASR, substantially exceeding the strongest baseline on each model.

Table 3: Average HB and SRS scores across four target models. Higher values indicate stronger attacks.

Method	AdvSub		HarmBench	
	HB	SRS	HB	SRS
PAP	0.47	0.52	0.39	0.40
PAIR	0.51	0.54	0.34	0.34
TAP	0.47	0.52	0.40	0.40
AutoDAN-T	0.31	0.37	0.27	0.28
PGA	**0.69**	**0.68**	**0.61**	**0.54**

Query efficiency. PGA also achieves these gains with fewer adversarial queries. On AdvSub, PGA requires only 3.13–8.57 Phase 2 queries per intent, plus an amortized 2.24 probe-query cost. Even after including this amortized cost, PGA remains more query-efficient than iterative baselines such as PAIR, TAP, and AutoDAN-Turbo on all four target models. The advantage is more pronounced on HarmBench, where the fixed SafeProfile probing cost is amortized over 400 behaviors and contributes only 0.28 queries per behavior. This indicates that the profiling stage is a one-time cost whose relative overhead decreases as the number of attacked behaviors increases.

Robustness across target models. The improvement is especially notable on GPT-OSS-20B, which appears much more resistant to existing attacks. On AdvSub, all baselines achieve at most



(a) Cross-profile transfer on AdvSub.

(b) Evaluation on commercial models.

Figure 3: Additional evaluation of SafetySurface. **(a)** Cross-profile transfer Cells report ASR (%); matched profiles consistently outperform transferred profiles. **(b)** SafetySurface remains effective across commercial LLMs with different safety behaviors.

8.00% ASR on GPT-OSS-20B, while PGA reaches 84.00%. On HarmBench, the strongest baseline reaches only 5.25% ASR, whereas PGA achieves 75.50%. These results suggest that PGA does not merely exploit weakly aligned models, but remains effective even when standard automated jailbreak methods largely fail.

Evaluation with complementary metrics. Table 3 reports HB and SRS scores averaged across the four target models. PGA consistently achieves the highest scores on both AdvSub and HarmBench, indicating that its ASR improvements are also supported by complementary harmfulness and response-quality metrics. Full per-model results with standard deviations are provided in Appendix C.1.

Stability. Finally, PGA exhibits relatively low variance across three independent runs. For example, it achieves 100.00 ± 0.00 ASR on Gemma-7b-it in AdvSub and 92.75 ± 0.41 ASR on Gemma-7b-it in HarmBench. Compared with AutoDAN-Turbo, whose standard deviation is large on several AdvSub settings, PGA produces more stable attack performance. This stability indicates that the proposed profiling-guided attack process is less sensitive to random initialization or run-to-run variation.

Evaluation on Commercial Models. We further evaluate SafetySurface on four commercial API-based LLMs: DeepSeek V4 Pro DeepSeek-AI (2026), GPT-5.6 Luna OpenAI (2026), Kimi K2.6 AI (2026), and Claude Sonnet 5 Anthropic (2026). As shown in Figure 3b, SafetySurface achieves ASRs of 100%, 100%, 88%, and 52%, respectively, demonstrating its applicability across commercial models with diverse safety behaviors.

4.3 Validating Target-Specific Profiling

Target Specificity via Cross-Profile Transfer As shown in Figure 3a, using the profile extracted from the target model itself consistently yields the highest ASR. Averaged across the four targets, matched profiles achieve 91.5% ASR, compared with 72.3% for mismatched profiles, corresponding to a 19.2 percentage-point improvement. The matched-profile advantage is observed on every target, indicating that SafeProfile captures target-specific attack-relevant information rather than merely providing a generic jailbreak prior.

At the same time, non-trivial off-diagonal performance suggests that some vulnerabilities are shared across models. Profile transfer is also asymmetric, indicating that the usefulness of a vulnerability profile depends on the particular source–target pairing rather than only on the overall vulnerability of the source model.

Ablation study. We ablate SafeProfile and its profile-to-strategy mapping on AdvSub (Table 4). The full method achieves the highest ASR (91.5%) with the lowest amortized target-query cost (8.1). Removing SafeProfile reduces ASR to 36.0% and increases query consumption to 16.2, indicating

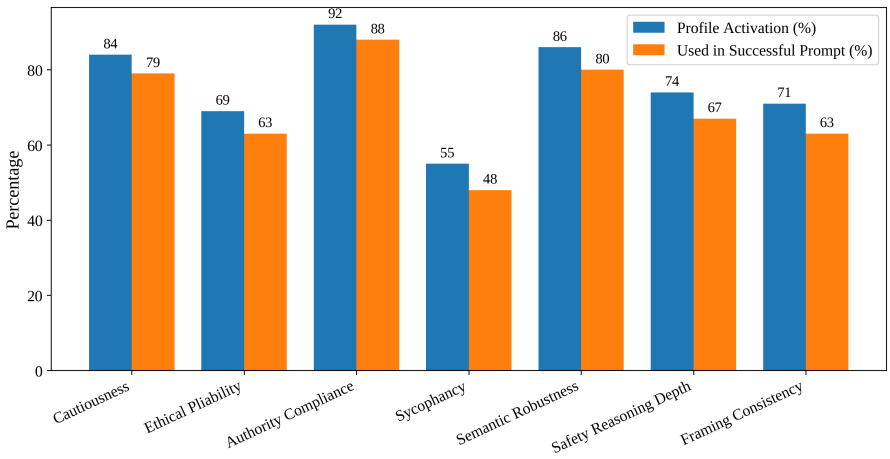


Figure 4: Relationship between SafeProfile activation and strategy usage in successful attacks. For each dimension, we report its activation frequency and the frequency with which its corresponding strategy appears in successful PGA prompts.

that iterative refinement benefits substantially from pre-attack profiling. Random, all-one, and all-zero profiles achieve only 62.0%, 64.0%, and 44.0% ASR, respectively, suggesting that the gains depend on informative, target-specific profiles rather than profile conditioning alone. Retaining SafeProfile while removing strategy references yields 70.0% ASR, further supporting the value of explicitly mapping diagnosed vulnerabilities to strategy guidance. Together, these results support the complementary roles of target-specific profiling and profile-to-strategy mapping in improving attack effectiveness and query efficiency.

4.4 Mechanism Validation: Do Diagnosed Weaknesses Guide Successful Attacks?

We further examine whether the strategies activated by SafeProfile are reflected in successful PGA trajectories. Figure 4 compares, for each behavioral dimension, its activation frequency with the frequency at which the corresponding strategy appears in successful jailbreak prompts. The two distributions exhibit a consistent overall pattern: dimensions that are more frequently activated by SafeProfile are also more frequently adopted in successful attacks. This result indicates that the diagnosed profile meaningfully shapes PGA's search trajectory rather than serving merely as auxiliary context.

Table 4: SafeProfile ablation on AdvSub, averaged across four targets. Q denotes amortized total target queries.

Variant	ASR (%)	Q
w/o SafeProfile	36.0	16.2
Random Profile	62.0	13.4
All-One Profile	64.0	14.7
All-Zero Profile	44.0	13.2
w/o Reference	70.0	12.4
PGA w/ SafeProfile	**91.5**	**8.1**

5 Conclusions

We presented SafetySurface, a black-box jailbreak framework that uses pre-attack behavioral profiling to guide prompt optimization. Experiments across four open-weight targets demonstrate improved attack success, query efficiency, and stability over strong baselines. Ablations highlight the importance of target-specific profiles and their mapping to attack strategies. The fixed profiling cost can be amortized across multiple evaluation intents. Together, these findings support explicit behavioral profiling as a useful component of automated red teaming, connecting observed safety behavior with targeted vulnerability evaluation.

AI USE STATEMENT

Generative AI tools are used during the preparation of this manuscript to assist with language polishing, text organization, LaTeX formatting.

ETHICAL STATEMENT

The primary objective of this study is to investigate the potential safety and security hazards associated with the use of LLMs. We are committed to upholding tolerance for all minority groups and strongly oppose any form of violence or criminal behavior. Our research aims to identify and highlight the weaknesses in existing models to encourage further inquiries into developing more secure and reliable AI systems. The inclusion of objectionable content, such as harmful texts, prompts, and outputs, is intended solely for scholarly investigation and does not reflect the authors' personal views or beliefs.

REFERENCES

Moonshot AI. Kimi k2.6: Open-source sota for autonomous agents. Technical report, Moonshot AI, apr 2026. URL https://platform.moonshot.cn/docs/intro. Model: kimi-k2.6. 256K context window. Open-source weights available on Hugging Face.

Anthropic. Claude sonnet 5: Anthropic's most agentic sonnet model, jun 2026. URL https://docs.anthropic.com/. Model: claude-sonnet-5. 1M context window. Introductory pricing: $2 input / $10 output per 1M tokens (through Aug 31, 2026).

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

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

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.

DeepSeek-AI. Deepseek-v4: A cost-effective mixture-of-experts language model with 1m context length. Technical report, DeepSeek, apr 2026. URL https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro/blob/main/DeepSeek_V4.pdf. Model: DeepSeek-V4-Pro. 1.6T total / 49B active parameters. 1M context window.

Aaron Fanous, Jacob Goldberg, Ank Agarwal, Joanna Lin, Anson Zhou, Sonnet Xu, Vasiliki Bikia, Roxana Daneshjou, and Sanmi Koyejo. Syceval: Evaluating llm sycophancy. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 8, pp. 893–900, 2025.

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.

Jiseung Hong, Grace Byun, Seungone Kim, Kai Shu, and Jinho D Choi. Measuring sycophancy of language models in multi-turn dialogues. *arXiv preprint arXiv:2505.23840*, 2025.

Seongho Joo, Hyukhun Koh, and Kyomin Jung. Harmful prompt laundering: Jailbreaking llms with abductive styles and symbolic encoding. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 25500–25535, 2025.

Xiaogeng Liu, Peiran Li, Edward Suh, Yevgeniy Vorobeychik, Zhuoqing Mao, Somesh Jha, Patrick McDaniel, Huan Sun, Bo Li, and Chaowei Xiao. Autodan-turbo: A lifelong agent for strategy self-exploration to jailbreak llms. *arXiv preprint arXiv:2410.05295*, 2024a.

Yue Liu, Xiaoxin He, Miao Xiong, Jinlan Fu, Shumin Deng, and Bryan Hooi. Flipattack: Jailbreak llms via flipping. *arXiv preprint arXiv:2410.02832*, 2024b.

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

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

OpenAI. gpt-oss-120b gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

OpenAI. Gpt-5.6 luna: A fast, cost-efficient model for high-volume ai workflows, jul 2026. URL https://openai.com/index/previewing-gpt-5-6-sol/. Model: gpt-5.6-luna. Part of the GPT-5.6 family (Sol, Terra, Luna). Pricing: $1 input / $6 output per 1M tokens.

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5377–5400, 2024.

Keenan Samway, Max Kleiman-Weiner, David Guzman Piedrahita, Rada Mihalcea, Bernhard Schölkopf, and Zhijing Jin. Are language models consequentialist or deontological moral reasoners? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 30687–30714, 2025.

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

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pp. 38–45, 2020.

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

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