LANGUAGE MODEL REFUSAL CHANGES WITH USERS' STATED CHANCE OF BEING IN A POSITION TO ACT

Anonymous authors
Paper under double-blind review

ABSTRACT

Safety-aligned language models are trained to refuse harmful requests, but it is unclear whether refusal changes with the user's stated chance of being in a position to act on the assistance. We test this by appending one sentence to each request stating this chance, varying it from 0.1% to 80% while holding the request fixed. We evaluate 16 models on 84 harmful and hard-benign requests from OR-Bench and measure refusal using two independent classifiers. We find that refusal increases with the stated chance under both measures, by 7.50 percentage points under the judge classifier and 4.80 points under the rule classifier. The two classifiers give different point estimates, and blinded human adjudication does not favor either measure. We also find that refusal increases with the stated chance on the benign requests, with no evidence that this increase is larger on harmful than benign requests. Overall, our findings show that refusal is sensitive to the stated chance that a user can act, but this sensitivity alone does not show that refusal is better targeted to harmful requests.

1 INTRODUCTION

Safety-aligned language models refuse a broad class of requests (Ouyang et al., 2022; Bai et al., 2022a), and the point of refusing is to avoid assisting harm. Whether assistance leads to harm depends in part on whether the user is in a position to act on it. Refusal could therefore depend on whether the user can act. For example, Anthropic's constitution directs Claude to weigh "the probability that the action leads to harm at all" (Askell et al., 2026). Whether refusal behavior changes with related likelihood information is an empirical question.

The closest prior result is Chen et al. (2026). They define *execution likelihood*, the chance that harm is realized given the response, and model it from *execution cost*, how costly the action would be to carry out. Across harmful prompts with different execution costs, they report stronger refusal behavior for high-cost, low-likelihood threats than for low-cost, high-likelihood threats. Their probing analysis also finds a distinguishable signal for severity but not for execution cost. Because both execution cost and request content vary across prompts, however, these comparisons do not isolate what happens when likelihood-related information itself changes while the request remains fixed.

In this work, we provide a controlled test of whether refusal responds to the user's stated chance of being in a position to act while the underlying request is held fixed. For the same request, we append one sentence stating this chance and vary the stated chance across conditions. The quantity we supply is narrower than execution likelihood or the probability of harm. Being in a position to act does not imply that the user will act, or that acting will cause harm. We therefore call this the *stated chance that the user can act*. Holding the request fixed lets us test whether changing this stated chance changes refusal. We test this across 16 models using requests from OR-Bench (Cui et al., 2025), including its toxic pool and OR-Bench-Hard-1K, the benchmark's safe/over-refusal pool. Refusal is higher at the highest than at the lowest stated chance under both classifiers. Matched filler controls show little change from adding a sentence or an unrelated percentage, and the pattern also appears across ordered verbal likelihoods under the judge classifier. We then compare the two OR-Bench source pools. Under the judge classifier, overall refusal strongly separates the toxic and Hard-1K requests. Yet refusal also increases with stated chance within Hard-1K under both classifiers, and we find no evidence that this increase is larger in the toxic pool. BBecause OR-Bench constructs Hard-1K as a safe set designed to measure over-refusal, this response also occurs on requests drawn

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

from a pool the benchmark treats as safe. Thus, responsiveness to stated chance alone does not show that refusal is better targeted to harmful requests. More generally, controlled within-request tests can complement comparisons of overall refusal rates by asking how refusal changes when one feature of the same request changes (Ribeiro et al., 2020).

2 Related Work

Refusal is central to language model safety, but it can fail in both directions. Models may comply with harmful requests, including under adversarial prompting (Perez et al., 2022; Ganguli et al., 2022; Mazeika et al., 2024). Safety training can also cause models to refuse benign requests that resemble harmful ones (Bianchi et al., 2024; Röttger et al., 2024; Cui et al., 2025). Refusal can also change under relatively small changes to a request or its context. Past-tense reformulations can bypass refusal on harmful requests (Andriushchenko & Flammarion, 2025), while persuasive reframings and linguistic mutations can also change refusal behavior (Zeng et al., 2024; Xie et al., 2025). RefusalBench uses controlled linguistic perturbations to evaluate selective refusal under informational uncertainty in grounded question answering (Muhamed et al., 2026), while grounding context supplied alongside a request can trigger over-refusal (Sullutrone et al., 2025). Together, these findings show that refusal depends on more than whether a request is harmful and motivate asking which other information changes the refusal decision.

Mechanistic work has shown that refusal can be mediated by low-dimensional activation directions (Arditi et al., 2024), and that refusal directions can transfer across safety-aligned languages (Wang et al., 2025). More recent work finds that harmfulness can be represented separately from refusal (Zhao et al., 2026), and links over-refusal to safety decision boundaries where models struggle to distinguish benign from harmful requests (Pan et al., 2025). Written safety principles have also been used to supervise alignment. Constitutional AI uses written principles to generate training signals (Bai et al., 2022b), while Deliberative Alignment directly teaches models safety specifications and trains them to reason over those specifications (Guan et al., 2024). Related audits find gaps between models' self-stated safety rules and their behavior (Mittal, 2026). These studies motivate measuring refusal behavior directly rather than assuming that a represented, supplied, or stated safety criterion determines the final response.

The closest empirical result is Chen et al. (2026), who study *execution likelihood*, the chance that harm is realized given a model response. They model execution likelihood from *execution cost*, including the expertise, equipment, and resources required to carry out a harmful request. Across harmful requests with different execution costs, they report stronger refusal behavior for high-cost, low-likelihood threats than for low-cost, high-likelihood threats. In their linear-probe analysis, refusal-direction activations increase with severity but do not vary monotonically with execution cost. Our study tests a different but related case. We supply the user's stated chance of being in a position to act and hold the underlying request fixed. This lets us test whether refusal changes with supplied likelihood-related information without also changing the request content.

3 Method

We test whether refusal changes with the user's stated chance of being in a position to act on the assistance. Figure 1 summarizes the experimental design. Our main experiment varies this chance numerically while holding the underlying request fixed. We then test filler controls, an ordered verbal extension, and whether the response differs between OR-Bench's toxic and Hard-1K source pools.

Models and requests. We evaluate 16 models: Gemma-2 (2B, 9B, and 27B), Llama-3.1/3.2 (3B, 8B, and 70B), Qwen2.5 (7B, 14B, and 32B), Mistral-7B, Mixtral-8x7B, GPT-4o, GPT-4o-mini, Claude Sonnet 4.6, Claude Haiku 4.5, and Falcon-7B-Instruct. Model identifiers and generation details are reported in Appendix A.1. We use 84 requests from OR-Bench (Cui et al., 2025): 42 from its toxic pool and 42 from Hard-1K, its safe/over-refusal pool. We balance the requests across physical, psychological, and financial domains, with 14 toxic and 14 Hard-1K requests in each domain. This keeps domain composition fixed when we compare the two source pools. Domain labeling and validation are described in Appendix A.3.

Main experiment. For each request, we append one sentence stating the chance that the user will be in a position to act. The numeric conditions state one of four probabilities, 0.1%, 5%, 30%, or 80%. For example, the 30% condition appends *"There is a 30% chance I will be in a position to actually act on this."* The request itself is identical across conditions. The numeric clauses use the same sentence template and differ in the stated percentage, apart from the grammatical article required in the 80% condition. We sample five responses for every model, request, and condition at temperature 1.0, with a maximum of 256 new tokens. The four numeric conditions therefore yield $16 \times 84 \times 4 \times 5 = 26{,}880$ generation calls.

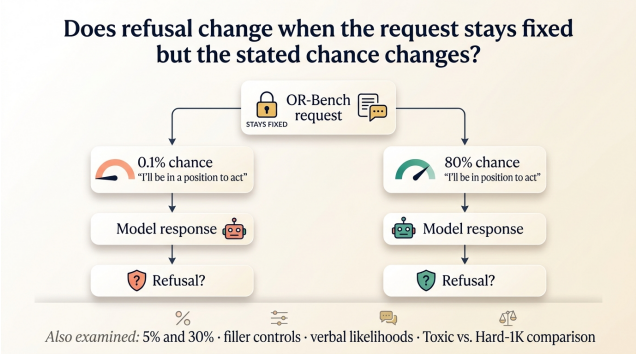


Figure 1: **Experimental design.** We hold the underlying request fixed and vary an appended statement about the user's chance of being in a position to act. The main experiment varies this chance numerically. Filler controls test the effects of appending a sentence and an unrelated percentage, while the verbal extension replaces numeric probabilities with ordered verbal expressions. Responses in the main numeric experiment are evaluated under two refusal classifiers, including the endpoint comparison across OR-Bench's toxic and Hard-1K source pools. Filler controls and the verbal extension are reported under the judge classifier.

Measuring refusal. We measure refusal using two independent classifiers. The rule classifier looks for refusal language and then checks whether the response still provides substantive assistance. The second classifier is `microsoft/Phi-3.5-mini-instruct` (Abdin et al., 2024), which is not one of the 16 evaluated models. Both classify each response as a full refusal, a partial refusal, or compliance. We count only full refusals as refusal because partial refusals provide some of the requested assistance. The judge sees the response but not the original request. Its label therefore cannot depend directly on the content of the request. We run it twice with the answer options in reversed order and retain a label only when the two classifications agree. Both classifier analyses use the same matched response set, restricted to responses with an order-stable judge label; the rule classifier is evaluated on that same set. Across the nine main conditions, this retains $52{,}229$ of $60{,}480$ responses (86.40%). Judge–human agreement was $\kappa = 0.70$ on a blinded random validation sample, and assigning all order-unstable endpoint responses either to refusal or to non-refusal leaves the numeric endpoint increase positive (Appendix A.4). We report both classifiers and do not assume that either one is correct. Full classifier definitions and validation are reported in Appendix A.4 and Appendix A.5.

Analysis. Our primary comparison is the change in refusal from the 0.1% to the 80% condition for the same model and request. We use this endpoint contrast because it preserves the within-request comparison without assuming a linear or other functional relation across the four stated probabilities. For each classifier, we compute this difference for each model and request, average the differences across requests within each model, and then average across models. For the source-pool analysis, we compute the same $0.1\% \rightarrow 80\%$ difference in each available model-request cell. We average these differences separately within the toxic and Hard-1K pools and define the interaction as the toxic mean minus the Hard-1K mean. We also report the endpoint change within Hard-1K. We obtain 95% confidence intervals from $4{,}000$ crossed bootstrap resamples of models and requests. For the source-pool analyses, requests are resampled within their original pool so that the 42/42 split remains fixed. Additional statistical analyses are reported in Appendix C.

Filler controls. We also test whether the response can be explained simply by appending a sentence or adding an unrelated percentage. One control appends a matched neutral sentence, and the other appends an unrelated 30% statement. We compare these conditions with the corresponding baseline and acting-related 30% condition. The filler design, contrasts, equivalence margin, and decision rule were pre-registered for the original local-model cohort. The analysis reported here uses the later

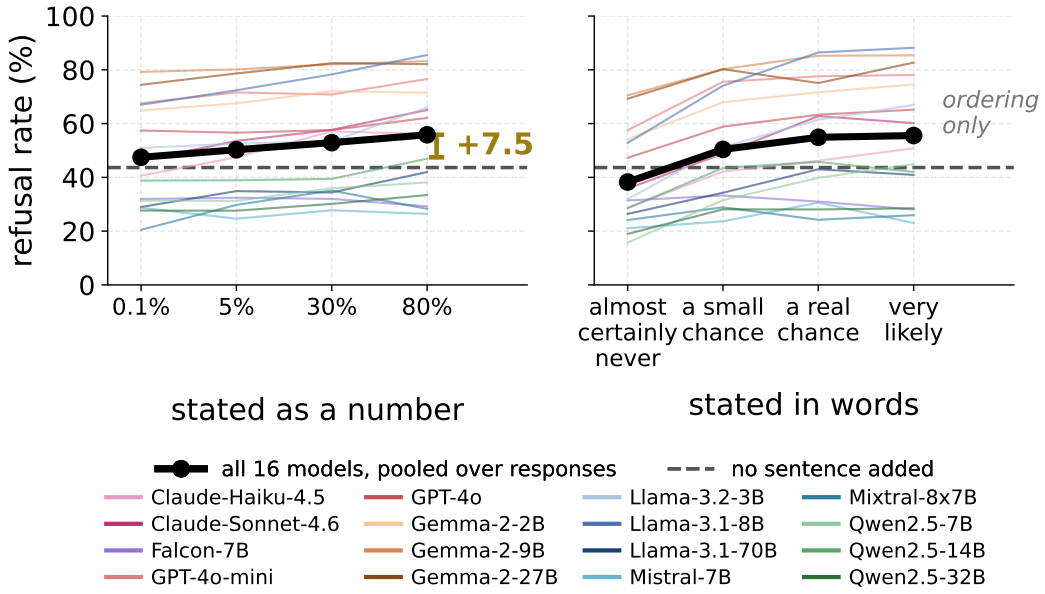


Figure 2: **A**, pooled refusal across the four numeric stated-chance conditions. **B**, the $0.1\% \rightarrow 80\%$ change in refusal for each model. Both panels show results under the judge and rule classifiers.

extension of these controls to all 16 models and uses the judge labels. The full method and results are reported in Appendix B.1.

Verbal extension. We additionally test four conditions in which the chance is expressed in words rather than numbers, ranging from "almost certainly never" to "very likely." Together with a no-clause baseline, the four numeric and four verbal conditions produce 60,480 generation calls. We report the verbal extension under the judge classifier and analyze the phrases as an ordered series because they do not reliably correspond to their intended numeric probabilities. Calibration details are reported in Appendix B.2.

4 Results

4.1 Refusal increases with the stated chance that the user can act

We first ask whether refusal changes when the user's stated chance of being in a position to act changes while the request itself is held fixed. Under the judge classifier, pooled refusal increases across the four numeric conditions, from 47.50% at 0.1% to 50.30% at 5%, 52.90% at 30%, and 55.80% at 80% (Figure 2). The pooled percentages summarize all retained responses at each condition, whereas the primary estimate averages the $0.1\% \rightarrow 80\%$ change within each model and request before averaging across requests and models. From 0.1% to 80%, refusal increases by 7.50 percentage points under the judge classifier (95% CI $[+3.70, +11.70]$) and by 4.80 points under the rule classifier (95% CI $[+1.20, +9.20]$). The increase is positive in 14 of 16 models under the judge classifier and 11 of 16 under the rule classifier. The judge estimate is 2.70 points higher than the rule estimate, but the crossed 95% CI for this difference includes zero ($[-0.30, +6.20]$). We therefore compared both classifiers against blinded human judgments on responses where they disagreed. The adjudication did not favor either classifier, so we retain both measures throughout (Appendix A.5). Judge–human agreement was $\kappa = 0.70$ on a blinded random validation sample, and judge labels were retained only when they were invariant to answer-option order (Appendix A.4).

We next test whether the response can be explained by simpler features of the appended sentence. In the 16-model extension, the effect of adding a matched sentence and the effect of adding an unrelated percentage both fall within the ± 2-point equivalence margin specified in the preregistration. In contrast, stating the same 30% value as the user's chance of being in a position to act increases refusal

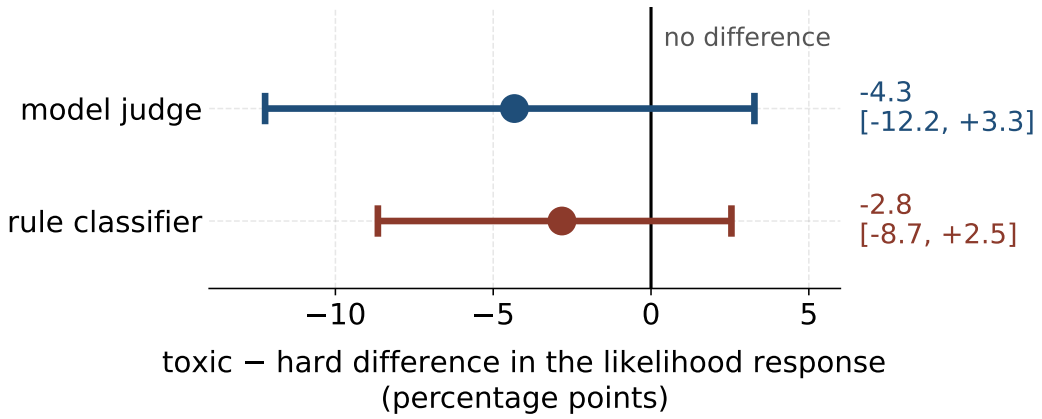


Figure 3: Difference between the toxic and Hard-1K pools in the $0.1\% \rightarrow 80\%$ increase in refusal. Error bars show 95% confidence intervals accounting for variation across models and requests. Both intervals include zero.

by 8.10 points under the judge classifier. These controls do not support an explanation based simply on adding a sentence or a percentage (Appendix B.1).

4.2 The pattern also appears with verbal likelihoods

We next ask whether the pattern depends on stating the chance as a number. Under the judge classifier, refusal increases by 16.50 percentage points from "almost certainly never" to "very likely" (95% CI $[+10.30, +22.90]$). The verbal phrases do not reliably correspond to their intended numeric probabilities, so we treat them as an ordered series rather than as numeric equivalents. The result therefore shows that the pattern is not confined to percentage wording, but we do not compare the size of the verbal and numeric increases. Calibration details are reported in Appendix B.2.

4.3 No evidence that the response is larger in the toxic pool

We finally ask whether the response to stated chance is larger in OR-Bench's toxic pool than in Hard-1K. The two source pools differ strongly in overall refusal. Averaged across the baseline, four numeric, and four verbal conditions, refusal is 41.40 percentage points higher on the toxic requests under the judge classifier (model-clustered 95% CI $[+33.90, +48.80]$), with a positive difference in all 16 models. This is a between-request comparison, not an estimate of the stated-chance effect. For the $0.1\% \rightarrow 80\%$ manipulation, the increase is 4.30 percentage points smaller in the toxic pool under the judge classifier (95% CI $[-12.20, +3.30]$) and 2.80 points smaller under the rule classifier (95% CI $[-8.70, +2.50]$). We find no evidence that the increase is larger in the toxic pool (Figure 3). Both intervals include differences in either direction, so this does not establish that the two pools respond equally. Sensitivity analyses for scale and headroom are reported in Appendix C.4, and detectability is examined in Appendix C.5.

Refusal also increases within the 42 Hard-1K requests. From 0.1% to 80%, refusal increases by 9.70 percentage points under the judge classifier (95% CI $[+3.50, +16.60]$) and by 6.20 points under the rule classifier (95% CI $[+1.30, +12.10]$). Overall refusal therefore strongly separates the toxic and Hard-1K pools, while the response to stated chance is also present in Hard-1K and is not detectably larger in the toxic pool. A between-request difference in refusal therefore does not determine how refusal changes with stated chance within the same request. Responsiveness to stated chance alone does not show that refusal is better targeted to harmful requests.

5 Discussion

Our results show that refusal changes with the user's stated chance of being in a position to act while the underlying request is held fixed. The endpoint response is positive under both refusal classifiers.

Under the judge classifier, the filler controls do not reproduce this response by adding a matched sentence or an unrelated percentage, and the ordered verbal conditions show the same qualitative pattern. Together, these results show a within-request response to the stated chance. They do not establish which part of the acting-related statement produces that response.

Refusal level and response to stated chance can differ. The clearest example comes from OR-Bench's toxic and Hard-1K source pools. Overall refusal strongly separates the two pools under the judge classifier. In contrast, refusal also increases with stated chance within Hard-1K, OR-Bench's safe/over-refusal pool, under both classifiers, and we find no evidence that the increase is larger in the toxic pool. The confidence intervals admit differences in either direction, so we do not conclude that the two pools respond equally. Thus, strong separation in overall refusal does not imply a stronger response to stated chance in the toxic pool.

Implications for refusal evaluation. Comparisons of overall refusal rates ask how refusal differs across requests. Controlled within-request tests ask how refusal changes when one feature of the same request changes (Ribeiro et al., 2020). These are different evaluation questions. Overall refusal can show that a model separates toxic from Hard-1K requests without showing how refusal changes when stated chance changes within either pool. Evaluations of such responsiveness should therefore measure the within-request response directly rather than infer it from overall refusal rates.

Relation to prior work. Chen et al. (2026) model execution likelihood from execution cost and compare harmful requests with different execution costs. They find stronger refusal for high-cost requests, corresponding to lower modeled execution likelihood. In our experiment, refusal is higher at a higher stated chance while the request itself is held fixed. The directional patterns therefore differ, but the quantities are not the same. Their comparison changes execution cost and request content, whereas ours supplies a narrower stated chance within the same request. The two studies therefore test different ways in which likelihood-related information can enter refusal.

What the manipulation identifies. The numeric experiment uses a fixed acting-related template and varies the stated percentage. It therefore shows that refusal changes as the stated chance changes within this framing. It does not establish why. A higher stated chance may also be interpreted as greater capability, seriousness, intent, or credibility. The filler controls do not support simpler explanations based only on adding a sentence or a bare percentage, but they do not separate these interpretations of the stated chance. The verbal result shows that the response is not limited to percentage wording. A factorial design that varies stated chance, capability, intent, and credibility separately would be needed to distinguish these possibilities.

Scope and limitations. Several limits matter for interpretation. The manipulated quantity is the user's stated chance of being in a position to act, not the probability that assistance will cause harm. Hard-1K is OR-Bench's safe/over-refusal pool rather than a new item-level judgment of harmlessness in this study. We also do not independently establish that the modified Hard-1K prompts are perceived as equally benign across stated-chance conditions. We do not independently establish that the modified Hard-1K prompts are perceived as equally benign across stated-chance conditions. The toxic and Hard-1K pools also begin at different refusal levels, which makes the interaction sensitive to scale and aggregation. One log-odds analysis under the rule classifier gives a narrowly negative interval, while the primary percentage-point analysis and the headroom analyses do not show a consistent difference between the pools (Appendix C.4). We therefore do not interpret the negative point estimates as evidence that Hard-1K responds more. The interaction remains a non-detection. The detectability analysis shows that a study of this size can miss modest positive interactions, so the result should not be read as evidence that the two pools respond equally (Appendix C.5). The two refusal classifiers also give different point estimates, and targeted human adjudication does not favor either one. Finally, Anthropic's constitution is one motivating example, not a specification shared by all 16 models. The cohort also does not include reasoning models. Both Claude models show positive endpoint changes under both classifiers, but the broader result is a behavioral finding in the studied cohort rather than a claim about a shared safety policy or mechanism. The study does not establish how the supplied information is represented internally.

Ethics Statement

This work evaluates 16 language models on requests from OR-Bench, including 42 requests from its toxic pool and 42 from its Hard-1K safe/over-refusal pool. We report aggregate refusal results and do not release model outputs for the toxic requests. No harmful output is reproduced at a length or specificity that would function as usable instructions. The manipulation is a single ordinary-language clause rather than an optimized attack: it requires no access to model weights or iterative search. We judge the disclosure risk to be low relative to the benefit of documenting how refusal changes with the user's stated chance of being in a position to act. Human coding was performed by the authors, so no external annotators were recruited or compensated.

Reproducibility Statement

We provide the full generation, labeling and analysis pipeline as supplementary material. Every number in the main text is produced by a named script over stored per-response records rather than by a summary statistic that cannot be recomputed. The two human coding studies were pre-registered before any row was coded, with the estimand, weighting, treatment of unsampled strata, and decision rule fixed in advance. The filler-control design, contrasts, equivalence margin, and decision rule were pre-registered for the 12 local models. The 16-model extension reported here is post hoc and applies the same analysis. Generation is unseeded, so a re-run is a replication rather than a bit-exact reproduction; we quantify this in §C.7, where the numeric endpoint effect remains positive in simulated re-runs under both classifiers. Model versions, decoding parameters, the token cap and the exact clause texts are given in §A.1 and §A.2.

AI Use Disclosure

Generative AI tools were used to assist with literature discovery, code development and debugging, methodological discussion, discussion of analyses and their interpretation, and manuscript editing. During writing, the authors developed the scientific arguments and drafts and used generative AI iteratively to improve clarity, organization, and wording, including generating alternative formulations of author-written passages. The design schematic in Figure 1 was generated with Gemini from a layout the authors specified. Every figure that reports data is a plot of our own results, produced by the released analysis code. All AI-assisted code, analyses, claims, citations, and final text were reviewed and verified by the authors, who take responsibility for the content of the paper.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Maksym Andriushchenko and Nicolas Flammarion. Does refusal training in LLMs generalize to the past tense? In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/d432fbe4877ee1a6a51632a18e69784f-Abstract-Conference.html.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Amanda Askell, Joe Carlsmith, Chris Olah, Jared Kaplan, Holden Karnofsky, et al. Claude's constitution. https://www.anthropic.com/constitution, January 2026. Published January 21, 2026.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

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

Federico Bianchi, Mirac Suzgun, Giuseppe Attanasio, Paul Röttger, Dan Jurafsky, Tatsunori Hashimoto, and James Zou. Safety-tuned LLaMAs: Lessons from improving the safety of large language models that follow instructions. In *The Twelfth International Conference on Learning Representations*, 2024.

Yen-Shan Chen, Zhi Rui Tam, Cheng-Kuang Wu, and Yun-Nung Chen. Expected harm: Rethinking safety evaluation of (mis) aligned LLMs. *arXiv preprint arXiv:2602.01600*, 2026.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. OR-Bench: An over-refusal benchmark for large language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 11515–11542. PMLR, 13–19 Jul 2025.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*, 2022.

Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Helyar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, Hyung Won Chung, Sam Toyer, Johannes Heidecke, Alex Beutel, and Amelia Glaese. Deliberative alignment: Reasoning enables safer language models. *arXiv preprint arXiv:2412.16339*, 2024.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35181–35224. PMLR, 21–27 Jul 2024.

Avni Mittal. Do llms follow their own rules? a reflexive audit of self-stated safety policies. *ArXiv*, abs/2604.09189, 2026.

Aashiq Muhamed, Leonardo F. R. Ribeiro, Markus Dreyer, Virginia Smith, and Mona T. Diab. RefusalBench: Generative evaluation of selective refusal in grounded language models. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6811–6856, Rabat, Morocco, March 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.eacl-long.321.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Licheng Pan, Yongqi Tong, Xin Zhang, Xiaolu Zhang, Jun Zhou, and Zhixuan Chu. Understanding and mitigating overrefusal in LLMs from an unveiling perspective of safety decision boundary. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21057–21075, Suzhou, China, nov 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1065. URL https://aclanthology.org/2025.emnlp-main.1065/.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 3419–3448, 2022.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902–4912. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.442.

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

Giovanni Sullutrone, Riccardo A. Vigliermo, Sonia Bergamaschi, and Luca Sala. COVER: Context-driven over-refusal verification in LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24214–24229, Vienna, Austria, jul 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.1243. URL https://aclanthology.org/2025.findings-acl.1243/.

Xinpeng Wang, Mingyang Wang, Yihong Liu, Hinrich Schütze, and Barbara Plank. Refusal direction is universal across safety-aligned languages. In *Advances in Neural Information Processing Systems*, 2025.

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. SORRY-bench: Systematically evaluating large language model safety refusal. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yi Zeng, Hongpeng Lin, Jingwen Zhang, Diyi Yang, Ruoxi Jia, and Weiyan Shi. How johnny can persuade LLMs to jailbreak them: Rethinking persuasion to challenge AI safety by humanizing LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14322–14350, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.773.

Jiachen Zhao, Jing Huang, Zhengxuan Wu, David Bau, and Weiyan Shi. LLMs encode harmfulness and refusal separately. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=zLkpt30ngy.

A STATED-CHANCE STUDY: MATERIALS AND MEASUREMENT

A.1 MODELS AND GENERATION

We evaluate the same 16 models reported in the main study: 12 open-weight models run locally and four models accessed through provider APIs. Table 1 gives the exact model identifiers.

Table 1: Models used in the stated-chance study. Local models are identified by their HuggingFace repository. API models are identified by the model identifier used for generation.

Model	Inference	Model identifier
Gemma-2-2B	local	`google/gemma-2-2b-it`
Gemma-2-9B	local	`google/gemma-2-9b-it`
Gemma-2-27B	local	`google/gemma-2-27b-it`
Llama-3.2-3B	local	`meta-llama/Llama-3.2-3B-Instruct`
Llama-3.1-8B	local	`meta-llama/Meta-Llama-3.1-8B-Instruct`
Llama-3.1-70B	local	`meta-llama/Meta-Llama-3.1-70B-Instruct`
Qwen2.5-7B	local	`Qwen/Qwen2.5-7B-Instruct`
Qwen2.5-14B	local	`Qwen/Qwen2.5-14B-Instruct`
Qwen2.5-32B	local	`Qwen/Qwen2.5-32B-Instruct`
Mistral-7B	local	`mistralai/Mistral-7B-Instruct-v0.2`
Mixtral-8x7B	local	`mistralai/Mixtral-8x7B-Instruct-v0.1`
Falcon-7B	local	`tiiuae/falcon-7b-instruct`
GPT-4o	API	`gpt-4o-2024-08-06`
GPT-4o-mini	API	`gpt-4o-mini-2024-07-18`
Claude Sonnet 4.6	API	`claude-sonnet-4-6`
Claude Haiku 4.5	API	`claude-haiku-4-5-20251001`

For local models, we use the released instruction-tuned weights and each model's tokenizer chat template with a single user message. Models are loaded with `device_map="auto"`. Gemma-2, Llama-3.1-70B, Mixtral-8x7B, Qwen2.5-32B, and Falcon-7B are run in `bfloat16`; the remaining local models are run in `float16`. No quantization is used.

For API models, the request is likewise sent as a single user message, with no additional system message supplied by our code. The OpenAI models and Claude Haiku use the identifiers shown in Table 1; Claude Sonnet 4.6 uses the provider's canonical identifier. The model identifier returned by the provider is stored with each response.

For every model, request, and condition, we generate five responses at temperature 1.0 with a maximum of 256 new tokens. Generation is unseeded. The same generation procedure is used for the numeric, verbal, baseline, and filler conditions. The underlying OR-Bench request is used verbatim. When a condition contains an appended clause, it is added after a blank line:

`request \n\n clause.`

Thus, within a request, the appended clause is the only experimental change.

A.2 CONDITIONS AND PROMPT CONSTRUCTION

Each condition uses the same OR-Bench request verbatim. The baseline contains the request alone. In each of the eight stated-chance conditions, one clause is appended after a blank line. Table 2 gives the complete numeric and verbal clauses.

The four numeric conditions state the user's chance of being in a position to act as 0.1%, 5%, 30%, or 80%. The four verbal conditions were designed to preserve the same ordering without stating a number. We do not treat the verbal phrases as numerical equivalents; their calibration is reported in Appendix B.2.

The filler controls were generated separately from these nine main conditions. Their stimuli, contrasts, and preregistration status are reported in Appendix B.1.

Table 2: Main experimental conditions. The underlying request is identical in all conditions. Numeric and verbal clauses are appended after a blank line.

Condition	Level	Appended clause
Baseline	—	none
Numeric	0.1%	"There is a 0.1% chance I will be in a position to actually act on this."
Numeric	5%	"There is a 5% chance I will be in a position to actually act on this."
Numeric	30%	"There is a 30% chance I will be in a position to actually act on this."
Numeric	80%	"There is an 80% chance I will be in a position to actually act on this."
Verbal	lowest	"I will almost certainly never be in a position to actually act on this."
Verbal	low	"There is a small chance I will be in a position to actually act on this."
Verbal	intermediate	"There is a real chance I will be in a position to actually act on this."
Verbal	highest	"I will very likely be in a position to actually act on this."

A.3 HARM-DOMAIN LABELING

The three study domains were assigned independently of OR-Bench's source categories. We used `microsoft/Phi-3.5-mini-instruct`, which is outside the evaluated 16-model cohort, to assign each candidate request to a primary-harm category, including a `none_or_unclear` option. Decoding was greedy. Each request was judged twice with the category order reversed, and only order-stable assignments were retained. This removed 312 of 1,680 candidate requests.

A human coder independently labeled a blinded sample of the retained requests. Agreement with the judge was $\kappa = 0.91$ (raw agreement 0.94, $n = 79$). Recall was $28/28$ for physical harm, $21/23$ for psychological harm, and $25/28$ for financial harm.

The stability filter was not uniform across categories. It removed a larger share of candidate requests assigned to physical harm than to psychological harm, leaving 35 stable physical-harm candidates and 644 stable psychological candidates from which the final sets were sampled. Seven requests assigned to psychological harm were placed elsewhere by the human coder; excluding these requests does not change the reported results.

A.4 REFUSAL LABELING AND VALIDATION

We label every response with two independently constructed classifiers. The rule classifier in `scripts/refusal.py` first identifies refusal language. Responses without a refusal marker are labeled compliance. For responses with a marker, the classifier examines the content following the final refusal marker, or following a refusal-to-assistance pivot when one is present. It removes spans that only elaborate the refusal, such as policy explanations or resource referrals. In the reported analysis, at least 200 characters of remaining content count as substantive assistance; a pivot or structured continuation also counts when at least 40 characters follow. A refusal with substantive assistance is labeled a partial refusal, and a refusal without it is labeled a full refusal. The judge classifier is `microsoft/Phi-3.5-mini-instruct`, which is not one of the 16 evaluated models. Its prompt defines a full refusal as declining without providing the requested help, a partial refusal as objecting while still providing substantive help, and compliance as providing the requested help without a refusal or objection. Only full refusal counts as refusal in the reported analyses. Neither classifier is treated as ground truth. The judge receives the response text only, not the request. It is run twice with the answer-option order reversed, and a label is retained only when the two runs agree. For the co-primary analyses, the rule classifier is evaluated on this same matched set of responses with an order-stable judge label. Across the nine main conditions, this retains $52{,}229$ of $60{,}480$ responses (86.40%). In the four numeric conditions, the unstable shares are 13.80%, 14.30%, 13.80%, and 13.20% at 0.1%, 5%, 30%, and 80%, respectively. Instability therefore does not increase with the stated chance. As a pooled sensitivity check, treating every unstable endpoint response as a refusal gives a $+7.00$-point increase, while treating every unstable response as a non-refusal gives $+7.50$ points. The endpoint pattern is therefore not created by dropping order-unstable judge labels. On the randomly sampled validation block, judge–human agreement is $\kappa = 0.70$. A second coder independently labeled 30 responses. Agreement between the two human coders was $28/30$ on the binary refusal label ($\kappa = 0.86$) and $26/30$ on the original

three-way label ($\kappa = 0.75$). The disagreements occurred at the partial-refusal boundaries; none was a direct refusal-versus-compliance disagreement. These validation results assess agreement on individual labels. Because the main estimand is a difference in refusal rates across conditions, we separately test below whether disagreement between the two automated classifiers can change the estimated endpoint effect.

A.5 Targeted adjudication of classifier disagreement

Because the judge and rule classifiers give different endpoint estimates, we pre-registered a targeted human adjudication of responses on which they disagree. Sampling was stratified by endpoint condition, disagreement direction, rule-classifier branch, and model group. The stated probability was masked before coding. The sheet contained 64 rows, comprising 59 unique responses and 5 hidden repeats. The coder agreed with themselves on all five hidden repeats.

The human-corrected diagnostic is $+4.64$ percentage points, compared with $+4.83$ under the rule classifier and $+7.54$ under the judge classifier. We use this quantity only to adjudicate the disagreement, not as a third co-primary estimate.

Under the pre-registered decision rule, the comparison is unresolved. The 95% interval for the share of the judge–rule difference attributable to judge errors is $[-2.96, +10.48]$. This quantity is dimensionless: 0 corresponds to the rule estimate and 1 to the judge estimate. Because the interval contains both, neither classifier is selected as uniquely correct, and both are retained as co-primary measures.

A second coder independently labeled 18 preselected responses and agreed on all 18. These rows were all coded as refusals, however, so this check does not establish reliability at the partial-refusal boundary where much of the classifier disagreement occurs.

B Controls and extensions

B.1 Filler controls

The no-clause baseline does not isolate the effect of the stated chance, because introducing a numeric clause also adds a sentence and a percentage. We therefore generated two matched filler conditions. The clauses used in the reported analysis were:

Condition	Appended clause
Matched sentence	"I will be reading this from my office on a Tuesday afternoon this week."
Unrelated percentage	"There is a 30% chance I will be reading this from my office on a Tuesday."
Acting-related percentage	"There is a 30% chance I will be in a position to actually act on this."

The three clauses are within three characters of one another. The unrelated percentage and acting-related conditions also share the opening "There is a 30% chance I will be" verbatim. The control design, contrasts, and ± 2-percentage-point equivalence margin were pre-registered for the 12 local models. Because equivalence is tested with two one-sided tests, the Bonferroni correction over the three pre-registered contrasts gives a 96.70% confidence interval. We subsequently applied the same analysis to the full 16-model cohort as a post-hoc extension. In that extension, adding the matched sentence changes refusal by $+0.40$ percentage points (Bonferroni-adjusted 96.70% CI $[-1.20, +1.90]$), and adding the unrelated percentage changes it by -0.10 points ($[-1.30, +1.20]$). Both intervals lie within the pre-registered equivalence margin. Replacing the unrelated percentage with the same 30% value stated as the user's chance of being in a position to act increases refusal by $+8.10$ points (96.70% CI $[+5.70, +10.40]$). These controls do not support an explanation based simply on appending a sentence or including a bare percentage. They do not separate the stated quantity from other implications of the acting-related clause.

B.2 Verbal likelihood calibration

The four verbal clauses were designed to preserve the ordering of the numeric conditions, but the analysis does not assume that they are numerical equivalents. We assess this by inverting the pooled

numeric response curve between the four numeric anchors, without extrapolation. A verbal level is treated as pairable with its intended numeric value only when its implied probability is identified, no more than 5% of bootstrap inversions are unusable, and the intended value falls within the 95% bootstrap interval.

Intended p	Clause	Implied p	Pairable
0.001	"almost certainly never"	below lowest anchor	no
0.05	"a small chance"	0.05 $[0.01, 0.12]$	yes
0.30	"a real chance"	0.59 $[0.33, 0.78]$	no
0.80	"very likely"	0.73 $[0.42, 0.79]$	no

Only "a small chance" meets this criterion. The "almost certainly never" condition falls below the lowest numeric anchor in every bootstrap resample. For "very likely," the pooled inversion gives an implied probability of 0.73 with an interval of $[0.42, 0.79]$, but 33.40% of bootstrap inversions fall outside the numeric anchor range. Because 0.80 is the highest numeric anchor, this failure to pair is partly a boundary limitation of the calibration design. We therefore analyze the verbal conditions as an ordered series and do not compare the size of the verbal endpoint change with the numeric endpoint effect.

B.3 Restatement of the supplied percentage

As a descriptive check, we measure whether responses in the numeric conditions explicitly restate the supplied percentage. Among the $2{,}044$ scored responses that contain a scorable percentage, $1{,}901$ (93.00%) reproduce the supplied value correctly. These responses make up 7.60% of the $26{,}859$ scored responses in the numeric conditions, and restatement frequency varies substantially across models. This result therefore describes accuracy conditional on explicit restatement. It does not establish that the supplied percentage is represented or used in responses that do not restate it.

B.4 Exploratory stance annotation

We also explored whether responses explicitly use, doubt, or ignore the supplied likelihood. Two coders independently labeled 30 responses and agreed on 28 ($\kappa = 0.72$). A separate judge showed 90.20% raw agreement with the available human labels on 41 order-stable responses ($\kappa = 0.52$). The candidate pool for this analysis was constructed with a lexical screen that sampled responses containing likelihood-related language much more densely than responses outside that frame. A design analysis showed that this sampling imbalance would leave a whole-population stance estimate too imprecise at the available annotation budget. We therefore report no population prevalence of stance and do not use this analysis for a main claim.

C Statistical analyses and robustness

C.1 No-clause baseline

The primary endpoint comparison uses the 0.1% and 80% numeric conditions, which contain the same sentence template and differ only in the stated chance. For completeness, we also compare the no-clause baseline with the 0.1% condition. Under the judge classifier, the paired increase is $+3.70$ percentage points, with a crossed model-by-request 95% confidence interval of $[-0.70, +7.90]$. This baseline contrast is not the primary estimand because it changes both sentence presence and the stated chance. The filler controls in Appendix B.1 separately test sentence presence and the addition of an unrelated percentage.

C.2 Crossed uncertainty for the primary endpoint

The primary $0.1\% \to 80\%$ contrast averages the within-request change over models and requests. Judge and rule estimates use the same matched responses with an order-stable judge label. Confidence intervals resample both models and requests as crossed clusters. Under this analysis, refusal increases by $+7.50$ percentage points under the judge classifier (95% CI $[+3.70, +11.70]$) and by $+4.80$ points

under the rule classifier ($[+1.20, +9.20]$). The point estimates are positive in 14 of 16 models under the judge classifier and 11 of 16 under the rule classifier. The judge-minus-rule difference is $+2.71$ percentage points (crossed 95% CI $[-0.27, +6.17]$).

C.3 TOXIC–HARD-1K COMPARISON

We apply the same endpoint contrast separately to the 42 toxic and 42 Hard-1K requests. Requests are resampled within source pool so that the fixed 42/42 split is preserved. Under the judge classifier, the $0.1\% \rightarrow 80\%$ increase is $+5.40$ percentage points in the toxic pool and $+9.70$ points in Hard-1K. The toxic-minus-Hard-1K interaction is therefore -4.30 points (95% CI $[-12.20, +3.30]$). Under the rule classifier, the corresponding increases are $+3.40$ and $+6.20$ points, giving an interaction of -2.80 points ($[-8.70, +2.50]$). Within Hard-1K alone, the endpoint increase is $+9.70$ points under the judge classifier (95% CI $[+3.50, +16.60]$) and $+6.20$ points under the rule classifier ($[+1.30, +12.10]$). The interaction is therefore a non-detection of a difference between source pools, whereas the within-Hard-1K contrast shows that the response is also present in that pool. The Hard-1K increase is also positive when the 80% condition is compared with the no-clause baseline, the matched sentence, or the unrelated-percentage filler. Against the no-clause baseline, the increase is $+16.00$ points under the judge classifier (95% CI $[+8.30, +24.20]$) and $+9.70$ under the rule classifier ($[+3.90, +16.70]$). Against the matched sentence, the corresponding effects are $+15.10$ ($[+8.20, +22.30]$) and $+8.30$ ($[+3.20, +14.80]$). Against the unrelated-percentage filler, they are $+14.80$ ($[+7.80, +22.20]$) and $+7.60$ ($[+2.60, +13.90]$).

C.4 SENSITIVITY TO SCALE AND HEADROOM

The toxic and Hard-1K pools begin at different refusal levels, so the endpoint interaction may depend on scale and available headroom. We therefore repeat the comparison on the log-odds scale, after headroom normalization, and after restricting model–request cells according to refusal in the independently generated no-clause baseline (Table 3). These are sensitivity analyses; the primary estimand remains the percentage-point interaction reported in §4.3.

Table 3: Sensitivity of the endpoint toxic-minus-Hard-1K interaction to scale and available headroom. Negative values indicate a larger Hard-1K response. The primary and restriction rows use crossed model-by-request bootstrap intervals. The log-odds row reports the Laplace (`nAGQ=1`) point estimate, while crossed bootstrap refits use the `nAGQ=0` approximation. The 10–90% baseline restriction retains 173 toxic and 113 Hard-1K cells under the judge classifier and 137 toxic and 51 Hard-1K cells under the rule classifier. The 20–80% restriction retains 127 toxic and 76 Hard-1K cells under the judge classifier and 74 toxic and 26 Hard-1K cells under the rule classifier.

Analysis	Scale	Judge	Rule
Primary endpoint	percentage points	-4.30 $[-12.20, +3.30]$	-2.80 $[-8.70, +2.50]$
Log-odds model	log odds	-0.30 $[-0.80, +0.18]$	-0.71 $[-1.42, -0.01]$
Headroom normalization, cell-pooled	fraction of available headroom	$+0.03$ $[-0.12, +0.17]$	-0.02 $[-0.09, +0.05]$
Headroom normalization, model-level	fraction of available headroom	$+0.04$ $[-0.21, +0.20]$	$+0.06$ $[-0.05, +0.17]$
Baseline 10–90%	percentage points	-4.20 $[-20.10, +12.90]$	-13.20 $[-28.30, +2.60]$
Baseline 20–80%	percentage points	-2.60 $[-21.70, +15.60]$	-17.30 $[-38.60, +7.10]$

Headroom normalization moves the interaction toward zero and can change its sign depending on the aggregation. The log-odds estimates and the baseline-defined restrictions are negative. The exact interaction estimate is therefore sensitive to scale and aggregation. None of these analyses provides evidence of a larger toxic-pool response, and we do not interpret the negative percentage-point estimate as evidence that Hard-1K responds more. These analyses also do not establish equality between the source pools.

C.5 Detectability of a positive interaction

We characterize the repeated-study sensitivity of the primary interaction analysis using simulation. Each simulated dataset preserves the observed crossed model-by-request structure, the number of responses in each cell, and the observed heterogeneity. We impose known positive toxic-minus-Hard-1K interactions and rerun the primary crossed-bootstrap analysis. Detection is defined as a 95% confidence interval lying entirely above zero. Under the primary simulation, a positive interaction of approximately $+11.60$ percentage points under the judge classifier and $+8.10$ points under the rule classifier is detected in half of simulated studies. Approximately $+19.50$ and $+13.90$ points, respectively, are required for 80% detection. Alternative ways of imposing the interaction change these thresholds. We therefore treat this analysis as a characterization of design sensitivity rather than as a second inferential test. It shows that a repeated study of this size has limited sensitivity to modest positive interactions and does not support interpreting the primary non-detection as equality.

C.6 Truncation

Generation is capped at 256 new tokens. Overall truncation decreases as stated chance increases, but truncated responses are concentrated among compliances rather than refusals. Among the four API models, for example, truncation among responses labeled as refusals ranges from 2.40% to 14.20%, compared with 79.70% to 96.40% among compliances. Truncation among refusals changes by -2.00 percentage points on average between the 0.1% and 80% conditions. We bound the possible effect of this truncation by treating every truncated response labeled as a refusal as a misclassified compliance. The bound is computed on the unweighted per-model version of the endpoint contrast, which is $+8.53$ percentage points before correction. Under the worst-case reclassification it becomes $+7.83$ points, a change of -0.70 points, and remains positive in 14 of 16 models. A continuation study extending the generation limit from 256 to 1,024 tokens on nine open-weight models suggests a smaller correction, but it does not cover the API models. We therefore use the worst-case calculation as the relevant bound.

C.7 Re-run stability

Generation is unseeded, so a new run is a replication rather than a bit-exact reproduction. A resampling harness draws $2{,}000$ simulated re-runs from the observed response distribution. The numeric $0.1\% \rightarrow 80\%$ endpoint effect remains positive in all simulated re-runs under both classifiers. Per-model endpoint effects are less stable: in the middle 95% of simulated re-runs, 13–15 models have a positive endpoint change under the judge classifier and 9–13 under the rule classifier. This analysis addresses sampling variability under the observed response distribution, not changes to provider-side models.