D-CALM: A Dynamic Clustering-based Active Learning Approach for Mitigating Bias

Sabit Hassan and Malihe Alikhani
School of Computing and Information
University of Pittsburgh, Pittsburgh, PA
{sah259,malihe}@pitt.edu

Abstract

Despite recent advancements, NLP models continue to be vulnerable to bias. This bias often originates from the uneven distribution of real-world data and can propagate through the annotation process. Escalated integration of these models in our lives calls for methods to mitigate bias without overbearing annotation costs. While **active learning (AL)** has shown promise in training models with a small amount of annotated data, AL's reliance on the model's behavior for selective sampling can lead to an accumulation of unwanted bias rather than bias mitigation. However, infusing clustering with AL can overcome the bias issue of both AL and traditional annotation methods while exploiting AL's annotation efficiency. In this paper, we propose a novel adaptive clustering-based active learning algorithm, **D-CALM**, that dynamically adjusts clustering and annotation efforts in response to an estimated classifier error-rate. Experiments on eight datasets for a diverse set of text classification tasks, including emotion, hatespeech, dialog act, and book type detection, demonstrate that our proposed algorithm significantly outperforms baseline AL approaches with both pretrained transformers and traditional Support Vector Machines. **D-CALM** showcases robustness against different measures of information gain and, as evident from our analysis of label and error distribution, can significantly reduce unwanted model bias.

1 Introduction

While NLP models have experienced groundbreaking advancements in performance and functionality in recent years, they have been under scrutiny for exhibiting bias (Lu et al., 2020; Ahn and Oh, 2021; Kiritchenko and Mohammad, 2018). As noted by Davidson et al. (2019), classifier bias can stem from distribution in training data rather than the classifier itself. This bias is complex and can manifest in various forms, including racial, gender-based, and other types of discrimination. For example, in a hatespeech dataset, hatespeech against Persons of Color might be underrepresented, leading to a model biased against Persons of Color (Figure 1). Since the true distribution of data is unknown prior to labeling, ridding these models of such unwanted bias would require annotating a large number of samples to ensure that minority groups are well-represented in the data, incurring much higher cost, time, and effort. As such, we are in need of methods that can mitigate unwanted bias without overwhelming annotation costs. We address the problem of bias with a novel *clustering-based active learning* approach.

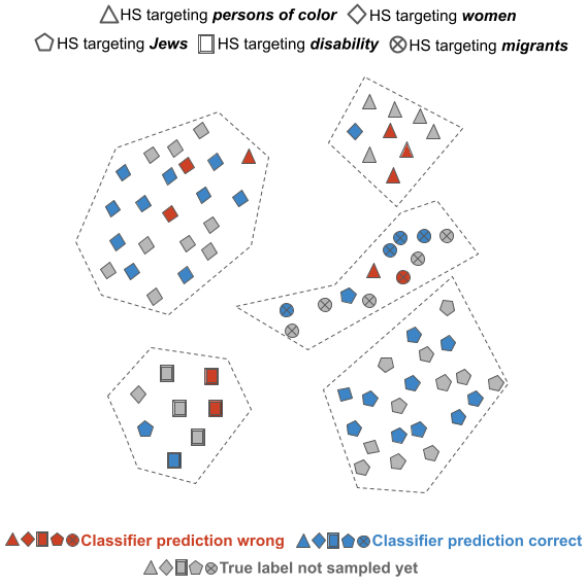

Figure 1: Example scenario: classifiers may not perform well for underrepresented groups in the data. Here, the classifier has a high error rate in detecting hatespeech (HS) against persons of color. Thus, annotation effort should be focused on regions (upper-right) more likely to contain hatespeech against persons of color.

Although active learning (Settles, 2009) is regarded as an efficient method for training models, generic active learning methods can induce bias (Krishnan et al., 2021) rather than mitigate it. Al-

Findings of the Association for Computational Linguistics: ACL 2023, pages 5540–5553
July 9-14, 2023

though there have been numerous works aimed at mitigating bias of active learning methods by the machine learning community (Farquhar et al., 2021; Gudovskiy et al., 2020), these approaches often necessitate an in-depth comprehension of machine learning and active learning theories. We hypothesize that infusing clustering with active learning will allow us to overcome bias issues of both generic active learning and traditional annotation approaches while leveraging the annotation efficiency of active learning.

To this end, we propose a novel dynamic clustering-based algorithm that can substantially improve performance and mitigate bias —**D-CALM** (**D**ynamic **C**lustering-based **A**ctive **L**earning for **M**itigating Bias)[1]. **D-CALM** leverages the distance between a classifier's predictions and true labels in dynamically-adjusted subregions within the data. As opposed to existing active learning methods (Bodó et al., 2011; Berardo et al., 2015) that utilize static clustering of data, our proposed algorithm adapts the clustering in each iteration of active learning. As the classifier gets updated in each iteration, the classifier's error rate changes in different regions. By calibrating the boundaries of clusters iteratively, **D-CALM** focuses annotation effort in updated regions with the evolving classifier's error-rate. As **D-CALM** dynamically adapts its regions for obtaining samples, we hypothesize that our approach will result in reduced bias. Similar to Hassan et al. (2018), we expect bias reduction to be reflected in improved performance metrics and more balanced label and error distribution. We test our hypothesis across eight datasets, spanning a diverse range of text classification tasks (e.g., fine-grained hatespeech, dialog act, emotion detection) and a case study of fine-grained hatespeech detection. Our algorithm is model agnostic, showing substantial improvement for both pretrained models and lightweight Support Vector Machines. Our experiments also demonstrate robustness of **D-CALM** with respect to different measures of information gain.

2 Related Work

Active learning is a well-studied problem in machine learning (Settles, 2009) with numerous scenarios and query strategies (Section 3). Although active learning has shown promise in many tasks, susceptibility to bias, particularly for neural networks, is a concern raised by several works (Yuan et al., 2020). There are existing works that aim to mitigate this bias. Farquhar et al. (2021) proposes using corrective weights to mitigate bias. Gudovskiy et al. (2020) propose self-supervised Fischer-Kernel for active learning on biased datasets. These approaches, however, often require a deep understanding of active learning and neural networks. Our approach is tailored for the NLP community and can easily be deployed.

In recent years, there has been a renewed interest in active learning within the NLP community (Zhang et al., 2022). Some recent works have applied active learning with BERT models for specific tasks such as intent classification (Zhang and Zhang, 2019), sentence matching (Bai et al., 2020), parts-of-speech tagging (Chaudhary et al., 2021) or named entity recognition (Liu et al., 2022). Margatina et al. (2022) propose continued pretraining on unlabeled data for active learning. Rotman and Reichart (2022) adapt active learning to multi-task scenarios for transformer models. Ein-Dor et al. (2020) perform a large-scale empirical study of existing active learning strategies on binary classification tasks. In comparison, we target a diverse range of binary and multi-class classification tasks.

Some other works in the NLP domain have adapted advanced active learning approaches. Yuan et al. (2020) adapt the BADGE (Ash et al., 2020) framework for active learning with BERT. While BADGE computes gradient embedding using the output layer of a neural network and then clusters the gradient space, Yuan et al. (2020) computes surprisal embeddings by using Masked Language Model loss. Margatina et al. (2021) use acquisition functions to obtain contrastive samples for BERT. Our algorithm is comparatively straightforward, not requiring in-depth understanding of surprisal embeddings or acquisition functions. Our algorithm is also model-agnostic and can be applied to neural networks such as BERT, and traditional models such as SVMs. In addition, our clustering step relies on feature representation independent from the learner's representation which may induce bias during the learning process. While some of the aforementioned works (Ein-Dor et al., 2020; Yuan et al., 2020; Margatina et al., 2021) compute diversity in selected samples, our work is the first to analyze and address bias in active learning from a socio-cultural perspective.

[1]Our code is available at: https://github.com/sabithsn/DCALM

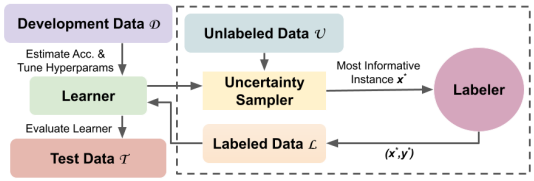

Figure 2: Active learning framework in a pool-based setting. Most informative sample from an unlabeled pool of data is annotated and added to training data.

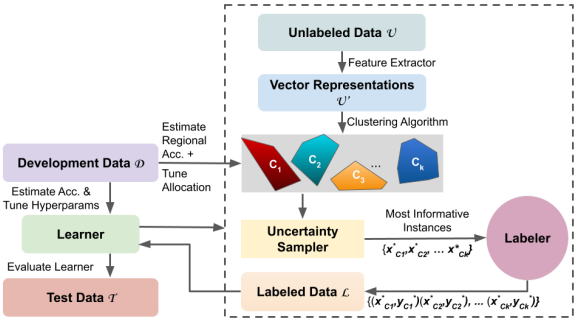

Figure 3: Clustering-based framework. First, unlabeled data is clustered and then most informative samples are chosen from each cluster.

3 Background

This section presents the relevant background of generic active learning followed by a discussion of adapting clustering-based active learning framework for text classification. Within the scope of this paper, we focus on creating the train data. We assume that the dev and test data are already created. Literature of active *testing* (Kumar and Raj, 2018; Hassan et al., 2018) can be referred to for efficiently creating the dev and test set.

3.1 Active Learning Framework

Due to the expanse of active learning literature, it is important to define the generic active learning framework within the scope of this paper. To do so, we need to define the *labeling scenario* and *query-strategy*.

3.1.1 Labeling Scenario

In our work, we assume there is a large pool of unlabeled dataset U but only a small set of labeled dataset L that can be obtained. L is iteratively constructed by querying label for the *most-informative* instance. We focus on *pool-based* active learning because of its relevance to many recent NLP tasks (e.g., hatespeech detetion), for which, a large amount of unlabeled data is scraped from the web and then a subset of it is manually annotated.

3.1.2 Query-Strategy

Many types of query-strategies have been proposed for active learning over the years, including, but not limited to: uncertainty sampling (Lewis and Gale, 1994), expected model change (Settles et al., 2007), expected error reduction (Roy and McCallum, 2001), and variance reduction (Hoi et al., 2006). In our work, we focus on uncertainty sampling because of its popularity and synergy with pool-based sampling (Settles, 2009). Settles (2009) lists three measures of uncertainty to identify most informative sample:

Least Confident: Query the instance whose prediction is the least confident.

$$x^*_{LC} = \underset{x}{argmax}\ 1 - P_\theta(\hat{y}|x) \quad (1)$$

In Eq. 1, $\hat{y} = argmax_y P_\theta(y|x)$, or the class label with the highest probability.

Smallest Margin: Query the sample with minimum difference between two most likely classes:

$$x^*_{MS} = \underset{x}{argmin}\ P_\theta(\hat{y}_1|x) - P_\theta(\hat{y}_2|x) \quad (2)$$

Entropy: The most commonly used measure of uncertainty is entropy:

$$x^*_E = \underset{x}{argmax}\ -\sum_i P_\theta(y_i|x) log P_\theta(y_i|x) \quad (3)$$

In Eq. 3, i ranges over all possible labels.

It should be noted that, in binary classification, all the above measures become equivalent. The active learning framework, within the scope of this paper, is summarized in Figure 2

3.1.3 Challenges

Bias Induction: Since the active learning framework relies on the model's uncertainty to choose samples, the framework may never query samples that the model is confident on. The active learning classifier can become *confidently wrong* on certain samples, leading to an accumulation of bias.

Effective Batch Selection: In a real-world setting, it is not feasible to obtain annotations one by one and queries need to be done in batches. The most straightforward approach would be to choose the N most informative samples (Citovsky et al., 2021). The limitations of this approach can be easily seen. Particularly when N is large, it can amplify the bias discussed earlier.

3.2 Clustering-based Framework

To address the challenges outlined earlier, we approach the problem with clustering-based framework for active learning under pool-based uncertainty sampling settings. Within this framework, the first step is to obtain vector representation of the unlabeled data. This can be done using SentenceBERT (Reimers and Gurevych, 2019) or more traditional Doc2Vec (Le and Mikolov, 2014).

The next step is to cluster the data. This can be done using any clustering algorithm such as KMeans. Then, informative samples are chosen from each cluster, and are added to the training data. The classifier is retrained after each round and the process is repeated until the annotation budget runs out. Figure 3 summarizes this framework.

3.3 D-CALM

Within the clustering-based framework of active learning, we propose a novel algorithm, **D-CALM**, that dynamically adjusts clusters in the data based on estimated classifier error rate.

Algorithm 1 D-CALM: **D**ynamic **C**lustering-based **A**ctive **L**earning for **M**itigating Bias

$D, T \leftarrow$ dev data, test data
$U, L \leftarrow$ unlabeled data, labeled data
$G \leftarrow$ bootstrapped classifier
$B \leftarrow$ labeling budget
$N \leftarrow$ annotation batch size
$m \leftarrow$ initial number of clusters
Cluster U into $\{C_1, C_2, ... C_m\}$
Partition D into $\{C'_1, C'_2, ... C'_m\}$
while $B \geq 0$ **do**
  for `i=0,1,...m` **do**
    Estimate accuracy A_i in C'_i
  end for
  for `i=0,1,...m` **do**
    Allocate $l_i = N * \frac{1-A_i}{\sum_j (1-A_j)}$
    Cluster C_i into $\{C_{i_1}, C_{i_2}, ...C_{i_{l_i}}\}$
    for `j=0,1,...`l_i **do**
      $x^*_{ij} \leftarrow$ most infor. sample in C_{ij}
      $y^*_{ij} \leftarrow$ query true label for x^*_{ij}
      Add (x^*_{ij}, y^*_{ij}) to L
    end for
  end for
  $G \leftarrow$ retrain on L
  $B = B - N$
end while
Evaluate G on T

In our proposed algorithm, the cluster C'_i is used to dynamically partition C_i. Our algorithm first observes how the classifier behaves in C'_i. For cluster C_i, it allocates samples proportional to the error rate in C'_i. Then the cluster C_i is split into subclusters according to the number of samples allocated to C_i. Most informative sample from each subcluster is then added to training data. The subclusters are dynamically updated in each iteration to account for the classifier's new state. This prevents the classifier from repeatedly sampling from any particular region. It is worth noting that *error-rate* can be substituted with different metrics to account for specific needs. For example, in scenarios where it is more important to reduce false negative rate reduce compared to false positive rate, the error rate can be substituted with false negative rate. In this paper, we focus on the general case of error-rate.

4 Experiment Setup

In this section, we outline our experimental setup.

4.1 Active Learning Approaches

For all the following approaches, total number of samples range from 100-300, initial allocation for bootstrapping is set to 50, and annotation batch size is 50. Similar to (Ein-Dor et al., 2020), the classifiers are retrained in each round.

Random: The allocated number of samples are picked randomly from the unlabeled pool.

TopN: The classifier is bootstrapped with 50 samples. In each iteration N most informative samples are labeled and added to training data until labeling budget runs out. TopN is a widely used baseline (Yuan et al., 2020; Ash et al., 2020).

Cluster-TopN: The classifier is bootstrapped in the same way. The unlabeled pool is first clustered into 10 clusters and in each iteration, N/10 most informative samples are chosen from each cluster. Cluster-TopN combines TopN and stratified sampling (Qian and Zhou, 2010). We choose Cluster-TopN as a baseline due to its similarity with multiple existing methods (Xu et al., 2003; Zhdanov, 2019).

D-CALM: The classifier is bootstrapped in a similar fashion. While **D-CALM** is not sensitive to the initial number of clusters because of its dynamic splitting into subclusters, we set initial number of clusters to 10 to be consistent with Cluster-TopN.

4.2 Models

Transformers We fine-tune the widely-used bert-based-cased (Devlin et al., 2019). We observed that the models stabilize on the dev data when fine-tuned for 5 epochs with learning rate of 8e-5 and batch size of 16. The same setting is used across all experiments.

Support Vector Machine (SVM) We choose SVM as our alternate model as it is completely different from transformers and because SVMs are still in use for practical purposes due to speed and lightweight properties (Hassan et al., 2021, 2022). We use Tf-IDF weighted character [2-5] grams to train SVMs with default scikit-learn settings[2].

4.3 Datasets

We evaluate our proposed algorithm on eight diverse datasets, among which two are binary classification datasets and the rest are multiclass.

BOOK32 (Iwana et al., 2016) contains 207K book titles categorized into 32 classes such as *Biographies & Memoirs*. We take a subset that contains 20K random samples from 10 most frequent classes for runtime efficiency. Random sampling ensures the subset respects original distribution.

CONAN (Fanton et al., 2021) contains 5K instances annotated for hatespeech targets: *Disabled, Jews, LGBT+, Migrants, Muslims, Person of Color (POC), Women, and Other*.

CARER (Saravia et al., 2018) is an emotion detection dataset that contains six basic emotions in the released version: *Anger*, *Fear*, *Joy*, *Love*, *Sadness*, and *Surprise*. [3]. The released version consists of 16K training, 2K dev and 2K test instances.

CoLA (Saravia et al., 2018) contains 9.5K sentences expertly annotated for acceptability (grammaticality) in the public version. We use the in-domain set as dev and out-of-domain as test set.

HATE (Davidson et al., 2017) contains a total of 24.7K tweets that are annotated as: *Offensive*, *Hatespeech*, and *Neither*.

MRDA (Shriberg et al., 2004) contains 117K instances annotated for dialog acts. We consider the five basic labels: *Statement*, *BackChannel*, *Disruption*, *FloorGrabber*, and *Question*. We limit the data to 20K randomly chosen samples for runtime efficiency.

Q-Type (Li and Roth, 2002) contains 5.5K train and 0.5K test instances annotated for question types. We take the first level of annotation, containing six classes: *Entity, Description, Abbrebivation, Number, Human,* and textitLocation.

Subjectivity (Pang and Lee, 2004) contains 10K snippets from Rotten Tomatoes/IMDB reviews automatically tagged as *Subjective* or *Objective*.

4.4 Data Preparation

Splits We use default train-dev-test splits if they are provided. If they are not provided, we split the data into 70-10-20 splits. The train data is treated as unlabeled pool of data, dev data is used for tuning purposes and test data is used to report results. Table 1 shows summary of data used.

Dataset	classes	Pool	Dev	Test
BOOK32	32	14K	2K	4K
CONAN	8	3.5K	0.5K	1K
CARER	6	16K	2K	4K
CoLA	2	8.5K	0.5K	0.5K
Hatespeech	3	17.2K	2.4K	4.9K
MRDA	5	14K	2K	4K
Q-Type	6	4.9K	0.5K	0.5K
Subjectivity	2	7K	1K	2K

Table 1: Statistics of used datasets

Vector Representation We use MiniLM (Wang et al., 2020) sentence-transformer to transform text instances into 384 dimensional vectors. These vectors are then used to cluster the unlabeled data.

Clustering We use KMeans to cluster the unlabeled pool of data. We use scikit-learn[4] implementation of KMeans with default parameters.

5 Results and Case Study

We first discuss findings of our experiments, followed by a case study of fine-grained hatespeech detection. Figures 4,5 and 6 summarize the results across the eight datasets, different measures of information gain, and different models. Table 2 summarizes relative performance across all experiments. For each experiment, we report Macro-F1

[2] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

[3] https://huggingface.co/datasets/emotion

[4] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

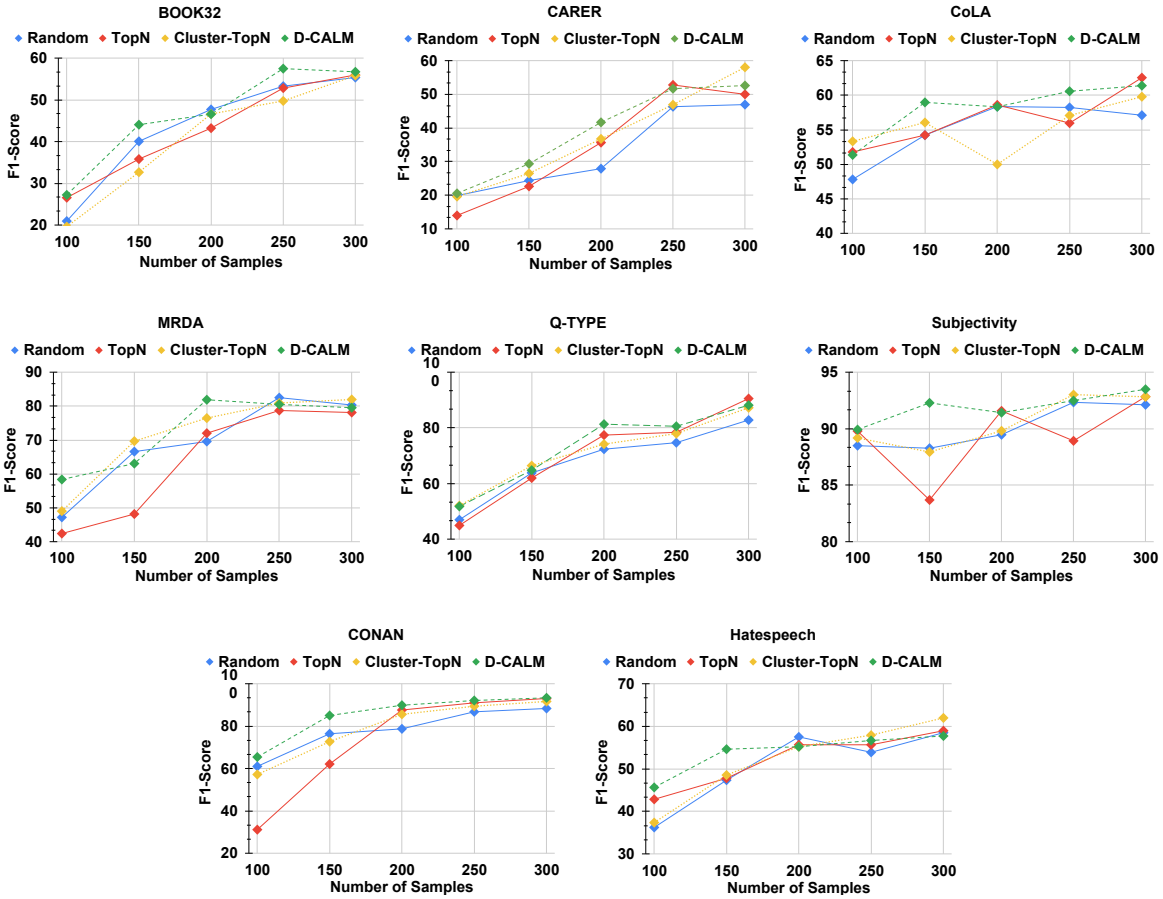

Figure 4: Comparison of our proposed algorithm (**D-CALM**) and baseline approaches. **D-CALM** (green-dashed-line) consistently outperforms baseline approaches across eight datasets with Entropy as the measure of information gain and BERT as learner model.

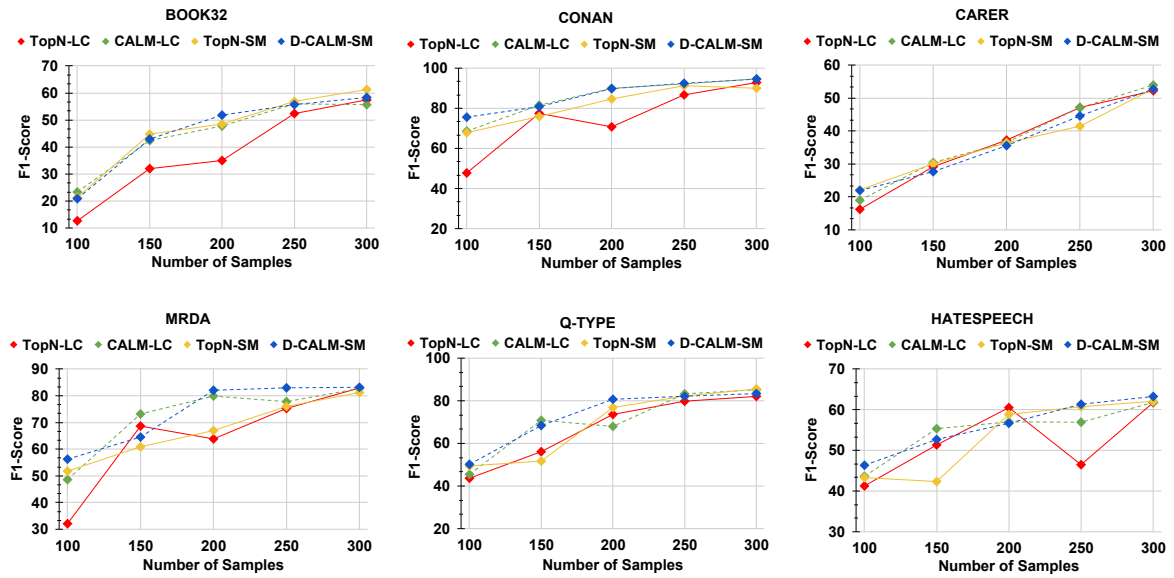

Figure 5: Comparison of our algorithm with TopN approach under different measures of information gain. LC refers to Least Confident and SM refers to Smallest Margin. A consistent improvement over TopN baseline, similar to Entropy-based information gain in Figure 4 affirms the robustness of our algorithm

.

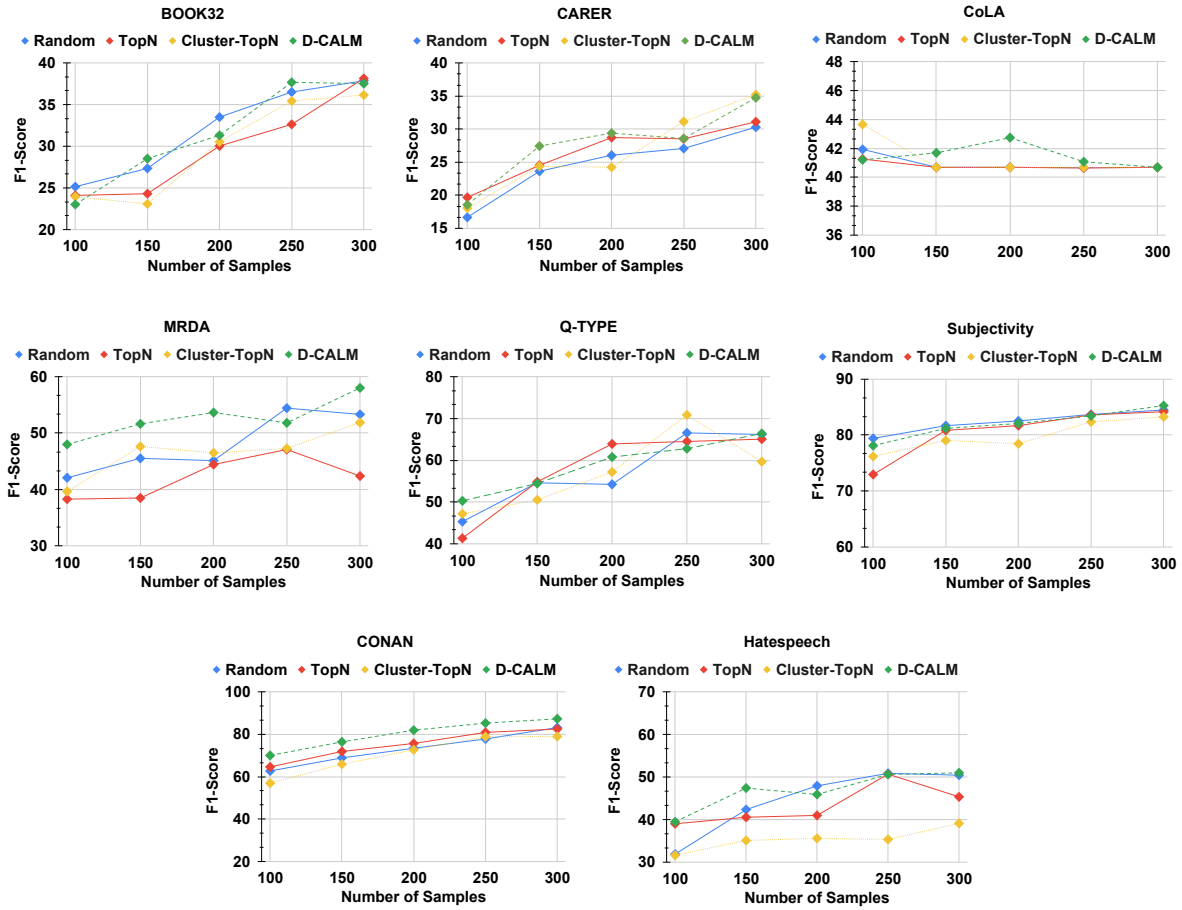

Figure 6: Comparison of our proposed algorithm (**D-CALM**) and baseline approaches for SVM as learner model. Consistent improvement for SVMs in addition to BERT models (Figure 4) suggests **D-CALM** can be used for completely different types of models.

score averaged across 3 runs. We choose Macro-F1 as our metric since it provides a more holistic measure of a classifier's performance across classes. Thus, reduction of bias is more likely to be reflected in metrics such as F1 compared to other metrics such as accuracy.

5.1 Experiment Results

D-CALM consistently outperforms baselines: From Figures 4, 5, 6 we can observe that **D-CALM** consistently outperforms TopN, random and cluster-TopN across all datasets. From Table 2, we observe that **D-CALM** beats TopN in 32 out of 40 data points for BERT, among which, the difference in F1 score is greater than 5 in 15 cases. D-CALM beats the nearest algorithm, Cluster-TopN in 28/40 (p value 0.003) for BERT and Random Sampling in 26/40 cases (p value 0.0073) for SVMs (Table 2). Both of these are statistically significant according to 2 population proportion test at significance level of 0.01.

Diff.	**Count for BERT (IG=Entropy)**		
(F1)	**DL > RND**	**DL > TN**	**DL > CTN**
> 0	33/40	32/40	28/40
> 1	30/40	27/40	23/40
> 3	24/40	18/40	16/40
> 5	15/40	12/40	11/40
> 10	4/40	4/40	2/40
(F1)	**DL > RND**	**DL > TN**	**DL > CTN**
> 0	26/40	30/40	34/40
> 1	21/40	23/40	30/40
> 3	16/40	18/40	23/40
> 5	10/40	11/40	16/40
> 10	0/40	2/40	6/40

Table 2: Aggregated counts of **D-CALM** (DL) outperforming Random (RND), TopN (TN) and Cluster-TopN (CTN) across 8 datasets (5*8=40 data points). Diff denotes the difference of F1 score between DL and the contesting method. E.g.: diff >10 indicates the count of DL outperforming contesting methods by a difference of 10 F1 score or more.

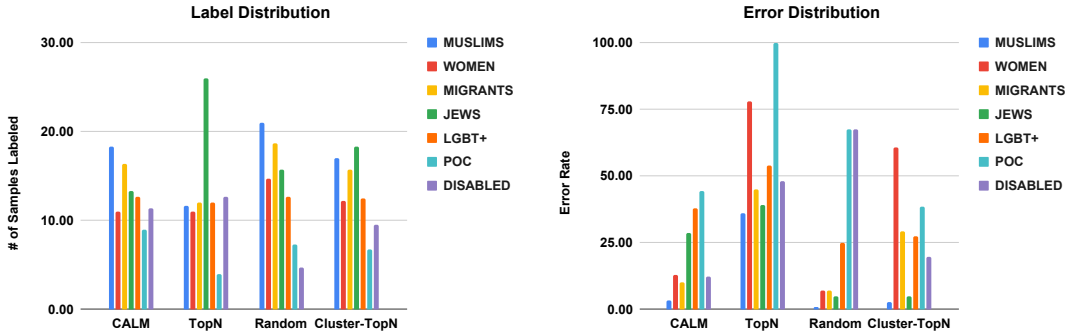

Figure 7: Label and error distribution after one iteration of Active Learning on the CONAN (fine-grained hatespeech) dataset (averaged across 3 runs). **D-CALM** doesn't suffer from strong bias toward particular groups as TopN does and is more equitable toward underrepresented groups compared to random sampling.

.

D-CALM is more robust against critical failures: We observe from Figure 4 that on occasions such as in the case of Subjectivity and MRDA, TopN can have critical failures where the model ends up with an extremely low F1 score. Although on a few occasions, we witness dips in the curves of **D-CALM**, in general, the curves are much more stable, indicating its robustness.

D-CALM is robust across different measures of information gain: From Figure 5, we see that **D-CALM** outperforms random, TopN, and cluster-TopN for different measure of information gain. Figure 5 does not contain the Subjectivity and CoLA because these are binary datasets and Entropy (reported in Figure 4), Least Confident and Smallest Margin become equivalent in the case of binary classification (Section 3).

D-CALM is model-agnostic: From Figure 6, we observe similar patterns in improvement when the learner model is SVM instead of BERT. Although the degree of improvement is smaller for SVMs compared to BERT, it is a limitation of active learning rather than **D-CALM's**, as we can see others showing smaller improvement as well. Improvement over the baselines for SVMs in addition to BERT suggests **D-CALM** is model-independent.

D-CALM is more robust against bias: Since D-CALM's focus on enforcing diversity in dynamically adjusting clusters separates it from the other methods, we can deduce that it is the bias reduction that is resulting in improved performance metrics. This is further supported by a study of label and error distribution in the following section.

5.2 Case Study

We choose hatespeech detection as our case study because hatespeech and abusive content detection has been one of the most studied text classification tasks in recent years (Zampieri et al., 2019; Hassan et al., 2020). Due to its importance in many avenues, whether for AI moderation of online content (Ye et al., 2023) or filtering language model data (Abdelali et al., 2021), racial and gender bias in hatespeech and abusive content detection tasks have been a particular concern for the NLP community (Davidson et al., 2019; Ahn and Oh, 2021).

5.2.1 Label and Error Distribution

To understand the effect of the underlying distribution in data, we observe the label distribution of samples chosen for annotation after one round of active learning. We also observe the distribution when the same number of samples are chosen randomly. From Figure 7 we can observe that random sampling may result in small number of samples chosen for minority classes in the data such as *Persons of Color* and *Disabled*. The distribution of labels obtained by TopN can become particularly skewed. We see that on average, TopN samples 25+ hatespeech targeting *Jews* while mostly ignoring groups such as *Person of Color (POC)* with <5 samples on average. While Cluster-TopN mitigates this problem to an extent, the best results are obtained by **D-CALM**, with samples for *POC* doubling compared to TopN and samples for *Disabled* doubling compared to random sampling after just one iteration. The error distribution in Figure 7, reflects the effects of this as we see errors for *POC*

Text	Annotated	Predicted	Error Type
Jews are everywhere, but they are too fundamentalist.	*Jews*	*Muslims*	Related Class
Asians just want to talk with other asians: they are the racists...	*Other*	*POC*	Dataset Limitation
Migrants are just a bunch of animals, we can't accept them...	*Other*	*MIGRANT*	Annotation Error

Table 3: Examples of errors made on the CONAN dataset by our proposed approach. While our approach improves performance and mitigates bias as seen before, a better task design and annotation quality control are required to address these types of errors. *Disclaimer: these examples do not reflect authors' views in any way.*

and *Disabled* are greatly reduced by **D-CALM**. It's important to note, if we had access to a large pool of labeled data, we could obtain a more balanced dataset for training. However, in a real-world scenario, before the annotation process, we only have access to unlabeled pool of data. As such, we cannot identify low-frequency classes and balance the training set. **D-CALM**, however, can obtain more samples from the underrepresented classes without knowing their true labels beforehand.

5.2.2 Error Analysis

To understand the limitations of **D-CALM**, we manually annotated 100 errors made by the best run with BERT on the CONAN dataset after one iteration of active learning. Our key observations are listed below:

- The model can be confused on closely related classes such as *Jews* and *Muslims* as the hatespeech in both cases target religions.
- Some errors can be attributed to the limitation of annotation design. For example, CONAN contains the class Persons of Color (POC), but does not contain a separate class for racism against Asians. These instances are labeled as *Other* in the data but are predicted as *POC* by the model.
- In some cases, the error is in the original annotation, rather than the model's prediction.

Examples of these errors are listed in Table 3. While the first type of error can possibly be reduced with the addition of more data close to boundary regions between closely related classes, the last two types of errors need to be addressed during the design and annotation phase of the task.

6 Conclusion and Future Work

In this paper, we presented a novel dynamic clustering-based active learning algorithm, **D-CALM**, that can be easily adopted by the NLP community for training models with a small set of annotated data. We have shown that by focusing annotation efforts in adaptive clusters where the learner model has higher error rates, the performance can be improved substantially while reducing bias against underrepresented groups in unlabeled data. Our experiments also show that **D-CALM** is robust across different datasets, different measures of information gain, and completely different model types. In the future, our approach can be adapted for creating less biased test sets for evaluating classifiers. An exciting future direction for our approach is to adapt it for natural language generation tasks such as style-transfer (Atwell et al., 2022) or counterspeech generation (Ashida and Komachi, 2022).

Limitations

It's important to note that, in this paper, we focus on bias resulting from underlying distribution of training data. Bias that may result from pretraining of transformer models (Li et al., 2021) is not within the scope of this paper.

Although we conduct a case study of fine-grained hatespeech detection task, a collective effort from the research community is required to better quantify bias mitigation of our approach across multiple tasks and different types of bias.

Another limitation of our work is that our proposed algorithm requires dynamic adjustment of clusters. For very large datasets, this may be computationally expensive.

Ethics Statement

Although our proposed algorithm shows more stability and reduced bias compared to existing approaches and random sampling, it's important to observe the behavior of active learner as the algorithm may not completely eliminate bias, specifically when the annotation budget is small. This can be achieved by observing label and error variance on the evaluation data. It is also important to take into consideration the necessities of practical scenarios. In scenarios where certain type of bias is desired (e.g., higher precision), the algorithm needs to be adapted as outlined in Section 3.3

References

Ahmed Abdelali, Sabit Hassan, Hamdy Mubarak, Kareem Darwish, and Younes Samih. 2021. Pre-training BERT on arabic tweets: Practical considerations. *CoRR*, abs/2102.10684.

Jaimeen Ahn and Alice Oh. 2021. Mitigating language-dependent ethnic bias in BERT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 533–549, Online and Punta Cana, Dominican Republic. Association for 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. *ArXiv*, abs/1906.03671.

Mana Ashida and Mamoru Komachi. 2022. Towards automatic generation of messages countering online hate speech and microaggressions. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 11–23, Seattle, Washington (Hybrid). Association for Computational Linguistics.

Katherine Atwell, Sabit Hassan, and Malihe Alikhani. 2022. APPDIA: A discourse-aware transformer-based style transfer model for offensive social media conversations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6063–6074, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Guirong Bai, Shizhu He, Kang Liu, Jun Zhao, and Zaiqing Nie. 2020. Pre-trained language model based active learning for sentence matching. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1495–1504, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Saul Berardo, Eloi L. Favero, and Nelson Cruz Sampaio Neto. 2015. Active learning with clustering and unsupervised feature learning. In *Canadian Conference on AI*.

Zalán Bodó, Zsolt Minier, and L. Csató. 2011. Active learning with clustering. In *Active Learning and Experimental Design @ AISTATS*.

Aditi Chaudhary, Antonios Anastasopoulos, Zaid Sheikh, and Graham Neubig. 2021. Reducing confusion in active learning for part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 9:1–16.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. 2021. Batch active learning at scale. In *NeurIPS*.

Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 25–35, Florence, Italy. Association for Computational Linguistics.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805.

Liat Ein-Dor, Alon Halfon, Ariel Gera, Eyal Shnarch, Lena Dankin, Leshem Choshen, Marina Danilevsky, Ranit Aharonov, Yoav Katz, and Noam Slonim. 2020. Active Learning for BERT: An Empirical Study. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7949–7962, Online. Association for Computational Linguistics.

Margherita Fanton, Helena Bonaldi, Serra Sinem Tekiroğlu, and Marco Guerini. 2021. Human-in-the-loop for data collection: a multi-target counter narrative dataset to fight online hate speech. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3226–3240, Online. Association for Computational Linguistics.

Sebastian Farquhar, Yarin Gal, and Tom Rainforth. 2021. On statistical bias in active learning: How and when to fix it. *ArXiv*, abs/2101.11665.

Denis A. Gudovskiy, Alec Hodgkinson, Takuya Yamaguchi, and Sotaro Tsukizawa. 2020. Deep active learning for biased datasets via fisher kernel self-supervision. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9038–9046.

Sabit Hassan, Hamdy Mubarak, Ahmed Abdelali, and Kareem Darwish. 2021. ASAD: Arabic social media analytics and unDerstanding. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 113–118, Online. Association for Computational Linguistics.

Sabit Hassan, Younes Samih, Hamdy Mubarak, and Ahmed Abdelali. 2020. ALT at SemEval-2020 task 12: Arabic and English offensive language identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1891–1897, Barcelona (online). International Committee for Computational Linguistics.

Sabit Hassan, Shaden Shaar, and Kareem Darwish. 2022. Cross-lingual emotion detection. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6948–6958, Marseille, France. European Language Resources Association.

Sabit Hassan, Shaden Shaar, Bhiksha Raj, and Saquib Razak. 2018. Interactive evaluation of classifiers under limited resources. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 173–180.

Steven C. H. Hoi, Rong Jin, and Michael R. Lyu. 2006. Large-scale text categorization by batch mode active learning. In *WWW '06*.

Brian Kenji Iwana, Syed Tahseen Raza Rizvi, Sheraz Ahmed, Andreas Dengel, and Seiichi Uchida. 2016. Judging a book by its cover. *arXiv preprint arXiv:1610.09204*.

Svetlana Kiritchenko and Saif Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 43–53, New Orleans, Louisiana. Association for Computational Linguistics.

Ranganath Krishnan, Alok Sinha, Nilesh A. Ahuja, Mahesh Subedar, Omesh Tickoo, and Ravi R. Iyer. 2021. Mitigating sampling bias and improving robustness in active learning. *ArXiv*, abs/2109.06321.

Anurag Kumar and Bhiksha Raj. 2018. Classifier risk estimation under limited labeling resources. *ArXiv*, abs/1607.02665.

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *ICML*.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *SIGIR '94*.

Luoqiu Li, Xiang Chen, Hongbin Ye, Zhen Bi, Shumin Deng, Ningyu Zhang, and Huajun Chen. 2021. On robustness and bias analysis of bert-based relation extraction. In *CCKS*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING*.

Mingyi Liu, Zhiying Tu, Tong Zhang, Tonghua Su, Xiaofei Xu, and Zhongjie Wang. 2022. Ltp: A new active learning strategy for crf-based named entity recognition. *Neural Processing Letters*, 54:2433–2454.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2020. Gender bias in neural natural language processing. In *Logic, Language, and Security*.

Katerina Margatina, Loic Barrault, and Nikolaos Aletras. 2022. On the importance of effectively adapting pretrained language models for active learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 825–836, Dublin, Ireland. Association for Computational Linguistics.

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.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the ACL*.

Longhua Qian and Guodong Zhou. 2010. Clustering-based stratified seed sampling for semi-supervised relation classification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 346–355, Cambridge, MA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Guy Rotman and Roi Reichart. 2022. Multi-task active learning for pre-trained transformer-based models. *Transactions of the Association for Computational Linguistics*, 10:1209–1228.

Nicholas Roy and Andrew McCallum. 2001. Toward optimal active learning through sampling estimation of error reduction. In *ICML*.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Burr Settles. 2009. Active learning literature survey.

Burr Settles, Mark W. Craven, and Soumya Ray. 2007. Multiple-instance active learning. In *NIPS*.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The ICSI meeting recorder dialog act (MRDA) corpus. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*, pages 97–100, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.

Zhao Xu, Kai Yu, Volker Tresp, Xiaowei Xu, and Jizhi Wang. 2003. Representative sampling for text classification using support vector machines. In *European Conference on Information Retrieval*.

Meng Ye, Karan Sikka, Katherine Atwell, Sabit Hassan, Ajay Divakaran, and Malihe Alikhani. 2023. Multilingual content moderation: A case study on Reddit. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3828–3844, Dubrovnik, Croatia. Association for Computational Linguistics.

Michelle Yuan, Hsuan-Tien Lin, and Jordan Boyd-Graber. 2020. Cold-start active learning through self-supervised language modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7935–7948, Online. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. In *NAACL*.

Leihan Zhang and Le Zhang. 2019. An ensemble deep active learning method for intent classification. In *Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence*, CSAI2019, page 107–111, New York, NY, USA. Association for Computing Machinery.

Zhisong Zhang, Emma Strubell, and Eduard Hovy. 2022. A survey of active learning for natural language processing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6166–6190, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Fedor Zhdanov. 2019. Diverse mini-batch active learning.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations sections at the end

☑ A2. Did you discuss any potential risks of your work?
Ethical considerations sections at the end

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Sections 3-5

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4,5

☑ B1. Did you cite the creators of artifacts you used?
Section 4,5

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
The models (e.g. BERT) are free-to-use for researchers.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

Section 4,5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
4.2, 4.3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5.1

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.