How Many Ratings per Item are Necessary for Reliable Significance Testing?

Christopher M. Homan[1], Flip Korn[2], Deepak Pandita[1], Chris Welty[2]

[1]Rochester Institute of Technology, [2]Google Research

cmh@cs.rit.edu, flip@google.com, deepak@mail.rit.edu, cawelty@gmail.com

Abstract

A cornerstone of machine learning evaluation is the (often hidden) assumption that model and human responses are reliable enough to evaluate models against unitary, authoritative, "gold standard" data, via simple metrics such as accuracy, precision, and recall. The generative AI revolution would seem to explode this assumption, given the critical role stochastic inference plays. Yet, in spite of public demand for more transparency in AI—along with strong evidence that humans are unreliable judges—estimates of model reliability are conventionally based on, at most, a few output responses per input item. We adapt a method, previously used to evaluate the reliability of various metrics and estimators for machine learning evaluation, to determine whether an (existing or planned) dataset has enough responses per item to assure reliable null hypothesis statistical testing. We show that, for many common metrics, collecting even 5-10 responses per item (from each model and team of human evaluators) is not sufficient. We apply our methods to several of the very few extant gold standard test sets with multiple disaggregated responses per item and show that even these datasets lack enough responses per item. We show how our methods can help AI researchers make better decisions about how to collect data for AI evaluation.

1 Introduction

Arguably, the two central questions of experimental design are: *What degree of detection capability must the study possess to ensure that a genuine effect, if present, is measured?* and *How reliably can we predict the same outcome in future trials, given the observed evidence?* Here, **power analysis (PA)** (Bausell and Li, 2002) helps to answer the first question by controlling for false-negatives, and **null hypothesis statistical tests** (NHSTs) – or, in some cases, confidence intervals (CIs) – address the second by controlling for false-positives.

For AI evaluation, nearly all existing implementations of these fundamental tools for capturing experimental reproducibility measure only the variation of the inputs. *Yet they fail to capture the variance of the* ***output responses****—model or human—associated with each test input item.*

On the model side, response variance can come from stochastic inference, which is responsible for the creative power of foundation models, such as LLMs. It can also come from race conditions (Shanmugavelu et al., 2024), mixtures of experts (Shazeer et al., 2017), Monte Carlo dropout (Gal and Ghahramani, 2016), and ensembling (Lakshminarayanan et al., 2017).

On the human side, annotation and feedback continue to play a critical role in making AI useful, by providing gold standard responses. The increasingly sophisticated behavior of AI models has made it easier for people with little-to-no computer training to interact with them (Daugherty and Wilson, 2018).

In this paper, we present a humans-in-the-loop method for estimating the number of test items N, *and responses per item* K, needed for *reproducibly* estimating the performance difference between two AI models, while accounting for sampling variance across both items and responses per item, *before more data are collected and models are retrained*. This gives critical information about how to budget resources for building benchmark datasets. Our approach, which builds on methods from Wein et al. (2023), simulates the responses from a large pool of human raters and two ML models, rather than relying on methods that aggregate, and hence ignore, response variance. Simulation enables us to generate enough response data to explore the significance boundary for NHST under various metrics, for N test examples (items) with K responses per item for each model and pool of human raters. Our contributions are as follows:

Findings of the Association for Computational Linguistics: EACL 2026, pages 4258–4273

- While Wein et al. (2023) used their simulator to answer the questions of what are the best metrics and bootstrap configurations to use, they did not investigate the optimal trade-off in annotator budget between number of items N and raters per item K.

- The simulator of Wein et al. (2023) can only estimate p-values for NHST. We extend the simulator to also estimate the type-II error rate, allowing for statistical power.

- We examine the trade-off between N versus K and report these results on seven real-world datasets; by contrast, Wein et al. (2023) only investigated p-value estimation using a single dataset. We show that these datasets, in their current size, lack enough responses for reproducibility of model performance. We further show that one can boost the reproducibility with fewer overall responses by collecting *fewer items with more responses per item*. In fact, our results in Section 5.3 indicate that, for a fixed budget of $N \times K$ overall responses, apportioning the budget to as many as 100 responses per item can provide more reproducibility than with fewer responses per item.

2 Related Work

Statistical testing is critical to understanding state-of-the-art performance on a task or within a domain, in particular due to the **flawed nature of benchmarking practices** in machine learning evaluation (Ethayarajh and Jurafsky, 2020; Raji et al., 2021; Rodriguez et al., 2021; Hernandez-Orallo, 2020). Existing statistical tests such as Student's t-test (Student, 1908) are based on strong assumptions, such as that the datasets are normally distributed or have the same standard deviation, which are not realistic, especially when testing the system on new datasets (Søgaard, 2013). Dietterich (1998) applied hypothesis testing to machine learning systems and Dror et al. (2020); Deutsch et al. (2021) provide a survey and guide to state-of-the-art techniques for statistical significance testing in AI systems. Longjohn et al. (2025) study the problem of aggregating across multiple tests. All of these studies apply to the case where each model yields a single response and a single correct label exists for each training example; therefore, the issue of response variance is ignored.

More recently, Gundersen (2020) exploited pseudo-random seeds to generate multiple model responses that could be used for improved statistical testing in the presence of a single correct label for each item. Goldberg et al. (2018) showed how to revise p-value calculation when "gold" annotations exist but are unknown and in their place multiple noisy "bronze" annotations are available, where the probability of a bronze annotation matching the gold is given. In contrast, we consider settings where annotations are subjective and, hence, there is no single right answer but rather the ground truth is a distribution.

Our approach incorporates response variance from both ML models and human raters. The nature of response variance of the former was studied in Szymański and Gorman (2020), claiming that human rater response variance on individual items is most often due to measurable differences in perspective or ambiguity of the item, as opposed to noise. Nuanced analysis of the nature of response variance in ML has been studied by R Artstein (2008); Plank et al. (2014); Peng et al. (2024); Weerasooriya et al. (2023). See Plank (2022) for a survey.

Although none of these methods have been widely adopted, beginning with Dawid and Skene (1979), researchers have recognized the importance of response variance, and have sought to characterize it. Most of these methods can be characterized as **tableau**-based, where items are visualized as rows and respondents as columns of an (often sparse) table (Passonneau and Carpenter, 2014), and the models typically seek to jointly model both dimensions.

Lalor et al. (2016) apply item-response theory (IRT) to ML datasets. IRT is widely used in survey design and educational testing, two domains where, ironically, variance among respondents is widely reported, but variance among the items is not. (This makes sense for survey design, where each question addresses a different problem, but not in educational tests that contain multiple instances of the same problem, such as the Scholastic Aptitude Test (SAT).) And so they present a mirror image to the case of ML where, generally speaking, people tend to analyze variance along one dimension of the tableau, regardless of the domain, although *which* dimension is used depends on the domain.

Related crowdsourcing studies have examined the trade-offs between cost and quality of annotation collection (Snow et al., 2008) or gave recommendations for which crowdsourcing platforms

and protocols to use (Wang et al., 2013). Chau et al. (2020) explored the use of peer-review and self-review to resolve disagreement in annotations, and Hovy et al. (2013) developed an unsupervised model to identify which Mechanical Turk raters are reliable. Recent assessments of leaderboard practices have also led to models being able to indicate which items are most useful to annotate for evaluation purposes (Rodriguez et al., 2021). Welinder and Perona (2010) developed a system to select the most useful/informative labels to collect, which can lead to a reduction in annotation cost.

Sheng et al. (2008) focus on ML data curation and examines when one should obtain multiple, noisy training labels to improve model accuracy, assuming there exists a single correct label for each example. Lin et al. (2014) claim that response variance is less important than item variance – at least for training data – and suggests collecting more items with a single response is more valuable than collecting multiple responses per item.

Wein et al. (2023) investigate p-value sensitivity of both metrics and test-set sampling methods in hypothesis testing, which therefore can affect the power analysis. While the latter did not turn out to be important in our study, metrics did. Clearly, different metrics (e.g., mean absolute error vs Spearman rank-correlation) will produce different scores for the same matrix of responses, so it stands to reason that any comparison will have different p-values for different metrics. They model a metric as a function $\Gamma(M, G)$, where M is a matrix of model predictions which returns a score for M. We assume Γ is given here but focus on the best performing of these metrics in experiments. Homan et al. (2024) initiates a study of the trade-off between number of items and responses using a toy simulator. By contrast, we use real datasets to investigate these trade-offs and perform experiments that shed light on the mechanism for how response variance provides statistical significance. Recently, in a follow-up paper (Pandita et al., 2025) to this work, we modeled categorical datasets using a Bayesian approach and examined the optimization problem of allocating a fixed human annotation budget ($N \times K$). The current work provides the foundational evidence, for regression models, that increasing responses per item (K) is often more critical for significance than increasing the number of items (N).

The term *multistage sampling* is commonly used in statistics when the data is subsampled at multiple levels of granularity, usually for stratification. Bootstrap resampling has been applied in this setting (Mashreghi et al., 2016) and so the sampling method we describe herein can be seen as an instance of these. The Pigeonhole Bootstrap (Owen, 2007) is quite different from our multistage bootstrapping in that it resamples independently over rows and columns to form a Cartesian product rather than being nested.

It would be remiss not to mention other classes of techniques besides hypothesis testing that are commonly used for measuring statistical differences in model performance; see Riezler and Hagmann (2021) for a survey. *Likelihood ratios* provide an alternative form of significance testing and have been used for evaluating the impact of variability in data characteristics and hyperparameter settings on ML models (Hagmann et al., 2023). Estimation statistics for reliability, most notably *confidence intervals*, take variance into account to produce a range of values and are often used to assess a difference in model performance via non-overlap. *Circularity testing* based on general additive models has been proposed for evaluating the validity of ML models (Riezler and Hagmann, 2021).

3 Problem Statement

We wish to apply null hypothesis significance testing (NHST) to compare the performance of two ML models, A and B, on a test set of N items with K responses per item and decide if one model is significantly better than the other. We evaluate this with respect to human-annotated benchmark "gold" responses, G, and according to a metric, Γ, which we assume is provided as a design hyperparameter. For example, a common metric for evaluating regression models is the mean absolute error (differences) between model predictions and gold annotations.

The null hypothesis assumes that the respective model output distributions are the same in relation to G. Our goal is to determine whether the observations would be less than 5% likely under the null hypothesis and, therefore, the null hypothesis can be rejected. The 5% level is what our calculated p-values are compared against to conclude significance. Our motivation here is to determine whether a dataset—which we represent as $G^{N \times K}$, a matrix of N items and K responses—is large enough to provide replicable test results. This can be applied either post-hoc, as a test of the reliability of results,

or at design time, before data is gathered and to help determine how best to allocate the usually limited amount of resources available for gathering human annotations.

A key innovation in this work is to treat a data set $G^{N\times K}$ (as well as the responses from models A and B) as a matrix of responses, instead of the pervasive simplifying assumption that G is a vector, whose value for each item is an aggregation, such as the mean of several independent annotator (or model) responses. The notation captures the further insight that the distribution of responses for each item in a dataset is different.

4 Methods

Our main contribution is a human-in-the-loop process that allows one to (1) estimate the amount of data in terms of items, N, and responses per item, K, needed to detect, with high confidence, a difference of performance according to metric Γ of at least ϵ; and (2) compute p-values for existing experimental data comparing the performance of two models against gold data. Note that when the amount of experimental data is insufficient we can fit the data to a parameterized model and perform (1) to rerun the experiments with a sufficiently large dataset. It is precisely this use case that our experiments address.

Given an evaluation dataset G, arbitrary N and K, $\epsilon > 0$ and metric Γ the process has the following steps.

1. Fit a two-stage probabilistic *response model* model to G.
2. Use that model via *simulation* to determine p-values for N, K, ϵ, and Γ.

To fit a dataset to a response model, we create two histograms, one of all the individual responses over as a flat distribution and another of the average ratings of each item. We then find distribution families whose members visually match the distributions. Finally, we use the scipy package to find optimal parameters for the chosen model families fitting the dataset. See Section 5.1 and the appendix for more details.

We then use a simulator to generate new gold responses as the same (fitted) distribution as G. We use the same given distribution to generate data for both A and G, so that A represents an ideal model for G. We add perturbation (governed by ϵ) to this distribution to generate data for B. This ensures that model A performs better than model B with respect to G under almost any metric, and that "ground-truth" p-values should converge to zero as ϵ, N, and/or K increase. The simulator then estimates p-values (or, in the case of power analysis $1-\beta$) based on a large number of repetitions b. Typically $b = 10000$, although power analysis requires two levels of repetitions: one to generate a distribution over effect sizes and one to estimate the p-value, given the effect size. We report the number of repetitions in the figures associated with each of our results (Figure 4).

The time complexity of computing p-values in terms of the number of calls to the metric function Γ is simply $bT(\Gamma)$, where T measures the time complexity of Γ. For most of the choices of Γ that we consider here, including MAE and Wins (see below), $T(\Gamma)$ is linear in the size of the matrix, hence the total complexity is $O(bNK)$.

5 Experiments

5.1 Data

Unfortunately, precious few public datasets have both a large number of items and disaggregated responses. We apply the metrics and p-value estimators to the following datasets, all of which are secondary to us. We essentially ignore the content of each item in each dataset and use only the human responses associated with each item. Even though these responses were generated by humans–and we believe modeling human annotators is a promising direction to explore—to simplify our analysis and minimize risk we ignore any information about those humans and treat the responses for each item as, effectively, an anonymous sample.

In the experiments, we use the data to fit parameterized models. This allows us to study the performance (counterfactually) of the metrics under different values of N, K than the ones inherent to datasets, and for different values of ϵ due to different (hypothetical) models. We need to rely on counterfactuals and hypotheticals, even though we have real data, because no extant dataset has enough responses for large enough N and K or models with specific ϵ for us to run our experiments, and collecting that data would be prohibitively expensive. In fact, the *motivation behind this research* is precisely the problem that we need to choose reasonable values for N and K *before* we collect data, because no one has the budget to collect data for arbitrary values of N or K.

The **MultiDomain Agreement** (Leonardelli et al., 2021) dataset contains tweets about Black Lives Matter, the US 2020 presidential election, and COVID-19, annotated for offensiveness. The test set has 3057 items annotated by 5 raters each. We fit the means and standard deviations of the item responses to *truncated* normal distributions with $(\mu = -0.5, \sigma = 1)$ and $(\mu = -0.3923, \sigma = 0.8502)$, respectively. Instructions for directly obtaining the dataset from the author are available at `https://github.com/dhfbk/annotators-agreement-dataset`.

The **Stanford Toxicity** dataset (Kumar et al., 2021) was also used in Wein et al. (2023). It contains 107,620 items annotated by 5 raters each with ratings on a 5-point Likert scale: not/slightly/moderately/very/extremely toxic. We use the same distributions as they do, namely, a folded normal with $(\mu = 0.19, \sigma = 0.11)$ for the means and a triangular distribution with $(a = -0.05, b = 0.21, c = 0.45)$ for the standard deviations. The data is available at `https://data.esrg.stanford.edu/study/toxicity-perspectives`. It is encrypted, but the website gives instructions for how to decrypt it. There is no published license.

5.2 Fitting the Simulator to Real Data

The simulator allows us to generate many test sets to extrapolate patterns beyond one domain or system. By holding the item distributions for A, B, and G fixed, we can draw from them repeatedly to generate test sets similar to a real dataset but with arbitrarily large values of N and K, which would be infeasible with actual human annotations.

Like Wein et al. (2023), for each set of responses (from models A or B, or G), we sample from multistage parameterized models to simulate multiple samples for fixed N and K from a data source. This multistage process uses two probabilistic models, where for each item i the second stage model generates responses for the item $P(i)$, while the first stage model generates for i parameters unique to i for the second stage model to generate each response (i.e., $P(j|i)$ for response j to item i). In contrast to Wein et al. (2023), we choose the parameterized models to fit real datasets. Each dataset has enough responses *over all items* for us visualize the *a priori* distribution (i.e., $P(j)$ for item j, without regard to the item i it is associated with), say, as a histogram and use that to make informed choices about what families of parameterized distribution might fit the data. However, none of these

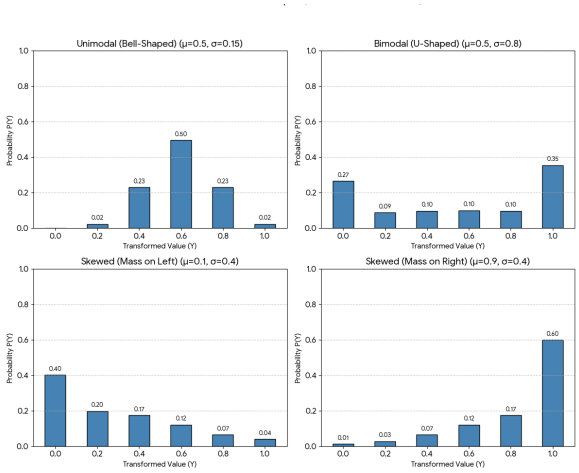


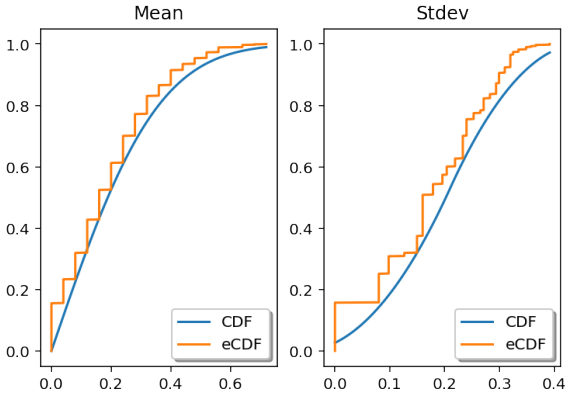
Figure 1: Shapes possible with censored normal.

datasets has enough responses *per item* for us to conclude anything about the shape of the *prior distribution* of responses for any item (we are not aware of any dataset that has both enough gold responses per item to visualize responses). And so for the second state model, we apply the principle of maximum entropy and assume the per-item distribution of responses is a *generalized normal distribution* $\mathcal{N}(\mu_i, \sigma_i)$. With more data per item, we could easily swap in a different family of distributions if we observed meaningful patterns in per-item responses.

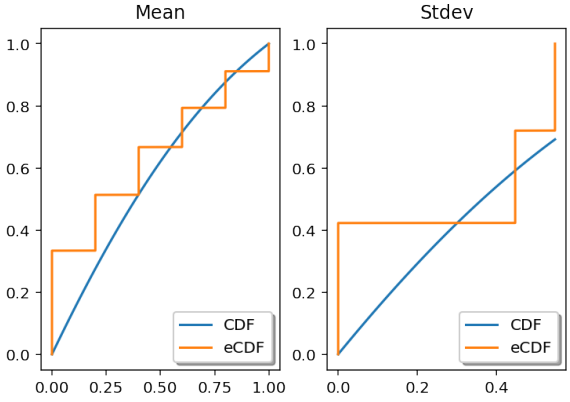
However, because we do have enough responses *over all items*, we do choose for the first stage specific distributions for each dataset that, paired with the second stage described above, fit the data. We used the *censored* normal distribution for $\mathcal{N}$, which assumes a latent continuous distribution that is not observed exactly but measured to within intervals, including left and right intervals which *pool* (not truncate) the smallest and largest values, respectively. This provides support for head and/or tail bias; Figure 1 illustrates a variety shapes that this distribution can capture. For example, items in the Stanford Toxicity dataset (see Section 5.1) rated at either extreme (either "not toxic" or "extremely toxic") tend to have more agreement among raters.

We use distributions fitted to each dataset from distribution families tailored to each dataset. Note that we chose the folded and truncated normal and triangular distributions for these datasets *based on visually matching histograms of the responses of each dataset*, as described in Section 4. We can use any family of distributions we like, i.e., *they need*

not be any flavor of normal distribution, as long as there are algorithmically feasible ways of fitting them to the data. Figure 2 illustrates goodness-of-fit for simulations of two datasets used in this paper. Details about computing p-values and results for additional datasets can be found in Appendix A and B.



(a) Stanford Toxicity



(b) MultiDomain Agreement

Figure 2: Empirical CDFs of item-level response means and standard deviations in (a) the Stanford Toxicity dataset vs clipped, folded normal CDF with $\langle \mu = 0.19, \sigma = 0.11 \rangle$ and clipped triangular distribution CDF with $\langle a = -0.05, b = 0.21, c = 0.45 \rangle$, respectively; and (b) the MultiDomain-Agreement dataset vs truncated normal CDF with $\langle \mu = -0.5, \sigma = 1 \rangle$ and truncated normal CDF with $\langle \mu = -0.3923, \sigma = 0.8502 \rangle$, respectively.

5.3 Results

We mainly used the following metrics in experiments:

- *Mean absolute error difference* (MAE). The distances (errors) from the per-item mean gold response to the model response averaged over the items: $\Gamma_{\text{MAE}}(A, B, G) = \frac{1}{N}\sum_i^N \left(\left| \frac{1}{K}\sum_j^K B_{ij} - \frac{1}{K}\sum_j^K G_{ij} \right| - \left| \frac{1}{K}\sum_j^K A_{ij} - \frac{1}{K}\sum_j^K G_{ij} \right| \right)$
- *Item-wise wins* (Wins). The fraction of items in the test set for which the absolute error of A is smaller than B: $\Gamma_{\text{Wins}}(A, B, G) = \sum_{i=1}^N \mathbf{1}_<(|\overline{A_i} - \overline{G_i}|, |\overline{B_i} - \overline{G_i}|)/N$
- *Mean EMD difference* (MEMD). The Earth mover's distance for each item between the system and the gold standard responses, and then take the mean of those item-wise EMDs: $\Gamma_{\text{MEMD}}(A, B, G) = \sum_{i=1}^N \left(\text{EMD}(B_i, G_i) - \text{EMD}(A_i, G_i)\right)/N$

We utilize the Variance Estimation Toolkit (VET)[1] to run our experiments. We used the Python libraries NumPy, Pandas, and SciPy, versions 2.2.3, 2.2.1, and 1.13.1, respectively. Our experiments took various times to run, with the longest experiments (producing any of the points in our figures) running approximately nine hours.

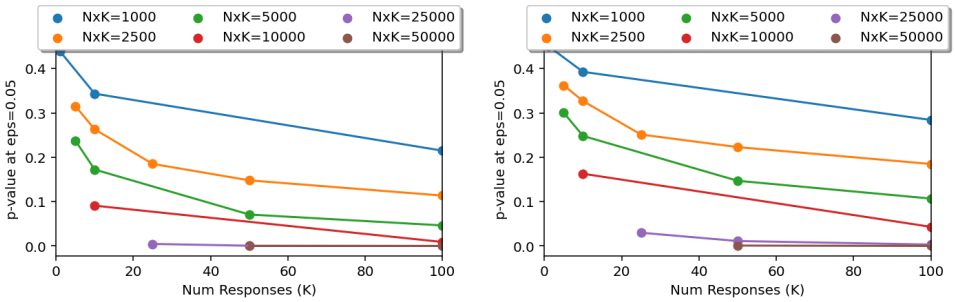
Figure 3 demonstrates that trading off items for responses is beneficial at a wide range of $(N \times K)$ values, with p-value decreasing as K increases. (The benefit of increasing K is strikingly more apparent when viewing p-values vs K with a fixed N, but we omit these graphs for brevity.) Here Γ_{MAE} was used with distortion $\epsilon = 0.05$ for Toxicity and $\epsilon = 0.1$ for MultiDomain, but similar trends were observed using other metrics, amounts of distortion, as well as different datasets. The graphs on the left are based on using the multistage bootstrap whereas those on the right use the baseline "flat" bootstrap over only the items, after the per-item responses have been aggregated. Note that the multistage bootstrap p-values are smaller, hence closer to the ground-truth values of zero, as it makes better use of response variance. There is indeed a point where trading N for K is beneficial for statistical significance: in this case, the curves hit an inflection point before $K = 500$; see Figure 4.

Figure 5 graphs p-value as a function of number of responses at $\epsilon = 0.1$, where the number of items varies such that $N \times K = 2500$, and demonstrates a similar trend across five different metrics.

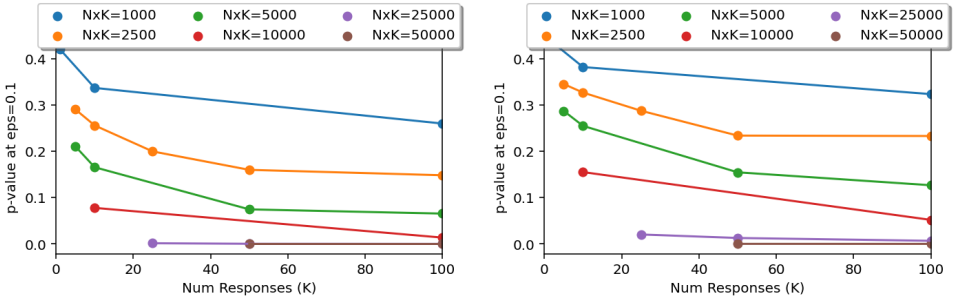
Power Analysis

Figure 6 demonstrate greater statistical power for Multistage Bootstrap as sample size with respect to either number of items or responses increases,

[1] https://github.com/google-research/vet



(a) Toxicity ($\epsilon = 0.05$): MultiStage (left) vs "flat" bootstrap (right)



(b) MultiDomain ($\epsilon = 0.1$): MultiStage (left) vs "flat" bootstrap (right)

Figure 3: p-value vs K with Γ_{MAE} at various $N \times K$. Each data point is the estimated from $10,000$ samples.

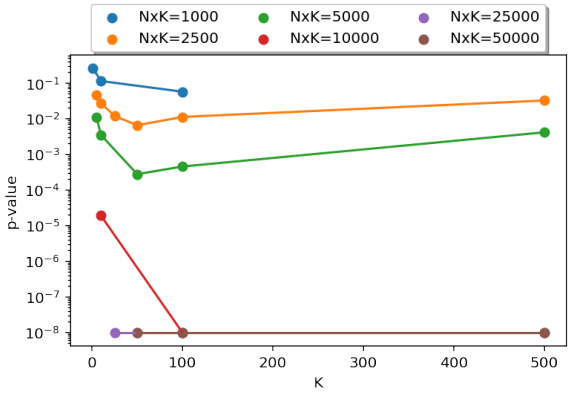


Figure 4: p-value vs K with Γ_{MAE} at various $N \times K$ for Toxicity at log-scale on the y-axis. Each data point is the estimated from 10000 samples.

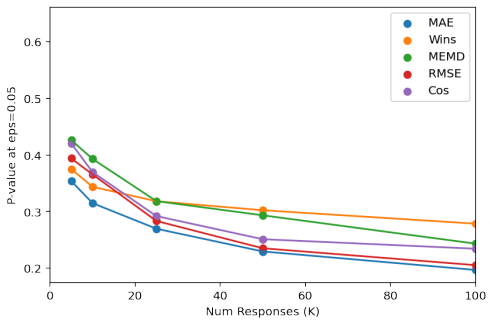
achieving a power of 90% (i.e., probability of not rejecting the null hypothesis when it's false) before baseline hypothesis tests. As usual, we use $\alpha = 0.05$ as the significance level for power calculation, i.e., the data is inconsistent with the null hypothesis at least 95% of the time. While the power of all these tests benefit from having more responses, the rate of improvement is markedly more rapid for Multistage Bootstrap.

For the baseline (paired) hypothesis tests, the mean response of each item was pre-computed for Model A, Model B and for "gold" G, resulting in $\bar{a}_i, \bar{b}_i, \bar{g}_i$, respectively, for each item i. The baseline tests then consider the null hypothesis that the distributions across the items of $|\bar{a}_i - \bar{g}_i|$ and $|\bar{b}_i - \bar{g}_i|$ are the same in the case of the permutation test, or have the same center in the case of Welch's t-test and the Wilcoxon signed-rank test. In contrast, Multistage Bootstrap resamples both the set of items and, for each item, the set of responses at each iteration, hence more effectively taking into account the disaggregated distribution of responses.

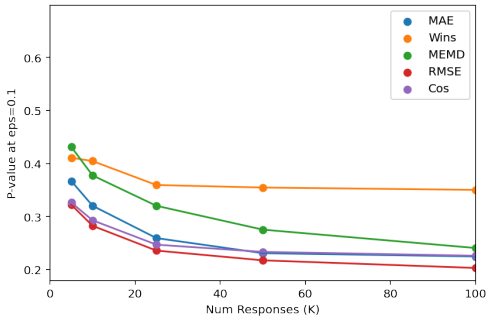
6 Discussion

Our results indicate that the number of raters and items have a notable impact on p-value estimation, to different degrees depending on the metric. Γ_{Wins} provides a discrete decision for each *item*, counting those decisions (i.e."wins") across the test set and normalizing by the number of items. Γ_{Wins} is also presented as a meta-metric of sorts: it can use any item-level metric, with absolute error being used here, and requires both models' predictions as well as input to directly compare their predictions at the item level.

In general, increasing N (number of test set items) increases the statistical power of any measurement by simply providing more scores to base the final metric score on. The more scores there are, the more stable the variance across simulation runs

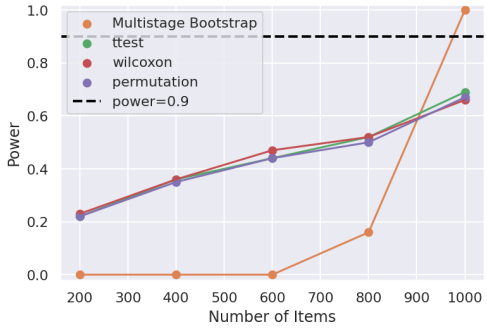


(a) Toxicity ($\epsilon = 0.05$)

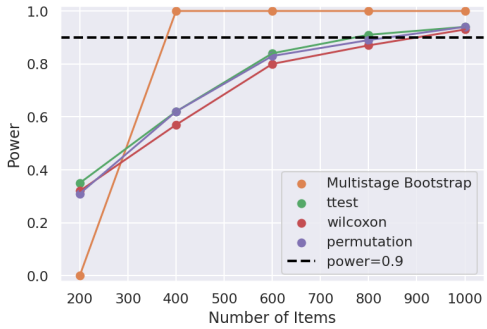


(b) MultiDomain ($\epsilon = 0.1$)

Figure 5: p-value vs K with a fixed budget $N \times K = 2500$ for various metrics. Each data point is estimated from $10,000$ samples.



(a) Varying N with $K = 5$



(b) Varying N with $K = 10$

(c) Varying K with $N = 1000$

Figure 6: Power Analysis of Toxicity data ($\epsilon = 0.1$). Each data point is the estimated from 1000 outer-level samples, each consisting of 10000 inner level samples.

will be, and the lower the p-value. All examined metrics respond well to increasing N.

Increasing K (number of responses per item) increases the statistical power of each *item level aggregate*. As K increases, the lower the variance of an individual item's aggregate will be across simulation runs, thereby lowering the p-value. All tested metrics also respond well to increasing K.

The difference between the metrics lies in the way the item-level scores are used. For Wins, which responds better to increasing N, the A's and B's item-level scores are directly compared. In each run, these item-level scores will vary, but in many cases that variance won't change the pairwise comparison. For example, if A_i's metric score is 0.10 and B_i's is 0.12 on the first simulation, a win is recorded for A. In the next simulation, if the scores are 0.11 and 0.12, respectively, this score change does not change the Win, as A_i's score is still lower. This indicates the item-level variance in the discrete win decision is far lower than the score variance - so adding more responses is less likely to further reduce the variance than adding items.

By contrast, for Γ_{MAE} and Γ_{MEMD}, any changes in item-level metric scores do impact the variance, both at the item and test-set level. Since the item-level scores come from the response distribution, adding more responses stabilizes the simulated distributions under repeated test set generation, reducing the metric variance across simulations and lowering the p-value.

The implications of these results are that the item/response trade-off should be handled differently depending on the metric itself, and the demands on the number of raters and items are high for all metrics in order to provide statistical guarantees. However, our results suggest that shifting the budget to account for as many as 100 raters per item could improve the sensitivity of experiment data to effect sizes.

The datasets we explored here have simple out-

put domains. What about generative models whose outputs may be highly complex? In this case, the ratings tend to be specific for each model, provided by a human or, increasingly, another generative model. And the ratings tend to be simple. In this case, our methods can be adapted. Essentially, we drop the gold dataset and only model the distribution of ratings received for each models *not the actual model responses*. This also requires a different comparision metric that does not require a gold dataset and instead of model responses takes as input the ratings each model received for its outputs. However, the basic process is the same.

7 Conclusion

In this work, we experimented with simulated data in order to examine the trade-off between the number of items and the number of responses per item necessary to compare two models against human judgments with statistical significance ($p < 0.05$). As expected, we see that when two models are more similar in performance, a greater number of annotations is required to achieve significance on their comparison. Further, the metric itself affects the utility of an increase in either items or responses.

These results suggest that current evaluation practices are not sufficient to confidently assess two models' performance against gold judgments, as using 25,000-50,000 annotations in a test set is rarely seen. Even when using 1000 items, at least 25 raters are needed for models to achieve significance with MAE.

Additionally, we found that the trade-off between the number of items and the number of responses per item depended on the metric. For two of our tested metrics, MAE and mean EMD, adding more responses than items is a more optimal division to achieve lower p-values. For the Wins metric, the opposite is true: more items and fewer responses per item lead to lower p-values. Still, in all cases for all metrics, increasing the total number of responses consistently lowers p-values, and thereby increases the sensitivity of the evaluation instrument. For real-world data, we actually found MAE to be more sensitive than Wins.

Limitations

The effectiveness of Wein et al. (2023)'s simulator depends on how well the probabilistic models capture realistic distributions of responses over items. Although we used rigorous methods to fit the parameters of these distributions to our datasets, our choice of distribution family to use for each dataset was based on visual inspection of the data. Given more datasets with disaggregated responses, we hope in future work to develop rigorous methods for model selection. However, the dearth of such publicly-available datasets impedes progress in this direction. One key limitation future work will address is that we treat the responses as independent from item-to-item, when in reality responses usually depend on which human annotator or instance of a model produced the response. Hypothesis testing such as that described here is not a comprehensive measure of data quality; it only estimates the likelihood of sampling error. It does not account for sampling bias, leading to data that is not representative of the sampling distribution.

The simulator is only intended to capture the complexity of the annotations. It is not intended to capture the complexity of real model predictions but rather to compare a near-perfect model, A, against a version, B, that has been perturbed by a controlled amount via a variance parameter. In practice, this functions as an approximate bound on the model response variance.

Otherwise, we have taken precautions to avoid common "p-hacking" pitfalls, such as that the null hypothesis and significance threshold α are independent of the dataset. We attempt to avoid *optional stopping* by performing power analysis.

While the distribution of responses depends on each item, we do not assume a fixed correspondence between annotations and raters. This assumption is valid, for example, with a large rating pool where each rater annotates at most one item. Therefore, there is no meaningful ordering of the responses within each item. For convenience, we use the term "matrix" for what is really a sequence of multisets. Modeling the dependence of annotations from the same raters across multiple items is something we chose to ignore in this paper so as not to distract from its main focus on the impact of response variance on hypothesis testing.

Ethical considerations

The paper focuses on a method to ensure that enough data is collected during testing to ensure that large enough observed differences between the performance of two models on the data are significant. While such analysis can ensure that experiment results are meaningful and replicable,

p-values have a tendency to be used more than they are understood. It is important to understand what p-values guarantee and what the limitations of our, or any other particular NHST framework, are. Misinterpreting the analysis can lead to dishonest or misleading claims about the reliability of the data for testing.

References

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2021. Whose opinions matter? perspective-aware models to identify opinions of hate speech victims in abusive language detection. *arXiv preprint arXiv:2106.15896*.

Dina Almanea and Massimo Poesio. 2022. ArMIS - the Arabic misogyny and sexism corpus with annotator subjective disagreements. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2282–2291, Marseille, France. European Language Resources Association.

R.B. Bausell and Y.F. Li. 2002. *Power Analysis for Experimental Research: A Practical Guide for the Biological, Medical and Social Sciences*. Cambridge University Press.

Amanda Cercas Curry, Gavin Abercrombie, and Verena Rieser. 2021. ConvAbuse: Data, analysis, and benchmarks for nuanced abuse detection in conversational AI. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7388–7403, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hung Chau, Saeid Balaneshin, Kai Liu, and Ondrej Linda. 2020. Understanding the tradeoff between cost and quality of expert annotations for keyphrase extraction. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 74–86, Barcelona, Spain. Association for Computational Linguistics.

Paul R Daugherty and H James Wilson. 2018. *Human+ machine: Reimagining work in the age of AI*. Harvard Business Press.

Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2021. A statistical analysis of summarization evaluation metrics using resampling methods. *Transactions of the Association for Computational Linguistics*, 9:1132–1146.

Thomas G Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.

Rotem Dror, Lotem Peled-Cohen, Segev Shlomov, and Roi Reichart. 2020. Statistical significance testing for natural language processing. *Synthesis Lectures on Human Language Technologies*, 13(2):1–116.

Kawin Ethayarajh and Dan Jurafsky. 2020. Utility is in the eye of the user: A critique of nlp leaderboards. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4846–4853.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR.

David Goldberg, Andrew Trotman, Xiao Wang, Wei Min, and Zongru Wan. 2018. Further insights on drawing sound conclusions from noisy judgments. *ACM Trans. Inf. Syst.*, 36(4).

Odd Erik Gundersen. 2020. The Reproducibility Crisis Is Real. *AI Magazine*, 41(3):103–106.

Michael Hagmann, Philipp Meier, and Stefan Riezler. 2023. Towards inferential reproducibility of machine learning research. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Jose Hernandez-Orallo. 2020. Ai evaluation: On broken yardsticks and measurement scales. In *Workshop on Evaluating Evaluation of Ai Systems at AAAI*.

Christopher M Homan, Shira Wein, Chris Welty, and Lora Aroyo. 2024. How many raters do you need? power analysis for foundation models. In *I Can't Believe It's Not Better Workshop: Failure Modes in the Age of Foundation Models*.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Deepak Kumar, Patrick Gage Kelley, Sunny Consolvo, Joshua Mason, Elie Bursztein, Zakir Durumeric, Kurt Thomas, and Michael Bailey. 2021. Designing toxic content classification for a diversity of perspectives. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*, pages 299–318. USENIX Association.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

John P. Lalor, Hao Wu, and Hong Yu. 2016. Building an evaluation scale using item response theory. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 648–657, Austin, Texas. Association for Computational Linguistics.

Elisa Leonardelli, Stefano Menini, Alessio Palmero Aprosio, Marco Guerini, and Sara Tonelli. 2021. Agreeing to disagree: Annotating offensive language datasets with annotators' disagreement. *arXiv preprint arXiv:2109.13563*.

Christopher H. Lin, Mausam, and Daniel S. Weld. 2014. To re(label), or not to re(label). In *HCOMP 2014*.

Rachel Longjohn, Giri Gopalan, and Emily Casleton. 2025. Statistical uncertainty quantification for aggregate performance metrics in machine learning benchmarks. *Preprint*, arXiv:2501.04234.

Zeinab Mashreghi, David Haziza, and Christian Léger. 2016. A survey of bootstrap methods in finite population sampling. *Statistics Surveys*, 10(none):1 – 52.

Art B. Owen. 2007. The pigeonhole bootstrap. *The Annals of Applied Statistics*, 1(2):386 – 411.

Deepak Pandita, Flip Korn, Chris Welty, and Christopher M Homan. 2025. Forest vs tree: The (n, k) trade-off in reproducible ml evaluation. *arXiv preprint arXiv:2508.03663*.

Rebecca J. Passonneau and Bob Carpenter. 2014. The benefits of a model of annotation. *Transactions of the Association for Computational Linguistics*, 2:311–326.

Siyao Peng, Zihang Sun, Sebastian Loftus, and Barbara Plank. 2024. Different tastes of entities: Investigating human label variation in named entity annotations. In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 73–81, Malta. 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.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 507–511, Baltimore, Maryland. Association for Computational Linguistics.

M Poesio R Artstein. 2008. Inter-coder agreement for computational linguistics. *Computational linguistics*, 34(4):555–596.

Inioluwa Deborah Raji, Emily M Bender, Amandalynne Paullada, Emily Denton, and Alex Hanna. 2021. Ai and the everything in the whole wide world benchmark.

Stefan Riezler and Michael Hagmann. 2021. *Validity, Reliability, and Significance: Empirical Methods for NLP and Data Science*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

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)*.

Pratik Sachdeva, Renata Barreto, Geoff Bacon, Alexander Sahn, Claudia von Vacano, and Chris Kennedy. 2022. The measuring hate speech corpus: Leveraging rasch measurement theory for data perspectivism. In *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP @LREC2022*, pages 83–94, Marseille, France. European Language Resources Association.

Sanjif Shanmugavelu, Mathieu Taillefumier, Christopher Culver, Oscar R. Hernandez, Mark Coletti, and Ada Sedova. 2024. Impacts of floating-point non-associativity on reproducibility for HPC and deep learning applications. *CoRR*, abs/2408.05148.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, page 614–622, New York, NY, USA. Association for Computing Machinery.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics.

Anders Søgaard. 2013. Estimating effect size across datasets. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 607–611, Atlanta, Georgia. Association for Computational Linguistics.

Student. 1908. The probable error of a mean. *Biometrika*, pages 1–25.

Piotr Szymański and Kyle Gorman. 2020. Is the best better? Bayesian statistical model comparison for natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2203–2212, Online. Association for Computational Linguistics.

Aobo Wang, Cong Duy Vu Hoang, and Min-Yen Kan. 2013. Perspectives on crowdsourcing annotations for natural language processing. *Language resources and evaluation*, 47:9–31.

Tharindu Cyril Weerasooriya, Sarah Luger, Saloni Poddar, Ashiqur KhudaBukhsh, and Christopher Homan. 2023. Subjective crowd disagreements for subjective data: Uncovering meaningful CrowdOpinion with population-level learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 950–966, Toronto, Canada. Association for Computational Linguistics.

Shira Wein, Christopher Homan, Lora Aroyo, and Chris Welty. 2023. Follow the leader(board) with confidence: Estimating p-values from a single test set with item and response variance. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3138–3161, Toronto, Canada. Association for Computational Linguistics.

Peter Welinder and Pietro Perona. 2010. Online crowdsourcing: rating annotators and obtaining cost-effective labels. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 25–32. IEEE.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

A Using the simulator to estimate p-values

A.1 Simulator

We use a simulator to generate "gold" annotations and model predictions by modeling the responses for each item as a random variable. The purpose of this is to be able to control how similar, or different, predictions from models A and B are to G as well as to each other. By using the same given distribution to generate data for both A and G, and by adding perturbation (governed by parameter ϵ) to the given distribution to generate data for B, we can ensure that model A performs better than model B with respect to G under almost any metric, and that "ground-truth" p-values should converge to zero as ϵ, N, and/or K increase.

The simulator takes input parameters N and K, along with *perturbation parameter* ϵ. In the first stage, it randomly chooses hyperparameters $\theta_1, \ldots, \theta_N \sim P_{items}$, each corresponding to an item θ_i, from a fixed distribution that serve as model parameters for the second stage. In the second stage, for each item i we sample K responses from a second distribution $P_{responses}(\theta_i)$. We do this for each of the datasets, respectively representing responses from gold annotations, $G^{N\times K}$, and two models, $A^{N\times K}$ and $B^{N\times K}$. The specific distributions that were used in our experiments were modeled from real datasets; for details see Appendix 5.2.

These choices operationalize a solution to the paradox that one must have data in G, A, and B to know if it has enough statistical power. Instead, we simulate a set of gold items and responses (G) and then simulate an ideal model (A) – ideal because it draws its simulated responses from the same distribution as the gold – and then explore how such an ideal system would compare in significance to another model (B) whose response distributions differ from gold by an amount (ϵ) we experimentally control. This gives us *a-priori* control over the hypothesis test, because we know which model is better through a controllable parameter.

For any given selection of N and K, we have response matrices $G^{N\times K}$ and $A^{N\times K}$ and, for each ϵ, a matrix $B^{N\times K,\epsilon}$. We then seek to compare A and B to each other to determine which is better; the answer should almost always be A unless $\epsilon = 0$. When evaluating AI models, the comparison of A and B involves differencing each of their item responses to those of G using a suitable metric, which is then aggregated across the items. We compare the performance between A and B via $\Gamma(A, B, G)$.

A.2 Estimating p-values

Given N, K, and ϵ, p-values are estimated by drawing b (bootstrap) resamples $S_{alt} = \langle G_1^{N\times K}, A_1^{N\times K}, B_{1,\epsilon}^{N\times K}\rangle, \ldots, \langle G_b^{N\times K}, A_b^{N\times K}, B_{b,\epsilon}^{N\times K}\rangle$ for the alternative hypothesis according to the process described in Section A.1. Since the null hypothesis makes the assumption that the distributions of A and B are the same with respect to G, we construct S_{null} by pooling the items from $A^{N\times K}$ and $B^{N\times K}$ and then independently sampling from this pool. When sampling responses from A,

for each item i, we sample each response by sampling from $P_{responses}(\theta_i)$, where $\theta_i = (\mu_i, \sigma_i)$. Sampling responses from B is similar but we first choose $\delta_i \sim Unif(-\epsilon, \epsilon)$ and then sample from $P_{responses}(\theta_i)$, where $\theta_i = (\mu_i + \delta_i, \sigma_i)$.

Next, we estimate the *expected p-value under the alternative hypothesis* as the average one-sided p-value over all samples in S_{alt}, computed by counting for each $s_{alt} = \langle G_{alt}^{N\times K}, A_{alt}^{N\times K}, B_{alt,\epsilon}^{N\times K}\rangle \in S_{alt}$ the fraction of samples $s_{null} \in S_{null}$ where $\Gamma(s_{null})$ is at least as extreme as $\Gamma(s_{alt})$. Here "at least as extreme" is determined by computing Γ_{alt} (respectively, Γ_{null}), the median of Γ over S_{alt} (respectively, S_{null}). If $\Gamma_{alt} > \Gamma_{null}$, then "at least as extreme" means $\Gamma(s_{null}) \geq \Gamma(s_{alt})$. Otherwise, it means $\Gamma(s_{null}) < \Gamma(s_{alt})$. The estimator is fast to compute if the Γ values are presorted, and because it is averaged over a large number of samples from the alternative hypothesis, it is a robust estimator for determining whether $N \times K$ is a large enough sample size.

Finally, as is typical for NHST, we reject the null hypothesis when the p-value is below the significance level $\alpha = 0.05$.

B Results on Additional Datasets

The **Amazon reviews** dataset (Zhang et al., 2015) contains 20,415 products rated by 5 reviewers on a scale of 1-5, which were selected from the full dataset of reviews from $6,643,669$ users on $2,441,053$ products from those products having at least 5 reviews. We fit the means and standard deviations of the item responses to *truncated* normal distributions with $(\mu = 0.552121, \sigma = 0.032093)$ and $(\mu = 0.318177, \sigma = 0.018281)$, respectively.

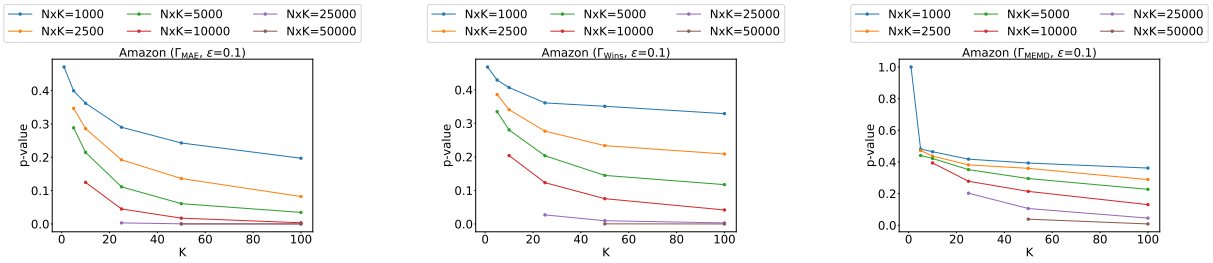
The **HS-Brexit** dataset (Akhtar et al., 2021) contains 1120 tweets related to Brexit and is labeled for hate speech by 6 raters each. We fit the means and standard deviations of the item responses to *truncated* normal distributions with $(\mu = -0.278260, \sigma = 0.181938)$ and $(\mu = -0.340141, \sigma = 0.408186)$, respectively.

The **ConvAbuse** dataset (Cercas Curry et al., 2021) contains 4185 dialogues between users and two conversational agents and is labeled for abuse by at least 3 experts each. We fit the means and standard deviations of the item responses to *truncated* normal distributions with $(\mu = 1.124694, \sigma = 0.512993)$ and $(\mu = -0.324344, \sigma = 0.417337)$, respectively.

The **ArMIS** dataset (Almanea and Poesio, 2022) contains 964 Arabic tweets for misogyny detection and is labeled by 3 raters each. We fit the means and standard deviations of the item responses to *truncated* bi-normal distributions with $(\mu_1 = -0.430701, \sigma_1 = 0.418148, \mu_2 = 1.194010, \sigma_1 = 0.525248)$ with the likelihood of choosing the first distribution as 0.652561 and $(\mu_1 = -0.264113, \sigma_1 = 0.530150, \mu_2 = 0.362404, \sigma_2 = 0.632262)$ with the likelihood of choosing the first distribution as 0.76639, respectively.

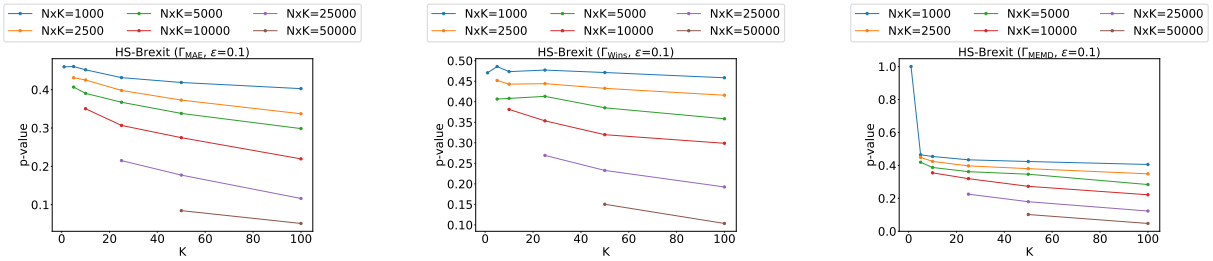
The **Measuring Hate Speech (MHS)** dataset (Sachdeva et al., 2022) contains 39,565 comments labeled for hate speech by 7912 raters. We fit the means and standard deviations of the item responses to *truncated* normal distributions with $(\mu = -0.211147, \sigma = 0.106442)$ and $(\mu = -0.243672, \sigma = 0.148406)$, respectively.

Tables 1–3 show the results for minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ for different ϵ. In Table 2 ($\epsilon = 0.1$), we observe that minimum p-values are consistently obtained with a higher $K = 100$ for all datasets except MHS, where minimum p-values are obtained at $K = \{5, 10\}$. We notice a similar trend for $\epsilon = 0.0.05$ and $\epsilon = 0.2$. Figures 7–11 show results for p-values for $\epsilon = 0.1$ for different datasets and metric combinations.



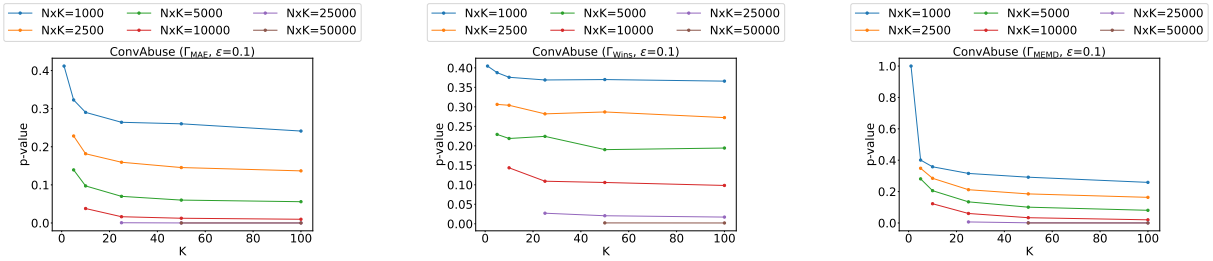
(a) MAE (b) Wins (c) MEMD

Figure 7: p-value plots for Amazon dataset, $\epsilon = 0.1$.



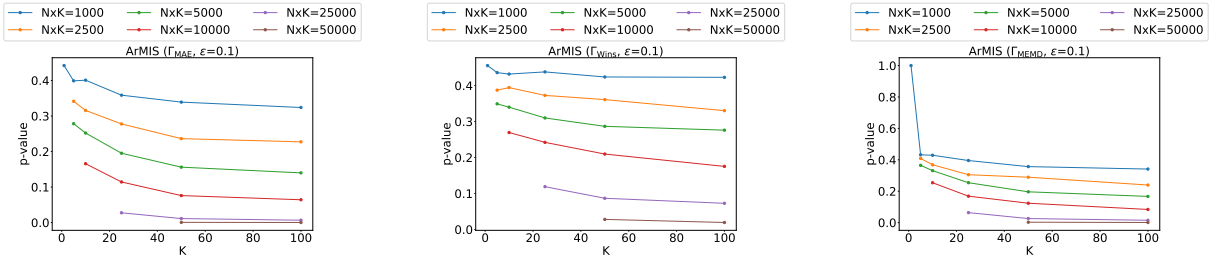
(a) MAE (b) Wins (c) MEMD

Figure 8: p-value plots for HS-Brexit dataset, $\epsilon = 0.1$.



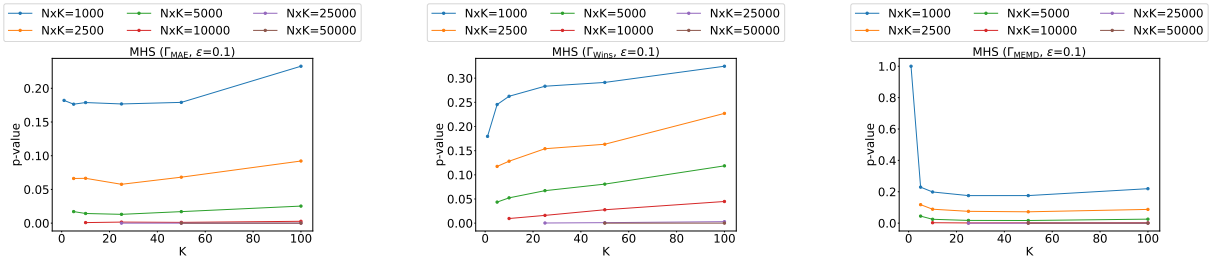
(a) MAE (b) Wins (c) MEMD

Figure 9: p-value plots for ConvAbuse dataset, $\epsilon = 0.1$.



(a) MAE (b) Wins (c) MEMD

Figure 10: p-value plots for ArMIS dataset, $\epsilon = 0.1$.



(a) MAE (b) Wins (c) MEMD

Figure 11: p-value plots for MHS dataset, $\epsilon = 0.1$.

Dataset	Stat	Γ_{MAE}	Γ_{Wins}	Γ_{MEMD}
Toxicity	NK	10000	25000	25000
	p-value	0.047	0.026	0.011
	K	100	100	100
	Δ	0.007	0.227	0.546
MultiDomain	NK	100000	-	100000
	p-value	0.019	-	0.033
	K	100	-	100
	Δ	0.006	-	0.195
Amazon	NK	40000	100000	-
	p-value	0.043	0.031	-
	K	100	100	-
	Δ	0.005	0.118	-
HS-Brexit	NK	-	-	-
	p-value	-	-	-
	K	-	-	-
	Δ	-	-	-
ConvAbuse	NK	40000	100000	40000
	p-value	0.020	0.037	0.033
	K	100	100	100
	Δ	0.009	0.093	0.279
ArMIS	NK	100000	-	100000
	p-value	0.025	-	0.047
	K	100	-	100
	Δ	0.005	-	0.169
MHS	NK	20000	40000	20000
	p-value	0.041	0.044	0.037
	K	100	100	100
	Δ	0.001	0.107	0.246

Table 1: Minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.05$).

Dataset	Stat	Γ_{MAE}	Γ_{Wins}	Γ_{MEMD}
Toxicity	NK	2000	5000	4000
	p-value	0.041	0.040	0.024
	K	100	100	100
	Δ	0.021	0.443	1.403
MultiDomain	NK	20000	40000	20000
	p-value	0.015	0.030	0.024
	K	100	100	100
	Δ	0.018	0.169	0.546
Amazon	NK	4000	10000	25000
	p-value	0.039	0.042	0.045
	K	100	100	100
	Δ	0.018	0.335	0.331
HS-Brexit	NK	100000	100000	50000
	p-value	0.009	0.037	0.047
	K	100	100	100
	Δ	0.004	0.077	0.153
ConvAbuse	NK	10000	20000	10000
	p-value	0.010	0.028	0.020
	K	100	100	100
	Δ	0.025	0.212	0.732
ArMIS	NK	20000	40000	20000
	p-value	0.018	0.035	0.024
	K	100	100	100
	Δ	0.016	0.158	0.491
MHS	NK	4000	5000	4000
	p-value	0.028	0.044	0.040
	K	10	5	10
	Δ	0.004	0.049	0.053

Table 2: Minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.1$).

Dataset	Stat	Γ_{MAE}	Γ_{Wins}	Γ_{MEMD}
Toxicity	NK	500	1250	1250
	p-value	0.048	0.045	0.046
	K	50	5	50
	Δ	0.045	0.192	1.056
MultiDomain	NK	2500	10000	4000
	p-value	0.042	0.026	0.029
	K	50	50	100
	Δ	0.042	0.246	1.390
Amazon	NK	1000	2500	4000
	p-value	0.024	0.041	0.018
	K	100	50	100
	Δ	0.053	0.479	1.220
HS-Brexit	NK	10000	20000	10000
	p-value	0.024	0.030	0.024
	K	100	50	100
	Δ	0.014	0.129	0.531
ConvAbuse	NK	2000	4000	2000
	p-value	0.023	0.037	0.044
	K	50	10	100
	Δ	0.057	0.127	1.659
ArMIS	NK	4000	10000	4000
	p-value	0.018	0.028	0.034
	K	100	25	100
	Δ	0.043	0.154	1.250
MHS	NK	200	200	1000
	p-value	0.038	0.039	0.016
	K	1	1	10
	Δ	0.020	0.097	0.185

Table 3: Minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.2$).