Challenges in Trustworthy Human Evaluation of Chatbots

Wenting Zhao **Alexander M. Rush** **Tanya Goyal**
{wz346,tg436}@cornell.edu

Abstract

Open community-driven platforms like Chatbot Arena that collect user preference data from site visitors have gained a reputation as one of the most trustworthy publicly available benchmarks for LLM performance. While now standard, it is tricky to implement effective guardrails to collect high-quality annotations from humans. In this paper, we demonstrate that three sources of bad annotations, both malicious and otherwise, can corrupt the reliability of open leaderboard rankings. In particular, we show that only 10% of poor quality votes by apathetic (site visitors not appropriately incentivized to give correct votes) or adversarial (bad actors seeking to inflate the ranking of a target model) annotators can change the rankings of models by up to 5 places on the leaderboard. Finally, we discuss open challenges in ensuring high-quality human annotations.

1 Introduction

Reliable evaluation of free-form text generation quality is a long-standing challenge in NLP (Gehrmann et al., 2023; Celikyilmaz et al., 2020; Goyal et al., 2022a). Despite limitations, human annotation is widely accepted as the gold standard, especially for open-ended text generation tasks without an objective notion of correctness. As a result, platforms such as Chatbot Arena (Zheng et al., 2023; Chiang et al., 2024b) and WildVision Arena (Lu et al., 2024) that allow users to interact with available large language models (LLMs) and submit preference judgments for model pairs, have become extremely valuable resource in the NLP evaluation landscape. By providing free and easy access to available LLMs, these community-driven platforms are able to incentivize millions of user interactions[1] and collect a large-scale and diverse dataset of user queries and preferences. Deservedly, these peer production and community-driven platforms have emerged as one of the most trusted benchmarks in NLP today.[2]

Moreover, such benchmarks play a crucial role in auditing automatic evaluators by providing the necessary ground truth rankings that any evaluator can be validated against. In fact, the most popular automatic evaluation benchmarks today, including AlpacaEval (Li et al., 2023), WildBench (Lin et al., 2024), MixEval (Ni et al., 2024) and Arena-Hard (Li et al., 2024), validate their metric by reporting high correlation with Chatbot Arena judgments.

Given its far-reaching impact, both on human and automatic benchmarking of LLMs, and consequently on LLM research more broadly, it is crucial to ensure that the model rankings on these open community leaderboards are trustworthy. However, challenges with obtaining high-quality human judgments from non-expert crowdworkers like Chatbot Arena users are widely discussed in literature (Karpinska et al., 2021; Clark et al., 2021; Hosking et al., 2024). Moreover, these platforms typically implement minimal quality controls for verifying annotation quality such as attention checks, user verification, etc. This sits in direct opposition to the goals of trustworthiness. In this paper, we play devil's advocate and ask: **is it even possible to ensure the reliability of a community-driven open platform, like Chatbot Arena, without sacrificing user scale?**

We approach this thought experiment from two angles. First, using Chatbot Arena as a case study, we consider three different sources of poor quality preference judgments or votes in the collected dataset: un-incentivized or **apathetic** users providing random judgments (Section 3.1), malicious actors launching **adversarial** attacks to detect and artificially inflate a target model's ranking (Sec-

[1] As of October 6, 2024, Chatbot Arena has collected 2,011,939 pairwise preference judgments.

[2] As an example, Google's Chief Scientist used high performance on Chatbot Arena to declare the success of their recent model release: https://tinyurl.com/55xs2pz4.

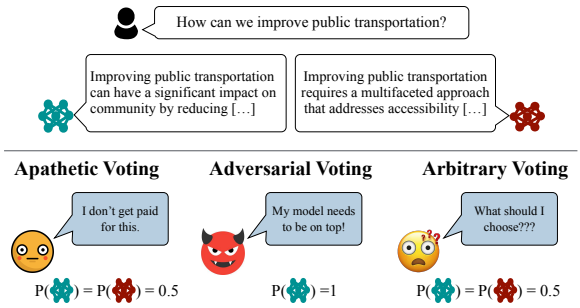

Figure 1: Our characterization of sources of poor-quality votes on open data annotation platforms: (1) Apathetic: Users who lack intrinsic motivation may submit random votes. (2) Adversarial: Malicious users aim to manipulate rankings by upvoting a target model. (3) Arbitrary: Users voting based on subjective preferences in response to open-ended questions.

tion 3.2), and the inherent **arbitrariness** of preference votes for open-ended and subjective queries (Section 3.3). For the former two sources of votes, we show that small fractions of poor-quality judgments (either apathetic or adversarial) can have a non-trivial impact on the target models' rankings. Concerningly, poor annotations from either apathetic or adversarial voting are not easy to detect in a post-hoc manner. Moreover, even carefully recruited and onboarded human annotators exhibit low inter-annotator agreement on subjective queries, making inter-annotator-based techniques to filter out low-quality annotations ineffective.

Finally, we discuss open challenges in ensuring the reliability and human annotation quality in open-source community-driven benchmarks (Section 4). We strongly believe that open data collection platforms offer an invaluable resource for the academic community and have facilitated essential work in developing new automatic evaluators (Li et al., 2023; Lin et al., 2024; Ni et al., 2024), training and evaluating reward models (Lambert et al., 2024), etc. However, critical questions exist about their reliability, especially against adversarial attacks. We hope that our work will spur future research on quality control mechanisms for open platforms that power LLM evaluations.

2 Background

In this paper, we run experiments with Chatbot Arena (Zheng et al., 2023; Chiang et al., 2024a) as a case study, although our insights are broadly applicable to other similar community-driven preference collection platforms. Below, we describe the preference collection pipeline and quality control measures employed by Chatbot Arena.

Notation Assume there are k different models $\mathcal{M} = \{m_1, m_2, ..., m_k\}$ that need to be ranked on the leaderboard. Each new user on the platform submits a query x and receives outputs from two different models $y_i \sim m_i(x)$ and $y_j \sim m_j(x)$.[3] The user has the option to submit a preference label $l \in \{i, j, \text{tie}\}$. In order to ensure that this annotation is unbiased, the names of the models that the two outputs are sampled from is only revealed to end users after they have submitted their preference annotation. This arena logs data points of the form: $(x, y_i, y_j, m_i, m_j, l)$.

These preferences are then used to estimate the pairwise win matrix between model pairs, i.e. $p(m_i > m_j)$. Next, they estimate the coefficients of the Bradley-Terry model (Bradley and Terry, 1952) to obtain scores s_i for each model $m_i \in \mathcal{M}$. Models are sorted by s_i to obtain the final ranking.

Quality control measures The arena employs a list of filtering strategies: detecting malicious users according to a certain distribution (Section 5.1; Chiang et al. (2024a)), bot detection by Cloudflare and Google reCAPTCHA v3, automatic categorization pipelines to filter out low-quality data[4][5], placing limits on the number of votes each IP can provide in a day, and deduplicating top 0.1% occurring prompts. However, these filtering strategies focus more on filtering bots than differentiating user votes with varying qualities. Therefore, we present results and discussions in this paper assuming minimal quality control checks in the backend to filter out bad quality user annotations[6].

Released Artifacts We conduct our experiments using the largest publicly released dataset by Chatbot Arena. It consists of 55k preference annotations[7]; it includes response pairs sampled from two of 64 unique models and the corresponding pairwise preference annotation.

[3]The arena employs an adaptive sampling strategy that favors model pairs with higher uncertainty in relative performance, and also newly introduced models. However, exact details are not publicly shared, possibly to mitigate gaming.

[4]https://blog.lmarena.ai/blog/2024/hard-prompts/

[5]https://blog.lmarena.ai/blog/2024/arena-category/

[6]https://github.com/lm-sys/FastChat/

[7]https://huggingface.co/datasets/lmsys/lmsys-arena-human-preference-55k

Model	Leaderboard Ranking			
	Orig.	r=1	r=5	r=10
Llama-2-7b-chat	21	21	$20_{\uparrow 1}$	21
Llama-2-13b-chat	39	39	$41_{\downarrow 2}$	$34_{\uparrow 5}$
Mistral-7b-instruct-v0.2	36	$38_{\downarrow 2}$	$38_{\downarrow 2}$	$41_{\downarrow 5}$

Table 1: Change in leaderboard rankings for 3 test models based on different percentages (r) of arbitrary votes. The subscripts denote gain (↑) or loss (↓) in rankings. We find that only 10% poor quality annotations can change the rank of 2/3 systems by 5 places.

3 Case Studies: Sources of Poor Quality Votes and Their Impact

For our thought experiment, we hypothesize that there exist three potential sources of poor quality votes on open platforms: (a) apathetic votes by users that are un-incentivized, (b) adversarial votes that aim to inflate the ranking of a target model, and (c) arbitrary votes on difficult to meaningfully distinguish response pairs. For each of these, we study their impact on model rankings and the challenges in mitigating them.

3.1 Apathetic Voting

The main attraction of open community platforms for end users is that they expose a free and easy-to-use API endpoint for LLMs. This incentivizes diverse users to interact with the platform and submit queries to explore their use cases. However, these platforms do not explicitly incentivize high-quality preference annotation. We hypothesize that at least r% of users on the arena are apathetic and provide random or low-quality votes on the platform.

Setup We run experiments on Chatbot Arena's dataset of 55k preferences (discussed in Section 2). We assume that this dataset reflects "true" rankings of models based on gold human preferences. We study the change in model rankings for 3 arbitrarily selected models: Llama-2-7b-chat, Mistral-7b-instruct-v0.2, and Llama-2-13b-chat, assuming r% of these preferences were instead assigned random labels by apathetic users during data collection.

Results Table 1 summarizes our results. **We find that only 10% of apathetic votes in the dataset can change the leaderboard rankings of 2/3 models by 5 places** (namely Llama-2-13b-chat and Mistral-7b-instruct-v0.2).[8] Note that there are no existing studies characterizing the incentives or behaviors of an average user on open platforms like Chatbot Arena. Therefore, we have no way of estimating the fraction r of apathetic.

Discussion: Can we detect and remove apathetic votes? A major challenge in detecting apathetic votes is that they are often indistinguishable from arbitrary votes. Multiple past studies have found that output-level comparisons using a single label is ill-defined as an annotation task (Krishna et al., 2023; Goyal et al., 2022a) as users often rely on different criteria and disagree with each other. This ambiguity makes it hard to ascertain whether observed disagreements are due to personal variations in quality assessment (arbitrary voting, discussed further in Section 3.3) or due to apathetic or low-quality annotations by certain annotators. Despite challenges with detecting individual apathetic votes, detecting apathetic users may be viable by computing agreements between model rankings by individual users. This strategy is based on the intuition that while annotators might disagree on specific examples, their aggregate system-level judgments tend to be more aligned (Goyal et al., 2022a). Finally, requesting additional justifications for votes, such as free-text rationales, can also help discourage apathetic votes. We discuss this more in Section 4.

3.2 Adversarial Voting

We assume there exists a malicious developer who seeks to inflate the rankings of their own target model m_T on the arena leaderboard A. We argue that due to the lack of quality controls (e.g. user verification, attention checks, etc.), it is straightforward to inject preference votes for m_T using a simple attack methodology.

Our main component is a **target model attribution algorithm** which, given a query-output pair (q, y), predicts whether y is sampled from the target model $m_T(q)$. Given such an algorithm, we can inflate the ranking of the target model m_T using the following strategy: (1) Enter a prompt q on the arena, (2) Detect if any of the two shown outputs y_1, y_2 are sampled from m_T, (3) If yes, vote for the target model m_T, (4) Repeat.

Target model attribution algorithm We assume that the attribution algorithm has access to the target model logits. This is a reasonable assumption for our setting where a model developer seeks to

[8] Note that model frequency also impacts its susceptibility to ranking changes. All three models we inspect collectively occur in less than 10% of all data samples.

Model	Leaderboard Ranking				
	Orig.	r=1	r=5	r=10	r=100
Llama-2-7b-chat	21	$23_{\downarrow 2}$	21	$17_{\uparrow 4}$	$1_{\uparrow 21}$
Llama-2-13b-chat	39	$36_{\uparrow 3}$	$32_{\uparrow 5}$	$28_{\uparrow 9}$	$1_{\uparrow 39}$
Mistral-7b-instruct-v0.2	36	$34_{\uparrow 2}$	$34_{\uparrow 2}$	$29_{\uparrow 7}$	$2_{\uparrow 34}$

Table 2: Change in leaderboard rankings for 3 test models based on different percentages (r) of adversarial votes (upvoting the target model). We find that only 10% adversarial annotations can change the rank of all systems by more than 4 places.

Model	TPR	TNR	#Tokens
Llama-2-7b-chat	91.13	88.46	328.06
Llama-2-13b-chat	100.00	89.93	326.53
Mistral-7b-instruct-v0.2	91.28	86.69	319.46

Table 3: Intrinsic eval. of model attribution algorithm

inflate rankings. Our simple attribution algorithm is outlined in Algorithm 1 in Appendix A.

Essentially, we use teacher-forcing to determine the probability distribution over the vocabulary for all tokens at time step t, i.e. $P_{m_T}(.|x, y_1, ...y_{t-1})$. We sort the tokens in descending order of probability to identify the smallest subset of tokens that cover a cumulative probability mass of at least p. We compute the fraction of generation time steps t for which the actual generated token y_t falls within this top-p probability subset. We compare this against a threshold t to classify generations $y = y_1...y_N$ as being sampled from m_T or not.

Intrinsic Evaluation of Detector Algorithm For all three test models, we report the true positive rate (TPR) and true negative rate (TNR) on the arena dataset in Table 3. We find that our detector algorithm reports very high performance (e.g. TPR=91.13%, and TNR=88.46% for Llama-2-7b-chat). We also find a positive correlation between the number of tokens and TPRs, which can be leveraged in the attack. Note that malicious actors can always improve the detector accuracy further using watermarking techniques (Kirchenbauer et al., 2023). Next, we use these highly performant models to cast adversarial votes.

Can we influence voting on the live Chatbot Arena platform? We also implement a proof-of-concept of a real "attack" on Chatbot Arena to demonstrate that current guardrails, such as bot detection, can be bypassed easily. On October 13, 2024, we programmatically launched 100 queries into Chatbot Arena, extracted the two model responses, and successfully submitted a preference vote. To avoid contaminating the dataset, we only cast "tie" votes but note that it would be trivial to instead use the vote from the attribution algorithm.

Interestingly, post-hoc analysis of this data revealed that `yi-lightning` family of models, released just 2 days later, were the most common (20% of the responses) in this set.[9] We assume that Chatbot Arena had early access to these models and sampled them more frequently than others in order to collect enough votes. However, this knowledge of when particular models will be up-sampled can be easily exploited by adversaries to log a large fraction of upvotes for their model.

Impact of adversarial voting on leaderboard rankings Similar to Section 3.1, we run experiments on the 55k preference dataset from Chatbot Arena, assumed to reflect "true" votes. For 3 target models, we report the change in leaderboard rankings if adversarial voting was conducted on r% of the data samples during data collection. Table 2 summarizes our results. Across all models, we show that adversarial attacks can substantially change leaderboard rankings if adversaries get to contribute 10% votes for their model.[10]. Note that, in this work, we only report results using the most simplistic version of this attack. We can further boost these numbers by not only upvoting the target model but also downvoting open-source competitor models or those ranked higher than the target model in the leaderboard.

Discussion: Can we detect and remove adversarial votes? Open platforms can employ two types of mitigation strategies to address this issue: recognizing bot-like behavior to prevent votes from being cast, or detecting abnormal users post-hoc to filter out their votes. Platforms like Chatbot Arena already implement measures from both categories. For example, Chatbot Arena uses Cloudflare and Google reCAPTCHA to detect bots on their platform; however, we were able to bypass both programmatically. We did not find public information indicating that similar measures have been incorporated into the Wildvision Arena platform.

There are also opportunities to detect anomalous users post-hoc based on behaviors across multiple

[9] Evenly distributed between `yi-lightning` and `yi-lightning-lite`.

[10] We assume that adversaries can get 10% votes towards their own model because newly released models will be sampled more frequently.

	Th	Org	Re	Per	WS
GPT-3.5 vs GPT-4o	-0.36	5.51	9.91	17.18	20.06
Llama-3-8b vs Llama-3-70b	10.15	-10.78	27.16	5.78	8.50
Llama-3-8b vs GPT-3.5	11.34	7.19	-12.15	3.53	7.45
Llama-3-70b vs GPT-4o	3.15	-1.27	4.56	2.75	-4.66

Table 4: Fleiss' Kappa between four annotators on different evaluation axis: Th(esis), Org(anization), Re(asoning), Per(spectives), WS (Writing Style).

sessions or votes. Chatbot Arena implements a version of this strategy by comparing the distribution of ratings from a user (uniquely identified by IP address) against historical distributions to identify anomalies. Because committed adversaries may bypass these checks using IP rotation or similar techniques, we encourage further exploration of these approaches to make them more robust.

3.3 Arbitrary Voting

We assume an idealized scenario where all users genuinely make their best effort to rank model outputs. However, we argue that holistically rating a response to an open-ended and inherently subjective query is ill-defined and liable to always be arbitrary. To demonstrate this, we conduct a small-scale annotation study for outputs of subjective *Researchy* questions' prompts (Rosset et al., 2024).[11]

Setup We use these prompts and generate generate responses from four language models: Llama-3-8B, Llama-3-70B, GPT-4o, and GPT-3.5. We recruit four undergraduate CS students who are passionate about NLP and committed to providing thoughtful annotations. They evaluate responses on four dimensions: thesis, organization, reasoning, perspectives, and writing style. We offer them unlimited time and allow them to seek clarification from the authors when needed. Note that this dimension-wise rating is different from Chatbot Arena's setup of pairwise preferences. However, there already exist multiple prior works that argue that the task is under-defined in this latter setting and report low agreement between annotators (Goyal et al., 2022a,b; Krishna et al., 2023). Therefore, we opt to run this study using a more well-defined task description.

Results Table 4 shows the inter-annotator agreement between the annotators. Overall, we find very low agreement between these well-intentioned annotators with clear guidelines, irrespective of the performance difference between the model pairs. More concerningly, the results highlight that traditional approaches like filtering out low-quality users/annotations using inter-annotator agreement may not be a viable strategy for open-ended queries as it is difficult to disentangle between of low inter-annotator agreement due to bad annotation (apathetic votes) or inherent subjectivity. Adversarial users can also "hide" their votes from similar scrutiny by using open-ended prompts for which vote choice is expected to be ambiguous.

Discussion We argue that arbitrary votes are not "noise" and provide useful signals about models' relative performance. If most frontier models perform similarly well on a substantial fraction of real-world queries, this information should not be discarded but inform leaderboard Elo scores. Arbitrary votes become problematic when the majority of the leaderboard is dominated by open-ended queries that fail to meaningfully distinguish models, despite the existence of legitimate topics or skills along where models exhibit distinct behaviors. Identifying which test examples (or type of test examples) are most informative and up-weighting them when deriving aggregate scores are potential ways of addressing this (Rodriguez et al., 2021).

4 Conclusion & Future Directions

Our experiments in Section 3 lay a convincing case for the need for stronger guardrails in open community-driven platforms. Although these are broadly accepted as the ground truth rankings of LLMs, we are concerned that it is easy to intentionally (adversarial) or unintentionally (apathetic, arbitrary settings) corrupt these leaderboards. The key challenge in mitigating the issue of poor quality annotations is: how can community-driven platforms strike the right balance between implementing necessary quality controls while also providing the right incentives and experience to users to continue to use these platforms.

Richer feedback We encourage the community to explore ideas from past research, such as soliciting fine-grained annotations (Krishna et al., 2023; Goyal et al., 2022b) or rationales (McDonnell et al., 2016) in addition to the binary preference feedback. Rationales can be useful in encouraging apathetic

[11] Representative question: "How can the education system be improved?".

users to think more critically about their votes (or abstain) and also for filtering out low-quality annotations from both apathetic and adversarial users.

Past work in generation evaluation has discussed how binary preference, or even a single Likert rating, for the whole output, cannot meaningfully capture the nuances of human preferences (Gehrmann et al., 2023). Instead, fine-grained preference annotation is recommended, both along multiple dimensions or quality (Gehrmann et al.) or for smaller units within the whole output (Krishna et al., 2023; Goyal et al., 2022b). More recent work proposes providing added context during evaluation to encourage higher agreement between annotators (Malaviya et al., 2024). Future work must explore how these strategies can be incorporated into open platforms without inordinately increasing the annotation burden on users.

Stronger Guardrails Other guardrails could include reputation-based systems (Adler and de Alfaro, 2007), CAPTCHA (Von Ahn et al., 2003, 2008), machine learning based anomaly detection (Kumar et al., 2014; Wu et al., 2016) and techniques that use annotator behavior traces on the platform to estimate quality (Goyal et al., 2018).

Open access to collected dataset Public release of the collected data on open platforms will spur research to address the annotation issues we discuss in this work. It would provide a more detailed overview into which types of queries are most well-equipped to distinguish between models, and what are the limitations of different families of models.

Limitations

In this paper, we focus our analysis on one open community-driven platform, namely Chatbot Arena. However, there exist other similar platforms, like WildVision Bench, that implement similarly lax guardrails around annotation quality. Extending this analysis to such platforms can lead to added insights specific to vision language model evaluation.

Acknowledgements

We thank Deniz Bölöni-Turgut, Leo Lu, Aishanur Aydin, and Alicia Fulbright for providing model preference annotations on Researchy Questions. We thank the Chatbot Arena team for feedback on this draft. AMR was supported by NSF CAREER 2037519.

References

B. Thomas Adler and Luca de Alfaro. 2007. A content-driven reputation system for the wikipedia. In *The Web Conference*.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024a. Chatbot arena: An open platform for evaluating llms by human preference. *Preprint*, arXiv:2403.04132.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. 2024b. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A Smith. 2021. All that's 'human'is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. A case for better evaluation standards in nlg.

Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022a. News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356*.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022b. Snac: Coherence error detection for narrative summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 444–463.

Tanya Goyal, Tyler McDonnell, Mucahid Kutlu, Tamer Elsayed, and Matthew Lease. 2018. Your behavior signals your reliability: Modeling crowd behavioral traces to ensure quality relevance annotations. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 6, pages 41–49.

Tom Hosking, Phil Blunsom, and Max Bartolo. 2024. Human feedback is not gold standard. In *The Twelfth International Conference on Learning Representations*.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using mechanical turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. 2023. Longeval: Guidelines for human evaluation of faithfulness in long-form summarization. *arXiv preprint arXiv:2301.13298*.

Srijan Kumar, Francesca Spezzano, and VS Subrahmanian. 2014. Accurately detecting trolls in slashdot zoo via decluttering. In *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*, pages 188–195. IEEE.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2024. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024. From live data to high-quality benchmarks: The arena-hard pipeline.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*.

Yujie Lu, Dongfu Jiang, Wenhu Chen, William Yang Wang, Yejin Choi, and Bill Yuchen Lin. 2024. Wildvision: Evaluating vision-language models in the wild with human preferences. *arXiv preprint arXiv:2406.11069*.

Chaitanya Malaviya, Joseph Chee Chang, Dan Roth, Mohit Iyyer, Mark Yatskar, and Kyle Lo. 2024. Contextualized evaluations: Taking the guesswork out of language model evaluations. *arXiv preprint arXiv:2411.07237*.

Tyler McDonnell, Matthew Lease, Mucahid Kutlu, and Tamer Elsayed. 2016. Why is that relevant? collecting annotator rationales for relevance judgments. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 4, pages 139–148.

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. 2024. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *arXiv preprint arXiv:2406.06565*.

Pedro Rodriguez, Joe Barrow, Alexander Miserlis Hoyle, John P Lalor, Robin Jia, and Jordan Boyd-Graber. 2021. Evaluation examples are not equally informative: How should that change nlp leaderboards? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4486–4503.

Corby Rosset, Ho-Lam Chung, Guanghui Qin, Ethan C Chau, Zhuo Feng, Ahmed Awadallah, Jennifer Neville, and Nikhil Rao. 2024. Researchy questions: A dataset of multi-perspective, decompositional questions for llm web agents. *arXiv preprint arXiv:2402.17896*.

Luis Von Ahn, Manuel Blum, Nicholas J Hopper, and John Langford. 2003. Captcha: Using hard ai problems for security. In *Advances in Cryptology—EUROCRYPT 2003: International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4–8, 2003 Proceedings 22*, pages 294–311. Springer.

Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. 2008. recaptcha: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468.

Zhaoming Wu, Charu C Aggarwal, and Jimeng Sun. 2016. The troll-trust model for ranking in signed networks. In *Proceedings of the Ninth ACM international conference on Web Search and Data Mining*, pages 447–456.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A Model Attribution Algorithm

The model attribution algorithm used to carry out the adversarial attack in Section 3.2 is outlined below.

Algorithm 1 Model Attribution

Input: Target model m_T, input sequence x, output sequence $y = (y_1, y_2, \ldots, y_N)$, probability threshold p, decision threshold t

Output: 1 (y is likely from m_T); 0 (y is unlikely from m_T)

Initialize *results* $\leftarrow$ *an empty list*
for $i = 1$ to N **do**
    $\mathcal{D}_i \leftarrow \text{softmax}(P_{m_T}(x, y_1, \ldots, y_{i-1}))$
    $S_i^* \leftarrow \arg\min_{S_i} |S_i|$ s.t. $\sum_{t \in S_i} \mathcal{D}_i[t] \geq p$
    if $y_i \in S_i^*$ **then**
        Append 1 to *results*
    else
        Append 0 to *results*
    end if
end for
Compute confidence score $c \leftarrow \frac{\sum(\textit{results})}{N}$
if $c \geq t$ **then**
    return 1
else
    return 0
end if