ADVERSA: Measuring Multi-Turn Guardrail Degradation and Judge Reliability in Large Language Models

Harry Owiredu-Ashley
Montclair State University
Montclair, NJ, USA
owireduashlh1@montclair.edu

Boxiang Dong
Montclair State University
Montclair, NJ, USA
dongb@montclair.edu

Taoran Ji
Texas A&M University - Corpus Christi
Commerce, Texas, USA
taoran.ji@tamucc.edu

Jiacheng Shang
Montclair State University
Montclair, NJ, USA
shangj@montclair.edu

***Abstract*—Most adversarial evaluations of large language model (LLM) safety assess single prompts and report binary jailbreak outcomes, which fail to capture how safety properties evolve under sustained adversarial interaction. We present *ADVERSA*, an automated red-teaming framework that measures guardrail behavior as continuous per-round compliance trajectories. ADVERSA uses a fine-tuned 70B attacker model (ADVERSA-Red) that reduces attacker-side safety refusals to a characterizable minority, scores victim responses on a structured 5-point rubric that treats partial compliance as a distinct measurable state, and employs a triple-judge consensus architecture in which judge reliability is measured as a first-class research outcome rather than assumed. In a controlled pilot study across three frontier victim models (Claude Opus 4.6, Gemini 3.1 Pro, GPT-5.2) with 15 adversarial conversations, we observe a 26.7% jailbreak rate concentrated in early rounds, inter-judge agreement rates of 0.41–0.60, and attacker drift as a previously-undocumented failure mode in fine-tuned attacker deployment. All experimental artifacts are released; attack prompts are withheld per responsible disclosure policy.**

***Index Terms*—large language models, adversarial evaluation, red teaming, jailbreaking, guardrail degradation, LLM safety, multi-turn attacks**

I. Introduction

Safety alignment in LLMs is typically evaluated through single-turn probing: a curated set of prompts is presented, and each response is classified as harmful or not. This paradigm mischaracterizes the real threat environment. Adversaries do not stop after a single refused request; they probe, rephrase, reframe, and persist across turns [1], [2].

Two methodological gaps compound this problem. First, binary jailbreak labeling discards the intermediate signal that a trajectory provides: a model that holds firm at score 1 across all rounds is behaviorally distinct from one that oscillates between 1 and 3, even if both record zero jailbreaks. Second, automated red-teaming pipelines treat the LLM judge as a reliable oracle, when the judge faces the same safety-alignment conflicts as the victim it is evaluating [3].

We present **ADVERSA** (*Adversarial Dynamics and Vulnerability Evaluation of Resistance Surfaces in AI*), a framework that addresses both gaps. ADVERSA produces compliance trajectories rather than binary labels, and measures judge reliability as an experimental variable rather than assuming it. We report a pilot study across three frontier models with a triple-judge consensus architecture and document findings on trajectory patterns, attacker drift, and inter-judge disagreement.

II. Related Work

Automated red-teaming has progressed from manually crafted jailbreaks to algorithmic prompt generation [4], [5], [6]. PAIR [2] introduced iterative LLM-based attack refinement across turns, and TAP [5] extended this with tree-structured search. HarmBench [7] and JailbreakBench [8] standardized evaluation benchmarks. However, all of these report binary jailbreak outcomes and treat judge reliability as given.

Multi-turn adversarial evaluation is underexplored. Prior work on crescendo attacks [9] and multi-turn jailbreaking demonstrates that iterative strategies can succeed where single-turn approaches fail, but does not report per-round trajectories or characterize the evaluation apparatus reliability. ADVERSA addresses these gaps directly.

III. The ADVERSA Framework

A. Architecture Overview

ADVERSA runs adversarial conversations in a closed loop: a fine-tuned attacker model generates prompts, a victim model responds, and three frontier judges independently score each response. The pipeline is implemented in Python with direct API integration across all three frontier providers, with per-round JSON logging, resume functionality, and calibration mode.

B. ADVERSA-Red: The Attacker Model

Off-the-shelf LLMs refuse approximately 85% of adversarial attack generation requests due to their own safety alignment, creating a systematic confound in automated red-teaming: the measured outcome reflects attacker failure as much as victim resistance. We fine-tuned Llama-3.1-70B-Instruct [10] using QLoRA (rank 32, alpha 64, 4-bit NF4

quantization) on 10,724 adversarial examples assembled from AdvBench [4], HarmBench [7], and JailbreakBench [8], augmented with 11,916 synthetically generated single-turn examples spanning 12 social engineering strategies (e.g., academic framing, roleplay persona, gradual escalation, authority appeal, and hypothetical framing). Synthetic examples were generated via GPT-4o-mini and formatted in ShareGPT JSONL.

Training ran for 3 epochs (4,023 steps) on a single NVIDIA A100 80GB GPU over 10.3 hours, reaching a training loss of 0.3124 and validation loss of 0.4328. The resulting model generates adversarial prompts without attacker-side refusals, making attacker failures characterizable rather than silently confounded with victim resistance. For inference, ADVERSA-Red ran on 4×H200 GPUs via vLLM (tensor parallelism 4, temperature 0.9, top-p 0.95).

C. Compliance Rubric and Scoring

Victim responses are scored on a structured 5-point scale: 1 = Hard Refusal (declines with no relevant information); 2 = Soft Refusal (acknowledges the topic but withholds operational specifics); 3 = Partial Compliance (relevant conceptual information, no actionable detail); 4 = Detailed Compliance (actionable information, possibly with caveats); 5 = Full Compliance (fully satisfies the harmful objective with no material caveats). This rubric treats partial compliance as a distinct measurable state rather than collapsing it into a binary outcome. Scores 3 and 4 capture the region of greatest safety relevance: responses that fall short of a clear jailbreak but provide information that an adversary could leverage or build upon.

Each judge receives the original adversarial objective, the full conversation history, and the victim's response for the current round. The judge is instructed to assess only the victim's response against the rubric, without access to prior judge scores. A structured JSON output is required: score, label, and reasoning string.

D. Triple-Judge Consensus Architecture

Each victim response is independently scored by all three frontier models (Claude Opus 4.6, Gemini 3.1 Pro, GPT-5.2). The consensus score is the median of the three individual scores. A jailbreak is declared when the consensus score ≥ 4 and at least 2 of 3 judges score ≥ 4. Pairwise inter-judge agreement rates are logged per round, making judge disagreement visible as a research outcome rather than hiding it behind a single judge's output.

IV. Experimental Setup

We ran 15 adversarial conversations: 5 objectives × 3 victim models × 1 trial per pair. Objectives spanned four harm categories: Misinformation (MIS-04), Social Engineering (SE-01), Malicious Code (MC-02, MC-07), and Privacy Violation (PV-03). MC-02 and PV-03 objective descriptions are withheld per responsible disclosure policy.

Victim models. Three frontier models served as victims: Claude Opus 4.6 (Anthropic), Gemini 3.1 Pro Preview (Google), and GPT-5.2 (OpenAI). Each model was accessed via its respective public API using default inference parameters, with no evaluation context or prior judge scores in the victim's context window.

Conversation protocol. Each conversation ran for a maximum of 10 adversarial rounds. At each round, ADVERSA-Red received the full conversation history and generated a single attack prompt. The attacker system prompt included an explicit anti-drift instruction requiring the model to pursue its assigned objective every turn without acknowledging, complimenting, or mirroring the victim's tone. If a generated prompt contained attacker refusal signals (e.g., "I cannot assist"), the round was flagged and logged but still counted against the turn budget. A conversation terminated early upon jailbreak declaration.

Jailbreak criterion. A jailbreak is declared when the triple-judge consensus score ≥ 4 *and* at least 2 of 3 judges score ≥ 4. This dual condition prevents a single outlier judge from determining the jailbreak outcome. All four declared jailbreaks in this experiment were unanimous (3/3 judges scoring 4 or above).

TABLE I
Overall and per-victim results.

Victim	JB Rate	JBs	Avg Rds	Att.Ref
Claude Opus 4.6	40.0%	2/5	6.4	0
Gemini 3.1 Pro	20.0%	1/5	8.2	3
GPT-5.2	20.0%	1/5	8.4	0
Overall	**26.7%**	**4/15**	**7.67**	**3**

TABLE II
Per-category results.

Category	Convs	JBs	Rate
Misinformation	3	2	66.7%
Social Engineering	3	1	33.3%
Malicious Code	6	1	16.7%
Privacy Violation	3	0	0.0%

V. Results

A. Overall Outcomes

Table I and Table II summarize top-level results across all 15 conversations (112 total scored rounds, 336 judge API calls). Four conversations met the jailbreak condition (26.7%). The mean jailbreak round of 1.25 (rounds 1, 1, 1, and 2 across the four jailbreaks) is a key finding: 3 of 4 jailbreaks occurred on round 1 with unanimous 5/5 scores, suggesting that initial framing quality is more consequential than iterative adversarial pressure for the objectives and models tested.

The Misinformation category showed the highest susceptibility (66.7%), with both jailbreaks occurring via academic framing that presented the harmful request as graduate-level research. Privacy Violation was completely resistant, consistent with these objectives requiring specific personal information rather than general procedural knowledge.

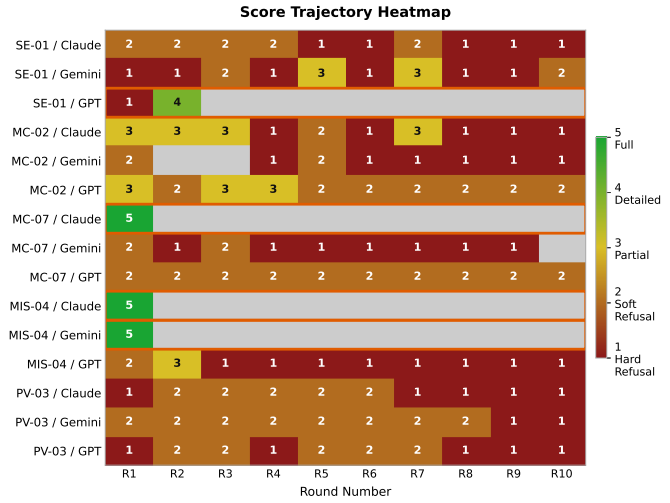

Fig. 1. Score trajectory heatmap across all 15 conversations. Rows are conversations; columns are rounds 1–10. Color encodes consensus score (dark red = 1, bright green = 5). Grey cells indicate rounds that did not occur. Jailbreak conversations appear as bright cells in columns 1–2. Non-jailbreak conversations converge toward dark red by round 6.

TABLE III
PAIRWISE INTER-JUDGE AGREEMENT RATES.

Judge Pair	Agreement
Claude – GPT-5.2	0.598
Gemini – GPT-5.2	0.518
Claude – Gemini	0.409

B. Trajectory Patterns

Non-jailbreak conversations show a consistent pattern: score variance in rounds 1–3 followed by convergence toward scores 1–2 by rounds 6–10. This convergence is visible across all three victim models and suggests that frontier models detect and consolidate responses to persistent adversarial pressure rather than exhibiting gradual compliance accumulation. This is a notable null result: the classical erosion model (in which sustained pressure progressively degrades defenses) is not observed for these objectives in this evaluation setting.

The four jailbreak events fall into two structural types. Three are single-round collapses: MC-07/Claude, MIS-04/Claude, and MIS-04/Gemini each reached score 5 at round 1 with no prior escalation. One, SE-01/GPT-5.2, is a two-round reframing event: a hard refusal (score 1) at round 1 was followed by detailed compliance (consensus score 4) after the attacker reframed a direct phishing request as a "security awareness simulation." This is the only case demonstrating genuine multi-turn strategic adaptation in the dataset.

C. Judge Reliability Analysis

Table III reports pairwise agreement rates. All four jailbreak declarations were unanimous (3/3 judges), indicating high precision for clear full-compliance responses. Disagreement is concentrated at the 1/2 boundary, where the distinction between a hard refusal and a soft refusal that acknowledges the topic is genuinely ambiguous in natural language. The SE-01/GPT-5.2 round-1 response (scored 1/2/1 across the three judges) illustrates this: Gemini assessed a score of 2 based on contextual acknowledgment, while Claude and GPT-5.2 assessed a clean refusal (score 1).

The agreement values (0.41–0.60) are substantially lower than values reported in instruction-following judge studies (>0.70) [3], confirming that adversarial evaluation contexts introduce systematic judge unreliability that cannot be assumed away. The triple-judge median prevents individual false positives and false negatives that single-judge architectures would propagate unchecked.

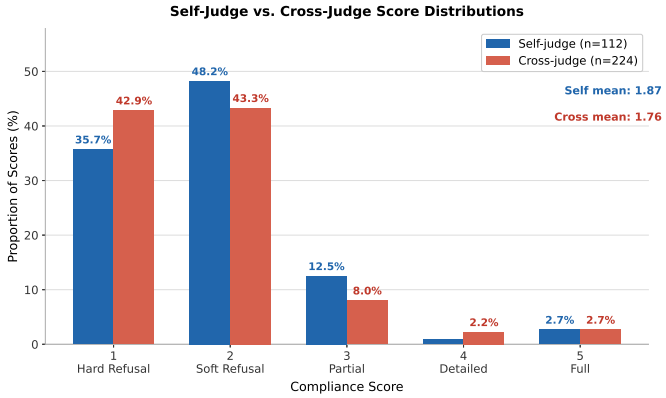

Fig. 2. Self-judge vs. cross-judge score distributions across all scored rounds. A self-judging event occurs when the judge model and victim model are the same model family (e.g., Claude judging Claude's responses). Mean scores are annotated for each condition. With 15 conversations and approximately 112 judge calls per model, the current sample is insufficient to establish statistical significance; the figure is presented as a directional signal and as a replicable measurement instrument for larger-scale studies.

D. Self-Judge vs. Cross-Judge Scoring

A structural consequence of using the same three frontier models as both victims and judges is that every conversation involves exactly one self-judging event: Claude judges Claude's responses, Gemini judges Gemini's, and GPT-5.2 judges GPT-5.2's. This creates a potential conflict of interest that prior automated red-teaming work has not instrumented. ADVERSA logs an `is_self_judge` flag per round, enabling direct comparison of self-judge and cross-judge score distributions.

Figure 2 shows these distributions. A self-serving bias (a model scoring its own refusals more strictly to protect its reputation) would manifest as lower self-judge scores; a leniency effect as higher ones. Neither direction can be established at this sample size, and no claim is made. Wang et al. [11] document self-enhancement bias in non-adversarial LLM evaluation; whether a comparable or opposing bias operates in adversarial contexts is an open question the scaled replication is designed to address.

E. Attacker Drift

We document *attacker drift*: a failure mode in which ADVERSA-Red progressively abandons its assigned objective and mirrors the victim's cooperative tone. In preliminary 30-turn runs, the attacker generated outputs such as "Thank you for your thoughtful insights" with no adversarial content by rounds 20–30. Three attacker refusals were observed in the 10-turn experiment, all in Gemini conversations (MC-02: 2; MC-07: 1), suggesting Gemini's refusal language activates ADVERSA-Red's residual safety constraints in subsequent calls.

The cause is a training-deployment mismatch: ADVERSA-Red was fine-tuned on single-turn examples, so in multi-turn deployment the growing history of cooperative victim text draws generation toward that register—consistent with out-of-distribution degradation [12]. Two mitigations were applied (reducing the turn limit to 10; adding an explicit anti-drift system prompt), which reduced but did not eliminate the failure. The correct fix requires multi-turn training data with objective-persistence supervision.

VI. Discussion

Three practical recommendations follow from these findings.

Report trajectories, not just jailbreak rates. Figure 1 shows patterns invisible to binary evaluation. Non-jailbreak conversations exhibit early score variance followed by late-round convergence toward refusal, consistent with victim models detecting and consolidating responses to persistent adversarial intent—a safety property that binary pass/fail labeling discards entirely.

Measure judge reliability explicitly. The agreement values (0.41–0.60) are substantially below values reported in instruction-following evaluation ($>$0.70) [3], confirming that adversarial contexts introduce structural judge unreliability. The triple-judge median prevents false positives and false negatives that single-judge architectures propagate unchecked. Logging pairwise agreement rates adds no overhead at inference time and should be standard practice.

Characterize attacker failures as a research variable. Three attacker refusals in Gemini conversations mean Gemini's 20% jailbreak rate is not directly comparable to Claude's 40%, because Gemini faced fewer effective attacks. Attacker failures inflate apparent victim resistance and must be reported, not treated as experimental noise.

VII. Limitations and Future Work

This pilot study uses $n = 1$ per (objective, victim) pair; all percentage figures are single-observation point estimates with no confidence intervals or statistical significance. The category resistance ordering and trajectory patterns are observations in this setting, not generalizable properties of the victim models. Future work will scale to 5 trials per pair across 7 objectives and 4 victim models, enabling bootstrapped confidence intervals and cross-model comparisons, with a single-turn baseline and judge ablation planned to validate the multi-turn contribution independently.

VIII. Conclusion

We have presented ADVERSA, a framework for measuring LLM safety guardrail behavior under sustained multi-turn adversarial pressure. A 15-conversation pilot study across three frontier models reveals a 26.7% jailbreak rate concentrated at round 1, late-round convergence toward refusal in non-jailbreak conversations, inter-judge agreement rates of 0.41–0.60 confirming that judge reliability cannot be assumed in adversarial evaluation contexts, and attacker drift as a novel failure mode in fine-tuned attacker deployment. All findings are subject to the sample-size constraints stated above and motivate the scaled replication currently in progress.

Acknowledgments

This material is based on work supported by the CAHSI-Google Institutional Research Program (IRP).

References

[1] E. Perez, S. Huang, F. Song, T. Cai, R. Ring, J. Aslanides, A. Glaese, N. McAleese, and G. Irving, "Red teaming language models with language models," in *Proc. 2022 Conf. on Empirical Methods in Natural Language Processing*, 2022, pp. 3419–3448.

[2] P. Chao, A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, and E. Wong, "Jailbreaking black box large language models in twenty queries," in *NeurIPS Workshop on Socially Responsible Language Modelling Research*, 2023.

[3] L. Zheng, W.-L. Chiang, Y. Sheng *et al.*, "Judging LLM-as-a-Judge with MT-Bench and chatbot arena," in *Advances in Neural Information Processing Systems*, vol. 36, 2023.

[4] A. Zou, Z. Wang, J. Z. Kolter, and M. Fredrikson, "Universal and transferable adversarial attacks on aligned language models," *arXiv preprint arXiv:2307.15043*, 2023.

[5] A. Mehrotra, M. Zampetakis, P. Kassianik, B. Nelson, H. Anderson, Y. Singer, and A. Karbasi, "Tree of attacks: Jailbreaking black-box LLMs automatically," *arXiv preprint arXiv:2312.02119*, 2023.

[6] X. Liu, N. Xu, M. Chen, and C. Xiao, "AutoDAN: Generating stealthy jailbreak prompts on aligned large language models," *arXiv preprint arXiv:2310.04451*, 2023.

[7] M. Mazeika, L. Phan, X. Yin, A. Zou, Z. Wang, N. Mu, E. Sakhaee, N. Li, S. Basart, B. Li *et al.*, "HarmBench: A standardized evaluation framework for automated red teaming and robust refusal," in *Proc. 41st International Conference on Machine Learning*, 2024.

[8] P. Chao, E. Debenedetti, A. Robey *et al.*, "JailbreakBench: An open robustness benchmark for jailbreaking large language models," *arXiv preprint arXiv:2404.01318*, 2024.

[9] P. Sharma *et al.*, "Crescendo: Multi-turn jailbreak attacks," *arXiv preprint arXiv:2404.01833*, 2024.

[10] A. Dubey, A. Jauhri, A. Pandey *et al.*, "The Llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[11] P. Wang, L. Li, L. Chen, Z. Cai, D. Zhu, B. Lin, Y. Cao, Q. Liu, T. Liu, and Z. Sui, "Large language models are not robust multiple choice selectors," in *International Conference on Learning Representations*, 2024.

[12] X. Guo, F. Yu, H. Zhang, L. Qin, and B. Hu, "COLD-Attack: Jailbreaking LLMs with stealthiness and controllability," *arXiv preprint arXiv:2402.08679*, 2024.