Auditing Verbalized Distributions Under Human Label Variation

Anonymous ACL submission

Abstract

LLMs are increasingly used for data annotation, offering an alternative to using experts or crowd-workers. Given that annotators do not always agree on all items, numerous studies have attempted to quantify the extent to which LLMs can predict the distribution of judgments inherent to a population of human respondents. Though such knowledge can be elicited by various means, previous work found that prompting models to verbalize the distribution of subjective assessments pertaining to a certain stimulus is the most felicitous to observed, ground-truth samples. In this study, we investigate verbalized distribution expression, particularly as it is related to the presence of annotator disagreement. We propose three desiderata of probability distributions that verbalized strategies should conform to and evaluate six LLMs on these desiderata. We find that across verbalization strategies, LLMs largely fail at generating valid, human annotator-aligned probability distributions, calling into question the validity of the verbalized expression method.

1 Introduction

LLMs are increasingly utilized in dataset annotation tasks, promising an inexpensive and scalable alternative to the experts or crowd-workers typically employed for such purposes (Tan et al., 2024). The main assumption motivating such endeavors is that, in pre-training on massive, web-scale corpora, LLMs encode a "wisdom of the crowd" (Stroop, 1932) and are thus capable of representing aggregate human judgments over a variety of stimuli (Dillion et al., 2023; Trott, 2024). However, while models generally tend to yield high accuracy on trivially easy or high-consensus items, they have likewise been shown to fare poorly on ambiguous or contentious data points (Liu et al., 2023), often failing to replicate the diversity of human-generated data (Baumann et al., 2025; Giulianelli et al., 2023).

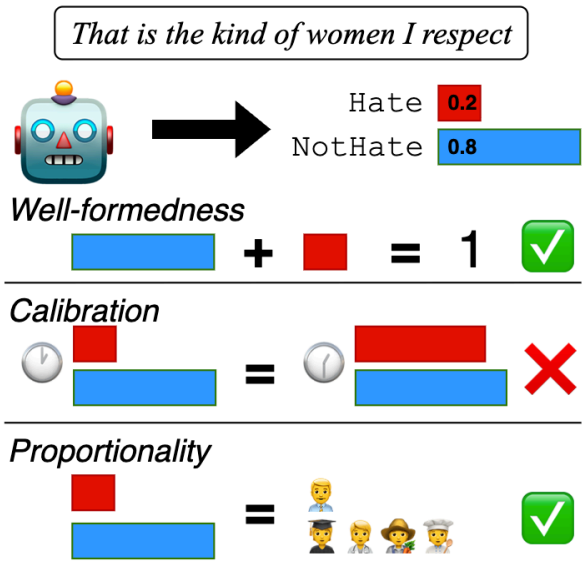

Figure 1: Overview of the three consistency desiderata for verbalization.

Significant interest, too, has been placed on aligning LLMs with *dis-aggregated* human judgments, ranging from linguistic stimuli to political opinion (Lee et al., 2023; Orlikowski et al., 2025). Efforts in this realm are typically framed in terms of "alignment", offering a glimpse into how well LLMs are calibrated to diverse — and potentially dissenting — human opinion (Sorensen et al., 2024). Operationally, this is generally accomplished by eliciting LLM distributions over categorical or ordinal data and matching these to human judgments via metrics such as KL divergence or total variation distance. In a benchmark study, Meister et al. (2025) investigate various distribution expression strategies, finding that *verbalization* — i.e., simply prompting an LLM to return categorical distributions in text format — serves as a more accurate and faithful estimator of human opinion than logit extraction or label sequence generation. Based on these findings, the verbalization approach has been adopted as a standard method

for distribution expression via LLMs (Suh et al., 2025; Ni et al., 2026).

Yet, given LLMs' non-deterministic nature and generally poor numeric calibration (Kumar et al., 2024; Lovering et al., 2025; Yuan et al., 2026), their general ability to reliably capture human judgment distributions remains questionable. This is touched upon by Meister et al. (2025), who observe that verbalized distributional alignment, while faring the best among surveyed methods, is nonetheless weak in more abstract, non-cultural and non-political prompting settings. To assess the reliability of verbalized distributions in more challenging annotation-oriented scenarios, we subject six LLMs to a series of consistency checks across two popular NLP tasks with high degrees of human label variation (HLV) (Plank, 2022): hate-speech detection (HSD) and natural language inference (NLI). Focusing on a comparison between instances of high and low annotator disagreement, we formulate three desiderata aimed at targeting distinct properties of distributions that a well-calibrated model should possess. Ultimately, we find that models largely behave in idiosyncratic fashion, often producing ill-formed, poorly calibrated, and disproportional distributions — namely in high-entropy settings. Furthermore, we observe a broad dissimilarity to attested human judgment distributions, calling into question the validity of verbalization as a model alignment paradigm.

2 Verbalized Distribution Desiderata

We investigate two distinct verbalization strategies:

- SINGLE: given a categorical distribution with $K = 2$ classes, the model is prompted to return the percentage (or absolute number) of annotators who select a specified label k, e.g. `[<float>]`.

- FULL: given a multi-class categorical distribution with $K \geq 2$ classes, the model is prompted to return a dictionary representing the percentage (or absolute number) of annotators that select each class $k \in \{1, \ldots, K\}$, e.g. `{"Hate": <float>, "NotHate": <float>}`

Alongside these strategies, we formulate a set of three desiderata that we deem jointly necessary to assess the validity of distribution verbalization as an alignment and evaluation method:

Well-formedness. *Distributions must be well-formed and represent the intended number of participants.* For SINGLE, we sample probabilities for classes a and b over $T = 5$ trials each ($2T = 10$ total). Let $p_i(a)$ denote the probability assigned to class a in trial $i \in \{1, \ldots, T\}$. We define the average error of $p_i(a)$ with respect to the distribution over class b as:

$$\mathcal{E}(a) = \frac{1}{T^2} \sum_{i=1}^{T} \sum_{j=1}^{T} |p_i(a) - p_j(b)|$$

The total error rate $\mathcal{E}$ for a dataset item is then averaged as $\mathcal{E} = \frac{1}{2}(\mathcal{E}(a) + \mathcal{E}(b))$. If the verbalized distributions are well-formed, we hypothesize that $\mathcal{E} \approx 0$. For FULL, we simply calculate the rate at which the predicted probability distribution sums to 1 in the $n = \text{random}$ condition (see Proportionality).[1]

Calibration. *Distributions must be well-calibrated, exhibiting minimal distance across identical prompting runs.* Let $\mathbf{p}_i$ denote the predicted (normalized) distribution vector in run $i \in \{1, \ldots, T\}$, with $T = 5$, e.g., $\mathbf{p}_i = [0.78, 0.10, 0.12]$. We require that:

$$\text{Cal.} = \frac{1}{T^2} \sum_{i=1}^{T} \sum_{j=1}^{T} \text{TVD}\left(\mathbf{p}_i, \mathbf{p}_j\right) \approx 0$$

For SINGLE, we infer $\mathbf{p}_i$ from $p_i(a)$ and $p_i(b)$ separately, where $\mathbf{p}_i = [p_i(a), 1 - p_i(a)]$ (and vice versa). We report $\mathbf{p}_i$ as averaged over the inferred distributions for classes a and b.

Proportionality. *Distributions must be proportional with respect to a varying number of participants.* Let $\mathbf{p}_{\text{norm}} = \frac{1}{T} \sum_{i=1}^{T} \mathbf{p}_i$ denote a probability distribution over K classes, averaged and normalized over T trials. For another distribution $\mathbf{p}_n$ over K, where $n \in \{100, 1000\}$ or is sampled randomly from $[100, 1000]$ participants, we require that:

$$\text{Prop.} = \text{TVD}\left(\mathbf{p}_{\text{norm}}, \mathbf{p}_n\right) \approx 0$$

Here, $\mathbf{p}_{\text{n}}$ represents the absolute number of n participants selecting each class $k \in \{1, \ldots, K\}$, normalized into a probability distribution. For SINGLE, we utilize inferred distributions, as described above.

[1] In practice, we find that this rate is approximately 1.0 for all surveyed models and thus do not report it further.

3 Experimental Setup

3.1 Datasets

We evaluate our approach on two tasks that capture fundamentally different sources of annotator disagreement: hate-speech detection (HSD) and Natural Language Inference (NLI). For HSD, we use the *Measuring Hate Speech* dataset (Kennedy et al., 2020), where judgments of English social media comments are heavily influenced by annotators' subjective backgrounds and lived experiences (Sap et al., 2022). In the original dataset, annotators rate samples on a scale from -1 - 'not hateful' to +1 'hateful'. As verbalization requires categorical data, we instead map these ratings to binary classes ('Hate' and 'NotHate'), considering annotations above 0.0 as 'Hate'. Conversely, for the NLI task — predicting the common sense relationship ('entailment', 'neutral' or 'contradiction') between a statement and a hypothesis — we employ the English *ChaosNLI* dataset (Nie et al., 2020), which features a rich distribution space of 100 annotations per sample. In contrast to HSD, disagreement in NLI stems from largely linguistic factors such as lexical ambiguity, implicature, or probabilistic enrichment (Jiang and de Marneffe, 2022).

We split each dataset into high and low-entropy partitions, each comprising 100 items (200 in total). For HSD, we consider the 100 lowest-rated items in terms of absolute value as high-entropy, and 50 items from the head and tail of the score distribution as low-entropy (separately)[2]. For NLI, we compute entropy over the (normalized) annotator distributions for the ChaosNLI-M split, and utilize the 100 highest and lowest-rated items. Since $K = 3$ for NLI, we only report the well-formedness $\mathcal{E}$ for HSD.

3.2 Models

We evaluate six LLMs spanning both open- and closed-source ecosystems. To ensure architectural diversity, our selection encompasses models from distinct families and varying parameter scales: `claude-opus-4.7` and its lightweight variant, `claude-haiku-4.5`; `gemini-3.1-pro`; `glm-5.1`; and Qwen 3.5 at both the 397B and 9B parameter scales. This selection was made with respect to model performance on the LMArena leaderboard,[3] approximately around April 2026. To ensure reproducibility and minimize stochastic variance between generations, the decoding temperature for all models was fixed at 0. For Gemini 3.1 Pro, the model's internal reasoning mechanism was set to its "low" setting. All inference was done using the OpenRouter platform.[4]

4 Results

We assess the verbalization capabilities across three dimensions: well-formedness, proportionality, and calibration. For each desideratum to hold, we expect the metrics to be around 0.0, with larger values indicating greater inconsistencies between trials and settings. We adopt SINGLE for HSD and FULL for NLI. Results using FULL for HSD can be found in Appendix A.2, and comparison metrics to human annotator distributions can be found in Appendix A.3.

4.1 Hate-Speech Detection

Our evaluation of hate-speech probability prediction reveals distinct trends across well-formedness, proportionality, and calibration. In terms of well-formedness, absolute error rates consistently rise alongside human disagreement, scaling from a low of 0.05 for `claude-haiku-4.5` in low-entropy settings up to 0.32 for `qwen3.5-9b` under high entropy (Figure 2, Table 6 in Appendix A.5). Regarding proportionality, the mean TVD generally increases when moving from the norm setting to prompts with a higher number of annotators, and rises further when annotator counts are replaced with random participant numbers (Table 1). Despite these variations, the models demonstrate stable calibration; the mean pairwise TVD between identical prompt runs remains low at approximately 0.03, maintaining a low standard deviation with only slightly higher variation observed for high-entropy items (Table 2). Table 3 in Appendix A.5 presents the results by label prompted for ('Hate' or 'NotHate').

4.2 Natural Language Inference

Similarly to HSD, for proportionality and calibration, there is a large gap between the consistency metrics of high and low entropy items, with high entropy items leading to larger variation. For proportionality, TVDs range from 0.05 to 0.16 per setting,

[2] Notably, the high-entropy split for `gemini-3.1-pro` was restricted to 98 samples due to a failure to generate valid outputs for the remaining two.

[3] https://lmarena.ai/

[4] https://openrouter.ai

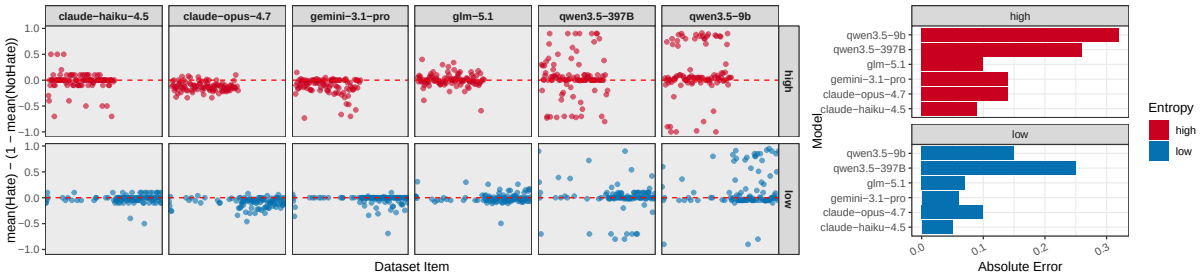

Figure 2: Error rates for the SINGLE setting on HSD. *Left:* Average per-item $\mathcal{E}$ values across models and entropy setting. *Right:* Average absolute error rate per-model and entropy setting.

Model	HSD 100 / 1000 / ran.	NLI 100 / 1000 / ran.
claude-haiku-4.5	0.04 / 0.04 / 0.06	0.11 / 0.16 / 0.15
claude-opus-4.7	0.06 / 0.05 / 0.06	0.12 / 0.15 / 0.15
qwen-3.5-9b	0.14 / 0.19 / 0.21	0.15 / 0.16 / 0.15
qwen-3.5-397b	0.12 / 0.16 / 0.17	0.12 / 0.13 / 0.14
gemini-3.1-pro	0.11 / 0.12 / 0.07	0.11 / 0.11 / 0.12
glm-5.1	0.06 / 0.12 / 0.12	0.17 / 0.13 / 0.17
claude-haiku-4.5	0.03 / 0.03 / 0.04	0.05 / 0.05 / 0.05
claude-opus-4.7	0.05 / 0.05 / 0.06	0.05 / 0.06 / 0.06
qwen-3.5-9b	0.08 / 0.10 / 0.18	0.09 / 0.07 / 0.08
qwen-3.5-397b	0.10 / 0.12 / 0.12	0.06 / 0.07 / 0.07
gemini-3.1-pro	0.05 / 0.06 / 0.05	0.05 / 0.05 / 0.06
glm-5.1	0.04 / 0.08 / 0.09	0.07 / 0.08 / 0.07

Table 1: Proportionality results for **HSD** and **ChaosNLI**, stratified across high (top) and low (bottom) entropy items. Values represent the TVD with the averaged normalized distribution and predictions for $n =$ 100/1000/random.

Model	HSD	NLI
claude-haiku-4.5	$0.00_{\pm 0.00}$	$0.10_{\pm 9.5}$
claude-opus-4.7	$0.02_{\pm 0.02}$	$0.08_{\pm 5.9}$
qwen-3.5-9b	$0.02_{\pm 0.02}$	$0.18_{\pm 12.8}$
qwen-3.5-397b	$0.05_{\pm 0.04}$	$0.14_{\pm 12.0}$
gemini-3.1-pro	$0.06_{\pm 0.04}$	$0.11_{\pm 12.9}$
glm-5.1	$0.04_{\pm 0.03}$	$0.18_{\pm 15.1}$
claude-haiku-4.5	$0.00_{\pm 0.00}$	$0.04_{\pm 4.0}$
claude-opus-4.7	$0.03_{\pm 0.02}$	$0.03_{\pm 3.1}$
qwen-3.5-9b	$0.01_{\pm 0.01}$	$0.09_{\pm 8.2}$
qwen-3.5-397b	$0.02_{\pm 0.02}$	$0.06_{\pm 7.0}$
gemini-3.1-pro	$0.05_{\pm 0.04}$	$0.04_{\pm 6.6}$
glm-5.1	$0.03_{\pm 0.02}$	$0.09_{\pm 9.7}$

Table 2: Calibration results for **HSD** and **ChaosNLI**, stratified across high (top) and low (bottom) entropy items. Values represent the mean pairwise TVD across runs (standard deviation in subscript).

with $n = 1000$ and $n = random$, showing greater deviation from the norm compared to $n = 100$. The calibration results indicate that there is a pronounced variation within trials of the same prompt with some models (`qwen-3.5-9b`, `glm-5.1`) reaching up to 0.18 mean pairwise TVD. The variation is less pronounced in the low entropy setting, but still reaches up to 0.09 TVD for `qwen-3.5-9b` and `glm-5.1`.

5 Conclusion

We establish three consistency desiderata necessary for a rigorous evaluation framework for verbalized probability distributions and subsequently evaluate two verbalization strategies on HSD and NLI. Our findings indicate that under the SINGLE setting for HSD, LLMs largely have difficulties generating well-formed probability distributions. Conversely, supplementary experiments (Appendix A.2) demonstrate that the FULL approach yields distinct improvements in both model calibration and proportionality. However, extending the FULL method to NLI reveals that consistency issues resurface within a multiclass context. Ultimately, these results indicate that while specific configurations of task, model, and elicitation method can produce well-defined outputs, such successes do not generalize broadly, thereby challenging the overall reliability of the approach. Deeper analyses of the mechanisms behind verbalization methods are difficult, as the method lacks grounding in the models' internal sampling distributions, therefore raising a fundamental theoretical question: what justifies the expectation that LLMs should inherently reflect human annotator behavior? While models internalize statistical patterns from pre-training corpora, it remains unproven whether they explicitly encode these patterns as representative distributions of human perspectives. Lacking an underlying meta-structure for authentic annotator distributions, LLMs depend entirely on superficial linguistic representations of opinions, which rarely conform to the rigorous logic of probabilistic inference.

Limitations

Our experiments are limited to two tasks on a small sample size, due to the high cost of running large closed-source models. We opted for a smaller coverage of 100 high and 100 low entropy data points for a few large LLMs to illustrate that the inconsistencies in verbalization exist not only for smaller open-source models, but also for state-of-the-art LLMs. Additionally, we include one lightweight LLM (`qwen3.5-9b`) and show that the inconsistencies are more severe for this smaller model.

References

Joachim Baumann, Paul Röttger, Aleksandra Urman, Albert Wendsjö, Flor Miriam Plaza-del Arco, Johannes B Gruber, and Dirk Hovy. 2025. Large language model hacking: Quantifying the hidden risks of using LLMs for text annotation. *arXiv preprint arXiv:2509.08825*.

Danica Dillion, Niket Tandon, Yuling Gu, and Kurt Gray. 2023. Can AI language models replace human participants? *Trends in Cognitive Sciences*, 27(7):597–600.

Mario Giulianelli, Joris Baan, Wilker Aziz, Raquel Fernández, and Barbara Plank. 2023. What comes next? Evaluating uncertainty in neural text generators against human production variability. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14349–14371, Singapore. Association for Computational Linguistics.

Nan-Jiang Jiang and Marie-Catherine de Marneffe. 2022. Investigating reasons for disagreement in natural language inference. *Transactions of the Association for Computational Linguistics*, 10:1357–1374.

Chris J Kennedy, Geoff Bacon, Alexander Sahn, and Claudia von Vacano. 2020. Constructing interval variables via faceted rasch measurement and multitask deep learning: A hate speech application. *arXiv preprint arXiv:2009.10277*.

Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. 2024. Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 315–334, Bangkok, Thailand. Association for Computational Linguistics.

Noah Lee, Na Min An, and James Thorne. 2023. Can large language models capture dissenting human voices? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4569–4585, Singapore. Association for Computational Linguistics.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah Smith, and Yejin Choi. 2023. We're afraid language models aren't modeling ambiguity. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 790–807, Singapore. Association for Computational Linguistics.

Charles Lovering, Michael Krumdick, Viet Dac Lai, Varshini Reddy, Seth Ebner, Nilesh Kumar, Rik Koncel-Kedziorski, and Chris Tanner. 2025. Language model probabilities are *not* calibrated in numeric contexts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29218–29257, Vienna, Austria. Association for Computational Linguistics.

Nicole Meister, Carlos Guestrin, and Tatsunori Hashimoto. 2025. Benchmarking distributional alignment of large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 24–49, Albuquerque, New Mexico. Association for Computational Linguistics.

Jingwei Ni, Yu Fan, Vilém Zouhar, Donya Rooein, Alexander Miserlis Hoyle, Mrinmaya Sachan, Markus Leippold, Dirk Hovy, and Elliott Ash. 2026. Can reasoning help large language models capture human annotator disagreement? In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 36–54, Rabat, Morocco. Association for Computational Linguistics.

Yixin Nie, Xiang Zhou, and Mohit Bansal. 2020. What can we learn from collective human opinions on natural language inference data? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Matthias Orlikowski, Jiaxin Pei, Paul Röttger, Philipp Cimiano, David Jurgens, and Dirk Hovy. 2025. Beyond demographics: Fine-tuning large language models to predict individuals' subjective text perceptions. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2092–2111, Vienna, Austria. Association for Computational Linguistics.

Barbara Plank. 2022. The "problem" of human label variation: On ground truth in data, modeling and evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10671–10682, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2022. Annotators with attitudes: How annotator beliefs

and identities bias toxic language detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5884–5906, Seattle, United States. Association for Computational Linguistics.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, Tim Althoff, and Yejin Choi. 2024. Position: A roadmap to pluralistic alignment. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

John Ridley Stroop. 1932. Is the judgment of the group better than that of the average member of the group? *Journal of Experimental Psychology*, 15(5):550.

Joseph Suh, Erfan Jahanparast, Suhong Moon, Minwoo Kang, and Serina Chang. 2025. Language model fine-tuning on scaled survey data for predicting distributions of public opinions. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21147–21170, Vienna, Austria. Association for Computational Linguistics.

Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation and synthesis: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 930–957, Miami, Florida, USA. Association for Computational Linguistics.

Sean Trott. 2024. Large language nodels and the wisdom of small crowds. *Open Mind*, 8:723–738. _eprint: https://direct.mit.edu/opmi/article-pdf/doi/10.1162/opmi_a_00144/2374096/opmi_a_00144.pdf.

Jiayi Yuan, Hao Li, Xinheng Ding, Wenya Xie, Yu-Jhe Li, Wentian Zhao, Kun Wan, Jing Shi, Xia Hu, and Zirui Liu. 2026. Understanding and mitigating numerical sources of nondeterminism in LLM inference. *Advances in Neural Information Processing Systems*, 38:169819–169851.

A Appendix

A.1 Per Class Metrics for SINGLE

In our experiments for proportionality and calibration in Tables 1 and 2 for SINGLE, we report the mean predictions across the two prompt settings 'Hate' and 'NotHate'. In Table 3, we show the disaggregated metrics. There is a general trend that the consistency metrics for proportionality and calibration are worse for the 'NotHate' prompt, compared to the 'Hate' prompt. Especially in the high entropy setting, there are large consistency gaps between the prompt settings. This underlines the fact that verbalizing single labels is

A.2 FULL Distribution Setting for HSD

For HSD, we also ran the consistency checks using the FULL distribution method as an alternative method to SINGLE (see Table 4 for the results per check). In contrast to the SINGLE setting, the well-formedness checks are performed on the $n = random$ setting. Across all checks, the consistency is lower in the low entropy setting, as is the case for SINGLE. However, compared to the SINGLE setting, FULL is decidedly more consistent with metrics close to 0.00 - 0.1 for all checks. An outlier for proportionality is `qwen3.5-9b`, which produces varying distributions depending on which number of annotator participants is used, with the TVD going up to 0.17 in the random setting.

In general, this approach offers a more consistent approach to elicit the probability distribution for binary tasks.

A.3 Comparison with Human Annotations

Table 5 shows the average TVD between human annotator distributions and the mean predicted distribution in the norm setting. The values range from 0.118 for `gemini-3.1-pro` for low entropy items using the SINGLE prompt to 0.358 on high entropy items using `qwen3.5-9b` and the FULL prompt.

A.4 Inference

For running our experiments, we used the openrouter.ai cloud computing clusters, which allocate resources across suitable servers, depending on their availability. As such, tracking the precise number of GPU hours is difficult and we report the number of calls instead. We ran four sets of prompts(FULL - NLI, FULL - HSD, SINGLE - HSD - Hate and SINGLE - HSD - NotHate) with 1.200 calls per LLMs, resulting in 28.800 model calls.

Model	Proportionality						Calibration	
	$n = 100$		$n = 1000$		$n = random$			
	Hate	NotHate	Hate	NotHate	Hate	NotHate	Hate	NotHate
claude-haiku-4.5	0.06	0.03	0.06	0.05	0.09	0.09	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$
claude-opus-4.7	0.08	0.06	0.07	0.07	0.08	0.09	$0.03_{\pm 0.02}$	$0.03_{\pm 0.03}$
qwen3.5-9b	0.10	0.21	0.12	0.27	0.14	0.33	$0.01_{\pm 0.01}$	$0.03_{\pm 0.03}$
qwen3.5-397b	0.08	0.20	0.11	0.29	0.17	0.36	$0.02_{\pm 0.02}$	$0.08_{\pm 0.07}$
gemini-3.1-pro	0.13	0.11	0.14	0.15	0.08	0.09	$0.06_{\pm 0.04}$	$0.09_{\pm 0.06}$
glm-5.1	0.10	0.11	0.13	0.25	0.11	0.22	$0.04_{\pm 0.03}$	$0.06_{\pm 0.05}$
claude-haiku-4.5	0.04	0.04	0.04	0.05	0.05	0.05	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$
claude-opus-4.7	0.05	0.06	0.06	0.07	0.06	0.07	$0.02_{\pm 0.02}$	$0.03_{\pm 0.02}$
qwen3.5-9b	0.08	0.09	0.10	0.10	0.22	0.17	$0.02_{\pm 0.01}$	$0.03_{\pm 0.03}$
qwen3.5-397b	0.05	0.16	0.12	0.20	0.08	0.23	$0.01_{\pm 0.01}$	$0.04_{\pm 0.03}$
gemini-3.1-pro	0.06	0.05	0.06	0.07	0.05	0.06	$0.03_{\pm 0.02}$	$0.05_{\pm 0.04}$
glm-5.1	0.06	0.06	0.08	0.13	0.08	0.14	$0.03_{\pm 0.02}$	$0.04_{\pm 0.03}$

Table 3: Per prompting case (Hate and NotHate) results for the SINGLE method for **HSD**, stratified across high (top) and low (bottom) entropy items. For **proportionality** the values are the mean pairwise TVD between the normalized norm predictions and the predictions of the respective scope ($n = 100$/1000/random); **calibration** values are the mean pairwise TVD between trials for the norm setting (standard deviation in the subscript).

Model	Proportionality			Calibration
	$n = 100$	$n = 1000$	$n = random$	
claude-haiku-4.5	0.04	0.05	0.07	$0.00_{\pm 0.00}$
claude-opus-4.7	0.04	0.05	0.04	$0.02_{\pm 0.02}$
qwen3.5-9b	0.11	0.10	0.17	$0.02_{\pm 0.02}$
qwen3.5-397b	0.06	0.06	0.06	$0.02_{\pm 0.02}$
gemini-3.1-pro	0.06	0.08	0.07	$0.04_{\pm 0.03}$
glm-5.1	0.04	0.06	0.10	$0.03_{\pm 0.02}$
claude-haiku-4.5	0.03	0.03	0.04	$0.00_{\pm 0.00}$
claude-opus-4.7	0.03	0.04	0.05	$0.02_{\pm 0.02}$
qwen3.5-9b	0.08	0.06	0.08	$0.01_{\pm 0.01}$
qwen3.5-397b	0.03	0.04	0.04	$0.01_{\pm 0.01}$
gemini-3.1-pro	0.03	0.02	0.03	$0.03_{\pm 0.02}$
glm-5.1	0.03	0.04	0.06	$0.02_{\pm 0.02}$

Table 4: Results for the FULL method for **HSD**, stratified across high (top) and low (bottom) entropy items. For **well-formedness**, the values represent the standard and absolute error for the random setting; for **proportionality** the values are the mean pairwise TVD between the normalized norm predictions and the predictions of the respective scope ($n = 100$/1000/random); **calibration** values are the mean pairwise TVD between trials for the norm setting (standard deviation in the subscript).

Model	HSD		NLI
	SINGLE	FULL	FULL
claude-haiku-4.5	0.298	0.300	0.326
claude-opus-4.7	0.196	0.222	0.344
qwen3.5-9b	0.288	0.358	0.400
qwen3.5-397B	0.266	0.353	0.404
gemini-3.1-pro	0.355	0.308	0.520
glm-5.1	0.259	0.299	0.480
claude-haiku-4.5	0.185	0.195	0.126
claude-opus-4.7	0.223	0.220	0.116
qwen3.5-9b	0.202	0.193	0.105
qwen3.5-397B	0.218	0.182	0.108
gemini-3.1-pro	0.118	0.118	0.123
glm-5.1	0.237	0.239	0.108

Table 5: Per-model mean TVD between normalized norm predictions and the human distribution on HSD, split by high (top) and low (bottom) entropy.

Model	$\mathcal{E}$	$\mathcal{E}_{sign}$
claude-haiku-4.5	0.09	-0.03
claude-opus-4.7	0.14	-0.13
qwen3.5-9b	0.32	0.15
qwen3.5-397b	0.26	0.05
gemini-3.1-pro	0.14	-0.11
glm-5.1	0.10	0.00
claude-haiku-4.5	0.05	-0.02
claude-opus-4.7	0.10	-0.09
qwen3.5-9b	0.25	0.15
qwen3.5-397b	0.15	0.00
gemini-3.1-pro	0.06	-0.04
glm-5.1	0.07	0.00

Table 6: Well-formedness error rates (see Well-formedness) for **HSD**, across high (top) and low (bottom) entropy items. Values show error rates and signed error rates.

Apart from gemini-3.1-pro, which required the models to output a reasoning string before generating a prediction, the output strings were kept to a minimum number of output tokens consisting of the desired output format (e.g., '[0.3]' for SINGLE or '{"Hate": 0.3, "NotHate": 0.7}' for FULL).

A.5 Signed Error Rates

In Figure 2, we report the absolute average error rates $\mathcal{E}$ per model. A more fine-grained analysis of the signed average error rates $\mathcal{E}_{sign}$ in Table 6 gives further insights into each model's prediction consistency. For $\mathcal{E}_{sign}$, $\mathcal{E}_{sign} > 0$ signifies that the model has overestimated the probability distribution, whereas $\mathcal{E}_{sign} < 0$ corresponds to an underestimation.

All $\mathcal{E}_{sign}$ are amplified in the high entropy setting. For FULL, a large part of the models have errors close to 0.00, while the SINGLE setting has a larger variance between models. Most models maintain near-zero mean $\mathcal{E}_{sign}$, but claude-opus-4.7 and gemini-3.1-pro notably undershoot (-0.12), and qwen3.5-9b overshoots (0.15) the predicted distribution.

A.6 Prompts

Below are the prompt templates used for evaluating the annotator distributions for HSD for the SINGLE and FULL prompt.

The SINGLE prompt was adapted from Ni et al. (2026), where, in the case of hate-speech detection, [label] is replaced with 'Hate' and 'NotHate'.

Prompt: HSD - norm - SINGLE

```
You are a helpful research assistant with
     expertise in detecting hate speech
    from social media posts.

Objective: You are tasked with evaluating
     social media posts to assess the
    distribution of labels assigned by
    annotators regarding whether a post
    constitutes hate speech. The goal is
    to predict the percentage of
    annotators who classify the post as
    hate speech and those who classify it
     as not hate speech, based on the
    specific aspects of hate speech it
    may exhibit.

A post contains hate speech if it
    contains any of the following aspects
    :
- Assaults on Human Dignity: Does the
    post demean or degrade individuals or
     groups based on race, ethnicity,
    gender, religion, sexual orientation,
     or other protected characteristics?
- Calls for Violence: Does the post
    incite or encourage physical harm or
    violence against individuals or
    groups?
- Vulgarity and/or Offensive Language:
    Does the post contain profanity,
    slurs, or other offensive language
    that may or may not be directed at
```

```
    individuals or groups?

Instruction:
- Review the Post: Carefully read the
    social media post and consider its
    content in relation to the three
    aspects of hate speech listed above.
- Assess Agreement: Human annotators may
    have high disagreement if the post
    contains ambiguous language or
    context, or if it is likely to be
    interpreted differently based on
    subjective views about the severity
    or nature of hate speech. They may
    have low disagreement if the post
    clearly falls into one or more of the
     hate speech categories or is
    obviously not hate speech.
- Context Sensitivity: Consider the
    cultural, social, and linguistic
    context that may influence the
    interpretation of the post.

Target label for this run: {label}.

Output Format: [Percentage]: a floating-
    point number indicating the
    percentage of annotators for the
    target label, in squared brackets, e.
    g., [0.3]. Omit explanations and only
     give the percentage.

Here is the post: {text}

{label}:
```

Figure 3: Prompt for HSD, SINGLE in the norm setting.

Prompt: HSD - norm - FULL

```
...

Output Format: Return ONLY one valid JSON
     object with exactly two keys in this
     order: {``Hate'': <float>, ``NotHate
    '': <float>}. Each value must be a
    floating-point number between 0 and
    1, and the two values should sum to
    1. Omit explanations and only give
    the JSON object.

Here is the post: {text}
```

Figure 4: Section of the prompt for HSD, FULL in the norm setting.

Prompt: HSD - count - FULL

```
...

The goal is to predict the distribution
    of {count} annotators who classify
    the post as hate speech and those who
     classify it as not hate speech,
...


Output Format: Return ONLY one valid JSON
     object with exactly two keys in this
     order: {{``Hate'': <int>, ``NotHate
    '': <int>}}.
Each value must be an integer between 0
    and {count}, and the two values
    should sum to {count}.
Omit explanations and only give the JSON
    object.


Here is the post: {text}
```

Figure 5: Section of the prompt for HSD FULL using $n = 100$, $n = 1000$ and $n = random$, where count is replaced by the corresponding annotator participants.

Prompt: NLI - norm - FULL

```
Instructions:

You will be given a Context and a
    Statement. A group of annotators were
     asked to label the relationship
    between them using the following
    criteria:

Assuming the Context is true, the
    Statement...
* ... is most likely true  entailment
* ... could be either true or false
    neutral
* ... is most likely false  contradiction

Your task is to estimate how those
    annotators distributed their labels
    across the three categories.

Output a single Python dictionary with
    exactly these three keys: ``
    entailment'', ``neutral'', ``
    contradiction''. The values must be
    non-negative floats that sum to
    exactly 1.0.

Example output format:

{example}

IMPORTANT:
* Output only the Python dictionary  no
```

```
    explanation, no markdown, no extra
    text.
* Once again, do not preface your output
    with ```json, as the output will be
    parsed directly with Python's `json.
    loads`.
"""

CHAOSNLI_USER_PROMPT_DIST = """
[CONTEXT]: {context}
[STATEMENT]: {statement}

Response distribution:
```

Figure 6: Prompt for NLI, FULL in the norm setting.