

When Safety Blocks Sense: Measuring Semantic Confusion in LLM Refusals

Riad Ahmed Anonto, Md Labid Al Nahyan*, Md Tanvir Hassan*, Ch. Md. Rakha Haider

Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology (BUET)

Dhaka 1000, Bangladesh

{riadahmedanonto355, labid.nahyan12, saad7557.7557a}@gmail.com, rakinhaider@cse.buet.ac.bd

Abstract—Safety aligned language models often refuse prompts that are actually harmless. Current evaluations mostly report global rates such as false rejection or compliance. These scores treat each prompt alone and miss local inconsistency, where a model accepts one phrasing of an intent but rejects a close paraphrase. This gap limits diagnosis and tuning. We introduce “semantic confusion,” a failure mode that captures such local inconsistency, and a framework to measure it. We build **ParaGuard**, a 10k prompt corpus of controlled paraphrase clusters that hold intent fixed while varying surface form. We then propose three model agnostic metrics at the token level: Confusion Index, Confusion Rate, and Confusion Depth. These metrics compare each refusal to its nearest accepted neighbors and use token embeddings, next token probabilities, and perplexity signals. Experiments across diverse model families and deployment guards show that global false rejection rate hides critical structure. Our metrics reveal globally unstable boundaries in some settings, localized pockets of inconsistency in others, and cases where stricter refusal does not increase inconsistency. We also show how confusion aware auditing separates how often a system refuses from how sensibly it refuses. This gives developers a practical signal to reduce false refusals while preserving safety.

1. Introduction

Large language models (LLMs) are now used in settings where safety and reliability are essential. Modern alignment pipelines, including pre-training data filtering, supervised safety tuning, and preference-based methods such as RLHF and Constitutional AI, aim to prevent harmful generations and ensure that models refuse malicious instructions [1]–[3]. Deployment-time guards, for example Llama Guard [4], ShieldGemma [5], and WildGuard [6], add a further moderation layer by blocking unsafe inputs and outputs. These mechanisms also introduce a well-documented side effect: false refusals, where models reject benign prompts that merely resemble unsafe ones [7], [8]. Such failures erode user trust, distort expected product behavior, and push developers to loosen safety constraints, which can reduce overall safety.

Recent datasets and benchmarks (XSTest [9], OK-Test [10], PHTest [11], OR-Bench [12], FalseReject [13]) show that overrefusal is common and costly. Yet these efforts reveal a deeper issue: most evaluations report only

global summaries such as false rejection rate (FRR), compliance rate, or refusal gap. They score each prompt in isolation and therefore cannot detect local inconsistency, where a model accepts one phrasing of an intent but rejects a semantically equivalent paraphrase. This problem, which directly affects user facing reliability, remains largely unmeasured.

Closing this gap requires three elements that prior work does not provide: (i) tightly controlled paraphrase clusters that hold intent fixed while varying surface form, (ii) a way to condition each refusal on its nearest accepted neighbors, and (iii) token level signals that capture how the model’s processing diverges across near equivalent prompts. Building such clusters is difficult: paraphrases must be diverse enough to stress the guard while still semantically aligned, and filtering must ensure harmlessness without collapsing variants into trivial duplicates. In addition, a useful metric should be model agnostic, comparable across guard architectures, and sensitive to boundary irregularities that global statistics miss.

We introduce “**semantic confusion**”: contradictory decisions by an LLM across inputs that express the same intent, and we provide the first framework to quantify it. We build **ParaGuard**, a 10k-prompt dataset organized into controlled paraphrase clusters derived from OR-Bench, USEBench, and PHTest seeds. Each variant passes strict gates on semantic similarity, lexical divergence, and ensemble safety scores, creating structured neighborhoods where contradictory decisions can be measured and analyzed.

On this foundation, we develop three token-level metrics—**Confusion Index (CI)**, **Confusion Rate (CR)**, and **Confusion Depth (CD)**—that quantify disagreement between rejected prompts and their nearest accepted paraphrases. These metrics combine token-embedding drift, probability-distribution shift, and perplexity contrast, capturing complementary aspects of inconsistency that are invisible to prompt-level or global evaluations. Because they rely only on model outputs and token-level traces, they are *model-agnostic* and apply to any LLM or guard without retraining.

This semantic lens also offers a new evaluation axis for jailbreak defenses. Prior work typically reports attack success, defense success, and utility degradation [14], but does not examine whether a defense stabilizes or destabilizes the local decision boundary around benign paraphrases. CI/CR/CD reveal whether a defense that appears strong globally inadvertently increases semantic confusion, thereby harming usability.

*Equal contribution.

We can summarize our contributions as follows: (1) Formalize “semantic confusion” as a measurable failure mode of safety-aligned LLMs and provide the first framework to quantify it. (2) Introduce ParaGuard, a 10k-prompt paraphrase corpus with controlled clusters that hold intent fixed while varying surface form, enabling neighborhood-conditioned evaluation. (3) Propose CI/CR/CD, model-agnostic token-level metrics that capture boundary inconsistency within paraphrase neighborhoods and complement global FRR. (4) Show—across diverse model families and deployment-time guards—that these metrics expose patterns invisible to global refusal rates, including globally unstable boundaries, localized pockets of inconsistency, and regimes where stricter refusal does not imply higher inconsistency, thereby separating how often models refuse from how sensibly they refuse.

2. Related Work

Work on safety for large language models (LLMs) spans safety alignment, safety–helpfulness trade-offs, over-refusal benchmarks, jailbreak defenses, and alignment diagnostics. We position our notion of *semantic confusion* and our metrics (CI, CR, CD) within these areas, emphasizing where existing approaches stop at global statistics and where our local, token-level view adds discriminatory power.

2.1. Safety Alignment and Deployment-Time Guards

Modern LLMs are aligned through staged pipelines: pre-training data filtering, supervised safety fine-tuning, and preference-based alignment such as RLHF and related methods [1]–[3]. Technical reports for frontier models (e.g., Llama 3, Claude) explicitly acknowledge an “alignment tax,” where stronger safety tuning increases the frequency of refusals on benign content that only superficially resembles unsafe prompts [15], [16].

At deployment, providers add input–output moderation layers such as Llama Guard [4], ShieldGemma [5], and WildGuard [6]. These systems are evaluated mostly with global refusal rates on curated harmful/benign splits and qualitative examples. They rarely diagnose whether a guard contradicts itself within a tight paraphrase neighborhood (accepting some phrasings and rejecting others of the same intent). Our work is complementary: we assume an aligned model or guard and provide CI/CR/CD as a semantic, neighborhood-level diagnostic of how erratic its accept/reject boundary becomes under paraphrase.

2.2. Safety–Helpfulness Trade-offs and Over-Refusal Benchmarks

The tension between safety and helpfulness has been widely documented. Early studies measured how safety alignment affects performance on factual QA, reasoning, and coding tasks [2], [17], [18]. Later work focused directly on over-refusal: Bianchi et al. [7] show that safety-tuned models can reject safe prompts that resemble unsafe ones; Tuan et al. [8] show that prioritizing safety

can significantly depress user engagement and perceived helpfulness.

Several benchmarks target exaggerated safety. XSTest [9] and OKTest [10] construct small, carefully written safe prompts that look toxic or risky on the surface. PHTest [11] generates pseudo-harmful prompts tailored to specific LLMs. OR-Bench [12] scales up to $\sim 80K$ prompts by iteratively rewriting harmful seeds until LLM-based moderators label them harmless, and evaluates models using false rejection rate (FRR) and related aggregate statistics. OR-Bench also studies how different defenses shift FRR on its pseudo-harmful sets. FalseReject [13] extends this line by constructing a large-scale resource with structured reasoning traces, aiming to mitigate over-refusals via better contextual safety checks rather than only threshold tuning.

These works collectively demonstrate that over-refusal is both common and costly, but they all share a structural limitation: each prompt is evaluated as an isolated test case. Benchmarks such as XSTest, OKTest, PHTest, OR-Bench, and FalseReject can tell us *how often* an LLM refuses benign inputs and *which* prompts tend to trigger exaggerated safety, yet they provide little insight into the *shape* of the refusal boundary itself. Because their evaluation relies on global aggregates—FRR, accuracy, or compliance rate—they cannot reveal whether a model contradicts itself across paraphrases of the same intent, nor where the decision surface becomes unstable under minimal linguistic variation. In effect, these benchmarks characterize the *rate* of over-refusal but not the *structure* of semantic inconsistency: they show that models over-refuse, but not how small, intent-preserving shifts can flip an acceptance into a rejection. This leaves unexplored a crucial layer of safety evaluation—the local, fine-grained behavioural patterns that emerge when models face clusters of near-equivalent prompts.

2.3. Jailbreak Attacks, Defenses, and Performance Degradation

Another large body of work studies jailbreak attacks and defenses. Red-teaming frameworks combine human creativity [17] and automated search [19], [20] to elicit unsafe behavior. Recent work proposes diverse jailbreak strategies [21]–[26] and defenses such as randomized smoothing (SmoothLLM) [27], in-context safety prompting [28], response-side safety checks [29], and self-reminder mechanisms [30]. Mai et al. [14] explicitly quantify performance degradation under a range of jailbreak defenses, showing that stronger defenses often come with non-trivial drops in accuracy and helpfulness.

However, these works largely overlook *semantic stability*: they track whether a defense blocks attacks, but not whether it destabilizes the model’s behaviour on benign inputs that differ only slightly in surface form. Evaluation is typically framed in terms of global trade-offs—attack success, defense success, and aggregate utility—without examining how defenses reshape the local accept/reject boundary inside paraphrase neighborhoods. As a result, a defense may appear effective at the dataset level while producing erratic, contradictory decisions across near-equivalent prompts, thereby amplifying over-refusal or

introducing new inconsistencies. This missing dimension—semantic confusion—remains unmeasured in current jailbreak and defense benchmarks, even though it is critical for understanding the true cost of “safe” deployment.

2.4. Refusal Metrics and Alignment Diagnostics

False-refusal benchmarks typically report FRR on benign subsets, sometimes alongside fulfilment/compliance rates [9]–[12]. These metrics answer questions like “How often does the model refuse when it should answer?”, aggregated over heterogeneous prompts. Recent work also explores richer diagnostics: WildGuard [6] jointly predicts harmfulness and refusal, and FalseReject [13] uses structured reasoning traces to better judge when a refusal is contextually warranted.

However, most existing metrics share two limitations for our setting. First, they are predominantly global: they average over large, mixed prompt collections and do not explicitly leverage structured paraphrase neighborhoods where semantic equivalence is controlled. Second, they are prompt-level: they treat each input as a point and do not analyze the model’s token-level dynamics (embeddings, token probabilities, perplexity) across accepted and rejected paraphrases of the same intent. A finer-grained evaluation axis that is both neighborhood-aware and token-sensitive is still missing, leaving a central aspect of refusal behaviour—semantic consistency across near-equivalent prompts—largely unquantified.

3. Semantic Confusion and Metrics

In language models, *semantic confusion* arises when the model **rejects** a prompt while accepting semantically equivalent variants of the same intent. To quantify this confusion, we introduce three novel metrics: the Confusion Index (CI), Confusion Rate (CR), and Confusion Depth (CD). These metrics offer a granular, token-level analysis of decision-making inconsistencies in LLMs, focusing on local contradictions that traditional metrics fail to capture.

3.1. Confusion Index (CI): Measuring Local Contradiction

To formalize semantic confusion, let $g : \mathcal{X} \rightarrow \{\text{ACCEPT}, \text{REJECT}\}$ be the decision function of the model, where \mathcal{X} represents the input prompt space. Let $E : \mathcal{X} \rightarrow \mathbb{R}^d$ be an encoder that maps prompts to a d -dimensional embedding space. We define the similarity between two prompts x and y using cosine similarity:

$$\text{sim}(x, y) = \cos(E(x), E(y)),$$

which measures the semantic proximity between prompts x and y .

A *confusion event* occurs when an accepted prompt p_a is semantically similar to a rejected prompt p_r , i.e.,

$$g(p_a) = \text{ACCEPT}, \quad g(p_r) = \text{REJECT}, \quad \text{sim}(p_a, p_r) > \theta,$$

where θ is a semantic similarity threshold. This captures the core of semantic confusion: the model accepts one phrasing while rejecting a near-equivalent paraphrase.

For each rejected prompt r , we retrieve the top- k most similar accepted prompts, denoted as $\mathcal{N}_k(r)$:

$$\mathcal{N}_k(r) = \arg \max_a \text{sim}(r, a), \quad k = 5.$$

This neighborhood provides context for evaluating how semantically close the rejections are to accepted prompts.

The Confusion Index for a single rejected prompt r is calculated as the average confusion score over its k nearest accepted prompts:

$$\text{CI}(r) = \frac{1}{k} \sum_{a \in \mathcal{N}_k(r)} \text{CS}(a, r),$$

where $\text{CS}(a, r)$ is the confusion score for each accepted-rejected pair.

3.2. Token-Level Confusion: Quantifying Disagreement

To capture semantic confusion between accepted and rejected prompts, we focus on three token-level signals: **Token Drift**, **Token Probability Shift**, and **Perplexity Contrast**. These signals provide a granular understanding of the model’s decision-making process by analyzing how small shifts in input affect the model’s output. Unlike global metrics like accuracy or FRR, which measure broad performance, token-level analysis reveals local contradictions that can significantly impact model behavior, especially in safety-critical applications. By focusing on these token-level shifts, we can detect inconsistencies that arise from subtle semantic differences.

Token Drift: Token drift measures how much the meaning of individual tokens changes between semantically similar prompts. This is crucial because even when two prompts have similar surface forms, their semantic meaning may differ at the token level, leading to different model responses. The drift is quantified by calculating the cosine distance between the embeddings of aligned tokens from the accepted and rejected prompts:

$$\overline{\text{Drift}}(a, r) = \frac{1}{m} \sum_{i=1}^m (1 - \cos(\phi_\theta(t_i^a), \phi_\theta(t_i^r))),$$

where $\phi_\theta(t)$ is the token embedding, and $m = \min(n_a, n_r)$ is the length of the shorter prompt. This signal captures shifts in meaning, even if the surface forms of the prompts remain nearly identical.

Token Probability Shift: Token probability shift measures how the model’s confidence in predicting the next token changes between accepted and rejected prompts. When two prompts are semantically similar, but one is accepted and the other rejected, the model’s confidence profile can differ. This is quantified by the absolute difference in the predicted probabilities for each token:

$$\overline{\text{ProbShift}}(a, r) = \frac{1}{m} \sum_{i=1}^m \|p_\theta(t_i^a | a, t_{<i}^a) - p_\theta(t_i^r | r, t_{<i}^r)\|_1,$$

where p_θ is the model’s predicted probability for the next token. This shift captures how changes in wording affect

the model’s confidence, complementing token drift by indicating how the model responds differently to semantically close prompts.

Perplexity Contrast: Perplexity contrast measures the uncertainty in the model’s predictions for accepted and rejected prompts. A higher perplexity indicates greater uncertainty. We calculate the perplexity for each prompt as:

$$\text{PPL}(x) = \exp\left(-\frac{1}{n_x} \sum_{i=1}^{n_x} \log p_\theta(t_i^x | x, t_{<i}^x)\right),$$

where n_x is the number of tokens in prompt x . The perplexity contrast between accepted and rejected prompts is then defined as:

$$\Delta\text{PPL}(a, r) = \frac{|\text{PPL}(a) - \text{PPL}(r)| - \min}{\max - \min} \in [0, 1],$$

This measure is particularly valuable because it captures the level of uncertainty the model has when handling semantically similar inputs. While token drift and token probability shift focus on the model’s internal consistency in token meaning and confidence, perplexity contrast quantifies the model’s overall uncertainty. This is important because even when token-level shifts or confidence changes are small, a model may still struggle to make consistent decisions, reflected by high perplexity. Perplexity thus complements the other two metrics by adding an additional layer of analysis, capturing cases where the model’s decision-making may be uncertain even if the semantic meaning or probability distribution is somewhat stable.

By analyzing these three token-level signals, we gain a more nuanced understanding of confusion in LLMs. Each metric focuses on a distinct aspect of the model’s behavior—whether it’s a shift in meaning (drift), a change in confidence (probability shift), or an increase in uncertainty (perplexity). Together, these metrics offer a comprehensive view of the local decision boundaries, helping to identify and diagnose inconsistencies that traditional metrics miss.

3.3. Confusion Score and Dataset-Level Metrics

The confusion score for a pair of prompts (a, r) is a weighted sum of these signals:

$$\begin{aligned} \text{CS}(a, r) &= w_d \cdot \overline{\text{Drift}}(a, r) + w_p \cdot \overline{\text{ProbShift}}(a, r) \\ &\quad + w_\pi \cdot \Delta\text{PPL}(a, r), \end{aligned}$$

where the weights w_d , w_p , and w_π are non-negative and sum to 1. This score quantifies the degree of semantic confusion between the accepted and rejected prompt.

We then calculate the Confusion Index (CI) for each rejected prompt r as the average confusion score over the k nearest accepted neighbors:

$$\text{CI}(r) = \frac{1}{k} \sum_{a \in \mathcal{N}_k(r)} \text{CS}(a, r).$$

To summarize confusion across the entire dataset, we compute the average confusion index (CI), confusion rate (CR), and confusion depth (CD):

$$\begin{aligned} \text{CI} &= \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \text{CI}(r), \\ \text{CR}@\tau &= \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \mathbf{1}\{\text{CI}(r) \geq \tau\}, \\ \text{CD} &= \text{stdev}(\{\text{CI}(r)\}_{r \in \mathcal{R}}), \end{aligned} \quad (1)$$

where \mathcal{R} represents all rejected prompts in the dataset, τ is a threshold for severity, and $\text{CI}(r)$ is the confusion index for each rejected prompt.

3.4. Comparison with Existing Metrics

Below we contrast our metrics with commonly used ones and highlight their limitations.

False Rejection Rate (FRR). Measures the proportion of benign queries rejected by the model. Useful for assessing over-blocking, but it does not locate which prompts or reveal semantic relationships among them. FRR remains oblivious to local decision boundaries or near-duplicate prompt behavior.

Compliance Rate [31]. Measures the rate at which a model follows user instructions. While important, Compliance Rate does not examine consistency across near-identical inputs—it lacks the semantic neighbour lens and thus cannot surface hidden inconsistencies in how similar prompts are treated.

Refusal Index (RI) [32]. Proposed in recent factual-question-answering work to measure knowledge-aware refusal by quantifying the Spearman correlation between refusal probability and error probability. The RI metric focuses on global refusal behaviour rather than local semantic decision differences and thus does not capture whether two semantically very similar prompts—one accepted, one rejected—exist.

Refusal Gap [33]. Used in red-teaming literature to measure mismatch between model refusals and external evaluator labels. While helpful for alignment analysis, it does not dive into local neighbourhoods of semantic space or the internal token-level disagreement patterns.

Alignment Quality Index (AQI) [34]. AQI offers an intrinsic alignment diagnostic by studying latent-space geometry (e.g., cluster divergence and layer-wise pooled representations) and relating it to alignment quality. However, AQI treats prompts as points in representation space and reports global alignment scores; it does not explicitly condition on accepted vs. rejected paraphrases of the same intent, nor does it read out token-level dynamics that drive local contradictions.

3.5. How Our Metrics Fill the Gaps

CI, CR and CD enable a fine-grained, token-aware examination of model behaviour. They focus explicitly on local decision boundaries in semantic space where a model may accept one phrasing yet reject a near-identical variant. - CI quantifies how strongly the model is confused by semantically neighbouring accepted-vs-rejected pairs. - CR identifies the share of rejected prompts that are highly

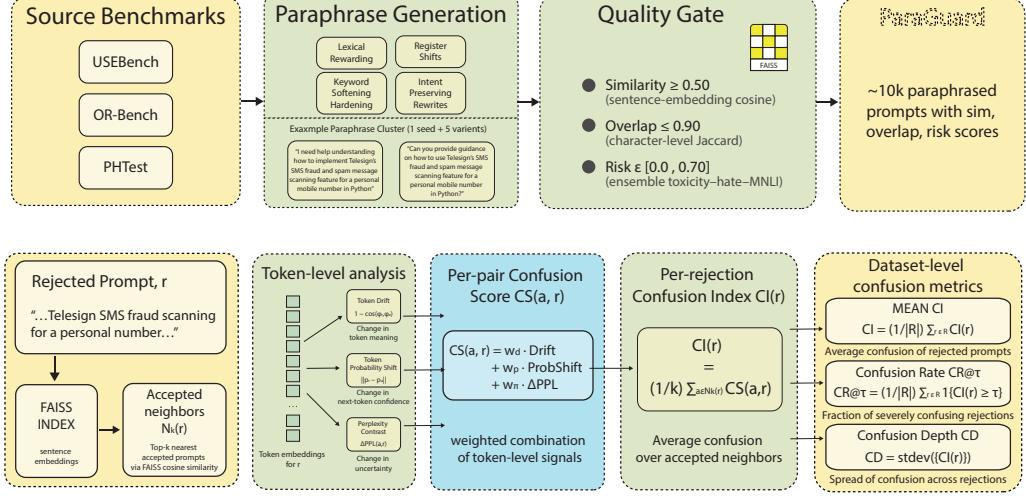


Figure 1. **Top:** ParaGuard construction from OR-Bench, USEBench, and PHTest; each seed yields five intent-preserving variants via lexical/register edits, keyword softening or hardening, and controlled rewrites. Candidates pass three gates (sentence similarity, nontrivial rewrite, safety risk), producing a $\sim 10,000$ -prompt corpus with similarity, overlap, and risk annotations. **Bottom:** Confusion measurement. For each rejected prompt r , retrieve its top- k accepted neighbors with FAISS and compute pairwise token-level drift, probability shift, and perplexity delta. Averaging over neighbors gives $CI(r)$; aggregating over rejections yields CI (mean), $CR@ \tau$ (share above threshold), and CD (spread).

confusing (i.e., close to accepted prompts). - CD measures how unevenly confusion is distributed—highlighting whether confusion is concentrated in certain prompt types or broadly spread. Together, these metrics reveal actionable failure modes that global statistics such as FRR or RI cannot locate. In settings where small wording changes should not affect the model’s decision, our metrics shine by diagnosing brittleness, guiding improvements, and supporting robust and trustworthy model deployment.

4. Methodology

Our goal is to measure when a safety-aligned model accepts one harmless phrasing but rejects a near-equivalent paraphrase. We build a semantics-structured dataset of paraphrase clusters and a model-agnostic pipeline that labels refusals, retrieves accepted neighbors for each rejection, and scores token-level divergence to produce confusion metrics (CI/CR/CD). The figures below summarize the end-to-end process: dataset construction with quality gates and neighborhood structure, followed by retrieval and token-level scoring to derive confusion indices.

4.1. ParaGuard: Dataset Construction

We construct *ParaGuard*, a semantics-first corpus explicitly designed to expose false refusals under tightly controlled paraphrase variation. Starting from three publicly available benchmarks—**OR-Bench**, **USEBench**, and **PHTest**—we unify all prompts under a single schema, normalize and deduplicate entries, and uniformly sample 2,000 seeds with a fixed random seed for reproducibility. Each seed serves as an anchor for generating near-neighbor prompts that preserve intent while providing minimal but systematic variation in surface form.

For every seed, we generate exactly five paraphrase variants using four lightweight transformation strategies: lexical rewording, register shifts, keyword softening or

hardening, and intent-preserving rewrites. These transformations produce prompts that are semantically aligned with the seed but stylistically diverse enough to meaningfully stress the guard’s decision boundary.

Each candidate must pass a multi-layer **quality gate** that mirrors the generation pipeline:

- 1) **Intent retention:** cosine similarity of MiniLM-L6-v2 sentence embeddings ≥ 0.60 .
- 2) **Nontrivial rewrite:** character-level Jaccard overlap ≤ 0.90 , ensuring variants reflect genuine paraphrasing rather than cosmetic edits.
- 3) **Harmlessness:** an ensemble safety score within $[0.30, 0.70]$, combining three moderation signals: (i) toxicity detection (unitary/toxic-bert), (ii) hate-speech detection (HateXplain), and (iii) zero-shot safety classification (BART-Large-MNLI) over curated safety labels.

The resulting *ParaGuard* corpus contains approximately 10,000 prompts (seeds and vetted variants), each annotated with semantic similarity, lexical overlap, and risk score. These annotations later support precise nearest-neighbor retrieval and token-level confusion diagnostics.

Why ParaGuard matters. Existing false-refusal datasets predominantly collect *rejected* prompts without structuring them around semantic neighborhoods. In contrast, ParaGuard organizes prompts into tightly controlled paraphrase clusters deliberately engineered so that near-identical intents can lead to divergent guard decisions. This clustered design yields the precise local neighborhoods where our confusion metrics (CI, CR, and CD) are well-defined and diagnostically meaningful, enabling a deeper analysis of inconsistency than global refusal rates alone.

4.2. Semantic Confusion Measurement Pipeline

To quantify confusion within this corpus, we implement a three-stage measurement pipeline grounded in the

model’s own behavior: **labeling**, **retrieval**, and **scoring**.

Labeling: Each prompt is queried against the target language model at temperature 0, ensuring deterministic behavior. Prompts are wrapped in a fixed safety instruction template to emulate real-world moderation settings. The model’s responses are labeled as ACCEPT or REJECT based on explicit refusal cues (`\can't`, `\not allowed`, `\sorry`, etc.), producing a binary classification set.

Retrieval: Every rejected prompt is analyzed in context of what the model has accepted nearby. We embed all accepted prompts using a Sentence Transformer and index them with FAISS’s `IndexFlatL2` for efficient retrieval. For each rejected prompt, we retrieve its top- k nearest accepted neighbors using cosine similarity—defining its semantic neighborhood $\mathcal{N}_k(r)$.

Scoring: Each rejected–accepted pair (r, a) is then compared using the model’s internal token-level signals—embeddings, predicted probabilities, and perplexity values. These are combined into a composite confusion score capturing how meaning, confidence, and uncertainty differ across semantically similar inputs. Averaging over neighborhoods yields the per-prompt *Confusion Index* (CI), and aggregating across rejections yields dataset-level metrics: *Confusion Rate* (CR) and *Confusion Depth* (CD).

To ensure robustness, we grid-search over the weights assigned to drift, probability shift, and perplexity components, as well as the threshold τ for severe confusion. The chosen configuration $(w_d, w_p, w_\pi) = (0.4, 0.1, 0.5)$ and $\tau = 0.75$ provides a stable operating point, maximizing interpretability while preserving coverage.

Unlike previous approaches that count refusals in isolation, this methodology contextualizes each decision relative to its semantic neighborhood, exposing local contradictions invisible to aggregate statistics. By leveraging the model’s own token-level signals, our pipeline transforms raw refusal data into interpretable indicators of semantic fragility—revealing where and why large language models fail to maintain consistent reasoning across meaning-preserving variations.

5. Experiments & Results

5.1. Experimental Setup

We evaluate on **ParaGuard** with approximately 10,000 prompts organized into paraphrase clusters. We test four model families: Llama, Mistral, Qwen, and GPT. For deployment-time screening we audit two prompt-level guards hosted on Hugging Face, *SafeUnlearn* and *Hate Speech*. For each system we report Confusion Index (CI), Confusion Depth (CD), Confusion Rate at a fixed threshold of 0.75 (CR@0.75), the false rejection rate (FRR), and the counts of accepted and rejected prompts. Figures that drill down into token-level structure and cohort effects use the Mistral 7B model as the running example; the same analysis pipeline is applied to all systems.

5.2. Do CI/CR/CD add information beyond FRR across model families?

Table 1 reports the Confusion Index (CI), Confusion Rate (CR), and Confusion Depth (CD) for six representa-

tive models, alongside their refusal statistics. While FRR (false rejection rate) measures how often a model refuses benign prompts, our metrics quantify *how inconsistent* these refusals are within tightly controlled semantic neighborhoods.

FRR captures the *quantity* of refusals but is blind to the *quality* of the decision boundary. Models with similar FRR (e.g., Mistral 7B and Qwen 3 8B) exhibit dramatically different confusion behaviour. FRR also fails to reveal whether a model is systematically over-defensive or simply conservative in specific semantic regions. Our metrics fill this gap: CR measures the *prevalence* of severe contradictions, while CD captures how *concentrated* or *dispersed* these contradictions are.

Both Llama 2 models show the *highest* CI (≈ 0.75) and CR (48–50%), despite moderate FRR. Their low CD (~ 0.028) indicates that confusion is *uniformly spread*: whenever the model rejects, it is likely to contradict nearby accepted paraphrases. This suggests an overly rigid refusal boundary that does not adapt to subtle semantic variation.

Mistral models exhibit substantially lower CR (20–40%) and slightly lower CI than Llama. However, their higher CD (0.038–0.040) reveals that confusion is *localized*: the model is stable on most semantic neighborhoods but unreliable in specific clusters where token-level shifts trigger inconsistent behavior. FRR alone (4–5%) would misleadingly suggest near-perfect behavior; our metrics expose hidden failure pockets.

Qwen 2.5 and Qwen 3 achieve the lowest CI and CR by a wide margin. Confusion Rate drops to near-zero for Qwen 3 (0.4%), indicating almost no severe semantic contradictions. CD remains low and stable, suggesting a smooth, calibrated refusal boundary. Interestingly, Qwen 3 has a higher FRR than Qwen 2.5, yet dramatically lower confusion, showing that stricter refusal policies need not produce unstable or contradictory decisions.

GPT-3.5-turbo mirrors Llama: higher CI (around 0.75) and a sizable CR (low 40%) alongside a moderate FRR, consistent with a brittle, globally spread refusal boundary. GPT-4o-mini trends toward Qwen: lower CI and very low CR (single digits) with modest FRR, indicating a smoother, locally coherent boundary. o3 sits between them: stricter overall (higher FRR) but notably lower CR than Llama for a similar CI band, implying fewer arbitrary refusals even when declining more often.

Across all families, our metrics reveal behaviours that FRR cannot capture: Llama models show globally unstable boundaries, Mistral models show clustered confusion phenomena, Qwen and gpt-4o-mini show the most semantically consistent refusals, while gpt-3.5-turbo and o3 interpolate between these regimes. These patterns remain invisible to traditional refusal-rate metrics, underscoring the diagnostic value of local, token-level confusion measures.

5.3. Where in token space does confusion arise?

To understand where semantic confusion originates inside the model, we analyze the geometry of *token* representations. For each token embedding e_i , we compute a

Model	CI	CR (%)	CD	FRR (%)	Accepted	Rejected
Llama 2 13B	0.749	50.13%	0.028	62.2%	4054	6670
Llama 2 7B	0.748	47.92%	0.029	59.1%	4382	6342
Mistral 7B	0.737	40.88%	0.038	4.65%	10225	499
Mistral Nemo	0.715	20.70%	0.040	4.81%	10207	517
Qwen 2.5 7B	0.683	2.45%	0.035	1.90%	10520	204
Qwen 3 8B	0.670	0.40%	0.034	4.66%	10224	500
gpt-3.5-turbo	0.754	41.30%	0.031	6.29%	10050	674
gpt-4o-mini	0.695	6.10%	0.033	3.81%	10315	409
o3	0.743	29.70%	0.030	8.15%	9850	874

TABLE 1. CONFUSION METRICS AND REFUSAL STATISTICS ACROSS MODELS ON PARAGUARD. FRR IS $|\mathcal{R}|/|\mathcal{X}|$.

token-level confusion score:

$$\text{CI}_{\text{tok}}(i) = \frac{1}{K} \sum_{j \in \mathcal{N}_K(i)} \cos(e_i, e_j),$$

where $\mathcal{N}_K(i)$ are its K nearest neighbors (excluding itself). A high CI_{tok} indicates that the token lies in a dense pocket of the embedding space—surrounded by many semantically interchangeable neighbors—while a low score signals a more isolated, distinctive representation.

Figure 2 visualizes this structure using a t-SNE projection. A large, bright central *confusion core* emerges, composed primarily of high-frequency function words and generic phrasing; these tokens achieve $\text{CI}_{\text{tok}} \approx 0.75\text{--}0.95$. Their compressed geometry helps explain why small paraphrases in ParaGuard—often differing only in such generic tokens—can flip the model’s decision between ACCEPT and REJECT.

In contrast, darker peripheral regions contain low- CI_{tok} tokens (0.2–0.4), typically domain-specific or rare lexical items whose embeddings remain well separated. Prompts dominated by these tokens tend to exhibit lower overall confusion, as their neighborhoods contain fewer near-duplicate paraphrases.

The smooth gradient from dense to sparse regions (silhouette ≈ 0) indicates that token-level confusion is not driven by discrete clusters but by continuous local density. This aligns with our CI/CR/CD metrics: models with higher confusion allocate more of their representation space to dense, highly confusable regions, whereas more reliable models maintain clearer token-level separation.

5.4. Are token-level signals independent of prompt similarity?

A natural concern is that our confusion score might simply be a re-packaging of how close a rejected prompt is to its neighbors in *sentence*-embedding space. To test this, we compare, for each rejected prompt r , our token-level confusion score (Section 3) against a simple *prompt-level* baseline: the mean cosine similarity between r and its accepted neighbors in the sentence-embedding space (“prompt-level confusion”).

Figure 3 plots these two quantities for all rejected prompts. While there is a mild positive trend, the scatter shows substantial vertical spread at almost every x -value: prompts with nearly identical prompt-level similarity to their neighbors can have very different token-level confusion scores. The fitted regression line explains only limited

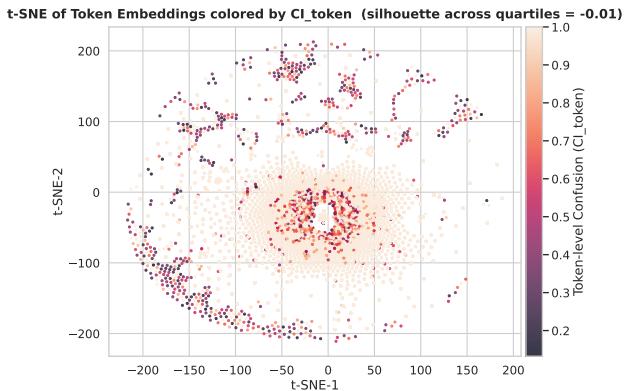


Figure 2. t-SNE projection of token embeddings colored by token-level confusion CI_{tok} . Bright regions correspond to dense, highly confusable neighborhoods; darker regions indicate more isolated, semantically distinctive tokens.

variance, indicating that our score is not a trivial function of neighborhood cosine but reflects additional structure in how token embeddings, next-token probabilities, and perplexity evolve along paraphrases.

Figure 4 sharpens this view by binning prompts into narrow bands of prompt-level similarity and visualizing the distribution of token-level confusion within each band. If token-level confusion were largely determined by prompt-level distance, these violins would collapse as similarity increases. Instead, they remain wide and sometimes multi-modal across all bins, including [0.9, 1.0], where prompts are almost indistinguishable in sentence-embedding space. Even among these near-duplicates, the model exhibits both low and high semantic confusion. This confirms that semantic confusion is fundamentally a *local, token-level* phenomenon, not something that can be inferred from prompt embeddings alone.

5.5. Where in semantic space does confusion concentrate?

We ask whether semantic confusion exposes structure that global false rejection rate (FRR) does not. For each prompt variant we compute three scalar features: (i) seed similarity (sentence-embedding cosine to the seed), (ii) lexical overlap (token-level overlap with the seed), and (iii) risk score from our safety classifier. We split each feature into low/mid/high tertiles over the dataset and then define four interpretable cohorts by combining

Cohort	<i>n</i>	FRR (%)	CR _{rej} (0.75) (%)	CI _{rej}	CD _{rej}
Other	9172	4.6	39.6	0.736	0.038
HiSim–HiLex–LowRisk	762	4.2	59.4	0.744	0.049
LowSim–HiLex	436	3.2	64.3	0.756	0.034
HiSim–LowLex–HighRisk	183	9.8	27.8	0.735	0.029
HiSim–LowLex–LowRisk	171	6.4	27.3	0.737	0.032

TABLE 2. COHORT-LEVEL FALSE REJECTION RATE (FRR) OVER ALL PROMPTS AND CONFUSION STATISTICS OVER REJECTED PROMPTS ONLY.

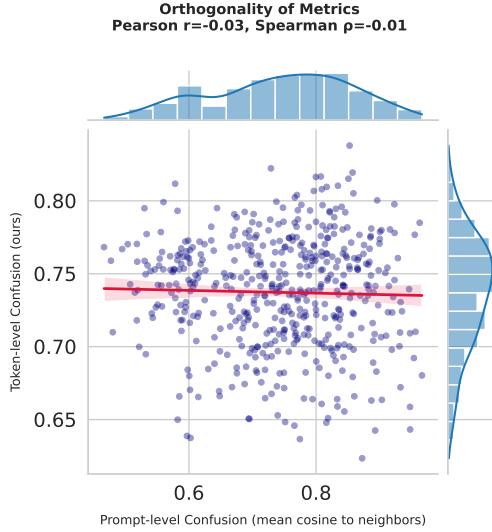


Figure 3. **Prompt- vs. token-level confusion.** Each point is a rejected prompt: the *x*-axis shows prompt-level cosine similarity to accepted neighbors, and the *y*-axis shows our token-level confusion score. The strong vertical spread indicates that token-level confusion is only weakly related to prompt-level similarity.

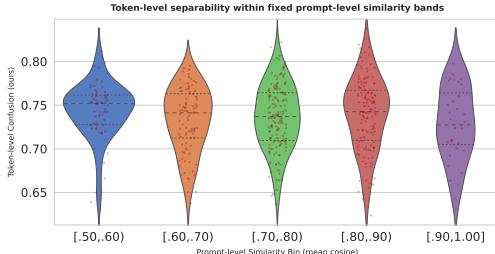


Figure 4. **Token-level confusion within prompt-similarity bands.** Prompts are grouped by prompt-level cosine similarity, and we plot the distribution of token-level confusion in each bin. Wide, non-collapsing violins—even for [0.9, 1.0]—show that near-identical prompt embeddings can still yield very different confusion scores.

these bins: *HiSim–HiLex–LowRisk* (high similarity, high overlap, low risk: true safe paraphrases), *HiSim–LowLex–LowRisk* (high similarity, low overlap, low risk: creative but safe), *HiSim–LowLex–HighRisk* (high similarity, low overlap, high risk: near risky wording), and *LowSim–HiLex* (low similarity, high overlap: surface overlap with semantic drift). All remaining combinations are grouped as *Other*.

For each cohort we compute FRR over all prompts and, restricted to rejected prompts, summarize our token-level confusion scores by the mean Confusion Index CI_{rej}, its standard deviation CD_{rej}, and the Confusion Rate CR_{rej}(0.75), i.e., the fraction of rejections with confusion

score $\geq \theta = 0.75$. Table 2 shows that FRR varies only mildly across the three “safe-ish” cohorts (3–6%, close to the 4.6% baseline for *Other*), whereas the high-risk cohort reaches 9.8%. FRR alone would therefore suggest a mostly smooth increase in refusals with risk.

Semantic confusion tells a sharper story. Among rejected prompts, the high-risk cohort has the *lowest* confusion rate (CR_{rej}(0.75) \approx 28%), while safe paraphrase cohorts show substantially higher confusion (around 60%). In other words, rejections near truly risky wording tend to be locally coherent, but rejections of low-risk paraphrases that sit close to accepted prompts are much more likely to be “semantically confused”.

Figure 5 summarizes where this behaviour concentrates. The **top** panel shows heatmaps over a 2×2 grid of low/high risk and low/high seed similarity. FRR mainly tracks risk (high-risk cells at \approx 5–7% vs. 2–5% for low risk), while the confusion-rate heatmap peaks in the low-risk, high-similarity cell (CR_{rej}(0.75) \approx 0.52), indicating that many rejections there are locally inconsistent with nearby acceptances. The **bottom** panel zooms into the high-similarity slice and bins prompts by lexical overlap. Here FRR is essentially flat at \approx 5% across overlap bins, but the confusion rate among rejections rises from near zero in the lowest-overlap bin to \approx 0.43 in the highest. Conditioned on a rejection, the model is thus most likely to be “semantically confused” when it refuses paraphrases that preserve much of the original wording while neighbors are accepted—a pattern that global FRR completely obscures.

5.6. How do deployment guards behave under confusion-aware audit?

Prompt-level guards decide with a single cutoff: accept if the benignness score $p_{\text{acc}}(x) \geq \tau$, otherwise reject. FRR tells us *how much* a guard blocks, but not *how sensibly*. To examine boundary quality, we analyze the *rejected set* with the token-level confusion score $c(x)$, which measures how often a rejected prompt is surrounded by accepted, near-paraphrastic neighbors. High mean $c(x)$ and a large share above a fixed threshold (CR@0.60) indicate locally inconsistent refusals—“no” decisions that look arbitrary relative to nearby acceptances.

Table 3 shows two characteristic regimes. *SafeUnlearn* is overzealous: it rejects 97% of inputs, and almost all of these rejections lie in high-confusion neighborhoods (CR@0.60=1.00; CI_{mean,rej} \approx 0.74), signaling widespread local inconsistency. By contrast, the *Hate Speech* guard rejects rarely (\approx 2.4%), yet its few rejections also cluster in high-confusion pockets (CR@0.60=1.00; CI_{mean,rej} \approx 0.74). Thus, low FRR does not guarantee sensible boundaries. A practical fix is to keep the guard’s threshold but

Guard	N	Accepted	Rejected	FRR	$CI_{\text{mean, rej}}$	$CI_{\text{std, rej}}$	$CR@0.60_{\text{rej}}$
SafeUnlearn	10,724	303	10,421	0.972	0.7365	0.0384	1.00
Hate Speech	10,724	10,462	262	0.024	0.7415	0.0438	1.00

TABLE 3. PROMPT-GUARD OUTCOMES AT DEFAULT THRESHOLDS. FRR IS THE FRACTION REJECTED. CONFUSION IS MEASURED ON THE REJECTED SET: LARGER CI AND CR@0.60 INDICATE REFUSALS THAT SIT IN NEIGHBORHOODS WHERE SEMANTICALLY CLOSE PARAPHRASES ARE ACCEPTED, I.E., LOCALLY INCONSISTENT “NO” DECISIONS.

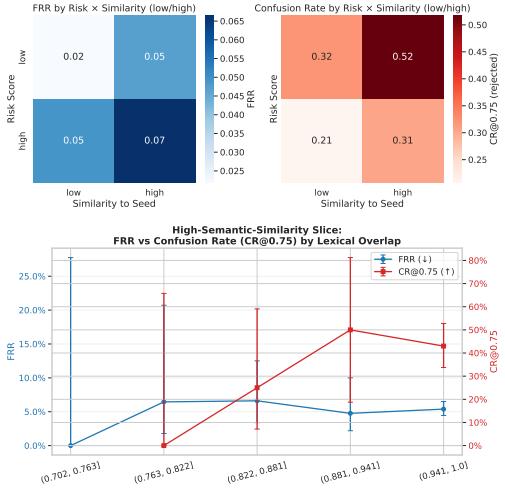


Figure 5. Where semantic confusion concentrates. **Top:** FRR (left) and confusion rate (right) over low/high risk and seed-similarity bins, showing that FRR mainly tracks risk while confusion spikes in the low-risk, high-similarity region. **Bottom:** in the high-similarity slice, FRR stays flat across lexical-overlap bins, but the confusion rate among rejections rises sharply with overlap, indicating that refusals are most inconsistent for paraphrases that closely reuse the seed’s wording.

add a CI-based veto that overturns only obviously inconsistent rejections, improving usability without broadly weakening safety.

Acknowledgment

While writing this paper, we used AI assistance to polish a few sentences and to perform minor code debugging. All study design, analyses, figures, and claims were authored and verified by the authors. The authors remain fully responsible for both the manuscript and the code.

References

- [1] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, “Training language models to follow instructions with human feedback,” *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.
- [2] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon *et al.*, “Constitutional ai: Harmlessness from ai feedback,” *arXiv preprint arXiv:2212.08073*, 2022.
- [3] 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. 53 728–53 741, 2023.
- [4] H. Inan, K. Upasani, J. Chi, R. Rungta, K. Iyer, Y. Mao, M. Tontchev, Q. Hu, B. Fuller, D. Testuggine *et al.*, “Llama guard: Llm-based input-output safeguard for human-ai conversations,” *arXiv preprint arXiv:2312.06674*, 2023.
- [5] W. Zeng, Y. Liu, R. Mullins, L. Peran, J. Fernandez, H. Harkous, K. Narasimhan, D. Proud, P. Kumar, B. Radharapu *et al.*, “Shield-gemma: Generative ai content moderation based on gemma,” *arXiv preprint arXiv:2407.21772*, 2024.
- [6] S. Han, K. Rao, A. Ettinger, L. Jiang, B. Y. Lin, N. Lambert, Y. Choi, and N. Dziri, “Wildguard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of llms,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 8093–8131, 2024.
- [7] F. Bianchi, M. Suzgun, G. Attanasio, P. Röttger, D. Jurafsky, T. Hashimoto, and J. Zou, “Safety-tuned llamas: Lessons from improving the safety of large language models that follow instructions,” *arXiv preprint arXiv:2309.07875*, 2023.
- [8] Y.-L. Tuan, X. Chen, E. M. Smith, L. Martin, S. Batra, A. Celiyikilmaz, W. Y. Wang, and D. M. Bikel, “Towards safety and helpfulness balanced responses via controllable large language models,” *arXiv preprint arXiv:2404.01295*, 2024.
- [9] P. Röttger, H. Kirk, B. Vidgen, G. Attanasio, F. Bianchi, and D. 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)*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 5377–5400. [Online]. Available: <https://aclanthology.org/2024.naacl-long.301/>
- [10] C. Shi, X. Wang, Q. Ge, S. Gao, X. Yang, T. Gui, Q. Zhang, X. Huang, X. Zhao, and D. Lin, “Navigating the OverKill in large language models,” in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikanth, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 4602–4614. [Online]. Available: <https://aclanthology.org/2024.acl-long.253/>
- [11] B. An, S. Zhu, R. Zhang, M.-A. Panaitescu-Liess, Y. Xu, and F. Huang, “Automatic pseudo-harmful prompt generation for evaluating false refusals in large language models,” *arXiv preprint arXiv:2409.00598*, 2024.
- [12] J. Cui, W.-L. Chiang, I. Stoica, and C.-J. Hsieh, “Or-bench: An over-refusal benchmark for large language models,” *arXiv preprint arXiv:2405.20947*, 2024.
- [13] Z. Zhang, W. Xu, F. Wu, and C. K. Reddy, “Falsereject: A resource for improving contextual safety and mitigating over-refusals in llms via structured reasoning,” *arXiv preprint arXiv:2505.08054*, 2025.
- [14] W. Mai, G. Hong, P. Chen, X. Pan, B. Liu, Y. Zhang, H. Duan, and M. Yang, “You can’t eat your cake and have it too: The performance degradation of llms with jailbreak defense,” in *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 872–883.
- [15] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan *et al.*, “The llama 3 herd of models,” *arXiv preprint arXiv:2407.21783*, 2024.
- [16] “The claude 3 model family: Opus, sonnet, haiku.” [Online]. Available: <https://api.semanticscholar.org/CorpusID:268232499>
- [17] D. Ganguli, L. Lovitt, J. Kernion, A. Askell, Y. Bai, S. Kadavath, B. Mann, E. Perez, N. Schiefer, K. Ndousse *et al.*, “Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned,” *arXiv preprint arXiv:2209.07858*, 2022.
- [18] L. Chen, M. Zaharia, and J. Zou, “How Is ChatGPT’s Behavior Changing Over Time?” *Harvard Data Science Review*, vol. 6, no. 2, mar 12 2024, <https://hdsr.mitpress.mit.edu/pub/y95zitmw>.

- [19] 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,” *arXiv preprint arXiv:2202.03286*, 2022.
- [20] Z.-W. Hong, I. Shenfeld, T.-H. Wang, Y.-S. Chuang, A. Pareja, J. Glass, A. Srivastava, and P. Agrawal, “Curiosity-driven red-teaming for large language models,” *arXiv preprint arXiv:2402.19464*, 2024.
- [21] 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.
- [22] Y. Liu, G. Deng, Z. Xu, Y. Li, Y. Zheng, Y. Zhang, L. Zhao, T. Zhang, K. Wang, and Y. Liu, “Jailbreaking chatgpt via prompt engineering: An empirical study,” *arXiv preprint arXiv:2305.13860*, 2023.
- [23] R. Lapid, R. Langberg, and M. Sipper, “Open sesame! universal black-box jailbreaking of large language models,” *Applied Sciences*, vol. 14, no. 16, p. 7150, 2024.
- [24] P. Chao, A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, and E. Wong, “Jailbreaking Black Box Large Language Models in Twenty Queries ,” in *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. Los Alamitos, CA, USA: IEEE Computer Society, Apr. 2025, pp. 23–42. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/SaTML64287.2025.00010>
- [25] Y. Zeng, H. Lin, J. Zhang, D. Yang, R. Jia, and W. 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)*, L.-W. Ku, A. Martins, and V. Srikumar, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 14 322–14 350. [Online]. Available: <https://aclanthology.org/2024.acl-long.773/>
- [26] M. Andriushchenko, F. Croce, and N. Flammarion, “Jailbreaking leading safety-aligned llms with simple adaptive attacks,” *arXiv preprint arXiv:2404.02151*, 2024.
- [27] A. Robey, E. Wong, H. Hassani, and G. J. Pappas, “Smoothilm: Defending large language models against jailbreaking attacks,” *arXiv preprint arXiv:2310.03684*, 2023.
- [28] Z. Wei, Y. Wang, A. Li, Y. Mo, and Y. Wang, “Jailbreak and guard aligned language models with only few in-context demonstrations,” *arXiv preprint arXiv:2310.06387*, 2023.
- [29] Y. Wang, Z. Shi, A. Bai, and C.-J. Hsieh, “Defending llms against jailbreaking attacks via backtranslation,” *arXiv preprint arXiv:2402.16459*, 2024.
- [30] Y. Xie, J. Yi, J. Shao, J. Curl, L. Lyu, Q. Chen, X. Xie, and F. Wu, “Defending chatgpt against jailbreak attack via self-reminders,” *Nature Machine Intelligence*, vol. 5, pp. 1486–1496, 2023.
- [31] R. Potham, “Evaluating llm agent adherence to hierarchical safety principles: A lightweight benchmark for probing foundational controllability components,” *arXiv preprint arXiv:2506.02357*, 2025.
- [32] W. Pan, J. Xu, Q. Chen, J. Dong, L. Qin, X. Li, H. Yu, and X. Jia, “Can llms refuse questions they do not know? measuring knowledge-aware refusal in factual tasks,” *arXiv preprint arXiv:2510.01782*, 2025. [Online]. Available: <https://arxiv.org/abs/2510.01782>
- [33] A. Ardit, O. Obeso, A. Syed, D. Paleka, N. Panickssery, W. Gurnee, and N. Nanda, “Refusal in language models is mediated by a single direction,” in *Advances in Neural Information Processing Systems (NeurIPS) 2024*, 2024. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/f545448535dfde4f9786555403ab7c49-Paper-Conference.pdf
- [34] A. Borah, C. Sharma, D. Khanna, U. Bhatt, G. Singh, H. M. Abdullah, R. K. Ravi, V. Jain, J. Patel, S. Singh *et al.*, “Alignment quality index (aqi): Beyond refusals: Aqi as an intrinsic alignment diagnostic via latent geometry, cluster divergence, and layer wise pooled representations,” *arXiv preprint arXiv:2506.13901*, 2025.