Forest vs Tree: The (N, K) Trade-off in Reproducible ML Evaluation

Deepak Pandita[1], **Flip Korn**[2], **Chris Welty**[2], **Christopher M. Homan**[1]
[1]Rochester Institute of Technology
[2]Google Research
deepak@mail.rit.edu, flip@google.com, cawelty@gmail.com, cmh@cs.rit.edu

Abstract

Reproducibility is a cornerstone of scientific validation and of the authority it confers on its results. Reproducibility in machine learning evaluations leads to greater trust, confidence, and value. However, the ground truth responses used in machine learning often necessarily come from humans, among whom disagreement is prevalent, and surprisingly little research has studied the impact of effectively ignoring disagreement in these responses, as is typically the case. One reason for the lack of research is that budgets for collecting human-annotated evaluation data are limited, and obtaining more samples from multiple raters for each example greatly increases the per-item annotation costs. We investigate the trade-off between the number of items (N) and the number of responses per item (K) needed for reliable machine learning evaluation. We analyze a diverse collection of categorical datasets for which multiple annotations per item exist, and simulated distributions fit to these datasets, to determine the optimal (N, K) configuration, given a fixed budget ($N \times K$), for collecting evaluation data and reliably comparing the performance of machine learning models. Our findings show, first, that accounting for human disagreement may come with $N \times K$ at no more than 1000 (and often much lower) for every dataset tested on at least one metric. Moreover, this minimal $N \times K$ almost always occurred for $K > 10$. Furthermore, the nature of the tradeoff between K and N, or if one even existed, depends on the evaluation metric, with metrics that are more sensitive to the full distribution of responses performing better at higher levels of K. Our methods can be used to help ML practitioners get more effective test data by finding the optimal metrics and number of items and annotations per item to collect to get the most reliability for their budget.

Code — https://github.com/google-research/vet
Extended version — https://arxiv.org/abs/2508.03663

Introduction

The scientific community, including the rapidly evolving fields of AI and NLP, is grappling with a pervasive reproducibility crisis (Baker 2016; Gundersen and Kjensmo 2018; Hutson 2018; Mieskes et al. 2019; Gundersen 2020). Researchers are increasingly unable to replicate the results of previous studies (Raff 2019), thus undermining trust in experimental and empirical research. In machine learning, where data-driven research is essential for advancing knowledge, the comparison of models is central to determining the state-of-the-art for a given task. As such, ensuring the reproducibility of results through robust evaluation is critical.

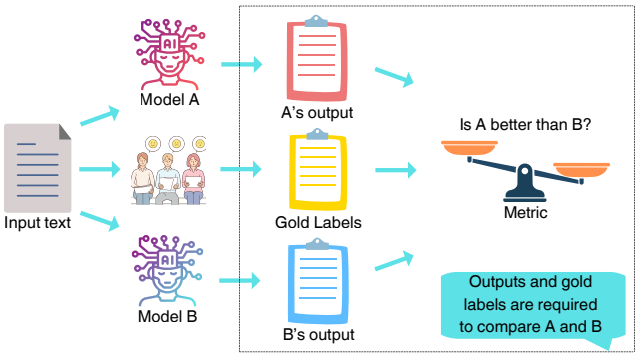


Figure 1: Model assessment process with ground truth.

We focus here on an underlooked source of unreliability: failing to account for human disagreement and other sources of randomness in ML evaluation. Conventional evaluation approaches treat disagreement, if at all, as nothing more than noise and may aggregate 3–5 labels per item—a number that comes from literature on machine learning (Snow et al. 2008), not machine learning evaluation—via plurality voting to represent consensus, overlooking disagreement, which is endemic in human responses, as anyone who has participated in a democratic process such as voting knows.

Recent papers have advocated using and publishing disaggregated labels to account for human label variation (Basile et al. 2021; Prabhakaran, Mostafazadeh Davani, and Diaz 2021; Plank 2022; Cabitza, Campagner, and Basile 2023). The field also faces a pervasive issue of inadequate statistical analysis; statistical significance is often misapplied, and reported outcomes are frequently unreliable (Søgaard et al. 2014; Dror et al. 2018; van der Lee et al. 2019).

A crucial question, from a statistical perspective, is *how much data needs to be collected to ensure statistically reliable testing*, via null hypothesis significance tests (NHSTs) and confidence intervals (CIs)? We are particularly interested in challenging the assumption (in a skeptical manner)

that a small number of annotations per item is sufficient. It would seem that there should be a trade-off between the number of items N, i.e., how many trees are observed, and the number of annotations per item collected, i.e., the resolution by which each tree is observed. It would seem that the nature of the trade-off might depend on the metrics used, which depend on the performance expectations of the models under consideration. In this same vein, we would like to know whether the newer, but still uncommon approach of keeping disaggregated responses for each item has value from the perspective of comparing one ML model against another—the most basic way to evaluate ML models. We investigate the following research questions:

RQ1 What is the lowest number of total annotations needed $N \times K$ to ensure reasonably repeatable results in comparing two models?

RQ2 How does this this number $N \times K$ depend on:

- the distribution of responses found in five actual datasets with disaggregated annotations?
- the metric used?
- the number of categories?
- the statistical instrument (NHSTs vs. CIs)?

RQ3 For fixed $N \times K$ (particularly the minimal ones found in RQ2), what is the smallest value of K that ensures reasonably repeatable results, and how does this vary according to the same variables in RQ2?

To address these questions, we make the following contributions.

1. We believe this is the first paper to examine the optimization problem of allocating a human annotation budget to a sample of N items, where each item is annotated by K raters, such that the total budget $N \times K$ is fixed.
2. How many items, and how many annotations per item, to collect needs to be known *before* the data is collected, but the answer should be based on realistic assumptions. Towards this end, we apply a Bayesian approach to model existing datasets via simulation for arbitrary N and K. This enables a more robust way of modeling when the sample size is small (assuming accurate priors) and allows for maximum *a posteriori* (MAP) fitting of data, versus maximum likelihood estimation (MLE)-based frequentist approaches, which provides regularization.
3. We extend an existing simulator to model categorical data and confidence intervals (along with NHSTs).
4. We report on a comprehensive set of experiments using five different real datasets as well as synthetic data, to demonstrate the impact on statistical significance and confidence of optimizing the trade-off between N and K.

Our findings show, first, that accounting for human disagreement may come with $N \times K$ at no more than 1000 (and often much lower) for every dataset tested on at least one metric. Moreover, this minimal $N \times K$ almost always occurred for $K > 10$. Moreover, the nature of the trade-off between K and N—or if one even existed—depends on the evaluation metric, with metrics that are more sensitive to the full distribution of responses performing better at higher levels of K.

Related Work

The reproducibility crisis in AI and NLP, highlighted by numerous studies (Gundersen and Kjensmo 2018; Hutson 2018; Mieskes et al. 2019; Gundersen 2020), stems from various factors. A major contributor is the inherent nondeterministic nature of machine learning methods, algorithms, and implementations; even with shared code, multiple seemingly identical training runs of the same deep learning model can yield different models and test results, often due to factors like varying random seeds or hardware-specific operations (Pham et al. 2020). Furthermore, a survey by Pham et al. (2020) of 901 participants revealed that 84% were either unaware or unsure about the variance stemming from different implementations. Arvan, Pina, and Parde (2022) further underscored this challenge by achieving only a 25% success rate in a reproducibility study of eight papers published in EMNLP 2021. These findings underscore the critical need to account for this inherent variance in machine learning evaluations, even when working with seemingly identical setups.

Beyond model-inherent variance, the human element in evaluation also introduces considerable variability. Human raters are frequently recruited to generate reference labels, commonly referred to as "gold standards," for evaluating machine learning model performance. However, human disagreement is prevalent, especially in subjective tasks, leading to significant variance in responses (Basile et al. 2021; Prabhakaran, Mostafazadeh Davani, and Diaz 2021; Uma et al. 2021; Plank 2022; Cabitza, Campagner, and Basile 2023; Homan et al. 2023a; Weerasooriya et al. 2023; Prabhakaran et al. 2023; Pandita et al. 2024). Typically, responses are aggregated via plurality voting to represent consensus, though recent work has shown the inadequacy of such aggregation for incorporating response variance (Barile et al. 2021; Mostafazadeh Davani, Díaz, and Prabhakaran 2022). It is therefore not surprising that human evaluation studies also show a low degree of reproducibility (Belz et al. 2023). The issue of response variance has been particularly explored in the context of conversational AI safety. For instance, Homan et al. (2023a) utilized Bayesian multilevel models to understand the impact of rater demographics on safety ratings, while Prabhakaran et al. (2023) proposed a framework to analyze diversity in safety ratings among rater subgroups. Further, Aroyo et al. (2023) introduced a dedicated dataset to enable in-depth analysis and measurement of response variance in this domain.

Wein et al. (2023) proposed a framework and simulator using NHSTs to estimate the true p-value of model comparisons. The simulator considers item and response variance to sample a "reference test set" of gold/human responses (G) and the responses of two models (A and B). To construct responses for G, it produces N items and K responses per item using random variables. For each item, the mean and standard deviation are sampled from uniform distributions. Then, K continuous responses are drawn using a normal distribution parameterized with the sampled mean and standard deviation. The responses for items in model A are sampled using the same distribution as G, making model A an ideal representation of G. Responses for items in model B are

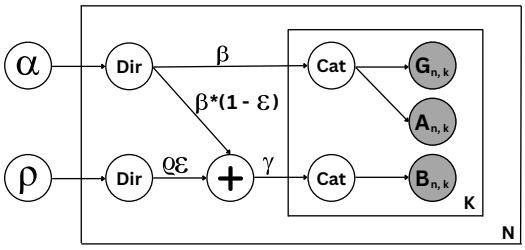


Figure 2: Plate notation for the simulator. Categorical parameters (β) and noise parameters (ϱ) are sampled from two Dirichlet distributions parameterized by α and ρ, respectively. Then, responses for G and A are produced by sampling from a categorical distribution parameterized by β. Responses for B are produced by sampling from a categorical distribution parameterized by γ, where γ is a convex combination of β and ϱ controlled by the perturbation parameter ϵ.

sampled using the same mean as G but with standard deviation perturbed by a small amount. Perturbation parameters for each item are randomly drawn at a given perturbation level from a uniform distribution. As the perturbation level increases, model B increasingly differs from model A. NHSTs are used to estimate the p-values for comparing model A and model B under different metrics and sampling methods. The data for the null hypothesis is generated assuming that responses for model A and model B are drawn from the same distribution. This is achieved by combining the responses from model A and model B.

Homan et al. (2023b) utilized the simulator presented by Wein et al. (2023) to propose an evaluation framework tailored for foundation models. Our approach builds on these prior works. We propose a framework for power analysis designed for evaluating machine learning models using NHSTs. Crucially, our framework accounts for both item and response variance, specifically under the assumption that the responses are nominal.

Methods

We use a simulator that generates the outputs for comparing two models, A and B, against gold standard outputs G (see Figure 1 for an illustration). Our methods differ from earlier work (Wein et al. 2023; Homan et al. 2023b) in that the responses produced by the simulator are categorical rather than continuous. The simulator produces the gold outputs G and the outputs for model A by sampling from a Dirichlet-categorical distribution, making A an ideal model. For model B, the annotations are sampled after introducing noise in the model parameters controlled by the perturbation parameter ϵ, making it slightly worse than model A. We then employ NHSTs to estimate the p-value for this comparison.

We use a Dirichlet-categorical distribution because it naturally models the probability of observing the categories, and the Dirichlet distribution is the conjugate prior for the categorical distribution, simplifying the calculations involved in Bayesian inference. When the prior is conjugate, the posterior distribution is also a Dirichlet distribution. Bayesian inference gives us the flexibility to incorporate prior information about response probabilities and produces more robust estimates, especially when data is limited.

Algorithm 1: Simulations for H_{alt}

Input parameters: $N, K, M, \alpha, \rho, \epsilon$
for $i = 1$ **to** N **do**
  // sample categorical parameters
  $\boldsymbol{\beta_i} = \beta_{i,1}, ..., \beta_{i,M} \sim Dir(\alpha_1, ..., \alpha_M)$;
  // sample noise parameters
  $\boldsymbol{\varrho_i} = \varrho_{i,1}, ..., \varrho_{i,M} \sim Dir(\rho_1, ..., \rho_M)$;
  // convex combination of categorical & noise parameters
  $\boldsymbol{\gamma_i} = (1 - \epsilon) * \boldsymbol{\beta_i} + \epsilon * \boldsymbol{\varrho_i}$;
  /* sample j's response to i */
  // Gold
  for $j = 1$ **to** k **do**
    $G_{i,j} = Cat(\beta_{i,1}, ..., \beta_{i,M})$;
  // Model A
  for $j = 1$ **to** K **do**
    $A_{i,j} = Cat(\beta_{i,1}, ..., \beta_{i,M})$;
  // Model B
  for $j = 1$ **to** K **do**
    $B_{i,j} = Cat(\gamma_{i,1}, ..., \gamma_{i,M})$;

Simulation Framework

We generate the gold outputs G and the outputs for two models A and B. Each sample consists of N items and K responses per item. The responses are discrete values chosen from M categories. For each item $i \in N$, we sample categorical parameters (β_i) and noise parameters (ϱ_i) from two Dirichlet distributions parameterized by α and ρ, respectively. For G and A, the response by each rater $j \in K$ is sampled from a categorical distribution parameterized by β. For model B, the response by each rater j is sampled from a categorical distribution parameterized by γ, where γ is a convex combination of β and ϱ controlled by the perturbation parameter ϵ. This process is illustrated in Figure 2 and is described in Algorithm 1.

Hypothesis Testing

For the alternative hypothesis (H_{alt}), we use the responses generated according to Algorithm 1. For the null hypothesis (H_{null}), we generate data for two models A and B, assuming they are drawn from the same distribution. The process is similar to the one in H_{alt} except the response by each rater is sampled from a categorical distribution with parameters chosen uniformly at random from $\{\beta_i, \gamma_i\}$ for both models A and B. This process is described in Algorithm 2.

To compare H_{alt} and H_{null}, we use a metric $\Gamma(A, B, G)$ for each pair of response samples $\{A, B\}$ and gold samples G to obtain a score. $\Gamma(A, B, G) = \Gamma(A, G) - \Gamma(B, G)$,

Algorithm 2: Simulations for H_{null}

Input parameters: $N, K, M, \alpha, \rho, \epsilon$
for $i = 1$ **to** N **do**
  // Use same steps as Algorithm 1 for $\boldsymbol{\beta_i}$, $\boldsymbol{\varrho_i}$, $\boldsymbol{\gamma_i}$ and $G_{i,j}$
  // Model A
  for $j = 1$ **to** K **do**
    $x \sim Bernoulli(0.5)$;
    if $x == 0$ **then**
      $A_{i,j} = Cat(\beta_{i,1}, ..., \beta_{i,M})$;
    else
      $A_{i,j} = Cat(\gamma_{i,1}, ..., \gamma_{i,M})$;
  // Model B
  for $j = 1$ **to** K **do**
    $x \sim Bernoulli(0.5)$;
    if $x == 0$ **then**
      $B_{i,j} = Cat(\gamma_{i,1}, ..., \gamma_{i,M})$;
    else
      $B_{i,j} = Cat(\beta_{i,1}, ..., \beta_{i,M})$;

Algorithm 3: Calculate Confidence Interval (CI)

Input: Γ^{alt}
$\hat{\Gamma} \leftarrow mean(\Gamma^{alt})$;
$\Gamma^{alt}_{sorted} \leftarrow$ sort(Γ^{alt});
// Choose 2.5th and 97.5th percentile (95% CI)
$\text{CI}_{lower} \leftarrow 2\hat{\Gamma} - \Gamma^{alt}_{sorted}[975]$;
$\text{CI}_{upper} \leftarrow 2\hat{\Gamma} - \Gamma^{alt}_{sorted}[25]$;
$\text{CI} \leftarrow [\text{CI}_{lower}, \text{CI}_{upper}]$;

where larger is better and $\Gamma(A, B, G) = \Gamma(B, G) - \Gamma(A, G)$, where smaller is better. For each hypothesis, a distribution over metric scores Γ^H is obtained by resampling. We calculate a p-value for Γ^{alt} & Γ^{null} by calculating the proportion of samples in the null distribution Γ^{null} that exceed the scores in the alternative distribution Γ^{alt}.

Confidence Interval Estimation

We utilize the Γ^{alt} bootstrap distribution to obtain 95% confidence intervals around the mean by using the reverse percentile method (Algorithm 3).

Metrics

We choose a set of metrics that range from simple plurality agreement to a more nuanced comparison of full response distributions and head-to-head performance. Collectively, these metrics provide a comprehensive view of how well models A and B align with a gold standard G when dealing with nominal data. We use the following metrics in our experiments:

- **Accuracy**. Accuracy is the most commonly used metric to compare models against each other. First, take the plurality vote for all items in A, B, and G. Then compute the accuracy for A and B by comparing against G.
  Accuracy for A against G:
  $$\Gamma_{Accuracy}(A, G) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(PV_A(i) = PV_G(i))$$
  where, $PV_X(i)$ is the plurality vote for item i in set X and $\mathbb{I}(\cdot)$ is the indicator function, which is 1 if the condition is true and 0 otherwise.
- **Total variation (TV)**. TV is related to Manhattan or L1 distance. It goes beyond the plurality vote and helps compare probability distributions for soft label evaluation. Compute the frequency of responses for all items in A, B, and G, normalize, and compute the mean Manhattan distance across all items in A and B against G.
  TV for A against G:
  $$\Gamma_{TV}(A, G) = \frac{1}{N}\sum_{i=1}^{N}\sum_{m\in\mathcal{M}} \big|P_A(m|i) - P_G(m|i)\big|$$
  where, $P_X(m|i)$ be the normalized frequency (probability distribution) of response m for item i in set X. This means $\sum_m P_X(m|i) = 1$. $\mathcal{M}$ is the set of all possible responses.
- **Wins**. Wins is a meta-metric used for item-level comparison. We use TV as the base metric for Wins, but any other metric can be used. Calculate TV for all items in A and B against G, then count the wins of A and B, i.e., the number of times A has less TV than B and vice-versa.
  Wins for A over B:
  $$\Gamma_{Wins}(A > B) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\text{TV}_A(i) < \text{TV}_B(i))$$
  where, $\text{TV}_A(i) = \sum_{m\in\mathcal{M}} |P_A(m|i) - P_G(m|i)|$ and $\text{TV}_B(i) = \sum_{m\in\mathcal{M}} |P_B(m|i) - P_G(m|i)|$.
- **KL-Divergence (KL-Div)**. KL-Divergence is another frequently used metric for comparing probability distributions. Calculate the frequency of responses for all items in A, B, and G. Then, compute the mean KL-divergence across all items in A and B against G.
  KL-Divergence for A against G:
  $$\Gamma_{KL}(A, G) = \frac{1}{N}\sum_{i=1}^{N}\sum_{m\in\mathcal{M}} P_G(m|i)\log\left(\frac{P_G(m|i)}{P_A(m|i)}\right)$$
  where, $P_X(m|i)$ be the normalized frequency (probability distribution) of response m for item i in set X.

Fitting to Real-World Datasets

We fit the prior parameters α of our model to real-world datasets by computing the maximum a posteriori (MAP) estimate of the model. We use mean absolute bias (MAB) to measure goodness of fit:

$$\text{MAB} = \frac{1}{M}\sum_{m=1}^{M} \big|\theta_m - E[\hat{\theta}_m]\big|,$$

where, θ_m is the percentage of category m in the dataset and $\hat{\theta_m} = \alpha_m / \sum_m \alpha_m$ is the expected rate of category m. It indicates how much, on average, the predicted parameters deviate from the actual values. More details about the MAP estimation can be found in the extended version.

Experiments

Datasets

We use the following datasets, each comprising various categories and multiple responses per item, for our experiments.

Toxicity Toxicity dataset (Kumar et al. 2021) consists of 107,620 social media comments labeled by 17,280 raters. (Number of categories $M = 2$, $\alpha = [1.37, 1.33]$)

DICES Diversity in Conversational AI Evaluation for Safety 350 dataset (Aroyo et al. 2023) consists of 350 chatbot conversations rated for safety by 123 raters across 16 safety dimensions. ($M = 3$, $\alpha = [5.22, 0.86, 2.75]$)

D3code D3code (Davani et al. 2024) is a large cross-cultural dataset comprising 4554 items, each labeled for offensiveness by 4309 raters from 21 countries and balanced across gender and age. ($M = 2$, $\alpha = [6.08, 2.88]$)

Jobs Jobs dataset (Liu et al. 2016) is a collection of 2000 job-related tweets labeled by 5 raters each. The raters answer 3 questions about each tweet, and the corresponding sets are denoted by JobsQ1/2/3. The categories in JobsQ1/2/3 represent the point of view of job-related information, employment status, and job transition events, respectively. We use **JobsQ1** and **JobsQ3** for our experiments. Here $M = 5$, $\alpha = [1039.76, 38.24, 35.57, 310.29, 46.02]$ and $M = 12$, $\alpha = [133.79, 834.51, 105.27, 3669.04, 206.80, 293.44, 585.58, 1278.56, 1874.82, 1838.49, 1576.10, 989.23]$, respectively.

Experimental Setup

We run experiments for hypothesis testing with different number of annotations ($N \times K$ = {100, 250, 500, 1000, 2500, 5000, 10000, 25000, 50000}) while ranging K from 1 to 500 (in increments of 1 till 10, then 20, then in increments of 20 from 20 onwards) for different metrics, and ϵ = {0.1, 0.2, 0.3, 0.4}. We use four metrics with four ϵ, yielding 16 sets of 282 experiments for each dataset.

	Toxicity	DICES	D3code	JobsQ1	JobsQ3
MAB	0.0111	0.0231	0.0029	0.0537	0.0869

Table 1: Mean absolute bias for the parameters.

For the real-world datasets, the value of parameter α is determined using the MAP estimate and $\rho = [1/M] \times M$ where M is fixed for each dataset. To estimate the p-values we repeat the sampling process 1000 times. Our estimates of the MAB fit are shown in Table 1.

We also experiment using different prior distributions for parameter α, **balanced** ($\alpha = [3] \times M$) and **unbalanced** ($\alpha = [10] + [3] \times M$) to simulate class imbalance, and varying the number of categories (M = {2, 3, 4, 5, 12}).

		Accuracy	TV	Wins	KL-Div
Toxicity (M=2)	NK	2500	1000	2500	1000
	p-value	0.012	0.015	0.012	0.022
	K	1	120	1	200
	Δ	0.040	0.074	0.040	0.044
DICES (M=3)	NK	1000	500	1000	1000
	p-value	0.036	0.017	0.028	0.020
	K	1	80	20	300
	Δ	0.055	0.063	0.346	0.082
D3code (M=2)	NK	2500	1000	2500	1000
	p-value	0.037	0.020	0.024	0.022
	K	2	140	60	100
	Δ	0.034	0.072	0.413	0.036
JobsQ1 (M=5)	NK	250	250	250	250
	p-value	0.035	0.015	0.036	0.035
	K	1	40	1	1
	Δ	0.104	0.050	0.104	2.864
JobsQ3 (M=12)	NK	500	250	500	500
	p-value	0.047	0.014	0.038	0.030
	K	100	240	80	500
	Δ	0.595	0.024	0.868	0.182
Unbalanced (M=2)	NK	1000	500	1000	1000
	p-value	0.031	0.044	0.031	0.014
	K	1	80	1	140
	Δ	0.050	0.074	0.050	0.047
Unbalanced (M=3)	NK	1000	500	1000	500
	p-value	0.039	0.023	0.040	0.031
	K	2	100	40	100
	Δ	0.061	0.061	0.473	0.068
Unbalanced (M=4)	NK	1000	500	1000	500
	p-value	0.049	0.013	0.022	0.021
	K	4	120	40	240
	Δ	0.089	0.054	0.520	0.084
Unbalanced (M=5)	NK	1000	500	1000	500
	p-value	0.045	0.009	0.015	0.010
	K	10	100	40	240
	Δ	0.138	0.043	0.545	0.098
Unbalanced (M=12)	NK	1000	250	500	500
	p-value	0.027	0.014	0.042	0.004
	K	80	240	60	460
	Δ	0.436	0.023	0.763	0.156

Table 2: Minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.3$).

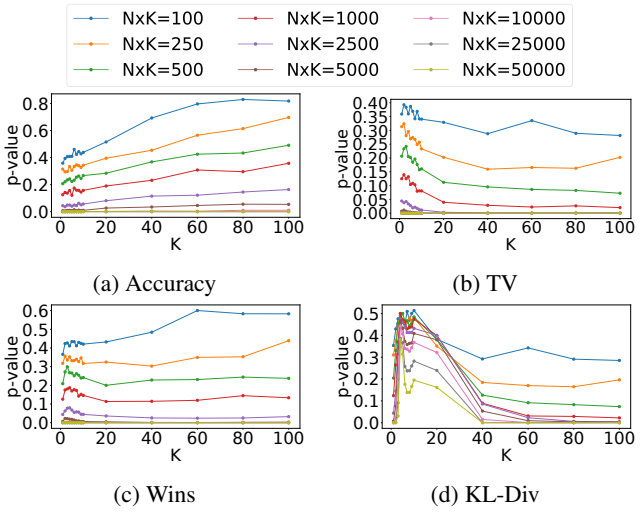
Results

Tables 2 shows the results for minimum p-value, K, and corresponding effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.3$). Table 3 shows the results for the lowest CI-width with the corresponding value of K and effect size - Δ for the lowest NK observed in Table 2. Our results suggest that whether or not a tradeoff exists, and where it is, depends much more on the metric used than the data source, and that the metrics behave very differently. They show that the TV metric requires the smallest number of $N \times K$ overall, and that this comes with a small number of $K > 10$.

Figures 3–4 show results for p-values and confidence intervals for $\epsilon = 0.3$ on the D3code dataset. Although there were exceptions, they exemplify many common observations found for other datasets (refer to the extended version).

		Accuracy	TV	Wins	KL-Div
Toxicity (M=2)	NK	2500	1000	2500	1000
	ci-width	0.050	0.067	0.050	0.085
	K	1	5	1	100
	Δ	0.117	0.063	0.728	0.042
DICES (M=3)	NK	1000	500	1000	1000
	ci-width	0.090	0.063	0.090	0.230
	K	1	7	1	100
	Δ	0.080	0.075	0.203	0.082
D3code (M=2)	NK	2500	1000	2500	1000
	ci-width	0.054	0.067	0.054	0.068
	K	1	7	1	100
	Δ	0.116	0.066	0.669	0.036
JobsQ1 (M=5)	NK	250	250	250	250
	ci-width	0.160	0.055	0.160	1.516
	K	1	8	1	80
	Δ	0.104	0.038	0.104	0.109
JobsQ3 (M=12)	NK	500	250	500	500
	ci-width	0.086	0.020	0.086	1.322
	K	1	1	1	100
	Δ	0.614	0.003	0.941	1.613
Unbalanced (M=2)	NK	1000	500	1000	1000
	ci-width	0.070	0.093	0.079	0.083
	K	2	7	1	80
	Δ	0.068	0.086	0.154	0.045
Unbalanced (M=3)	NK	1000	500	1000	500
	ci-width	0.079	0.063	0.079	0.145
	K	1	10	1	100
	Δ	0.111	0.028	0.180	0.028
Unbalanced (M=4)	NK	1000	500	1000	500
	ci-width	0.081	0.048	0.081	0.218
	K	1	6	1	100
	Δ	0.125	0.058	0.191	0.243
Unbalanced (M=5)	NK	1000	500	1000	500
	ci-width	0.070	0.039	0.070	0.467
	K	1	7	1	100
	Δ	0.128	0.049	0.187	0.515
Unbalanced (M=12)	NK	1000	250	500	500
	ci-width	0.056	0.019	0.080	1.292
	K	1	1	1	100
	Δ	0.105	0.002	0.920	1.264

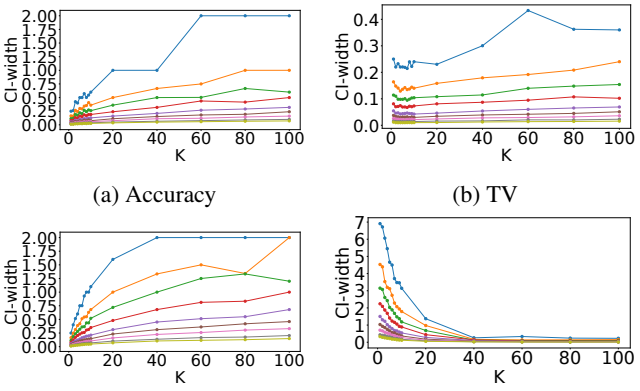
Table 3: Lowest CI-width with corresponding value of K and effect size (Δ) for lowest NK with $p < 0.05$ ($\epsilon = 0.3$).



(a) Accuracy (b) TV

(c) Wins (d) KL-Div

Figure 3: p-value plots for D3code dataset, $\epsilon = 0.3$.



(a) Accuracy (b) TV

(c) Wins (d) KL-Div

Figure 4: CI-width plots for D3code dataset, $\epsilon = 0.3$.

Accuracy We notice that the p-value increases as K increases for all $N \times K$ in all datasets. The increase is sharp until $K = 40$ for all $N \times K$ values, especially for lower values of ϵ (0.1, 0.2). Then, the p-values start plateauing for higher $N \times K$ values, but continue to increase for lower $N \times K$ values. As ϵ increases, p-values decrease as expected. Having more responses per item (K) does not seem to be helpful for accuracy. CI-width and effect sizes increase with increasing K except for JobsQ1.

With some exceptions (particularly JobsQ1), **accuracy** yields the most reliable results (lowest scores) for both p-values and confidence intervals when $K = 1$ or very small k. For **total variation**, higher K usually yields lower, more reliable p-values, with improvements occurring even for $K = 300$, but confidence intervals generally increase with K, and tend to have best results for $K \in [1, 10]$. **Wins** tends to have minimum p-values for $K \in [25, 100]$), where confidence intervals improve with increasing K, with improvements occurring even for $K = 300$. **KL-divergence** is particularly interesting. While confidence intervals improve with increasing K, settling down at around $K = 100$, p-values improve for $K \in [2, 5]$ then get worse and then settle to lower numbers by around $k = 300$.

TV For TV, p-value increases sharply with increasing K till $K = 10$, and then starts to decrease thereafter for all $N \times K$ for $\epsilon = 0.1$. For the remaining values of ϵ, the p-values generally decrease with increasing K. We also notice an elbow plot emerging for total variation, suggesting an optimal value of K for the datasets. As ϵ increases, p-values decrease as expected. Having more responses per item (K) seems to be helpful for total variation and results in lower p-values. CI-width and effect sizes increase as K increases.

For simulations with balanced categories, p-values monotonically decrease with increasing K as M gets larger. CI-

width and effect sizes increase as K increases. For simulations with unbalanced categories, p-values decrease with increasing K as M gets larger, whereas CI-width and effect sizes increase as K increases.

Wins For Wins, p-value increases sharply with increasing K till $K = 10$, and then starts to plateau for all $N \times K$ for $\epsilon = 0.1$. For the remaining values of ϵ, p-values start going back up as K gets higher. As ϵ increases, p-values decrease as expected. There seems to be an optimal K for Dices, D3code, and JobsQ3 dataset. CI-width and effect sizes increase with increasing K.

For simulations with balanced categories, p-values generally decrease with increasing K and start to go back up for lower $N \times K$. CI-width and effect sizes increase as K increases, with some exceptions. For simulations with unbalanced categories, p-values CI-width, and effect sizes show similar trends as balanced categories.

KL-Divergence We notice that p-values exhibit double peaks till about $K = 40$ for all datasets and for all $N \times K$. The p-values continue to decrease for higher values of K. As ϵ increases, p-values decrease as expected. CI-width and effect sizes generally decrease as K increases, except for JobsQ3 with one initial peak.

For simulations with balanced categories, p-values exhibit a single or double peak initially but settle down with higher K. CI-width and effect sizes decrease with increasing K; however start to have one peak as M increases. For simulations with unbalanced categories, p-values CI-width, and effect sizes show similar behavior to balanced categories.

Discussion

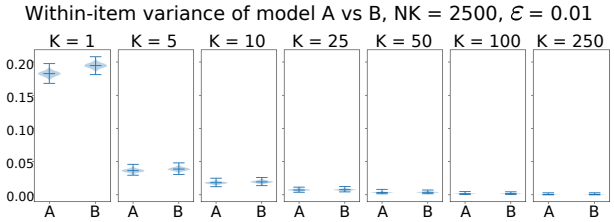


Figure 5: Distribution of scores for machines A and B for within-item variance estimates, for $N \times K = 2500$.

Figure 5 provides some insight into our results. It shows, using the distribution fitted to the JobsQ3 data for $N \times K = 2500$ as an example (but which is representative of other distributions). We see, first, that within-item variance drops precipitously from $K = 1$ to $K = 5$, and to nearly zero by $K = 100$, by which point we can assume to have enough within-item samples for nearly any item-level metric.

Looking at the metric scores, it is worth noting that raw scores generally improve as k increases (though we do not show this here). Taking accuracy as an example, this is easy to see why: the larger K is, the more likely it is that the most likely response is the most common one. So as long as the machine and gold have the same most frequent response, accuracy increases. However, accuracy increases for *both* machines, and this, under our error model, allows for machine

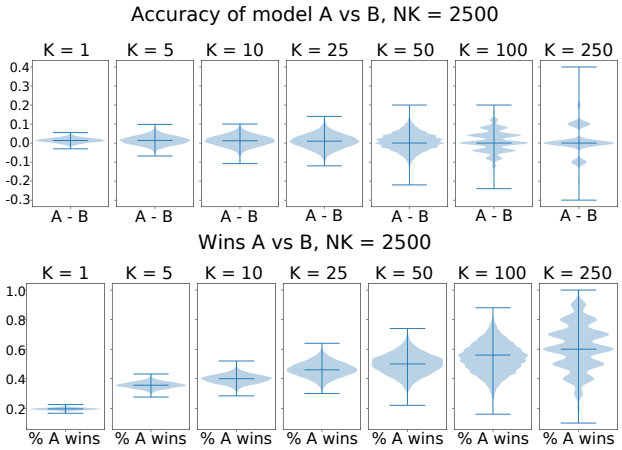


Figure 6: Distribution of scores differences for machines A and B against gold data over the Toxicity distribution, for $N \times K = 2500$, $\epsilon = 0.01$ for Accuracy and Wins.

B to catch up. And as N decreases, we would expect there to be more variance between individual samples.

Figure 6 shows that for two of the metrics, accuracy and wins, their variance increases. Yet with wins, this increase occurs almost exclusively in a positive direction, whereas for accuracy (where p-values increase as K increases), the variance is both positive and negative.

There are several limitations to our approach. Our simulator does not account for the machines using soft labels. We did not explore the impact of different noise models, and we see this as an important piece of the puzzle. We have not validated our results with real data collected based on the analysis of our simulation. This is because doing so would require us to collect multiple sets of data that have more annotations than are needed, and this is beyond our lab's budget.

Conclusion

In this work, we investigated the critical trade-off between the number of items (N) and the number of responses per item (K) for achieving reliable machine learning model evaluation under a fixed budget. Our findings demonstrate that increasing K is often a more effective strategy for achieving reliable evaluation than increasing N. We discovered, across a diverse set of datasets, that accounting for the full human response distribution can be achieved with a surprisingly modest budget ($N \times K$) of 1000 or less, with $K > 10$. Furthermore, we established that the relationship between N and K is heavily dependent on the chosen evaluation metric. Metrics that are more sensitive to the distributional nature of human responses benefit greatly from higher values of K. Our research provides a clear, data-driven methodology for ML practitioners to design more effective and budget-conscious evaluations. By moving beyond the single-truth paradigm and strategically collecting multiple responses, the field can build greater trust and confidence in model performance. Embracing human disagreement is not an expensive luxury but a cornerstone of robust and meaningful machine learning evaluation.

References

Aroyo, L.; Taylor, A.; Díaz, M.; Homan, C.; Parrish, A.; Serapio-García, G.; Prabhakaran, V.; and Wang, D. 2023. DICES Dataset: Diversity in Conversational AI Evaluation for Safety. In Oh, A.; Neumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 53330–53342. New Orleans, Louisiana, USA: Curran Associates, Inc.

Arvan, M.; Pina, L.; and Parde, N. 2022. Reproducibility in Computational Linguistics: Is Source Code Enough? In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2350–2361. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Baker, M. 2016. 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604): 452–454.

Barile, F.; Najafian, S.; Draws, T.; Inel, O.; Rieger, A.; Hada, R.; and Tintarev, N. 2021. Toward benchmarking group explanations: Evaluating the effect of aggregation strategies versus explanation. In *Perspectives on the Evaluation of Recommender Systems Workshop 2021: co-located with the 15th ACM Conference on Recommender Systems (RecSys 2021)*. Amsterdam, The Netherlands: ACM New York, NY, USA.

Basile, V.; Fell, M.; Fornaciari, T.; Hovy, D.; Paun, S.; Plank, B.; Poesio, M.; and Uma, A. 2021. We Need to Consider Disagreement in Evaluation. In Church, K.; Liberman, M.; and Kordoni, V., eds., *Proceedings of the 1st Workshop on Benchmarking: Past, Present and Future*, 15–21. Online: Association for Computational Linguistics.

Belz, A.; Thomson, C.; Reiter, E.; and Mille, S. 2023. Non-Repeatable Experiments and Non-Reproducible Results: The Reproducibility Crisis in Human Evaluation in NLP. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023*, 3676–3687. Toronto, Canada: Association for Computational Linguistics.

Cabitza, F.; Campagner, A.; and Basile, V. 2023. Toward a Perspectivist Turn in Ground Truthing for Predictive Computing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6): 6860–6868.

Davani, A. M.; Díaz, M.; Baker, D.; and Prabhakaran, V. 2024. D3CODE: Disentangling Disagreements in Data across Cultures on Offensiveness Detection and Evaluation. ArXiv:2404.10857 [cs].

Dror, R.; Baumer, G.; Shlomov, S.; and Reichart, R. 2018. The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing. In Gurevych, I.; and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1383–1392. Melbourne, Australia: Association for Computational Linguistics.

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

Gundersen, O. E.; and Kjensmo, S. 2018. State of the Art: Reproducibility in Artificial Intelligence. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1): 1644–1651.

Homan, C. M.; Serapio-Garcia, G.; Aroyo, L.; Diaz, M.; Parrish, A.; Prabhakaran, V.; Taylor, A. S.; and Wang, D. 2023a. Intersectionality in Conversational AI Safety: How Bayesian Multilevel Models Help Understand Diverse Perceptions of Safety.

Homan, C. M.; Wein, S.; Aroyo, L. M.; and Welty, C. 2023b. How Many Raters Do You Need? Power Analysis for Foundation Models. In *Proceedings of I Can't Believe It's Not Better (ICBINB): Failure Modes in the Age of Foundation Models*.

Hutson, M. 2018. Artificial intelligence faces reproducibility crisis. *Science*, 359(6377): 725–726.

Kumar, D.; Kelley, P. G.; Consolvo, S.; Mason, J.; Bursztein, E.; Durumeric, Z.; Thomas, K.; and Bailey, M. 2021. Designing toxic content classification for a diversity of perspectives. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*, 299–318.

Liu, T.; Homan, C.; Ovesdotter Alm, C.; Lytle, M.; Marie White, A.; and Kautz, H. 2016. Understanding Discourse on Work and Job-Related Well-Being in Public Social Media. In Erk, K.; and Smith, N. A., eds., *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1044–1053. Berlin, Germany: Association for Computational Linguistics.

Mieskes, M.; Fort, K.; Névéol, A.; Grouin, C.; and Cohen, K. 2019. Community Perspective on Replicability in Natural Language Processing. In Mitkov, R.; and Angelova, G., eds., *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, 768–775. Varna, Bulgaria: INCOMA Ltd.

Mostafazadeh Davani, A.; Díaz, M.; and Prabhakaran, V. 2022. Dealing with Disagreements: Looking Beyond the Majority Vote in Subjective Annotations. *Transactions of the Association for Computational Linguistics*, 10: 92–110.

Pandita, D.; Weerasooriya, T. C.; Dutta, S.; Luger, S. K.; Ranasinghe, T.; KhudaBukhsh, A. R.; Zampieri, M.; and Homan, C. M. 2024. Rater Cohesion and Quality from a Vicarious Perspective. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Findings of the Association for Computational Linguistics: EMNLP 2024*, 5149–5162. Miami, Florida, USA: Association for Computational Linguistics.

Pham, H. V.; Qian, S.; Wang, J.; Lutellier, T.; Rosenthal, J.; Tan, L.; Yu, Y.; and Nagappan, N. 2020. Problems and opportunities in training deep learning software systems: an analysis of variance. In *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 771–783. Virtual Event Australia: ACM. ISBN 9781450367684.

Plank, B. 2022. The "Problem" of Human Label Variation: On Ground Truth in Data, Modeling and Evaluation. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 10671–10682. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Prabhakaran, V.; Homan, C.; Aroyo, L.; Parrish, A.; Taylor, A.; Díaz, M.; and Wang, D. 2023. A Framework to Assess (Dis) agreement Among Diverse Rater Groups.

Prabhakaran, V.; Mostafazadeh Davani, A.; and Diaz, M. 2021. On Releasing Annotator-Level Labels and Information in Datasets. In Bonial, C.; and Xue, N., eds., *Proceedings of the Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop*, 133–138. Punta Cana, Dominican Republic: Association for Computational Linguistics.

Raff, E. 2019. A Step Toward Quantifying Independently Reproducible Machine Learning Research. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Vancouver, Canada: Curran Associates, Inc.

Snow, R.; O'connor, B.; Jurafsky, D.; and Ng, A. Y. 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*, 254–263.

Søgaard, A.; Johannsen, A.; Plank, B.; Hovy, D.; and Martínez Alonso, H. 2014. What's in a p-value in NLP? In Morante, R.; and Yih, S. W.-t., eds., *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, 1–10. Ann Arbor, Michigan: Association for Computational Linguistics.

Uma, A. N.; Fornaciari, T.; Hovy, D.; Paun, S.; Plank, B.; and Poesio, M. 2021. Learning from disagreement: A survey. *Journal of Artificial Intelligence Research*, 72: 1385–1470.

van der Lee, C.; Gatt, A.; van Miltenburg, E.; Wubben, S.; and Krahmer, E. 2019. Best practices for the human evaluation of automatically generated text. In van Deemter, K.; Lin, C.; and Takamura, H., eds., *Proceedings of the 12th International Conference on Natural Language Generation*, 355–368. Tokyo, Japan: Association for Computational Linguistics.

Weerasooriya, T.; Dutta, S.; Ranasinghe, T.; Zampieri, M.; Homan, C.; and KhudaBukhsh, A. 2023. Vicarious Offense and Noise Audit of Offensive Speech Classifiers: Unifying Human and Machine Disagreement on What is Offensive. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 11648–11668. Singapore: Association for Computational Linguistics.

Wein, S.; Homan, C.; Aroyo, L.; and Welty, C. 2023. Follow the leader(board) with confidence: Estimating p-values from a single test set with item and response variance. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023*, 3138–3161. Toronto, Canada: Association for Computational Linguistics.