Select, Label, Evaluate: Active Testing in NLP

Antonio Purificato[1,2], Maria Sofia Bucarelli[3,⋆], Andrea Bacciu[1], Fabrizio Silvestri[2], Amin Mantrach[1]

[1] Amazon, [2] Sapienza University of Rome, [3]Université Côte d'Azur, CNRS, Inria, I3S

Correspondence: purifian@amazon.com

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. We release our code at `https://github.com/amazon-science/NLPActiveTesting`.

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 (Liu et al., 2023) and question answering (Parekh et al., 2025). 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 (Kreutzer et al., 2022). 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 (Gururangan et al., 2018). 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 (Wei et al., 2018) 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.

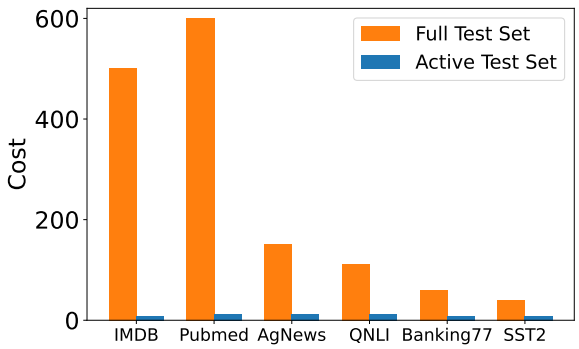

Figure 1: Annotation cost of using Active Testing vs full test set (↓ is better). Each sample has a cost of 0.02$ (LabelYourData). The active cost is the one required to reach an estimation quality $< 1\%$ wrt. full budget.

In recent years, Large Language Models (LLMs) and AI agents have increasingly been used for data generation, annotation, and validation at scale (Karim et al., 2025). While promising for reducing annotation effort, they raise concerns about quality, bias, and evaluation reliability, often still requiring human verification for test sets (Zhen et al.,

⋆ Work done while at Sapienza.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 26566–26584
October 24-29, 2026 ©2026 Association for Computational Linguistics

2025). Active Testing addresses these challenges by selecting informative test samples using only unlabeled inputs, before annotation. Since not all test instances contribute equally to evaluation (Kossen et al., 2021), Active Testing aims to estimate model performance with statistical guarantees while minimizing annotation cost and preserving agreement with full test set evaluation.

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 (Ahmadnia et al., 2025), 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 (Farquhar et al., 2021). Specifically, our contributions are:

- 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 $\mathbf{95}\%$ while preserving performance estimate accuracy within $\mathbf{1}\%$ 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.
- We show how Active Testing operates in multilingual settings, optimizing cost by strategically allocating budget across languages.
- 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 into two areas: approaches focusing on optimal sampling under budget constraints, and methods designed for Active Testing. However, budget-constrained sampling approaches, although sharing the goal of reducing annotation costs, are not comparable as they do not target test set optimization.

2.1 Sampling with a limited budget

Zouhar et al. (2025) 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. (2025) focus on human evaluation scenarios, Zhen et al. (2025) 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. (2025) 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. (2018) 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. (2021) generalize the Active Testing frame-

work 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. (2021), 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, Active Testing can match the accuracy of *i.i.d.* evaluation with significantly fewer labeled examples.

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

Li et al. (2025); Zhang et al. (2025) share our goal of reducing the cost of model evaluation by selecting a subset of test examples while maintaining reliable estimates of overall performance. In particular, Li et al. (2025) reduces the number of benchmark prompts that need to be evaluated by modeling dependencies among evaluation examples and actively selecting prompts that are informative for estimating overall benchmark performance. Similarly, Zhang et al. (2025) addresses benchmark prediction by leveraging the historical performance of previously evaluated models to predict the performance of new models from a small subset of benchmark examples. While these approaches are closely related to our objective, they differ in scope and methodology from our setting, where the focus is on active testing and sample selection for unbiased performance estimation, rather than on reducing benchmark prompts through dependency modeling or predicting model performance from historical evaluation data.

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 $f : X \to Y$. 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:

$$E(M) = \mid M(X_F) - M(X_A) \mid$$

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) (Liu et al., 2023) and ROUGE (for summarization) (Li et al., 2024) as these metrics, differently from loss, provide a reliable way to evaluate model performance. Given cost $C(x)$ of annotating sample x, we aim to find:

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

We seek to select the subset X_A minimizing the estimation error while ensuring that the total annotation cost does not exceed the 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 $\left(\mathbb{E}[\widehat{M}_{\text{random}}] = M\right)$ and it converges to the true empirical metric as $B \to N$, but it can produce estimates with large variance.

As pointed out in Farquhar et al. (2021); Kossen et al. (2021), 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 (Horvitz and Thompson, 1952) that provides an unbiased estimator which is defined as:

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

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 Appendix 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 Appendix A.1. Accuracy provides an interpretable, bounded measure of performance that directly reflects classification quality and it is the metrics used in related works (Kossen et al., 2021). For summarization we focus on ROUGE. Estimators like PURE and LURE (Farquhar et al., 2021) 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 Appendix A.3). This limitation is problematic when evaluating accuracy, where boundedness is an essential property for interpretation.

3.2 Active Testing Strategies

We evaluate Active Testing strategies for textual data. Although some are adapted from Active Learning, the two address different problems, as discussed in Kossen et al. (2021) and in Appendix B.2. Additional details on the Active Testing strategies can be found in Appendix B.1.

- Surrogate [RF, SVM] (Kossen et al., 2021): Selects samples using uncertainty estimated by a surrogate classifier trained on embeddings and accumulated labeled data. Supported models are a 300-tree Random Forest and an RBF-SVM. Uncertainty is measured as Shannon entropy over calibrated class probabilities, and samples with highest normalized entropy are selected. A limitation is the need to progressively collect ground-truth labels to train the surrogate.

- Uncertainty [GP, MI] (Kossen et al., 2021): Estimates uncertainty via Monte Carlo Dropout on the pretrained embedder (not the evaluated LLM), using ten forward passes. MI computes the difference between predictive entropy and average per-pass entropy, while GP measures latent-space variability through embedding standard deviations across passes.

- Coverage (Maharana et al., 2024): Maximizes geometric diversity in embedding space. Starting from a random sample, it iteratively selects the point farthest (Euclidean distance) from the last chosen sample, promoting broad exploration without clustering assumptions.

- Stratified Random: Samples proportionally from each class to preserve the original label distribution. However, it is not suitable for Active Testing, as it requires prior knowledge of class distributions in the unlabeled dataset.

- 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.

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 $\mid \widehat{M}_{\text{random}} - \widehat{M} \mid < \tau$ **then**
    break
  else
    continue
  end if
end for

3.3 Stopping Criterion

To the best of our knowledge, existing works on Active Testing (Kossen et al., 2021) 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.

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;

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.

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 $\mathbb{E}[\widehat{M}^{(t)} - \widehat{M}^{(t)}_{\text{random}}]$ converges to zero. When the empirical difference $|\widehat{M}^{(t)} - \widehat{M}^{(t)}_{\text{random}}| < \tau$ for a small threshold τ, we deem the estimator stable—further annotations are unlikely to change the performance estimate meaningfully. The user-defined convergence threshold τ specifies how close the unbiased estimate ($\widehat{M}$) must be to the unweighted random-sampling estimate ($\widehat{M}_{\text{random}}$) before we consider the active set sufficiently representative to stop annotating. More formally, the following proposition holds:

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

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

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]	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

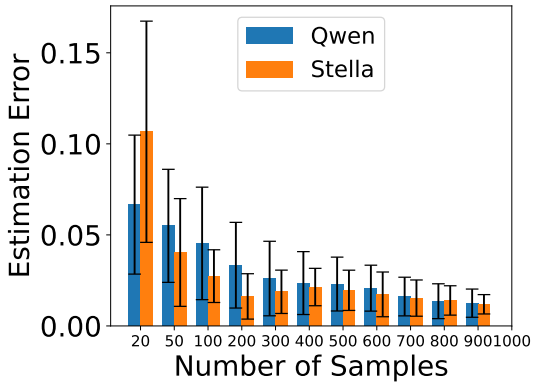

Figure 2: Estimation error with Agreement on AgNews using Qwen and Stella embedders (↓ is better).

4 Experiments

4.1 General Framework

We use the same experimental setup as Rauch et al. (2025). Our framework follows Pipeline 1. All strategies share a common 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). 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

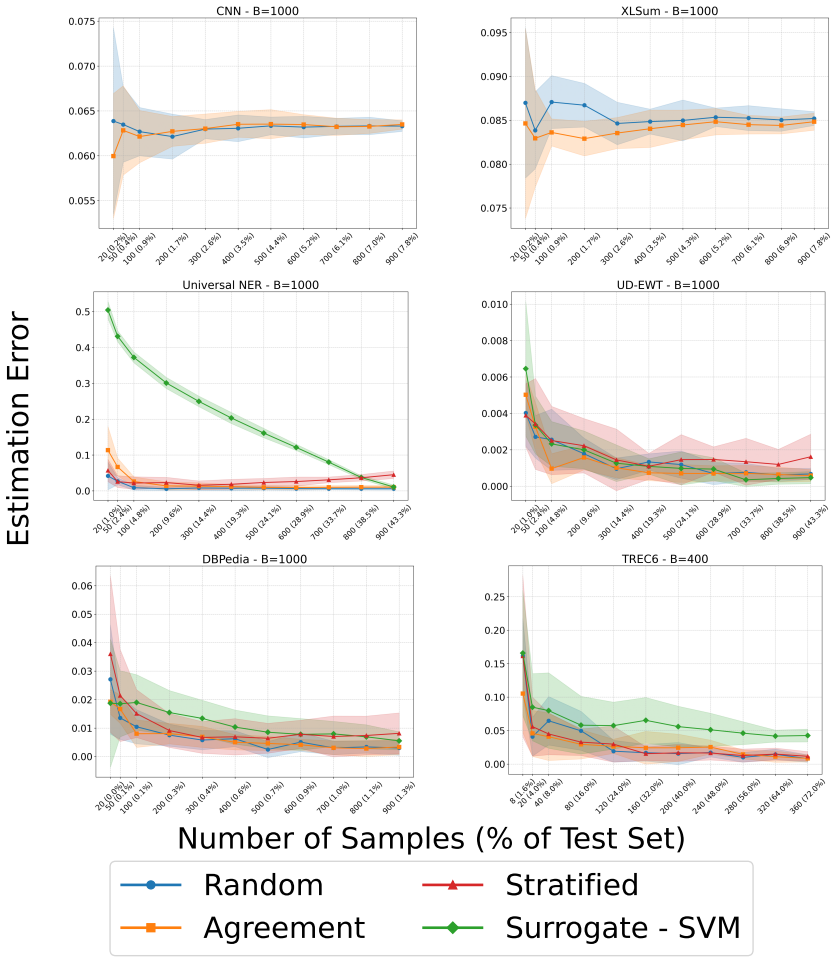

Figure 3: Active Testing results in terms of accuracy on all the datasets by fixing the embedding strategy and the predictor. We observe that the Agreement strategy is performing better than Surrogate on the majority of the datasets. With a reduced budget Agreement is also able to beat Random (↓ is better).

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 (Rauch et al., 2023) with the addition of more data (marked with *). We try to set, when possible, the same value of B for all the datasets ($B = 1000$). Additional statistics [1] are presented in Table 1.

Embedding Strategies Since the quality of the embeddings influences the results of Active Learning (Rauch et al., 2025), we investigate whether this extends to Active Testing. 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. The effect of different embedding strategies and predictors on Active Testing is analyzed in Appendix B.3.

Predictor To avoid bias from training, 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 Appendix B.4, where we also present the prompts used for classification and summarization. For POS tagging, we use BERT and DeBERTa fine-tuned, 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

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

samples from low-cost languages, transferring evaluation knowledge across linguistic boundaries.

For this experiment, we apply the steps presented in Pipeline 1. The difference is that dataset $\mathcal{D}$ contains samples from two 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 languages. This experiment assumes cross-lingual transferability, where a model performing well in one language achieves similar accuracy in the other. This is reasonable for multilingual models with shared vocabularies, as sentence-level classification tends to be language-agnostic (Zhao et al., 2024).

4.3 Minority Class Sample Detection

Active Testing presents advantages for highly imbalanced datasets. This is relevant in domains such as finance or 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. 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.

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 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 follows a similar trend, with Agreement achieving competitive performance. On the UD-EWT dataset, Agreement achieves 95% annotation reduction with an estimation error below 1%. For NER, the curves display more irregular patterns, likely due to class imbalance, yet Agreement remains consistently effective, confirming its robustness across different task types. Full results (also in terms of Precision and Recall) are available in the repository. An additional plot including all the Active Testing strategies is provided in Fig. 7.

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

	IMDB				QNLI			
B	$T_{Agreement}$	T_{Random}	$T_{Surrogate}$	Cost	$T_{Agreement}$	T_{Random}	$T_{Surrogate}$	Cost
100	5.51	0.01	12.47	2	4.97	0.01	14.31	2
500	5.51	0.001	12.52	10	5.05	0.01	14.32	10
900	5.52	0.01	59.05	18	5.08	0.01	43.06	18
S-Agreement	5.49	-	-	**1**	5.05	-	-	**2**
S-Random	-	0.01	-	2	-	NA	-	NA
S-Surrogate	-	-	12.74	4	-	-	14.71	8
Full	137.96	0.01	1,476.36	110	27.43	0.01	232.52	500

Stopping Mechanism Figure 5 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 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. Table 2 reports computational time and annotation cost for each strategy on IMDB and QNLI. 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. For clarity and due to space constraints, we report results on two datasets with additional plots in our repository.

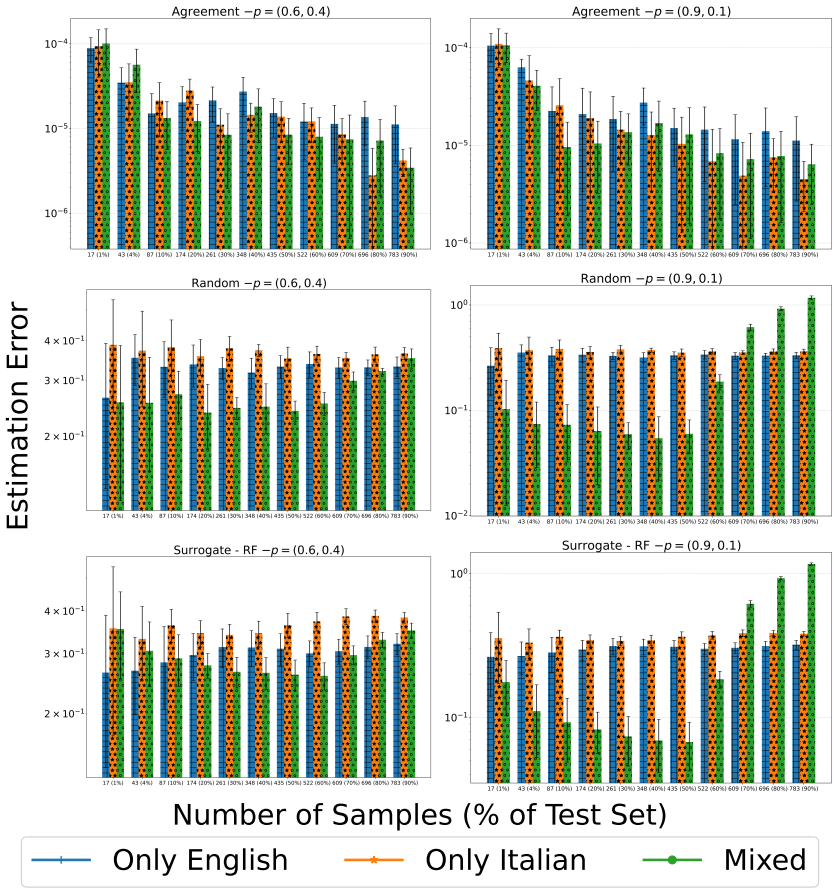

Figure 4: Multilingual results. Each row is a value of prior p. Each column refers to a method (↓ is better).

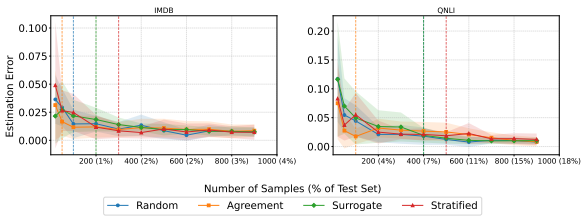

Figure 5: Results using the stopping criterion approach in order to avoid annotating all the samples (↓ is better). Vertical lines represents when the algorithms stop.

Agreement strategy We perform an ablation to evaluate Agreement with different numbers of attention heads (4, 8, and 16). The main results use 8 heads. The results are highly consistent, indicating that the method is not sensitive to this hyperparameter. Table 3 reports the estimation error in terms of accuracy obtained with different numbers of attention heads across multiple annotation budgets.

We further replace the randomly initialized attention module with the attention maps from the last layer of the pretrained BERT encoder in Table 4. The performance (estimation error in terms of accuracy) remains comparable overall, showing that Agreement does not rely on a learned attention mechanism.

Table 3: Estimation error in terms of accuracy for Agreement with different numbers of attention heads across annotation budgets. Best results are shown in bold.

Budget	4 Heads	8 Heads	16 Heads
100	0.0084 ± 0.0058	$\mathbf{0.0075 \pm 0.0062}$	0.0077 ± 0.0066
500	0.0047 ± 0.0037	$\mathbf{0.0038 \pm 0.0034}$	0.0039 ± 0.0034
900	0.0039 ± 0.0021	$\mathbf{0.0034 \pm 0.0025}$	0.0034 ± 0.0026

Overall, these experiments indicate that Agreement is robust to the attention configuration and that its performance is driven by the quality of the contextual representations rather than by a specific attention parameterization.

RQ2: Active Testing with Cost-Based Priors

Fig. 4 shows the results of the 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

Table 4: Estimation error in terms of accuracy for Agreement using randomly initialized attention and attention maps from the last layer of a pretrained BERT encoder. Best results are shown in bold.

Budget	Random Attention	BERT Attention
100	$\mathbf{0.0075 \pm 0.0012}$	0.0146 ± 0.0015
500	$\mathbf{0.0038 \pm 0.0004}$	0.0038 ± 0.0003
900	$\mathbf{0.0034 \pm 0.0005}$	0.0038 ± 0.0006

Table 5: Per-class F1 on minority class detection (↑ is better). Agreement and Surrogate outperform Random. Top: Emotion, bottom: PubMed.

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

balanced selection across languages. Allowing language selection based on a prior benefits Surrogate and Agreement, achieving estimation errors 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. Unlike results observed in previous experiments, in this setting the Active Testing strategies consistently outperform random sampling.

RQ3: Minority Class Sample Detection

Table 5 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. We report results on two datasets; results on the remaining data 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.

Limitations Our work focuses on static model evaluation settings, assuming fixed predictors and test distributions. Extending Active Testing to evolving models, continuously updated benchmarks, or interactive NLP systems remains an open direction for future research.

Acknowledgments

Maria Sofia Bucarelli acknowledges the French government National Research Agency (ANR) through the UCA JEDI (ANR-15-IDEX-01), through the EUR DS4H (ANR-17-EURE-004), and through the 3IA Cote d'Azur Investments in the project with the reference number ANR-23-IACL-0001. Fabrizio Silvestri and Antonio Purificato acknowledge the project SAP_RICERCA_2025_EVOLVE_SILVES_F_01.

References

Saeed Ahmadnia, Arash Yousefi Jordehi, Mahsa Hosseini Khasheh Heyran, Seyed Abolghasem Mirroshandel, Owen Rambow, and Cornelia Caragea. 2025. Active few-shot learning for text classification. In *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)*, pages 6677–6694, Albuquerque, New Mexico. Association for Computational Linguistics.

Sebastian Farquhar, Yarin Gal, and Tom Rainforth. 2021. On statistical bias in active learning: How and when to fix it. In *International Conference on Learning Representations*.

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

Daniel G Horvitz and Donovan J Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association*, 47(260):663–685.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. 2011. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*.

Henrik Imberg, Johan Jonasson, and Marina Axelson-Fisk. 2020. Optimal sampling in unbiased active learning. In *International Conference on Artificial Intelligence and Statistics*, pages 559–569. PMLR.

Md Monjurul Karim, Sangeen Khan, Dong Hoang Van, Xinyue Liu, Chunhui Wang, and Qiang Qu. 2025. Transforming data annotation with ai agents: A review of architectures, reasoning, applications, and impact. *Future Internet*, 17(8):353.

Andreas Kirsch, Joost Van Amersfoort, and Yarin Gal. 2019. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32.

Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. 2021. Active testing: Sample-efficient model evaluation. In *International Conference on Machine Learning*, pages 5753–5763. PMLR.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan Van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, and 1 others. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Dongyuan Li, Ying Zhang, Zhen Wang, Shiyin Tan, Satoshi Kosugi, and Manabu Okumura. 2024. Active learning for abstractive text summarization via llm-determined curriculum and certainty gain maximization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8959–8971.

Yang Li, Jie Ma, Miguel Ballesteros, Yassine Benajiba, and Graham Horwood. 2025. Active evaluation acquisition for efficient LLM benchmarking. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 35581–35602. PMLR.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Chaoqun Liu, Wenxuan Zhang, Guizhen Chen, Xiaobao Wu, Luu Anh Tuan, Chip Hong Chang, and Lidong Bing. 2023. Zero-shot text classification via self-supervised tuning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1743–1761.

Adyasha Maharana, Prateek Yadav, and Mohit Bansal. 2024. $\mathbb{D}^2$pruning: Message passing for balancing diversity & difficulty in data pruning. In *The Twelfth International Conference on Learning Representations*.

Phuc Nguyen, Deva Ramanan, and Charless Fowlkes. 2018. Active testing: An efficient and robust framework for estimating accuracy. In *International Conference on Machine Learning*, pages 3759–3768. PMLR.

Tanmay Parekh, Pradyot Prakash, Alexander Radovic, Akshay Shekher, and Denis Savenkov. 2025. Dynamic strategy planning for efficient question answering with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6038–6059, Albuquerque, New Mexico. Association for Computational Linguistics.

Lukas Rauch, Matthias Aßenmacher, Denis Huseljic, Moritz Wirth, Bernd Bischl, and Bernhard Sick. 2023. 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*, page 55–74, Berlin, Heidelberg. Springer-Verlag.

Lukas Rauch, Moritz Wirth, Denis Huseljic, Marek Herde, Bernhard Sick, and Matthias Aßenmacher. 2025. No free lunch in active learning: Llm embedding quality dictates query strategy success. *Preprint*, arXiv:2506.01992.

Qiang Wei, Amy Franklin, Trevor Cohen, and Hua Xu. 2018. Clinical text annotation–what factors are associated with the cost of time? In *AMIA Annual Symposium Proceedings*, volume 2018, page 1552.

Guanhua Zhang, Florian E. Dorner, and Moritz Hardt. 2025. How benchmark prediction from fewer data misses the mark. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*.

Yiran Zhao, Wenxuan Zhang, Huiming Wang, Kenji Kawaguchi, and Lidong Bing. 2024. Adamergex: Cross-lingual transfer with large language models via adaptive adapter merging. *arXiv preprint arXiv:2402.18913*.

Cheng Zhen, Ervine Zheng, Jilong Kuang, and Geoffrey Jay Tso. 2025. 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)*, pages 960–970.

Vilém Zouhar, Peng Cui, and Mrinmaya Sachan. 2025. How to select datapoints for efficient human evaluation of nlg models? *arXiv preprint arXiv:2501.18251*.

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 $c \in \{1, \ldots, C\}$, 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:

$$\mathbb{E}\left[\frac{U}{V}\right] \neq \frac{\mathbb{E}[U]}{\mathbb{E}[V]} \Rightarrow \mathbb{E}\left[\frac{\widehat{TP_c}}{\mathrm{PI}_c}\right] \neq \frac{\mathbb{E}\left[\widehat{TP_c}\right]}{\mathbb{E}[PI_c]}$$

The predicted instances can be defined as:

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

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

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

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

$$\widehat{\mathrm{P}} = \frac{1}{C}\sum_{c=1}^{C}\frac{\widehat{TP_c}}{\mathrm{PI}_c}$$

$$\widehat{\mathrm{R}} = \frac{1}{C}\sum_{c=1}^{C}\frac{\widehat{TP_c}}{\mathrm{TI}_c}$$

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:

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

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 (Lin, 2004).

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 $B \to \infty$:

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

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

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

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

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

A.3 Limits of the estimators of Farquhar et al. (2021)

Farquhar et al. (2021) define two estimators. We start analyzing the first one. To simplify the computations we assume that all the samples have an accuracy equal to 1, obtaining an accuracy of 1:

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

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 Farquhar et al. (2021), with the same assumption of all the accuracy values being equal to 1:

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

Denoting by $H_N = \sum_{h=1}^{N}\frac{1}{h}$, we have:

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

And:

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

Now:

$$
\begin{aligned}
A^{\text{LURE}} &= \frac{N-M}{M}\big[(N-1)(H_{N-2}-H_{N-M}) \\
&\quad - NH_{N-1} + NH_{N-M+1}\big] + 1 \\
&= 1+\frac{N-M}{M}\big[NH_{N-2} - NH_{N-1} \\
&\quad + NH_{N-M+1} - NH_{N-M} \\
&\quad - H_{N-2} + H_{N-M}\big] \\
&= 1+\frac{N-M}{M}\Bigg[-\frac{N}{N-1} \\
&\quad + \frac{N}{N-M+1} - H_{N-2} \\
&\quad + H_{N-M}\Bigg] \\
&= 1+\frac{N}{M}\Bigg[-\frac{N-M}{N-1} \\
&\quad + \frac{N-M}{N-M+1}\Bigg] \\
&\quad + \frac{N-M}{M}\big[-H_{N-2} + H_{N-M}\big].
\end{aligned}
$$

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. $B \to N$, hence $q_i \to 1$ and $q_{ij} \to 1$ 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:

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

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

Let $\Delta_{ij} := q_{ij} - q_i q_j$ denote the joint inclusion covariance (with the convention $\Delta_{ii} = q_i(1-q_i)$). Not that $q_{ij} = \mathbb{P}(i,j \in S)$ We can see that the mean of the gap vanishes. Taking design expectations and using $\mathbb{E}(I_i) = q_i$:

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

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

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

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

$$\mathrm{Var}(D) = \sum_{i \neq j} (q_i - q_i^2)\, b_i^2 + (q_{ij} - q_i q_j)\, b_i b_j \quad (3)$$

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

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

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 $B \to N$, indeed in this case $q_i \to 1$ and $1 - q_i \to 0$.

From Eq. (1), implies $b_i \to m_i\left(\frac{1}{N} - \frac{1}{N}\right) = 0$ and $-\Delta_{ij} \to 0$ for each pair $i \neq j$ as $B \to N$. The sequence $\{m_i\}$ has uniformly bounded second moments. Therefore, each summand in Eq. (3) tends to 0 and the finite sum converges to 0: $\mathrm{Var}(D) \to 0$.

Combining the two results,

$$\mathbb{E}\big[D^2\big] = \mathrm{Var}(D) + \big(\mathbb{E}[D]\big)^2 \;\longrightarrow\; 0$$

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

Proposition A.1 (Convergence rate of the estimator gap)**.** *Let $\widehat{M}^{(n)}_{\text{random}}$ 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:*

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

Where:

$$\|q^{(n)}\|_2 = \big(\sum_i q_i^{(n)^2}\big)^{1/2}$$
$$\sigma_M^2 = \frac{1}{N}\sum_i (M_i - M)^2$$
$$q^{(n)}_{\min} = \min_i q_i^{(n)}$$

In particular:

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

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

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

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

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

Proof. By the triangle inequality:

$$|\widehat{M}^{(n)}_{random} - \widehat{M}^{(n)}| \;\leq\; |\widehat{M}^{(n)}_{random} - M| \; + \; + |\widehat{M}^{(n)} - M| \quad (4)$$

Bounding $|\widehat{M}^{(n)}_{random} - M|$. 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:

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

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

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

Applying Cauchy–Schwarz:

$$\begin{aligned}&\left|\mathbb{E}[\widehat{M}^{(n)}_{random}] - M\right| \leq \\ &\leq \frac{1}{n}\left(\sum_{i=1}^{N} q_i^{(n)2}\right)^{\frac{1}{2}}\left(\sum_{i=1}^{N}(M_i - M)^2\right)^{\frac{1}{2}} = \\ &= \frac{\sqrt{N}}{n}\,\|q^{(n)}\|_2\,\sigma_M. \end{aligned} \tag{5}$$

Since the bias is deterministic, $\mathbb{E}[|\widehat{M}^{(n)}_{random} - M|] \leq |\mathbb{E}[\widehat{M}^{(n)}_{random}] - M|$, so Eq. (5) 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:

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

Integrating over $u > 0$:

$$\begin{aligned}\mathbb{E}[|\widehat{M}^{(n)} - M|] &\leq \int_0^\infty 2\exp\left(-\frac{2n\,q_{\min}^{(n)2}\,u^2}{(b-a)^2}\right)\mathrm{d}u = \\ &= \frac{(b-a)}{q_{\min}^{(n)}\sqrt{2n}}.\end{aligned} \tag{6}$$

Combining. Taking expectations in Eq. (4) and substituting Eqs. (5) and (6) 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. (6) 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. (5):

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

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

B Additional Results

B.1 Active Testing Detailed Strategies

Surrogate The Surrogate strategy selects test samples based on an auxiliary predictive model that estimates the uncertainty associated with each instance. The method first extracts embeddings for all unlabeled texts using the same pretrained encoder employed in the main evaluation pipeline and then trains a surrogate classifier on the resulting feature vectors paired with the ground-truth labels accumulated so far. Depending on the configuration, 2 alternative base learners can be used: a Random Forest with 300 trees and an RBF-kernel Support Vector Machine. In both cases, the base model is wrapped into `CalibratedClassifierCV` with five-fold internal calibration in order to reduce bias in predicted probabilities, which improves the stability of the entropy-based uncertainty measure.

Once the surrogate model is initialized, embeddings for all candidate samples are computed using the shared embedding extractor. The surrogate is then fitted on these embeddings and the current pool of labels. For each candidate sample, the surrogate outputs a vector of calibrated class probabilities. The uncertainty score is defined as the Shannon entropy of this distribution, computed as:

$$s_i = -\sum_c p_{i,c}\log(p_{i,c} + \epsilon),$$

where $\epsilon = 10^{-6}$ prevents numerical instabilities. Samples with the largest normalized scores are selected, corresponding to the most uncertain instances according to the surrogate model.

Uncertainty The Uncertainty strategy selects test cases by estimating predictive uncertainty through Monte Carlo Dropout. The method supports two acquisition functions: mutual information (MI), which measures epistemic uncertainty over model predictions, and a Gaussian prior–based score derived from the variability of intermediate representations (GP). All samples are first tokenized and then passed to the underlying transformer model with dropout layers kept active by switching the model to training mode and ten forward passes are performed for each batch of samples. For mutual information, each forward pass produces a set of logits which are converted into probability distributions via softmax. The resulting tensor has shape (M, B, C), where $M = 50$ is the number of Monte Carlo samples, B is the batch size, and C is the

number of classes. Uncertainty is computed as the difference between the entropy of the mean predictive distribution and the mean entropy of the individual Monte Carlo distributions. Formally, if $p_{i,c}^{(m)}$ denotes the predicted probability for class c and sample i at Monte Carlo iteration m, the mutual information is defined as:

$$\mathrm{MI}(i) = -\sum_c \bar{p}_{i,c} \log(\bar{p}_{i,c} + \varepsilon) + \\ -\frac{1}{M} \sum_{m=1}^{M} \left[\sum_c p_{i,c}^{(m)} \log(p_{i,c}^{(m)} + \varepsilon) \right]$$

with $\varepsilon = 10^{-6}$ to avoid numerical instability.

When the Gaussian prior acquisition is selected, the model instead extracts mean-pooled last-layer embeddings at each Monte Carlo pass. Averaging these embeddings across M iterations yields a mean representation for each sample, while the standard deviation across iterations provides a measure of uncertainty. The final acquisition score is computed as the squared sum of standard deviations across embedding dimensions, thus capturing the magnitude of variability in the latent space induced by dropout.

Coverage It aims at selecting test cases by maximizing the geometric spread of sampled points in the embedding space. This method chooses points that lie farthest from previously selected ones. All input texts are first transformed into dense vector representations using the pretrained multilingual encoder, following the same tokenization and embedding extraction pipeline used across all strategies. The resulting embedding matrix provides a geometric structure on which the coverage procedure operates.

The algorithm begins by selecting one initial point uniformly at random among all available embeddings. Given the set of selected indices S, let x_i denote the embedding of the last selected point. For each candidate index $j \notin S$, the method computes the Euclidean distance:

$$d(j) = \| x_j - x_i \|_2,$$

which measures how far each unselected point lies from the most recently added sample. The core idea is to iteratively append the point achieving the maximal distance, thus ensuring that new selections explore embedding regions that have not yet been represented. Points already in S are masked and assigned distance -1 to prevent re-selection. This process continues until the desired number of samples is reached. Because this strategy selects points explicitly at the periphery of previously covered regions, it ensures a broad geometric exploration of the embedding manifold without relying on clustering assumptions. The procedure is computationally efficient for moderate dataset sizes due to its incremental nature, and it is entirely deterministic except for the choice of the initial random point.

Agreement The Agreement strategy ranks unlabeled samples according to the level of disagreement among attention heads in a multi-head attention module applied to BERT representations. All text inputs are encoded with the pretrained BERT model from HuggingFace, using the corresponding tokenizer. For each batch, the tokenizer outputs PyTorch tensors that are forwarded through the BERT encoder. The final hidden states are then processed by a lightweight neural module consisting of an 8-head self-attention layer. The attention module receives inputs in the shape and returns both transformed token representations and attention weight matrices. For each sample, the uncertainty score is computed by taking the variance of the attention weights across the eight heads. More precisely, for each token position t, the tensor 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.

B.2 Active Testing vs Active Learning

As pointed out by Kossen et al. (2021), while acquisition functions for Active Learning have been extensively studied, they cannot be directly applied to Active Testing due to fundamental conceptual differences, as Active Testing does not involve using data to train a model.

There are three key distinctions:

- Conventional Active Learning methods tend to avoid regions with high aleatoric (inherent data) uncertainty while prioritizing areas with high epistemic (model) uncertainty. This principle underlies acquisition functions such as BALD (Houlsby et al., 2011) and BatchBALD (Kirsch et al., 2019). In contrast, for Active Testing, regions characterized by high

aleatoric uncertainty may be essential for accurate estimation.

- Imberg et al. (2020) demonstrate that the optimal acquisition strategy for Active Learning aims to minimize expected generalization error after training concludes. They illustrate how this objective necessitates additional considerations beyond simply reducing the variance of the loss estimator.

- Farquhar et al. (2021) establish that biased loss estimators can actually benefit the training process, as they frequently offset the inherent bias present in the training loss. However, this advantage disappears during testing, where minimizing bias becomes the primary objective.

A detailed plot containing the results on all the Active Testing strategies can be found in Fig. 7

B.3 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. 6, 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.

B.4 Predictors' classification performance

Table 6 shows the performance of the predictors in terms of classification, while Table 7 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.
```

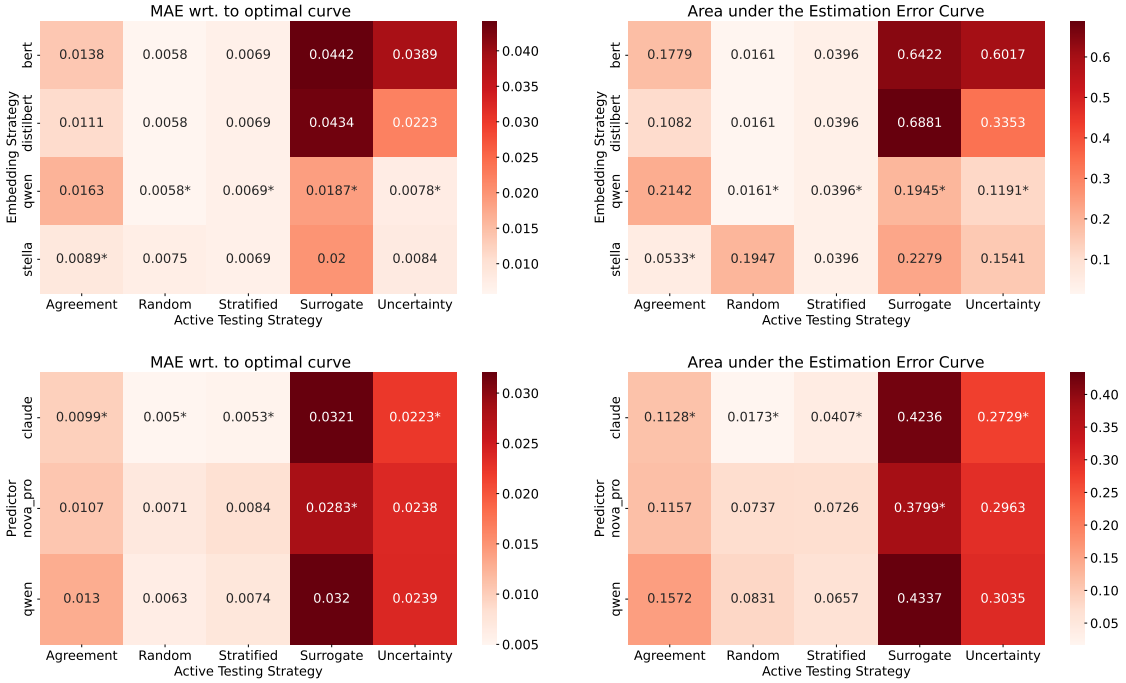

Figure 6: Confusion matrices with the Active Testing Strategies on the x-axis. The top figures have the embedding strategies on the y-axis while the bottom ones the predictors. The best results are marked with *. MAE wrt. optimal curve on the left and Area Under the Estimation Error Curve on the right. ↓ is better.

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

Table 6: 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 7: 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.

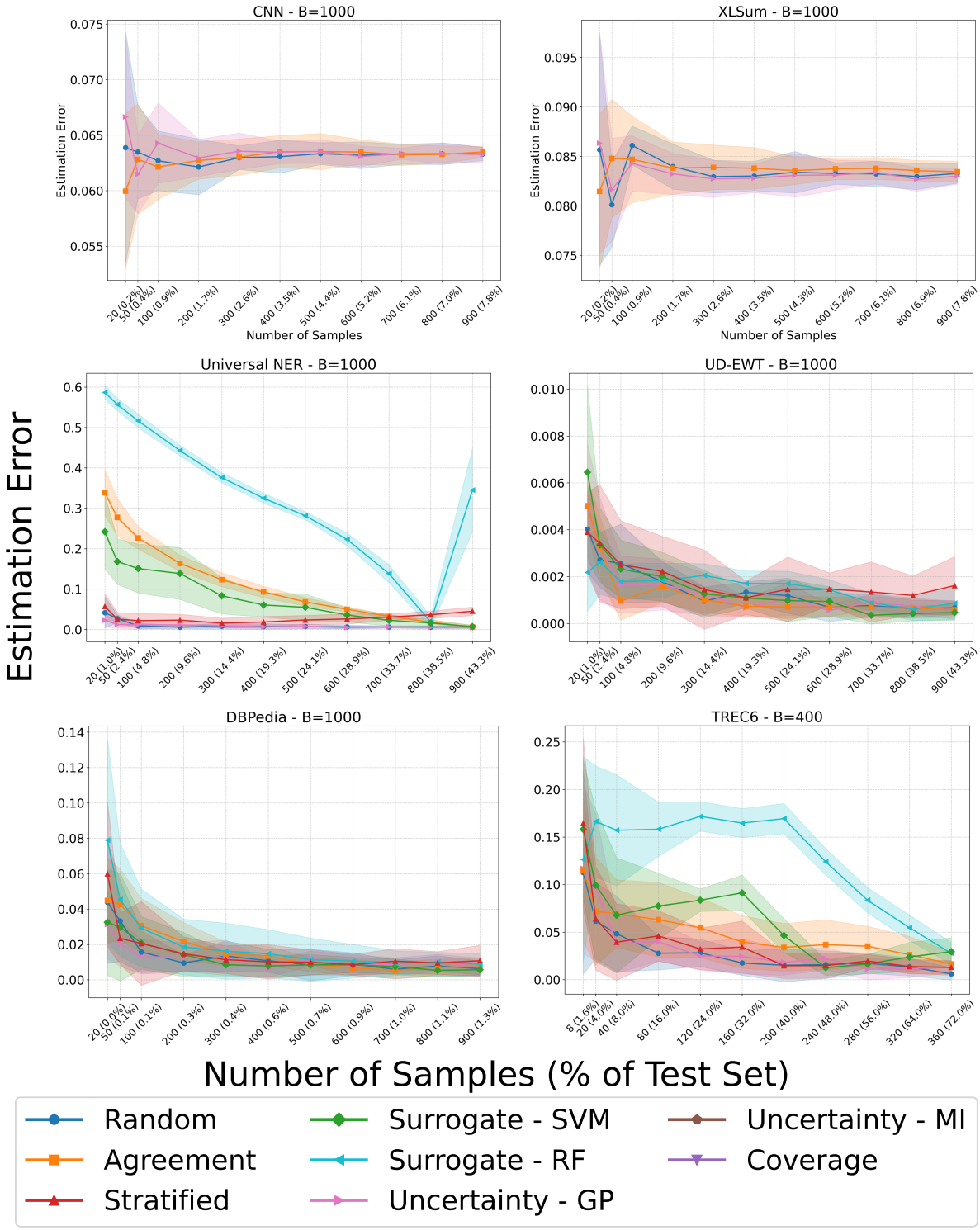

Figure 7: Active Testing results in terms of accuracy on all the datasets by fixing the predictor (Qwen for classification, DeBERTa finetuned for POS tagging and CNER for NER and the embedding strategy (Qwen).