SubLIME*: Data efficient foundation model evaluation across modalities, languages and benchmarks

Anonymous authors
Paper under double-blind review

Abstract

The exponential growth of foundation models has created an unsustainable evaluation paradigm, where comprehensive assessment incurs prohibitive computational costs and environmental impact. We introduce **SubLIME*** ("Less Is More for Evaluation"), an extensible framework that reduces evaluation costs by 10-100× through adaptive sampling while preserving model ranking fidelity (Spearman $\rho > 0.9$). Our core innovation lies in identifying minimal representative subsets through three key extensions: (1) **SubLIME-I** for text-to-image models combines difficulty and quality sampling methods validated on the image generation tasks, reducing inference time from 2792 hours to 28 hours for evaluating 27 models; (2) **SubLIME-C** eliminates cross-benchmark coding redundancies via LLM-guided similarity analysis (80% precision vs 66% baseline), improving correlation by 14% at fixed sample sizes; (3) **SubLIME-M** enables multilingual assessment through cross-lingual subset alignment, maintaining > 0.8 rank correlation across 4 languages with 80% less data. SubLIME* experiments across modalities, languages and benchmarks show that using strategic sampling based on difficulty gradients, semantic diversity, and quality metrics maintains evaluation integrity while significantly reducing costs by orders of magnitude.

1 Introduction

The rapid expansion of foundation models has transformed the landscape of artificial intelligence. Platforms such as HuggingFace now host over 200,000 open-source generative AI models—including roughly 180,000 for text generation and 58,000 for text-to-image tasks—while leaderboards like the Open LLM leaderboard and AlpacaEval track evaluations for more than 90,000 models. However, as model architectures diversify (e.g., fine-tuned, quantized, or merged variants) and benchmark datasets proliferate, the computational, financial, and environmental costs of comprehensive evaluations escalate dramatically. For instance, the HELM project Liang et al. (2023) spent approximately $50,000 (with $38,001 on commercial APIs and an additional 19,500 A100 GPU hours at $1/hr) to evaluate only 30 LLMs over 13 tasks. Scaling such evaluations to even a fraction of the available models and benchmarks would be unsustainable.

In response to these challenges, we introduce **SubLIME** (“Less Is More for Evaluation”), a data-efficient evaluation framework that leverages adaptive sampling techniques to identify representative, diverse, and high-quality subsets from large-scale benchmarks. By systematically eliminating redundant evaluation instances while preserving model rankings and score distributions, SubLIME dramatically reduces evaluation costs without compromising assessment fidelity.

Evaluation inefficiencies are particularly evident in areas like image generation, where many diffusion models took anywhere from 30 seconds to several minutes to generate a single image. Our analysis of the logs for building the HEIM leaderboard Hugging Face (2022) shows that evaluating 27 text-to-image models consumed 2,792 hours of inference time. SubLIME-I addresses these challenges by using adaptive sampling methods tailored to the specific needs of text-to-image benchmarks, maintaining a high correlation with full-scale evaluations while using only 1% of the data.

Evaluating code generation tasks can be more expensive than assessing multiple-choice language benchmarks because they involve additional steps like code execution and verification, which add computational overhead. Furthermore, several coding benchmarks use overlapping test problems, presented in different formats for prompts. SubLIME-C addresses this issue by using LLM-based similarity analysis to detect and remove these cross-benchmark redundancies. This domain-specific extension simplifies the evaluation process, ensuring only unique instances are kept.

As language models become integral to global applications, evaluations must move beyond English-centric benchmarks. Multilingual assessments capture variations in syntax, semantics, and cultural context, yet the scarcity of high-quality data in low-resource languages forces evaluators to multiply efforts across languages. Current practices face a scaling challenge: creating non-English benchmarks often involves machine translation followed by costly human verification. SubLIME-M adapts the core sampling strategies of SubLIME to efficiently evaluate models in diverse linguistic settings, ensuring consistent performance assessments without having to create overwhelming machine translated data which requires expensive human verifications. In this work, we make the following contributions:
We introduce **SubLIME***, an extensible, data-efficient evaluation framework that maintains model ranking and score distributions while drastically reducing computational costs.
We design **SubLIME-I** for text-to-image models, employing adaptive sampling techniques that, in our experiments, preserve high correlation with full benchmark evaluations while reducing data usage by up to 90%. Our analysis shows that the HEIM leaderboard construction for 27 models, which originally required 4 months, can be dramatically to a single day.
We present **SubLIME-C**, a domain-specific extension that utilizes LLM-based similarity analysis to remove redundant evaluation instances across multiple benchmarks.
We develop **SubLIME-M** for multilingual benchmarks, demonstrating effective subset selection methods that yield robust performance assessments across a range of languages.

2 Background and Related Work

Data efficient training was widely studied for model training on image Ding et al. (2023); Sorscher et al. (2023), and language tasks Marion et al. (2023); Xie et al. (2023). Methods includes coreset selection, importance sampling, and difficulty sampling to use smaller, representative datasets Zayed et al. (2023); Guo et al. (2022). SubLIME* explores diverse sampling strategies in LLM and text-to-image model evaluation, aiming to maintain model rankings and score distributions.

Efficient LLM evaluation was recently introduced in techniques like AnchorPoints Vivek et al. (2024) and TinyBenchmarks Polo et al. (2024) which use coreset and item response theory (IRT) to select a subset of evaluation instances, closely estimating full benchmark scores. These methods can be considered for integration into SubLIME*. SubLIME-C explores cross-benchmark redundancies, while SubLIME-I applies similar approaches to text-to-image generation models. Lifelong Benchmarks Prabhu et al. (2024), expands candidate examples in benchmarks on an ongoing basis and selects a subset of examples based on their difficulty ranking, for evaluating new models. For new examples, difficulty estimates are based on evaluation using a selected subset of models. Flash-HELM Perlitz et al. (2024) optimizes evaluation resources based on estimated leaderboard positions, prioritizing higher-ranked models. Our approach, which evaluates all models on a sampled subset, complements FlashHELM's strategy.

Multimodal and multilingual foundation model evaluation adds further challenges to traditional text-based, English-centric models. The recent success and quality of diffusion models in tasks such as audio, image and video generation comes at a cost of efficiency. Denoising schedules during inference significantly increase generation times Ye et al. (2025), which has a direct impact in the speed and cost of evaluation of these types of models. Something similar happens in code generation tasks, where most benchmarks focus on code accuracy Chen et al. (2021a); Zhuo et al. (2024), and few look at efficient code generation Liu et al. (2024). The issue with evaluation inefficiency in both scenarios scales up with multilingualism. For instance, HumanEval Chen et al. (2021a) uses just 164 prompts to evaluate Python performance with English prompts. Adding the programming language dimension, McEval Chai et al. (2024) includes 16,000 prompts to evaluate models perfromance in 40 programming languages. Covering multilingual prompting, HumanEval-XL Peng et al. (2024) scales down to 12 programming languages, but evaluates across 23 natural languages, which results in over 22,000 prompts. Another significant issue is the scarcity of data in low-resource languages. This scarcity often leads to models under-performing in mid- and low-resource languages compared to high-resource ones Chang et al. (2023), and to the challenge of differentiating actual multilingual capabilities from a models' ability to generalize a task evaluation format or objective from English to other languages Aula-Blasco et al. (2025). In addition, language-specific nuances, such as idiomatic expressions and cultural references, require models to have a deep understanding of each language's unique characteristics. Standard evaluation metrics and the practice of translating evaluation datasets can introduce artifacts and cultural biases, potentially distorting the assessment of a model's true capabilities across different languages Singh et al. (2024). Some recent work has tackled this issue by integrating a strong human-in-the-loop approach to multilingual benchmark creation Bandarkar et al. (2024); Baucells et al. (2025). Our approach is an initial test on how to leverage these quality-oriented efforts while reducing the cost of evaluation.

3 Our Solution

Our solution is built on top of DELE Saranathan et al. (2024), inspired by real-world examples like the International Mathematical Olympiad which identifies top talents with only 6 problems. By leveraging dataset redundancy, we select representative subsets that preserve model ranking and score distributions, using statistical measures like Spearman correlation to ensure alignment with the complete dataset. Diverse sampling methods S are used:
Random Sampling serves as the baseline, wherein we select 1% to 100% of the dataset in 1% increments (using fixed random seeds). This straightforward approach offers a simple, unbiased way to compare performance across LLMs and helps calibrate more sophisticated methods.
Clustering-based Sampling including both topic modeling and spectral clustering techniques are explored to ensure diverse coverage of semantic structures. Topic modeling employs Non-Negative Matrix Factorization (NMF) and Latent Dirichlet Allocation (LDA) over TF-IDF representations Luhn (1958) to extract topical clusters, from which we stratify samples to capture broad thematic variety. Spectral clustering leverages embeddings such as MTEB Muennighoff et al. (2022), BERT Walkowiak & Gniewkowski (2019), or K-means Hu et al. (2021) to cluster data points in a latent space and sample representatives from each cluster to preserve distributional properties.
Quality-based Sampling A high-quality subset prioritizes instances with clearer and more coherent text, reducing noise and ambiguity. This includes indicators to minimize spelling errors Hu et al. (2024) for text clarity, maintain an optimal average word length Bochkarev et al. (2012) to balance complexity with readability, and promote lexical diversity Zenker & Kyle (2021) by selecting text segments rich in vocabulary. These methods ensure that only high-quality items are retained, reducing confounding factors and improving the stability of model comparisons.
Difficulty-based Sampling considers text complexity measures such as *Gunning Fog* and *SMOG*. The *Gunning Fog Index* estimates readability based on sentence length and the proportion of complex words (3 or more syllables), indicating how difficult a passage is to comprehend. The *SMOG* grade similarly estimates the education level needed to understand a text. By selecting subsets that span a range of these scores, this approach ensures a balanced distribution of difficulty levels.

The algorithms of SubLIME* are described in Algo 1. Comparing its base version to DELE, we added more sampling methods including *Flesch* readability Flesch (1948) (Quality) and *SMOG* (Difficulty) and clustering with SOTA embeddings models such as SFR-Embedding-Mistral Rui Meng (2024). We use Spearman coefficient for a more relevant measure of rank correlation and Pearson coefficient for score preservation between subset and fullset evaluations.

3.1 SubLIME-I: Applying SubLIME to Text-to-Image Models

Text-to-image model evaluation demands significant computational resources because it iteratively refines noisy inputs over many neural network passes to generate a coherent image. To address this, we introduce SubLIME-I (Image), an extension of SubLIME, which employs adaptive sampling strategies, tailored for the unique requirements of Text-to-Image models. By analyzing the characteristics of each benchmark, SubLIME-I dynamically selects the optimal sampling method, ensuring representative and high-quality subsets. Our experiments, covering 27 models including *DALLE* and *Stable Diffusion*, utilize HEIM leaderboard Hugging Face (2022) benchmarks across various datasets like *MSCOCO, Cub200,* and *I2P*.

Algorithm 1 SubLIME*: subset sampling algorithms across languages and benchmarks

```
Require: Initialize
Specify X: SubLIME(Baseline), SubLIME-I, SubLIME-C, SubLIME-M
Target ($\mathcal{B}$) benchmarks from public or private Leaderboards
Collect sample-level results for ($\mathcal{M}_b$) models
Sampling based on random, quality, clustering, difficulty
$d_b \subset b \forall \in B$; ($d_b$ is subset of b)
Ensure: Adaptive Sampling for each Benchmark
if $X$ is SubLIME-C then
    Identify candidate problem pairs based on cosine similarity using an embedding model
    Flag pairwise data $(D_M, D_N)$ across $\forall b$ in $B$ using LLM based redundancy analysis
    $d_R \leftarrow \forall D_M, D_N$, select one by comparing the intra-latent representation of candidate to the centroid
    Update $d_b \leftarrow d_b$ - $d_R$
end if
for each benchmark $b \in \mathcal{B}$ do
    for each sampling technique $s \in \mathcal{S}$ do
        for sampling rate $x\%$ from 1 to 100 at step size 1% do
            if $X$ is SubLIME-M (Multilingual) and $b$ in non-English then
                use the same $I_{s,x}$ from $b$ in English
            else
                $I_{s,x} \leftarrow$ apply $s$ to get indices of $x\%$ samples of $d_b$
            end if
            $d_{s,x} \leftarrow d_b[I_{s,x}]$ (subset using $I_{s,x}$)
            $\mathbf{score}_{s,x} \leftarrow \{\text{eval}(m, d_{s,x}) \mid m \in \mathcal{M}_b\}$
            $\mathbf{rank}_{s,x} \leftarrow \text{argsort}(\mathbf{score}_{s,x})$
            $\tau_{s,x} \leftarrow \rho(\mathbf{rank}_{s,x}, \mathbf{rank}_{\text{full}})$          ▷ Spearman
            $\rho_{s,x} \leftarrow \rho(\mathbf{score}_{s,x}, \mathbf{score}_{\text{full}})$          ▷ Pearson
            $R_s[x] \leftarrow (\tau_{s,x}, \rho_{s,x})$
        end for
    end for
    Analyze $R_s$ to find minimal $x$ where $\tau \geq 0.9$ and $\rho \geq 0.9$
end for
$s_b^* \leftarrow \arg\min(s) \in \mathcal{S}x$
Ensure: Testset Validation
Using selected $I_{s,x}$ evaluate the Test models for Rank and Score Correlation Preservation
return Return optimal sampling subset $Sub \leftarrow \{(s_b^*, x_b^*)\}_{b=1}^{10}$
```

3.2 SUBLIME-C: ELIMINATING CROSS-BENCHMARK REDUNDANCY

Evaluation across multiple benchmarks sometimes involves significant redundancy, particularly within specific categories such as coding, math, reasoning, and sentiment analysis etc. SubLIME-C addresses this through a hybrid approach that combines embedding-based semantic search with LLM-guided conceptual analysis to eliminate redundant evaluation instances across benchmarks.

Candidate Selection: We first pre-select candidate pairs of coding problems by computing embedding-based similarity. This initial filtering reduces the search space by flagging pairs with similarity scores above a predetermined threshold.

LLM-Based Similarity Analysis: For each candidate pair, we construct a prompt, listed in Appendix Section C, that instructs the LLM to assess the conceptual and algorithmic similarity between the two problems. The conceptual evaluation focuses on the underlying problem statements and algorithmic challenges rather than superficial differences (e.g., coding style or minor implementation details). In return, the LLM generated a structured output with a `similarity_score` (ranging from 0 to 3) and a concise `justification`. We then filter pairs by retaining only those with a score above a threshold (e.g., $\geq$ 2.0). This LLM-based analysis leverages the reasoning abilities of LLMs such as OpenAI o3-min-high OpenAI and DeepSeek-R1 DeepSeek-AI et al. (2025) to assess the nuanced conceptual similarities between coding problems—going well beyond conventional semantic similarity metrics.

Removing problem from redundant pair during the sampling: when removing duplicates (Algorithm 1), we retain the problem closest to its benchmark's centroid in the latent space. This preserves each benchmark's distinctive characteristics while eliminating truly redundant items from the selected coding benchmark.

3.3 SubLIME-M: Extending SubLIME to Multilingual Benchmarks

Evaluating models across multiple languages is increasingly important as language models are deployed globally. Current multilingual evaluations face two challenges: (1) Most high-quality benchmarks remain English-centric due to the prohibitive costs of creating native non-English evaluation data, and (2) Translated benchmarks often introduce artifacts through machine translation pipelines that require expensive human verification—a process particularly burdensome for low-resource languages with scarce labeling resources. **SubLIME-M** addresses these challenges through subset selection, enabling data-efficient evaluations while minimizing translation dependency.

The core idea behind SubLIME-M is straightforward: as shown in Algo 1, we first run SubLIME on the English version of the benchmark to optimal subset indices, and then we directly map these indices to the corresponding samples in the non-English versions of the benchmark. This approach assumes that the structure and semantic content of the evaluation data are preserved across translations or localized versions, thereby ensuring that the selected subset is representative in all languages. Finally, we evaluate the models using these aligned subsets. By doing so, we preserve the ranking while reducing the amount of data that needs to be processed across all languages.

4 Experiment Results and Discussions

We assess various sampling techniques' effectiveness in reducing the benchmark time while maintaining rankings using a subset of the complete dataset. Using our proposed method, as outlined in 1, we aim to dynamically pinpoint the best sampling approach for each benchmark.

4.1 Experimental Setup and Design

SubLIME(Baseline) setup: 10 Benchmarks from HuggingFace OpenLLM leaderboard i.e. GSM8K, MATH, ARC, etc. Evaluated across 313 LLMs, such as *Qwen, Llama* etc.

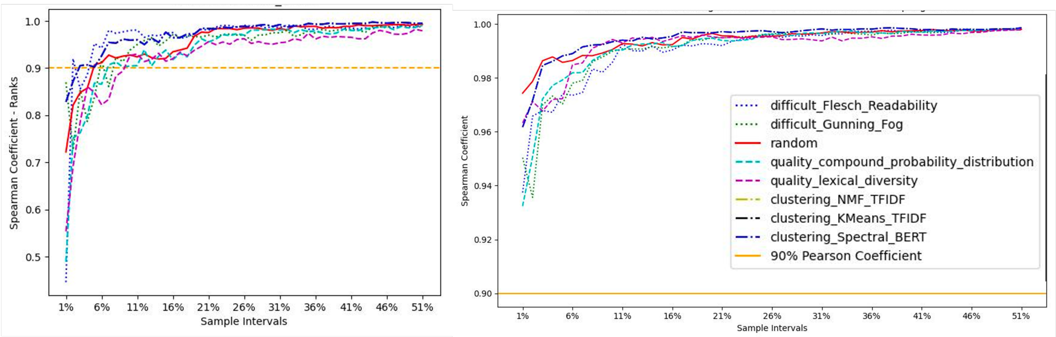
SubLIME-I setup: 17 Benchmarks from HEIM leaderboard i.e. MSCOCO, CUB200, I2P, etc. Evaluated across 27 text-to-image Models, such as *Stable Diffusion, DALL-E 2* etc.

SubLIME-C setup: Coding leaderboard based on HumanEval Chen et al. (2021b) and MBPP Google Research (2022). Evaluated 19 LLMs with sample-level results on our GPU clusters.

SubLIME-M setup: ARC in English and Catalan, and OpenBookQA in English, Spanish, Catalan, and Galician languages Baucells et al. (2025). Evaluated 21 LLMs which can handle at least 3 of these languages based on the description of their model cards.

4.2 Results on SubLIME-I

Our empirical analysis demonstrates the efficacy of SubLIME-I (SubLIME for Text-to-Image) in preserving model rankings while significantly reducing the evaluation effort. The key results from our experiments are as follows:
Data efficiency: Using only 1-5% of the data, SubLIME-I maintains high Spearman correlation coefficients across various text-to-image benchmarks, as shown in Figure 15 for clip-score and 1(a) aesthetics-score leaderboards for DrawBench Saharia et al. (2022) benchmark (SubLIME for some of the remaining benchmark plots is depicted in Appendix C.2). Techniques like difficulty-based and quality-based sampling proved particularly effective in identifying representative subsets within 5-10% of data that preserve the integrity of model rankings and score distributions.
Consistent Performance Across Benchmarks: We performed performance analysis of SubLIME-I across multiple Text to Image benchmarks (16 benchmarks) by considering the average winrate of different models across multi-benchmarks. Figure 1(b) for aesthetics and Figure 14 for clip score leaderboard, demonstrates that SubLIME-I maintains strong average win-rate correlation ($\tau > 0.95$) across varying sampling percentages (10%–100%), and even we also observed $\tau > 0.9$ for data within 1% - 5% for many sampling methods, indicating potential for remarkable evaluation time



(a) SubLIME-I on Draw Bench Benchmark

(b) Average Win Rate Rank Correlation on Aesthetics Score Leaderboard

Figure 1: SubLIME-I: (a) Rank and Score Preservation for Individual draw_bench benchmark on - max-aesthetics-score Leaderboard; (b)Average winrate vs Rank Correlation of 27 Text-to-Image models on all 16 benchmarks on Aesthetics Score Leaderboard

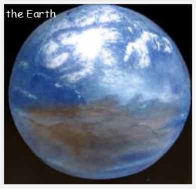
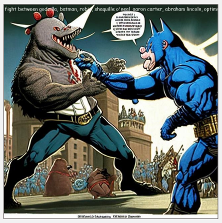
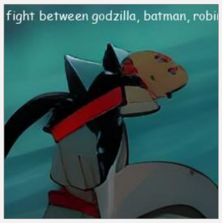
Difficulty Level	Input Text	Benchmark	Top Performing Model from HEIM Leaderboard huggingface_dreamlike-diffusion-v1-0	Least Performing Model from HEIM Leaderboard kakaobrain_mindall-e
Easiest Sample Gunning Fog Score – 0.0067	“the Earth”	Parti-prompts		
Hardest Sample Gunning Fog Score – 0.929	"fight between godzilla, batman, robin, shaquille o'neal, aaron carter, abraham lincoln, optimus prime, mcgruff the crime dog, jackie chan, indiana jones, a care bear, chuck norris, gandalf the grey, gandalf the white, monty python and the holy grail's black knight, benito mussolini, the blue meanie, cowboy curtis, jambi the genie, robocop, the terminator, captain kirk, darth vader, lo pan, superman, every single power ranger, bill s. preston, ted logan, spock, dwayne'the rock'johnson, doctor octopus, hulk hogan, and mister rodgers”	I2P		

Figure 2: SubLIME-I: Example easy and hard sample identification: To show the advantage of metric-based stratified sampling for rank preservation of models

and cost reduction regardless of benchmark selection, while maintaining the rank and score distribution performance. This pattern persists across diverse datasets including MSCOCO, CUB200, and Wikimedia, with full metrics available in Appendix C.2, showing SubLIME could be useful for individual text to image benchmarks.

Time savings: Our analysis of the HEIM leaderboard log files shows that evaluating 27 models on its benchmarks required a total of 2,792 hours of inference time. As shown in Figure 1b, this time is estimated to be reduced to just 28 hours by selecting only 1% of the data, while still maintaining a Spearman correlation greater than 0.9.

SubLIME-I Illustration Figure 2 provides an intuitive demonstration of the core idea behind SubLIME-I. Our framework leverages metric-based stratified sampling—using criteria such as difficulty, quality, and diversity—to construct representative subsets of evaluation data. In the figure, we compare outputs from a top-performing and a bottom-performing text-to-image model on a sample drawn from only 10% of the full HEIM leaderboard dataset. For a prompt with a high difficulty score, the best model generates an image that faithfully captures the complex context, whereas the lowest-performing model fails to do so. Conversely, when the prompt is easier, both models produce comparable outputs. This contrast underscores the motivation for SubLIME-I: by ensuring that our sampled subset spans a broad range of prompt difficulties and other quality metrics, we preserve the relative performance differences among models.

4.3 Results on SubLIME-C

Table 1: Match rates for redundant pairs identified by semantic search with different cosine distance thresholds, and LLM-based similarity analysis in HumanEval and MBPP benchmarks.

Method	# of problem pairs	Match rate
Semantic Search (cos similarity $>$ 0.78)	1285	38.5%
Semantic Search (cos similarity $>$ 0.82)	281	66.2%
Semantic Search (cos similarity $>$ 0.78) + LLM Review	281	80.1%

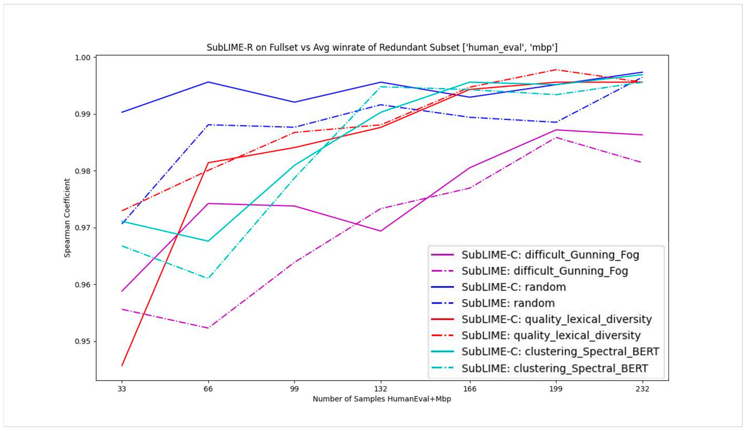


Figure 3: SubLIME-C: Rank Correlation based on average win-rate across MBPP and HumanEval for SubLIME(baseline) and SubLIME-C(Cross-redundancy removed) subsets

We compare SubLIME-C against our baseline SubLIME on a private coding leaderboard encompassing two popular benchmarks, HumanEval Chen et al. (2021b) and MBPP Google Research (2022), evaluated across 19 LLMs. Our goal is to preserve rank correlation while reducing redundancy in the joint evaluation of both benchmarks. Below, we detail our steps and main observations:
1. We evaluated 19 LLMs on both the HumanEval and MBPP benchmarks, and saved the rigorous evaluation results in a comprehensive dataset, including sample-level LLM generation outputs and code execution statuses.
2. To identify cross-benchmark redundancies, SubLIME-C combines *semantic search* (using cosine similarity on SFR embeddings) with an *LLM-based* review step that prompts a model to assess conceptual overlap between coding tasks. Table 1 summarizes our ablation on different methods:

- At a cosine-similarity threshold of 0.78, semantic search flagged 1,285 problem pairs as potential duplicates, with a *38.5%* “true" match rate. Using a *higher* threshold of 0.82, the match rate increased to *66.2%* with remaining 281 problem pairs.
- When refining the same 1,285 candidate pairs with an additional LLM-based review, the match rate increased to *80.1%*—**significantly higher** than semantic-only filtering at the same final count of problem pairs. This indicates that SubLIME-C identifies *more truly* redundant coding problems.

3. Figure 3 illustrates the Spearman correlation vs the total number of samples when preserving average win-rate across HumanEval and MBPP. Notably, SubLIME-C consistently achieves higher correlation for a given sample size compared to baseline SubLIME methods (e.g., random or clustering-only). In many cases, SubLIME-C surpasses a 0.95 rank correlation with *10–20%* fewer samples, underscoring the benefits of removing cross-benchmark redundancies. Overall, these results confirm that SubLIME-C delivers more *data-efficient* evaluations by eliminating needless repetition.

4.4 Results on SubLIME-M

In this section, we conducted SubLIME-M experiments on two benchmarks with multilingual variants: OpenBookQA (in English, Spanish, Catalan, and Galician) and ARC Challenge (in English and

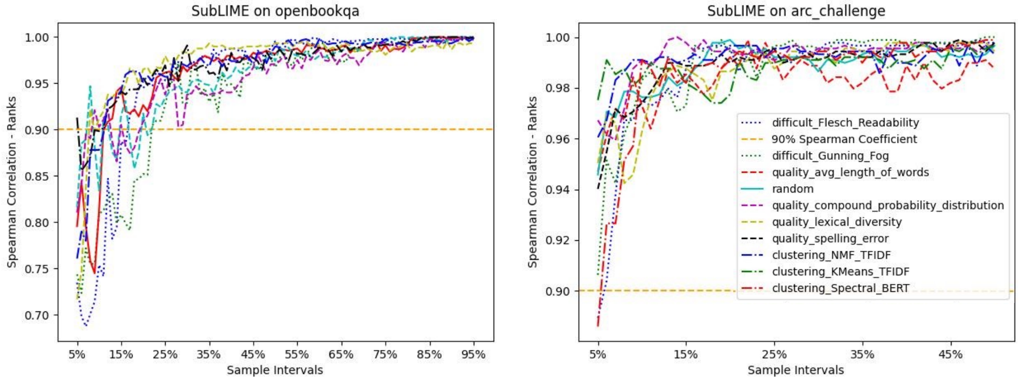


Figure 4: SubLIME-M (Baseline) for 21 models used to select best subset across different sampling rate to be used in test set validation for (a) OpenBookQA; (b) ARC Challenge

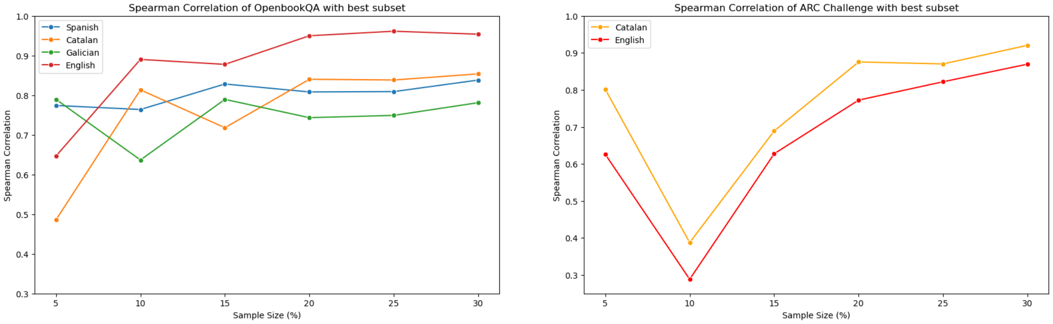


Figure 5: SubLIME-M: Multi-lingual Evaluation on Selected Subset from SubLIME for (a) OpenBookQA; (b) ARC Challenge

Catalan). Our goals are two fold: (i) to assess whether the subsets selected by **SubLIME-M** preserve rank correlations similarly across multiple languages, and (ii) to identify potential language-specific limitations or idiosyncrasies.

Figure 5a depicts the Spearman correlation between model ranks on various percentage subsets (from 5% to 30%) and the full dataset for English, Spanish, Catalan, and Galician. Notably, *English* achieves consistently high correlations (above 0.9) already at around 10–15%, while Spanish hovers around 0.9 by 20%. In contrast, Catalan and Galician exhibit more variability, requiring slightly larger subsets (often above 20%) to stabilize at correlations beyond 0.8–0.85. Despite these differences, *all four languages* benefit from smaller evaluation subsets (e.g., 15–25% of the data), preserving the model ordering at a high fidelity while drastically reducing evaluation overhead.

Figure 5b shows the Spearman correlation curves for English and Catalan on the ARC Challenge dataset. We observe more pronounced fluctuations at lower sampling rates (around 5–10%), including a sharp dip for both languages near the 10% mark. As the sample size increases beyond 15%, the rank correlations recover steadily, exceeding 0.8 for Catalan and 0.75 for English by 20–25%. We hypothesize that these oscillations arise from the benchmark's narrower topical coverage, which makes certain small subsets disproportionately challenging (or too easy) relative to the full distribution. Nonetheless, with at least 20% of the data, **SubLIME-M** still achieves robust rank preservation (> 0.8) across both languages. We attribute some variations to the following aspects which require further investigation:
Translation quality and artifacts. In lower-resource languages, evaluation data often relies on machine translations or partial human curation. In the case of OpenBookQA in Galician, the dataset was automatic translated with human revision. Though we cannot assume any problems with the revision process, it could be that resulting translations were not localized or had an English-like

structure that, though it may be correct in Galician, fails at being fully natural. These aspects may introduce distributional biases that complicate the sampling strategies (e.g., clustering or difficulty-based methods) employed by SubLIME-M.
Language-specific difficulty / quality / clustering sampling: What is considered an “easy” prompt in English may become more linguistically complex or culturally unfamiliar in another language, undermining the assumption that difficulty or quality indicators transfer seamlessly. Moreover, measures such as readability scores—tuned primarily for English—may not accurately capture complexity in Spanish, Catalan, or Galician, leading to skewed subset selections if used directly across languages. A certain embedding model may not cluster well for a minority language.

Despite these open questions, our preliminary results show that a single English-based sampling strategy still generalizes reasonably well to other languages, cutting down evaluation costs up to 80%. Moving forward, addressing language-specific challenges may involve: (i) leveraging more refined alignment between source and target language data (e.g., improved translation or specialized preprocessing for lower-resource languages), (ii) exploring cross-lingual embeddings and domain-adaptive sampling strategies that can reliably cluster prompts in non-English contexts, and (iii) incorporating human-in-the-loop quality checks or alternative readability metrics tailored to each language. Such efforts can mitigate distributional biases and linguistic mismatches, ultimately allowing SubLIME-M to more robustly account for cultural and linguistic nuances while maintaining its data-efficient advantages.

4.5 Testset Validation

The test set validation shows the effectiveness of SubLIME* on unseen models across different modalities and leaderboards, suggesting that the data could be effectively selected in order to perform evaluations efficiently while preserving the rank correlations.

1. Validation on HEIM Leaderboard (Text to Image Models)
We selected 27 text-to-image models from the HEIM leaderboard. Out of these, 17 models were used to choose the best subsets at different sampling rates using SubLIME-I. The remaining 10 models, listed in C.2, were used for test set evaluation. In Figure 6, we observe that beyond a 10% sampling rate, we achieve a rank correlation of 90% or higher. Note that this experiment was conducted on a limited set of models due to restricted access to sample-level results for the text-to-image models. In the future, it can be extended to a more comprehensive list of models.

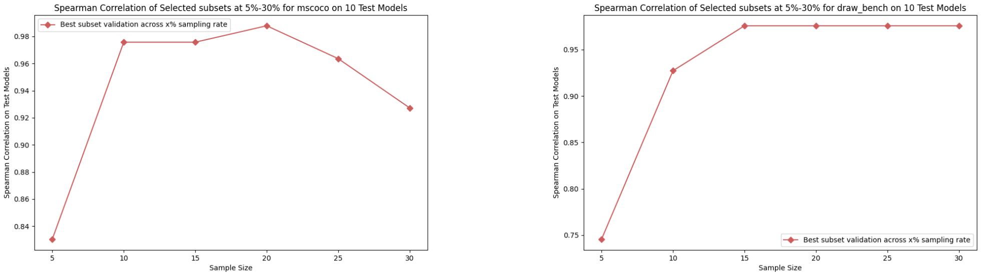


Figure 6: SubLIME-I: Testset Validation on 10 Models from HEIM Leaderboard for MSCOCO (left) and Draw Bench (right)

2. Validation on Latest HELM leaderboard
In this experiment, we conducted test set validation on 90 unseen LLMs from the latest HELM Lite Leaderboard HELM. We used the optimal subsets selected from the OpenBookQA benchmark during the SubLIME-M experiments. These winning subsets were identified by achieving high rank correlations with the full set rankings at a given sampling rate using the 21 LLMs from the "training" set. We assessed the selected subsets at various sampling rates—5-30%—on the 90 unseen LLMs. As shown in Figure 7, the optimal subsets derived from the 21 LLMs work effectively with LLMs in the test set.

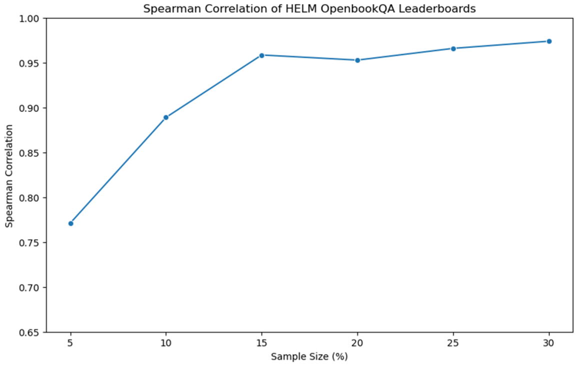


Figure 7: Testset Validation of OpenbookQA Benchmark on HELM Leaderboard Models

3. Validation on Open LLM Leaderboard

We obtained 313 LLMs from Open LLM Leaderboard to evaluate SubLIME(baseline) on 213 LLMs to obtain best subset for each sampling rate using adaptive sampling. We evaluate this selected subsets at 5-30% on 100 Test Models. The results are plotted for this in Figure 8 for GSM8K and TruthfulQA benchmarks. The graph shows that on the test set, even with 5% data, we were able to achieve ≥0.92 Rank Correlation across different benchmarks. At 10% we achieve much better results preserving upto 97% Ranks of the 100 models.

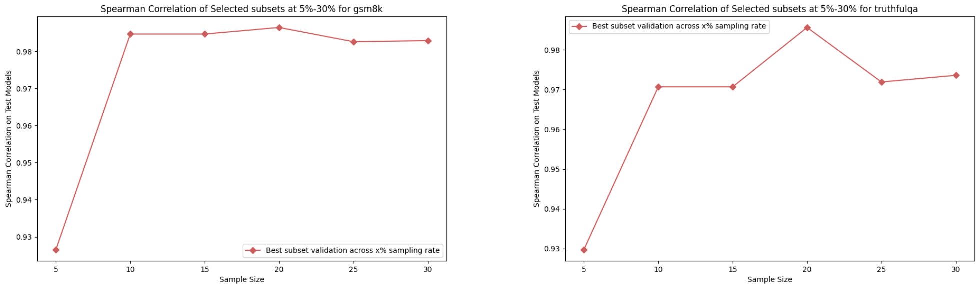


Figure 8: SubLIME (Baseline): Testset Validation on 100 Test Models using best subset selected for GSM8K (left) and TruthfulQA (right)

5 Conclusion

Our experiments demonstrate that the adaptive sampling strategies employed in SubLIME* yield significant benefits across a diverse set of benchmarks and modalities. In the text-to-image domain (SubLIME-I), representative subsets comprising as little as 1–5% of the full dataset were able to preserve model ranking correlations (with Spearman τ often exceeding 0.9) while reducing evaluation time from hundreds of hours to mere tens of hours. For coding benchmarks, SubLIME-C's integration of semantic search with an LLM-based review effectively eliminated cross-benchmark redundancies, achieving higher true match rates and enabling redundancy aware data-efficient evaluations without compromising rank integrity. Finally, our SubLIME-M experiments confirmed that winner subsets selected from an English benchmark can generalize to some of the unseen multilingual test benchmarks across different models from the HELM Leaderboard, ensuring consistent performance evaluations across languages.

Collectively, these results underscore the efficacy of our approach in reducing computational overhead while maintaining high evaluation fidelity. SubLIME* not only accelerates the evaluation process but also ensures that the relative performance of models is reliably preserved, paving the way for scalable and efficient benchmarking of large-scale foundation models in varied application domains.

References

Javier Aula-Blasco, Júlia Falcão, Susana Sotelo, Silvia Paniagua, Aitor Gonzalez-Agirre, and Marta Villegas. VeritasQA: A truthfulness benchmark aimed at multilingual transferability. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5463–5474, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.366/.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 749–775, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.44. URL https://aclanthology.org/2024.acl-long.44/.

Irene Baucells, Javier Aula-Blasco, Iria de Dios-Flores, Silvia Paniagua Suárez, Naiara Perez, Anna Salles, Susana Sotelo Docio, Júlia Falcão, Jose Javier Saiz, Robiert Sepulveda Torres, Jeremy Barnes, Pablo Gamallo, Aitor Gonzalez-Agirre, German Rigau, and Marta Villegas. IberoBench: A benchmark for LLM evaluation in Iberian languages. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 10491–10519, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.699/.

Vladimir Bochkarev, Anna Shevlyakova, and Valery Solovyev. Average word length dynamics as indicator of cultural changes in society. *Social Evolution and History*, 14:153–175, 08 2012.

Linzheng Chai, Shukai Liu, Jian Yang, Yuwei Yin, Ke Jin, Jiaheng Liu, Tao Sun, Ge Zhang, Changyu Ren, Hongcheng Guo, Zekun Wang, Boyang Wang, Xianjie Wu, Bing Wang, Tongliang Li, Liqun Yang, Sufeng Duan, and Zhoujun Li. Mceval: Massively multilingual code evaluation, 2024. URL https://arxiv.org/abs/2406.07436.

J. S. Chall and E. Dale. *Readability revisited: The new Dale-Chall readability formula*. Brookline Books, 1995.

Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K. Bergen. When is multilinguality a curse? language modeling for 250 high- and low-resource languages, 2023. URL https://arxiv.org/abs/2311.09205.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021a. URL https://arxiv.org/abs/2107.03374.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec

Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. 2021b.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL `https://arxiv.org/abs/2501.12948`.

Tianyu Ding, Tianyi Chen, Haidong Zhu, Jiachen Jiang, Yiqi Zhong, Jinxin Zhou, Guangzhi Wang, Zhihui Zhu, Ilya Zharkov, and Luming Liang. The efficiency spectrum of large language models: An algorithmic survey, 2023.

R. Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233, 1948.

Google Research. Mostly basic python problems dataset, 2022. URL `https://github.com/google-research/google-research/tree/master/mbpp`.

R. Gunning. *The technique of clear writing*. McGraw-Hill, 1952.

Chengcheng Guo, Bo Zhao, and Yanbing Bai. Deepcore: A comprehensive library for coreset selection in deep learning. *arXiv preprint arXiv:2204.08499*, 2022.

HELM. Helm lite leaderboard v.1.13.0. `https://crfm.stanford.edu/helm/lite/latest/#/leaderboard`. Accessed: 2025-02-08.

Wenhao Hu, Dong Xu, and Zhihua Niu. Improved k-means text clustering algorithm based on bert and density peak. In *2021 2nd Information Communication Technologies Conference (ICTC)*, pp. 260–264, 2021. doi: 10.1109/ICTC51749.2021.9441505.

Yifei Hu, Xiaonan Jing, Youlim Ko, and Julia Taylor Rayz. Misspelling correction with pre-trained contextual language model. 2024. doi: 10.1123/acl.2024.12345.

Hugging Face. Heim leaderboard. `https://crfm.stanford.edu/helm/heim/latest/#/leaderboard`, 2022.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=iO4LZibEqW. Featured Certification, Expert Certification.

Jiawei Liu, Songrun Xie, Junhao Wang, Yuxiang Wei, Yifeng Ding, and Lingming Zhang. Evaluating language models for efficient code generation, 2024. URL https://arxiv.org/abs/2408.06450.

H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958. doi: 10.1147/rd.22.0159.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining llms at scale, 2023.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022. URL https://arxiv.org/abs/2210.07316. Version 3.

OpenAI. Openai o3-mini. https://openai.com/index/openai-o3-mini/. Accessed: 2025-02-08.

Qiwei Peng, Yekun Chai, and Xuhong Li. HumanEval-XL: A multilingual code generation benchmark for cross-lingual natural language generalization. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 8383–8394, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.735/.

Yotam Perlitz, Elron Bandel, Ariel Gera, Ofir Arviv, Liat Ein-Dor, Eyal Shnarch, Noam Slonim, Michal Shmueli-Scheuer, and Leshem Choshen. Efficient benchmarking of language models, 2024.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: evaluating llms with fewer examples, 2024.

Ameya Prabhu, Vishaal Udandarao, Philip Torr, Matthias Bethge, Adel Bibi, and Samuel Albanie. Efficient lifelong model evaluation in an era of rapid progress. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=A7wC1CTkYl.

Shafiq Rayhan Joty Caiming Xiong Yingbo Zhou Semih Yavuz Rui Meng, Ye Liu. Sfr-embedding-mistral:enhance text retrieval with transfer learning. Salesforce AI Research Blog, 2024. URL https://www.salesforce.com/blog/sfr-embedding/.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Lit, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Raphael Gontijo-Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Gayathri Saranathan, Mahammad Parwez Alam, James Lim, Suparna Bhattacharya, Soon Yee Wong, Martin Foltin, and Cong Xu. DELE: Data efficient LLM evaluation. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2024. URL https://openreview.net/forum?id=I8bsxPWLNF.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, David I. Adelani, Jian Gang Ngui, Daniel Vila-Suero, Peerat Limkonchotiwat, Kelly Marchisio, Wei Qi Leong, Yosephine Susanto, Raymond Ng, Shayne Longpre, Wei-Yin Ko, Madeline Smith, Antoine Bosselut, Alice Oh, Andre F. T. Martins, Leshem Choshen, Daphne Ippolito, Enzo Ferrante, Marzieh Fadaee, Beyza Ermis, and Sara Hooker. Global mmlu: Understanding and addressing cultural and linguistic biases in multilingual evaluation, 2024. URL https://arxiv.org/abs/2412.03304.

John Smith and Lisa Johnson. Strategies for difficulty sampling providing diversity in datasets. *Journal of Machine Learning Research*, 10:100–120, 2020.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning, 2023.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1576–1601, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.eacl-long.95.

Tomasz Walkowiak and Mateusz Gniewkowski. Evaluation of vector embedding models in clustering of text documents. In Ruslan Mitkov and Galia Angelova (eds.), *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pp. 1304–1311, Varna, Bulgaria, September 2019. INCOMA Ltd. doi: 10.26615/978-954-452-056-4_149. URL https://aclanthology.org/R19-1149.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=uPSQv0leAu.

Zilyu Ye, Zhiyang Chen, Tiancheng Li, Zemin Huang, Weijian Luo, and Guo-Jun Qi. Schedule on the fly: Diffusion time prediction for faster and better image generation, 2025. URL https://arxiv.org/abs/2412.01243.

Abdelrahman Zayed, Prasanna Parthasarathi, Gonçalo Mordido, Hamid Palangi, Samira Shabanian, and Sarath Chandar. Deep learning on a healthy data diet: finding important examples for fairness. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i12.26706. URL https://doi.org/10.1609/aaai.v37i12.26706.

Fred Zenker and Kristopher Kyle. Investigating minimum text lengths for lexical diversity indices. *Assessing Writing*, 47:100505, 2021. ISSN 1075-2935. doi: https://doi.org/10.1016/j.asw.2020.100505. URL https://www.sciencedirect.com/science/article/pii/S1075293520300660.

Terry Yue Zhuo, Minh Chien Vu, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, Simon Brunner, Chen Gong, Thong Hoang, Armel Randy Zebaze, Xiaoheng Hong, Wen-Ding Li, Jean Kaddour, Ming Xu, Zhihan Zhang, Prateek Yadav, Naman Jain, Alex Gu, Zhoujun Cheng, Jiawei Liu, Qian Liu, Zijian Wang, David Lo, Binyuan Hui, Niklas Muennighoff, Daniel Fried, Xiaoning Du, Harm de Vries, and Leandro Von Werra. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions, 2024. URL https://arxiv.org/abs/2406.15877.

A Appendix / Supplemental Material

B SubLIME (Baseline) Results

We perform efficient subset sampling on 10 widely used LLM Benchmarks such as GSM8K, ARC, Winogrande, GPQA, Math, MUSR, etc which are evaluated across multiple leaderboards. We enhance subset selection by incorporating metrics that identify the data characteristics of a given benchmark.

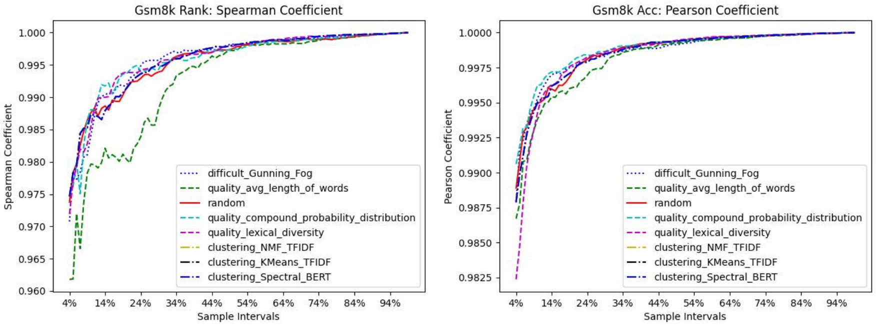


Figure 9: SubLIME (Baseline) - Rank and Score Correlation on GSM8K Benchmark

B.1 Difficulty Sampling Methods

The Difficult Words Percentage approach defines a list of over 3000 words known to 4th-grade students, flagging words outside this list as challenging. Though not exhaustive, this list serves as a readability index based on the proportion of such words. The Dale Chall Formula Chall & Dale (1995) assesses text readability by considering the number of difficult words and text length. The Flesch Reading Ease score Flesch (1948) quantifies readability based on sentence length and word complexity. The Gunning Fog index Gunning (1952) evaluates text complexity through average sentence length and complex words. These indices help in curating a dataset that not only challenges the model across a spectrum of complexity levels but also targets a wider distribution of metrics, enabling a more comparative analysis of LLM performance. Difficulty based sampling approach involves selection of samples from a dataset according to their perceived level of difficulty, assessed using readability indices Smith & Johnson (2020).

$$\text{Dale-Chall Formula} = 0.1579 \left(\frac{\text{Difficult Words}}{\text{Total Words}} \times 100 \right) + 0.0496 \left(\frac{\text{Total Words}}{\text{Total Sentences}} \right),$$

$$\text{Flesch Reading Ease} = 206.835 - 1.015 \cdot \text{Average number of words per sentence} - 84.6 \cdot \text{Average number of syllables per wo}$$

$$\text{Gunning Fog Index} = 0.4 \left(\frac{\text{words}}{\text{sentence}} + 100 \cdot \frac{\text{complex words}}{\text{words}} \right).$$

Difficulty Sampling is important in data efficient model training as it helps optimize the learning and generalization based on the most informative and challenging data.

C LLM Prompt Template for Coding Redundancy Analysis

Listing 1: LLM Prompt Template for Comparing Coding Problems

```
You are a coding problem analysis expert. Your task is to evaluate the
    similarity between two coding problems, along with their reference
    solutions. Focus primarily on the conceptual problem statements, the
    underlying algorithmic challenges, and the overall intended
    difficulty. The provided reference solutions are meant to help gauge
    difficulty and potential pitfalls, but differences in coding style or
     minor implementation details should be downplayed unless they
    indicate a significant change in the problem's core requirements.

Below are the details for the two problems:

---------------------------
Problem A:
{problem_A}

Reference Solution A:
{solution_A}
```

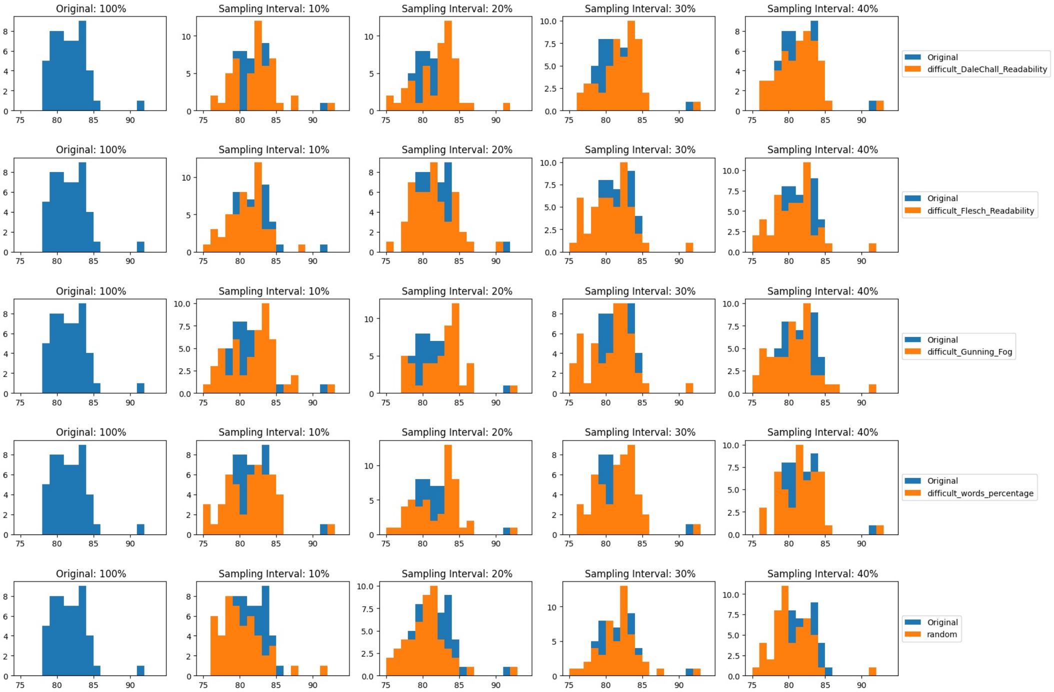


Figure 10: Four difficulty sampling on Winograde showing wider score distributions compared to original results

```
---------------------------
Problem B:
{problem_B}

Reference Solution B:
{solution_B}

---------------------------

The goal is to determine how likely it is that a tester who has solved
    Problem A would also be capable of solving Problem B, based on the
    conceptual similarities and difficulty of the problems.

Please rate the similarity on a scale from 0 to 3 according to the
    following guidelines:
- **3:** The problems are extremely similar in terms of conceptual
    requirements, problem setting, and overall difficulty. They are
    nearly redundant for a tester.
- **2:** The problems are very similar; solving one strongly indicates
    the ability to solve the other.
- **1:** There is only a little relevance between the problems.
- **0:** There is no relevance between the problems at all.

**Important Guidance:**
- **Primary Focus:** Evaluate the similarity based on the problem
    statements and underlying concepts.
- **Secondary Focus:** Consider the reference solutions as indicators of
    difficulty or potential pitfalls. Do not let differences in coding
    style or minor implementation choices affect your assessment. For
    instance, if both problems require the same core concept, such as
    converting a decimal number to its binary representation, do not view
     the use of built-in functions in one solution and manual logic in
    another as indicators of differing difficulty between the problems
    themselves.
```

~~~
Provide a brief justification explaining your reasoning for the chosen
    score.

Your final answer should be in JSON format with exactly two keys:
- "similarity_score": (an integer from 0 to 3)
- "justification": (a concise explanation)

**IMPORTANT:** Only output valid JSON. Do not include any additional text
    .

Example output:
```json
{
  "similarity_score": 2,
  "justification": "Both problems require similar data structure
      manipulations, although Problem B introduces an extra constraint
      that slightly increases its complexity."
}
```

Now, please analyze the provided problems and output your answer.
~~~

C.1 Analysis of Sampling for 57 subjects in MMLU

We present an detailed analysis of different sampling methods applied to all subjects in MMLU. An example on the Law subject is shown in Figure 12 where *Spectral MTEB* performs the best among all methods, and in Figure 13 Quality CPD performs best. The subjects in domains such as Figure 11 are also included here which achieves good rank preservation at lower sampling rate.

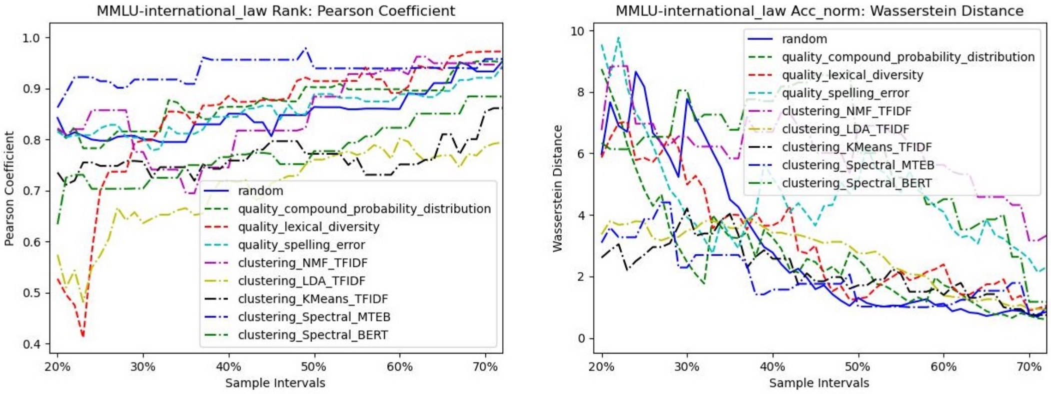


Figure 11: International Law: Rank and Accuracy (normalized) distribution preservation

Adaptive Sampling evaluates the performance of various sampling techniques across the 57 subjects as shown in Table 2. Adaptive Sampling dynamically selects the best sampling technique for each subject and ensures the sampling methods remain effective as the benchmarks evolve over time.

C.2 SubLIME-I - Experiments

SubLIME from Aesthetic and Clip Score leaderboard across various benchmarks are shown below.

Below are the list of test models from HEIM leaderboard which were used in evaluating the best subsets.

1. craiyon_dalle-mini
2. openai_dalle-2

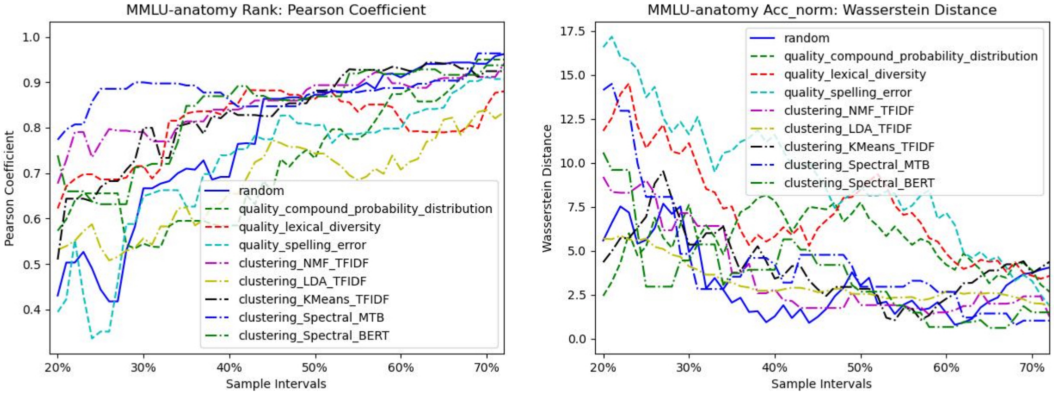


Figure 12: Anatomy: Rank and Accuracy (normalized) distribution preservation

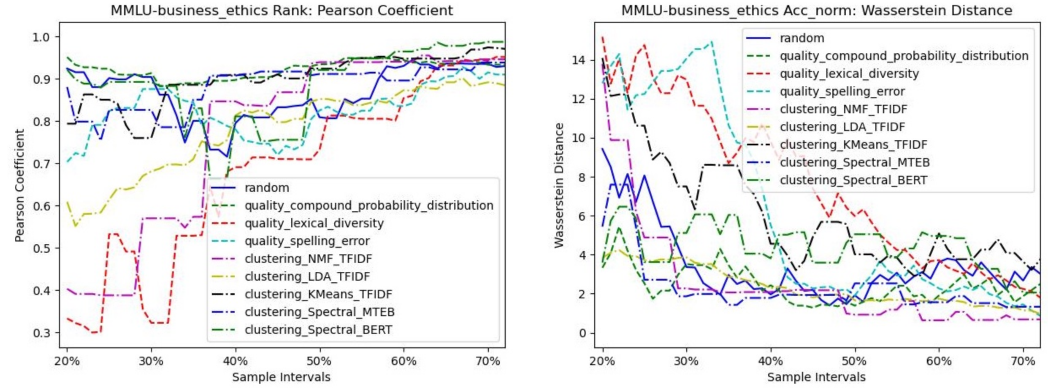


Figure 13: Business Ethics: Rank and Accuracy (normalized) distribution preservation

3. huggingface_dreamlike-photoreal-v2-0
4. huggingface_openjourney-v2-0
5. huggingface_stable-diffusion-v1-5
6. craiyon_dalle-mega
7. huggingface_stable-diffusion-v2-base
8. huggingface_stable-diffusion-safe-weak
9. huggingface_stable-diffusion-safe-medium
10. huggingface_vintedois-diffusion-v0-1

C.3 SubLIME-C - Experiments

SubLIME on Individual Benchmarks Results are given below, this was performed before the cross benchmark evaluation. The Rank and Score preservation is performed on *MBP* and *HumanEval* using its result score. The redundancy is then remvoed with the help of GPT4.0 evaluator.

C.4 SubLIME-M - Experiments

Below are the list of 21 models which were chosen to generate the best subset

1. mistralai/Mixtral-8x7B-Instruct-v0.1
2. tiiuae/falcon-40b
3. FelixChao/Scorpio-7B

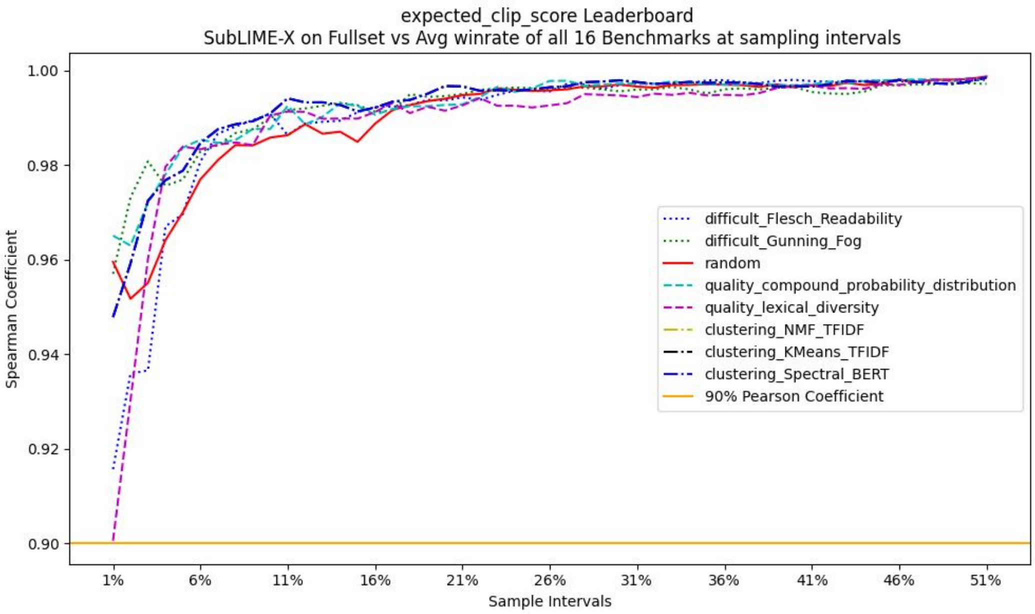


Figure 14: Average winrate vs Rank Correlation of 27 Text-to-Image models on all 16 benchmarks on Clip Score Leaderboard

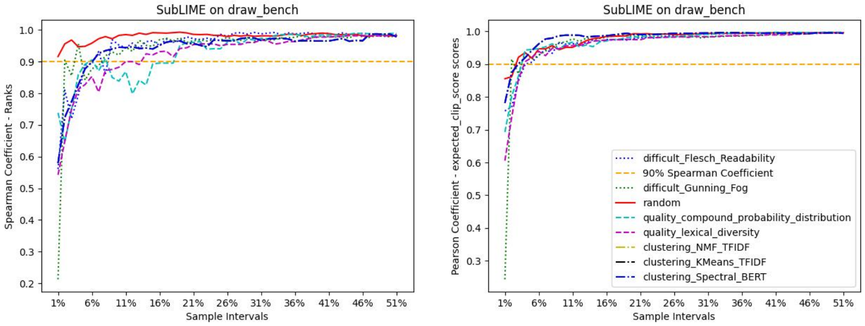


Figure 15: Rank and Score Preservation across multiple sample rate for draw_bench benchmark - expected-clip-score

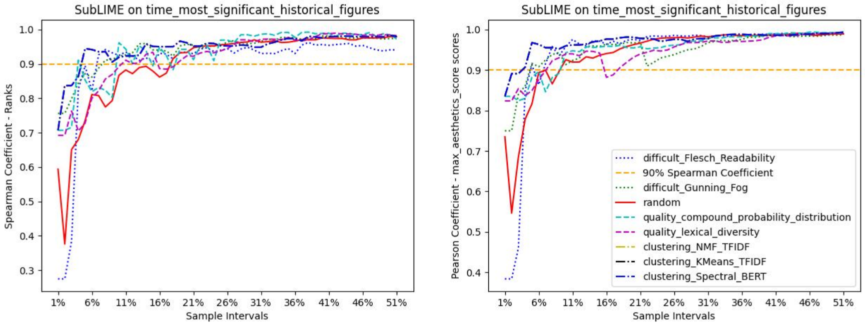


Figure 16: Rank and Score Preservation vs sampling ratio on most_significant_historical_figures Benchmark on Aesthetics scores

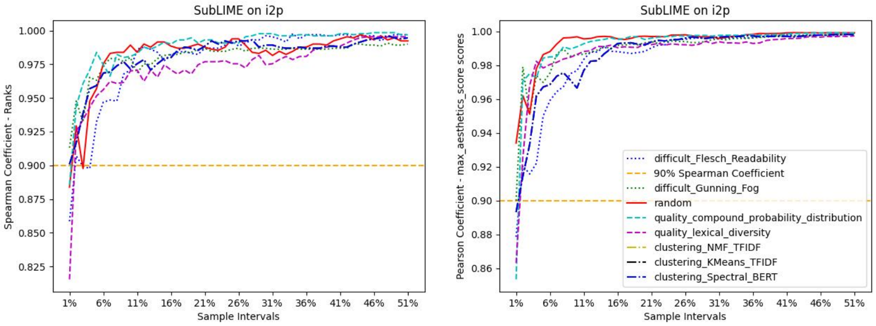


Figure 17: Rank and Score Preservation vs sampling ratio on I2P Benchmark on Aesthetics scores

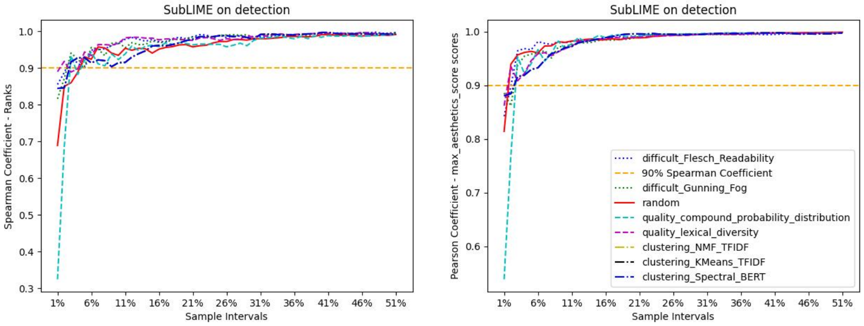


Figure 18: Rank and Score Preservation vs sampling ratio on detection Benchmark on Aesthetics scores

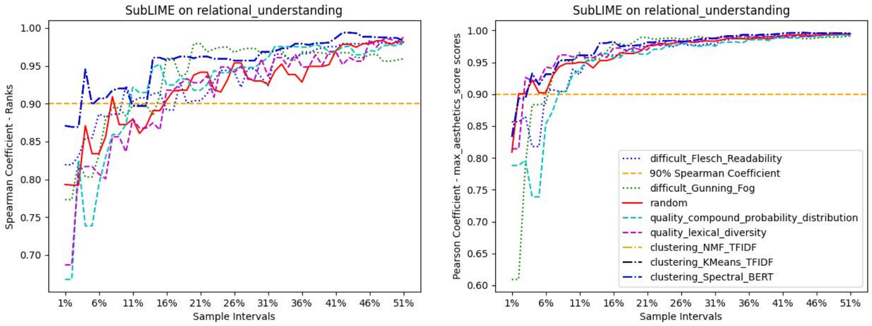


Figure 19: Rank and Score Preservation vs sampling ratio on Relational understanding Benchmark on Aesthetics scores

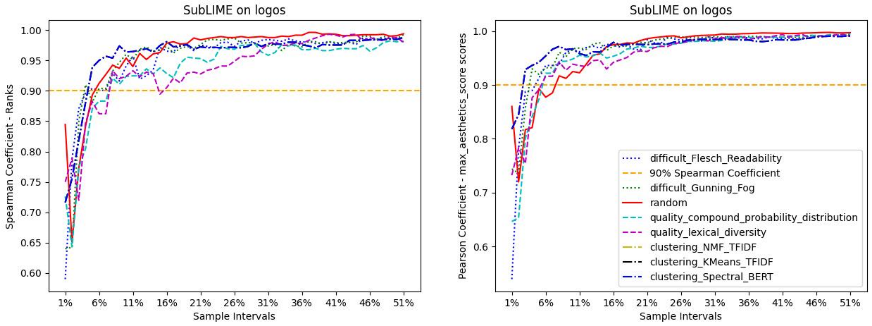


Figure 20: Rank and Score Preservation vs sampling ratio on Logos Benchmark on Aesthetics scores

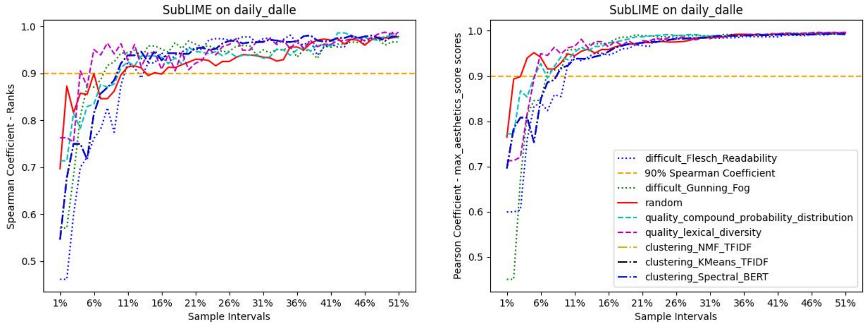


Figure 21: Rank and Score Preservation vs sampling ratio on Daily dalle Benchmark on Aesthetics scores

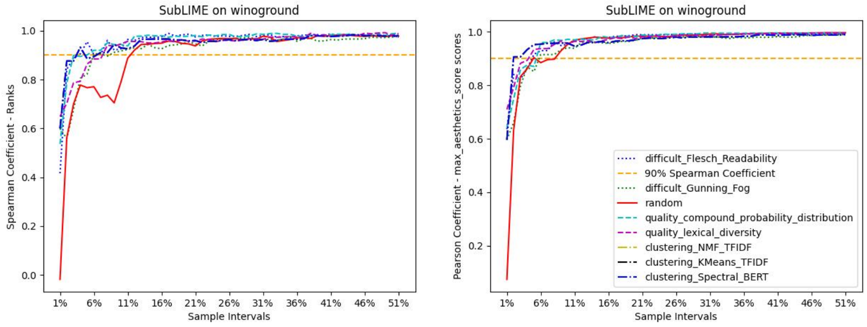


Figure 22: Rank and Score Preservation vs sampling ratio on Winoground Benchmark on Clip-scores

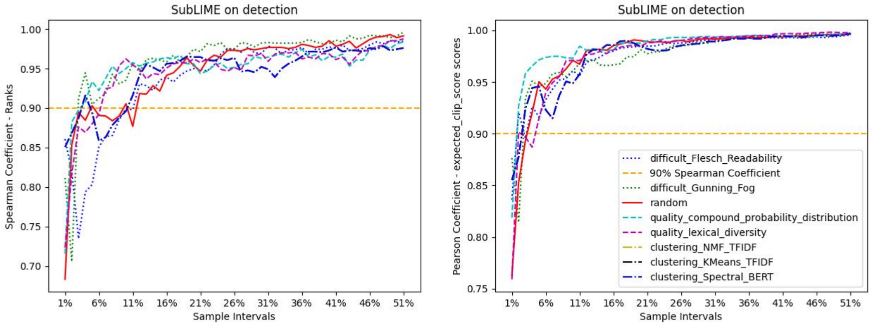


Figure 23: Rank and Score Preservation vs sampling ratio on Detection Benchmark on Clip-scores

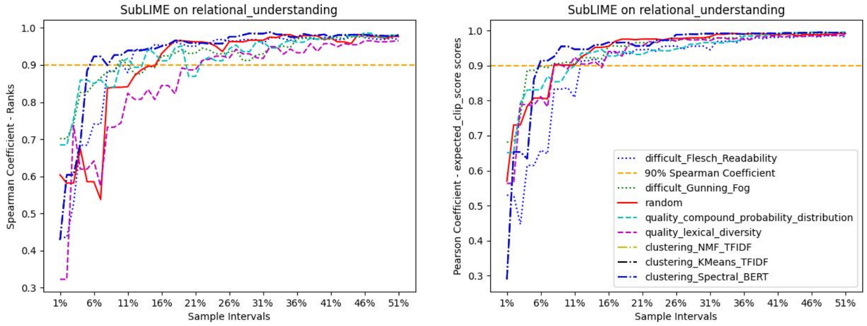


Figure 24: Rank and Score Preservation vs sampling ratio on Relational Understanding Benchmark on Clip-scores

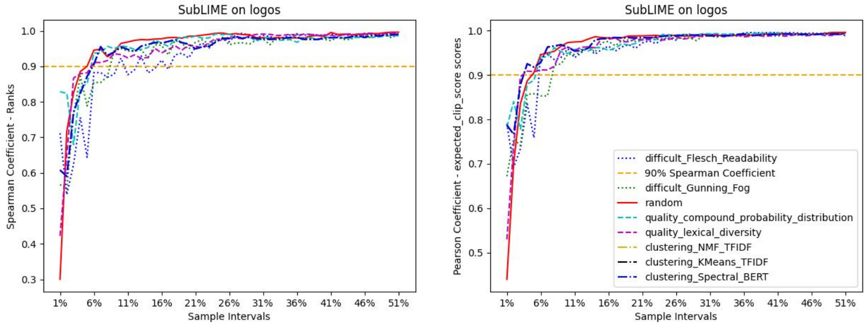


Figure 25: Rank and Score Preservation vs sampling ratio on Logos Benchmark on Clip-scores

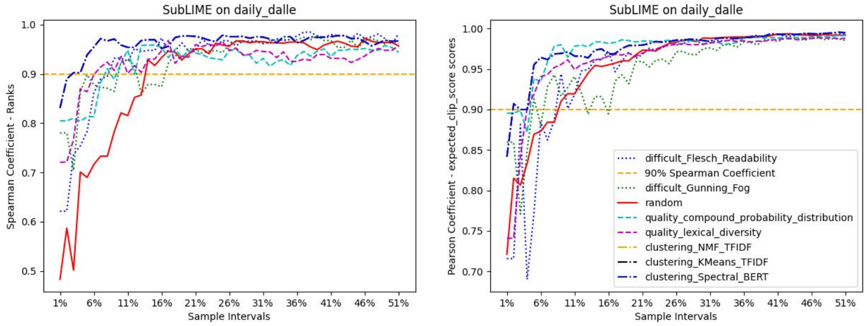


Figure 26: Rank and Score Preservation vs sampling ratio on Daile Dalle Benchmark on Clip-scores

4. meta-llama/Llama-3.1-8B
5. BSC-LT/salamandra-7b
6. cloudyu/Mixtral_13B_Chat
7. mayflowergmbh/occiglot-7b-es-en-instruct-AWQ
8. ibm-granite/granite-3.0-8b-base
9. MaziyarPanahi/Calme-7B-Instruct-v0.4
10. CohereForAI/aya-expanse-8b
11. google/gemma-2-2b-it
12. HiTZ/latxa-70b-v1.2
13. clibrain/lince-mistral-7b-it-es
14. HiTZ/latxa-13b-v1.2
15. GenVRadmin/AryaBhatta-GemmaUltra-Merged
16. gplsi/Aitana-6.3B
17. deepseek-ai/DeepSeek-R1-Distill-Qwen-7B
18. vanhocpham/Llama-3.2-3B-Instruct-GGUF
19. proxectonos/Carballo-bloom-1.3B
20. LLaMAX/LLaMAX2-7B
21. HPLT/sft-fpft-multilingual-downsampled-pythia–6.9b-deduped

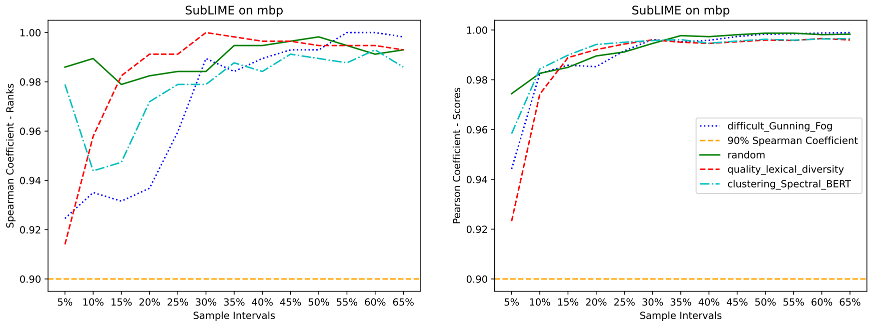


Figure 27: SubLIME on MBPP Benchmark : Before Redundancy Removal

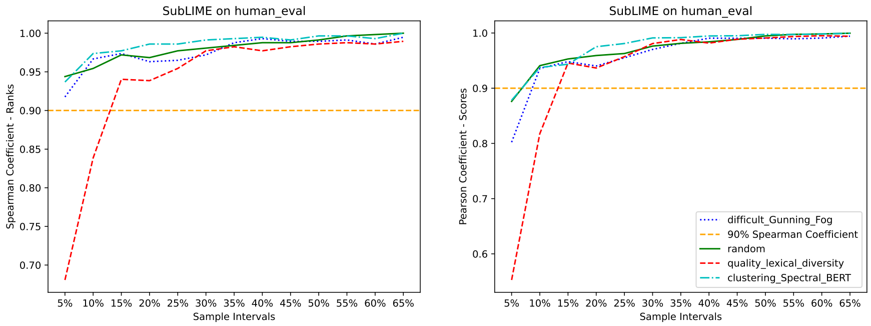


Figure 28: SubLIME on Human Eval Benchmark : Before Redundancy Removal

C.5 Discussion: Broader Applications of Adaptive Sampling

Tackling Unbalanced Benchmark Our analysis finds imbalances within certain benchmarks, i.e. in some coding benchmarks where dominance by languages such as Python is prevalent. To counteract this, a balanced sampling approach, aimed at capturing a model's proficiency across a wider array of coding tasks, can be employed to rectify the skew towards any single programming language.

Enhancing Benchmark Fairness by Mitigating Bias Our adaptive sampling approach also could help address biases inherent in benchmarks, which can distort the evaluation outcomes. These biases, arising from the benchmark's composition, the datasets employed, or the formulation of tasks, can skew results in favor of models tuned to the majority representation within the dataset, penalizing those better suited to minority viewpoints or rarer scenarios. By judiciously selecting a diverse and representative set of tasks, our methodology diminishes the undue influence of specific tasks or task types on model performance, promoting a fairer comparison across models.

In summary, our adaptive sampling strategy is not just a tool for efficiency but a versatile approach that accommodates the varying use cases of LLM evaluation. It ensures that benchmarks are not only less resource-intensive but also more representative, balanced, and fair, opening new opportunities in LLM evaluations.

Table 2: Adaptive sampling to each subject in MMLU with >90% Pearson Coefficient

MMLU Subject	Selected Sampling Method
high_school_government_politics	random
abstract_algebra	clustering_Spectral_MTEB
anatomy	clustering_Spectral_MTEB
astronomy	random
business_ethics	quality_CPD
clinical_knowledge	clustering_Spectral_MTEB
college_biology	quality_spelling_error
college_chemistry	quality_CPD
college_computer_science	quality_CPD
college_mathematics	clustering_Spectral_MTEB
college_medicine	clustering_Spectral_BERT
college_physics	clustering_Spectral_BERT
computer_security	clustering_NMF_TFIDF
conceptual_physics	clustering_Spectral_BERT
econometrics	clustering_NMF_TFIDF
electrical_engineering	quality_spelling_error
elementary_mathematics	quality_lexical_diversity
formal_logic	clustering_Spectral_BERT
global_facts	quality_CPD
high_school_biology	clustering_Spectral_MTEB
high_school_chemistry	quality_CPD
high_school_computer_science	quality_spelling_error
high_school_european_history	clustering_Spectral_BERT
high_school_geography	clustering_NMF_TFIDF
high_school_macroeconomics	clustering_NMF_TFIDF
high_school_mathematics	clustering_NMF_TFIDF
high_school_microeconomics	quality_spelling_error
high_school_physics	quality_spelling_error
high_school_psychology	random
high_school_statistics	clustering_NMF_TFIDF
high_school_us_history	quality_spelling_error
high_school_world_history	clustering_KMeans_TFIDF
human_aging	random
human_sexuality	clustering_Spectral_BERT
international_law	quality_spelling_error
jurisprudence	clustering_NMF_TFIDF
logical_fallacies	random
machine_learning	quality_spelling_error
management	clustering_Spectral_BERT
marketing	clustering_KMeans_TFIDF
medical_genetics	quality_lexical_diversity
miscellaneous	clustering_NMF_TFIDF
moral_disputes	random
moral_scenarios	clustering_NMF_TFIDF
nutrition	clustering_Spectral_BERT
philosophy	quality_spelling_error
prehistory	quality_lexical_diversity
professional_accounting	random
professional_law	clustering_NMF_TFIDF
professional_medicine	clustering_Spectral_MTEB
professional_psychology	quality_CPD
public_relations	clustering_KMeans_TFIDF
security_studies	clustering_KMeans_TFIDF
sociology	quality_spelling_error
us_foreign_policy	clustering_NMF_TFIDF
virology	clustering_Spectral_MTEB
world_religions	quality_CPD