Inherent Trade-Offs between Diversity and Stability in Multi-Task Benchmarks

Guanhua Zhang Moritz Hardt

Max Planck Institute for Intelligent Systems, Tübingen and Tübingen AI Center

Abstract

We examine multi-task benchmarks in machine learning through the lens of social choice theory. We draw an analogy between benchmarks and electoral systems, where models are candidates and tasks are voters. This suggests a distinction between cardinal and ordinal benchmark systems. The former aggregate numerical scores into one model ranking; the latter aggregate rankings for each task. We apply Arrow's impossibility theorem to ordinal benchmarks to highlight the inherent limitations of ordinal systems, particularly their sensitivity to the inclusion of irrelevant models. Inspired by Arrow's theorem, we empirically demonstrate a strong trade-off between diversity and sensitivity to irrelevant changes in existing multi-task benchmarks. Our result is based on new quantitative measures of diversity and sensitivity that we introduce. Sensitivity quantifies the impact that irrelevant changes to tasks have on a benchmark. Diversity captures the degree of disagreement in model rankings across tasks. We develop efficient approximation algorithms for both measures, as exact computation is computationally challenging. Through extensive experiments on seven cardinal benchmarks and eleven ordinal benchmarks, we demonstrate a clear trade-off between diversity and stability: The more diverse a multi-task benchmark, the more sensitive to trivial changes it is. Additionally, we show that the aggregated rankings of existing benchmarks are highly unstable under irrelevant changes. The codes and data are available at `https://socialfoundations.github.io/benchbench/`.

1 Introduction

At this point, there is little agreement about what the right benchmark is for different tasks in machine learning (Ethayarajh and Jurafsky, 2020; Bowman and Dahl, 2021; Kiela et al., 2021). Natural language understanding, for example, has hundreds of different benchmarks, each measuring different qualities of a model (Storks et al., 2019). On the one hand, multiple benchmarks are desirable when it comes to creating a diverse canvas of evaluation results. On the other hand, the plurality of different benchmarks makes it challenging to consistently measure progress, as different benchmarks suggest different model rankings.

The de facto solution to the problem are multi-task benchmarks (Wang et al., 2018, 2019; Hendrycks et al., 2020). Major recent developments, such as BigBench (Srivastava et al., 2022) and HELM (Liang et al., 2023), combine hundreds of evaluation tasks into a single benchmark. The hope is that by aggregating many tasks into one, a reliable and representative picture of model performance will emerge.

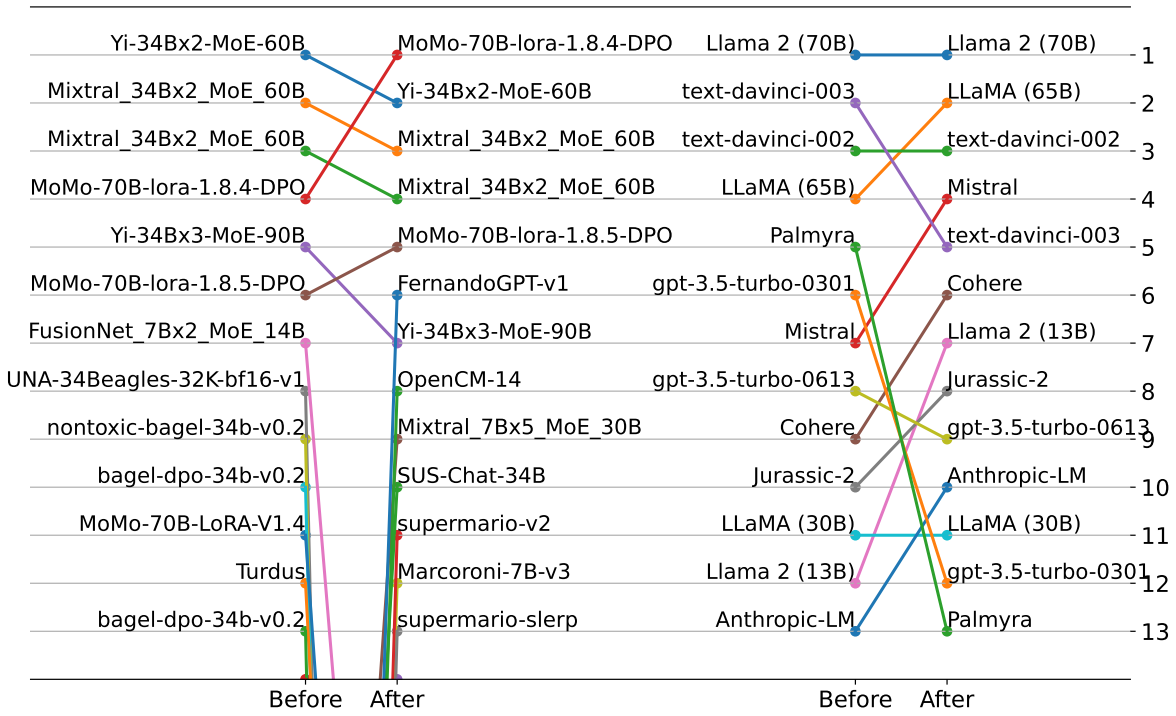

Figure 1: Ranking changes after irrelevant changes on tasks. For cardinal benchmark `OpenLLM` (left), *Before* refers to the original ranking, and *After* is the new ranking after injecting label noises into different tasks. For ordinal benchmark `HELM-accuracy` (right), *Before* refers to the ranking based on only the original top-20% models, while *After* is the new relative ranking after adding irrelevant models from the rest 80%. y-axis refers to the ranking.

In this work, we scrutinize multi-task benchmarks through the lens of social choice theory. In doing so, we analogize multi-task benchmarks with electoral systems. Models stand in analogy with candidates in the electoral system, and tasks with voters. Each task in a multi-task benchmark may rank candidate models differently. The benchmark must determine a ranking of candidates given the different votes.

A robust lesson from social choice theory is that there is no perfect voting system. Celebrated results, such as Arrow's impossibility theorem (Arrow, 1950, 1951), point at inherent limitations in the design of desirable voting rules (Taylor, 2005). Inspired by social choice theory, we surface an important trade-off in multi-task benchmarks between a measure of diversity and a measure of robustness to irrelevant changes. In a nutshell, we demonstrate empirically that current multi-task benchmarks fail to be both robust and diverse. Instead, one comes at the expense of the other.

1.1 Our contributions

We propose a distinction between *cardinal benchmark systems* and *ordinal benchmark systems*. Cardinal benchmark systems aggregate multiple rankings into one on the basis of numerical scores, such as accuracy numbers. Ordinal benchmark systems instead aggregate rankings into a single ranking. BigBench is an example of a cardinal benchmark, ranking by average accuracy over the tasks.

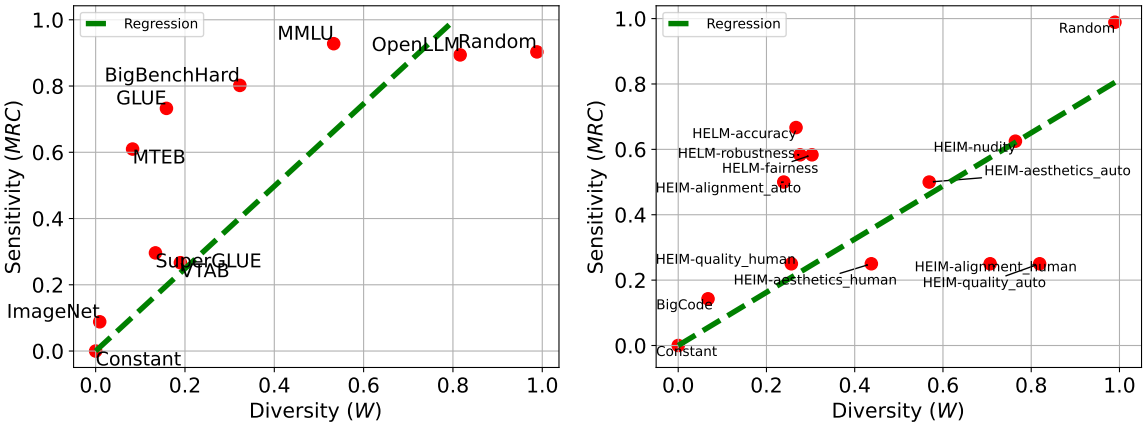

Figure 2: Trade-off between benchmark diversity and sensitivity to irrelevant changes. Left: Cardinal benchmarks. Right: Ordinal benchmarks. Sensitivity is measured in terms of the maximum normalized rank change (MRC) possible via irrelevant task changes. Diversity is measured by Kendall's coefficient of concordance (W). The green curve is a linear regression on all points without fitting the intercept.

HELM is an example of an ordinal system, comparing any two models by how often one ranks higher than the other.

To start, we point out that Arrow's impossibility result directly applies to ordinal systems, such as HELM. We observe that in the case of ordinal benchmarks, the desirable property that fails in Arrow's language is *independence of irrelevant alternatives*. What this means is that adding irrelevant models to a benchmark can perturb the order of top contenders. We demonstrate that this is indeed possible with HELM and similar benchmarks.

Arrow's result is neither quantitative, nor does it apply to cardinal systems. Inspired by Arrow's theorem, we introduce a quantitative measure of *sensitivity to irrelevant changes*. Sensitivity measures how responsive a benchmark is to trivial transformations of tasks. For example, adding a fraction of random labels to a task does not change the relative performance of models, thus resulting in an equivalent task.

It is easy to design benchmarks that are robust to such irrelevant changes. Simply take a single-task benchmark. Or take the single task and copy it many times to obtain a multi-task benchmark. We therefore contrast our measure of sensitivity with a measure of *diversity*. Diversity measures the degree to which different tasks disagree in their model rankings. Multi-task benchmarks lacking in diversity are essentially single-task benchmarks.

Our measures of diversity and sensitivity are computationally hard to compute exactly. We therefore developed efficient approximation algorithms for both.

Through comprehensive experiments, we demonstrate that there is a strong trade-off between diversity and sensitivity in current multi-task benchmarks. The more diverse a multi-task benchmark, the more sensitive to trivial changes it is. In other words, the pursuit of diversity compromises sensitivity, and striving for robustness necessitates sacrificing diversity. We confirm this trade-off in seven cardinal benchmarks and eleven ordinal benchmarks from natural language understanding and computer vision.

The most stable benchmark, by our measure, is a constant benchmark. The most diverse benchmark is a random benchmark. We show that all existing multi-task benchmarks strike a trade-off no better than a linear interpolation between random and constant. In particular, our empirical analysis reveals that current benchmarks are highly unstable to irrelevant changes. For illustration, Figure 1 gives an example where both a cardinal benchmark (`OpenLLM`) and an ordinal benchmark (`HELM-accuracy`) suffer from significant ranking changes after trivial task transformations. Figure 2 summarizes the trade-off between diversity and sensitivity for both cardinal and ordinal benchmarks.

2 Related Works

Benchmarks are at the foundation of applied machine learning research, underpinning many of its successes (Donoho, 2023; Koch et al., 2021; Zhang et al., 2019; Ott et al., 2022). Although many benchmarks have been proposed, far fewer works have studied benchmarks as a scientific subject itself; see Hardt and Recht (2022) for an overview. With recent machine learning models achieving impressive abilities across many different evaluation settings (Ramesh et al., 2021; Team, 2023; OpenAI, 2023; Touvron et al., 2023a,b), the spotlight has increasingly turned to multi-task benchmarks. Multi-task benchmarks aim to provide a diverse and holistic evaluation of machine learning models by covering many different tasks and metrics (Liang et al., 2023; Lee et al., 2023; Srivastava et al., 2022; Wang et al., 2018, 2019). Concurrently, various concerns regarding benchmarks have surfaced, highlighting their limitations, see, e.g., (Liao et al., 2021; Bowman and Dahl, 2021; Zhang et al., 2023; Boubdir et al., 2023), in particular, susceptibility to chosen tasks (Dehghani et al., 2021; Alzahrani et al., 2024), non-smooth utility functions (Ethayarajh and Jurafsky, 2020), data contamination (Roberts et al., 2023; Magar and Schwartz, 2022), possibility for overfitting due to repeated use of test sets (Dwork et al., 2014; Blum and Hardt, 2015; Feldman et al., 2019; Mania et al., 2019; Arora and Zhang, 2021). Shirali et al. (2023) demonstrated inherent limitations of *dynamic benchmarks*, another recent benchmark design paradigm that aims to mitigate shortcomings of static single-task benchmarks.

In our research, we focus on the challenge of aggregating performance measures within multi-task benchmarks. While existing studies have raised concerns about the aggregation problem in benchmarks, they have primarily focused on cardinal aggregation by mean scores (Mania et al., 2019; Colombo et al., 2021; Peyrard et al., 2017; Mishra and Arunkumar, 2021). As a result, there is a shift towards ordinal benchmarks where only relative performances in each task are used for aggregation Liang et al. (2023); Lee et al. (2023). Himmi et al. (2023) adopt a compatible partial ranking approach to address missing scores in benchmarks and introduce a Borda count-based aggregation method (Kelly, 1988). Colombo et al. (2022) propose a new aggregation process to fix the scale difference problem of cardinal based on Kemeny consensus (Shapiro and Hellman, 1993). Rofin et al. (2022) propose VOTE'N'RANK, which comprises eight procedures that depend on rankings for each task. In our study, we highlight the fundamental compromise one must navigate between diversity and stability for both cardinal and ordinal benchmarks.

3 A Social Choice Perspective for Benchmarks

Social choice theory addresses the problem of aggregating individual preferences to select the best option or candidate (Kelly, 1988). In the context of multi-task machine learning benchmarks, we adopt this framework by considering each task as an individual voter. The tasks, as voters, provide preference scores to different candidate models, akin to how individuals might vote for political candidates. The problem of aggregating these task-based votes into a cohesive ranking of models parallels the challenge in social choice of electing a candidate that best represents the preferences of the electorate.

From this perspective, we divide multi-task benchmarks into two classes: cardinal benchmarks and ordinal benchmarks. Cardinal benchmarks collect model scores from tasks, translate quantitative performance into a single average score per model, and rank all candidate models based on it. Ordinal benchmarks, on the other hand, only utilize relative rankings rather than absolute scores. Every task ranks the models based on performance, and the final model rankings emerge from an aggregation of these ordinal positions.

Notation. We need some notation to formalize the problem:

- $\mathcal{T} = (T_1, T_2, \ldots, T_n)$ represents the list of all n tasks in the benchmark, analogous to voters.
- $\mathcal{M}$ refers to the set of all potential candidate models that could be evaluated by the benchmark.
- Let $\mathcal{L} = (L_1, L_2, \ldots, L_m)$ be any non-empty list of candidate models with m models, where $L_i \in \mathcal{M}$ for any i.
- For any $\mathcal{L}$, we define s_{ij} as the score for the i-th model in $\mathcal{L}$ in task T_j. For simplicity, we abuse the notations and use $\boldsymbol{s}_j = (s_{1j}, s_{2j}, \ldots, s_{mj})$ as scores in any task T_j, and $\boldsymbol{S} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n)$ as scores over all tasks.
- For any $\mathcal{L}$, we define r_{ij} as the rank for the i-th model $\mathcal{L}$ in task T_j *w.r.t.* $\mathcal{L}$. For simplicity, we abuse the notations and use $\boldsymbol{r}_j = (r_{1j}, r_{2j}, \ldots, r_{mj})$ as ranks in any task T_j, and $\boldsymbol{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_n)$ as ranks over all tasks.
- A cardinal benchmark is defined as a function $f^{\mathrm{c}} = h^{\mathrm{c}} \circ g^{\mathrm{c}}$, which is composed of the scoring function g^{c} and the aggregation function h^{c}. Specifically, g^{c} takes a list of models $\mathcal{L}$ as input and outputs the corresponding scores for each index over all tasks, *i.e.*, $\boldsymbol{S} = g^{\mathrm{c}}(\mathcal{L})$. The scores $\boldsymbol{S}$ are fed into h^{c}, which outputs the final ranking $\boldsymbol{r}^{\mathrm{c}} = (r_1^{\mathrm{c}}, r_2^{\mathrm{c}}, \ldots, r_m^{\mathrm{c}})$, *i.e.*, $\boldsymbol{r}^{\mathrm{c}} = h^{\mathrm{c}}(\boldsymbol{S})$.
- An ordinal benchmark is defined as a function $f^{\mathrm{o}} = h^{\mathrm{o}} \circ g^{\mathrm{o}}$, which is composed of the scoring function g^{o} and the aggregation function h^{o}. Specifically, g^{o} takes a list of models $\mathcal{L}$ as input and outputs the corresponding rankings for each index over all tasks, *i.e.*, $\boldsymbol{R} = g^{\mathrm{o}}(\mathcal{L})$. The rankings $\boldsymbol{R}$ are fed into h^{o}, which outputs the final ranking $\boldsymbol{r}^{\mathrm{o}} = (r_1^{\mathrm{o}}, r_2^{\mathrm{o}}, \ldots, r_m^{\mathrm{o}})$, *i.e.*, $\boldsymbol{r}^{\mathrm{o}} = h^{\mathrm{o}}(\boldsymbol{R})$.
- We use $\textsc{rankdata}(\cdot)$ as the operator of getting rank.

More specifically, in cardinal benchmarks, an aggregated score is first calculated for each model, in most cases, by averaging the scores (Wang et al., 2018, 2019). For any candidate model list $\mathcal{L}$,

the final ranking is then calculated by sorting the average scores as follows,

$$\begin{aligned} \boldsymbol{r}^{\mathrm{c}} = h^{\mathrm{c}}(\boldsymbol{S}) = \textsc{rankdata}\big((\bar{s}_1, \ldots, \bar{s}_m)\big)\,, \\ \text{where} \quad \bar{s}_i = \frac{1}{n}\sum_{j=1}^{n} s_{ij}\,. \end{aligned} \tag{1}$$

In contrast, most existing popular ordinal benchmarks in machine learning calculate the *winning rate* for each model (Liang et al., 2023; Lee et al., 2023; Ben Allal et al., 2022). For any candidate model list $\mathcal{L}$, the winning rate for model L_i represents the probability that its rank r_{ik} is lower than r_{jk} for a randomly selected opponent model L_j and task T_k. By referring to $\mathcal{I}(\cdot)$ as the indicator function, the final ranking is calculated as follows,

$$\begin{aligned} \boldsymbol{r}^{\mathrm{o}} = h^{\mathrm{o}}(\boldsymbol{R}) = \textsc{rankdata}\big((\bar{w}_1, \ldots, \bar{w}_m)\big)\,, \\ \text{where} \quad \bar{w}_i = \frac{1}{m}\sum_{j=1}^{m} w_{ij}\,,\ w_{ij} = \frac{1}{n}\sum_{k=1}^{n} \mathcal{I}(r_{ik} < r_{jk})\,. \end{aligned} \tag{2}$$

Arrow's Impossibility Theorem for Benchmarks Arrow's Impossibility Theorem, a cornerstone in social choice theory, posits that no system can flawlessly translate individual preferences into a group ranking (Arrow, 1950, 1951). Adapted to the case of multi-task benchmarks, the theorem says the following (proof in Appendix B).

Theorem 3.1 (Arrow's Impossibility Theorem for Benchmarks)**.** *No ordinal benchmark f^o can fulfill the following conditions simultaneously:*

1. ***Non-Dictatorship:*** *There is no task T_i such that, for any $\mathcal{L}$ and any index pair (x, y) , when $r_{xi} < r_{yi}$, then $r^o_x < r^o_y$.*
2. ***Pareto Efficiency:*** *For any L and any index pair (x, y), if $r_{xi} < r_{yi}$ for every task $T_i \in \mathcal{T}$, then $r^o_x < r^o_y$.*
3. ***Independence of Irrelevant Alternatives (IIA):*** *Let $\mathcal{L}$ and $\mathcal{L}'$ be any two lists of models. For any index pair (x, y), if x and y have the same relative order in $g^o(\mathcal{L})$ and $g^o(\mathcal{L}')$ for all tasks, then x and y have the same relative order in $f^o(\mathcal{L})$ and $f^o(\mathcal{L}')$.*
4. ***Universality:*** *The benchmark has at least three tasks. The benchmark has as domain all finite lists with at least three models. The scoring function g^o has full range over all logically possible values for $\boldsymbol{R}$. The aggregation function h^o has full domain over all logically possible values for $\boldsymbol{R}$.*

For the ordinal benchmarks introduced in equation 2, the *IIA* condition is especially problematic since introducing a new model can perturb the winning rate of existing models, and as a result, change the aggregated ranking of existing models. For example, assume there are three candidate models $\mathcal{L} = (L_1, L_2, L_3)$ and nine tasks $\mathcal{T} = (T_1, T_2, \ldots, T_9)$, and the rankings $\boldsymbol{R}$ are as follows,

- for any task in $\{T_i\}_{i=1}^{4}$, $r_{1i} < r_{2i} < r_{3i}$,
- for any task in $\{T_i\}_{i=5}^{7}$, $r_{2i} < r_{3i} < r_{1i}$,
- for any task in $\{T_i\}_{i=8}^{9}$, $r_{3i} < r_{1i} < r_{2i}$.

The winning rates are $\bar{w}_1 = 10/27$, $\bar{w}_2 = 10/27$, $\bar{w}_3 = 7/27$, so we have $\boldsymbol{r}^{\mathrm{o}}_1 = \boldsymbol{r}^{\mathrm{o}}_2 < \boldsymbol{r}^{\mathrm{o}}_3$. Now we add one extra candidate model and get $\mathcal{L}' = (L_1, L_2, L_3, L_4)$. The rankings $\boldsymbol{R}'$ are as follows,

- for any task in $\{T_i\}_{i=1}^4$, $r'_{1i} < r'_{2i} < r'_{4i} < r'_{3i}$,
- for any task in $\{T_i\}_{i=5}^7$, $r'_{2i} < r'_{4i} < r'_{3i} < r'_{1i}$,
- for any task in $\{T_i\}_{i=8}^9$, $r'_{3i} < r'_{1i} < r'_{2i} < r'_{4i}$.

Then the winning rates are $\bar{w}'_1 = 17/36$, $\bar{w}'_2 = 19/36$, , $\bar{w}'_3 = 9/36$, so we have $\boldsymbol{r}_2^{o\prime} < \boldsymbol{r}_1^{o\prime} < \boldsymbol{r}_3^{o\prime}$. Note that the relative ranking among $\{L_1, L_2, L_3\}$ does not change over the nine tasks, but the final ranking has been different.

While Arrow's Impossibility Theorem is mainly concerned with ordinal voting systems, criticisms extend to cardinal systems as well. The main concern lies in the interpersonal comparability between voters (Drakopoulos, 1989). The validity of interpersonal comparison has been challenged as transforming any particular scale for individual preferences has been widely recognized as arbitrary (Sen, 2017; Arrow, 1951). In the context of cardinal benchmarks, the scale discrepancies among tasks could result in a situation where the aggregate performance disproportionately reflects the score of a single task, thereby distorting the benchmark's intent to represent all tasks effectively (Colombo et al., 2022; Himmi et al., 2023) and thus violating *Non-Dictatorship*. As a result, outliers or skewed distributions can significantly influence the final ranking, undermining the reliability of the cardinal benchmark assessments. Even if the scores are similar in scale across tasks, the underlying difficulty of each task may differ, *i.e.*, improvements in one task are easier to come by than in another. Consequently, a cardinal benchmark would then reward improvements in the easier task more than in the harder task, leading to discrepancies in how improvements are valued.

Although this discussion suggests potential issues in multi-task benchmarks that are informed by Arrow's theorem, we have yet to establish quantitative metrics that can gauge the severity of these issues within existing benchmarks. The next section will provide such quantitative metrics.

4 Diversity and Sensitivity in Multi-Task Benchmarks

Inspired by Arrow's impossibility theorem, in this section, we propose two quantitative measurements for multi-task benchmarks, *diversity* and *sensitivity*. *Diversity* is used to measure the ranking disagreement among all tasks, while *sensitivity* measures how vulnerable the final ranking of the benchmarks is toward irrelevant changes that do not change the relative performance of models.

4.1 Diversity in Multi-Task Benchmarks.

Let $\mathcal{L} = (L_1, L_2, \ldots, L_m)$ contain all models in $\mathcal{M}$ without duplicates, *i.e.*, $m = |\mathcal{M}|$, we define the *diversity* with reversed Kendall's coefficient of concordance W, which is to assess the disagreement among model rankings on each task as follows,

$$\begin{aligned} W &= 1 - 12\Sigma/(n^2(m^3 - m))\,, \\ \text{where} \quad \Sigma &= \sum_{i=1}^{m} (\bar{r}_i - \tilde{r})^2,\ \ \tilde{r} = \sum_{i=1}^{m} r_i,\ \ \bar{r}_i = \sum_{j=1}^{n} r_{ij}. \end{aligned} \tag{3}$$

$W = 0$ means all model rankings are the same across all tasks, while $W = 1$ means random or highly varied rankings. For example, if the benchmark is composed of only one task, repeating multiple times, then the *diversity* would be zero.

The definition of *diversity* is inspired by the *Universality* condition in Theorem 3.1, which indicates that a benchmark should accommodate all possible values for the rank matrix $\boldsymbol{R}$, meaning that any configuration of model ranks across the tasks in $\mathcal{T}$ should be feasible. This condition can be trivially satisfied when there is only one task in $\mathcal{T}$ by rearranging the models in $\mathcal{L}$. However, it becomes challenging in a multi-task scenario, particularly if the tasks share high correlations in their evaluations of models. For instance, if all tasks in $\mathcal{T}$ are merely replicas of a single task, the situation will never arise where the ranking vectors $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ differ, as such, not all values of $\boldsymbol{R}$ are possible—thereby violating the *Universality* condition.

As directly verifying *Universality* is intractable, we use *diversity* as an approximate. *Diversity* quantifies the degree of alignment or discordance between rankings of different tasks over $\mathcal{M}$. A lower *diversity* score indicates a strong inter-task correlation with similar rankings being produced across tasks, which could potentially impair *Universality*, as it restricts the possible values that $\boldsymbol{R}$ can take. For example, *diversity* being zero means that all tasks are the same, and *Universality* will be violated. Conversely, a higher *diversity* represents a stronger disagreement between tasks regarding model rankings, paving the way for more possible $\boldsymbol{R}$ scenarios and thus aligning more closely with the tenet of *Universality*. For example, *diversity* being one means that the rankings of all tasks are random, and thus *Universality* will hold.

4.2 Sensitivity in Multi-Task Benchmarks

Sensitivity is based on the desideratum from Arrow's theorem about the independence of irrelevant changes, restated below.

Property 4.1 (Independence of Irrelevant Changes). The aggregated final ranking should not be altered by irrelevant changes on tasks that do not modify the relative performance of models.

Intuitively speaking, our measure of *sensitivity* captures the degree to which a benchmark responds to irrelevant changes. In particular, high *sensitivity* implies that the desideratum of independence of irrelevant changes is strongly violated.

The definition of *sensitivity* is different in the case of ordinal and cardinal benchmarks. Both definitions make use of Kendall's τ coefficient that we define next. For any model list $\mathcal{L}$, Kendall's τ coefficient measures the distance between any two model rankings $\boldsymbol{r}$ and $\boldsymbol{r}'$, as follows,

$$\tau = \frac{\text{number of disconcordant pairs}}{\binom{m}{2}}, \tag{4}$$

where a pair of models L_i and L_j is said to be concordant in $\boldsymbol{r}$ and $\boldsymbol{r}'$ if both $r_i > r_j$ and $r'_i > r'_j$ hold or both $r_i < r_j$ and $r'_i < r'_j$ hold; otherwise, this pair is considered as disconcordant. One intuitive explanation for the number of disconcordant pairs is to count the number of times one has to cross lines when connecting matching data points from one ranking to another. Here we have normalized τ into $[0, 1]$, so that $\tau = 0$ means that the two ranks are exactly the same, while $\tau = 1$ means they are opposite to each other. We primarily use τ as the measurement for ranking distance in our formulation, but we also report max rank change (*MRC*) to provide a more intuitive measurement for the ranking distance in our experiments. For any two model

rankings $\boldsymbol{r}$ and $\boldsymbol{r}'$, *MRC* is defined as follows,

$$MRC = \max_{i \in \{1,2,\ldots,m\}} \frac{|r_i - r'_i|}{m-1}. \tag{5}$$

$MRC = 0$ means there is no ranking change, while $MRC = 1$ indicates the maximum possible fluctuation in rankings. Next, we will define two kinds of irrelevant changes for cardinal and ordinal benchmarks, respectively.

Sensitivity in cardinal benchmarks. For cardinal benchmarks, the *sensitivity* is defined based on *label noise injection* on tasks as the irrelevant change. Specifically, the injection of label noise to a task could well preserve the relative performances of models and should not change a task's intrinsic nature. Therefore, *sensitivity* aims to quantify the robustness of benchmark rankings to these task-equivalent manipulations in scores. This concept is also loosely analogous to the *Non-dictatorship* principle, which prohibits any single voter (in this case, a task and its scoring) from imposing an undue influence on the outcome. In the most extreme scenario, randomizing all labels for a particular task is equivalent to excluding that task from the benchmark. Changes in the aggregated ranking brought by such manipulation can thus reveal the level of influence that the task has. More significant fluctuations imply a greater impact of the task on overall rankings, suggesting that it plays a vital role in the benchmark, while minimal changes suggest that the task's influence is negligible.

Specifically, let $\mathcal{L} = (L_1, L_2, \ldots, L_m)$ contain all models in $\mathcal{M}$ without duplicates, *i.e.*, $m = |\mathcal{M}|$, we define *sensitivity* by injecting different portions of label noise in each task, and calculating the largest ranking distance after injection, as follows,

$$\max_{\boldsymbol{\alpha} \in [\epsilon, 1]^n} \quad \tau(\boldsymbol{r}^{\mathrm{c}}, \boldsymbol{r}') \tag{6}$$

$$\text{s.t.} \quad \boldsymbol{r}' = \textsc{rankdata}\big((\bar{s}'_1, \bar{s}'_2, \ldots, \bar{s}'_m)\big), \tag{7}$$

$$\bar{s}'_i = \sum_{j=1}^{n} \big(\alpha_j s_{ij} + (1-\alpha_j) p_j\big), \tag{8}$$

where the original ranking $\boldsymbol{r}^{\mathrm{c}}$ and scores s_{ij} are defined in Section 3. $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ defines the proportions of preserved examples whose labels are unchanged for each task. The label noise are injected by randomly substituting $(1-\alpha_j)$ portion of examples' labels into a random one. As a result, the corresponding score for these examples would be p_j, which refers to the performance under random label substitution. For example, if the task T_j is binary classification and its score refers to accuracy, the p_j refers to $1/2$. In practice, the specific value of p_j does not have any impact on the ranking $\boldsymbol{r}'$ as the sum $\sum_{j=1}^{n}(1-\alpha_j)p_j$ does not depend on models. $\epsilon \in (0,1)$ is a hyper-parameter that defines the minimal preserving proportion of unchanged examples for each task. It is worth noting that $\boldsymbol{r}'$ will not change if we multiply a positive constant with $\boldsymbol{\alpha}$, so we could always keep the maximum value in $\boldsymbol{\alpha}$ as one by multiplying $\boldsymbol{\alpha}$ with $1/max(\{\alpha_i\}_{i=1}^{n})$, which means that there is always at least one task with no noise.

Sensitivity in ordinal benchmarks. For ordinal benchmarks, the definition for *sensitivity* is built upon *irrelevant model addition* as the irrelevant change. This is inspired by the *IIA* condition in

Arrow's Impossibility Theorem, which demands that the addition of a new model (an "irrelevant alternative" with respect to existing comparisons) should not change the relative ranking order of the models already being considered. If a benchmark's model rankings are dramatically affected every time a new model is introduced into the competition, it indicates a high *sensitivity*. A low *sensitivity* in ordinal benchmarks assures us that the relative rankings are stable and that the benchmark can handle the introduction of new models without disrupting the existing rankings, consistent with the IIA condition. This resilience is essential, as it means that the benchmark's evaluations are reliable and reflective of each model's true performance relative to its peers.

Specifically, let $\mathcal{L} = (L_1, L_2, \ldots, L_m)$ be a list of models, and $\mathcal{L}^{\mathrm{C}} = (L_{m+1}, L_{m+2}, \ldots, L_{m+l})$ represent the complement model list, *i.e.*, $m + l = |\mathcal{M}|$. Then the *sensitivity* for ordinal benchmarks is defined as the largest ranking distance after adding a subset of these extra candidate models into comparison.

$$\max_{\boldsymbol{\beta} \in \{0,1\}^l} \tau(\boldsymbol{r}^{\mathrm{o}}, \boldsymbol{r}') \tag{9}$$

$$\text{s.t. } \boldsymbol{r}' = \textsc{rankdata}\big((\bar{w}'_1, \bar{w}'_2, \ldots, \bar{w}'_m)\big)\,, \tag{10}$$

$$\bar{w}'_i = \frac{1}{m + ||\boldsymbol{\beta}||_1} \sum_{j=1}^{m} w_{ij} + \sum_{j=1}^{l} \beta_j w_{i(m+j)}\,, \tag{11}$$

where we use $\boldsymbol{\beta} \in \{0,1\}^l$ as the indicator irrelevant model selection, where $\beta_j = 1$ means M_{m+j} is selected and $\beta_j = 0$ means not-selected. As a result, $||\boldsymbol{\beta}||_1$ refers to the number of selected models from $\mathcal{L}^{\mathrm{C}}$ as irrelevant models.

Assuming *IIA* in Theorem 3.1 holds, for any model list $\mathcal{L}$, after appending a list of irrelevant models (as indicated by $\boldsymbol{\beta}$), the relative ranking among models in $\mathcal{L}$ should not change. In practice, we simply calculate *sensitivity* by selecting the top-20% models in the existing benchmark as $\mathcal{L}$ and the rest 80% models as $\mathcal{L}^{\mathrm{C}}$. If *IIA* holds, then *sensitivity* should be zero. On the other hand, we note that our *sensitivity* is a lower bound for *IIA*, which says that *IIA* could still not hold even if *sensitivity* is zero. This limit mainly comes from the setting where we only consider the top-20% models, which is inspired by real-world scenarios where most people only care about top models. Note that, in our paper, we only focus on ordinal benchmarks that aggregate the final ranking by calculating the winning rate as in equation 2, which satisfies *Non-Dictatorship* and *Pareto Efficiency* by design. For ordinal benchmarks with other aggregation methods, one should take all *Non-Dictatorship*, *Pareto Efficiency*, and *IIA* into consideration for *sensitivity*.

Relaxation of sensitivity. The main challenge for solving equation 6 and equation 9 lies in the non-differentiable nature of the operator $\textsc{rankdata}(\cdot)$ and $\tau(\cdot, \cdot)$. Thus we propose to relax the ranking distance to a continuous objective as follows,

$$\ell^c = \sum_{i=1}^{m} \sum_{j=1}^{m} \left(\mathcal{I}(r_i^{\mathrm{c}} < r_j^{\mathrm{c}}) \max(\bar{s}'_i - \bar{s}'_j, -\lambda) \right) \tag{12}$$

$$\ell^o = \sum_{i=1}^{m} \sum_{j=1}^{m} \left(\mathcal{I}(r_i^{\mathrm{o}} < r_j^{\mathrm{o}}) \max(\bar{w}'_i - \bar{w}'_j, -\lambda) \right) \tag{13}$$

where $\mathcal{I}(\cdot)$ is the indicator function, and $\lambda \geq 0$ is a hyperparameter. If the optimal point of equation 12 could be achieved, for any $r_i^{\mathrm{c}} < r_j^{\mathrm{c}}$, we have $r'_i > r'_j$ because $s'_i < s'_j$. As a result, the

Algorithm 1 Sensitivity for Cardinal Benchmarks

1: **Input:** scores $\{s_{ij}\}_{i\in[1,m],j\in[1,n]}$ for models in $\mathcal{L}$, ϵ, λ, number of optimization T
2: Calculate $\boldsymbol{r}^{\mathrm{c}}$ based on equation 1
3: Initialize the parameter $\boldsymbol{\theta} \in R^n$ randomly
4: **for** $t = 1$ **to** T **do**
5: $\quad \boldsymbol{\alpha} = Sigmoid(\boldsymbol{\theta}) + \epsilon/(1-\epsilon)$
6: $\quad \boldsymbol{\alpha} = \alpha/||\alpha||_1$
7: $\quad$ Calculate updated score $\{\bar{s}_i'\}_{i=1}^m$ as equation 8
8: $\quad$ Calculate relaxed loss ℓ^{c} based on equation 12
9: $\quad$ Gradient descent on $\boldsymbol{\theta}$ based on ℓ^{c}
10: **end for**
11: $\boldsymbol{\alpha} = Sigmoid(\boldsymbol{\theta}) + \epsilon/(1-\epsilon)$
12: $\boldsymbol{\alpha} = \alpha/max(\boldsymbol{\alpha})$
13: Calculate $\boldsymbol{r}'$ based on equation 7
14: **Output:** $\tau(\boldsymbol{r}^{\mathrm{c}}, \boldsymbol{r}')$

Algorithm 2 Sensitivity for Ordinal Benchmarks

1: **Input:** winning rates $\{w_{ij}\}_{i\in[1,m+l],j\in[1,m+l]}$ for $\mathcal{L}$ and $\mathcal{L}^{\mathrm{C}}$, λ, number of optimization T
2: Calculate $\boldsymbol{r}^{\mathrm{o}}$ based on equation 2
3: Initialize the parameter $\boldsymbol{\theta} \in R^n$ randomly
4: **for** $t = 1$ **to** T **do**
5: $\quad \boldsymbol{q}_{\boldsymbol{\beta}} = Sigmoid(\boldsymbol{\theta})$
6: $\quad \boldsymbol{\beta} \sim Bernoulli(\boldsymbol{q}_{\boldsymbol{\beta}})$
7: $\quad \boldsymbol{\beta} = \boldsymbol{\beta} + \boldsymbol{q}_{\boldsymbol{\beta}} - \boldsymbol{q}_{\boldsymbol{\beta}}.detach()$
8: $\quad$ Calculate updated win rate $\{\bar{w}_i'\}_{i=1}^m$ as equation 11
9: $\quad$ Calculate relaxed loss ℓ^{o} based on equation 13
10: $\quad$ Gradient descent on $\boldsymbol{\theta}$ based on ℓ^{c}
11: **end for**
12: $\boldsymbol{\beta} = (Sigmoid(\boldsymbol{\theta}) > 0.5).int()$
13: Calculate $\boldsymbol{r}'$ based on equation 10
14: **Output:** $\tau(\boldsymbol{r}^{\mathrm{o}}, \boldsymbol{r}')$

original objectives in equation 6 would also achieve the optimal solution as $\tau(\boldsymbol{r}^{\mathrm{c}}, \boldsymbol{r}') = 1$ based on equation 4. The same applies to ordinal benchmarks with equation 9 and 13.

The algorithms for calculating *sensitivity* for cardinal and ordinal benchmarks could be seen in Algorithm 1 and 2. For cardinal *sensitivity* calculation in Algorithm 1, we normalize the sum of $\boldsymbol{\alpha}$ as one in line 5-6 during optimization, or otherwise the loss could be minimized by setting $\boldsymbol{\alpha} = \mathbf{0}$. For ordinal *sensitivity* calculation in the algorithm 2, in order to handle the optimization challenge brought by the discrete nature of $\boldsymbol{\beta}$, we sample it from a Bernoulli distribution with probability $\boldsymbol{q}_{\boldsymbol{\beta}}$ modeled by $\boldsymbol{\theta}$ as shown in line 6. The straight through technique (Jang et al., 2016; Bengio et al., 2013) is used to handle the gradients on $\boldsymbol{\theta}$. Due to the potential approximation errors and optimization challenges, the calculated ranking distances by both algorithms are the lower bound of the true values.

5 Experiments on Cardinal Benchmarks

In this section, we present the *diversity* and *sensitivity* to label noise injection for seven cardinal benchmarks.

Experiment setup For our experiment, we have collected seven widely-used benchmarks for our experiments, `GLUE` (Wang et al., 2018), `SuperGLUE` (Wang et al., 2019), `MTEB` (Muennighoff et al., 2022), `BigBenchHard` (Suzgun et al., 2022), `MMLU` (Hendrycks et al., 2020), `OpenLLM` (Beeching et al., 2023; Gao et al., 2021) and `VTAB` (Zhai et al., 2019). To provide a better understanding of the *diversity* and *sensitivity* spectrums, we further introduce three additional "baseline" benchmarks,

Table 1: Summary of Benchmarks

Type	Benchmark	No. of Tasks	No. of Models
Cardinal	GLUE	9	87
	SuperGLUE	8	28
	BIG-Bench-Hard	27	107
	MTEB	56	83
	OpenLLM	6	100
	MMLU	57	100
	VTAB	19	16
	ImageNet	20	112
	Random	100	100
	Constant	100	100
Ordinal	BigCode	3	41
	HELM-accuracy	16	67
	HELM-fairness	14	67
	HELM-robustness	14	67
	HEIM-alignment-auto	40	26
	HEIM-quality-auto	12	26
	HEIM-aesthetics-auto	60	26
	HEIM-alignment-human	23	26
	HEIM-nudity	20	26
	HEIM-quality-human	7	26
	HEIM-aesthetics-human	18	26
	Random	100	1000
	Constant	100	1000

Constant, Random and ImageNet. The Constant benchmark features a single task where the scores for 100 different models are randomly determined, and the task has been duplicated 100 times. The Random benchmark assigns random scores to all 100 models across all 100 tasks. The ImageNet benchmark is based on the validation set of the ILSVRC-2012 challenge (Deng et al., 2009). We divide its 1,000 classes into 20 equally-sized subsets at random, with each subset functioning as a distinct task. We conducted evaluations on 112 models that had been pretrained on ImageNet and were sourced from the TorchVision (maintainers and contributors, 2016). The average performance across these 20 tasks corresponds to the original accuracy metric, thus ensuring that the final rankings are consistent with those derived from the original accuracy measures. More details of all benchmarks are in Table 1 and Appendix A.

Diversity and *sensitivity* scores are computed for each benchmark based on equation 3 and Algorithm 1. For both measures, all models are used for calculation, *i.e.*, $\mathcal{L}$ contains all models in the leaderboard. The only exceptions are OpenLLM and MTEB, where we focus on the top-100 models out of thousands of candidates to mitigate the influence of less reliable ones. For the *sensitivity* calculation in each benchmark, we set minimal preserving portion $\epsilon = \min\{0.01, \text{std}_{\min}/\text{std}_{\max}\}$, where $\text{std}_{\min}$ and $\text{std}_{\max}$ refer to the smallest and largest standard deviations across all tasks in the benchmark respectively. If all tasks have the same standard deviation, this will ensure that at least 1% of the data remains unaltered by label noise in each task. However, if there is variability in the standard deviations across tasks, ϵ will be adjusted based on the standard deviation. This

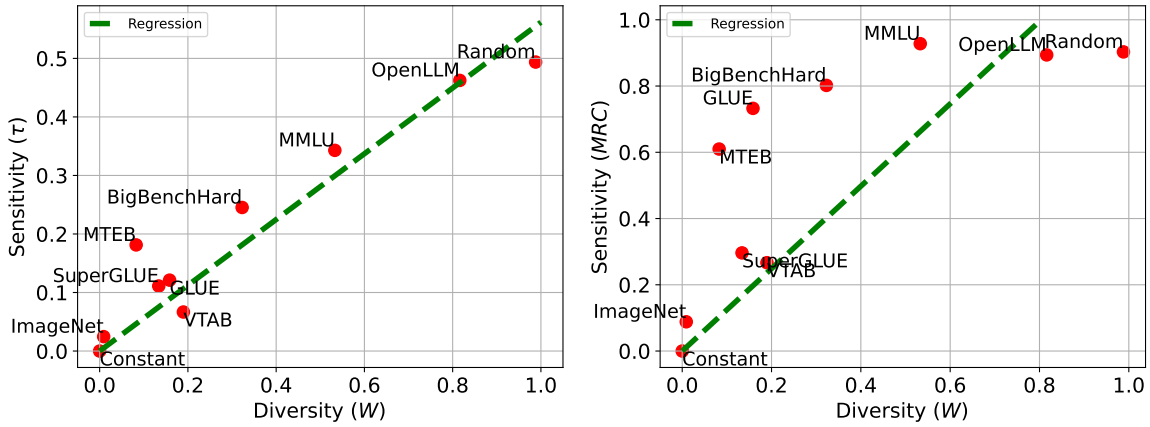

Figure 3: The x-axis indicates the *diversity* of model rankings across tasks, evaluated by the Kendall's W coefficient. The y-axis represents the *sensitivity* of the final model ranking to different portions of label noise across tasks. The ranking change is measured by both Kendall's τ (top) and MRC (bottom). The green curve is by linear regression on all points without fitting intercept.

adjustment prevents scenarios where a single task with a significantly larger standard deviation disproportionately influences the sensitivity calculation. λ is set as 0.0 and the number of gradient descent T is 1000. Results for `Constant`, `Random` and `ImageNet` are averaged over five random trials.

Experiment results The results are presented in Figure 3. A strong positive correlation between *diversity* and *sensitivity* can be observed, with Pearson correlation of 0.96 and 0.77 for top and bottom figures, respectively. A larger *diversity* always comes at the cost of high *sensitivity* to label noise injection. `Constant` is the most stable benchmark, while `Random` achieves the highest *diversity*. All real-world multi-task benchmarks roughly strike a trade-off comparable with the linear interpolation between `Random` and `Constant`.

Both *diversity* and *sensitivity* vary a lot across different benchmarks. For example, `OpenLLM` achieves the second largest *diversity* ($W = 0.82$) and suffers from a high *sensitivity* ($\tau = 0.54, MRC = 0.86$). In contrast, benchmarks like `GLUE` ($W = 0.16, \tau = 0.11, MRC = 0.72$) and `SuperGLUE` ($W = 0.13, \tau = 0.12, MRC = 0.33$) demonstrate far lower *diversity* and *sensitivity*. The underlying reason can be two-fold. First, the tasks within `GLUE` and `SuperGLUE` are more similar to each other by definition. For example, `GLUE` primarily consists of NLI and text classification tasks. In contrast, tasks within `OpenLLM` are more messy, including commonsense inference and reasoning, math problems, science questions, etc. Second, the candidate models in `OpenLLM` are also more noisy due to the relatively lower entry barrier. In contrast, there are a lot of restrictions for participant models to get presented in the leaderboard in `GLUE` and `SuperGLUE`, and thus enjoy a lower chance of having outlier candidate models in the leaderboard.

The results of `ImageNet` serve as a sanity check of our choice for the minimal preserving portion, denoted by ϵ. `ImageNet` has been one of the most influential single-task benchmarks in the field of machine learning, and its evaluation results have been widely regarded as a solid measure of progress in model development (Dwork et al., 2015; Tsipras et al., 2020; Koch et al.,

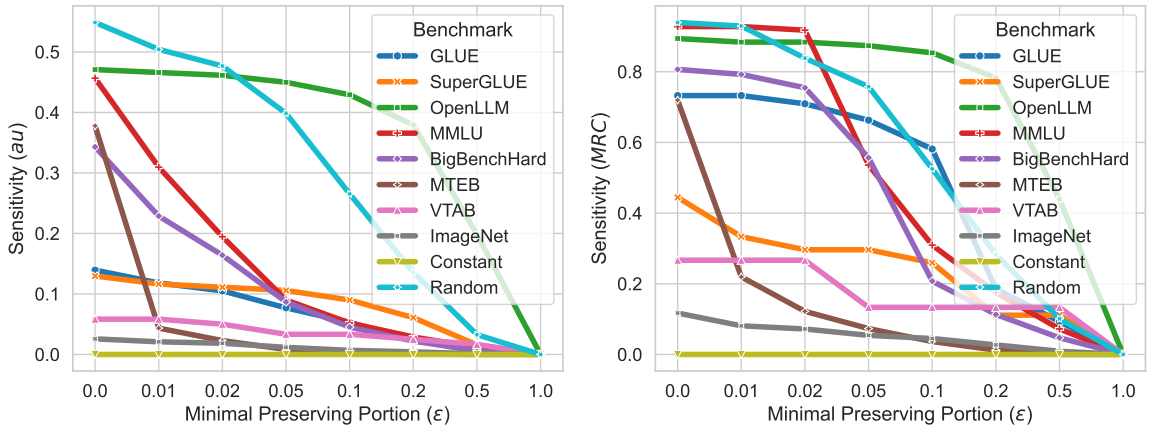

Figure 4: Sensitivity of cardinal benchmarks as a function of the minimal preserving ratio ϵ. x-axis refers to the minimal preserving portion of unchanged examples, ϵ, as stated in equation 6. The y-axis refers to *sensitivity* measured by τ (top) and *MRC* (bottom).

2021). Despite being split into 20 tasks, our `ImageNet` essentially parallels the original single-task benchmark in terms of model rankings. The experiment results show that `ImageNet` achieves the second lowest *sensitivity*, only slightly higher than `Constant`. It demonstrates that rankings of a high-quality benchmark remain stable even when subjected to significant label noise. Such robustness emphasizes the importance of resisting noise interference for benchmarks and validates our choice for ϵ.

To delve deeper into benchmark *sensitivity* concerning the minimal preserving portion ϵ, Figure 4 plots the *sensitivity* across varying ϵ values. When preserving 10% of the data ($\epsilon = 0.1$), the *MRC* for all non-baseline benchmarks ranges from 0.18 to 0.71, indicating a non-trivial ranking change. Notably, `OpenLLM` maintains a τ of 0.13 and *MRC* of 0.45 even at $\epsilon = 0.5$, underscoring its pronounced volatility.

6 Experiments on Ordinal Benchmarks

In this section, we present *diversity* and *sensitivity* to irrelevant models over eleven ordinal benchmarks.

Experiment setup Our selected benchmarks for experiments consist of `BigCode` (Ben Allal et al., 2022), three benchmarks from `HELM` (Liang et al., 2023), and seven benchmarks from `HEIM` (Lee et al., 2023). The original rankings for all these benchmarks are based on the winning rate, as defined in equation 2. We excluded any benchmarks that suffered from a lot of missing values or that showcased an undifferentiated scoring pattern among different models as these complicate the calculation of the winning rate. The remaining benchmarks are `HELM-accuracy`, `HELM-fairness`, `HELM-robustness`, `HEIM-alignment-auto`, `HEIM-quality-auto`, `HEIM-aesthetics-auto`, `HEIM-alignment-human`, `HEIM-nudity`, `HEIM-quality-human`, `HEIM-aesthetics-human`. The statistics can be seen in Table 1, and more details are in Appendix A. Similar to cardinal benchmarks, we also add `Constant` and `Random` benchmarks, with 100 tasks and 1000 models for each.

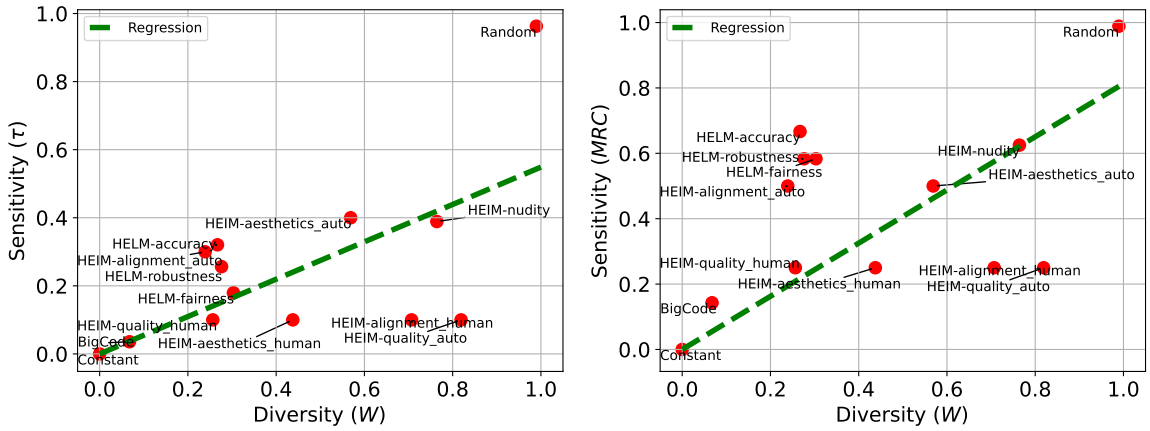

Figure 5: The x-axis indicates the *diversity* of model rankings across tasks, evaluated by the reversed Kendall's W coefficient, where $W = 0$ denotes uniformity in rankings, while $W = 1$ means random or highly varied rankings across tasks. The y-axis represents the *sensitivity* to irrelevant candidate models addition, measured by the Kendall's τ (top) and *MRC* (bottom). The green curve is by linear regression on all points without fitting intercept.

For both *diversity*, all models in each benchmark are used for calculation, *i.e.*, $\mathcal{L}$ contains all models in the leaderboard. To calculate *sensitivity*, we use the original top-20% models in the leaderboard as $\mathcal{L}$, and the ranking calculated only based on them is referred to as $\boldsymbol{r}^{\mathrm{o}}$. Then we use the rest models as $\mathcal{L}^{\mathrm{C}}$, use Algorithm 2 to select a subset from $\mathcal{L}^{\mathrm{C}}$ as irrelevant models to alter the rankings of $\boldsymbol{r}^{\mathrm{o}}$. The only exception is `Random`, where we use the top-10 models as $\mathcal{L}$ and use the rest as $\mathcal{L}^{\mathrm{C}}$, in order to simulate the scenario with infinite potential irrelevant models. λ is set as 0.01 and the number of gradient descent T is 100. For the calculation of *diversity*, we impute the missing scores with a KNN imputer as the rankings for each task must be of the same dimension for calculating Kendall's W.

Experiment results The results are shown in Figure 5, where we plot the *diversity* and *sensitivity* towards additional irrelevant alternative models. The green curve acquired by fitting all points, demonstrates that there is a strong correlation between *diversity* and *sensitivity*. The Pearson correlation is 0.61 and 0.50 for both figures. The lower Pearson correlation (compared to cardinal benchmarks) and the observed deviation from the regression curve could be attributed to the missing values in `HEIM` and `HELM`-based benchmarks. The KNN imputation method is used to impute values so that *diversity* could be calculated, but this also might lead to inaccuracies in the *diversity* estimation.

Several benchmarks exhibit significant *sensitivity*. For instance, a notable change in ranking is observed with the `HEIM-aesthetic-auto` benchmark, where the *MRC* reaches as high as 0.5 and τ reaches 0.4. Moreover, over half of these benchmarks exhibit an MRC of at least 0.5, which highlights their vulnerability to the inclusion of irrelevant models. The dependency of rankings on the selection of candidate models casts doubts on the reliability of the evaluation outcomes of these benchmarks.

One outlier in both plots is the `Random` benchmark, which is relatively far away from the

regression curve. This anomaly can be attributed to the assumption that the `Random` benchmark contemplates a nearly infinite array of irrelevant models for selection. Consequently, it allows for greater flexibility in altering the rankings of the existing models. This also suggests that, as the number of candidate models increases over time, the aggregated final rankings could be more unstable.

7 Conclusion

In this work, we examine multi-task benchmarks through the lens of social choice theory. Our exploration starts by applying Arrow's impossibility theorem on ordinal benchmarks, suggesting that there may be intrinsic limitations for multi-task benchmarks. But Arrow's theorem is neither quantitative, nor does it apply to cardinal benchmarks. We therefore develop two key measures of multi-task benchmarks—task diversity and stability to irrelevant changes—that we argue stand in tension with one another. Our empirical investigations on seven cardinal benchmarks and eleven ordinal benchmarks yield insights about the inherent trade-off between the two proposed measures. Furthermore, our analysis reveals significant sensitivity issues in several popular benchmarks, calling into question the validity of evaluation outcomes derived from these benchmarks.

8 Acknowledgement

We would like to thank Joachim Baumann, Andrè Cruz, Ricardo Dominguez-Olmedo, Florian E. Dorner, and Celestine Mendler-Dünner for helpful discussions and/or feedback on draft versions of this work.

References

Norah Alzahrani, Hisham Abdullah Alyahya, Sultan Yazeed Alnumay, Shaykhah Alsubaie, Yusef Almushaykeh, Faisal Mirza, Nouf Alotaibi, Nora Altwairesh, Areeb Alowisheq, Saiful Bari, and Haidar Khan. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. *ArXiv*, abs/2402.01781, 2024. URL https://api.semanticscholar.org/CorpusID:267412932.

Sanjeev Arora and Yi Zhang. Rip van winkle's razor: A simple estimate of overfit to test data. *ArXiv*, abs/2102.13189, 2021. URL https://api.semanticscholar.org/CorpusID:232069109.

Kenneth J Arrow. A difficulty in the concept of social welfare. *Journal of political economy*, 58(4): 328–346, 1950.

Kenneth J. Arrow. Social choice and individual values. 1951. URL https://api.semanticscholar.org/CorpusID:144910513.

Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. Open llm leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard, 2023.

Loubna Ben Allal, Niklas Muennighoff, Logesh Kumar Umapathi, Ben Lipkin, and Leandro von Werra. A framework for the evaluation of code generation models. https://github.com/bigcode-project/bigcode-evaluation-harness, 2022.

Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *ArXiv*, abs/1308.3432, 2013. URL https://api.semanticscholar.org/CorpusID:18406556.

Avrim Blum and Moritz Hardt. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*, 2015. URL https://api.semanticscholar.org/CorpusID:1493191.

Meriem Boubdir, Edward Kim, Beyza Hilal Ermiş, Sara Hooker, and Marzieh Fadaee. Elo uncovered: Robustness and best practices in language model evaluation. *ArXiv*, abs/2311.17295, 2023. URL https://api.semanticscholar.org/CorpusID:265498394.

Samuel R. Bowman and George E. Dahl. What will it take to fix benchmarking in natural language understanding? *ArXiv*, abs/2104.02145, 2021. URL https://api.semanticscholar.org/CorpusID:233033916.

Pierre Colombo, Chloe Clave, and Pablo Piantanida. Infolm: A new metric to evaluate summarization & data2text generation. In *AAAI Conference on Artificial Intelligence*, 2021. URL https://api.semanticscholar.org/CorpusID:244896426.

Pierre Colombo, Nathan Noiry, Ekhine Irurozki, and Stéphan Clémençon. What are the best systems? new perspectives on nlp benchmarking. *ArXiv*, abs/2202.03799, 2022. URL https://api.semanticscholar.org/CorpusID:246652319.

Mostafa Dehghani, Yi Tay, Alexey A. Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. The benchmark lottery. *ArXiv*, abs/2107.07002, 2021. URL https://api.semanticscholar.org/CorpusID:235810239.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. URL https://api.semanticscholar.org/CorpusID:57246310.

David Donoho. Data science at the singularity. *Issue 6.1, Winter 2024*, 2023. URL https://api.semanticscholar.org/CorpusID:263605559.

Stavros A. Drakopoulos. The historical perspective of the problem of interpersonal comparisons of utility. *Journal of Economic Studies*, 16, 1989. URL https://api.semanticscholar.org/CorpusID:55684805.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Preserving statistical validity in adaptive data analysis. *Proceedings of the forty-seventh annual ACM symposium on Theory of Computing*, 2014. URL https://api.semanticscholar.org/CorpusID:2209606.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349:636 – 638, 2015. URL https://api.semanticscholar.org/CorpusID:15569600.

Kawin Ethayarajh and Dan Jurafsky. Utility is in the eye of the user: A critique of nlp leaderboard design. In *Conference on Empirical Methods in Natural Language Processing*, 2020. URL https://api.semanticscholar.org/CorpusID:235408131.

Vitaly Feldman, Roy Frostig, and Moritz Hardt. The advantages of multiple classes for reducing overfitting from test set reuse. *ArXiv*, abs/1905.10360, 2019. URL https://api.semanticscholar.org/CorpusID:165163539.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021. URL https://doi.org/10.5281/zenodo.5371628.

Moritz Hardt and Benjamin Recht. *Patterns, predictions, and actions: Foundations of machine learning*. Princeton University Press, 2022.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Xiaodong Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *ArXiv*, abs/2009.03300, 2020. URL https://api.semanticscholar.org/CorpusID:221516475.

Anas Himmi, Ekhine Irurozki, Nathan Noiry, Stéphan Clémençon, and Pierre Colombo. Towards more robust nlp system evaluation: Handling missing scores in benchmarks. *ArXiv*, abs/2305.10284, 2023. URL https://api.semanticscholar.org/CorpusID:258741244.

Eric Jang, Shixiang Shane Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *ArXiv*, abs/1611.01144, 2016. URL https://api.semanticscholar.org/CorpusID:2428314.

Jerry S Kelly. *Social choice theory: An introduction*. Springer Science & Business Media, 1988.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Talat, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in nlp. *ArXiv*, abs/2104.14337, 2021. URL https://api.semanticscholar.org/CorpusID:233444226.

Bernard Koch, Emily L. Denton, A. Hanna, and Jacob Gates Foster. Reduced, reused and recycled: The life of a dataset in machine learning research. *ArXiv*, abs/2112.01716, 2021. URL https://api.semanticscholar.org/CorpusID:244894836.

Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Teufel, Marco Bellagente, Minguk Kang, Taesung Park, Jure Leskovec, Jun-Yan Zhu, Fei-Fei Li, Jiajun Wu, Stefano Ermon, and Percy Liang. Holistic evaluation of text-to-image models. *ArXiv*, abs/2311.04287, 2023. URL https://api.semanticscholar.org/CorpusID:265051037.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *Annals of the New York Academy of Sciences*, 1525:140 – 146, 2023. URL https://api.semanticscholar.org/CorpusID:253553585.

Thomas Liao, Rohan Taori, Inioluwa Deborah Raji, and Ludwig Schmidt. Are we learning yet? a meta review of evaluation failures across machine learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. *ArXiv*, abs/2203.08242, 2022. URL https://api.semanticscholar.org/CorpusID:247475929.

TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016.

Horia Mania, John Miller, Ludwig Schmidt, Moritz Hardt, and Benjamin Recht. Model similarity mitigates test set overuse. *ArXiv*, abs/1905.12580, 2019. URL https://api.semanticscholar.org/CorpusID:168169971.

Swaroop Mishra and Anjana Arunkumar. How robust are model rankings : A leaderboard customization approach for equitable evaluation. In *AAAI Conference on Artificial Intelligence*, 2021. URL https://api.semanticscholar.org/CorpusID:235363537.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022. doi: 10.48550/ARXIV.2210.07316. URL https://arxiv.org/abs/2210.07316.

OpenAI. Gpt-4 technical report. *arXiv*, 2023. URL http://arxiv.org/abs/2303.08774.

Simon Ott, Adriano Barbosa-Silva, Kathrin Blagec, Janina Brauner, and Matthias Samwald. Mapping global dynamics of benchmark creation and saturation in artificial intelligence. *Nature Communications*, 13, 2022. URL https://api.semanticscholar.org/CorpusID:247318891.

Maxime Peyrard, Teresa Botschen, and Iryna Gurevych. Learning to score system summaries for better content selection evaluation. In Lu Wang, Jackie Chi Kit Cheung, Giuseppe Carenini, and Fei Liu, editors, *Proceedings of the Workshop on New Frontiers in Summarization*, pages 74–84, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4510. URL https://aclanthology.org/W17-4510.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *ArXiv*, abs/2102.12092, 2021. URL https://api.semanticscholar.org/CorpusID:232035663.

Manley Roberts, Himanshu Thakur, Christine Herlihy, Colin White, and Samuel Dooley. Data contamination through the lens of time. *ArXiv*, abs/2310.10628, 2023. URL https://api.semanticscholar.org/CorpusID:264172693.

Mark Rofin, Vladislav Mikhailov, Mikhail Florinskiy, Andrey Kravchenko, E. Tutubalina, Tatiana Shavrina, Daniel Karabekyan, and E. Artemova. Vote'n'rank: Revision of benchmarking with social choice theory. *ArXiv*, abs/2210.05769, 2022. URL https://api.semanticscholar.org/CorpusID:252846467.

Amartya Sen. Collective choice and social welfare. 2017. URL https://api.semanticscholar.org/CorpusID:154085126.

Stewart Shapiro and Geoffrey Hellman. Mathematics without numbers. *Noûs*, 27:522, 1993. URL https://api.semanticscholar.org/CorpusID:170189790.

Ali Shirali, Rediet Abebe, and Moritz Hardt. A theory of dynamic benchmarks. In *International Conference on Learning Representations (ICLR)*, 2023.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *ArXiv*, abs/2206.04615, 2022. URL https://api.semanticscholar.org/CorpusID:263625818.

Shane Storks, Qiaozi Gao, and Joyce Yue Chai. Recent advances in natural language inference: A survey of benchmarks, resources, and approaches. *arXiv: Computation and Language*, 2019. URL https://api.semanticscholar.org/CorpusID:213613608.

Mirac Suzgun, Nathan Scales, Nathanael Scharli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed Huai hsin Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. In *Annual Meeting of the Association for Computational Linguistics*, 2022. URL https://api.semanticscholar.org/CorpusID:252917648.

Alan D Taylor. *Social choice and the mathematics of manipulation*. Cambridge University Press, 2005.

Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv*, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971, 2023a. URL https://api.semanticscholar.org/CorpusID:257219404.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288, 2023b. URL https://api.semanticscholar.org/CorpusID:259950998.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *International Conference on Machine Learning*, 2020. URL https://api.semanticscholar.org/CorpusID:218862858.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*, 2018. URL https://api.semanticscholar.org/CorpusID:5034059.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *ArXiv*, abs/1905.00537, 2019. URL https://api.semanticscholar.org/CorpusID:143424870.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, André Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, Lucas Beyer, Olivier Bachem, Michael Tschannen, Marcin Michalski, Olivier Bousquet, Sylvain Gelly, and Neil Houlsby. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv: Computer Vision and Pattern Recognition*, 2019. URL https://api.semanticscholar.org/CorpusID:214317405.

J Zhang, Mark Harman, Lei Ma, and Yang Liu. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering*, 48:1–36, 2019. URL https://api.semanticscholar.org/CorpusID:195657970.

Yue Zhang, Ming Zhang, Haipeng Yuan, Shichun Liu, Yongyao Shi, Tao Gui, Qi Zhang, and Xuanjing Huang. Llmeval: A preliminary study on how to evaluate large language models. In *AAAI Conference on Artificial Intelligence*, 2023. URL https://api.semanticscholar.org/CorpusID:266174168.

A Benchmark Details

We use the following benchmarks for our experiment. The cardinal benchmarks are as follows,

- The GLUE benchmark is designed to evaluate natural language understanding models using 9 tasks that cover fundamental linguistic abilities such as sentiment analysis, entailment, and similarity prediction. There are 87 candidate models. The leaderboard can be found in https://gluebenchmark.com/leaderboard.
- SuperGLUE, as an extension of GLUE, consists of more demanding tasks aimed at assessing deeper linguistic comprehension and commonsense reasoning, spanning 8 tasks. There are 28 candidate models. The leaderboard can be found in https://super.gluebenchmark.com/leaderboard.
- BIG-Bench-Hard, a subset of the larger BIG-Bench, zeroes in on 27 specifically challenging tasks to test models on complex reasoning and understanding nuanced language. There are 107 candidate models. The leaderboard is found in https://opencompass.org.cn/dataset-detail/BBH.
- MTEB is designed to extensively evaluate text embeddings, including 56 datasets across 7 different tasks and covering 112 languages to seek a universal text embedding method. There are 83 candidate models. The leaderboard is found in https://huggingface.co/spaces/mteb/leaderboard. As the original leaderboard reports the weighted average based on the number of datasets within each task, we simply repeat each task correspondingly in our experiment and use direct averaging.
- OpenLLM leaderboard evaluates open language models with 6 key benchmarks applied in the EleutherAI Language Model Evaluation Harness, involving various tasks related to reasoning, general knowledge, and truthfulness in both zero-shot and few-shot frameworks. Only the top 100 candidate models are used for our experiment. The leaderboard can be found in https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.
- The MMLU benchmark offers a large-scale, multidisciplinary evaluation with a focus on academic knowledge, including 57 different subjects. Only the top 100 candidate models are used for our experiment. The leaderboard can be found in https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.
- VTAB, short for Visual Task Adaptation Benchmark, is a suite designed to evaluate the versatility and generalizability of visual representations by measuring performance across 19 diverse classification tasks without using evaluation datasets during pre-training. There are 16 candidate models. The leaderboard can be found in https://google-research.github.io/task_adaptation/benchmark.

The ordinal benchmarks are as follows,

- BigCode is designed to test the abilities of code generation models by posing complex coding challenges in 3 programming languages. There are 41 models in the benchmark. The leaderboard can be found in https://huggingface.co/spaces/bigcode/bigcode-models-leaderboard.
- HELM evaluates language models across 8 scenarios, where each scenario corresponds to a benchmark with multiple tasks. We excluded any benchmarks that suffered from a lot

of missing values or that showcased an undifferentiated scoring pattern among different models, as these would result in tied results frequently, which subsequently complicates the calculation of winning rates. Three benchmarks have remained: `HELM-accuracy` (16 tasks), `HELM-fairness` (14 tasks), and `HELM-robustness` (14 tasks). There are 67 candidate models in each benchmark. The leaderboard can be found in https://crfm.stanford.edu/helm/lite/latest/#/leaderboard.

- `HEIM` is tailored to scrutinize the performance of text-to-image models across ten dimensions such as creativity, equity, and language coverage, each of which forms a benchmark with multiple tasks. We excluded any benchmarks that suffered from a lot of missing values or that showcased an undifferentiated scoring pattern among different models, as these would result in tied results frequently, which subsequently complicates the calculation of winning rates. Seven benchmarks have remained: `HEIM-alignment-auto` (40 tasks), `HEIM-quality-auto` (12 tasks), `HEIM-aesthetics-auto` (60 tasks), `HEIM-alignment-human` (23 tasks), `HEIM-nudity` (20 tasks), `HEIM-quality-human` (7 tasks), `HEIM-aesthetics-human` (18 tasks). There are 26 candidate models in each benchmark. The leaderboard can be found in https://crfm.stanford.edu/heim/latest/?group=core_scenarios.

Each of these benchmarks collectively aims to provide a comprehensive platform to test the limits and versatility of machine learning models from multiple aspects. The statistics could be seen in Table 1.

B Proof of Arrow's Impossibility Theorem for Benchmarks

We include a proof of Arrow's result in our notation for the sake of completeness.

Notation. We first restate the notation, as follows:

- $\mathcal{T} = (T_1, T_2, \ldots, T_n)$ represents the list of all n tasks in the benchmark, analogous to voters.
- $\mathcal{M}$ refers to the set of all potential candidate models that could be evaluated by the benchmark.
- Let $\mathcal{L} = (L_1, L_2, \ldots, L_m)$ be any non-empty list of candidate models with m models, where $L_i \in \mathcal{M}$ for any i.
- For any $\mathcal{L}$, we define s_{ij} as the score for the i-th model in $\mathcal{L}$ in task T_j. For simplicity, we abuse the notations and use $\boldsymbol{s}_j = (s_{1j}, s_{2j}, \ldots, s_{mj})$ as scores in any task T_j, and $\boldsymbol{S} = (\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n)$ as scores over all tasks.
- For any $\mathcal{L}$, we define r_{ij} as the rank for the i-th model $\mathcal{L}$ in task T_j *w.r.t.* $\mathcal{L}$. For simplicity, we abuse the notations and use $\boldsymbol{r}_j = (r_{1j}, r_{2j}, \ldots, r_{mj})$ as ranks in any task T_j, and $\boldsymbol{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_n)$ as ranks over all tasks.
- A cardinal benchmark is defined as a function $f^{\mathrm{c}} = h^{\mathrm{c}} \circ g^{\mathrm{c}}$, which is composed of the scoring function g^{c} and the aggregation function h^{c}. Specifically, g^{c} takes a list of models $\mathcal{L}$ as input and outputs the corresponding scores for each index over all tasks, *i.e.*, $\boldsymbol{S} = g^{\mathrm{c}}(\mathcal{L})$. The scores $\boldsymbol{S}$ are fed into h^{c}, which outputs the final ranking $\boldsymbol{r}^{\mathrm{c}} = (r_1^{\mathrm{c}}, r_2^{\mathrm{c}}, \ldots, r_m^{\mathrm{c}})$, *i.e.*, $\boldsymbol{r}^{\mathrm{c}} = h^{\mathrm{c}}(\boldsymbol{S})$.
- An ordinal benchmark is defined as a function $f^{\mathrm{o}} = h^{\mathrm{o}} \circ g^{\mathrm{o}}$, which is composed of the scoring function g^{o} and the aggregation function h^{o}. Specifically, g^{o} takes a list of models $\mathcal{L}$ as input and outputs the corresponding rankings for each index over all tasks, *i.e.*, $\boldsymbol{R} = g^{\mathrm{o}}(\mathcal{L})$. The rankings $\boldsymbol{R}$ are fed into h^{o}, which outputs the final ranking $\boldsymbol{r}^{\mathrm{o}} = (r_1^{\mathrm{o}}, r_2^{\mathrm{o}}, \ldots, r_m^{\mathrm{o}})$, *i.e.*, $\boldsymbol{r}^{\mathrm{o}} = h^{\mathrm{o}}(\boldsymbol{R})$.
- We use $\textsc{rankdata}(\cdot)$ as the operator of getting rank.

Arrow's Impossibility Theorem for Benchmarks We present Arrow's Impossibility Theorem for benchmarks as follows,

Theorem B.1 (Arrow's Impossibility Theorem for Benchmarks)**.** *No ordinal benchmark f^o can fulfill the following conditions simultaneously:*

1. ***Non-Dictatorship:*** *There is no task T_i such that, for any $\mathcal{L}$ and any index pair (x, y) , when $r_{xi} < r_{yi}$, then $r_x^o < r_y^o$.*
2. ***Pareto Efficiency:*** *For any L and any index pair (x, y), if $r_{xi} < r_{yi}$ for every task $T_i \in \mathcal{T}$, then $r_x^o < r_y^o$.*
3. ***Independence of Irrelevant Alternatives (IIA):*** *Let $\mathcal{L}$ and $\mathcal{L}'$ be any two lists of models. For any index pair (x, y), if x and y have the same relative order in $g^o(\mathcal{L})$ and $g^o(\mathcal{L}')$ for all tasks, then x and y have the same relative order in $f^o(\mathcal{L})$ and $f^o(\mathcal{L}')$.*
4. ***Universality:*** *The benchmark has at least three tasks. The benchmark has as domain all finite lists with at least three models. The scoring function g^o has full range over all logically possible values for*

R. *The aggregation function h^o has full domain over all logically possible values for* ***R***.

Supporting Lemmas To prove the Theorem B.1[1], we first define decisive coalitions and present two supporting lemmas:

- A subset of tasks $\mathcal{G} \subset \mathcal{T}$ is a coalition.
- A coalition $\mathcal{G}$ is decisive over an index pair (x, y) if and only if, for any $\boldsymbol{L}$, when $r_{xi} < r_{yi}$ for every $T_i \in \mathcal{G}$, then $r_x^o < r_y^o$.
- A coalition $\mathcal{G}$ is decisive if and only if it is decisive over all ordered pairs.
- A coalition $\mathcal{G}$ is decisive over an index pair (x, y) if and only if, for any $\boldsymbol{L}$, when $r_{xi} < r_{yi}$ for every $T_i \in \mathcal{G}$ and $r_{xj} > r_{yj}$ for every $T_j \in (\mathcal{T} - \mathcal{G})$, then $r_x^o < r_y^o$.

Lemma B.2 (Field Expansion Lemma)**.** *For a benchmark that satisfies* Pareto Efficiency, IIA *and* Universality*, if a coalition $\mathcal{G}$ is weakly decisive over index pair (x, y) for some $x \neq y$, then it is decisive.*

Proof. Assume $\mathcal{G}$ is weakly decisive over (x, y). Let z be any index distinct from x and y. Find a $\mathcal{L}$ such that $r_{xi} < r_{yi} < r_{zi}$ for every task $T_i \in \mathcal{G}$, and $r_{yj} < r_{xj}$ and $r_{yj} < r_{zj}$ for every task $T_j \in (\mathcal{T} - \mathcal{G})$. Note that there is no need to specify the relationship between r_{xj} and r_{zj} for $T_j \in (\mathcal{T} - \mathcal{G})$. By *Pareto Efficiency*, we have $r^o(y) < r^o(z)$. By weak decisiveness of $\mathcal{G}$ over (x, y), we have $r^o(x) < r^o(y)$. Thus we have $r^o(z) < r^o(y)$ for $\mathcal{L}$. By *IIA*, every $\mathcal{L}'$ which shares the same relative order for (x, z), *i.e.*, $r_{xi} < r_{zi}$ for every task $T_i \in \mathcal{G}$, should have $r^o(z) < r^o(y)$. Therefore, $\mathcal{G}$ is decisive over (x, z). Similarly, we could show $\mathcal{G}$ is also decisive over (y, z). Therefore, we prove that $\mathcal{G}$ is decisive for all index pairs in $\{x, y, z\}$. Iterating the above process, we could prove that $\mathcal{G}$ is decisive for all index pairs in $\{1, 2, \ldots, m\}$, and thus the proof is complete. □

Lemma B.3 (Group Contraction Lemma)**.** *For a benchmark that satisfies* Pareto Efficiency, IIA *and* Universality*, if a coalition $\mathcal{G}$ is decisive, and has at least two tasks, then it has a proper subset that is also decisive.*

Proof. Assume $\mathcal{G}$ is decisive and has at least two tasks. Partition $\mathcal{G}$ into $\mathcal{G}_1$ and $\mathcal{G}_2$. Fix distinct indices x, y, z. Find a $\mathcal{L}$ such that

$$r_{xi} < r_{yi} < r_{zi} \quad \text{if} \quad T_i \in \mathcal{G}_1 \tag{14}$$

$$r_{zi} < r_{xj} < r_{yj} \quad \text{if} \quad T_j \in \mathcal{G}_2 \tag{15}$$

$$r_{yk} < r_{zk} < r_{xk} \quad \text{if} \quad T_k \in (\mathcal{T} - \mathcal{G}) \tag{16}$$

Since $\mathcal{G}$ is decisive, we have $r_x^o < r_y^o$. So at least one is true between $r_x^o < r_z^o$ and $r_z^o < r_y^o$. If $r_x^o < r_z^o$, then $\mathcal{G}_1$ is weakly decisive over (x, z). If $r_z^o < r_y^o$, then $\mathcal{G}_2$ is weakly decisive over (z, y). Now apply the Field Expansion Lemma. By iterating the process, the lemma is proved. □

Proof of Arrow's Impossibility Theorem for Benchmarks

Proof. By *Pareto Efficiency*, the entire set of tasks $\mathcal{T}$ is decisive, thus by *Group Contraction Lemma*, there is a size-one decisive coalition — a dictator. In other words, any benchmark that satisfies *Pareto Efficiency*, *IIA* and *Universality* will violate *Non-Dictatorship*. Hence, the proof is complete. □

[1]The proof is largely the same as the original Arrow's Theorem in https://shorturl.at/bdlI0.

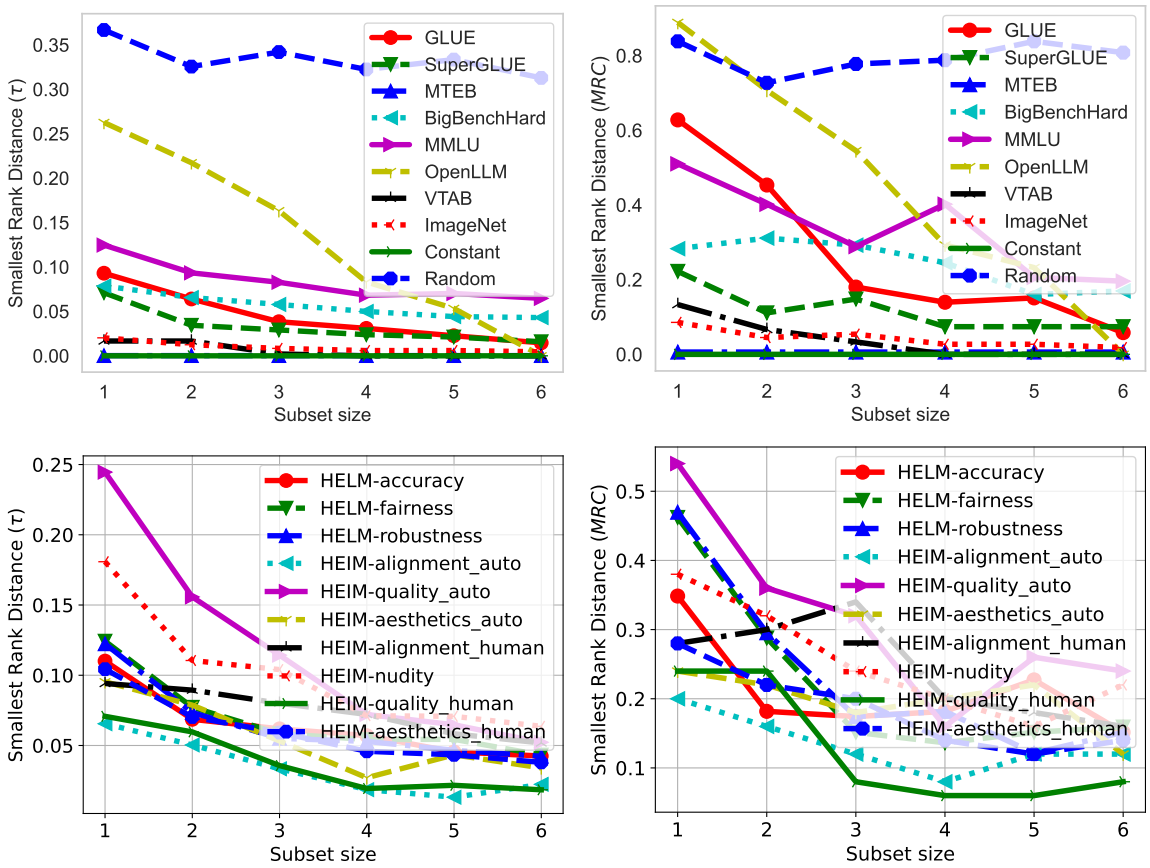

Figure 6: Smallest rank change by re-calculating the average score based on a subset of tasks. x-axis refers to how many tasks are selected in the subset. By randomly sampling 1000 random subsets of the specific size, we report the smallest ranking distance from the original ranking in y-axis, in cardinal benchmarks (top) and ordinal benchmarks (bottom), measured by Kendall's τ (left) and *MRC* (right).

C Additional Experimental Results on Diversity

We further conduct an experiment, seeking to determine the minimum number of tasks necessary to obtain an approximation of the overall final ranking. We examined subsets of tasks ranging from sizes one to six, randomly sampling these subsets for 1000 times and identifying which offered a ranking closest to the overall ranking. We exclude `BigCode` for this experiment as it only contains three tasks. The outcomes are illustrated in Figure 6. Intriguingly, our findings align with the *diversity* present within the benchmarks. For instance, `OpenLLM` and `HEIM-quality-auto`, which display the greatest *diversity* (except for `Random`) in Figure 3 and 5, also requires the largest number of tasks to arrive at a ranking proximate to the overall ranking. Conversely, benchmarks exhibiting less *diversity*, such as `VTAB` and `HEIM-quality-human`, require fewer tasks to replicate the overall ranking. This suggests that benchmarks with lower *diversity* might contain more redundant tasks that do not significantly contribute to the overall ranking.

While this experiment offers valuable insights into the connection between *diversity* and the minimum number of tasks needed to approximate the overall ranking, it is important to acknowl-

edge its limitations. The results can be influenced by the number of tasks in the benchmarks, potentially skewing the findings. For example, for a benchmark with three tasks, the maximum number of tasks to approximate the overall ranking is always three, no matter how large *diversity* of the benchmark is. To mitigate this issue, one potential approach could be to consider the ratio of tasks rather than absolute numbers. However, this will introduce another challenge where tasks could be duplicated within a benchmark to artificially reduce the minimal subset ratio required to replicate the full ranking. Despite these limitations, the experiment provides an intuitive understanding of how *diversity* correlates with the minimum subset size necessary for ranking recovery. We recognize the need for further exploration in future research endeavors.