Efficient Benchmarking (of Language Models)

Anonymous ACL submission

Abstract

The increasing versatility of language models (LMs) has given rise to a new class of benchmarks that comprehensively assess a broad range of capabilities. Such benchmarks are associated with massive computational costs, extending to thousands of GPU hours per model. However, the efficiency aspect of these evaluation efforts had raised little discussion in the literature.

In this work, we present the problem of *Efficient Benchmarking*, namely, intelligently reducing the computation costs of LM evaluation without compromising *reliability*. Using the HELM benchmark as a test case, we investigate how different benchmark design choices affect the computation-reliability trade-off. We propose to evaluate the reliability of such decisions, by using a new measure – Decision Impact on Reliability, *DIoR* for short. We find, for example, that a benchmark leader may change by merely removing a low-ranked model from the benchmark, and observe that a correct benchmark ranking can be obtained by considering only a fraction of the evaluation examples. Based on our findings, we outline a set of concrete recommendations for efficient benchmark design and utilization practices. To take a step further, we use our finding to propose an evaluation algorithm, that, when applied to the HELM benchmark, leads to dramatic cost savings with minimal loss of benchmark reliability, often reducing computation by x100 or more.[1]

1 Introduction

Given the ongoing advances in the versatility and performance of Language Models (LMs), they are now expected to perform a diverse range of tasks. This expectation raises a profound challenge – how do we evaluate and rank the quality of different LMs over a variety of capabilities?

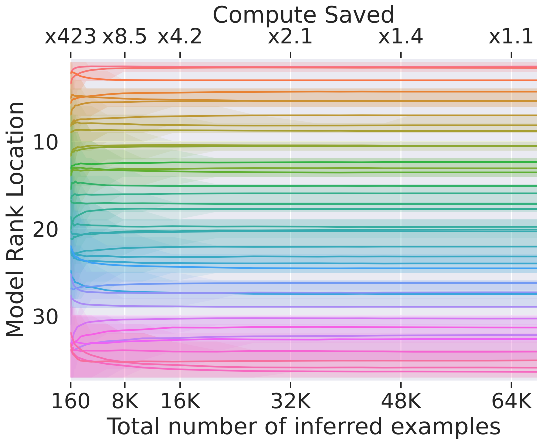


Figure 1: **HELM model ranks for different numbers of inference calls.** Each colored curve represents a different model. Model ranks are extremely stable even when compute drops dramatically: a ×10 decrease in the number of examples per scenario produces nearly the same results as the full benchmark, while a ×400 reduction still clusters models in the same small groups seen in the full compute regime.

This is a complex evaluation endeavor (Chang et al., 2023), as it transcends the boundaries of a specific task and seeks to measure the overall capabilities of an LM over a wide manifold of natural language tasks. To this end, LM benchmarks are constantly being proposed, where each new benchmark further expands the coverage and diversity of evaluated tasks and settings (Wang et al., 2018; bench authors, 2023; Gao et al., 2021; Talmor et al., 2020; Yuan et al., 2023; Zhang et al., 2023). Running such expansive benchmarks can entail spending $10K+ or 4K+ GPU hours for evaluating a single model (Liang et al., 2022), and may even surpass those of pretraining (Biderman et al., 2023) when evaluating checkpoints. At the same time, even when compute resources are abundant, benchmarks are bound to make certain concessions aiming to *approximate* true model ability. These

[1] Reproduction code would be supplied upon acceptance

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 2519–2536
June 16-21, 2024 ©2024 Association for Computational Linguistics

concessions – in the form of benchmark design choices – are to be made such that their impact on benchmark reliability (§2) is both minimized and transparent. This, to minimize cases where suboptimal design choices lead to reliability issues such as anointing a different best model or making rank differences between models statistically meaningless.

In this work, we call attention to the topic of **Efficient Benchmarking**, namely intelligently reducing the computation costs of evaluation without compromising reliability. While the trade-off between computation and performance is usually discussed in the context of pre-training (e.g., scaling laws; Hoffmann et al., 2022; Ivgi et al., 2022) and finetuning (e.g., parameter efficient; Lialin et al., 2023), here we call for putting this trade-off on the center stage of evaluation design.

In practice, the compute side of the trade-off already plays a role in most large-scale evaluation decisions, both in benchmark design (Liang et al., 2022) and in its use for evaluation (e.g., choosing the number of seeds; Csordás et al., 2021; Choshen et al., 2022). However, despite their practical importance, these choices and their impact on benchmark reliability have hardly been discussed in the literature, making researchers apply their own efficiency heuristics instead of using systematic guidelines or literature when building their benchmarks.

In order to advance efficient evaluation practices, the community is in need of a systematic set of guidelines and recommendations. These, in turn, must be based on a rigorous study of the different decisions made in benchmark design and how they affect efficiency.

To begin addressing these challenges, we propose Decision Impact on Reliability – *DIoR* – a way to measure the Impact of a Decision over a setup size (e.g., 1K examples, 10 datasets) on the Reliability. In addition, we perform a comprehesive analysis study on efficient benchmarking. With HELM (Liang et al., 2022) as a test case, we test various decisions made and how they affect the trade-off between computation and reliability: decisions about scenarios which are aggregated phenomena (§5.1), subscenarios (§5.2), few-shot prompts (§5.4) and the metric (§5.5). Among other findings, we observe a substantial computation redundancy (see Fig. 1, 4); that a change in one rank is currently unreliable (§5.1); that splitting the data into groups (scenarios) hurts reliability; and that the mean win rate score (§5.5) is unreliable and gameable.

Benchmark Building Tips

1. Know your Reliability-Compute Tradeoff (§2)
2. Compute matters - reduce samples to save (§5.3)
3. Reliability matters - add samples to improve (§5.3)
4. Maximize data-points variability, avoid varying one trait at a time, sample across traits (§5.4)
5. Align resource allocation with importance (§6)

Given our analysis findings, we collect a set of general guidelines for future benchmark creation and use (see Tips above). Moreover, we show how our findings can benefit current benchmarks by proposing *Flash-HELM* (§6), a general evaluation algorithm that enables obtaining a model's ranking with a fraction of the computation and minimal loss of benchmark reliability.

In summary, the contributions of this work are as follows:

1. We highlight the importance of the **balance between computation and reliability** in benchmark design and utilization, and propose DIoR as a quantitative measure of the reliability of a specific efficiency strategy.

2. We conduct the first **systematic study** of the effects of benchmark design on reliability.

3. Given the analysis findings, we provide a set of **practical recommendations** for constructing and using benchmarks; These guidelines outline how best to reduce the computational cost of benchmarking while maintaining an adequate level of evaluation reliability.

4. We propose an **algorithm for dynamic ranking** of a new LM, assigning higher importance to rank top-performing models. In HELM, we show that this algorithm dramatically reduces the computation by up to ×200 with minor deviations from the original ranking (see Fig. 5).

2 The Objective, Validity, Reliability

In this section, we first define 3 critical aspects for evalution: *the objective*, *validity*, and *reliability*. Then, we discuss benchmark reliability and how to measure it, in more detail, being the focus of this study.

The Objective. The question the benchmark aims to answer. For example, “How good is a given model at sentiment analysis?” or “Which is the best language understanding model?”. The objective guides the initial, high-level decisions such as the choice of metric, tasks, domains, and datasets.

Validity. Ensuring that the benchmark actually satisfies the objective, i.e., that it answers the right question, is not trivial. Following common psychometrics literature (Cronbach, 1946), we refer to this quality as *validity*. Validity challenges are often discussed in the literature, in general, (v. Kistowski et al., 2015) and in validating metrics (Choshen and Abend, 2018a; Freitag et al., 2022) or data (Poliak et al., 2018; Gururangan et al., 2018; Northcutt et al., 2021b,a). For example, if the objective is to measure broad language understanding capabilities but the benchmark measures only a narrow aspect of language understanding, the benchmark has a validity problem.

Reliability. Due to the noisy nature of broad evaluation, two valid protocols may yield different results (Maynez et al., 2023). *Reliability* assesses the degree to which the evaluation answer remains consistent under different random decisions, many of which are selections from the distribution of elements composing the benchmark (Kuder and Richardson, 1937).

Building a benchmark involves numerous decisions (e.g., the number of datasets, or of examples per dataset). Importantly, such design decisions determine the reliability of the benchmark, and the conclusions that can (or cannot) be drawn from it. Therefore, we argue that such decisions must be made in an informed manner, including considering their impact on reliability. From a practical point-of-view, well-informed decisions can lead to improved benchmarks, yielding more reliable results with lower computational costs.

2.1 Quantifying Reliability: *DIoR*

Just like significance, which relies on p-value, reliability requires a metric. However, such a metric is not available. Thus, we propose a new metric – the Decision Impact on Reliability test (hereafter *DIoR*) – as a way to assess the effect of a benchmark design decision (e.g., choosing 16 specific language understanding datasets) on the reliability of the benchmark. Given a collection of instantiations of the decision (e.g., dataset samples of size 16), a benchmark scoring function (e.g., rank) and a similarity meta-metric to measure the consistency of the scoring function under a pair of different instantiations (e.g., correlation between rankings), *DIoR* assesses the stability of the meta-metric across different instantiations. Specifically, we define *DIoR* as the lower bound of the confidence interval for the value of the meta-metric; we report the lower bound as this corresponds to the minimal value we are certain of.

Formally, given a set of models M, random instantiations of the decision $c \sim D$, and the original instantiation c_o, a benchmark scoring function $s_c \colon M \to r$ and a similarity meta-metric $f \colon r, r \to [0, 1]$, *DIoR* is defined as:

$$\text{DIoR} = CI_{95\%, c\sim D}(f(s_{c_o}(M), s_c(M)))$$

A reliable decision should receive a high *DIoR*, implying that different instantiations do not substantially affect the results.

3 Data, Models and Scores

As a test case for investigating benchmark efficiency, we analyze the results of the HELM benchmark (Liang et al., 2022). We stress, that although HELM satisfies a good candidate for our analysis, due to the wide range of tasks and models it offers, our conclusions and methods are general and in no way bound to a specific benchmark.

We take the scores of 37 models reported on HELM version 0.2.2[2] as the data for most of our experiments. As a test set for our recommendations (§6), we take the 7 new models introduced in the latest the later version, 0.2.3.

The HELM benchmark defines a taxonomy of *scenarios*, where each scenario corresponds to a collection of labeled data, as well as a metric used to evaluate performance on this data. The benchmark designates 16 scenarios as *“core scenarios”*, on which all LMs are evaluated and a bottom-line score is calculated (see below). Each scenario within HELM is further divided into one or more *subscenarios*; each is an individual dataset with a dedicated scoring function and 3 few-shot prompts, which are originally referred to as seeds. Note that the grouping of subscenarios into a scenario can stem from historical reasons, such as grouping datasets based on prior work. For instance, one of the scenarios in HELM is *RAFT*, which consists

[2] https://crfm.stanford.edu/helm/v0.2.2/?group=core_scenarios

of several different datasets used within the RAFT benchmark (Alex et al., 2021). In §5.2, we discuss the consequences of this grouping decision.

The HELM benchmark ranks LMs by an aggregation of their scores over all 16 core scenarios and 65K examples. The aggregation metric used is *Mean Win Rate (MWR)*, which compares LMs against one another per scenario (a Borda Count variant, Emerson, 2013). MWR measures the average win rate for each model over all scenarios (see App. A for a formal definition).

4 Experimental Setting

In our main experiments, we calculate *DIoR* to examine the reliability under the current realization of the benchmark, as well as more efficient realizations. Thus, we calculate *DIoR* for varying amounts of compute, ranging from the full HELM benchmark to a small fraction of it (e.g., a benchmark with 1 scenario, or 100 examples).

For each design choice (number of examples, scenarios etc.) we sample different instantiations of this choice, and use them to calculate *DIoR*. We follow a bootstrap approach, namely, sampling 1K times with repetition. For example (in §5.1), to test whether taking 10 scenarios reliably indicates the best model, we sample 10 scenarios (out of the available 16) 1K times, calculating the win rate values for each sample (in a sample, some datasets may be chosen more than once, or not at all).

4.1 Benchmark Objectives

Throughout our analysis, we consider three objectives that benchmarks often aim to measure. For every objective, we recommend a specific metric and then provide a related meta-metric to check its reliability.

One objective is to acquire the **full ranking**. The meta-metric measures the number of models switching places in the overall ranking (Kendall τ). We also calculate a weighted alternative that emphasizes correctly ranking the top models (Vigna, 2014), finding generally similar trends (see App. §C, §E).

For the objective of determining which model is the **best model**, we define the meta-metric as the *Error Rate*, namely, the probability (across different instantiations) of a rank switch between the top two models. As we care about the best model in general, and not the current one specifically, we repeat the experiment 5 times, each time removing the top model from the benchmark (as if it was not yet submitted).

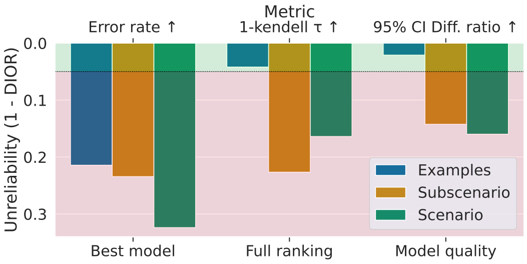


Figure 2: **Scenarios / Subscenarios / Examples *DIoR*.** Different subscenarios or scenarios would highly affect results, but not examples. Each cluster of bars represents a measure of *DIoR* (top labels) for the corresponding objective (bottom labels). Each color denotes the granularity: Examples, Subscenarios, or Scenarios. The area above the vertical line in light green represents high reliability levels ($\geq 95\%$), while the area below in red indicates lower reliability.

The last objective is to evaluate **model quality**, i.e., how well each model performs. For this, we calculate the absolute bottom-line score. To be consistent with the literature, we report MWR as the model quality metric, where the meta-metric is the absolute difference in MWR scores.

5 Results

In this section, we examine the impact of different design choices on the reliability of the benchmark objectives.

5.1 Scenarios

HELM selected 16 core scenarios for model evaluation. We do not challenge this choice's validity or relevance. Instead, by applying bootstrapping, we run a simulation of selecting equally valid alternative scenarios, in order to investigate the reliability of this choice.

In Fig. 2 we report the reliability of HELM's original choice of (16) scenarios, for each of the objectives. We find that the reliability of the set of scenarios is low. Put another way, under a different choice of scenarios it is quite likely that HELM's ranking, score, and winners would be different. Further, as shown in App. §C, upon reducing the number of scenarios, reliability drops drastically; thus, the common compute-reduction approach of dropping datasets (e.g., big-bench lite, bench authors, 2023), is in fact an ill-advised practice.

5.2 Subscenarios

In this section, we first examine the reliability of HELM's original design choice of 40 subscenarios (single-datasets grouped to construct the scenarios). Then, we question the reliability of the common design choice (also prevalent in HELM) of grouping multiple subscenarios into scenarios vs. keeping every subscenario as a standalone.

Repeating the reliability test for the three objectives (Fig. 2 and App. §C), we find that similarly to scenarios, the choice of subscenarios only supports low reliability, meaning that dropping subscenarios is a problematic approach for reducing compute. Given this finding, we revisit the decision to group subscenarios into scenarios. We find, that in terms of reliability, considering each subscenario as a standalone scenario is helpful, for example, in reducing the error rate between top pairs to 14% instead of 22% (see App. F).

The rationale for grouping subscenarios is their shared focus on testing a particular skill or phenomenon. Their reweighting as a single group prevents over-emphasis on this skill across the benchmark (see example in App. §B). By complementing each other, these grouped subscenarios should offer a more holistic assessment of that specific skill. Qualitatively, in HELM, it is not clear that the grouping is crucial and indeed prevents over-representation of specific phenomena. For example, the 7 open/closed question answering *scenarios* (e.g., openbookQA, Mihaylov et al., 2018) seem closer in spirit to each other, than MMLU's (Hendrycks et al., 2020) 4 *subscenarios* which were designed to cover distinct topics in language understanding.

If the above intuition proves correct, related subscenarios should test the same skill and are expected to rank models consistently. Conversely, rankings from unrelated subscenarios would likely diverge, as they evaluate different capabilities.

In App. §E, we measure just that and present the correlation between rankings made by different subscenarios. We do not find an a stronger correlation within subscenarios that belong to the same scenarios, concluding that aggregation is not needed for *validity*.

Although we found that aggregating subscenarios scores hurts the general benchmark reliability, we note that aggregated scores might still be interesting to report for fine-grained evaluation (Gehrmann et al., 2021) or for historical reasons (reusing a benchmark) as these can be considered as separate sub-objectives. Hence, we suggest that each such sub-objective would include aggregations, but that the bottom-line benchmark calculations should ignore these. This will allow the benefits of sub-objectives while preventing overall benchmark reliability decline. Concretely, in HELM, one can aggregate the final model score over all *subscenarios*, but still report the aggregated scores per *scenario* separately.

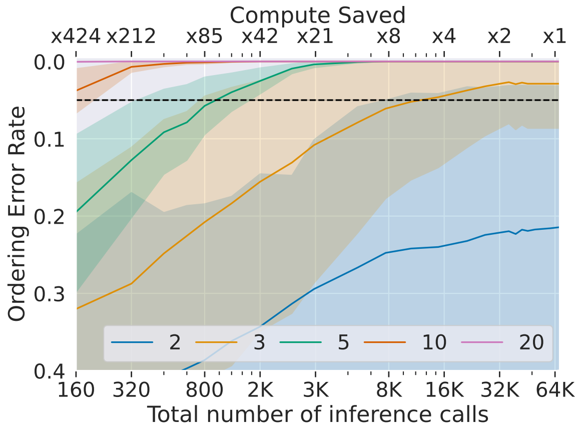


Figure 3: **The probability that models would switch places** (y-axis) given a different random choice of examples for evaluation (x-axis). Each line corresponds to taking a group of N models and testing if the top and bottom switch places. Results are averaged across 1K iterations (95% confidence interval in shade) and over the top 5 models as the top model.

5.3 Examples

Previously, we have found that the reliability of (sub)scenarios is already low, hence decreasing computational cost by removing them is undesirable. In contrast, as Fig. 2 shows, the current choice of examples is highly reliable. Thus, removing examples is a preferable strategy for reducing compute. Further, we find a certain discrepancy between the objectives, where best-model is not reliable while model-quality and full-ranking are. To reiterate, in the current state of HELM, discussing the top model is pointless. In smaller benchmarks, we expect the problem to be even more severe.

For the model-quality and full-ranking objective, the current state is reliable, hence, we test reliability with fewer examples per scenario. We find (see Fig. 1) that model ranks are quite stable regardless of the number of examples used. Remarkably, with the bare minimum of examples, models are already clustered into equivalence classes of about 2-5 models, and with a few hundred examples, models are separated into groups of ∼2 – the best separation the benchmark ever achieves. We find similar trends, of high reliability with a small number of examples, for the other objectives as well (App. §C). Fig. 6 quantifies the error in rank per model, and finds it is small, ranging from 6 to 2.

In the findings discussed so far, we repeatedly found models to be indistinguishable from the model ranked right above or below them. However, it is also interesting to consider the level of separation between models that are farther apart. Thus, we examine clusters of adjacent models within the full HELM ranking (e.g., for a cluster size of 5, we consider the models ranked 1-5, 2-6, 3-7 etc.). In Fig. 3 we plot the probability of rank location switch (i.e., error rate) between the first and last models in rank clusters of sizes 2, 3, 5, 10 or 20. While with cluster size 2 models often switch places – even with all benchmark examples – for clusters of 3 (i.e., a diff of two places), ¼ of the computation is sufficient to get an average error rate under 5%. For larger clusters, one can get reliable results with a hundredth of the cost or less.

We leave the special case where a "benchmark" is a single dataset to App. §D. Even then, fewer examples suffice. Moreover, as some datasets are more stable than others, one can tune the number of examples per dataset as needed, taking more examples where distinctions are harder to make. We leave more elaborate research on that for future work.

5.4 Few-Shot Prompts (HELM's seeds)

Under the in-context learning paradigm, LMs are expected to predict the right answer given some examples. As the choice of exemplars might change results (Min et al., 2022; Dai et al., 2023; Pan, 2023), a reliable benchmark should account for this variability as well. For this reason, HELM considers three sets of few-shot exemplars, uses them against every example (Liang et al., 2022), and averages their score.

To assess the reliability of prompts, bootstrap approximation is not a viable option as there are only three prompts. Instead, we compare the effect of two different approaches for using a given budget to evaluate model performance. In our example, the budget of inference calls is 3K examples. One method, as HELM did, samples a set of K examples and then samples prompts (3). Then, every model is tested on every example and prompt, the full cross-product. In contrast, one may sample *uniformly* from the cross product of all (K) examples and all possible prompts, where each call samples a different prompt and example (and perhaps other traits), ideally a unique example and prompt in each call. Thus the calls will evaluate 3K examples and 3K prompts. This approach captures more from each variable (e.g., example), but cannot separate the impact of each specific example on the performance of the model. As our use case is benchmarking, we do not care about, for instance, which example makes the model fail, and hence expect the uniform sampling to be more fitting. In practice, the group of all possible prompts is of size 3, as is available in HELM.

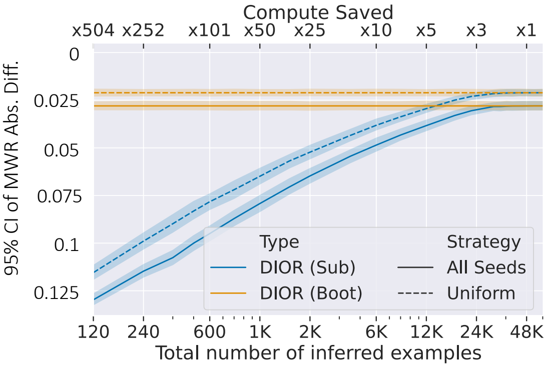


Figure 4: **In-context example selection strategies.** The 95% CI of the MWR difference of bootstrap (*Boot*) and sub-sample (*Sub*) for two choices of In-context example selection (*All Prompts* and *Uniform*). It shows that (1) sampling In-context examples uniformly from the pool is superior to using all examples per samples and that (2) more than half of the compute can be saved at no cost to reliability. This analysis discarded the scenarios that did not vary their in-context examples.

Comparing the two methods in Fig. 4, we find that the uniform method increases reliability. Being limited to only three prompts, we expect this is an underestimation of its true potential. Inducting from the prompt-example pairing to the general case, when multiple factors are taken into account, we conclude it is best to sample uniformly, covering as much variability of each factor.

5.5 Metrics

Choosing a valid and reliable metric is a complex art, with vast literature. From discussion about metric biases (Choshen and Abend, 2018b; Mathur et al., 2020; Sulem et al., 2018; Peyrard et al., 2021), to metric validation (Choshen and Abend, 2018a; Honovich et al., 2022; Zerva et al., 2022; Kocmi et al., 2021; Fabbri et al., 2020), referenceless metrics (Honovich et al., 2021; Rei et al., 2022) and models that evaluate themselves (Chia et al., 2023). However, benchmarks that rely on existing datasets (the subscenarios) as their building blocks often adopt their metrics as well. Thus, in this analysis, we discuss only the proposed way to convert subscenarios' scores to HELM's score – MWR. This includes two decisions; the grouping of subscenarios into scenarios, where each scenario is weighted similarly (see discussion in §5.2), and the decision to convert the absolute scores per model to a comparative score.

A comparative measure such as win rate provides a preference over models, but can not tell how good a model is at performing a task. This is especially useful when preference is easier to collect than an absolute score, as often happens with human evaluation (Bojar et al., 2016; Choshen and Abend, 2018a), or if even direct assessment produces relative scores unintentionally (Mathur et al., 2017; Liang et al., 2020). Fortunately, this is not our case, where each subscenario provides a score for each example and MWR converts it into pairwise comparisons as a normalization technique. There are however known and inherent limitations to comparative measures, most famously the impossibility theorems (Arrow, 1950). In this case, introducing a new model to the benchmark changes the scores of existing models (Knowles, 2021).

We analyze if this indeed affects the MWR we currently observe. Take for example the first two models in HELM – *davinci2* (Ouyang et al., 2022) and *Cohere XXL*. Those top models switch places when they are compared with or without *Cohere Medium*. This follows from MWR's tendencies. When we introduce models that are just slightly worse in everything than one model, this model sweeps the benchmark collecting all the wins while other models only get some of the wins. Thus, introducing a weaker model changed the rank of two stronger models.

One might consider the example above a rare and extreme case, but actually, this is the expected case. The common practice today is to release several sizes of a new model. Those model variations were trained similarly, and hence tend to have similar strengths, with the larger variant being stronger in every aspect. In App. §B we provide a simple numerical example of models changing rank when a new weaker model is introduced.

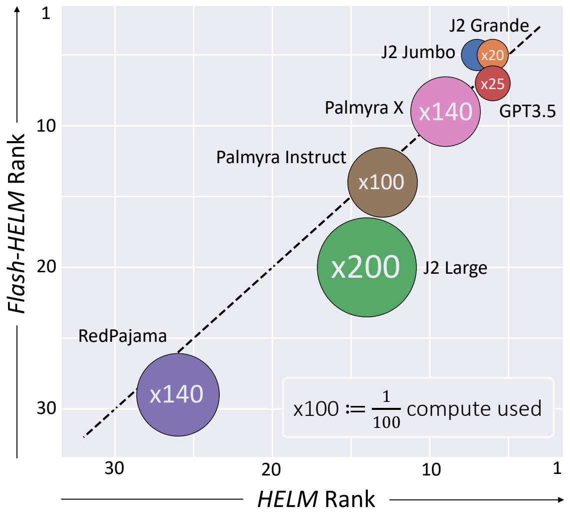


Figure 5: **Efficient evaluation proposal.** *Flash-HELM* (§6) produces similar ranks to HELM with a fraction of the compute. Models are Test-set and were not part of the analysis. Each circle size and numerals represent the reduction in compute usage for evaluation, in comparison to HELM's .

In essence, one can maliciously raise a model to the top by evaluating numerous models almost equal to their own, but with one wrong sentence in each scenario. While such intentional gaming is unanticipated, it is a favorable characteristic for a benchmark to improve only if results are better, encouraging innovation through healthy competition.

6 Efficient Alternative: *Flash-HELM*

In this section, we demonstrate the practical utility of our study, by proposing an efficient variation of the HELM benchmark, which we coin as *Flash-HELM*. This variation preserves the important information of HELM, while reducing computation costs by up to 200 times.

Objective. As discussed in §2, a well crafted evaluation answers a question. Here we consider the question *"How is model X ranked when compared to other models?"*. Usually, however, the required reliability of the answer varies depending on the model's performance. For example, when a model's ranking falls within the lower range of the benchmark – say, between positions 25 and 40, the precise ranking might not hold much importance; instead, a broad conclusion that the model is poor should suffice. On the flip side, when a model attains a position in the top 5 ranks, the specific placement carries more weight.

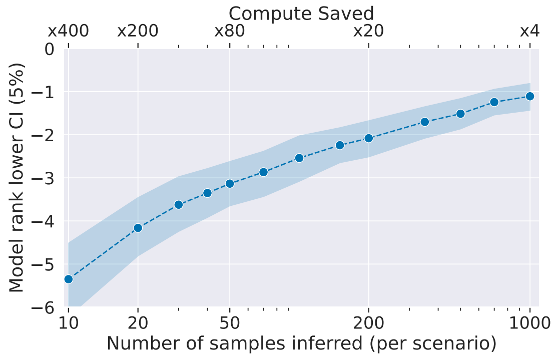


Figure 6: Reliable Rank Resolution in HELM

Algorithm 1 Efficient 'coarse-to-fine' tournament

```
$M \leftarrow$ The evaluated model
**for** Sample size $S \in [20, 50, 200, 1000, Max]$ **do**
    $Rank(M, S) \leftarrow$ Evaluated model $M$ using sub-sample of size $S$.
    **if** $Rank(M, S) + Res(S) \geq TierRank(S)$ **then**
        stop;
    **end if**
**end for**
report $Rank(M, S)$.;
```

Approach. Following this motivation, we propose the following use-case: segmenting the ranking into five "tiers"; Rank 1, Ranks 2-4, 5-9, 10-19, and Ranks 20 and below. Now, we associate each tier with a designated 'desired reliability' level, starting with a low amount of computation for lower ranks and gradually raising it for higher ranks. This allows to evaluate most models tapping into a fraction of the computation. To achieve this improved efficiency while minimally harming reliability, we reduce examples and sample prompts as suggested above.

Algorithm. For each tier, 1-4, 5-9, 10-19, and 20 we set the required precision to 1,2,3 and 4 rank resolution respectively. For the top model, we set the precision requirement to be maximal, as identifying the best model bears special importance. Based on that and the data in Figure 6, we can determine the size of the sub-sample needed in order to evaluate a model in each tier. We denote this relationship as $TierRank(S)$ – the lower rank of the "tier" that is associated with a sub-sample size S. Furthermore, we denote: $Rank(M, S)$ - the rank of model M when calculated using a sub-sample size S, and $Res(S)$ – the achieved rank resolution for sub-sample size S (based on Fig. 6). We formulate an efficient 'coarse-to-fine' tournament algorithm.

Evaluation. We assess the performance of our algorithm using the seven newly-introduced models found in HELM v0.2.3 – models that were not used in our previous analysis. The results are showcased in Figure 5. *Flash-HELM* ranks are very close to the full HELM ranks and are within the required resolution. These results highlight the algorithm's effectiveness in preserving important ranking information while achieving a reduction up to a factor of 200 in computational demands.

7 Discussion and Conclusion

Why this sudden focus on reliability when our field largely thrived without it? The shift from single datasets to complex multi-dataset benchmarks, like HELM, has changed the evaluation landscape. In the past, individual datasets offered great reliability due to a large i.i.d. sample pool spanning all the relevant example space; In current benchmarks, on the other hand, the space is constructed of more dimensions such as datasets, prompts, etc. For some of those dimensions (e.g. 3 prompts in HELM), the benchmark holds a handful of examples making for a severely low coverage. Even when the number of examples is sufficient for reliable, and stable insights for some dimensions, this might not be true for others. the lack of sufficient coverage narrows the gap with fields like psychology and physiology, which often rely on smaller samples. Just as we wouldn't expect meaningful psychological insights from 3 or even 16 human subjects, we shouldn't expect reliable conclusions from just 3 different prompts or 16 datasets without careful design.

Our study shows that by utilizing efficient evaluation methods we can both increase reliability and drastically reduce costs. We advocate for the development of more *transparent*, *efficient* and *reliable* evaluation benchmarks and techniques, and by doing so, not only to enhance their effectiveness, but also to make research more accessible across diverse groups, more reproducible and more respectful of environmental concerns.

Limitations

Future work will analyze other benchmarking decisions and other benchmarks. Thus, while the paper's results are sound, they might ignore common unreliable decisions in other benchmarks which were not apparent in this scenario or were left out (such as the choice of prompt templates, choices of non-textual benchmarks, etc.). A decision of special interest is that of efficient inference methods. With many efforts to tackle the tradeoff between performance and computation (Chen et al., 2023; Choukroun et al., 2019), future work would wonder if there is a validity and reliability tradeoff as well or if such methods can be used to evaluate models (that do not use them) as well.

In some of the analyses (e.g., tournament) we compare the change with respect to the reported HELM score, as we note throughout the paper, this is but an approximation of the true score each model deserves. Thus, where an efficient method might seem to be deviating in 3 ranks, it might only deviate in 2 (or 4) because the point of reference may itself be wrong. In a sense, reliability compares the change among realizations and solves this problem by not defining which one is the true value.

Here, we considered datasets as sampled and hence similarly informative (except for discussing their correlations). However, it is possible to split datasets into meaningful scenarios. If this would be done for validity reasons one would also want to diversify the scenarios used, perhaps in the space of tasks and domains and skills.

The validity and reliability axes and the claims calling for considering the tradeoffs carefully are general. However, we note that the specific analysis is, specific. It might change in the future, if models characteristics change drastically or improvements make some of the subscenarios redundant.

Especially prone to that is the rank change, if many similar models are added. In that case, each model would switch more ranks, but as models would still show grouped behaviour and the absolute scores won't change more, we assume the meaningful qualifiers would change as well (for a thousand models, a ±10 in ranks might not be as meaningful as with the current 40).

Another limitation of our work is that we introduce a known and critical aspect in testing, reliability, but evaluate it in an unconventional way. We believe using confidence intervals to be more intuitive and more general and available as anyone running the benchmark already has access to the computation necessary. However, it is more likely that adaptations and improvements would be needed as the traditional statistical study of reliability focuses on variances and such notions.

Our use of bootstrap for experiments (especially with full HELM) has two main limitations. The first is the limitation of bootstrapping in general, while this is the best approximation of the real distribution (e.g., of examples), it is merely an approximation using the sample at hand (HELM's data).

The second, is that we add an assumption that other decisions could be made that are as valid as the one made by HELM. In other words, we assume there exists a larger distribution from which other choices could have been taken (e.g., instead of considering a summarization task scenario considering paraphrase generation). We do not see that as a strong assumption, as we do not need to explicitly state which distribution that is. If however, the dataset were covering exactly all types of known capabilities or following a theory, that specific choice might not have been a good prospect to test reliability, as it could not change under the circumstances.

References

Neel Alex, Eli Lifland, Lewis Tunstall, Abhishek Thakur, Pegah Maham, C. Jess Riedel, Emmie Hine, Carolyn Ashurst, Paul Sedille, Alexis Carlier, Michael Noetel, and Andreas Stuhlmüller. 2021. RAFT: A real-world few-shot text classification benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

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

BIG bench authors. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Stella Biderman, USVSN Sai Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raf. 2023. Emergent and predictable memorization in large language models. *arXiv preprint arXiv:2304.11158*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Kaijie Zhu, Hao Chen, Linyi Yang, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2023. A survey on evaluation of large language models. *arXiv preprint arXiv:2307.03109*.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. 2023. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*.

Yew Ken Chia, Pengfei Hong, Lidong Bing, and Soujanya Poria. 2023. Instructeval: Towards holistic evaluation of instruction-tuned large language models. *arXiv preprint arXiv:2306.04757*.

Leshem Choshen and Omri Abend. 2018a. Automatic metric validation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1372–1382, Melbourne, Australia. Association for Computational Linguistics.

Leshem Choshen and Omri Abend. 2018b. Inherent biases in reference-based evaluation for grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 632–642, Melbourne, Australia. Association for Computational Linguistics.

Leshem Choshen, Elad Venezian, Shachar Don-Yehia, Noam Slonim, and Yoav Katz. 2022. Where to start? analyzing the potential value of intermediate models. *arXiv preprint arXiv:2211.00107*.

Yoni Choukroun, Eli Kravchik, Fan Yang, and Pavel Kisilev. 2019. Low-bit quantization of neural networks for efficient inference. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3009–3018. IEEE.

Lee J Cronbach. 1946. Response sets and test validity. *Educational and psychological measurement*, 6(4):475–494.

Róbert Csordás, Kazuki Irie, and Juergen Schmidhuber. 2021. The devil is in the detail: Simple tricks improve systematic generalization of transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 619–634, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. 2023. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

Peter Emerson. 2013. The original borda count and partial voting. *Social Choice and Welfare*, 40:353–358.

A. R. Fabbri, Wojciech Kryscinski, Bryan McCann, Richard Socher, and Dragomir R. Radev. 2020. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

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. 2021. A framework for few-shot language model evaluation.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, et al. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120, Online. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Xiaodong Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *ArXiv*, abs/2009.03300.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 161–175, Dublin, Ireland. Association for Computational Linguistics.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Maor Ivgi, Yair Carmon, and Jonathan Berant. 2022. Scaling laws under the microscope: Predicting transformer performance from small scale experiments. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7354–7371, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Rebecca Knowles. 2021. On the stability of system rankings at WMT. In *Proceedings of the Sixth Conference on Machine Translation*, pages 464–477, Online. Association for Computational Linguistics.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

G Frederic Kuder and Marion W Richardson. 1937. The theory of the estimation of test reliability. *Psychometrika*, 2(3):151–160.

Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. 2023. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.15647*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Weixin Liang, James Zou, and Zhou Yu. 2020. Beyond user self-reported likert scale ratings: A comparison model for automatic dialog evaluation. *arXiv preprint arXiv:2005.10716*.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2017. Sequence effects in crowdsourced annotations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2860–2865, Copenhagen, Denmark. Association for Computational Linguistics.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Joshua Maynez, Priyanka Agrawal, and Sebastian Gehrmann. 2023. Benchmarking large language model capabilities for conditional generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9194–9213, Toronto, Canada. Association for Computational Linguistics.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Curtis Northcutt, Anish Athalye, and Jonas Mueller. 2021a. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran.

Curtis Northcutt, Lu Jiang, and Isaac Chuang. 2021b. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Jane Pan. 2023. *What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning*. Ph.D. thesis, Princeton University.

Maxime Peyrard, Wei Zhao, Steffen Eger, and Robert West. 2021. Better than average: Paired evaluation of NLP systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2301–2315, Online. Association for Computational Linguistics.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018. BLEU is not suitable for the evaluation of text simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 738–744, Brussels, Belgium. Association for Computational Linguistics.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Jóakim v. Kistowski, Jeremy A Arnold, Karl Huppler, Klaus-Dieter Lange, John L Henning, and Paul Cao. 2015. How to build a benchmark. In *Proceedings of the 6th ACM/SPEC international conference on performance engineering*, pages 333–336.

Sebastiano Vigna. 2014. A weighted correlation index for rankings with ties. *Proceedings of the 24th International Conference on World Wide Web*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Zhiqiang Yuan, Junwei Liu, Qiancheng Zi, Mingwei Liu, Xin Peng, and Yiling Lou. 2023. Evaluating instruction-tuned large language models on code comprehension and generation. *arXiv preprint arXiv:2308.01240*.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, Piyawat Lertvittayakumjorn, José G. C. de Souza, Steffen Eger, Diptesh Kanojia, Duarte Alves, Constantin Orăsan, Marina Fomicheva, André F. T. Martins, and Lucia Specia. 2022. Findings of the WMT 2022 shared task on quality estimation. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 69–99, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Jianguo Zhang, Kun Qian, Zhiwei Liu, Shelby Heinecke, Rui Meng, Ye Liu, Zhou Yu, Silvio Savarese, and Caiming Xiong. 2023. DialogStudio: Towards richest and most diverse unified dataset collection for conversational AI. *arXiv preprint arXiv:2307.10172*.

A Mean Win Rate: A Formal Definition

For a given set of models M, scenarios CS containing subscenarios. Each subscenario provides a single metric to evaluate and score models. For brevity, we identify subscenarios $s{\in}CS$ with their scoring function and define it as $s\colon M{\to}\mathbb{R}$:

$$MWR(m) =$$

$$\mathop{\mathbb{E}}_{S\in CS_m} \mathop{\mathbb{E}}_{m_i\in M\setminus m} \mathbb{1}\left(\mathop{\mathbb{E}}_{s\in S} s(m) > \mathop{\mathbb{E}}_{s'\in S} s'(m_i) \right)$$

Where $\mathbb{1}$ is the indicator function. When a subscenario score was not submitted to the benchmark for a specific model m (missing value) it is omitted from CS, denoted as CS_m.

B Examples of Score Sensitivity

We give several examples of how small changes change the ranking of models without need.

Adding a model. Take two models with per model scores 10,10,10 and 12,12,8. The second is clearly better. It also gets a better score when the two are compared. However, adding an even worse model 9,9,9 now changes the picture. The win rates of the original models are now 0.5,0.5,1 and 1,1,0 respectively. So on average the scores of the two models are suddenly tied. Adding more such models would improve the first model's ranking more and more, enlarging the difference.

Combining datasets. Let two models have scores 1,1,0,0 and 0,0,1,1 on 4 datasets respectively. If we call the first two datasets a scenario, we get that one model wins on one scenario and loses on two. This makes the first model suddenly better; choosing the last two datasets would do the contrary.

Reporting partially. Let three models have average win rates of 0.9,0.9 and 0 and 0.8,0.8,0.8 with many models. If the first model does not report the last 0-winning-rated dataset, then it is considered a better model, with 0.9 win rate on average, while it would be the worse one with 0.6 win rate otherwise.

C Objectives per Decision

In this section, we present graphs (8,9,10) for decisions and objectives that were left out of the main paper. We provide a triplet of graphs per decision (one for each objective): scenarios in Fig. 8, subscenarios in Fig. 9 and examples in Fig. 10.

D Each Dataset as Standalone Benchmark

In this section, we report (Fig. 7) the results separated and without aggregation. We presume this would be helpful both for the use of single standalone benchmarks in the future, and for more elaborate choices when integrating datasets into a new benchmark, such as choosing datasets which provide commendable traits or varying the number of examples shown per dataset in the benchmark.

E Full Subscenario Correlations

We provide the full heatmap of correlations between pairs of subscenarios in Fig. 12, finding little similarity within scenario.

F Scenario Vs Subscenario Aggregation

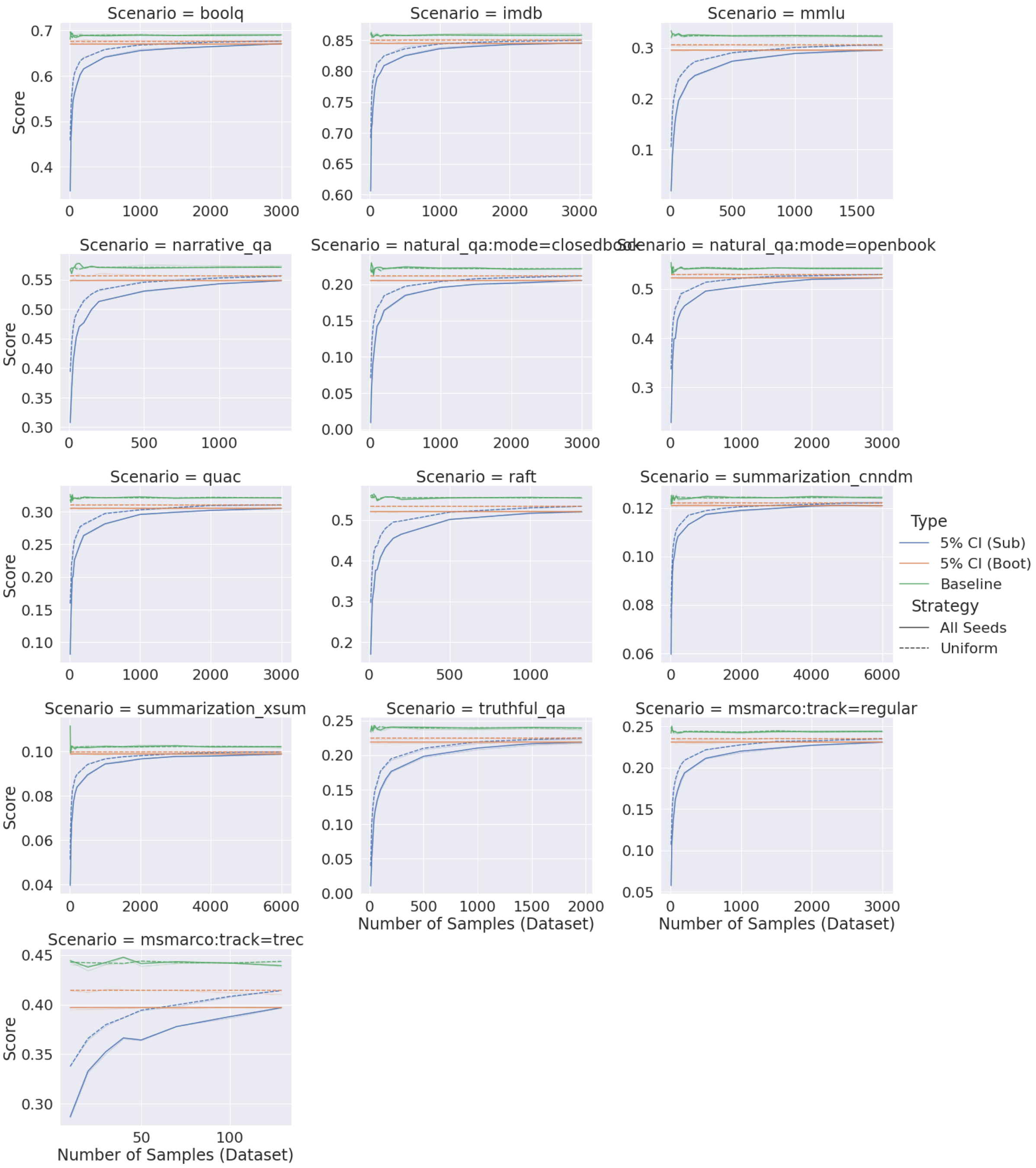
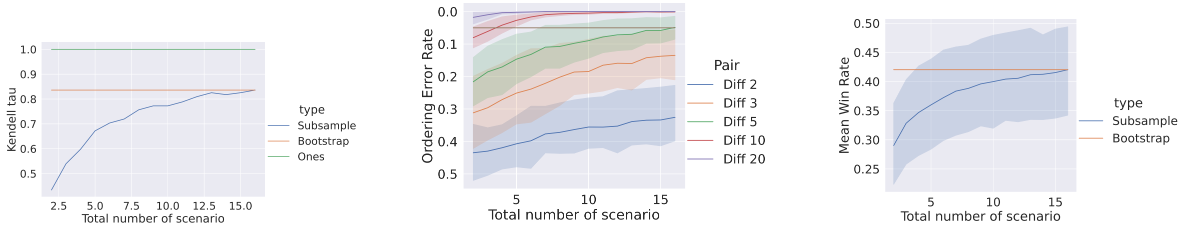


Figure 7: **In-context example selection strategies.** The figure depicts the mean dataset score along with the bootstrap (*Boot*) and sub-sample (*Sub*) 5% Confidence intervals for two choices of In-context example selection (*All Seeds* and *Uniform*). It shows that (1) sampling In-context examples uniformly from the pool is superior to using all examples per samples and that (2) more than half of the compute can be saved at no cost in score reliability.

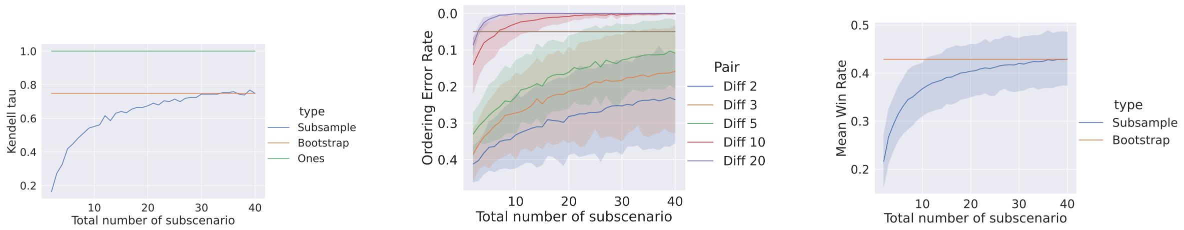


(a) Full Ranking

(b) Best Model

(c) Model Quality

Figure 8: Scenarios reliability. Reliability of different amounts of computation for the three objectives and corresponding meta-metrics.

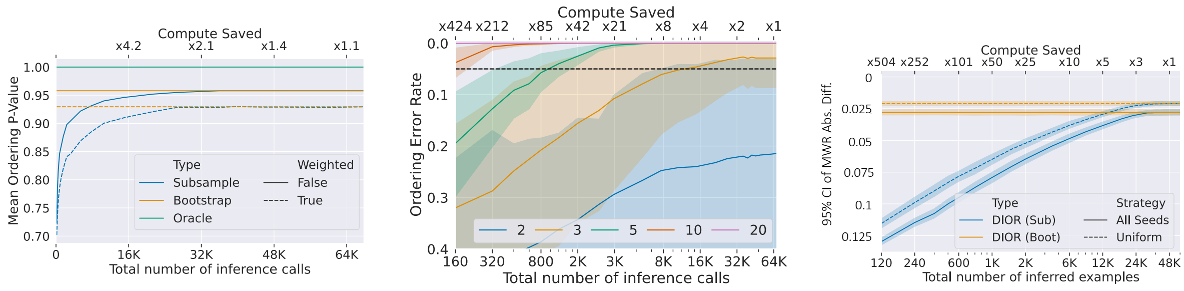


(a) Full Ranking

(b) Best Model

(c) Model Quality

Figure 9: Subscenarios reliability. Reliability of different amounts of computation for the three objectives and corresponding meta-metrics.



(a) Full Ranking

(b) Best Model

(c) Model Quality

Figure 10: Examples reliability. Reliability of different amounts of computation for the three objectives and corresponding meta-metrics.

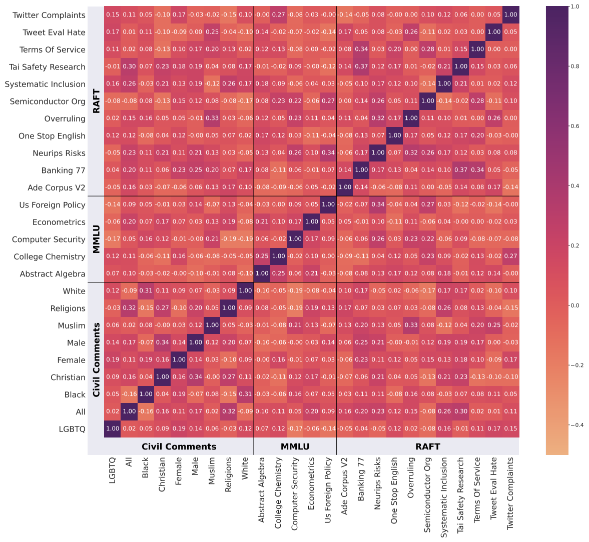


Figure 11: **Subscenerios Ranking Correlations.** The Kendall τ correlation matrix between model rankings in each standalone subscenario. Correlations within a scenario are not higher than across scenarios.

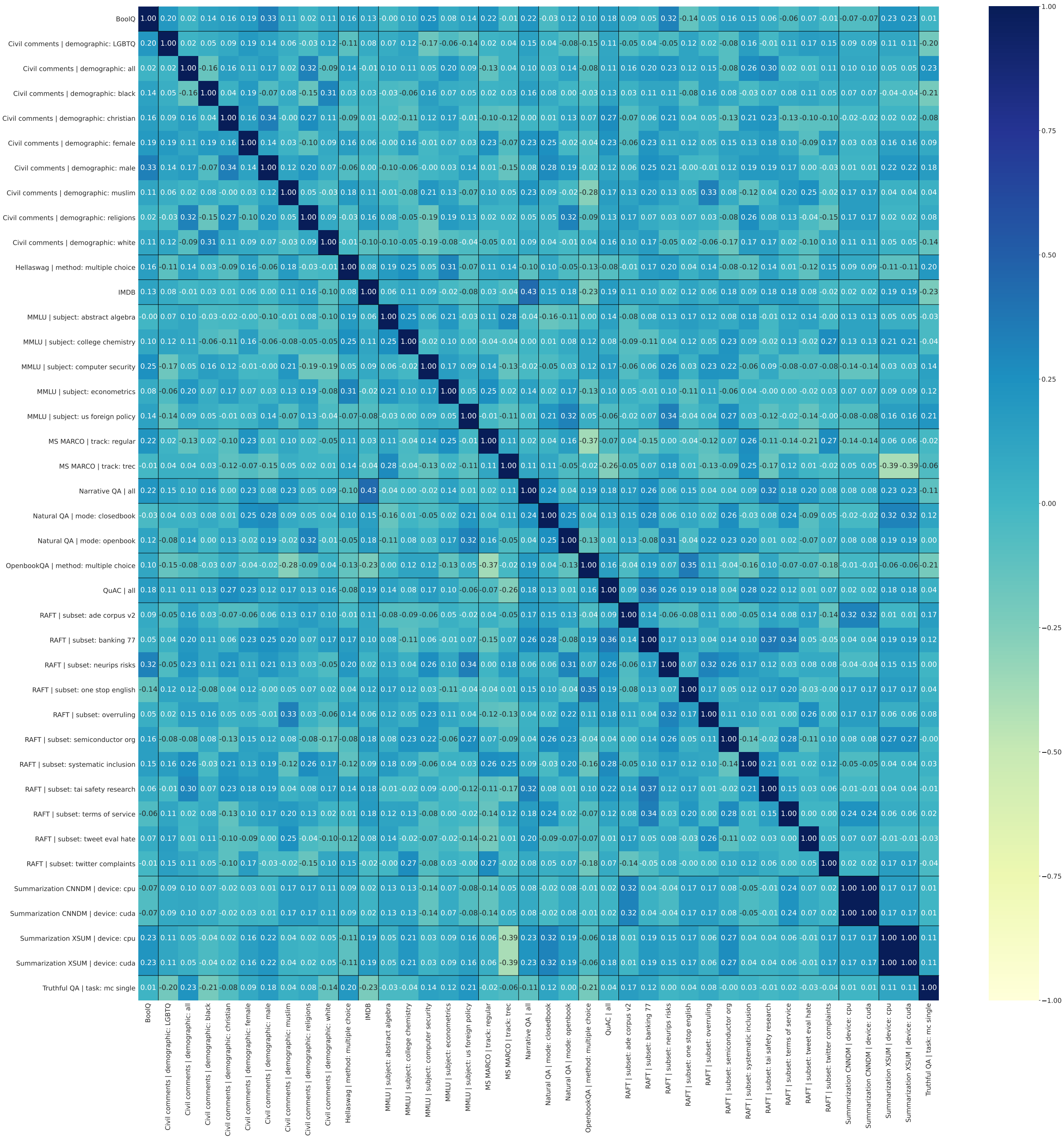


Figure 12: **Subscnerios Ranking Correlations.** This figure depicts the Kendall τ correlation matrix between the ranking of models based on the performance in different subscenarios.

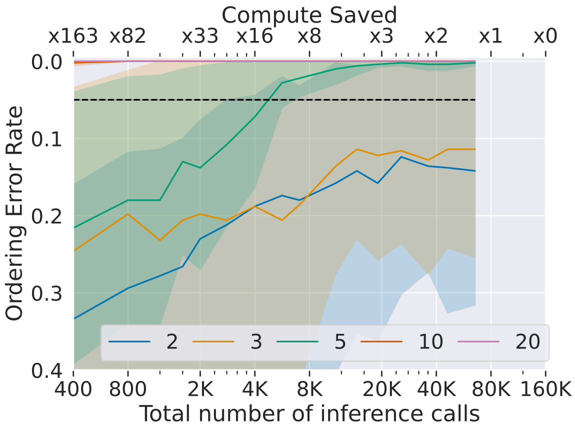


Figure 13: **The probability that models would switch places, MWR over subscenarios** (y-axis) given a different random choice of examples for evaluation (x-axis). Each line corresponds to taking a group of N models and testing if the top and bottom switch places. Results are averaged across 1K iterations (95% confidence interval in shade) and over the top 5 models as the top model.

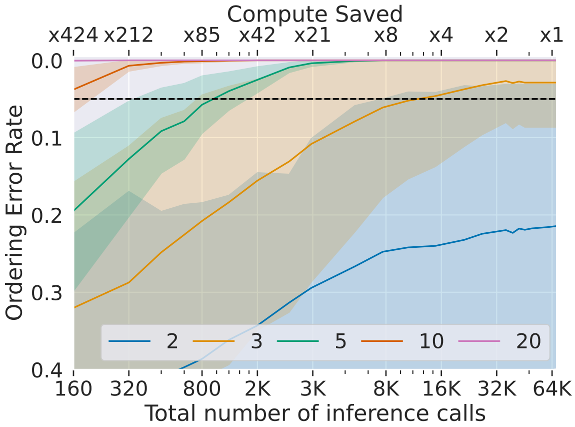


Figure 14: **The probability that models would switch places, MWR over scenarios** (y-axis) given a different random choice of examples for evaluation (x-axis). Each line corresponds to taking a group of N models and testing if the top and bottom switch places. Results are averaged across 1K iterations (95% confidence interval in shade) and over the top 5 models as the top model.