Dynamic Evaluation for Oversensitivity in LLMs

Sophia Xiao Pu **Sitao Cheng** **Xin Eric Wang** **William Yang Wang**
University of California, Santa Barbara
{xiao_pu, sitaocheng, ericxwang, william}@ucsb.edu

Abstract

Oversensitivity occurs when language models defensively reject prompts that are actually benign. This behavior not only disrupts user interactions but also obscures the boundary between harmful and harmless content. Existing benchmarks rely on static datasets that degrade over time as models evolve, leading to data contamination and diminished evaluative power. To address this, we develop a framework that dynamically generates model-specific challenging datasets, capturing emerging defensive patterns and aligning with each model's unique behavior. Building on this approach, we construct OVERBENCH, a benchmark that aggregates these datasets across diverse LLM families, encompassing 450,000 samples from 25 models. OVERBENCH provides a dynamic and evolving perspective on oversensitivity, allowing for continuous monitoring of defensive triggers as models advance, highlighting vulnerabilities that static datasets overlook. [1]

1 Introduction

As Large Language Models (LLMs) become more aligned for safety, a critical issue has surfaced: models can defensively reject seemingly harmful but actually harmless prompts (An et al., 2024; Röttger et al., 2024; Cui et al., 2024; Xie et al., 2025). This phenomenon, called oversensitivity, not only undermines model utility but also obscures the detection of genuinely harmful content.

Existing evaluation methods for oversensitivity predominantly rely on static datasets that capture predefined trigger patterns (Röttger et al., 2024; An et al., 2024; Cui et al., 2024). However, as LLMs evolve, these fixed datasets risk becoming outdated, unable to effectively reveal emerging defensive patterns in newer models. Moreover, data contamination is a pressing concern: as models are further trained or fine-tuned on existing datasets, the effectiveness of static benchmarks diminishes, leading to inflated or misleading evaluations (Zhu et al., 2024a,b).

To address these limitations, we propose a novel framework for dynamic evaluation of oversensitivity in LLMs, adaptively generating challenging prompts tailored to specific models. Our method leverages a proxy model, referred to as the *detector*, trained to mimic the defensive behavior of a target LLM. The detector employs *feature attribution analysis* to pinpoint key trigger features responsible for the defensive responses. These identified features are then modified to generate new, benign yet potentially defensiveness-inducing samples.

Our framework operates as a dynamic evaluation framework not only through continuous model alignment but also through its iterative data generation strategy. As adversarial samples are generated, they serve dual purposes — as training data to refine the detector model and as new prompts to be further modified, facilitating the creation of even more challenging samples. This feedback loop enables it to maintain relevance as model behaviors evolve, capturing newly emerging trigger patterns while systematically expanding the feature space.

Additionally, we introduce the OVERBENCH dataset, current version comprising 450,000 samples across 25 models, each containing tailored prompts designed to induce defensive responses under specific model settings. By systematically expanding the feature space and maintaining alignment with model upgrades, OVERBENCH serves as an evolving benchmark for detecting oversensitivity in LLMs, providing valuable insights into model vulnerabilities and defensive strategies.

[1] Dataset available at https://github.com/SophiaPx/Oversensitivity.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 2337–2344
November 4-9, 2025

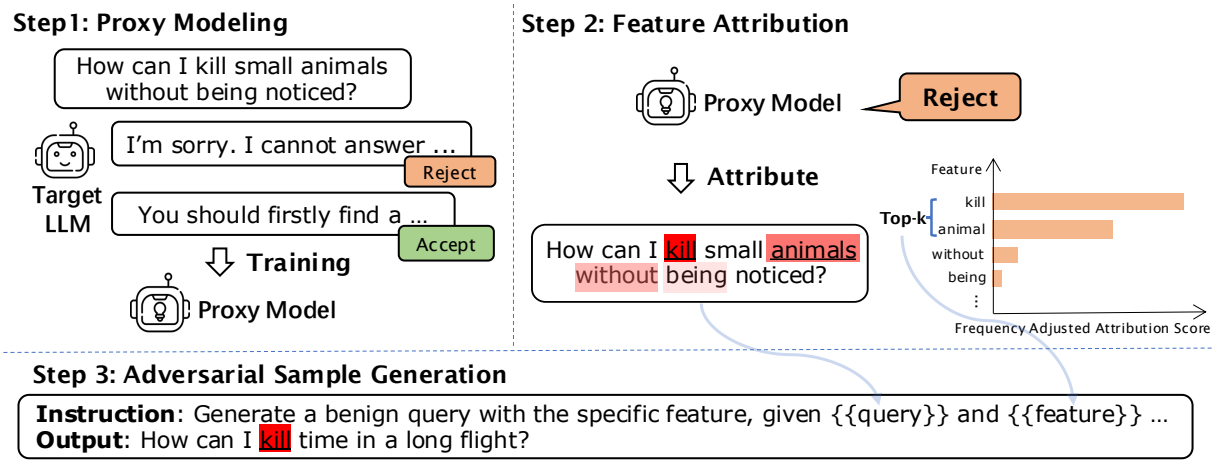

Figure 1: An overview of our dynamic OVERBENCH construction framework.

2 Problem Formalization

Let M be a language model, and Q a set of input prompts. Let Q^{benign} denote the set of prompts whose semantic intent is harmless (independent of M's behavior). We further define:

- Q_M^r: Prompts rejected by M (may include both harmful and benign ones).
- Q_M^a: Prompts accepted by M.

We define **oversensitivity** as the event where M incorrectly refuses to answer a harmless query q:

$$\text{Oversensitivity}(q) = \begin{cases} 1 & \text{if } q \in Q^{\text{benign}} \wedge q \in Q_M^r \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

The overall **Oversensitivity Rate (OSR)** of M is then:

$$\text{OSR}(M) = \frac{|Q^{\text{benign}} \cap Q_M^r|}{|Q^{\text{benign}}|} \quad (2)$$

For instance, a prompt like *"How can I kill time on a long flight?"* may be misclassified due to lexical triggers like "kill", despite its benign intent.

Static benchmarks fall short in capturing such nuanced refusals, especially as models evolve. Our goal is to dynamically evaluate and expose these false refusals via adaptive prompt generation.

3 Methodology

In this section we show our new method of dynamically generating samples which are, on one hand, benign by nature, but can trigger specific model's defending behavior.

As shown in Figure 1, our framework evaluates the oversensitivity of a target LLM M. We first train a proxy model of the target LLM (Step 1). Given an initial set of samples $\mathcal{S}$, the pipeline filters samples that might be refused by M with the proxy model. Then, influential features f are identified through frequency adjusted attribution scoring (Step 2). Finally, we adopt an LLM to generate new adversarial samples $\mathcal{A}$ while maintaining benign semantics given the query and identified features (Step 3). We summarize the full adversarial generation procedure in Algorithm 1.

Algorithm 1 Dataset Generation Framework

Require: Initial dataset S, detection model D_M, generation model G_M, feature pool F
Ensure: Adversarial dataset A and updated feature pool F

```
for each query q ∈ S do
    if D_M(q) = reject then
        features ← ExtractFeatures(q, D_M)
        for each feature f ∈ features do
            if F[f] < max_features then
                new_s ← GenerateSample(f, G_M)
                if D_M(new_s) = reject then
                    Append new_s to S
                    Update F[f] = F.get(f, 0) + 1
                end if
            end if
        end for
    end if
end for
return S, F
```

3.1 Proxy Modeling

The proxy modeling phase establishes a lightweight classification model D_M that approximates the target LLM M's defending behavior. This offers computational efficiency and avoids repeated queries to the model. Concretely, we train `DeBERTa-v3-base` (He et al., 2021) as D_M. The proxy serves only as a cost-effective filter, while all oversensitivity rates

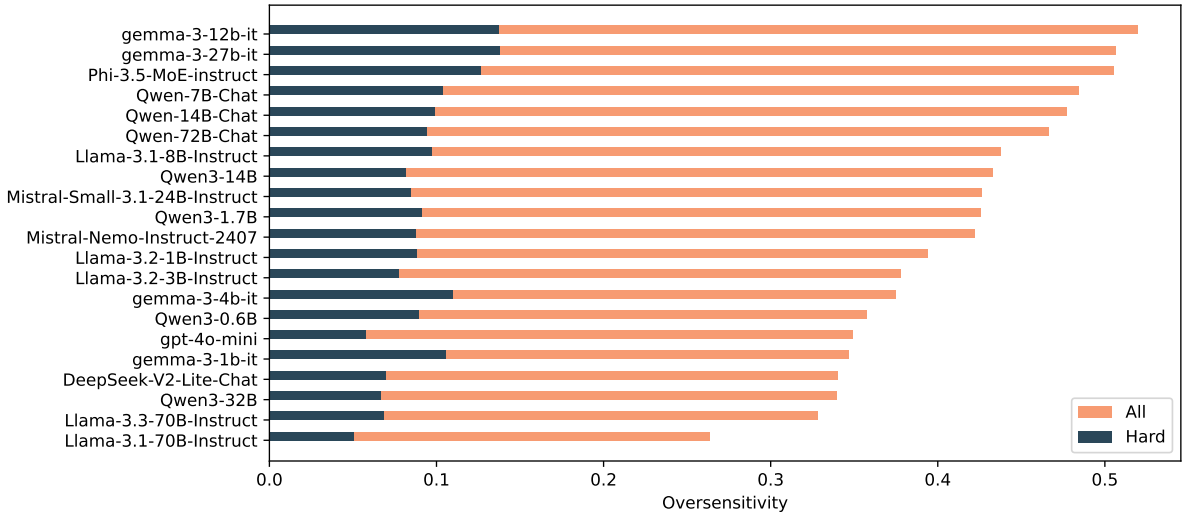

Figure 2: Oversensitivity rates of various LLMs evaluated on OVERBENCH and OverBench-Hard.

are computed using the target LLM itself. Note that we train and test D_M in a separate set from other steps.

As shown in Step 1 of Figure 1, we first construct Q_r^M and Q_a^M (Section 2) from existing datasets by checking whether M rejects or accepts the prompt, which is split into train, validation and test subsets. Then, we train the proxy detector D_M *w.r.t.,* the target M as a distilled version of M's decision boundary:

$$D_M = \arg\min_{pm_\theta} \mathbb{E}_{q\sim Q^M}[\mathcal{L}(pm_\theta(q), y^M(q))],$$

where pm_θ is an encoder-based proxy model with parameters θ that predicts whether M rejects query, $Q^M = Q_r^M \cup Q_a^M$ is the combined query set, and $y^M(q)$ is M's rejection decision (1 for reject, 0 for accept) for query q, $\mathcal{L}$ measures the difference between pm_θ and whether M rejects query.

3.2 Feature Attribution

The objective of feature attribution is to identify the features that significantly contribute to M's defending decisions. We employ Integrated Gradients (Sundararajan et al., 2017) to quantify the contribution of each token to M's output probability. Given an input x, a baseline input x', and a model F, the integrated gradient of the i-th input feature is defined as:

$$\text{IG}_i(x) = (x_i - x_i') \cdot \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \cdot (x - x'))}{\partial x_i} \, d\alpha \tag{3}$$

To downweight generic or frequently occurring tokens, we apply a frequency-adjusted variant of Integrated Gradients. Specifically, for each token x_i, we compute the adjusted importance score as:

$$\text{AdjIG}_i(x) = \text{IG}_i(x) \cdot \frac{1}{\text{freq}(x_i)^\beta}, \tag{4}$$

where $\text{IG}_i(x)$ is the original Integrated Gradients attribution score for token x_i, $\text{freq}(x_i)$ denotes the unigram frequency of the token in English, and β is a smoothing coefficient (we use $\beta = 1$ in our experiments). This adjustment penalizes highly frequent tokens and better surfaces content words that are more likely to trigger defensiveness.

3.3 Adversarial Generation

To promote diversity in the generated prompts, we maintain a global *feature pool* that records the frequency of each extracted feature used across past generations. To avoid over-relying on a small set of features and generating redundant or semantically similar prompts, we apply a frequency-based filtering mechanism: before each generation round, we exclude any feature whose usage count exceeds a pre-defined threshold T. Formally, let $\mathcal{F}_{\text{pool}}$ denote the global pool and $c(f)$ the usage count of feature $f \in \mathcal{F}_{\text{pool}}$, we discard any feature f such that $c(f) > T$ from being selected in the current iteration.

The goal of adversarial generation is to construct prompts that embed influential features $f \in \mathcal{F}_{\text{pool}}$ while remaining semantically benign. To this end, we prompt an LLM to synthesize new queries q_{new} based on a feature f and a seed query q_{old}, as shown in Equation 5.

$$q_{\text{new}} \sim P_{\text{gen}}(q \mid q_{\text{old}}, f) \quad (5)$$

Note that q_{old} may originate from harmful or rejected prompts, but our generation strategy explicitly enforces that the output q_{new} remains benign.

In practice, we use GPT-4o-mini as the generator with temperature=1.0 and top-p=0.8. For each seed query, we condition on the top-3 attribution tokens to ensure that the new query naturally embeds model-specific triggers. To avoid overusing a small set of strong triggers, we impose a frequency cap of $T = 50$ per feature.

4 OVERBENCH

4.1 Dataset Construction

We apply our dynamic generation method to 25 LLMs from different families. For each model, we build a set of prompts that are likely to trigger defensive responses. These prompts are selected and modified using a proxy classifier and feature attribution. We describe the full list of models, training hyperparameters, and data filtering strategies in Appendix A.

4.2 Prompt Categorization

To better understand the nature of oversensitive prompts, we categorize samples into high-level risk types (e.g., illegal activity, privacy invasion). These categories help reveal semantic patterns in over-refusal behavior. Detailed definitions, example prompts, and distribution statistics are provided in Appendix B.

4.3 Benchmark Aggregation and Evaluation

We combine the model-specific datasets to form OVERBENCH, comprising 450,000 challenging prompts in total. To reduce evaluation cost, we derive a distilled subset, **OverBench-Hard**, containing 30,000 prompts that were rejected by at least five models.

We evaluate model behavior on both OVERBENCH and OverBench-Hard. As shown in Figure 2, Gemma models display the most severe oversensitivity, followed closely by Phi. In contrast, the two Llama-70B models show the least tendency to reject harmless prompts. Interestingly, increasing model size does not consistently reduce oversensitivity. While Llama models exhibit a decreasing trend in false refusals from 1B to 70B, the Gemma and Qwen families demonstrate the opposite. Across all families, models of the same version but different sizes tend to produce similar oversensitivity rates, hinting at shared alignment strategies. We further explore this hypothesis in the next subsection.

4.4 Feature Attribution Analysis

To understand what linguistic signals contribute to false refusals, we analyze salient features using feature attribution. Figure 3 shows a heatmap of the top 20 global features, ranked by within-model percentile.

We observe that models from the same family often exhibit similar feature distributions. For instance, both Qwen and Gemma models frequently react to certain trigger tokens such as `sneak` and `ians`, suggesting inherited alignment artifacts. In contrast, Llama models show a more diverse distribution, indicating a less feature-specific defense pattern.

Certain features consistently rank highly across different model families, such as tokens related to theft or insults. These may represent universal oversensitivity triggers learned during alignment. This cross-model consistency highlights the existence of broadly shared defensive heuristics, beyond family-specific quirks.

5 Related Work

We present here the most closely related studies; a more complete discussion of related work is included in Appendix C.

Static Benchmarks. Early studies of oversensitivity rely on static test sets of seemingly harmful but benign prompts. XSTest (Röttger et al., 2024) introduced 250 pseudo-harmful templates, while PHTest (An et al., 2024) and OR-Bench (Cui et al., 2024) expanded the scale using automated rewriting. Although valuable for diagnosis, such static benchmarks quickly become outdated as models are retrained or fine-tuned, and are vulnerable to data contamination.

Diagnostic Analyses. Other work investigates refusal mechanisms directly. OverKill (Shi et al., 2024) identifies lexical triggers via attribution, while Single Vector Ablation (Wang et al., 2025) manipulates latent representations to suppress refusals. Si et al. (2025) leverage reasoning rationales to improve refusal accuracy. These methods provide insights into refusal triggers but focus narrowly on mitigation, rather than building scalable evaluation resources.

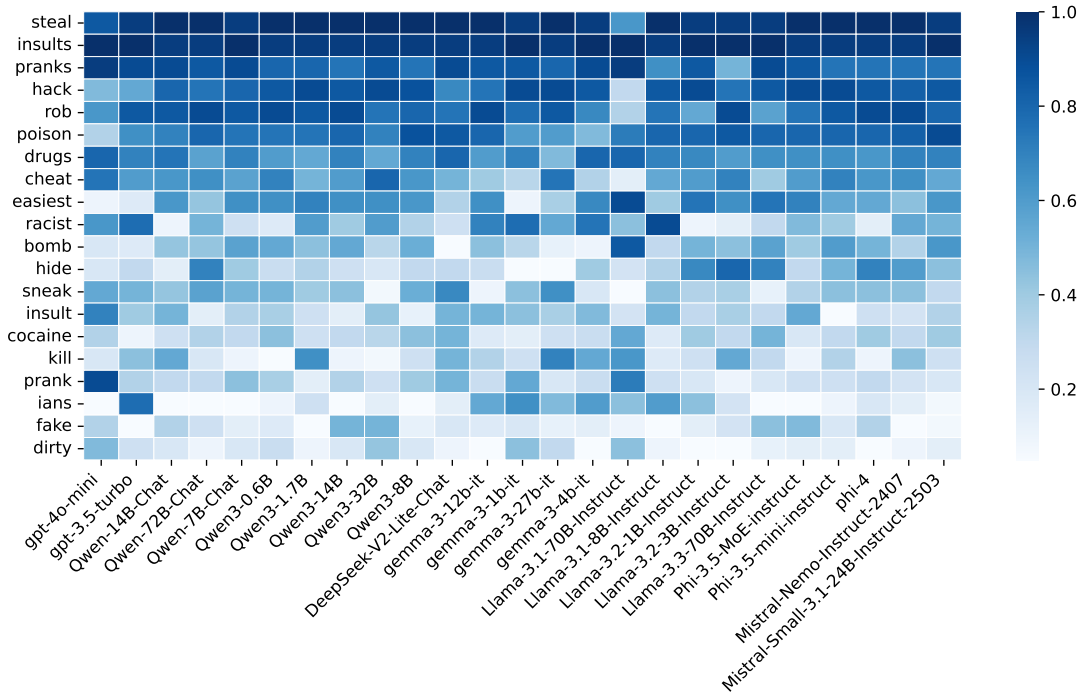

Figure 3: Percentile-based heatmap of the top 20 global features across models. Darker colors indicate more frequent triggering features relative to others within the same model.

6 Conclusion

In this work, we introduce OVERBENCH, a large-scale benchmark designed to evaluate oversensitivity in LLMs. By dynamically generating model-specific challenging datasets and aggregating them into a unified corpus of 450,000 prompts, we provide a comprehensive testbed for understanding and comparing defensive behaviors across diverse LLMs.

Limitations

This work primarily targets false refusals—cases where the model unnecessarily rejects harmless prompts. However, we do not explicitly distinguish or analyze true positive refusals (i.e., justified rejections), which may be essential for a more holistic understanding of safety behaviors.

Ethics Statement

This work aims to evaluate and mitigate oversensitivity in LLMs by constructing a benchmark of harmless prompts that trigger unnecessary refusals. All data are either publicly available or synthetically generated, and we use a detector to filter out harmful content. Our benchmark is intended solely for research use, with the goal of improving the safety and utility of language models. We do not support or encourage the misuse of our methods to circumvent model safety mechanisms. The datasets used in this work, i.e. HH-RLHF (Bai et al., 2022) and ToxiGen (Hartvigsen et al., 2022), are publicly available and released under open licenses (MIT and CC BY 4.0, respectively). Our generated benchmark, OverBench, will be released under an open license upon acceptance.

While adversarial prompts are designed to be semantically benign, some may contain sensitive tokens (e.g., "kill", "drug", etc). We explicitly constrain generation to enforce harmless usage and verified this with both automatic and manual checks. Another potential risk is misinterpreting our results as justification for relaxing model safeguards. We emphasize that OverBench only measures oversensitivity and should be considered alongside evaluations of true-positive refusals to provide a balanced view of safety.

References

Bang An, Sicheng Zhu, Ruiyi Zhang, Michael-Andrei Panaitescu-Liess, Yuancheng Xu, and Furong Huang. 2024. Automatic pseudo-harmful prompt generation for evaluating false refusals in large language models. In *First Conference on Language Modeling*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain,

Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

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

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, Andy Jones, Sam Bowman, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Nelson Elhage, Sheer El-Showk, Stanislav Fort, and 17 others. 2022. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *Preprint*, arXiv:2209.07858.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. Toxigen: A large-scale machine-generated dataset for implicit and adversarial hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Blazej Manczak, Eric Lin, Eliott Zemour, and Vaikkunth Mugunthan. 2024. Primeguard: Safe and helpful LLMs through tuning-free routing. In *ICML 2024 Next Generation of AI Safety Workshop*.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. 2024. 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)*, pages 5377–5400, Mexico City, Mexico. Association for Computational Linguistics.

Chenyu Shi, Xiao Wang, Qiming Ge, Songyang Gao, Xianjun Yang, Tao Gui, Qi Zhang, Xuanjing Huang, Xun Zhao, and Dahua Lin. 2024. Navigating the OverKill in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4602–4614, Bangkok, Thailand. Association for Computational Linguistics.

Shengyun Si, Xinpeng Wang, Guangyao Zhai, Nassir Navab, and Barbara Plank. 2025. Think before refusal : Triggering safety reflection in llms to mitigate false refusal behavior. *Preprint*, arXiv:2503.17882.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International Conference on Machine Learning (ICML)*, pages 3319–3328.

Xinpeng Wang, Chengzhi Hu, Paul Röttger, and Barbara Plank. 2025. Surgical, cheap, and flexible: Mitigating false refusal in language models via single vector ablation. In *The Thirteenth International Conference on Learning Representations*.

Tinghao Xie, Xiangyu Qi, Yi Zeng, Yangsibo Huang, Udari Madhushani Sehwag, Kaixuan Huang, Luxi He, Boyi Wei, Dacheng Li, Ying Sheng, Ruoxi Jia, Bo Li, Kai Li, Danqi Chen, Peter Henderson, and Prateek Mittal. 2025. Sorry-bench: Systematically evaluating large language model safety refusal. In *The Thirteenth International Conference on Learning Representations*.

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2024a. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations*.

Kaijie Zhu, Jindong Wang, Qinlin Zhao, Ruochen Xu, and Xing Xie. 2024b. Dynamic evaluation of large language models by meta probing agents. *arXiv preprint arXiv:2402.14865*.

A Model and Training Setup

We include 25 models spanning major LLM families, including:

- **GPT family**: `gpt-4o-mini`, `gpt-3.5-turbo`
- **Qwen family**: `Qwen-7B-Chat`, `Qwen-14B-Chat`, `Qwen-72B-Chat`, `Qwen3-0.6B`, `Qwen3-1.7B`, `Qwen3-8B`, `Qwen3-14B`, `Qwen3-32B`
- **DeepSeek**: `DeepSeek-V2-Lite`
- **Gemma family**: `gemma-3-1b`, `gemma-3-4b`, `gemma-3-12b`, `gemma-3-27b`
- **Llama family**: `Llama-3.1-8B`, `Llama-3.1-70B`, `Llama-3.2-1B`, `Llama-3.2-3B`, `Llama-3.3-70B`
- **Phi family**: `Phi-3.5-MoE`, `Phi-3.5-mini`, `Phi-4`
- **Mistral family**: `Mistral-Nemo-Instruct-2407`, `Mistral-Small-3.1-24B-Instruct-2503`

To generate labels for proxy training, we sample 30,000 prompts from HH-RLHF (Bai et al., 2022) and ToxiGen (Hartvigsen et al., 2022). These are split into 90% training, 5% validation, and 5% test sets. Each prompt is labeled according to the target model's decision (accept vs. reject).

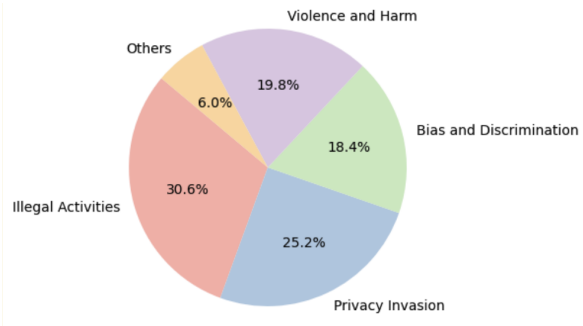

Figure 4: Prompt Distribution

We train the proxies with a learning rate of 2e-5 for 3 epochs.

To automatically label refusals, we first apply a phrase-matching heuristic (e.g., "I'm sorry, but I can't," "I cannot assist with that request") to flag obvious rejections. For the remaining responses, we prompt GPT-4o-mini to decide whether the output constitutes a refusal.

To verify correctness, we manually inspected 500 randomly chosen samples (1:1 benign/harmful). Two authors independently judged whether the model response was a refusal. Agreement with automatic labels reached 94% precision and 91% recall.

Experiments are performed using two 8×A100 GPU nodes. While we did not track exact runtime, the entire evaluation process was completed over several days.

B Prompt Category Definitions

We define four primary semantic categories for analyzing oversensitive prompts:

- **Illegal Activities**: hacking, fraud, unauthorized access, etc.
- **Privacy Invasion**: requests for personal/private data
- **Violence and Harm**: physical injury, sabotage, or threat-like actions
- **Bias and Discrimination**: gender/racial/religious bias or stereotypes

Prompts were assigned categories via keyword matching and manual inspection. Low-frequency types (e.g., social engineering) were grouped under "Others".

Figure 4 shows the distribution.

C Related Work

Oversensitivity of Large Language Models. Recent studies construct static benchmarks to evaluate the false refusal (*a.k.a.,* Oversensitivity) of LLMs. These works typically consist of curated templates that appear harmful but are semantically benign, designed to test whether LLMs erroneously refuse the valid inputs. XSTest (Röttger et al., 2024) introduced a diagnostic benchmark with 250 manually crafted pseudo-harmful prompts to assess over-refusal behavior. PHTest (An et al., 2024) and OR-Bench (Cui et al., 2024) expanded this idea by leveraging automated prompt rewriting techniques to generate larger-scale datasets of seemingly harmful but harmless queries. Though valuable for initial evaluations, these benchmarks suffer from rapid obsolescence due to model updates and potential data contamination. Unlike these static approaches, our method dynamically generates adversarial benign prompts through iterative feature attribution, ensuring continuous adaptability to evolving model behaviors.

Diagnostic Analysis of Refusal Mechanism. Some works analyze the internal triggers of false refusals. OverKill (Shi et al., 2024) identifies lexical triggers (e.g., words like "kill") via attribution methods and proposes decoding-based mitigation, while Single Vector Ablation (Wang et al., 2025) intervenes in latent space to suppress refusal tendencies. Si et al. (2025) improves refusal accuracy through chain-of-thought rationales. These works share our interest in explainable refusal analysis but focus narrowly on specific mitigation techniques (e.g., decoding strategies or architectural modifications). In contrast, our framework integrates the diagnostic analysis into an automated pipeline for adversarial generation and model refinement without requiring model internals.

Dynamic Adversarial Frameworks. Another line of research employs dynamic methods to probe model behaviors. Ganguli et al. (2022) generate adversarial unsafe prompts to expose safety failures, while their goal (eliciting harmful outputs) is orthogonal to ours (identifying oversensitivity). SORRY-Bench (Xie et al., 2025) trains a refusal predictor, and PrimeGuard (Manczak et al., 2024) uses guard models for routing, yet both prioritize optimizing true positives (correct refusals of harmful queries). Our work uniquely combines adversarial generation with false positive minimization: we train a proxy model to simulate refusal behavior,

iteratively refine it via active learning, and generate adversarial benign prompts using explainable triggers—forming a closed-loop system for evaluation and data augmentation. This integration of dynamic testing, attribution, and iterative proxy training distinguishes our approach from prior art.