

Select, Label, Evaluate: Active Testing in NLP

Antonio Purificato^{1,2} (✉), Maria Sofia Bucarelli^{3,4,5,#}, Andrea Bacciu¹, Amin Mantrach¹, and Fabrizio Silvestri²

¹ Amazon {purificato, bacciu, mantrach}@amazon.com

² Sapienza University of Rome {purificato, fsilvestri}@diag.uniroma1.it

³ Inria maria-sofia.bucarelli@inria.fr

⁴ CNRS

⁵ i3s

Work done while at Sapienza.

Abstract. Human annotation cost and time remain significant bottlenecks in Natural Language Processing (NLP), with test data annotation being particularly expensive due to the stringent requirement for low-error and high-quality labels necessary for reliable model evaluation. Traditional approaches require annotating entire test sets, leading to substantial resource requirements. Active Testing is a framework that selects the most informative test samples for annotation. Given a labeling budget, it aims to choose the subset that best estimates model performance while minimizing cost and human effort. In this work, we formalize Active Testing in NLP and we conduct an extensive benchmarking of existing approaches across 18 datasets and 4 embedding strategies spanning 4 different NLP tasks. The experiments show annotation reductions of up to 95%, with performance estimation accuracy difference from the full test set within 1%. Our analysis reveals variations in method effectiveness across different data characteristics and task types, with no single approach emerging as universally superior. Lastly, to address the limitation of requiring a predefined annotation budget in existing sample selection strategies, we introduce an adaptive stopping criterion that automatically determines the optimal number of samples.

Keywords: Active Testing · Data Annotation · NLP.

1 Introduction

Natural Language Processing (NLP) has witnessed unprecedented advances in recent years, with state-of-the-art models achieving remarkable performance across numerous tasks including text classification [11] and question answering [14]. However, the development and evaluation of these models continue to face a significant bottleneck: the acquisition of high-quality labeled data, particularly for test sets [8]. While training data can tolerate some level of noise, test sets require meticulous annotation with minimal error rates to serve as reliable benchmarks for model evaluation [4]. The conventional approach to model testing in NLP involves exhaustive annotation of entire test sets, representing a

substantial investment of time and financial resources. This burden is acute in domains requiring specialized knowledge like the annotation of medical texts [17] or technical content often requires trained annotators, driving up costs and limiting scalability. Full test-set evaluation is often impractical: annotation costs vary notably across examples (*e.g.*, in multilingual settings), and many samples may be redundant, with minimal impact on performance estimation.

In recent years, the landscape of data annotation has been transformed by the emergence of Large Language Models (LLMs) and AI agents as potential annotation tools [6]. These advanced systems are increasingly being deployed to generate, (synthetically) annotate, and validate linguistic data at unprecedented scale. While these approaches offer promising avenues for scaling annotation efforts, they introduce new challenges related to annotation quality, bias propagation, and evaluation reliability. Most critically, LLM-based annotations often still require human verification, especially for test sets where annotation quality directly impacts the validity of model evaluation [19].

Active Testing emerges as a promising paradigm to address these challenges by leveraging only the input samples before any annotation takes place, enabling the identification of a minimal subset of test samples providing statistical guarantees on model performance estimates while minimizing the difference with respect to full test set evaluation. This approach operates on the principle that not all test examples are equally informative for evaluating model performance [7]. By strategically selecting these high-information samples, Active Testing enables efficient performance estimation with dramatically reduced annotation requirements, thereby substantially reducing the associated cost.

This can be noted by looking at Fig. 1, which shows the amount of money saved by annotating using Active Testing with respect to annotating the full test set. While related concepts have been explored in areas such as Active Learning [1], the systematic application of these principles to test data in NLP remains largely unexplored. Active Testing differs from Active Learning in its objectives: while Active Learning aims to improve model training with fewer labeled examples, Active Testing focuses on estimating test performance with minimal annotation, requiring different sampling strategies and metrics tailored to the testing context [3]. Specifically, this work aims to answer to the following research questions:

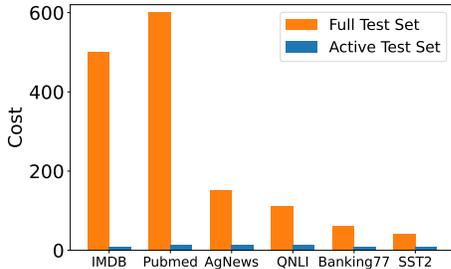


Fig. 1: Annotation cost of using Active Testing vs full test set (\downarrow is better). Each sample has a cost of 0.02\$ as from LabelYourData. The active cost is the one required to reach an estimation quality $< 1\%$ with respect to the full budget.

- **RQ1.** *Is adaptive stopping an effective criterion for reducing annotation budgets in Active Testing while preserving reliable performance estimates?*
We present the first framework for Active Testing in NLP, offering objectives and evaluation criteria and establishing a basis for research. Our results demonstrate annotation reductions of up to **95%** while preserving performance estimate accuracy within **1%** of testing on the full test set. We introduce an adaptive stopping criterion addressing a limitation of existing Active Testing methods: the need to pre-specify the annotation budget.
- **RQ2.** *Can cost-based priors reduce expensive annotations via low-cost language transfer?*
We show how Active Testing operates in multilingual environments, optimizing cost by strategically allocating budget across different languages.
- **RQ3.** *How can Active Testing support the detection of minority class samples to be annotated in imbalanced sets?*
We show how Active Testing can be applied for detection of samples from the minority class, helping annotators in identifying relevant samples. This is relevant in domains such as finance or e-commerce, where rare but high-impact events must be accurately identified under limited annotation budgets.

2 Related Work

Recent work in Active Testing can be broadly categorized into two main areas: approaches that focus on optimal sampling under budget constraints, and methods specifically designed for Active Testing. While the latter category is relevant for comparison with our work, budget-constrained sampling approaches, although sharing the goal of reducing annotation costs, are not directly comparable as they do not target test set optimization.

2.1 Sampling with a limited budget

Zouhar et al. [20] investigate strategies for efficient subset selection in human evaluation of Natural Language Generation models, aiming to reduce annotation costs while preserving accurate rankings. They propose two frameworks: output-based methods, relying on model outputs and automatic metrics, and source-based methods, predicting item utility directly from inputs when outputs are unavailable. While *Zouhar et al.* [20] focus on human evaluation scenarios, *Zhen et al.* [19] extend this line of research to automated evaluation, presenting an active sampling framework for automatic prompt optimization targeting LLM-as-a-judge systems. Their approach begins with zero labeled examples, iteratively selecting a sample set whose annotations are used to refine the prompt. Sample selection is posed as a convex optimization problem, balancing uncertainty and diversity to maximize utility under limited labeling budgets. Complementing these task-specific sampling strategies, *Rauch et al.* [16] benchmark the influence of LLM embedding quality on deep Active Learning strategies. They evaluate five top-ranked embedding models from the Massive Text Embedding

Benchmark (MTEB) leaderboard across diverse NLP text-classification datasets using multiple query strategies.

While all three works share the goal of reducing annotation costs through intelligent sampling, they differ from our approach in a fundamental way: they optimize the training or evaluation pipeline itself (*e.g.*, selecting examples for human evaluation or refining prompts), whereas our work focuses on constructing a minimal yet reliable test set for model assessment.

2.2 Active Testing

The concept of Active Testing was first introduced by *Nguyen et al.* [13] in the context of visual recognition systems trained on noisy datasets. Instead of annotating an entire test set, their framework selectively queries only a subset of examples to be re-annotated by humans. These annotated examples are then used to train a statistical estimator that improves the accuracy of performance metrics and actively guides which further examples should be annotated. Building on this foundation, *Kossen et al.* [7] generalize the Active Testing framework with a focus on reducing the number of test labels required for reliable model assessment. They propose actively selecting test points to label using acquisition strategies tailored to the testing phase. While such active selection introduces sampling bias, based on the work of *Farquhar et al.* [3], they derive estimators that are unbiased and exhibit reduced variance, leading to more accurate evaluation with fewer labels. They demonstrate that, across different image classification models and datasets, Active Testing can match the accuracy of *i.i.d.* evaluation with significantly fewer labeled examples.

Although both works establish important theoretical and empirical foundations for Active Testing, they remain confined to the computer vision domain and rely on specific heuristics or limited experimental settings. In contrast, our work provides the first framework for Active Testing in NLP, accompanied by empirical benchmarking across diverse tasks and an adaptive method that generalizes beyond prior approaches.

3 Method

Given a set of samples X with $|X| = N$, we define X_A as the set of annotated samples and X_{NA} as the set of non-annotated samples ($X = X_A \cup X_{NA}$). We use the same notation for the labels $Y = Y_A \cup Y_{NA}$. We have a model $f : X \rightarrow Y$. An Active Testing algorithm has a budget B and each query could have a different cost. Given a metric $M = M(f(X), Y)$, we define the estimation error as:

$$E(M) = | M(X_F) - M(X_A) |$$

The estimation error is a dimensionless quantity which measures the difference between the metric computed on the fully annotated test set (X_F) and on the actively selected subset (X_A). Throughout the paper, we use an unbiased

estimator for the latter, presented in the next paragraph. We focus on performance metrics commonly used by practitioners like accuracy, precision, recall (for classification, NER and POS tagging) [11] and ROUGE (for summarization) [9] as these metrics, differently from loss, provide a reliable way to evaluate model performance. Given the cost $C(x)$ of annotating sample x , we aim to find:

$$\begin{aligned} \min E(M) \text{ s.t.} \\ \sum_{x \in X_A} C(x) \leq B \end{aligned}$$

We seek to select the subset X_A that minimizes the estimation error while ensuring that the total annotation cost does not exceed the available budget B .

3.1 Unbiased Active Testing

The simplest and most used approach to handle limited annotation budgets is uniform random sampling. Given a labeling budget B , it selects a subset of examples uniformly at random and, given a metric M , it is defined as:

$$\widehat{M}_{\text{random}} = \frac{1}{B} \sum_{i=1}^B M(f(x_i), y_i)$$

In uniform sampling, the random estimator is unbiased ($\mathbb{E}[\widehat{M}_{\text{random}}] = M$) and it converges to the true empirical metric as $B \rightarrow N$, but it can produce estimates with large variance.

As pointed out in [3,7], Active Testing introduces a bias into our estimates, since the samples are not drawn from the population distribution. To solve this issue we use the Inverse Probability Weighted Estimator [5] that provides an unbiased estimator which is defined as:

$$\widehat{M} = \frac{1}{B} \sum_{i=1}^B \frac{M(f(x_i), y_i)}{Nq_i}$$

Where $q_i = q(x_i; X_{1:i-1}, X)$ is the probability mass for datum x_i of being the next to be sampled and it depends on the Active Testing strategy. Additional details and derivations for our estimator are provided in Section A.2. During the rest of the paper we focus on accuracy rather than loss as the primary evaluation metric but the same estimators for precision, recall can be found in Section A.1. Accuracy provides an interpretable, bounded measure of performance that directly reflects classification quality and it is the metrics used in related works [7]. For summarization we focus on ROUGE. Estimators like PURE and LURE [3] cannot be used in this case, since they tend to overestimate the value of the accuracy and they do not satisfy some properties of the accuracy like being in $[0, 1]$ (see Section A.3). This limitation is problematic when evaluating accuracy, where boundedness is an essential property for interpretation.

3.2 Active Testing Strategies

We evaluate several Active Testing strategies for textual data in this work. Although some are adapted from Active Learning techniques, the two address fundamentally different problems, as discussed in [7].

- Surrogate [RF, SVM] [7]: It selects test samples based on an auxiliary model that estimates per-instance uncertainty. Embeddings for all unlabeled texts are extracted using the same pretrained embedder employed in the main pipeline, and a surrogate classifier is trained on these embeddings paired with the ground-truth labels accumulated so far. Two base learners are supported: a Random Forest with 300 trees and an RBF-kernel SVM. For each candidate sample, the surrogate outputs calibrated class probabilities, and the uncertainty score is defined as the Shannon entropy [2]. Samples with the largest normalized scores are selected. A notable drawback of this strategy is that it requires training the surrogate classifier on ground-truth labels, which must be progressively collected during the annotation process, introducing an additional labeling overhead.
- Uncertainty [GP, MI] [7]: It selects test cases by estimating predictive uncertainty through Monte Carlo Dropout applied to the pretrained embedder, not the LLM predictor being evaluated. All samples are passed through the transformer model with dropout kept active, performing ten forward passes per batch. Two acquisition functions are supported. For mutual information (MI), each pass produces logits converted into probabilities via softmax. The uncertainty score is then computed as the difference between the entropy of the mean predictive distribution across all forward passes and the average entropy of the individual per-pass distributions. For Gaussian prior (GP), mean-pooled last-layer embeddings are extracted at each pass. The acquisition score is the squared sum of standard deviations across embedding dimensions, capturing latent-space variability induced by dropout.
- Coverage [12]: It maximizes the geometric spread of sampled points in the embedding space. All texts are first encoded into dense representations using the pretrained multilingual embedder. The algorithm starts by selecting one point uniformly at random. At each subsequent step, given the set of selected indices S , it computes the Euclidean distance $d(j) = \|x_j - x_i\|_2$ between each unselected point $j \notin S$ and the last selected point x_i , then appends the point achieving the maximal distance. This continues until the desired budget is reached, ensuring broad geometric exploration of the embedding manifold without relying on clustering assumptions.
- Stratified Random: It selects test samples proportionally from each class to maintain the original distribution. However, it is not Active Testing: it requires prior knowledge of the class distribution across the entire unlabeled dataset, which is the information that Active Testing aims to avoid needing.
- Agreement (ours): The Agreement strategy ranks unlabeled samples by the level of disagreement among attention heads. Text inputs are first encoded using a pretrained language model, producing contextual token representations. These are then passed through a separate, lightweight multi-head

self-attention layer with a fixed number of 8 heads, independent of the LLM predictor being evaluated, which outputs both transformed token embeddings and per-head attention weight matrices. For each sample, the uncertainty score is computed by taking the variance of the attention weights. More precisely, for each token position t , the vector containing all token-to-token attention scores for head h is considered, and its variance across heads is computed. The resulting variance matrices are averaged over all token positions, producing a single scalar score for each sample. All unlabeled samples not previously selected by the algorithm are ranked in descending order according to their normalized attention-based scores.

3.3 Stopping Criterion

To the best of our knowledge, existing works on Active Testing [7] typically set a fixed budget B of samples to be annotated. Therefore, the annotation process exhausts the whole budget, regardless of the actual need for it. However, in certain scenarios, this might be unnecessary. Since the actual number of samples to be annotated cannot be determined a priori, we propose Algorithm 1, which automatically terminates the annotation process as soon as we estimate that a sufficient number of samples have been annotated. Since the algorithm terminates adaptively before the full test set is annotated, $M(X_F)$ is unavailable. Therefore, in this setting, the estimation error is computed as the difference between the raw metric on X_A and its unbiased estimate on the same subset.

More concretely, the stopping criterion in Algorithm 1 relies on the convergence between the unbiased estimator \widehat{M} and the baseline random estimator $\widehat{M}_{\text{random}}$. Let $\widehat{M}^{(t)}$ denote the unbiased estimate after t annotated samples. As Active Testing progresses and the sampling strategy sufficiently explores the data distribution, the bias term $\mathbb{E}[\widehat{M}^{(t)} - \widehat{M}_{\text{random}}^{(t)}]$ converges to zero. When the empirical difference $|\widehat{M}^{(t)} - \widehat{M}_{\text{random}}^{(t)}| < \tau$ for a small threshold τ , we deem the estimator stable—further annotations are unlikely to change the performance estimate meaningfully. More formally, the following proposition holds:

Proposition 1. *Let $\widehat{M}_{\text{random}}$ be the unweighted estimator of the metric and \widehat{M} the inverse probability (Horvitz-Thompson) estimator defined in Section 3.1. If the sample is drawn without replacement and labeling budget $B \rightarrow N$ (approaching full annotation), then $\widehat{M}_{\text{random}} - \widehat{M} \xrightarrow{P} 0$.*

Proof can be found in Section A.4, where we also derive an explicit finite-sample upper bound on the gap as a function of the annotation budget.

4 Experiments

4.1 General Framework

For our experiments, we use the same experimental setup as [16]. Our framework follows the procedure presented in Pipeline 1. All strategies share a common

Pipeline 1 Active Testing Framework

Require: Dataset \mathcal{D} , embedding strategy E , budget B , test set X ;
 Compute embeddings on X ;
 Compute model predictions $f(X)$; ▷ Stored for metric eval.
 Given the embeddings, construct X_A via the chosen Active Testing strategy;
 Evaluate predictions using X_F and the active X_A test sets and compute the metrics;

Algorithm 1 Stopping Criterion

Require: Dataset \mathcal{D} , budget B , threshold τ
for i in range(B) **do**
 Pick a sample x_i using an Active Testing strategy and annotate it \rightarrow label y_i ;
 $X_A = X_A \cup x_i$ and $Y_A = Y_A \cup y_i$;
 Compute the prediction $f(X_A)$;
 Compute $\widehat{M}_{\text{random}}(f(X_A), Y_A)$;
 Compute unbiased metric $\widehat{M}(f(X_A), Y_A)$;
if $|\widehat{M}_{\text{random}} - \widehat{M}| < \tau$ **then**
break
else
continue
end if
end for

structure: given the embedding space of X , they iteratively select samples to form the active set X_A until the budget B is exhausted or the stopping condition is met. At each iteration, an Active Testing strategy assigns a score to each unlabeled sample, and the highest-scoring samples are added to X_A . We test 8 Active Testing strategies on 18 datasets using 4 embedding strategies and 3 predictors. This results in a large number of experiments. Due to space constraints, we show the results on the selected Active Testing strategies with one embedding strategy (Qwen) and one predictor (Claude 4.5). All the results are averaged over 10 different seeds. Remaining plots can be found in our repository.

Hardware We run all the experiments on a machine with 96 AMD EPYC 7R13 Processor CPUs and 4 NVIDIA L40S GPUs with 48 GB of RAM.

Datasets We use 18 datasets, covering text classification (12), POS tagging (2), NER (2) and summarization (2), presented in Table 1. We take all the data from the ActiveGLAE benchmark [15] with the addition of more data (marked with *). We try to set, when possible, the same value of B for all the datasets ($B = 1000$). Statistics on the number of training and test examples, number of classes and class distribution⁶ are also presented in Table 1.

⁶ For Banking77, values are omitted due to the high number of classes (77), with the dataset being nearly balanced ($\sim 1.29\%$ of samples per class). NA indicates summarization datasets with no classes.

Table 1: Class distribution of the selected datasets and time required to compute the embeddings on the full datasets in minutes (T). Number of train and test samples and values of the budget are also shown.

Name	#Classes	Class Distribution (%)	T [minutes]	Train	Test	B
AG's News	4	[25,25,25,25]	7	120,000	7,600	1,000
Banking77	77	-	2	10,000	3,000	1,000
DBPedia	14	[7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1,7.1]	77	560,000	5,000	1,000
FNC-1	4	[6.72,1.50,17.77,74.01]	2	40,000	4,998	1,000
QNLI	2	[49,51]	3	104,000	5,463	1,000
SST-2	2	[49,51]	1	67,000	872	600
TREC-6	6	[1.8,18.80, 27.60, 13.00, 16.2, 22.60]	1	5,452	500	400
IMDB*	2	[50,50]	41	25,000	25,000	1,000
PubMed*	5	[10.4,15.45, 33.42,7.89,32.84]	21	170,000	30,000	1,000
Emotion*	6	[29.05,34.75,7.95, 13.75,11.20,3.30]	2	16,000	2,000	1,000
Rotten*	2	[50,50]	1	8,000	1,000	600
Multilingual*	3	[33,33,34]	1	1,840	880	600
WikiNeural*	9	[87.7,1.9,1.5,1.3,1.0,2.2,0.9,1.7,1.8]	6	92,720	11,579	1,000
Universal NER*	9	[93.3,1.8,0.9,1.3,1.1,1.2,0.3,0.05,0.05]	1	12,543	2,077	1,000
UD-ATIS*	17	[17,21,2,3,1,7,1,23,5,1.5,0.5,10,4]	1	4,274	586	500
UD-EWT*	17	[17,12,8,3,0,4,1.5,7.1,3,7,3,8,8,1,1,4,10,6]	1	12,544	2,077	1,000
CNN*	NA	NA	4	287,113	11,490	1,000
XLSum*	NA	NA	2	38,110	4,763	1,000

Embedding Strategies Since the quality of the embeddings influences the results of Active Learning [16], we investigate whether this observation extends to Active Testing. To this end, we evaluate 4 different models from the MTEB leaderboard (Bert, DistilBert, Qwen, Stella). As shown in Fig. 2, even on the same dataset and with the same strategy, performance varies across embedding models.

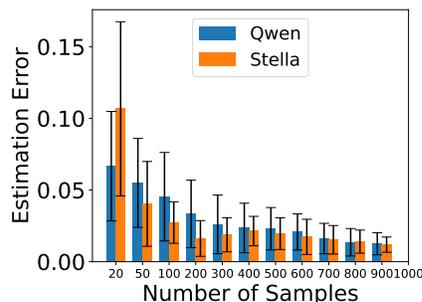


Fig. 2: Estimation error on Agreement on the AgNews dataset using Qwen and Stella embedding strategies (\downarrow is better).

Predictor To avoid bias from the training phase and ensure the full comparability between embedding models, we use Claude Sonnet 4.5, Amazon Nova Pro and Qwen for the text-classification task. The selected predictors are able to demonstrate good performance in text-classification and summarization, as also shown in Section B, where we also present the prompts used for classification and summarization. For POS tagging, we use BERT and DeBERTa finetuned, while for NER we use CNER and UNER.

4.2 Multilingual Setup

Annotation costs vary significantly across languages, with some (*e.g.*, English) being relatively cheap and others considerably more expensive. We investigate

whether Active Testing can reduce annotation effort in high-cost languages by leveraging samples from low-cost languages, transferring evaluation knowledge across linguistic boundaries.

For this experiment, we apply the steps presented in Pipeline 1. The main difference is that in this case the dataset \mathcal{D} contains samples from two different languages and has a prior p on the languages; in our case the prior reflects the relative cost of annotating texts in each language. The estimation error is computed with respect to the set containing the annotations for both the languages. This experiment assumes cross-lingual transferability, where a model performing well in one language achieves similar classification accuracy in the other. This is reasonable for multilingual models with shared vocabularies, as sentence-level classification tends to be language-agnostic [18].

4.3 Minority Class Sample Detection

Active Testing presents advantages for highly imbalanced datasets. This is relevant in domains such as finance, e-commerce, and A/B testing, where rare but high-impact events must be accurately identified. Examples include detecting fraudulent transactions for risk management or identifying negative user feedback to maintain product quality. In these scenarios, Active Testing enables an efficient allocation of limited annotation budgets toward critical yet underrepresented cases. To investigate this aspect, we conduct experiments under various budget constraints B . We define “minority samples” as those belonging to the class with the fewest instances in the test set, and evaluate how effectively each strategy identifies and selects such samples. Specifically, we measure the number of minority-class samples included in the active set, providing insights into model behavior on these cases.

5 Results

RQ1: Active Testing

Figure 3 shows the accuracy estimation error of the top 4 strategies. We report results on six datasets spanning two classification, two summarization, one NER, and one POS tagging task. For the summarization datasets we report ROUGE-1 estimation error instead of accuracy. Overall, all strategies perform comparably, with relatively small differences across methods. Among them, the proposed Agreement strategy tends to achieve slightly lower estimation error, particularly at reduced values of budget B , where it also manages to outperform the Random baseline. Furthermore, across most datasets, the estimation error remains low even under tight budget constraints, highlighting the practical effectiveness of Active Testing. POS tagging results follow trends similar to text classification and summarization, with Agreement achieving competitive performance. For NER, the curves display more irregular patterns—likely due to entity sparsity and class imbalance—yet Agreement remains consistently effective, confirming

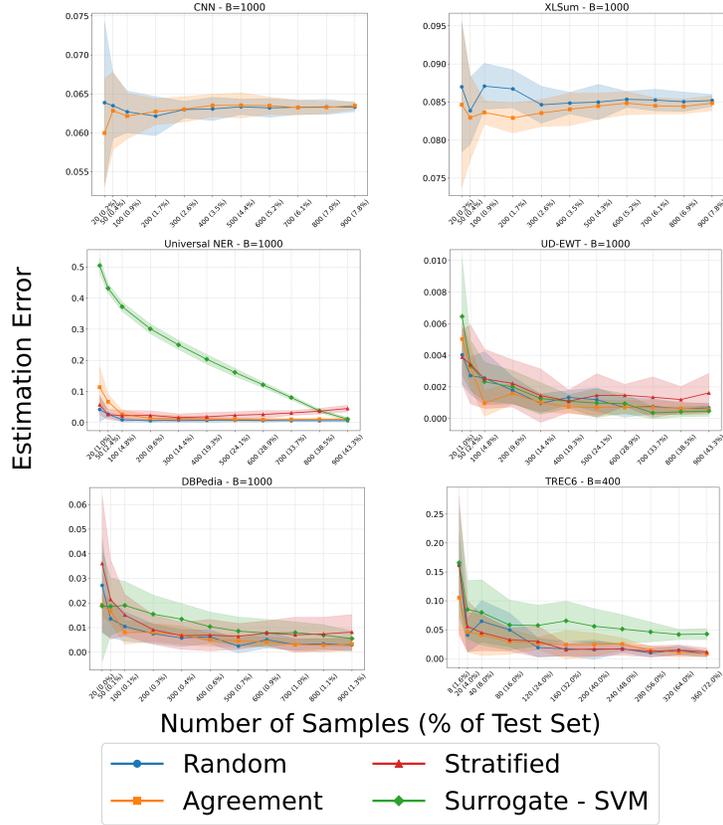


Fig. 3: Active Testing results in terms of accuracy on all the datasets by fixing the embedding strategy and the predictor. We observe that the Agreement strategy is performing better than Surrogate on the majority of the datasets. With a reduced budget Agreement is also able to beat Random (\downarrow is better).

its robustness across different task types. For clarity, we omit the remaining strategies from the main figures. Full results (also in terms of Precision and Recall) are available in the repository.

Stopping Mechanism Figure 4 shows the results of the proposed stopping criterion approach on the QNLI and IMDB datasets with $B = 1000$ and $\tau = 0.01$. On IMDB, the Agreement approach stops the annotation process after 50 samples, meaning that it reaches values of estimation error lower than τ . A similar behavior is observed for Stratified Random, which requires 250 samples, while Surrogate needs 200 samples. Comparable results are obtained on QNLI, where Agreement and Surrogate stop after 100 and 400 samples, respectively. This

Table 2: Computation time (in seconds) and annotation cost (in USD, assuming \$0.02 per sample as from LabelYourData) for different Active Testing strategies (\downarrow is better). *S*: stopping criterion applied on the specified strategy; *Full*: entire dataset annotated; *NA*: stopping condition not met.

B	IMDB				QNLI			
	$T_{\text{Agreement}}$	T_{Random}	$T_{\text{Surrogate}}$	Cost	$T_{\text{Agreement}}$	T_{Random}	$T_{\text{Surrogate}}$	Cost
100	5.5114	0.0001	12.4741	2	4.9738	0.0001	14.3149	2
500	5.5131	0.0003	12.5245	10	5.0541	0.0003	14.3259	10
900	5.5185	0.0005	59.0544	18	5.0801	0.0005	43.0613	18
S-Agreement	5.4945	-	-	1	5.0541	-	-	2
S-Random	-	0.0001	-	2	-	NA	-	NA
S-Surrogate	-	-	12.7448	4	-	-	14.7184	8
Full	137.9625	0.0125	1,476.36	110	27.43254	0.0027	232.5294	500

means that, with respect to the full budget $B = 1000$, our stopping criterion based on Agreement avoids annotating 900 samples on QNLI and 950 on IMDB.

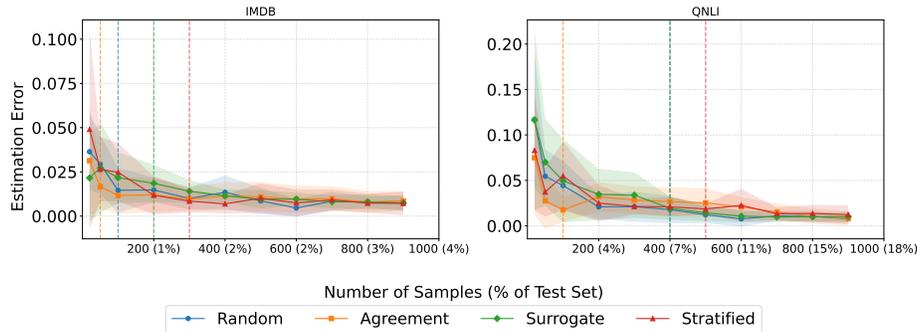


Fig. 4: Results using the stopping criterion approach in order to avoid annotating all the samples (\downarrow is better). Vertical lines represents when the algorithms stop.

Table 2 reports computational time and annotation cost for each strategy on IMDB and QNLI. Random has negligible overhead, Agreement requires roughly 5 seconds regardless of budget, while Surrogate is the most expensive, with time increasing significantly at higher budgets. Applying Active Testing reduces annotation costs compared to evaluating the full test set: for instance, on QNLI the cost drops from \$500 to \$10 at $B = 500$, a $50\times$ saving. Notably, Agreement combined with the stopping criterion consistently yields the lowest annotation cost across both datasets, confirming that the two contributions complement each other effectively. The cost of computing embeddings is negligible, requiring less than one hour on a machine with the setup previously described. For clarity

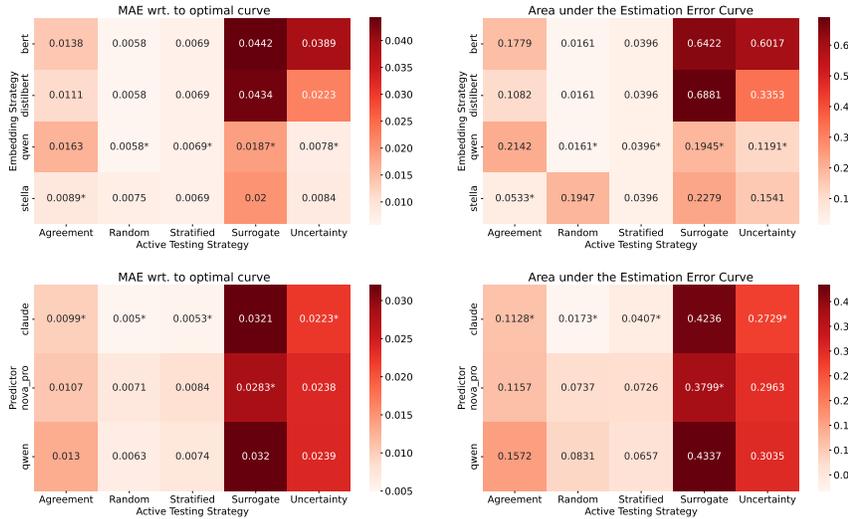


Fig. 5: Confusion matrices with the Active Testing Strategies on the x-axis. The top figures have the embedding strategies on the y-axis while the bottom ones the predictors. The best results are marked with *. MAE wrt. optimal curve on the left and Area Under the Estimation Error Curve on the right. ↓ is better.

and due to space constraints, we report results on two datasets; results on the remaining datasets are consistent and can be found in our repository.

Impact of Embedding Strategy and Predictor We investigate how the choice of embedding strategies and predictor affect the performance using two metrics: Area under the Estimation Error Curve (normalized by budget; lower=faster error reduction), and Mean Absolute Error (MAE) from the optimal curve (lower=closer to best achievable performance), averaged across datasets.

In Fig. 5, Qwen achieves the best performance across embedding strategies, while Claude performs well across predictors. Both analyses reveal the same pattern: Random and Stratified Random show comparable performance regardless of embedder or predictor, while more sophisticated Active Testing strategies (Agreement, Surrogate) exhibit larger performance gaps. This suggests that both embedding and predictor quality become critical as the Active Testing strategy grows more complex, since these methods rely on the geometric structure of the embedding space and model-specific signals to identify informative samples.

RQ2: Active Testing with Cost-Based Priors

Fig. 6 shows the results of the multilingual experiments on the Multilingual dataset. For the mixed strategy, which leverages the cross-lingual capabilities of our approach, two values of prior are considered: $p = 0.9$, which favors selecting English samples (cheaper), and $p = 0.6$, which provides a more balanced

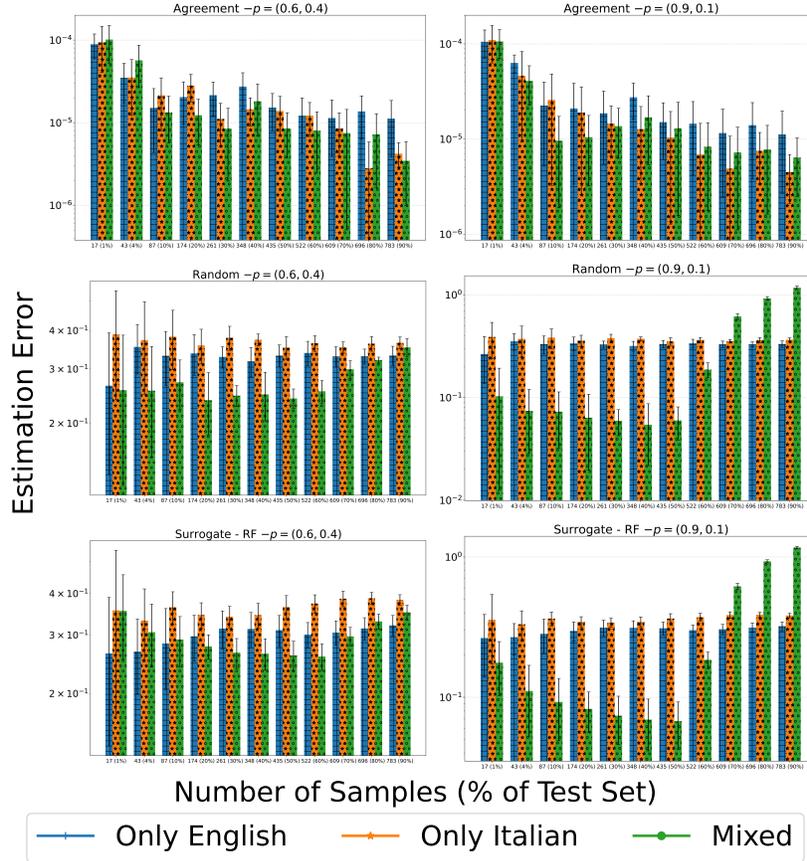


Fig. 6: Multilingual results. Each row is a value of prior p . Each column refers to an Active Testing method (\downarrow is better).

selection across languages. Allowing language selection based on a prior benefits Surrogate and Agreement, achieving estimation errors equal to or lower than single-language sampling. For Random, high priors (*e.g.*, $p = 0.9$) introduce bias toward one language, negatively affecting accuracy. Nevertheless, setting $p = 0.6$ reduces annotation costs by favoring cheaper English samples, while maintaining performance within 1% of the balanced setup, demonstrating that cost-aware Active Testing enables multilingual evaluation with minimal accuracy loss.

Notably, unlike the results observed in previous experiments, in this setting the Active Testing strategies consistently outperform random sampling. Allowing language selection based on a prior benefits Surrogate and Agreement, achieving estimation errors equal to or lower than single-language sampling.

Table 3: Per-class F1 on minority class detection (\uparrow is better). Agreement and Surrogate outperform Random at identifying minority-class samples.

B	Agreement	Random	Surrogate-SVM	B	Agreement	Random	Surrogate-SVM
75	0.031 \pm 0.006	0.022 \pm 0.012	0.022 \pm 0.005	100	0.096 \pm 0.001	0.041 \pm 0.008	0.072 \pm 0.003
150	0.042 \pm 0.008	0.036 \pm 0.009	0.029 \pm 0.006	500	0.151 \pm 0.003	0.074 \pm 0.007	0.103 \pm 0.004
450	0.044 \pm 0.006	0.033 \pm 0.007	0.035 \pm 0.008	1500	0.145 \pm 0.006	0.094 \pm 0.007	0.133 \pm 0.007
750	0.047 \pm 0.003	0.043 \pm 0.003	0.044 \pm 0.006	3000	0.125 \pm 0.005	0.102 \pm 0.006	0.113 \pm 0.005

Emotion dataset.

Pubmed dataset.

RQ3: Minority Class Sample Detection

Table 3 presents the results on minority class detection on two datasets and three Active Testing strategies. We focus on Emotion and PubMed as they exhibit the highest class imbalance (see Table 1). To evaluate how effectively each strategy identifies minority-class samples, we treat minority detection as a binary retrieval task: we compute Precision as the fraction of selected samples that belong to the minority class, and F1 accordingly. Across both datasets, Agreement consistently achieves the highest F1, followed by Surrogate, with Random performing worst. The advantage of Active Testing strategies is particularly pronounced at lower budgets: on PubMed at $B = 100$, Agreement achieves an F1 of 0.096, more than twice the Random baseline (0.041), demonstrating that informed sampling is especially beneficial when annotation resources are scarce. These results confirm that, as in the multilingual setting, Active Testing strategies prove advantageous when targeting specific subsets of interest, with Agreement being particularly effective at detecting underrepresented classes. For clarity and due to space constraints, we report results on two datasets; results on the remaining datasets are consistent and can be found in our repository.

6 Conclusions and Future Work

We presented a framework for Active Testing in NLP, demonstrating its effectiveness in reducing annotation costs while maintaining reliable model evaluation. The adaptive approach successfully addresses the challenge of determining optimal sample sizes, making Active Testing practical for real-world applications.

Several directions for future research emerge like the development of more sophisticated sample selection methods incorporating multi-criteria optimization or the integration of Active Testing with LLM-based evaluation approaches.

References

1. Ahmadnia, S., Yousefi Jordehi, A., Hosseini Khasheh Heyran, M., Mirroshandel, S.A., Rambow, O., Caragea, C.: Active few-shot learning for text classification.

- In: Chiruzzo, L., Ritter, A., Wang, L. (eds.) Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). pp. 6677–6694. Association for Computational Linguistics, Albuquerque, New Mexico (Apr 2025). <https://doi.org/10.18653/v1/2025.naacl-long.340>, <https://aclanthology.org/2025.naacl-long.340/>
2. Bromiley, P., Thacker, N., Bouhova-Thacker, E.: Shannon entropy, renyi entropy, and information. *Statistics and Inf. Series (2004-004)* **9**(2004), 2–8 (2004)
 3. Farquhar, S., Gal, Y., Rainforth, T.: On statistical bias in active learning: How and when to fix it. In: *International Conference on Learning Representations (2021)*
 4. Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S., Smith, N.A.: Annotation artifacts in natural language inference data. In: Walker, M., Ji, H., Stent, A. (eds.) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). pp. 107–112. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018). <https://doi.org/10.18653/v1/N18-2017>, <https://aclanthology.org/N18-2017/>
 5. Horvitz, D.G., Thompson, D.J.: A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association* **47**(260), 663–685 (1952)
 6. Karim, M.M., Khan, S., Van, D.H., Liu, X., Wang, C., Qu, Q.: Transforming data annotation with ai agents: A review of architectures, reasoning, applications, and impact. *Future Internet* **17**(8), 353 (2025)
 7. Kossen, J., Farquhar, S., Gal, Y., Rainforth, T.: Active testing: Sample-efficient model evaluation. In: *International Conference on Machine Learning*. pp. 5753–5763. PMLR (2021)
 8. Kreutzer, J., Caswell, I., Wang, L., Wahab, A., Van Esch, D., Ulzii-Orshikh, N., Tapo, A., Subramani, N., Sokolov, A., Sikasote, C., et al.: Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics* **10**, 50–72 (2022)
 9. Li, D., Zhang, Y., Wang, Z., Tan, S., Kosugi, S., Okumura, M.: Active learning for abstractive text summarization via llm-determined curriculum and certainty gain maximization. In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. pp. 8959–8971 (2024)
 10. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: *Text summarization branches out*. pp. 74–81 (2004)
 11. Liu, C., Zhang, W., Chen, G., Wu, X., Tuan, L.A., Chang, C.H., Bing, L.: Zero-shot text classification via self-supervised tuning. In: *Findings of the Association for Computational Linguistics: ACL 2023*. pp. 1743–1761 (2023)
 12. Maharana, A., Yadav, P., Bansal, M.: \mathbb{D}^2 pruning: Message passing for balancing diversity & difficulty in data pruning. In: *The Twelfth International Conference on Learning Representations (2024)*
 13. Nguyen, P., Ramanan, D., Fowlkes, C.: Active testing: An efficient and robust framework for estimating accuracy. In: *International Conference on Machine Learning*. pp. 3759–3768. PMLR (2018)
 14. Parekh, T., Prakash, P., Radovic, A., Shekher, A., Savenkov, D.: Dynamic strategy planning for efficient question answering with large language models. In: Chiruzzo, L., Ritter, A., Wang, L. (eds.) *Findings of the Association for Computational Linguistics: NAACL 2025*. pp. 6038–6059. Association for Computational Linguistics, Albuquerque, New Mexico (Apr 2025). <https://doi.org/10.18653/v1/2025.findings-naacl.336>, <https://aclanthology.org/2025.findings-naacl.336/>

15. Rauch, L., Akenmacher, M., Huseljic, D., Wirth, M., Bischl, B., Sick, B.: Activeglae: A benchmark for deep active learning with transformers. In: Machine Learning and Knowledge Discovery in Databases: Research Track: European Conference, ECML PKDD 2023, Turin, Italy, September 18–22, 2023, Proceedings, Part I. p. 55–74. Springer-Verlag, Berlin, Heidelberg (2023). https://doi.org/10.1007/978-3-031-43412-9_4, https://doi.org/10.1007/978-3-031-43412-9_4
16. Rauch, L., Wirth, M., Huseljic, D., Herde, M., Sick, B., Akenmacher, M.: No free lunch in active learning: Llm embedding quality dictates query strategy success (2025), <https://arxiv.org/abs/2506.01992>
17. Wei, Q., Franklin, A., Cohen, T., Xu, H.: Clinical text annotation—what factors are associated with the cost of time? In: AMIA Annual Symposium Proceedings. vol. 2018, p. 1552 (2018)
18. Zhao, Y., Zhang, W., Wang, H., Kawaguchi, K., Bing, L.: Adamergex: Cross-lingual transfer with large language models via adaptive adapter merging. arXiv preprint arXiv:2402.18913 (2024)
19. Zhen, C., Zheng, E., Kuang, J., Tso, G.J.: Enhancing llm-as-a-judge through active-sampling-based prompt optimization. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track). pp. 960–970 (2025)
20. Zouhar, V., Cui, P., Sachan, M.: How to select datapoints for efficient human evaluation of nlg models? arXiv preprint arXiv:2501.18251 (2025)

A Theoretical Results

A.1 Unbiased Estimators

In this section, we derive the unbiased estimators used in our experiments for Accuracy, Precision and Recall. All quantities are computed under importance sampling, where each selected sample i is drawn according to a probability q_i , and is assigned an importance weight $w_i = \frac{1}{Nq_i}$.

Unbiased Accuracy The unbiased estimator of accuracy is computed as:

$$\widehat{A} = \frac{1}{B} \sum_{i=1}^B \frac{A(f(x_i), y_i)}{Nq_i}$$

Unbiased Precision and Recall For each class $c \in \{1, \dots, C\}$, we define the following importance-weighted true positives as:

$$\widehat{TP}_c = \sum_{i=1}^M w_i \mathbb{1}[\hat{y}_i = c] \mathbb{1}[y_i = c].$$

Since the expectation of a ratio differs from the ratio of expectations, we cannot apply importance weighting to the denominator. Instead, we use the predicted instances from the full test set:

$$\mathbb{E} \left[\frac{U}{V} \right] \neq \frac{\mathbb{E}[U]}{\mathbb{E}[V]} \Rightarrow \mathbb{E} \left[\frac{\widehat{TP}_c}{\text{PI}_c} \right] \neq \frac{\mathbb{E}[\widehat{TP}_c]}{\mathbb{E}[\text{PI}_c]}$$

The predicted instances can be defined as:

$$\text{PI}_c = \sum_{i=1}^N \mathbb{1}[\hat{y}_i = c].$$

Same mechanism applies for the true instances, which are defined as:

$$\text{TI}_c = \sum_{i=1}^N \mathbb{1}[y_i = c].$$

The unbiased Precision (\widehat{P}) and Recall (\widehat{R}) are respectively defined as:

$$\widehat{P} = \frac{1}{C} \sum_{c=1}^C \frac{\widehat{TP}_c}{\text{PI}_c}$$

$$\widehat{R} = \frac{1}{C} \sum_{c=1}^C \frac{\widehat{TP}_c}{\text{TI}_c}$$

Unbiased ROUGE Given a predicted summary $f(x_i)$ and a reference summary y_i , the ROUGE-N score for a single example is defined as:

$$\text{ROUGE-N}(f(x_i), y_i) = \frac{\sum_{g \in \mathcal{G}_N(y_i)} \min(C(g, f(x_i)), C(g, y_i))}{\sum_{g \in \mathcal{G}_N(y_i)} C(g, y_i)}$$

where $\mathcal{G}_N(y_i)$ is the set of N-grams in the reference y_i and $C(g, s)$ denotes the count of N-gram g in sequence s [10].

Since ROUGE-N can be decomposed as a per-example average:

$$\text{ROUGE-N} = \frac{1}{N} \sum_{i=1}^N \text{ROUGE-N}(f(x_i), y_i)$$

we can apply the Inverse Probability Weighted Estimator to obtain an unbiased estimate:

$$\widehat{\text{ROUGE-N}} = \frac{1}{B} \sum_{i=1}^B \frac{\text{ROUGE-N}(f(x_i), y_i)}{Nq_i}$$

where $q_i = q(x_i; X_{1:i-1}, X)$ is the probability mass for datum x_i of being the next to be sampled, depending on the active testing strategy. This holds for the macro-averaged formulation of ROUGE, where the final score is obtained as the mean of per-example ROUGE scores.

Remark. Note that while Precision and Recall admit unbiased estimators (by fixing the denominator to the full test set counts), the F1 score does not. As a nonlinear function of P and R , unbiasedness does not transfer, and thus F1 cannot be estimated without bias under importance sampling.

A.2 Proof of Unbiasedness and Convergence

To prove unbiasedness, we need to show that $\mathbb{E}[\widehat{M}] = M$.

Starting with the expectation:

$$\mathbb{E}[\widehat{M}] = \mathbb{E} \left[\frac{1}{B} \sum_{i=1}^B \frac{M_i}{Nq} \right]$$

By linearity of expectation:

$$\mathbb{E}[\widehat{M}] = \frac{1}{B} \sum_{i=1}^B \mathbb{E} \left[\frac{M_i}{Nq_i} \right]$$

For each term in the sum:

$$\mathbb{E} \left[\frac{M_i}{Nq} \right] = \sum_{M_i} \frac{M_i}{Nq} \mathbb{P}(M_i)$$

Since q is the probability of selecting the index, when we multiply $\frac{M_i}{Nq}$ by q (the selection probability), we get:

$$\mathbb{E} \left[\frac{M_i}{Nq} \right] = \frac{M}{N}$$

Therefore:

$$\mathbb{E}[\widehat{M}] = \frac{1}{B} \sum_{i=1}^B \frac{M_i}{N} = M$$

Regarding the convergence in expectation, by the Law of Large Numbers, since each term $\frac{M_i}{Nq}$ is independent and identically distributed with finite expectation M , as $B \rightarrow \infty$:

$$\widehat{M} \xrightarrow{p} M$$

The variance of the estimator decreases as $1/B$, showing that it becomes more precise with more samples:

$$\begin{aligned} \text{Var}(\widehat{M}) &= \text{Var} \left(\frac{1}{B} \sum_{i=1}^B \frac{M_i}{Nq} \right) = \\ &= \frac{1}{B^2} \sum_{i=1}^B \text{Var} \left(\frac{M_i}{Nq} \right) \end{aligned}$$

Since the terms are independent, and assuming finite variance σ^2 :

$$\text{Var}(\widehat{M}) = \frac{\sigma^2}{B}$$

A.3 Limits of the estimators of *Kossen et al.* [3]

[3] define two estimators. We start analyzing the first one. In order to simplify the computations we assume that all the samples have an accuracy equal to 1,

obtaining a final accuracy of 1:

$$\begin{aligned}
 A^{\text{PURE}} &= \frac{1}{M} \sum_{m=1}^M \left(w_m + \frac{M-m}{N} \right) \text{Acc}_m, \\
 A^{\text{PURE}} &= \frac{1}{M} \sum_{m=1}^M \left(1 + \frac{M-m}{N} \right) 1, \\
 A^{\text{PURE}} &= \frac{1}{M} \sum_{m=1}^M 1 + \frac{1}{M} \sum_{m=1}^M \frac{M}{N} - \frac{1}{M} \sum_{m=1}^M \frac{m}{N} \\
 A^{\text{PURE}} &= \frac{1}{M} M + \frac{1}{M} \frac{M^2}{N} - \frac{1}{M} \frac{M(M+1)}{2N} \\
 A^{\text{PURE}} &= 1 + \frac{M}{N} - \frac{M+1}{2N} \\
 A^{\text{PURE}} &= 1 + \frac{2M - M - 1}{N} \\
 A^{\text{PURE}} &= 1 + \frac{M-1}{N}
 \end{aligned}$$

This means that the PURE, while working well in the case of the loss function, overestimates the value of the accuracy, easily reaching values which are bigger than 1. For example, if we assume a total number of samples $N = 500$ and a budget $B = 450$, we obtain an accuracy estimated $A^{\text{PURE}} = 1.898$.

Now we will focus our analysis on the second estimator proposed by [3], with the same assumption of all the accuracy values being equal to 1:

$$\begin{aligned}
A^{\text{LURE}} &= \frac{1}{M} \sum_{m=1}^M v_m \text{Acc}_m \\
A^{\text{LURE}} &= \frac{1}{M} \sum_{m=1}^M 1 + \frac{N-M}{N-m} \left(-1 + \right. \\
&\quad \left. + \frac{1}{(N-m+1)q} \right) \\
A^{\text{LURE}} &= \frac{1}{M} \sum_{m=1}^M 1 + \frac{N-M}{N-m} \left(\frac{1}{\frac{N-m+1}{N}} - 1 \right) \\
A^{\text{LURE}} &= 1 + \frac{1}{M} \sum_{m=1}^M \frac{N-M}{N-m} \left(\frac{1}{\frac{N-m+1}{N}} - 1 \right) \\
A^{\text{LURE}} &= 1 + \frac{1}{M} \sum_{m=2}^M \frac{N-M}{N-m} \left(\frac{N}{N-m+1} - 1 \right) \\
A^{\text{LURE}} &= 1 + \frac{N-M}{M} \sum_{m=2}^M \frac{1}{N-m} \left(\frac{m-1}{N-m+1} \right) \\
A^{\text{LURE}} &= 1 + \frac{N-M}{M} \sum_{k=N-M}^{N-2} \frac{N-1-k}{k(k+1)} \\
A^{\text{LURE}} &= 1 + \frac{N-M}{M} \left[\sum_{k=N-M}^{N-2} \frac{N-1}{k} - \frac{N}{k+1} \right] \\
A^{\text{LURE}} &= 1 + \frac{N-M}{M} \left[(N-1) \sum_{k=N-M}^{N-2} \frac{1}{k} \right. \\
&\quad \left. - N \sum_{k=N-M}^{N-2} \frac{1}{k+1} \right] \\
A^{\text{LURE}} &= 1 + \frac{N-M}{M} \left[(N-1) \sum_{k=N-M}^{N-2} \frac{1}{k} \right. \\
&\quad \left. - N \sum_{k=N-M+1}^{N-1} \frac{1}{k} \right]
\end{aligned}$$

Denoting by $H_N = \sum_{h=1}^N \frac{1}{h}$ we can rewrite

$$\sum_{k=N-M}^{N-2} \frac{1}{k} = H_{N-2} - H_{N-M}$$

and

$$\sum_{k=N-M+1}^{N-1} \frac{1}{k} = H_{N-1} - H_{N-M+1}.$$

Now

$$A^{\text{LURE}} = 1 + \frac{N-M}{M} \left[(N-1)(H_{N-2} - H_{N-M}) - N(H_{N-1} - H_{N-M+1}) \right]$$

$$A^{\text{LURE}} = 1 + \frac{N-M}{M} \left[N(H_{N-2} - H_{N-1}) + N(H_{N-M+1} - H_{N-M}) - H_{N-2} + H_{N-M} \right]$$

$$A^{\text{LURE}} = 1 + \frac{N-M}{M} \left[\frac{-N}{N-1} + \frac{N}{N-M+1} + -H_{N-2} + H_{N-M} \right]$$

$$A^{\text{LURE}} = 1 + \frac{N}{M} \left[-\frac{N-M}{N-1} + \frac{N-M}{N-M+1} \right] + \frac{N-M}{M} \left[-H_{N-2} + H_{N-M} \right]$$

We can see that LURE suffers of the same problem of LURE. For example, if we set $N = 500$ $B = 450$, we obtain an accuracy estimated $A^{\text{LURE}} = 1.724$.

A.4 Stopping Criterion convergence

Proof. We denote by $m_i := M(f(x_i), y_i)$. Notice that the sample size is deterministically B and $q_i > 0$ for all i . $B \rightarrow N$, hence $q_i \rightarrow 1$ and $q_{ij} \rightarrow 1$ for all $i \neq j$. $\frac{1}{N} \sum_{i=1}^N m_i^2 \leq M < \infty$ uniformly in N .

We denote the sample membership indicators by $I_i := \mathbf{1}\{i \in S\}$ and define:

$$b_i := m_i \left(\frac{1}{B} - \frac{1}{Nq_i} \right), \quad \text{so that} \quad (1)$$

$$D := \widehat{M}_{\text{random}} - \widehat{M}_{\text{HT}} = \sum_{i=1}^N I_i b_i. \quad (2)$$

Let $\Delta_{ij} := q_{ij} - q_i q_j$ denote the joint inclusion covariance (with the convention $\Delta_{ii} = q_i(1 - q_i)$). Note that $q_{ij} = \mathbb{P}(i, j \in S)$. We can see that the mean of the gap vanishes. Taking design expectations and using $\mathbb{E}(I_i) = q_i$:

$$\begin{aligned} \mathbb{E}[D] &= \frac{1}{B} \sum_{i=1}^N q_i m_i - \frac{1}{N} \sum_{i=1}^N m_i \\ &= \sum_{i=1}^N m_i \left(\frac{q_i}{B} - \frac{1}{N} \right). \end{aligned}$$

As $B \rightarrow N$ we have $q_i \rightarrow 1$ for every i , hence $\frac{q_i}{B} - \frac{1}{N} \rightarrow \frac{1}{N} - \frac{1}{N} = 0$. Using Eq. (2) to control the magnitudes of a_i , it follows that $\mathbb{E}[D] \rightarrow 0$.

Also the variance Gap vanishes. Because D is linear in the indicators, its design variance can be written in covariance form:

$$\text{Var}(D) = \sum_{i=1}^N \sum_{j=1}^N \Delta_{ij} b_i b_j. \quad (3)$$

$$\text{Var}(D) = \sum_{i \neq j} (q_i - q_i^2) b_i^2 + (q_{ij} - q_i q_j) b_i b_j \quad (4)$$

Since the sample size is fixed. Let's say $\sum_{k=1}^N I_k = B$ almost surely, then:

$$0 = \text{Cov}(I_i, \sum_{k=1}^N I_k) = \sum_{k=1}^N \text{Cov}(I_i, I_k).$$

So:

$$0 = \sum_{k=1}^N \Delta_{ik}.$$

Hence $\sum_{i \neq j} \Delta_{i,j} = -\Delta_{ii} = -q_i(1 - q_i)$. The total covariance weight goes to 0 as $B \rightarrow N$, indeed in this case $q_i \rightarrow 1$ and $1 - q_i \rightarrow 0$.

From Eq. (1), implies $b_i \rightarrow m_i \left(\frac{1}{N} - \frac{1}{N} \right) = 0$ and $-\Delta_{ij} \rightarrow 0$ for each pair $i \neq j$ as $B \rightarrow N$. The sequence $\{m_i\}$ has uniformly bounded second moments. Therefore, each summand in Eq. (4) tends to 0 and the finite sum converges to 0: $\text{Var}(D) \rightarrow 0$.

Combining the two results,

$$\mathbb{E}[D^2] = \text{Var}(D) + (\mathbb{E}[D])^2 \rightarrow 0,$$

so $D \rightarrow 0$ in L^2 , which implies $D \xrightarrow{P} 0$.

Proposition 2 (Convergence rate of the estimator gap). *Let $\widehat{M}_{\text{random}}^{(n)}$ be the unweighted sample mean and $\widehat{M}^{(n)}$ the Horvitz–Thompson estimator over a fixed budget of n annotated samples drawn according to inclusion probabilities*

$q_i^{(n)} \in (0, 1]$ with $\sum_{i=1}^N q_i^{(n)} = n$, and with metric values bounded $M_i \in [a, b]$. Then:

$$\mathbb{E} \left[|\widehat{M}_{\text{random}}^{(n)} - \widehat{M}^{(n)}| \right] \leq \frac{\sqrt{N}}{n} \|q^{(n)}\|_2 \sigma_M + \frac{(b-a)}{q_{\min}^{(n)} \sqrt{2n}},$$

where $\|q^{(n)}\|_2 = (\sum_i q_i^{(n)2})^{1/2}$, $\sigma_M^2 = \frac{1}{N} \sum_i (M_i - M)^2$, and $q_{\min}^{(n)} = \min_i q_i^{(n)}$. In particular:

1. (Uniform sampling). If $q_i^{(n)} = n/N$ for all i , then the bias is exactly zero and:

$$\mathbb{E} \left[|\widehat{M}_{\text{random}}^{(n)} - \widehat{M}^{(n)}| \right] \leq \frac{(b-a)N}{\sqrt{2}n^{3/2}} = \mathcal{O}(n^{-3/2}).$$

2. (Uncertainty sampling). If $q_i^{(n)} = nu_i/U$ with $U = \sum_j u_j$, then:

$$\mathbb{E} \left[|\widehat{M}_{\text{random}}^{(n)} - \widehat{M}^{(n)}| \right] \leq \frac{\sqrt{N}\|u\|_2}{U} \sigma_M + \frac{(b-a)U}{\sqrt{2}u_{\min}n^{3/2}},$$

where the first term is an irreducible bias independent of n in the regime $n \ll N$.

Proof. By the triangle inequality:

$$|\widehat{M}_{\text{random}}^{(n)} - \widehat{M}^{(n)}| \leq |\widehat{M}_{\text{random}}^{(n)} - M| + |\widehat{M}^{(n)} - M|. \quad (5)$$

Bounding $|\widehat{M}_{\text{random}}^{(n)} - M|$. Define $Z_i = \mathbf{1}[i \in S_n]$ so that $\mathbb{E}[Z_i] = q_i^{(n)}$. Since the budget is fixed, $\sum_{i=1}^N q_i^{(n)} = n$, and:

$$\mathbb{E}[\widehat{M}_{\text{random}}^{(n)}] - M = \frac{1}{n} \sum_{i=1}^N q_i^{(n)} M_i - \frac{1}{N} \sum_{i=1}^N M_i = \frac{1}{n} \sum_{i=1}^N q_i^{(n)} (M_i - M) + M \underbrace{\left(\frac{1}{n} \sum_{i=1}^N q_i^{(n)} - 1 \right)}_{=0},$$

where the last term vanishes by the fixed-budget constraint. Hence:

$$\mathbb{E}[\widehat{M}_{\text{random}}^{(n)}] - M = \frac{1}{n} \sum_{i=1}^N q_i^{(n)} (M_i - M).$$

Applying Cauchy–Schwarz:

$$\left| \mathbb{E}[\widehat{M}_{\text{random}}^{(n)}] - M \right| \leq \frac{1}{n} \left(\sum_{i=1}^N q_i^{(n)2} \right)^{\frac{1}{2}} \left(\sum_{i=1}^N (M_i - M)^2 \right)^{\frac{1}{2}} = \frac{\sqrt{N}}{n} \|q^{(n)}\|_2 \sigma_M. \quad (6)$$

Since the bias is deterministic, $\mathbb{E}[|\widehat{M}_{\text{random}}^{(n)} - M|] \leq |\mathbb{E}[\widehat{M}_{\text{random}}^{(n)}] - M|$, so Eq. (6) bounds the expected absolute deviation directly.

Bounding $|\widehat{M}^{(n)} - M|$. Since $\widehat{M}^{(n)}$ is unbiased for M and each term $M_i/q_i^{(n)}$ lies in $[a/q_i^{(n)}, b/q_i^{(n)}]$, applying Hoeffding's inequality gives:

$$\Pr(|\widehat{M}^{(n)} - M| \geq u) \leq 2 \exp\left(-\frac{2n^2 u^2}{\sum_{i \in S_n} (b-a)^2 / q_i^{(n)2}}\right) \leq 2 \exp\left(-\frac{2n q_{\min}^{(n)2} u^2}{(b-a)^2}\right).$$

Integrating over $u > 0$:

$$\mathbb{E}[|\widehat{M}^{(n)} - M|] \leq \int_0^\infty 2 \exp\left(-\frac{2n q_{\min}^{(n)2} u^2}{(b-a)^2}\right) du = \frac{(b-a)}{q_{\min}^{(n)} \sqrt{2n}}. \quad (7)$$

Combining. Taking expectations in Eq. (5) and substituting Eqs. (6) and (7) yields the main bound.

Case (i): Uniform sampling. If $q_i^{(n)} = n/N$ for all i , then $\sum_i q_i^{(n)}(M_i - M) = (n/N) \sum_i (M_i - M) = 0$, so the bias is *exactly* zero. With $q_{\min}^{(n)} = n/N$, Eq. (7) gives $(b-a)N/(\sqrt{2} n^{3/2})$.

Case (ii): Uncertainty sampling. If $q_i^{(n)} = nu_i/U$, then $\|q^{(n)}\|_2 = (n/U)\|u\|_2$ and $q_{\min}^{(n)} = nu_{\min}/U$. Substituting into Eq. (6):

$$\frac{\sqrt{N}}{n} \cdot \frac{n}{U} \|u\|_2 \cdot \sigma_M = \frac{\sqrt{N} \|u\|_2}{U} \sigma_M,$$

which is independent of n . The stochastic term gives $(b-a)U/(\sqrt{2} u_{\min} n^{3/2})$.

B Predictors’ classification performance

Table 4 shows the performance of the predictors in terms of classification, while Table 5 in terms of summarization. For the classification task, the prompt we use is the following:

This is a text classification task. The possible labels are [...] while the indices of the labels are [...]. Output only 'Label: the predicted index of the label that you predict'. This is the sentence to classify: [...].

For the summarization task, the prompt we use is the following:

This is a text summarization task. This is the sentence that must be summarized: [...] Output only the summary.

Dataset	Claude			Nova Pro			Qwen		
	Accuracy	Recall	F1	Accuracy	Recall	F1	Accuracy	Recall	F1
AG’s News	0.8814	0.8814	0.8813	<u>0.8082</u>	<u>0.8082</u>	<u>0.8066</u>	0.7863	0.7863	0.7886
Banking77	0.7925	0.7925	0.7996	<u>0.6942</u>	<u>0.6942</u>	<u>0.7072</u>	0.6581	0.6581	0.6781
DBPedia	0.9822	0.9822	0.9822	<u>0.9385</u>	<u>0.9385</u>	<u>0.9401</u>	0.9312	0.9312	0.9317
FNC-1	0.4806	0.4806	0.4462	0.2933	0.2933	0.1961	<u>0.4082</u>	<u>0.4082</u>	<u>0.3406</u>
QNLI	0.5072	0.5072	0.5988	<u>0.5017</u>	<u>0.5017</u>	<u>0.5484</u>	0.4882	0.4882	0.5033
SST-2	0.9564	0.9564	0.9564	0.9255	0.9255	0.9255	<u>0.9427</u>	<u>0.9427</u>	<u>0.9427</u>
TREC-6	<u>0.7980</u>	<u>0.7980</u>	<u>0.8009</u>	0.7440	0.7440	0.7395	0.9465	0.9465	0.9465
IMDB*	0.9621	0.9621	0.9621	<u>0.9519</u>	<u>0.9519</u>	<u>0.9519</u>	0.9465	0.9465	0.9465
PubMed*	0.7202	0.7202	0.7554	<u>0.6453</u>	<u>0.6453</u>	<u>0.6826</u>	0.6391	0.6391	0.6645
Emotion*	0.5830	0.5830	0.5776	<u>0.5695</u>	<u>0.5695</u>	<u>0.5630</u>	0.5830	0.5830	0.5784
Rotten*	0.9343	0.9343	0.9344	0.8987	0.8987	0.8986	<u>0.9071</u>	<u>0.9071</u>	<u>0.9071</u>
Multilingual*	0.8069	0.8069	0.8081	<u>0.7494</u>	<u>0.7494</u>	<u>0.7502</u>	0.7046	0.7046	0.7103

Table 4: Text classification metrics on the full datasets. For each metric and dataset, the best predictor is **bolded** while the second best is underlined.

Dataset	Claude	Nova Pro	Qwen
CNN*	0.5074	0.5078	0.4827
XLSum*	0.4427	0.4543	0.4620

Table 5: Summarization results on the full datasets in terms of ROUGE-1. For each metric and dataset, the best predictor is **bolded** while the second best is underlined.