Active Learning with Non-Uniform Costs for African Natural Language Processing

Bonaventure F. P. Dossou[1,2] **Ines Arous**[3] **Audrey Durand**[1,4,5] **Jackie Chi Kit Cheung**[1,2,5]
[1]Mila Québec AI Institute [2]McGill University [3]York University
[4]Université Laval [5]Canada CIFAR AI Chair, Mila

Abstract

Labeling datasets for African languages poses substantial challenges due to the diverse settings in which annotations are collected, leading to highly variable labeling costs. These costs vary with task complexity, annotator expertise, and data availability. Yet most active learning (AL) frameworks assume uniform annotation costs, which limits their applicability in real-world, resource-constrained scenarios. To address this, we introduce KnapsackBALD, a novel cost-aware active learning method that integrates the BatchBALD acquisition strategy with a 0-1 Knapsack optimization objective to select informative and budget-efficient samples. We evaluate KnapsackBALD on the MasakhaNEWS dataset, a multilingual news classification benchmark covering 11 African languages. Our method consistently outperforms seven strong active learning baselines, including BALD, BatchBALD, and stochastic sampling variants such as PowerBALD and Softmax-BALD, across all three cost scenarios. The performance gap widens as annotation cost imbalances become more extreme, demonstrating the robustness of KnapsackBALD in different cost settings. These findings show that when annotation costs are explicitly heterogeneous, cost-sensitive acquisition is critical for effective active learning, as demonstrated in African Languages NLP and similar settings. Our code base is open-sourced here.

1 Introduction

Annotating datasets for African languages is challenging due to the continent's vast linguistic diversity, which leads to a variety of annotation contexts and settings, further complicated by a lack of standardization (Eberhard et al., 2019). The scarcity of annotated texts and the primarily oral nature of many languages complicate annotation efforts, necessitating innovative approaches. This lack of well-structured annotated datasets hinders the development of NLP systems, making it difficult to train accurate models (Adelani et al., 2022), perpetuating the digital divide and restricting essential technology access for millions of speakers (Joshi et al., 2020).

Different annotation strategies were developed to mitigate these challenges, such as crowdsourcing through platforms like Amazon Mechanical Turk[1], leveraging community-driven initiatives (Nekoto et al., 2020), and employing semi-supervised learning techniques (Alabi et al., 2022). These strategies require effective budget management to collect high-quality data (Chen et al., 2021; Gan et al., 2017; Kulkarni et al., 2023). The costs of data labeling are highly variable and substantial, depending on the scenario. For instance, employing workers on Amazon Mechanical Turk is typically cheaper than traveling to local communities. Engaging with specific demographic groups, such as elderly speakers of endangered languages, often requires additional resources, including specialized outreach. However, despite higher upfront costs compared to crowdsourcing platforms such as Amazon Mechanical Turk, this approach is methodologically preferable, as it enables access to linguistically proficient speakers, improves annotation quality, and supports more ethically grounded and culturally appropriate data collection. Ethical requirements, such as ensuring informed consent and implementing robust privacy protections, also involve additional expenses. Consequently, effective budget management while reducing data dependency is of paramount importance.

Active learning plays a critical role in improving data efficiency for African languages (Dossou et al., 2022; Malhotra et al., 2019; Grießhaber et al., 2020; Ghimire et al., 2023). However, standard active learning setups mainly focus on model uncertainty, assuming uniform labeling costs for all samples. This assumption does not reflect real-

[1]https://www.mturk.com/

Findings of the Association for Computational Linguistics: EACL 2026, pages 6644–6656

world scenarios where labeling costs may vary significantly depending on the source and nature of the data. While several prior studies have investigated cost-sensitive active learning under non-uniform annotation budgets (Settles et al.; Donmez and Carbonell, 2008; Gao and Saar-Tsechansky, 2020), they largely overlook instance-level variability in annotation costs. We address this gap by explicitly modeling instance-level annotation costs and integrating them into the acquisition strategy. Our work builds on these insights by focusing on instance-level cost heterogeneity and integrating it directly into a batch mutual-information acquisition strategy. Addressing these non-uniform costs enables more effective use of limited annotation budgets, particularly in resource-constrained settings such as African-language NLP.

Motivated by these considerations, this paper introduces a novel active learning setting that allows for the definition of non-uniform labeling costs, enabling adaptive modeling of annotation expenses under explicit budget constraints. This setup provides the flexibility to incorporate multiple cost factors, ensuring the active learning process remains efficient and effectively uses limited annotation resources.

We propose KnapsackBALD for strategically selecting the most informative samples within this new setup. KnapsackBALD integrates BatchBALD (Kirsch et al., 2019), prioritizes labeling samples most likely to improve model performance. To further optimize resource allocation, we incorporate the 0-1 Knapsack optimization technique (Kellerer et al., 2004), ensuring that the selected samples provide the maximum value within the given budget. KnapsackBALD therefore effectively prioritizes the most informative data points while adhering to budgetary constraints, balancing model improvement and cost management.

Finally, we develop a cost-aware experimental setup that simulates annotation costs per sample based on annotator reliability and sample difficulty to evaluate our approach. We define three distinct cost scenarios, reflecting increasing levels of annotation cost, and introduce two ranking methods to evaluate the effectiveness of non-uniform active learning methods under different conditions. Our contributions are as follows:

- We design a novel active learning setup that incorporates non-uniform sample costs.
- We introduce KnapsackBALD, which balances sample informativeness and budget constraints to optimize sample selection.
- We define three distinct cost scenarios that reflect increasing levels of annotation costs and develop two ranking methods to evaluate the effectiveness of our approach under different scenarios.

On a news classification task involving 11 African languages, we show that KnapsackBALD achieves higher F1 scores than standard active learning baselines, particularly under extreme cost variations. Our results demonstrate that the performance gap increases consistently with higher annotation costs. This highlights the robustness of KnapsackBALD in non-uniform labeling conditions and establishes a baseline for future research in active learning for African languages.

2 Related Work

Active learning has been extensively studied as a strategy to reduce labeling effort by iteratively selecting informative examples. A variety of acquisition functions have been proposed, including uncertainty-based approaches such as BALD (Gal et al., 2017a), BatchBALD (Kirsch et al., 2019), PowerBALD, SoftmaxBALD, and SoftrankBALD (Kirsch et al., 2023).

Bayesian Active Learning by Disagreement (BALD) selects individual examples based on mutual information between predictions and model parameters, while BatchBALD extends this to batches by modeling joint informativeness. Other acquisition strategies focus on diversity and representativeness: core-set selection (Sener and Savarese, 2018a,b) optimizes geometric coverage, while BADGE (Ash et al., 2020) combines gradient uncertainty with clustering to select diverse batches. Unfortunately, these approaches are very computationally expensive (Kirsch et al., 2023). In our experiments, we compare against six strong active learning baselines, including random sampling, and multiple variants of BALD.

Contrastive sampling has also been proposed to improve informativeness by focusing on difficult counterexamples, but it can introduce instability by overemphasizing artificially hard examples (Margatina et al., 2021). In contrast, BatchBALD has consistently emerged as one of the strongest and most widely used batch-mode acquisition strategies in prior work (Gal et al., 2017b; Siddhant and Lipton, 2018; Kirsch et al., 2019). The proposed KnapsackBALD extends BatchBALD by in-

tegrating instance-specific labeling costs through a knapsack-based optimization. While more recent variants such as SoftrankBALD exist, we build on BatchBALD, given its established effectiveness and tractability in budgeted active learning.

Active learning with real annotation costs has been studied by empirically measuring and modeling annotation time across entire tasks and domains (Settles et al.; Settles, 2012). While these works demonstrate that labeling costs vary substantially across instances, cost information is used to analyze learning efficiency at the task level, rather than being explicitly integrated as an instance-level constraint within the acquisition function. Donmez and Carbonell (2008); Wang et al. (2017); Zhang and Chaudhuri (2015) model cost and accuracy trade-offs across multiple imperfect annotators. More recent studies consider multi-objective trade-offs between user effort and system utility (Lee et al., 2020; Chakraborty, 2020), or introduce budget-aware querying with weak and strong labelers (Gao and Saar-Tsechansky, 2020; Krishnamurthy et al., 2019; Xie et al., 2018). Beyond acquisition functions, other budget-sensitive research has looked at broader resource trade-offs. Chen et al. (2021) explores whether to allocate the budget to annotating new data or to cleaning existing labels. Kulkarni et al. (2023) optimizes budget allocation in graph labeling tasks via bandit-based strategies, while Gan et al. (2017) proposes incentive mechanisms for high-quality crowd annotations under budget limits. These works demonstrate that cost-sensitive active learning has been explored along multiple dimensions, including task-level budgets, annotator-level reliability, and broader system-level resource allocation. However, they typically do not directly integrate instance-level annotation costs into the acquisition function, nor do they consider joint batch selection under a global labeling budget.

Despite this growing body of work, most existing approaches model labeling costs at the task level, where costs reflect aggregate annotation effort across an entire task or dataset rather than guiding individual query decisions. This distinction aligns with prior categorizations of cost models in active learning, which separate task-level costs (overall annotation effort), annotator-level costs (variation due to annotator reliability or speed), and instance-level costs (costs that vary across individual examples and directly influence acquisition decisions) (Tomanek and Hahn, 2010; Huang et al., 2017; Herde et al., 2021). Related but complementary work has also studied cost-sensitive active learning in online and streaming settings. For example, Heuillet et al. (2024) formulates online active learning as a partial monitoring problem, where an agent trades off the cost of acquiring labels against the cost of prediction errors over time. This framework targets sequential decision-making under partial feedback and differs fundamentally from the pool-based setting considered in this paper, where the objective is to select batches of unlabeled instances for annotation under a fixed labeling budget. While both lines of work address cost-aware learning, they operate under distinct assumptions and cost definitions, and are therefore complementary.

In contrast, our method introduces a unified acquisition strategy that explicitly incorporates instance-level cost variation. These variations may be driven by factors such as length, difficulty, or annotator reliability. We integrate this cost modeling into the active learning loop using a budget-constrained knapsack optimization. We formulate batch selection as a constrained optimization problem to explicitly model annotation cost. This allows us to balance informativeness and cost at each acquisition step, thereby extending mutual-information-based active learning to realistic scenarios with heterogeneous annotation costs. This approach extends prior cost-aware methods to settings where costs are fine-grained and must be directly balanced against informativeness at each acquisition step.

3 Cost-Sensitive Active Learning with Non-Uniform Costs

We make the following assumptions based on classical active learning settings (Gal et al., 2017a; Jain et al., 2022; Kirsch et al., 2019, 2023). Let $\mathbb{X}$ and $\mathbb{Y}$ respectively denote the sample and label spaces. Let $\mathcal{D}^0_{\textbf{train}} = \{(x_i, y_i)\}_{i=1}^n \in \mathbb{X} \times \mathbb{Y}$ denote an initial available training dataset of n samples and let $\mathcal{D}^0_{\textbf{pool}} = \{\boldsymbol{x}_i\}_{i=1}^m \in \mathbb{X}$ denote a pool of m unlabeled samples. Let $\mathbf{f}^0$ denote an initial predictive model trained on $\mathcal{D}^0_{\textbf{train}}$. The goal is to leverage samples from $\mathcal{D}^0_{\textbf{pool}}$ to finetune $\mathbf{f}^0$.

The acquisition process is performed over several rounds $r = 1, 2, \ldots, R$. Let $\mathcal{D}^{r-1}_{\textbf{train}}$ and $\mathcal{D}^{r-1}_{\textbf{pool}}$ respectively denote the training dataset and the unlabeled pool to draw from at the beginning of round r. At each acquisition round r, we aim to construct a set (batch) $\mathbb{B}^r \subset \mathcal{D}^{r-1}_{\textbf{pool}}$ of samples to label. These samples are removed from the pool and added to

the training dataset along with their labels:

$$\mathcal{D}^r_{\textbf{pool}} = \mathcal{D}^{r-1}_{\textbf{pool}} \setminus \mathbb{B}^r$$
$$\mathcal{D}^r_{\textbf{train}} = \mathcal{D}^{r-1}_{\textbf{train}} \cup \{(x, \mathbf{f}^{r-1}(x))\}_{x \in \mathbb{B}^r}. \quad (1)$$

Round r yields a new model $\mathbf{f}^r$, which is finetuned on the updated training dataset $\mathcal{D}^r_{\textbf{train}}$.

Non-uniform costs: To model non-uniform labeling costs that are *fixed across active learning rounds*, we define $\mathbf{g}$ to be a cost modeling function (CMF) that provides the simulated cost to label any sample $x \in \mathbb{X}$:

$$\mathbf{g} : \mathbb{X} \mapsto \mathbb{R}^+.$$

We compute beforehand $g(x), \forall x \in \mathcal{D}^0_{\textbf{pool}}$.

Budget: To simulate a budget-constrained scenario, we also define a global available budget C, that should be respected over the entire acquisition process (i.e., over all acquisition rounds):

$$\sum_{r'=1}^{R} \left(\sum_{x \in \mathbb{B}^{r'}} \mathbf{g}(x) \right) \leq C. \quad (2)$$

The remaining budget C^r at the end of round r can be computed as follows:

$$C^r = C - \sum_{r'=1}^{r} \left(\sum_{x \in \mathbb{B}^{r'}} \mathbf{g}(x) \right). \quad (3)$$

Goal: In cost-sensitive active learning with non-uniform costs, the objective is to construct a sequence of batches $(\mathbb{B}^r)_{r=1}^R$ such that the final model $\mathbf{f}^R$ achieves maximal performance under a chosen evaluation metric $\text{Perf}(\mathbf{f}^R)$ (e.g., F1-score), subject to a labeling budget. Formally,

$$\max_{(\mathbb{B}^1, \ldots, \mathbb{B}^R)} \text{Perf}(\mathbf{f}^R) \quad \text{s.t.} \sum_{r=1}^{R} \sum_{x \in \mathbb{B}^r} \mathbf{g}(x) \leq C,$$

where $\mathbf{g}(x)$ is the labeling cost of item x and C is the total budget.

4 KnapsackBALD

KnapsackBALD, at a high level, combines three key steps: pre-selection, ranking, and knapsack optimization. First, we pre-select a set of promising candidates from the pool by estimating their informativeness. Next, we rank these candidates to prioritize examples that provide the best trade-off between informativeness and labeling cost. Finally, we apply a knapsack solver to select the actual batch $\mathbb{B}^r$, ensuring that the chosen examples maximize their collective value while respecting the remaining budget. This decomposition is motivated by both practical and theoretical considerations: pre-selection reduces computational overhead, ranking captures the relative merit of candidates, and the knapsack step enforces cost-awareness.

4.1 Description

We construct $\mathbb{B}^r$ in three steps:

1. Pre-Selection: Let $b \in \mathbb{N}_{>0}$ denote the number of samples to acquire at each round r from $\mathcal{D}^{r-1}_{\textbf{pool}}$ using an informativeness (acquisition) function $\hat{\mathbf{a}}$. Function $\hat{\mathbf{a}}$ takes $\mathcal{D}^{r-1}_{\textbf{pool}}$ and query size b as inputs to return a batch $\mathbb{B}^r$ of b highly informative samples. We compute the informativeness score of available samples $x \in \mathcal{D}^{r-1}_{\textbf{pool}}$. We thus construct a pre-selected set $\mathbf{A}$ defined as follow:

$$\mathbf{A}^* = \hat{\mathbf{a}}(\mathcal{D}^{r-1}_{\textbf{pool}}, b) = \{(x_i, \mathbf{a}(x_i))\}_{i=1}^b$$
$$\mathbf{A} = \mathbf{A}^* \cup \{\mathbf{g}(x_i) | x_i \in \mathbf{A}^*\}$$
$$= \{(x_i, \mathbf{g}(x_i), \mathbf{a}(x_i))\}_{i=1}^b$$

2. Ranking: We define a ranking function $\boldsymbol{\rho}$ to rank the acquired samples and select the top-k samples. We apply $\boldsymbol{\rho}$ on the pre-selected samples to obtain the set $\mathbf{A}^k \subseteq \mathbf{A}$. This set is composed of the top-k samples from $\mathbf{A}$, with their respective cost and informativeness scores given by:

$$\mathbf{A}^k = \boldsymbol{\rho}(\mathbf{A}) = \{(x_i, \mathbf{g}(x_i), \mathbf{a}(x_i))\}_{i=1}^k,$$

with $x_i \in \mathbf{A}$ for all i. While the knapsack optimization guarantees a budget-feasible solution using $(\mathbf{a}(x_i), \mathbf{g}(x_i))$, we find that using the knapsack step alone results in extreme solutions that select very few expensive but highly informative instances, or conversely many inexpensive but weakly informative ones, leading to worse overall performance. We empirically find that introducing an additional ranking step can mitigate this issue, as we describe here, rather than improving the estimation of sample informativeness. The ranking step shapes the candidate set before optimization, ensuring the knapsack better reflects the acquisition preferences. We discuss different ranking methods in Section 4.2.

3. $\mathbb{B}^r$ Construction with Knapsack: We define a knapsack function $\boldsymbol{\kappa}$ that selects samples within the

current budget C^r while maximizing informativeness. We run κ on $\mathbf{A}^k$ in order to obtain $\mathbb{B}^r \subseteq \mathbf{A}^k$:

$$\mathbb{B}^r = \arg\max_{\bar{\mathbf{A}} \subseteq \mathbf{A}^k} \sum_{x \in \bar{\mathbf{A}}} \mathbf{a}(x)\,, \text{subject to Eq. 2.}$$

We update the current budget C^r according to Eq. 3. Once $\mathbb{B}^r$ obtained, we update $\mathcal{D}^r_{\textbf{train}}$ and $\mathcal{D}^r_{\textbf{pool}}$ according to Eq. 1, and obtain a new predictive model $\mathbf{f}^r$.

4.2 Ranking Methods

In this section, we define two variants of Knapsack-BALD that depend on the type of ranking used in the second step of $\mathbb{B}^r$ construction.

Ranking Method 1: High-Scoring BatchBALD-Driven Knapsack (HS-BDK): The set $\mathbf{A}^k$ is obtained by selecting the top-k highest-informative samples. Next, the knapsack is solved using the remaining budget C^r and $\mathbf{A}^k$. This process results into a batch $\mathbb{B}^r$ of highly informative candidates that satisfy the knapsack constraints of the scenario.

Ranking Method 2: Low-Cost BatchBALD-Knapsack Selection (LC-BDK): The set $\mathbf{A}^k$ is obtained by selecting the top-k cheapest (according to simulated costs $\mathbf{g}(x)$ $\forall x \in \mathbf{A}$) samples. Then, as in HS-BDK, the knapsack is solved using the remaining budget C^r and the set $\mathbf{A}^k$. This process results in a batch $\mathbb{B}^r$ that satisfies the knapsack constraints. With HS-BDK, $\mathbb{B}^r$ typically contains fewer but more informative (and often costlier) samples, whereas LC-BDK tends to yield larger batches composed of cheaper samples that still contribute to model improvement.

5 Scenario Description

We propose three data-labeling scenarios to benchmark cost-sensitive active learning methods under non-uniform costs in simulated settings that capture real-world dynamics.

Scenario 1 (Non-Uniform Varying Costs): Cost function $\mathbf{g}$ generates random costs, following a continuous uniform distribution. In other words, for a given sample x, $\mathbf{g}(x) = e^t, t \sim \mathcal{U}_{[a,b]}\}$ where $\mathcal{U}_{[a,b]}$ is the uniform probability distribution in the interval $[a,b]$. We primarily use $a = 1$ and $b = 5$, but we also conduct experiments with different values of a and b to explore more extreme scenarios.

Scenario 2 (Annotator Reliability): Unlike Scenario 1, where costs are purely random, here, costs are tied to annotator reliability. For a sample x, we assign a reliability score λ_x drawn from a uniform distribution $\mathcal{U}_{[1,5]}$, where larger values correspond to more trustworthy annotators. This score is not a probability but rather a proxy variable that we map into costs via $\mathbf{g}(x) = e^{\lambda_x}$, ensuring that higher reliability is associated with higher annotation costs. In this setup, each sample is associated with a single reliability-derived cost, rather than multiple annotators, so informativeness is still computed independently through BatchBALD, while costs reflect annotator reliability as an external factor.

Scenario 3 (Sample Difficulty): Cost function $\mathbf{g}$ generates the costs based on the difficulty of the sample x. We use language model perplexity as our proxy for difficulty. In other words, to compute the difficulty of a sample x, we compute its perplexity β_x using a large language model that has been pretrained on the language of x. We also establish a positive correlation between difficulty and annotation cost (greater difficulty → higher sample cost). This explicitly means that the higher the perplexity, the higher the annotation cost. In this scenario, for a given input sample x, we define $\mathbf{g}(x) = e^{\beta_x}$.

These three scenarios represent plausible examples of real-world data annotation settings (Nguyen and Smeulders, 2004; Geva et al., 2019), but they are not exhaustive. In practice, many other factors (e.g., the availability of multilingual annotators) can influence annotation costs. However, because real-world instance-level annotation costs are not available, we simulate them in this work. These simulated scenarios are designed to isolate the effect of heterogeneous instance-level costs under a fixed labeling budget in a controlled pool-based active learning setting, rather than to exhaustively model all possible annotation processes.

5.1 Levels of Cost Extremity

To further assess the robustness of our methods, we propose to evaluate performance under progressively more extreme cost conditions. For each scenario, we define three levels of extremity:

Level 1 (Baseline Extremity): Uses the default cost ranges introduced for each scenario, serving as the reference point for comparison. These values are our experimental choice to represent a baseline level of cost variation. They are not fixed and could be instantiated differently depending on the experimental setting, with higher levels progressively expanding the ranges to simulate more extreme

conditions.

Level 2 (Moderate Extremity): Amplifies cost variability within each scenario. For **Scenario 1** and **Scenario 2**, we increase the upper bound b for uniform and exponential cost sampling from 5 to 10, effectively doubling the potential maximum cost relative to Level 1. For **Scenario 3**, we scale the exponent in the perplexity-based cost function by a factor of 5, which similarly expands the spread of costs compared to Level 1. We refer to this as a moderate extremity level because it substantially increases cost heterogeneity while still maintaining a balanced cost–informativeness ratio; higher levels push this ratio further, leading to more skewed and extreme scenarios.

Level 3 (High Extremity): Represents the most extreme annotation environment. The upper bound b in Scenarios 1 and 2 is further increased from 10 to 20, while the exponent in Scenario 3 is scaled by a factor of 10, creating highly skewed cost distributions. This setting mirrors real-world cases where annotation requires rare expertise or intensive effort, such as fine-grained linguistic annotation in low-resource languages (Geva et al., 2019), where a small subset of samples may be disproportionately costly to obtain.

This allows us to test whether our methods remain effective as annotation costs become more uneven across data points.

5.2 Normalized Cost and Budget Definition

The modeled costs $\mathbf{g}(x)$ are real positive numbers. To ensure consistency regardless of the scale of the costs, we normalize the costs so that (a) the average cost is 1, and (b) the total cost is roughly equal to the size of the initial pool dataset $\mathcal{D}^0_{\textbf{pool}}$. $\forall x \in \mathcal{D}^0_{\textbf{pool}}$, we re-define $\mathbf{g}(x)$, the normalized cost value of x as:

$$\underbrace{\hat{\mathbf{g}}(x)}_{\text{new value}} = \underbrace{\mathbf{g}(x)}_{\text{old value}} * \frac{|\mathcal{D}^0_{\text{pool}}|}{\sum_{x \in \mathcal{D}^0_{\text{pool}}} \underbrace{\mathbf{g}(x)}_{\text{old value}}} \tag{4}$$

We define the global budget as a percentage p of the total labeling cost of the initial pool:

$$C = \frac{p}{100} \sum_{x \in \mathcal{D}^0_{\textbf{pool}}} \hat{\mathbf{g}}(x) \tag{5}$$

6 Experiments

We ran our experiments in a multilingual classification setting with budget-aware active learning. For the experiments, we set $b = 80$, $k = 60$, $p = 30$, and $R = 5$.

6.1 Datasets and Evaluation Metric

We evaluate our methods on MasakhaNEWS (Adelani et al., 2023), a multilingual news topic classification benchmark spanning 17 African languages from diverse families (Afro-Asiatic, Niger-Congo, Indo-European) and regions across East, West, Central, and Southern Africa. Each article is labeled with one of seven topics: business, entertainment, health, politics, religion, sports, or technology. The dataset provides 2,000 to 12,000 articles per language from both international outlets (e.g., BBC, VOA) and local African sources (e.g., Gambuuze, Isolezwe).

MasakhaNEWS is particularly suited for our study for three reasons: (1) it is the most comprehensive publicly available benchmark for African news classification, covering diverse linguistic and regional contexts; (2) it offers reliable, human-annotated labels across multiple domains, making it valuable for assessing active learning where annotation costs are critical; and (3) it reflects realistic low-resource conditions, with modest per-language data and varying annotation costs tied to annotator availability and expertise. These properties make it an ideal testbed for exploring budget-aware active learning in African NLP.

We follow the official train, dev, and test splits for all languages. For our active learning experiments, we further split the official training set into two equal halves: one serves as $\mathcal{D}^0_{\textbf{train}}$, and the other becomes the unlabeled pool $\mathcal{D}^0_{\textbf{pool}}$, where we completely discard the ground-truth labels to mimic a true low-resource, unlabeled data scenario. This ensures that both $\mathcal{D}^0_{\textbf{train}}$ and $\mathcal{D}^0_{\textbf{pool}}$ originate from the same domain and distribution, preventing domain shift between the starting labeled data and the unlabeled pool, which could influence the downstream performance of the model (Adelani et al., 2022). Such a setup reflects realistic annotation workflows in African NLP, where an initial seed set may be labeled, but most available text remains unlabeled, requiring active querying under budget constraints.

The downstream performance is assessed on a test set after each round r. We evaluate effectiveness primarily using the **F1-score**. We report the final performance from the last acquisition round, averaging across five random seeds. Hyperparameters details are provided in Table 6.

Scenario	Method	amh	eng	fra	hau	ibo	lin	lug	orm	pcm	run	sna	swa	yor
Scenario 1	Random	0.8888	0.8689	0.8692	0.8657	0.8695	0.8415	0.7306	0.8754	0.9315	0.8708	0.9089	0.8250	0.8951
	BALD	0.9278	0.9079	0.9082	0.9047	0.9085	0.8805	0.7696	0.9144	0.9705	0.9098	0.9479	0.8640	0.9341
	SoftRank-BALD	0.9298	0.9099	0.9102	0.9067	0.9105	0.8825	0.7716	0.9164	0.9725	0.9118	0.9499	0.8660	0.9361
	Softmax-BALD	0.9308	0.9109	0.9112	0.9077	0.9115	0.8835	0.7726	0.9174	0.9735	0.9128	0.9509	0.8670	0.9371
	PowerBALD	0.9318	0.9119	0.9122	0.9087	0.9125	0.8845	0.7736	0.9184	0.9745	0.9138	0.9519	0.8680	0.9381
	RABS ($\lambda = 0.6$)	0.8900	0.8847	0.8900	0.8809	0.8856	0.8835	**0.8772**	0.8881	0.8993	0.8830	0.8906	**0.8856**	0.8900
	BatchBALD	**0.9338**	**0.9139**	**0.9142**	**0.9107**	**0.9145**	**0.8865**	0.7756	**0.9204**	**0.9765**	**0.9158**	**0.9539**	0.8700	**0.9401**
Scenario 2	Random	0.8910	0.8687	0.8776	0.8686	0.8731	0.8509	0.7702	0.8755	0.9380	0.8701	0.9116	0.8284	0.8980
	BALD	0.9300	0.9077	0.9166	0.9076	0.9121	0.8899	0.8092	0.9145	0.9770	0.9091	0.9506	0.8674	0.9370
	SoftRank-BALD	0.9320	0.9097	0.9186	0.9096	0.9141	0.8919	0.8112	0.9165	0.9790	0.9111	0.9526	0.8694	0.9390
	Softmax-BALD	0.9330	0.9107	0.9196	0.9106	0.9151	0.8929	0.8122	0.9175	0.9800	0.9121	0.9536	0.8704	0.9400
	PowerBALD	0.9340	0.9117	0.9206	0.9116	0.9161	0.8939	0.8132	0.9185	0.9810	0.9131	0.9546	0.8714	0.9410
	RABS ($\lambda = 0.6$)	0.8903	0.8854	0.8902	0.8815	0.8864	0.8842	**0.8778**	0.8888	0.8987	0.8836	0.8900	**0.8862**	0.8891
	BatchBALD	**0.9360**	**0.9137**	**0.9226**	**0.9136**	**0.9181**	**0.8959**	0.8152	**0.9205**	**0.9830**	**0.9151**	**0.9566**	0.8734	**0.9430**
Scenario 3	Random	0.8884	0.8683	0.8708	0.8730	0.8728	0.8516	0.8222	0.8757	0.9328	0.8683	0.9155	0.8348	0.8960
	BALD	0.9279	0.9065	0.9090	0.9110	0.9117	0.8904	0.8558	0.9137	0.9731	0.9076	0.9545	0.8727	0.9349
	SoftRank-BALD	0.9299	0.9085	0.9110	0.9130	0.9137	0.8924	0.8578	0.9157	0.9751	0.9096	0.9565	0.8747	0.9369
	Softmax-BALD	0.9309	0.9095	0.9120	0.9140	0.9147	0.8934	0.8588	0.9167	0.9761	0.9106	0.9575	0.8757	0.9379
	PowerBALD	0.9319	0.9105	0.9130	0.9150	0.9157	0.8944	0.8598	0.9177	0.9771	0.9116	0.9585	0.8767	0.9389
	RABS ($\lambda = 0.6$)	0.8894	0.8860	0.8896	0.8818	0.8868	0.8845	**0.8782**	0.8893	0.8982	0.8840	0.8895	**0.8866**	0.8887
	BatchBALD	**0.9339**	**0.9133**	**0.9158**	**0.9189**	**0.9177**	**0.8964**	0.8677	**0.9197**	**0.9810**	**0.9140**	**0.9605**	0.8798	**0.9411**

Table 1: Level 1 macro-F1 scores across the three scenarios. BatchBALD consistently outperforms other AL baselines, random sampling, and RABS. Bold scores indicate the best performance per language and scenario.

Level	Method	amh	eng	fra	hau	ibo	lin	lug	orm	pcm	run	sna	swa	yor	Avg. Relative % Gain
Level 1	Baseline	0.9338	0.9139	0.9142	0.9107	0.9145	0.8865	0.7756	0.9204	0.9765	0.9158	0.9539	0.8700	0.9401	–
	HS-BDK	**0.9396**	**0.9224**	**0.9226**	0.9131	**0.9304**	**0.9009**	**0.8540**	0.9215	0.9804	0.9187	**0.9630**	**0.8828**	0.9382	**+2.97%**
	LC-BDK	0.9365	0.9185	0.9219	**0.9147**	0.9260	0.8932	0.8146	**0.9229**	**0.9817**	**0.9191**	0.9607	0.8765	**0.9392**	+2.06%
Level 2	Baseline	0.9021	0.8902	0.8928	0.8850	0.8884	0.7708	0.7106	0.8827	0.9504	0.8975	0.9350	0.8403	0.9120	–
	HS-BDK	**0.9391**	0.9175	0.9206	0.9140	**0.9234**	**0.9021**	0.8297	0.9185	**0.9823**	0.9226	0.9588	0.8777	**0.9431**	**+4.92%**
	LC-BDK	0.9380	**0.9193**	**0.9219**	**0.9158**	0.9175	0.8739	**0.8312**	**0.9204**	0.9815	**0.9235**	**0.9597**	**0.8786**	0.9428	+4.52%
Level 3	Baseline	0.8552	0.8434	0.8398	0.8307	0.8345	0.8021	0.7887	0.8272	0.8804	0.8343	0.8489	0.8207	0.8397	–
	HS-BDK	0.9391	0.9175	0.9206	**0.9140**	**0.9234**	**0.9021**	0.8297	**0.9185**	**0.9823**	**0.9226**	**0.9588**	0.8777	**0.9431**	+10.64%
	LC-BDK	**0.9401**	**0.9183**	**0.9248**	0.9136	0.9212	0.8939	**0.8296**	0.9153	0.9798	0.9187	**0.9588**	**0.8798**	0.9397	**+10.69%**

Table 2: **Performance Evaluation for Scenario 1 across Different Levels of Extremity.** The last column shows each method's average relative percentage gain in F1-score over the baseline. Bold values indicate the best-performing method for each language and level.

Level	Method	amh	eng	fra	hau	ibo	lin	lug	orm	pcm	run	sna	swa	yor	Avg. Relative % Gain
Level 1	Baseline	0.9360	0.9137	0.9226	0.9136	0.9181	0.8959	0.8152	0.9205	**0.9830**	0.9151	0.9566	0.8734	**0.9430**	–
	HS-BDK	**0.9381**	0.9176	**0.9243**	0.9138	**0.9273**	**0.9056**	**0.8587**	0.9226	0.9797	0.9181	0.9609	0.8714	0.9373	**+2.59%**
	LC-BDK	0.9365	**0.9185**	0.9210	**0.9170**	0.9260	0.8932	0.8146	**0.9229**	0.9817	**0.9191**	**0.9615**	**0.8802**	0.9369	+1.87%
Level 2	Baseline	0.8956	0.8845	0.8872	0.8798	0.8827	0.7601	0.7012	0.8755	0.9450	0.8928	0.9301	0.8354	0.9065	–
	HS-BDK	**0.9364**	0.9184	0.9206	0.9132	**0.9220**	**0.9021**	0.8265	0.9159	**0.9831**	0.9206	0.9620	0.8764	**0.9455**	**+6.97%**
	LC-BDK	0.9349	**0.9171**	**0.9188**	**0.9176**	0.9212	0.8607	**0.8205**	**0.9148**	0.9819	**0.9198**	**0.9603**	**0.8746**	0.9440	+6.56%
Level 3	Baseline	0.8436	0.8354	0.8381	0.8263	0.8301	0.7935	0.7803	0.8202	0.8697	0.8305	0.8408	0.8156	0.8302	–
	HS-BDK	0.9316	0.9159	**0.9218**	**0.9143**	**0.9190**	**0.8645**	**0.8187**	**0.9245**	**0.9811**	**0.9272**	0.9549	0.8811	**0.9441**	**+11.98%**
	LC-BDK	**0.9354**	**0.9138**	0.9193	0.9085	0.9276	0.9053	0.8079	0.9160	0.9824	0.9185	**0.9587**	**0.8767**	0.9435	+11.74%

Table 3: **Performance Evaluation for Scenario 2 across Different Levels of Extremity.** The last column reports the average relative percentage gain of each method over the baseline. Bold values indicate the best-performing method for each language and level.

6.2 Model

The pretrained language model used in all our experiments is AfroXLMR-Large (Alabi et al., 2022), a multilingual transformer obtained by continued MLM pretraining of XLM-R Large on 17 African languages, including Amharic, Hausa, Igbo, Oromo, Swahili, Yoruba, and others, spanning major African language families, as well as three high-resource languages: English, French, and Arabic. We chose AfroXLMR-Large because it provides stronger language representations for African languages compared to standard multilingual models, thanks to its adaptation on large African text corpora. Prior work (Alabi et al., 2022) has demonstrated that AfroXLMR significantly improves downstream task performance over multilingual pretrained models such as XLM-R in low-resource African settings, making it particularly

Level	Method	amh	eng	fra	hau	ibo	lin	lug	orm	pcm	run	sna	swa	yor	Avg. Relative % Gain
Level 1	Baseline	0.9339	0.9133	0.9158	**0.9189**	0.9177	**0.8964**	**0.8677**	0.9197	**0.9810**	0.9140	0.9605	**0.8798**	0.9411	–
	HS-BDK	**0.9471**	0.9138	0.9195	0.9127	0.9208	0.8927	0.8517	**0.9231**	0.9755	**0.9212**	**0.9620**	0.8742	0.9382	**+1.55%**
	LC-BDK	0.9407	**0.9164**	**0.9218**	0.9139	**0.9260**	0.8894	0.8146	0.9204	0.9798	0.9202	0.9609	0.8758	**0.9422**	+1.29%
Level 2	Baseline	0.9023	0.8801	0.8892	0.8704	0.8951	0.7816	0.7253	0.8785	0.9402	0.8854	0.9157	0.8213	0.8996	–
	HS-BDK	**0.9417**	0.9175	**0.9236**	**0.9143**	**0.9302**	0.8657	0.8271	**0.9302**	0.9760	0.9153	**0.9620**	0.8729	**0.9417**	**+5.96%**
	LC-BDK	0.9355	**0.9180**	0.9230	0.9139	0.9187	**0.8939**	**0.8296**	0.9153	**0.9798**	**0.9187**	0.9588	**0.8798**	0.9397	+5.58%
Level 3	Baseline	0.8304	0.8227	0.8193	0.8157	0.8210	0.7845	0.7714	0.8120	0.8603	0.8254	0.8382	0.8102	0.8239	–
	HS-BDK	**0.9376**	0.9152	0.9152	**0.9151**	0.9123	**0.8205**	0.8020	**0.9177**	0.9766	**0.9217**	**0.9609**	0.8767	**0.9465**	+12.65%
	LC-BDK	0.9371	**0.9183**	**0.9248**	0.9139	**0.9163**	0.8144	**0.8640**	0.9160	**0.9804**	0.9147	0.9604	**0.8774**	0.9435	**+12.84%**

Table 4: **Performance Evaluation for Scenario 3 across Different Levels of Extremity.** Bold values indicate the best-performing method for each language and level. The last column reports the average relative percentage gain of each method over the baseline.

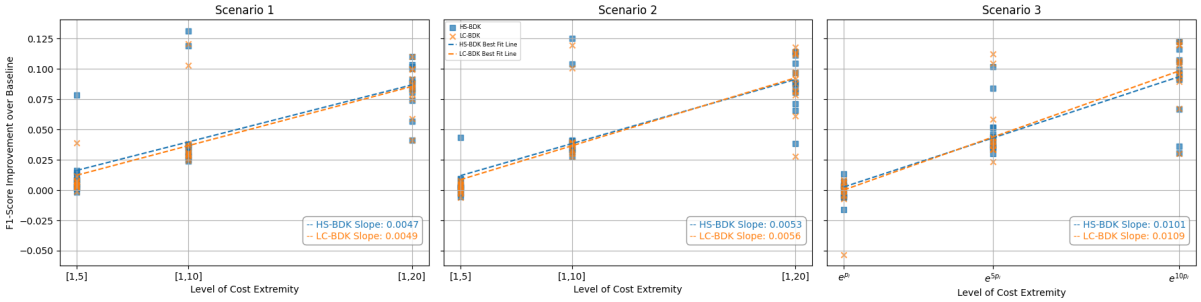

Figure 1: Language-level improvement of HS-BDK and LC-BDK over the baseline across increasing levels of acquisition cost extremity, shown per scenario. Each point represents the F1 score improvement for a given language at a specific cost level, and the dashed lines indicate the line of best fit. The slope values indicate the estimated rate of improvement as cost extremity increases. This visualization highlights how each method scales under varying cost conditions within each scenario.

well-suited for evaluating active learning methods where data scarcity and annotation costs are key challenges.

6.3 Baselines

We evaluate six widely known and used active learning acquisition functions, including **Random Sampling**, **BALD** (Gal et al., 2017a), **SoftRank-BALD**, **Softmax-BALD**, **PowerBALD** (Kirsch et al., 2023), and **BatchBALD** (Kirsch et al., 2019). All methods use the same query size, model architecture, and training configuration.

Additionally, we evaluate another self-created baseline, a lightweight **redundancy-aware budgeted selection** (RABS) that replaces BatchBALD's joint mutual information with a pointwise BALD score and a pairwise diversity penalty under C^r. RABS serves as a baseline by distilling BatchBALD's intuition into a lightweight redundancy-aware formulation, allowing us to test its effectiveness under budget constraints. Given the preselected set $\mathbf{A}$ (described in Section 4.1), we compute pairwise cosine similarities $\mathrm{sim}_{ij} \geq 0$, among elements of $\mathbf{A}$, and we construct $\mathbb{B}^r$ by solving

$$\max_{x_{i,j} \in \{0,1\}^b} \sum_{i=1}^{b} a_i x_i \; - \; \lambda \sum_{1 \leq i < j \leq b} \mathrm{sim}_{ij} x_i x_j$$

subject to Eq. 2, where $\lambda \geq 0$ trades off informativeness vs. redundancy. We linearize $x_i x_j$ via auxiliaries y_{ij} with $y_{ij} \leq x_i$, $y_{ij} \leq x_j$, $y_{ij} \geq x_i + x_j - 1$, yielding a mixed-integer linear program. The constraints ensure $y_{ij} = 1$ if both x_i and x_j are selected, so the penalty applies only to jointly selected similar items. We provide full details about **RABS** in the appendix.

7 Results and Discussion

In the uniform-cost setting, **BatchBALD** attains the strongest performance across most languages, so we use it both as the main baseline and as the informativeness function $\hat{\mathbf{a}}$ within KnapsackBALD.

Under non-uniform costs, **HS-BDK** and **LC-BDK** both improve over the baseline, with the magnitude of improvement increasing as cost extremity rises. Across our scenarios, gains are modest at low extremity and substantial at high extremity

(roughly +12%), indicating that cost-aware acquisition is particularly beneficial when costs are more heterogeneous.

The two variants exhibit different patterns. **LC-BDK** often shows larger relative gains at higher extremity in Scenarios 1 and 3, whereas **HS-BDK** displays more consistent improvements across languages and scenarios, especially in Scenario 2. These are descriptive observations from our experiments rather than claims about underlying mechanisms.

The slope analysis in Figure 1 supports these trends: slopes are positive for both methods as extremity increases, with **LC-BDK** generally exhibiting steeper slopes in Scenarios 1 and 3 and **HS-BDK** showing flatter but consistently positive slopes. Overall, both methods outperform the baseline more clearly as cost heterogeneity grows.

In summary, our experiments show that cost-aware batch selection yields increasing benefits as cost extremity rises. Between the two strategies, **LC-BDK** tends to yield larger improvements at high extremity, while **HS-BDK** provides steadier gains across settings. We report these trends as empirical findings specific to our scenarios and cost models.

8 Conclusion

This paper introduces a novel active learning setup with non-uniform costs to optimize sample selection for machine learning models in under-resourced African languages. We proposed two methods, HS-BDK and LC-BDK, and evaluated them at increasing levels of acquisition cost extremity. Our methods consistently outperformed the baselines, with the performance gap widening as cost extremity increased. Under the heterogeneous annotation-cost regimes considered, these results demonstrate that explicitly accounting for cost variability in sample selection can substantially improve performance, particularly in resource-constrained settings. Future research could explore extending these methods to other domains and cost settings to further enhance model adaptability and performance.

9 Limitations

This work introduces a novel cost-aware active learning framework tailored to under-resourced settings and demonstrates consistent improvements across diverse African languages and cost structures. While our approach broadly applies, we focus on text classification tasks and cost simulations aligned with realistic scenarios. Extending this framework to other tasks (e.g., machine translation or speech) and validating in real-world annotation environments remain valuable avenues for future work. Nonetheless, the methods are applicable to multiple tasks and offer strong empirical evidence of robustness and adaptability under varying annotation cost dynamics.

10 Ethical Considerations and Risks

This research does not involve the collection or annotation of sensitive data. All experiments are conducted on publicly available datasets, and the methods proposed are model-agnostic and task-general. Our primary objective is to improve efficiency and inclusion in low-resource language technology, particularly for African languages. While we simulate cost conditions, real-world deployment should ensure that annotation practices remain transparent, participatory, and aligned with local communities' needs. This includes respecting linguistic diversity and avoiding marginalizing less-represented dialects or speaker groups.

Acknowledgements

The authors acknowledge NVIDIA for providing computational resources. We also acknowledge funding support from the Canada CIFAR AI Chair program. This research was undertaken thanks in part to funding from the Connected Minds Canada First Research Excellence Fund for Ines Arous.

References

David Adelani, Jesujoba Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen Muhammad, Guyo Jarso, Oreen Yousuf, Andre Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin Ajibade, Tunde Ajayi, Yvonne Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthalu. 2022. A few thousand translations go a long way! leveraging pre-trained models for African news

translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3053–3070, Seattle, United States. Association for Computational Linguistics.

David Ifeoluwa Adelani, Marek Masiak, Israel Abebe Azime, Jesujoba Alabi, Atnafu Lambebo Tonja, Christine Mwase, Odunayo Ogundepo, Bonaventure F. P. Dossou, Akintunde Oladipo, Doreen Nixdorf, Chris Chinenye Emezue, sana al azzawi, Blessing Sibanda, Davis David, Lolwethu Ndolela, Jonathan Mukiibi, Tunde Ajayi, Tatiana Moteu, Brian Odhiambo, Abraham Owodunni, Nnaemeka Obiefuna, Muhidin Mohamed, Shamsuddeen Hassan Muhammad, Teshome Mulugeta Ababu, Saheed Abdullahi Salahudeen, Mesay Gemeda Yigezu, Tajuddeen Gwadabe, Idris Abdulmumin, Mahlet Taye, Oluwabusayo Awoyomi, Iyanuoluwa Shode, Tolulope Adelani, Habiba Abdulganiyu, Abdul-Hakeem Omotayo, Adetola Adeeko, Abeeb Afolabi, Anuoluwapo Aremu, Olanrewaju Samuel, Clemencia Siro, Wangari Kimotho, Onyekachi Ogbu, Chinedu Mbonu, Chiamaka Chukwuneke, Samuel Fanijo, Jessica Ojo, Oyinkansola Awosan, Tadesse Kebede, Toadoum Sari Sakayo, Pamela Nyatsine, Freedmore Sidume, Oreen Yousuf, Mardiyyah Oduwole, Tshinu Tshinu, Ussen Kimanuka, Thina Diko, Siyanda Nxakama, Sinodos Nigusse, Abdulmejid Johar, Shafie Mohamed, Fuad Mire Hassan, Moges Ahmed Mehamed, Evrard Ngabire, Jules Jules, Ivan Ssenkungu, and Pontus Stenetorp. 2023. Masakhanews: News topic classification for african languages. *Preprint*, arXiv:2304.09972.

Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to African languages via multilingual adaptive fine-tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2020. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*.

Shayok Chakraborty. 2020. Asking the right questions to the right users: Active learning with imperfect oracles. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3365–3372.

Derek Chen, Zhou Yu, and Samuel R Bowman. 2021. Clean or annotate: How to spend a limited data collection budget. *arXiv preprint arXiv:2110.08355*.

Pinar Donmez and Jaime G. Carbonell. 2008. Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, page 619–628, New York, NY, USA. Association for Computing Machinery.

Bonaventure F. P. Dossou, Atnafu Lambebo Tonja, Oreen Yousuf, Salomey Osei, Abigail Oppong, Iyanuoluwa Shode, Oluwabusayo Olufunke Awoyomi, and Chris Emezue. 2022. AfroLM: A self-active learning-based multilingual pretrained language model for 23 African languages. In *Proceedings of The Third Workshop on Simple and Efficient Natural Language Processing (SustaiNLP)*, pages 52–64, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

David Eberhard, Gary Simons, and Chuck Fennig. 2019. *Ethnologue: Languages of the World, 22nd Edition*.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017a. Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1183–1192. JMLR.org.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017b. Deep Bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1183–1192. JMLR. org.

Xiaoying Gan, Xiong Wang, Wenhao Niu, Gai Hang, Xiaohua Tian, Xinbing Wang, and Jun Xu. 2017. Incentivize multi-class crowd labeling under budget constraint. *IEEE Journal on Selected Areas in Communications*, 35(4):893–905.

Ruijiang Gao and Maytal Saar-Tsechansky. 2020. Cost-accuracy aware adaptive labeling for active learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(03):2569–2576.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Rupak Raj Ghimire, Bal Krishna Bal, and Prakash Poudyal. 2023. Active learning approach for fine-tuning pre-trained ASR model for a low-resourced language: A case study of Nepali. In *Proceedings of the 20th International Conference on Natural Language Processing (ICON)*, pages 82–89, Goa University, Goa, India. NLP Association of India (NLPAI).

Daniel Grießhaber, Johannes Maucher, and Ngoc Thang Vu. 2020. Fine-tuning bert for low-resource natural language understanding via active learning. *arXiv preprint arXiv:2012.02462*.

Marek Herde, Denis Huseljic, Bernhard Sick, and Adrian Calma. 2021. A survey on cost types, interaction schemes, and annotator performance models in selection algorithms for active learning in classification. *IEEE Access*, 9:166970–166989.

Maxime Heuillet, Ola Ahmad, and Audrey Durand. 2024. Neural active learning meets the partial monitoring framework. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, volume 244 of *Proceedings of Machine Learning Research*, pages 1621–1639. PMLR.

Sheng-Jun Huang, Jia-Lve Chen, Xin Mu, and Zhi-Hua Zhou. 2017. Cost-effective active learning from diverse labelers. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1879–1885.

Moksh Jain, Emmanuel Bengio, Alex Hernandez-Garcia, Jarrid Rector-Brooks, Bonaventure F. P. Dossou, Chanakya Ajit Ekbote, Jie Fu, Tianyu Zhang, Michael Kilgour, Dinghuai Zhang, Lena Simine, Payel Das, and Yoshua Bengio. 2022. Biological sequence design with GFlowNets. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 9786–9801. PMLR.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Hans Kellerer, Ulrich Pferschy, David Pisinger, Hans Kellerer, Ulrich Pferschy, and David Pisinger. 2004. Introduction to np-completeness of knapsack problems. *Knapsack problems*, pages 483–493.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. 2019. *BatchBALD: efficient and diverse batch acquisition for deep Bayesian active learning*. Curran Associates Inc., Red Hook, NY, USA.

Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal. 2023. Stochastic batch acquisition: A simple baseline for deep active learning. *Preprint*, arXiv:2106.12059.

Akshay Krishnamurthy, Alekh Agarwal, Tzu-Kuo Huang, John Langford, and Hal Daumé III. 2019. Active learning for cost-sensitive classification. *Journal of Machine Learning Research*.

Adithya Kulkarni, Mohna Chakraborty, Sihong Xie, and Qi Li. 2023. Optimal budget allocation for crowdsourcing labels for graphs. In *Uncertainty in Artificial Intelligence*, pages 1154–1163. PMLR.

Ji-Ung Lee, Christian M. Meyer, and Iryna Gurevych. 2020. Empowering Active Learning to Jointly Optimize System and User Demands. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4233–4247, Online. Association for Computational Linguistics.

Karan Malhotra, Shubham Bansal, and Sriram Ganapathy. 2019. Active learning methods for low resource end-to-end speech recognition. In *Interspeech*, pages 2215–2219.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active learning by acquiring contrastive examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 650–663, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkadir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Murhabazi Espoir, Elan van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Chinenye Emezue, Bonaventure F. P. Dossou, Blessing Sibanda, Blessing Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. 2020. Participatory research for low-resourced machine translation: A case study in African languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160, Online. Association for Computational Linguistics.

Hieu T. Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *Proceedings of the Twenty-First International Conference on Machine Learning*, ICML '04, page 79, New York, NY, USA. Association for Computing Machinery.

Ozan Sener and Silvio Savarese. 2018a. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*.

Ozan Sener and Silvio Savarese. 2018b. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*.

Burr Settles. 2012. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114.

Burr Settles, Mark Craven, and Lewis Friedland. Active learning with real annotation costs.

Aditya Siddhant and Zachary C Lipton. 2018. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. *arXiv preprint arXiv:1808.05697*.

Katrin Tomanek and Udo Hahn. 2010. A comparison of models for cost-sensitive active learning. In *Coling 2010: Posters*, pages 1247–1255, Beijing, China. Coling 2010 Organizing Committee.

Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. 2017. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600.

Kaige Xie, Cheng Chang, Liliang Ren, Lu Chen, and Kai Yu. 2018. Cost-sensitive active learning for dialogue state tracking. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 209–213, Melbourne, Australia. Association for Computational Linguistics.

Chicheng Zhang and Kamalika Chaudhuri. 2015. Active learning from weak and strong labelers. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

A RABS-General Formulation

We consider a selection problem over a set of n candidate items, indexed by $i \in \{1, \ldots, b\}$. Each item is associated with an informativeness score $a_i \geq 0$, a cost $g(x_i) \geq 0$, and may be compared pairwise with other items through a similarity measure $\text{sim}_{ij} \geq 0$ for each distinct pair (i, j) with $i < j$. The decision variable $x_i \in \{0, 1\}$ indicates whether item i is selected.

The objective is to select a subset of items that maximizes total informativeness while penalizing redundancy, subject to the total available budget C^r on the costs. The natural quadratic formulation of the problem is

$$\max \quad \sum_{i=1}^{b} a_i x_i \; - \; \lambda \sum_{1 \leq i < j \leq n} \text{sim}_{ij} x_i x_j \tag{6}$$

$$\text{s.t.} \quad \sum_{i=1}^{b} g(x_i) * x_i \leq C^r, \tag{7}$$

$$x_i, x_j \in \{0, 1\}, \quad i = 1, \ldots, b. \tag{8}$$

The first term in Eq. 6 rewards selection of informative items, while the second term subtracts a penalty for jointly selecting similar items, thereby encouraging diversity. The budget constraint Eq. 7 ensures that the total cost of selected items does not exceed C^r.

A.1 Linearization

The quadratic terms $x_i x_j$ render the model a binary quadratic program. To obtain an equivalent linear program, we introduce continuous auxiliary variables $y_{ij} \in [0, 1]$ for $i < j$, each representing the product $x_i x_j$. The linearized model is

$$\max \quad \sum_{i=1}^{b} a_i x_i \; - \; \lambda \sum_{1 \leq i < j \leq n} \text{sim}_{ij} y_{ij} \tag{9}$$

$$\text{s.t.} \quad y_{ij} \leq x_i, \quad \forall\, i < j, \tag{10}$$

$$y_{ij} \leq x_j, \quad \forall\, i < j, \tag{11}$$

$$y_{ij} \geq x_i + x_j - 1, \quad \forall\, i < j, \tag{12}$$

$$\sum_{i=1}^{b} g(x_i) * x_i \leq R, \tag{13}$$

$$x_i, x_j \in \{0, 1\}, \quad y_{ij} \in [0, 1]. \tag{14}$$

Constraints (Eqs. 10-12) ensure that $y_{ij} = 1$ if and only if both x_i and x_j are 1, which is necessary because the sim_{ij} term is subtracted in the objective and thus represents a penalty. Without the lower bound (Eq. 12), the solver could set $y_{ij} = 0$ to avoid the penalty even when both items are selected.

A.2 Single-Pair Formulation

For a specific pair (i, j), define $a := a_i$, $b := a_j$, and $c := \text{sim}_{ij} \geq 0$. With binary decisions $x_i, x_j \in \{0, 1\}$, the pairwise contribution to the objective, under an optional two-item budget $c_i x_i + c_j x_j \leq R$, is

$$\begin{aligned} \max \quad & a\, x_i \; + \; b\, x_j \; - \; c\, x_i x_j \\ \equiv\; & a\, x_i \; + \; b\, x_j \; - \; c\, y, \quad y \in [0, 1], \\ & \text{s.t. } y \leq x_i, \quad y \leq x_j, \quad y \geq x_i + x_j - 1, \\ & x_i, x_j \in \{0, 1\}, \quad c_i x_i + c_j x_j \leq R \;\; \text{(optional)}. \end{aligned} \tag{15}$$

Here y is an auxiliary variable exactly equal to $x_i x_j$ in any optimal binary solution, ensuring the penalty c is applied only when both items are selected.

A.3 Results and Analysis

Across all three scenarios, the redundancy penalty exhibits a stable, shallow optimum: $\lambda = 0.6$ yields the highest Macro-F1 for nearly all languages in Scenarios 1 and 2, and for most languages in Scenario 3 (with a minor exception where $\lambda = 0.8$ is marginally better). Performance varies little over the sweep $\lambda \in \{0.5, 1.0\}$, indicating that the method is not overly sensitive to moderate changes in the penalty strength. Larger λ values tend to slightly under-select in dense regions (lower scores at $\lambda = 1.0$), while smaller values reduce the benefit of de-duplication (lower scores at $\lambda = 0.5$).

Scenario	λ	amh	eng	fra	hau	ibo	lin	lug	orm	pcm	run	sna	swa	yor
Scenario 1	1.0	0.8891	0.8832	0.8887	0.8795	0.8842	0.8821	0.8759	0.8868	0.8984	0.8817	0.8899	0.8846	0.8900
	0.8	0.8898	0.8841	0.8894	0.8803	0.8851	0.8830	0.8767	0.8876	0.8990	0.8826	0.8904	0.8852	0.8900
	0.7	0.8899	0.8844	0.8897	0.8806	0.8853	0.8832	0.8769	0.8879	0.8992	0.8828	0.8905	0.8854	0.8900
	0.6	**0.8900**	**0.8847**	**0.8900**	**0.8809**	**0.8856**	**0.8835**	**0.8772**	**0.8881**	**0.8993**	**0.8830**	**0.8906**	**0.8856**	**0.8900**
	0.5	0.8897	0.8840	0.8892	0.8801	0.8849	0.8828	0.8765	0.8874	0.8989	0.8824	0.8902	0.8850	0.8898
Scenario 2	1.0	0.8883	0.8840	0.8892	0.8802	0.8850	0.8830	0.8768	0.8874	0.8976	0.8825	0.8892	0.8851	**0.8894**
	0.8	0.8902	0.8850	0.8899	0.8811	0.8860	0.8838	0.8774	0.8885	0.8983	0.8833	0.8898	0.8859	0.8893
	0.7	0.8901	0.8852	0.8900	0.8813	0.8862	0.8840	0.8776	0.8887	0.8986	0.8835	0.8899	0.8861	0.8892
	0.6	**0.8903**	**0.8854**	**0.8902**	**0.8815**	**0.8864**	**0.8842**	**0.8778**	**0.8888**	**0.8987**	**0.8836**	**0.8900**	**0.8862**	0.8891
	0.5	0.8899	0.8848	0.8896	0.8808	0.8857	0.8836	0.8771	0.8882	0.8982	0.8831	0.8896	0.8857	0.8891
Scenario 3	1.0	0.8876	0.8851	0.8885	0.8810	0.8861	0.8836	0.8775	0.8882	0.8968	0.8831	0.8886	0.8857	0.8887
	0.8	0.8891	0.8856	0.8892	0.8816	0.8865	0.8842	0.8779	0.8890	0.8979	0.8838	0.8892	0.8864	**0.8889**
	0.7	0.8892	0.8858	0.8894	0.8817	0.8866	0.8843	0.8780	0.8891	0.8981	0.8839	0.8894	0.8865	0.8888
	0.6	**0.8894**	**0.8860**	**0.8896**	**0.8818**	**0.8868**	**0.8845**	**0.8782**	**0.8893**	**0.8982**	**0.8840**	**0.8895**	**0.8866**	0.8887
	0.5	0.8890	0.8855	0.8890	0.8813	0.8863	0.8841	0.8777	0.8887	0.8978	0.8837	0.8891	0.8862	0.8886

Table 5: Macro-F1 scores across Scenarios 1, 2, and 3 for different values of the diversity weight λ. Best per language (column) within each scenario block is in **bold**.

Taken together, the results support using a small default redundancy weight, $\lambda \approx 0.6$, and optionally tuning within $[0.6, 0.8]$ if a development round is available.

B Experimental Hyperparameters

Hyperparameter	**Value**
model experiment seeds	1, 2, 3, 4, 5
max_length	164
cpu_per_node	6
ram_per_node	48gb
batch_size	16
num_epochs	10
save_steps	500000
bert_model	`davlan/afro-xlmr-large`
gpu type	rtx8000
number of gpus	2
learning rate	2e-5
gradient accumulation steps	2
model type	xlmroberta

Table 6: Experimental Setup for Model Training