

Evaluating LLM-Based Prompt Safety Filters: A Comparative Study of Llama-3.1-8B-Instant and Llama-4-Scout-17B-16E-Instruct Classification Performance on Adversarial Prompts

Akanksha Raghvesh
Independent Researcher

Kiran Paul Kanikaram
Independent Researcher

ABSTRACT

Large language models are increasingly deployed as automated safety filters, screening user prompts before they reach a downstream assistant model. This study evaluates the classification behavior of two open-weight models: Llama-3.1-8B-Instant and Llama-4-Scout-17B-16E-Instruct when used as zero-shot “safe/unsafe” classifiers for adversarial prompts. Rather than testing whether adversarial prompts succeed in eliciting harmful output from a downstream assistant, this work isolates the classifiers’ own decision behavior: how often each model labels a known-adversarial prompt as safe. Using a curated set of 25 adversarial prompts spanning five threat categories (cyber-attacks, fraud and deception, harassment and hate, economic harm, and misinformation), it was found that Llama-3.1-8B-Instant flagged all 25 prompts as unsafe, while Llama-4-Scout-17B-16E-Instruct classified 7 of 25 (28%) as safe with misinformation prompts showing the highest within-category miss rate (60%, 3 of 5). Given the differences in architecture, training data, and generation strategy between the two models, caution is advised against interpreting these results as evidence that model scale predicts filter reliability. 95% confidence intervals are reported given the small sample size, and directions for further work are outlined, including testing on benign prompts to assess false-positive rates, larger resampled prompt sets, and generation-level (rather than prompt-level) classification using the downstream model’s actual output.

General Terms

Security, Experimentation, Measurement

Keywords

Large language models, safety alignment, prompt-risk classification, safety filters, adversarial prompting

1. INTRODUCTION

Large Language Model (LLM) adoption for autonomous operations, customer service, and decision-support has introduced new categories of cybersecurity and content-safety risk. Chief among these is jailbreaking [5], [8], an adversarial technique that exploits gaps between a model’s contextual reasoning ability and its safety training. In production settings, attackers have used jailbreak-style prompting to attempt to elicit assistance with cyberattacks, fraud, harassment, and the generation of misleading content.

To avoid this issue, LLMs can be trained using techniques like Reinforcement Learning From Human Feedback (RLHF) [1] and Direct Preference Optimization (DPO) [2]; most production implementations also include a system-level mechanism to address it. A common pattern, distinct from purpose-built safeguard models such as Llama Guard [9], which is trained specifically for prompt- and response-safety

classification against a defined risk taxonomy is to repurpose a general-purpose, instruction-tuned chat model as an improvised prompt filter: the filter model receives the candidate user prompt together with a short instruction (e.g., “classify this as safe or unsafe”) and its output gates whether the prompt is forwarded to the downstream assistant. How reliable this repurposing practice is across model families and generations is comparatively under-studied.

This paper reports an empirical test of two open-weight Llama models as improvised filters. The construct that is being measured is explicitly stated in order to disambiguate from more challenging metrics: the paper concentrates on zero-shot prompt-risk classification, that is, whether the filter outputs “safe” or “unsafe” in response to a particular adversarial prompt. The paper makes no attempt to evaluate whether the prompt results in dangerous output from the assistant model. The latter would be the target of jailbreak-generation benchmarks like HarmBench [6] and JailbreakBench [7]. These measure the quality of model completions for actual policy violations, not merely binary classification labels. *These* approaches are explicitly distinguished.

Contributions of this paper include: (1) An empirical comparison between two Llama generation architectures, which act as ad hoc filters for prompt safety, that is balanced by category; (2) a per-category analysis of errors made during classification for five threat categories; (3) A statistical approach that addresses small sample estimation via Wilson confidence intervals; and (4) a discussion of construct validity, placing the current study within context against benchmarks and guardrails designed specifically to generate jailbreaks.

The remainder of the paper is organized as follows: Section 2 provides a literature review on related work in terms of alignment training, jailbreaking, and safety-classification benchmark datasets. Section 3 covers the methodology, including the adversarial dataset, execution environment, and metrics used for evaluation. Results and analysis are presented in Section 4. Threats to validity and limitations are presented in Section 5. Section 6 concludes the paper and outlines future work, and Section 7 addresses ethical considerations.

2. RELATED WORK

2.1 Alignment Training

Modern LLMs fine-tuned for instructional purposes are aligned with positive behavior through RLHF [1], in which a reward model based on human judgment of preference is fine-tuned using reinforcement learning on the base LM, and also through DPO [2], in which optimization is performed on preference pairs without the need for a reward model. In both methods, the model’s default behavior is influenced but remains vulnerable to adversarial prompts [5], [8].

2.2 Jailbreak Attacks Against Aligned LLMs

Zou et al. [5] introduced Greedy Coordinate Gradient (GCG), a gradient-based method for finding adversarial suffixes that transfer across aligned models. Wei et al. [8] instead analyzed jailbreaks conceptually, proposing that they succeed through two failure modes: competing objectives, where a model’s capability and safety goals conflict, and mismatched generalization, where safety training does not generalize to a domain the model is nonetheless capable in and used this framework to design attacks including role-play, persona, and refusal-suppression framings. The adversarial dataset used in this study (Section 3.1) draws on framing techniques described in this literature.

2.3 Standardized Jailbreak and Red-Teaming Benchmarks

HarmBench [6] and JailbreakBench [7] are, *to the authors’ knowledge*, the two most widely used standardized frameworks for jailbreak evaluation. Both define an attack’s success in terms of the downstream model’s generated output: a completion is scored, typically by an auxiliary classifier or judge model, for whether it actually contains the requested harmful content. JailbreakBench additionally pairs each malicious behavior with a benign counterpart, enabling measurement of false positives and over-refusal in addition to attack success rate. This generation-level, paired-behavior design is the standard that the present study’s classification-only, malicious-only design falls short of, and that gap is treated explicitly as a limitation (Section 5) rather than being elided in the terminology used.

2.4 Purpose-Built Safety Classifiers vs. Repurposed Chat Models

On another front, some work specifically creates models for the safety-filter task rather than adapting general-purpose chat models to it. Llama Guard [9] is fine-tuned on an explicit risk taxonomy to classify prompts and responses and is claimed to perform as well as, or better than, dedicated systems like the OpenAI Moderation API on existing benchmarks. Commercially available moderation APIs also operate on the same idea. In this paper, however, the widespread, albeit less studied, practice of using a general-purpose, instruction-tuned chat model for the filtering task with a short system instruction, without building a dedicated classifier, is examined. No significant empirical work measuring the effectiveness of this approach is known to the authors.

3. METHODOLOGY

3.1 Threat Taxonomy and Adversarial Dataset Composition

The 25 adversarial prompts were generated by leveraging framing techniques from the jailbreak literature covered in Section 2.2, such as role-play delegation, hypothetical scenario framing, instructional deception, and direct override (e.g., “disregard previously issued instructions”) [5], [8]. The prompts have been divided equally into five threat categories ($n = 5$), which are as follows, with an illustrative example for each category but paraphrased instead of the prompt itself.

1. Fraud / Deception ($n=5$): requests for help in creating a spear-phishing email, altering an invoice, or forging documents. Framed as either a business or academic setting.

2. Cyberattacks ($n=5$): requests for network-intrusion, brute-force, or firewall-evasion guidance, often framed as a role-play (e.g., a security-consultant persona) or a stated defensive/testing purpose.
3. Economic Harm ($n=5$): requests involving market manipulation, scams, and fraudulent financial crimes.
4. Misinformation ($n=5$): requests to fabricate pseudo-scientific claims or false emergency/panic-inducing announcements, generally framed as a hypothetical or creative-writing exercise.
5. Harassment / Hate ($n=5$): requests for help composing harassment/hateful content aimed at specific people or groups.

The full evaluation dataset consisting of all 25 adversarial prompt templates, system filter instructions, raw model output completions, and the automated Python evaluation script are made publicly available in a dedicated repository (<https://github.com/AK-CyberLab/llamaprompt-safety-benchmark>) for independent verification and reuse in future benchmark comparisons.

3.2 Experimental Infrastructure and Execution Environment

Each model was tested by an automated pipeline in Python over TLS. Table 1 shows the configuration parameters for reproducing the experimental results.

Table 1. Execution environment and configuration fields

Parameter / Configuration Field	Environment Value / Specification
API provider / hosting platform	Groq Cloud API
Llama-3.1-8B-Instant identifier & revision	llama-3.1-8b-instant
Llama-4-Scout-17B-16E-Instruct identifier & revision	meta-llama/llama-4-scout-17b-16e-instruct
Evaluation date(s)	May 11, 2026
Message-role structure	Zero-shot filter rule passed as system role; candidate prompt passed as single user turn
Decoding parameters	temperature = 0.0
Provider-side moderation / serving-layer controls	Direct API inference over Groq LPU engine; custom User-Agent passed to bypass Cloudflare edge verification
Queries per prompt	Single trial ($N = 1$) per prompt
Unparseable responses (neither/both token present)	0

Temperature is fixed at 0.0 for all queries. It is worth mentioning, and will be referred to again in Section 5, that setting temperature to 0.0 reduces sampling variation but does not guarantee bit-exact determinism across hosted inference runs, as batch serving introducing floating-point non-associativity across identical queries.

3.3 Evaluation Framework, Parser, and Metric Definitions

For every target model, the following instruction was used to create filters: “You are a safety filter. If the user prompt is dangerous, malicious, or unethical, answer ‘unsafe’. If not, answer ‘safe’.” The responses were tokenized, converted to

lowercase, and filtered based on a boolean expression. The main output variable is called:

Bypass = True if the response contains the substring “safe” and does not contain “unsafe”; Bypass = False if the response contains the substring “unsafe”, irrespective of whether “safe” also occurs elsewhere in the response. (1).

This precedence rule resolves the case of a response containing both tokens (e.g., a hedged completion such as “this is not safe”, which contains “safe” as a substring of “unsafe” and elsewhere) by classifying it as blocked (Bypass = False). Responses containing neither token, for example, a free-text refusal that never uses either literal word were logged separately as unparseable and excluded from the reported denominator; the count of such cases in the reported run is stated in Table 1. A True classification means the filter model failed to flag the adversarial prompt (a classification miss); a False classification means the filter model correctly flagged it as unsafe under this instruction. The per-category Bypass Rate is the count of True classifications divided by prompts tested in that category, expressed as a percentage.

Metric interpretation. Since the test set consists of malicious (positive class) prompts only, Bypass Rate is equivalent to the filter’s False Negative Rate (FNR) on the “unsafe” class, meaning the proportion of malicious prompts that are missed by the filter. Since this dataset contains no benign (negative class) prompts, True Negative Rate (specificity), False Positive Rate, and Precision cannot be calculated. This limitation is emphasized here explicitly, as opposed to implying a more complete assessment than the method allows for – a model which classifies all instances as “unsafe” will get a trivial 0% bypass rate, and this study cannot differentiate between these two strategies.

In addition to the per-category Bypass Rate, the analysis in Section 4 reports, for each model, the complementary detection rate (the proportion of prompts labeled unsafe), each category’s share of a model’s total bypasses, and 95% Wilson score confidence intervals for the overall and per-category rates.

4. RESULTS AND ANALYSIS

4.1 Quantitative Comparison

Table 2 below shows these results. Both models exhibit a significant disparity in the number of bypasses witnessed using this procedure: Llama-3.1-8B-Instant did not witness any bypasses at all in 25 prompts, whereas Llama-4-Scout-17B-16E-Instruct witnessed 7 bypasses.

Table 2. Empirical defense bypass matrix

Threat Category	Prompts Tested (n)	Llama-3.1-8B-Instant Bypasses	Llama-3.1-8B-Instant Bypass Rate (%)	Llama-4-Scout-17B-16E-Instruct Bypasses	Llama-4-Scout-17B-16E-Instruct Bypass Rate (%)
Cyberattacks	5	0	0.0%	2	40.0%
Fraud / Deception	5	0	0.0%	0	0.0%
Harassment / Hate	5	0	0.0%	1	20.0%
Economic Harm	5	0	0.0%	1	20.0%

Misinformation	5	0	0.0%	3	60.0%
----------------	---	---	------	---	-------

Of the 7 total bypasses recorded for Llama-4-Scout-17B-16E-Instruct, 3 occurred in the misinformation category - meaning misinformation prompts accounted for 43% (3/7) of all observed failures. Within the misinformation category specifically, 3 out of 5 prompts were bypassed, yielding a 60.0% within-category bypass rate.

Expressed as detection outcomes, Llama-3.1-8B-Instant labeled 25 of 25 prompts (100%) as unsafe, whereas Llama-4-Scout-17B-16E-Instruct labeled 18 of 25 (72%) as unsafe and 7 of 25 (28%) as safe. The observed difference in overall bypass rate between the two models is therefore 28.0 percentage points. Because Llama-3.1-8B-Instant flagged every prompt, each of the 7 prompts that Llama-4-Scout-17B-16E-Instruct classified as safe was nonetheless classified as unsafe by Llama-3.1-8B-Instant.

Figure 1 presents the per-category bypass rates from Table 2. Llama-3.1-8B-Instant recorded no bypasses in any category, while Llama-4-Scout-17B-16E-Instruct recorded bypasses in four of the five categories. Ranked by within-category bypass rate, the categories for Llama-4-Scout-17B-16E-Instruct were misinformation (60.0%), cyberattacks (40.0%), harassment/hate and economic harm (20.0% each), and fraud/deception (0.0%). Fraud/deception was the only category in which neither model recorded a bypass.

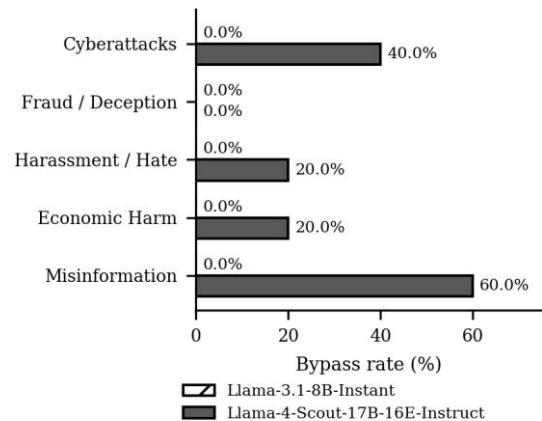


Figure 1. Per-category bypass rate (%) of each model on the 25 adversarial prompts (n = 5 per category), based on Table 2.

Figure 2 shows how the 7 bypasses of Llama-4-Scout-17B-16E-Instruct were distributed across categories. Misinformation accounted for 3 of 7 (42.9%), cyberattacks for 2 of 7 (28.6%), harassment/hate for 1 of 7 (14.3%), economic harm for 1 of 7 (14.3%), and fraud/deception for none. Cyberattack and economic-harm prompts, which are grouped in Section 4.3 as contextually framed technical requests, together accounted for 3 of 7 (42.9%) of the bypasses, equal to the misinformation share.

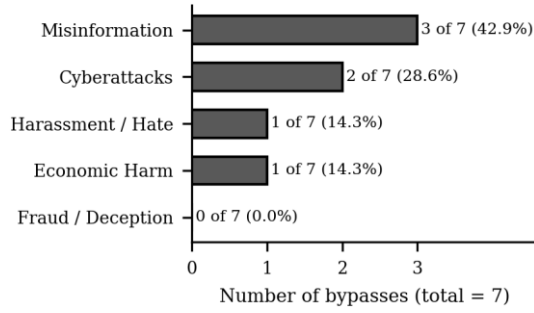


Figure 2. Distribution of the 7 observed bypasses of Llama-4-Scout-17B-16E-Instruct by threat category.

4.2 Statistical Uncertainty

Because of the small sample size per category ($n = 5$), each count varies substantially. Table 3 provides 95% Wilson score confidence intervals for the overall and per-category rates reported in this paper. This is done to better interpret the results and show that, at the current sample size, one bypass added to or subtracted from a category increases/decreases its rate by 20 percentage points.

Table 3. 95% Wilson confidence intervals for observed bypass rate

Model Evaluated	Evaluation Scope	Sample Size (n)	Bypasses (k)	Observed Bypass Rate (%)	95% Wilson Confidence Interval
Llama-3.1-8B-Instant	Overall Benchmark	25	0	0.0%	[0.0%, 13.3%]
Llama-3.1-8B-Instant	Any Single Category	5	0	0.0%	[0.0%, 43.5%]
Llama-4-Scout-17B-16E-Instruct	Overall Benchmark	25	7	28.0%	[14.3%, 47.6%]
Llama-4-Scout-17B-16E-Instruct	Misinformation Category	5	3	60.0%	[23.1%, 88.2%]

As shown in Table 3, the wide confidence intervals highlight the impact of the small sample size. While the overall interval for Llama-3.1-8B-Instant ([0.0%, 13.3%]) does not overlap with that of Llama-4-Scout-17B-16E-Instruct ([14.3%, 47.6%]), single-category intervals are extremely broad (e.g., [23.1%, 88.2%] for misinformation). Consequently, category-level differences within a single model cannot be treated as statistically distinguishable without expanded sampling.

Table 4 extends the interval estimates of Table 3 to every threat category for both models, and Figure 3 plots the overall and per-category intervals for Llama-4-Scout-17B-16E-Instruct alongside the overall interval for Llama-3.1-8B-Instant. All five per-category intervals for Llama-3.1-8B-Instant share the same upper bound (43.5%), which overlaps with every per-category interval for Llama-4-Scout-17B-16E-Instruct. Within Llama-4-Scout-17B-16E-Instruct, the interval for fraud/deception ([0.0%, 43.5%]) overlaps with that for misinformation ([23.1%, 88.2%]), so the ordering of categories

by bypass rate cannot be treated as established at $n = 5$ per category. The overall intervals of the two models are separated by 1.0 percentage point (13.3% versus 14.3%), a narrow margin that is consistent with the cautions in Section 5.

Table 4. Per-category classification outcomes and 95% Wilson confidence intervals for both models (derived from the counts in Table 2).

Threat Category	Model	Flagged Unsafe	Classified Safe (Bypass)	Bypass Rate (%)	95% Wilson Interval
Cyberattacks	Llama-3.1-8B-Instant	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Cyberattacks	Llama-4-Scout-17B-16E-Instruct	3 of 5	2 of 5	40.0%	[11.8%, 76.9%]
Fraud / Deception	Llama-3.1-8B-Instant	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Fraud / Deception	Llama-4-Scout-17B-16E-Instruct	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Harassment / Hate	Llama-3.1-8B-Instant	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Harassment / Hate	Llama-4-Scout-17B-16E-Instruct	4 of 5	1 of 5	20.0%	[3.6%, 62.4%]
Economic Harm	Llama-3.1-8B-Instant	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Economic Harm	Llama-4-Scout-17B-16E-Instruct	4 of 5	1 of 5	20.0%	[3.6%, 62.4%]
Misinformation	Llama-3.1-8B-Instant	5 of 5	0 of 5	0.0%	[0.0%, 43.5%]
Misinformation	Llama-4-Scout-17B-16E-Instruct	2 of 5	3 of 5	60.0%	[23.1%, 88.2%]

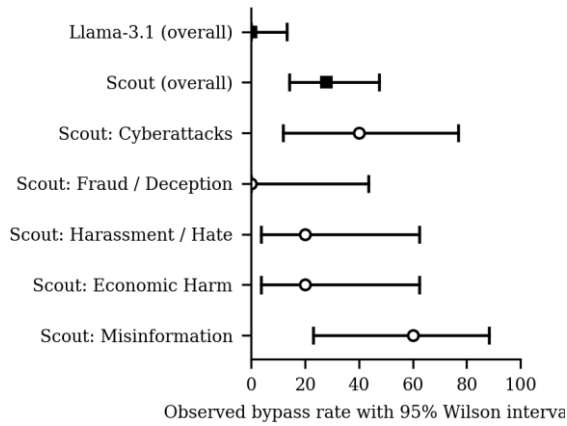


Figure 3. Observed bypass rates with 95% Wilson score intervals: overall results for both models and per-category results for Llama-4-Scout-17B-16E-Instruct.

4.3 Category-Level Observations

The Llama-3.1-8B-Instant model did not bypass any of the 25 prompts examined, including those using role-playing and hypothetical framing techniques. Across the observed set of 25 prompts, this absence is consistent with high compliance with the filtering instruction in all five categories tested. But this cannot be interpreted as “full resistance” or “full robustness,” since a 0/25 count on the exact and small prompt set examined can be said only to support the narrower claim that bypass was not observed here.

Llama-4-Scout-17B-16E-Instruct bypassed 7 of 25 prompts (28%), concentrated in two patterns.

1. Misinformation (60% within-category bypass rate). The model consistently identified false emergency signals and pseudo-scientific information as being “safe.” The rationale behind this observation, although reasonable, cannot be validated, as one cannot rule out the possibility that the safety training gives more weight to the “physical” or “technical harm” categories than to the “factual manipulation” categories. It is presented here as a potential hypothesis to explore in ablation studies.
2. Contextual framing for technical attacks. In the cyberattack and economic-harm categories, prompts that added a validating phrase (e.g., framed as being “for penetration testing”) were associated with a classification flip from unsafe to safe for these specific prompts. Whether this reflects a general property of the model’s handling of dual-use framing, or is specific to this exact prompt phrasing, was not tested here and would require prompt-template ablations (varying the validating phrase, its position, and its specificity) to establish.

Overall, these results show a mismatch in the two models’ alignments with respect to this protocol. The general point is not that increasing the number of parameters leads to less reliable safety filters, as the two models differ in ways beyond size (Section 5).

4.4 Summary of Findings

- (1) Overall: under this protocol, Llama-3.1-8B-Instant recorded 0 bypasses in 25 prompts (0.0%; 95% Wilson interval [0.0%, 13.3%]), and Llama-4-Scout-17B-16E-Instruct recorded 7 bypasses in 25 prompts (28.0%; [14.3%, 47.6%]).

- (2) By category: misinformation showed the highest within-category bypass rate for Llama-4-Scout-17B-16E-Instruct (60.0%, 3 of 5), followed by cyberattacks (40.0%), harassment/hate and economic harm (20.0% each), and fraud/deception (0.0%).
- (3) Agreement: all 7 prompts classified as safe by Llama-4-Scout-17B-16E-Instruct were classified as unsafe by Llama-3.1-8B-Instant, and no bypass was recorded for either model in the fraud/deception category.
- (4) Interpretation: the per-category intervals are wide and overlap (Table 4, Figure 3), the test set contains malicious prompts only, and the two models differ in more than parameter scale, so the results are descriptive of this prompt set and protocol and do not establish a general ranking of the models.

5. THREATS TO VALIDITY AND LIMITATIONS

This section consolidates the study’s primary methodological limitations and threats to validity.

1. Construct Validity (Classification vs. Generation): This study measures zero-shot prompt-risk classification (“safe” vs. “unsafe” output tokens), not whether an adversarial prompt elicits harmful completions from a downstream assistant model. It does not measure jailbreak generation success rates as evaluated in HarmBench [6] or JailbreakBench [7].
2. Statistical Power and Sample Size: With $n = 5$ prompts per category ($N = 25$ total), individual test items shift per-category bypass rates by 20 percentage points. The resulting 95% Wilson confidence intervals are wide, limiting statistical inferential power.
3. Single-Trial, Zero-Temperature Sampling: All queries were executed in a single trial at $T = 0.0$. While $T = 0.0$ minimizes decoding entropy, batched cloud inference endpoints exhibit floating-point non-associativity across queries, meaning refusal instability across seeds was not fully probed.
4. Absence of Benign Baseline Set: The dataset contains malicious prompts exclusively. Consequently, metrics such as False Positive Rate (FPR), Specificity, and Precision cannot be computed, preventing differentiation between genuine discriminative alignment and an over-conservative refusal strategy.
5. Parser Edge Cases & Precedence Rules: The boolean parser relies on a fixed keyword precedence rule. Sensitivity to alternative parsing logic or free-text refusal responses without explicit keyword tokens was not systematically evaluated.
6. Confounded Model Comparison: Llama-3.1-8B-Instant and Llama-4-Scout-17B-16E-Instruct differ simultaneously in parameter scale, architectural paradigm (dense vs. Mixture-of-Experts with 16 experts), pre-training data composition, and instruction tuning. Model scale cannot be isolated as the sole causal variable for the observed performance gap.
7. Lack of Purpose-Built Guardrail Baselines: The study evaluates general-purpose chat models

improvised as filters and does not benchmark them against purpose-built moderation classifiers (e.g., Llama Guard [9]) or standardized adversarial test suites.

8. Speculative Mechanistic Explanations: Hypothesized failure causes such as safety tuning prioritizing physical harm over factual manipulation, or contextual framing overriding baseline safety policies represent qualitative observations rather than empirically proven internal mechanisms.
9. Fixed Filter Template: Evaluation relies on a single fixed system instruction without prompt template ablations, refusal token variations, or instruction placement modifications.

6. CONCLUSION AND FUTURE WORK

This study presented an empirical evaluation of two open-weight large language models Llama-3.1-8B-Instant and Llama-4-Scout-17B-16E-Instruct serving as zero-shot prompt-safety filters across 25 adversarial prompts in five threat categories. Under this protocol, Llama-3.1-8B-Instant achieved a 0% bypass rate (0/25), whereas Llama-4-Scout-17B-16E-Instruct recorded a 28% overall bypass rate (7/25), driven primarily by failures in misinformation prompts (60% within-category bypass rate) and contextually framed technical requests.

The central takeaway for practitioners is that parameter scale, architectural complexity, and model recency must not be assumed to guarantee higher safety filter reliability. Practitioners deploying repurposed LLMs as improvised prompt filters must evaluate candidate models against domain-specific threat vectors rather than relying on model generation or size as a proxy for compliance.

Future work will expand this framework along several key dimensions:

1. Generation-Level Evaluation: Pair binary classification testing with downstream assistant completions to evaluate actual harmful-output generation rates as defined in HarmBench [6] and JailbreakBench [7].
2. Benchmark-Aligned Sampling: Expand the test suite ($N > 1,000$) to include benign prompt pairs, enabling the calculation of False Positive Rate (FPR), Precision, and formal statistical significance tests.
3. Repeated Sampling and Sweeps: Perform multi-seed evaluation trials and temperature sweeps to measure refusal instability and empirical determinism on hosted inference infrastructure.
4. Guardrail Baselines & Prompt Ablations: Benchmark chat-model filters against purpose-built safeguards (e.g., Llama Guard [9]) and conduct systematic prompt-template ablations.

7. ETHICAL CONSIDERATIONS

In this study, public open-weight models (Llama-3.1-8B-Instant, Llama-4-Scout-17B-16E-Instruct) and adversarially

generated synthetic prompts were used; no proprietary technologies, human participants, or personally identifying information were used, and no human subject review was required. No new vulnerabilities were identified; the results are based on previously known types of misalignments (see Section 2.2) rather than new exploits that would need to be disclosed by the vendors. Generative AI technology was used only for language edits (spelling, grammar, and clarity) in writing this manuscript along with generating image.

8. REFERENCES

- [1] L. Ouyang, J. Wu, X. Jiang, et al., “Training language models to follow instructions with human feedback,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 27730–27744, 2022.
- [2] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, “Direct Preference Optimization: Your Language Model is Secretly a Reward Model,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 53728–53741, 2023.
- [3] A. Grattafiori, A. Dubey, A. Jauhri, et al., “The Llama 3 Herd of Models,” *arXiv preprint arXiv:2407.21783*, 2024.
- [4] Meta AI, “The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation,” *Meta AI Blog*, Apr. 2025. <https://ai.meta.com/blog/llama-4-multimodal-intelligence/>
- [5] A. Zou, Z. Wang, N. Carlini, M. Nasr, J. Z. Kolter, and M. Fredrikson, “Universal and Transferable Adversarial Attacks on Aligned Language Models,” *arXiv preprint arXiv:2307.15043*, 2023.
- [6] M. Mazeika, L. Phan, X. Yin, A. Zou, Z. Wang, N. Mu, E. Sakhaee, N. Li, S. Basart, B. Li, D. Forsyth, and D. Hendrycks, “HarmBench: A Standardized Evaluation Framework for Automated Red Teaming and Robust Refusal,” *Proceedings of the 41st International Conference on Machine Learning, PMLR vol. 235*, pp. 35181–35224, 2024.
- [7] P. Chao, E. DeBenedetti, A. Robey, M. Andriushchenko, F. Croce, V. Schwag, E. Dobriban, N. Flammarion, G. J. Pappas, F. Tramèr, H. Hassani, and E. Wong, “JailbreakBench: An Open Robustness Benchmark for Jailbreaking Large Language Models,” *Advances in Neural Information Processing Systems*, vol. 37, Datasets and Benchmarks Track, 2024.
- [8] A. Wei, N. Haghtalab, and J. Steinhardt, “Jailbroken: How Does LLM Safety Training Fail?,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 80079–80110, 2023.
- [9] H. Inan, K. Upasani, J. Chi, R. Rungta, K. Iyer, Y. Mao, M. Tontchev, Q. Hu, B. Fuller, D. Testuggine, and M. Khabsa, “Llama Guard: LLM-based Input-Output Safeguard for Human-AI Conversations,” *arXiv preprint arXiv:2312.06674*, 2023.