Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems | School of Computing and Information Systems

9-2023

Real: A representative error-driven approach for active learning

Cheng CHEN

Yong WANG
Singapore Management University, yongwang@smu.edu.sg

Lizi LIAO
Singapore Management University, lzliao@smu.edu.sg

Yueguo CHEN

Xiaoyong DU

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Databases and Information Systems Commons

Citation

CHEN, Cheng; WANG, Yong; LIAO, Lizi; CHEN, Yueguo; and DU, Xiaoyong. Real: A representative error-driven approach for active learning. (2023). *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Turin, Italy, 2023 September 18-22*. 20-37.

Available at: https://ink.library.smu.edu.sg/sis_research/8586

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

arXiv:2307.00968v2 [cs.LG] 6 Jul 2023

REAL: A Representative Error-Driven Approach for Active Learning

Cheng Chen[1] Yong Wang[2](✉) Lizi Liao[2] Yueguo Chen[1] Xiaoyong Du[1]
[1]Renmin University of China [2]Singapore Management University
{chchen, chenyueguo, duyong}@ruc.edu.cn
{yongwang, lzliao}@smu.edu.sg

Abstract. Given a limited labeling budget, active learning (AL) aims to sample the most informative instances from an unlabeled pool to acquire labels for subsequent model training. To achieve this, AL typically measures the informativeness of unlabeled instances based on uncertainty and diversity. However, it does not consider erroneous instances with their neighborhood error density, which have great potential to improve the model performance. To address this limitation, we propose REAL, a novel approach to select data instances with Representative Errors for Active Learning. It identifies minority predictions as *pseudo errors* within a cluster and allocates an adaptive sampling budget for the cluster based on estimated error density. Extensive experiments on five text classification datasets demonstrate that REAL consistently outperforms all best-performing baselines regarding accuracy and F1-macro scores across a wide range of hyperparameter settings. Our analysis also shows that REAL selects the most representative pseudo errors that match the distribution of ground-truth errors along the decision boundary. Our code is publicly available at https://github.com/withchencheng/ECML_PKDD_23_Real.

Keywords: Active learning · Text classification · Error-driven

1 Introduction

Labeling data for machine learning is costly, and the budget on the amount of labels we can gather is often limited. Therefore, it is crucial to make the training process of machine learning models more label-efficient, especially for applications where labels are expensive to acquire. Active learning (AL) is to select a small amount of the most informative instances from an unlabeled pool, aiming to maximize the model performance gain when using the selected instances (labeled) for further training. Identification of the most informative instances from the unlabeled data pool is critical to the success of AL.

The AL techniques can be classified into three groups: uncertainty-based, diversity-based, and hybrid methods. Uncertainty-based methods select instances whose prediction probability is more evenly distributed over classes [28,27], instances with a larger expected loss/gradient [44,3], or those closer to decision

(✉)The corresponding author.

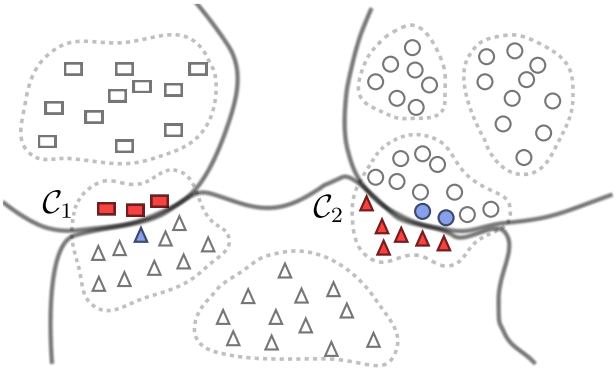


Fig. 1: An illustrative example of REAL. The solid thick lines denote the model decision boundaries, separating data points into three predicted classes (rectangles, circles, and triangles). The dashed irregular circles are clusters of data points. REAL samples the minority predictions as *pseudo errors* (▭ △) in cluster $\mathcal{C}_1$ and $\mathcal{C}_2$ for labeling. If our budget exceeds the number of pseudo errors, REAL will pick blue instances (△ ○) where the model has the least confidence in its predictions.

boundaries [37,13]. However, solely relying on instance-level uncertainty metrics may cause redundancy in samples [45]. Hence, diversity-based methods try to mitigate the redundancy problem by selecting a small but diverse set of data instances to represent *the whole unlabeled pool* [23]. However, they ignore the fact that training on errors is more label-efficient [7]. Hybrid methods try to select instances that are both uncertain and diverse[46,45]. Our proposed method belongs to the hybrid category. Our novelty is to seek representativeness for the *errors* rather than the *whole unlabeled pool*, by selecting instances with a larger error probability and higher neighborhood error density. Fig. 1 shows an illustrative example of our method. Specially, we first cluster the unlabeled instances by their representations. The majority prediction in a cluster is expected to be correct even with limited labeled training data [1,39,6], owing to the strong representation power of the pretrained models for images [18] or texts [31]. Also, it is common for AL to achieve a decent test accuracy after the warm-up training on the initial limited labeled data [33,45]. Consequently, we treat the majority prediction in a cluster as the *pseudo label* for all the instances in the cluster. We call instances in a cluster whose predictions disagree with the cluster *pseudo label* as *pseudo errors*. More pseudo errors with a lower prediction probability (larger disagreement) on the pseudo label will bring a larger sampling budget to its affiliated cluster. In this way, we emphasize dense areas of errors, and thus adaptively select more representative errors.

To our best knowledge, REAL is the first approach to sample representative errors to achieve label-efficient active learning. By taking text classification as an example application, we demonstrate the effectiveness of REAL. In summary, the major contributions of this paper can be summarized as follows:

- We propose a new AL sampling algorithm, REAL, that explores selecting representative errors from the unlabeled pool.
- We show REAL consistently beats all the best-performing baselines on five text classification benchmark datasets in terms of both accuracy and F1-macro scores.
- We empirically investigate error distribution and find that 1) most errors are distributed along the decision boundary; 2) the distribution of selections made by REAL align well with that of ground-truth errors.

2 Preliminaries

2.1 Related Work

Uncertainty-based Uncertainty-based sampling is to sample the most uncertain instances for model training. Three classical metrics for the uncertainty of model prediction probabilities are: entropy [28,27], least confidence [27,29], and smallest margin [36]. Recent research studies take the expected loss [44], expected generalization error reduction [24], or distance to the decision boundary [37] as surrogates for uncertainty. CAL [33] selects contrastive examples that are similar in the feature space of pre-trained language model (PLM) and maximally different in the output probabilities. Unlike CAL which ignores the correctness of sampled instances, our method aims to mine the yet-to-be errors. OPAL [25] computes the expected misclassification loss reduction, but is limited to binary classification using the outdated Parzen window classifier.

Diversity-based Diversity-based sampling aims to maximize the diversity of sampled instances. Cluster-Margin [9] selects a diverse of instances with the smallest margin using hierarchical agglomerate clustering. Sener and Savarese [38] proposed a coreset approach to find a representative subset from the unlabeled pool. Kim *et al.* [23] assessed the density of unlabeled pool and selected diverse samples mainly from regions of low density. Meanwhile, generative adversarial learning [17] is applied in AL as a binary classification task. They trained an adversarial classifier to confuse data from the training set and that from the pool. However, our method cares about the density of errors rather than the whole unlabeled pool.

Hybrid Hybrid AL methods try to combine uncertainty and diversity sampling. Such a combination can be achieved by meta learning [5,19] and reinforcement learning [14,30], which automatically learn a sampling strategy in each AL round instead of using a fixed heuristic. BADGE [3] and ALPS [46] both compute uncertainty representations of instances and then cluster them. BADGE transforms data into gradient embeddings that encode the model confidence and then apply K-Means++ [2]. ALPS first utilizes the self-supervision loss of PLM as uncertainty representation. ACTUNE [45] uses weighted K-Means clustering to find highly uncertain regions. The K-Means clustering in ACTUNE is weighted by some uncertainty measure, e.g., entropy or CAL [33]. ACTUNE deliberately tries to separate the uncertain regions from the confident regions via weighted clustering, with an implicit assumption that the two kinds of regions are separable.

2.2 Problem Definition

We take text classification as an example to illustrate the core idea of our approach. Given a small labeled set $\mathcal{D}_l = \{(\mathrm{x}_i, y_i)\}_{i=1}^{L}$ (warm-up dataset) for initial model training and a large unlabeled data pool $\mathcal{D}_u = \{(\mathrm{x}_i,)\}_{i=1}^{U}$, where x_i is the $i-$th input instance (e.g., the token sequence for text classification), $y_i \in \{1, \ldots, Y\}$ is the target label, and $\mathcal{D}_l \ll \mathcal{D}_u$, we want to select and obtain the labels of the most informative instances in $\mathcal{D}_u$ for training model $\mathcal{M}$, so that the performance of $\mathcal{M}$ can be maximized given a fixed labeling budget B [1]. $\mathcal{M}$ is trained iteratively. Suppose there are T AL rounds in total, then the budget for each round is $b = B/T$. In each AL round, a sampling function $\alpha(\mathcal{D}_u, \mathcal{M})$ selects b samples from $\mathcal{D}_u$ based on the previously learned model $\mathcal{M}$, and then moves the labeled b samples into $\mathcal{D}_l$. Model $\mathcal{M}$ is trained on the updated $\mathcal{D}_l$ and then evaluated on a hold-out test set. The AL process terminates when the total budget B is exhausted or the model performance is good enough. The core of an AL method is to study the sampling function α.

3 Representative Pseudo Errors

3.1 Overview of Representative Error-Driven Active Learning

We aim to explore one critical research question for active learning: how will *learning from errors* improve the active learning accuracy for models? Intuitively, sampling more errors for model training will prevent the model from making the same mistakes on the test set, thus improving the test accuracy. Errors bring larger loss values, making them more informative for model training [44]. Though one existing work [7] tries to directly calculate the erroneous probability for some image using the Bayesian theorem, it ignores the density of errors. Other than computing a single unlabeled instance's erroneous probability, we develop a sense of representativeness for the selected errors into our approach.

The AL process starts from training the model $\mathcal{M}$ on the initial labeled data set $\mathcal{D}_l^{(0)}$. Formally, we minimize the average cross entropy loss ℓ for all the instances in $\mathcal{D}_l^{(0)}$:

$$\min_{\theta} \frac{1}{|\mathcal{D}_l^{(0)}|} \sum_{(\mathrm{x}_i, y_i) \in \mathcal{D}_l} \ell(\mathcal{M}(\mathrm{x}_i, \theta^{(0)}), y_i). \tag{1}$$

In each of the following AL rounds, REAL (Algorithm 1) selects a set of representative errors Q consisting of b instances from $\mathcal{D}_u$, obtains their labels, and then adds Q into $\mathcal{D}_l$ for subsequent model training. Algorithm 1 consists of two components: *pseudo error identification* and *adaptive sampling of representative errors*.

[1] Following the convention in machine learning community [28,24,30,9], we ignore the cognitive difference for labeling different instances studied in the HCI community[8,34], and assume the labeling cost is 1 for every instance. For example, if our total labeling budget is $B = 800$ and we have $T = 8$ rounds of AL, then $b = 100$ is the budget per round.

3.2 Pseudo Error Identification

The first challenge is to select instances from $\mathcal{D}_u$ where the model $\mathcal{M}$ makes mistakes, which is non-trivial since we do not have access to the ground truth labels before selection. However, prior studies [1,39,49] have shown that PLM can effectively learn the sentence representations and support accurate classifications very well by simply clustering the embedded representations of sentences [39]. Also, it is commonly-seen that active learning will be employed after the machine learning models have achieved a reasonable performance [33,45]. Building upon these facts, it is safe to expect the majority prediction in a cluster by the PLM has a high probability to be the ground truth label, even with a small amount of training data. Thus, we assume that the majority prediction is the correct label for all the instances in the cluster. Our preliminary experiments also show a relatively high and stable accuracy of our pseudo-label assignment strategy, i.e., over 0.80 for all the chosen datasets. Since the majority prediction is treated as the pseudo label to each cluster, pseudo errors are defined as those instances whose predictions disagree with the majority prediction in each cluster. As will be shown in Section 5, the sampled pseudo errors usually have higher error rates when compared with ground truths, indicating that such a way of defining pseudo labels and pseudo errors is effective.

In round t ($1 \leq t \leq T$) of active learning, we first obtain the representations of instances in $\mathcal{D}_u$ by feeding them into model $\mathcal{M}$'s encoder $\Phi(.)$. Specifically, we only take the `[CLS]` token embedding from the output in the last layer of encoder $\Phi(.)$. Then K-Means++ [2] is employed as an initialization of the seeding scheme for the following clustering process. We denote the k-th cluster as $\mathcal{C}_k^{(t)} = \{\mathrm{x}_i | c_i^{(t)} = k\}, k \in \{1, \ldots, K\}$, where $c_i^{(t)}$ is the cluster id for the instance x_i at AL round t. After obtaining K clusters with the corresponding data $\mathcal{C}_k^{(t)}$, we assign a pseudo label for each cluster. First, the pseudo label for an individual instance x_i at round t of AL is computed as:

$$\widetilde{y}_i = \underset{j \in \{1,\ldots,Y\}}{\operatorname{argmax}} [\mathcal{M}(\mathrm{x}_i; \theta^{(t)})]_j, \tag{2}$$

where $\mathcal{M}(\mathrm{x}_i; \theta^{(t)}) \in \mathbb{R}^Y$ is the probability distribution for instance x_i over the Y target classes, and $[\mathcal{M}(\mathrm{x}_i; \theta^{(t)})]_j$ is the j-th entry denoting the probability of x_i belonging to the target class j, inferenced by the current model. Then the majority vote (the pseudo label of cluster $\mathcal{C}_k^{(t)}$) is derived as:

$$y_{maj} = \underset{j}{\operatorname{argmax}} \Big(\sum_{i \in \mathcal{C}_k^{(t)}} \mathbb{1}\{\widetilde{y}_i = j\} \Big) / |\mathcal{C}_k^{(t)}|. \tag{3}$$

The instances that are not predicted as y_{maj} are defined as pseudo errors in the corresponding cluster $\mathcal{C}_k^{(t)}$.

3.3 Adaptive Sampling of Representative Errors

For each round of active learning, assuming that the labeling budget is b, we need to decide how we should select the b samples from the unlabeled pseudo errors.

Algorithm 1 Round t of REAL

Input: unlabeled pool $\mathcal{D}_u$, budget for one iteration b, classification model $\mathcal{M}$, number of clusters K, model's encoding part $\Phi(.)$

Output: sampled set Q

$\mathcal{C}_k^{(t)} = \text{KMeans}\big(\Phi(\mathcal{D}_u)\big)$, $(k \in \{1, \ldots, K\})$ ▷ Clustering $\mathcal{D}_u$

for $k \in \{1, \ldots, K\}$ **do** ▷ Process cluster $\mathcal{C}_k$

  Run Eq. 2 for cluster $\mathcal{C}_k$ to get the instance-level pseudo labels

  Run Eq. 3 to find the cluster-level pseudo label y_{maj} for $\mathcal{C}_k$

  Init pseudo error set $\widetilde{E_k}$ in Eq. 4

  Compute the error density ϵ_k for cluster $\mathcal{C}_k$ by Eq. 4

end

Get the sampling budget b_k based on error density for each cluster using Eq. 6

if $\sum_k b_k < b$ **then**

  $\Delta = b - \sum_k b_k$ ▷ Budget residual

end

$b_k \mathrel{+}= 1, \forall k \in \Delta\text{-argmax}_k(b_k)$ and $b_k > 0$ ▷ Allocate residual to top-Δ largest b_k

$Q = \emptyset$ ▷ Init the sample set

for $k \in \{1, \ldots, K\}$ **do**

  Random sample $\min(|\widetilde{E_k}|, b_k)$ instances from $\widetilde{E_k}$ into Q

end

if $|Q| < b$ **then**

  $Q = Q \cup \{(b - |Q|)$ instances from $\mathcal{D}_u$ with top $\epsilon(.)$ scores (Eq. 5) and not in $Q\}$

end

To ensure the representativeness of selected samples, we allocate the sampling budget b to each cluster according to the density of pseudo errors in the cluster, i.e., the percentage of the pseudo errors within a cluster over the total number of pseudo errors in the whole unlabeled data pool. A larger sampling budget will be allocated to the cluster with a higher pseudo error density. The density of pseudo errors ϵ_k for cluster $\mathcal{C}_k^{(t)}$ is defined as:

$$\epsilon_k = \sum_{\mathrm{x}_e \in \widetilde{E}_k} \epsilon(\mathrm{x}_e), \tag{4}$$

where $\widetilde{E}_k = \{\mathrm{x}_e | \mathrm{x}_e \in \mathcal{C}_k^{(t)}$ and $\widetilde{y}_e \neq y_{maj}\}$ is the pseudo error set in the k-th cluster, and $\epsilon(\mathrm{x}_e)$ is one pseudo error x_e's contribution to the cluster-level error density:

$$\epsilon(\mathrm{x}_e) = 1 - [\mathcal{M}(\mathrm{x}_e; \theta^{(t)})]_{maj}. \tag{5}$$

The sampling budget b_k for the k-th cluster is then normalized as:

$$b_k = \left\lfloor b \frac{\epsilon_k}{\sum_i \epsilon_i} \right\rfloor, \forall k \in \{1 \ldots K\}. \tag{6}$$

Apart from selecting pseudo errors, we also try to select errors near the classification decision boundary by emphasizing clusters with denser pseudo errors.

The empirical evidences in §5 also show that our adaptive budget allocation is able to pick more *representative* pseudo errors along the decision boundary.

In real-world applications of AL, it is possible that there may not be enough pseudo errors to be sampled in a cluster (i.e., $|E_k| < b_k$). For instance, when the model is already well-trained via active learning, most of the data instances will be correctly classified. In those cases, we complement the sampled set Q by instances with a higher erroneous probability within all the unlabeled pool $\mathcal{D}_u$ (Line 18 in Algorithm 1), which are illustrated as blue instances (▲ ●) in Fig. 1.

The complexity of REAL consists of two parts: the inference time $O(|\mathcal{D}_u|)$ and the time for K-Means clustering $O(dK|\mathcal{D}_u|)$, where d is the encoder feature dimension $|\Phi(.)|$. K-Means implemented in faiss [22] costs only 2 or 3 seconds even for large datasets such as AGNEWS and PUBMED in §4.

4 Experimental Setup

4.1 Datasets

Following prior research [45,33,46], We conduct experiments on five text classification datasets from different application domains, i.e., SST-2 [40], AGNEWS [48], PUBMED [11], SNIPS [10], and STOV [43]. Table 1 shows their detailed statistics. Due to the limited computational resources, we follow the prior study [45] and take a subset of the original training set and validation set if they are too large. Specifically, we randomly sample 20K×Y instances form each training set if its size exceeds 20K×Y, where Y is the number of target classes. We also keep the size of validation set no more than 3K to speed up the validation process.

Table 1: Dataset statistics.

DATASET	LABEL TYPE	#TRAIN	#VAL	#TEST	#CLASSES
SST-2	Sentiment	40K	3K	1.8K	2
AGNEWS	News Topic	80K	3K	7.6K	4
PUBMED	Medical Abstract	100K	3K	30.1K	5
SNIPS	Intent	13K	0.7K	0.7K	7
STOV	Question	8.0K	1K	1K	10

4.2 Baselines & Implementation Details

We compare REAL against 8 baselines: (1) ENTROPY selects instances with the most even distribution of prediction probability [27]; (2) PLM-KM [46] is a diversity-based baseline which selects b instances closest to K-Means centers of the `[CLS]` token embeddings; (3) BADGE [3] transforms data into gradient embeddings that encode the model confidence and then use K-Means++ to select; (4) BALD [16] defines uncertainty as the mutual information among different versions (via multiple MC dropouts [15]) of the model's predictions; (5) ALPS [46]

selects by masked language modeling loss in PLM; (6) CAL [33] tries to sample the most contrastive instances along the classification decision boundary; (7) ACTUNE [45] selects the unlabeled samples with high uncertainty for active annotation and those with low uncertainty for semi-supervised self-training by weighted K-Means; Since semi-supervised self-training is out of the scope of our current work, we remove the self training part from ACTUNE for a fair comparison. (8) RANDOM uniformly samples data from the unlabeled pool $\mathcal{D}_u$.

For the text classification model $\mathcal{M}$, we follow the prior study [45] and use ROBERTA-BASE [31] implemented in the HuggingFace library [42] for our experiments. We train the model on the initial warm-up labeled set for 10 epochs, and continually train the model for 4 epochs after each round of active sampling to avoid overfitting. We evaluate the model 4 times per training epoch on the validation set and keep the best version. At the end of training, we test the previously-saved best model on the hold-out test set. We choose the best hyperparameters for baselines as indicated in their original papers. We set the AL rounds to be 8 for all the 9 AL methods. Following [46,33], all of our methods and baselines are run with 4 different random seed and the result is based on the average performance on them. This creates 5 (datasets) $\times$ 4 (random seeds) $\times$ 13 (8 baselines + 1 REAL + 4 REAL variants) $\times$ 8 (rounds) = 2080 experiments, which is almost the limit of our computational resources. More details on the experiment setup can be found in our code repository.

5 Results

In the experimental study, we try to answer the following research questions:

- **RQ1.** *Classification Performance*: How is the classification performance of REAL compared to baselines? (§5.1)
- **RQ2.** *Representative errors*: What are the characteristics of the samples, e.g., error rate and representativeness? (§5.2)
- **RQ3.** *Ablation & hyperparameter*: What is the performance of different design variants of REAL? How robust is REAL under different hyperparameter settings? (§5.3)

5.1 Classification Performance

For **RQ1**, we compare the classification performance of REAL against state-of-the-art baselines. Following the existing work of text classification [26], we use two criteria: accuracy and F1-macro to measure the model performances. Accuracy is the fraction of predictions our model got right. F1-macro is the average of accuracy independently measured for each class (i.e., treating different classes equally).

Table 2 shows the average accuracy and F1-macro scores of different AL strategies on all the datasets. The detailed accuracy for each AL round is shown in Fig. 2. REAL outperforms all the baselines by 0.43% – 0.70% performance

Table 2: Mean accuracy (top half) and mean F1-macro (bottom half).

DATASET	ENTROPY	PLM-KM	BALD	BADGE	ALPS	CAL	ACTUNE	RANDOM	REAL
SST-2	91.95	90.74	90.23	90.90	90.88	91.49	91.82	89.91	**92.41**
AGNEWS	90.34	90.19	89.73	90.11	89.70	90.65	90.57	89.23	**91.08**
PUBMED	80.99	80.90	79.06	81.28	79.93	81.26	81.53	80.60	**82.17**
SNIPS	95.99	95.51	95.26	95.52	94.93	96.07	96.16	94.81	**96.63**
STOV	86.56	85.83	85.40	86.39	85.62	86.67	86.27	84.82	**87.37**
SST-2	91.94	90.47	90.22	90.90	90.95	91.48	91.82	89.90	**92.41**
AGNEWS	90.35	89.84	89.66	90.12	89.60	90.26	90.66	89.31	**90.96**
PUBMED	73.92	74.21	71.54	74.82	73.48	74.99	75.30	73.90	**75.78**
SNIPS	96.06	95.60	95.35	95.58	94.82	96.13	96.22	94.86	**96.69**
STOV	86.76	85.96	85.46	86.53	85.72	86.83	86.41	84.99	**87.53**

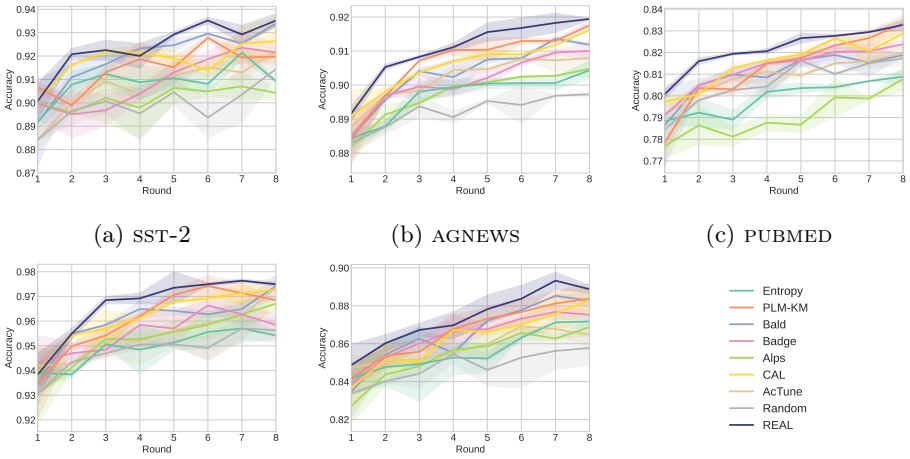
gain w.r.t. the mean accuracy of all the eight AL rounds. The two most recent baselines, CAL and ACTUNE rank in the second or third places in most cases, which is consistent with the reports in their original papers. ENTROPY is also a strong baseline and performs better than PLM-KM and ALPS. The relatively good performance of ENTROPY can be explained by pretrained language model's good uncertainty estimations [12]. It stands out in SST-2 dataset, probably because it is relatively easy to pick samples around the decision boundary for the binary classification task. As shown in Fig. 2, PUBMED is a difficult dataset to learn. The model's test accuracy on it is less than 85% even with the full training set. It is probably because the professional medical text is rarely seen in ROBERTA and the label distribution of PUBMED is a skewed. ENTROPY and BALD perform very badly on PUBMED, since they heavily rely on the distribution of the prediction probability. Compared to other baseline methods, REAL has a clear advantage on PUBMED.

5.2 Representative Errors

We address **RQ2** by investigating whether REAL can sample *representative errors* and comparing it with those baselines. Specifically, we evaluate the capability of REAL in selecting representative error samples from the following two perspectives:

- The error rate and initial training loss of samples (Table 3 and Fig. 4);
- The distribution divergence between samples and boundary errors (Fig. 4).

Error Rate and Initial Training Loss of Samples Table 3 shows the mean error rates and initial training loss (for all AL rounds) of samples Q for different AL strategies across all the datasets. The error rate $\varepsilon(Q)$ is the proportion of wrongly-predicted instances in Q by comparing the model prediction with the ground truth label for each instance. It is inappropriate to directly compare the error rates $\varepsilon(Q)$ of different AL strategies, because the error rates $\varepsilon(\mathcal{D}_u)$ in



(a) SST-2 (b) AGNEWS (c) PUBMED

(d) SNIPS (e) STOV (f) Legend

Fig. 2: The accuracy of different active learning rounds on each dataset. The height of the shadow area denotes the std of accuracy.

the whole unlabeled pool $\mathcal{D}_u$ are different. It is more easier to achieve a high sampling error rate $\varepsilon(Q)$ given a high background error rate $\varepsilon(\mathcal{D}_u)$. Therefore, we compares the `lift` of sampling error rate, which is defined as $\varepsilon(Q)/\varepsilon(\mathcal{D}_u)$, which implies how effective does an AL strategy select errors, compared with random selection. Another metric is the average cross entropy loss ℓ_0 of samples Q in the first training step , which is a more fine-grained version of error rate. Many previous research work [47,32,41,20] have already validated that samples with higher loss are usually more informative to the model.

Table 3 shows REAL usually has a large `lift` of sampling error rate, second only to ENTROPY, despite the fact that the unlabeled pool error rate $\varepsilon(\mathcal{D}_u)$ of REAL is the lowest. The large `lift` of sampling error rate implies REAL successfully identifies the errors in $\mathcal{D}_u$. It is worth mentioning that $\varepsilon(\mathcal{D}_u)$ can serve as test set because we don't use $\mathcal{D}_u$ in the previous AL rounds for training or validation. REAL has the lowest error rate on 4 out of 5 datasets, which means the highest accuracy when testing on $\mathcal{D}_u$. The initial training loss of REAL's samples is also the largest in the last four datasets. Fig. 4 shows more detailed loss distribution for each AL round.

Representativeness We investigate how REAL's samples align with ground-truth errors on decision boundary compared to baselines. Based on theoretical studies on margin theory for active learning [4], selecting instances close to the decision boundary can significantly reduce the number of annotations required [33,37,50,21]. Though identifying the precise decision boundaries for deep neural networks is intractable [13], we use the basic grid statistics on t-SNE embeddings as an empirical solution. Specifically, we apply t-SNE to the `[CLS]` token em-

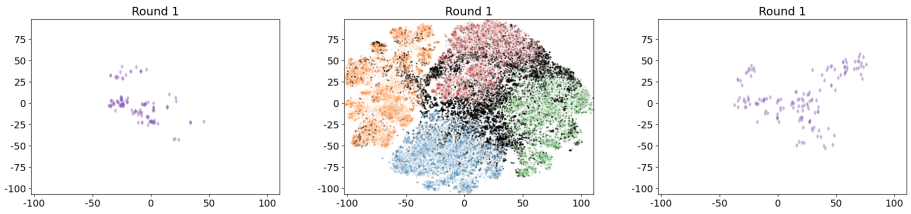
Table 3: Sampling error rate $\varepsilon(.)$, `lift`, and the average first step training loss ℓ_0.

DATASET	METRIC	ENTROPY	PLM-KM	BADGE	CAL	ACTUNE	RANDOM	REAL
SST-2	$\varepsilon(Q)$	**0.4959**	0.1841	0.2308	0.4821	0.4334	0.1284	0.4739
	$\varepsilon(\mathcal{D}_u)$	**0.1194**	0.1251	0.1259	0.1215	0.1170	0.1338	0.1212
	`lift`	**4.1530**	1.4713	1.8325	3.9670	3.7055	0.9596	3.9113
	ℓ_0	0.6984	0.8100	**1.0538**	0.6915	0.8526	0.6660	0.9938
AGNEWS	$\varepsilon(Q)$	**0.6092**	0.1904	0.2246	0.5637	0.5325	0.1142	0.5537
	$\varepsilon(\mathcal{D}_u)$	0.1009	0.1039	0.1041	0.0995	0.0991	0.1115	**0.0959**
	`lift`	**6.0377**	1.8320	2.1576	5.6667	5.3730	1.0239	5.7737
	ℓ_0	1.2504	0.8597	0.9477	1.0926	1.3009	0.5707	**1.3636**
PUBMED	$\varepsilon(Q)$	**0.6701**	0.3164	0.3634	0.6103	0.6231	0.1987	0.6046
	$\varepsilon(\mathcal{D}_u)$	0.1943	0.1971	0.1928	0.1941	0.1907	0.1998	**0.1858**
	`lift`	**3.4487**	1.6048	1.8845	3.1452	3.2670	0.9943	3.2531
	ℓ_0	1.5117	1.3533	1.6009	1.2871	1.4494	1.0222	**1.7040**
SNIPS	$\varepsilon(Q)$	0.4107	0.1226	0.1120	**0.4237**	0.2963	0.0276	0.4002
	$\varepsilon(\mathcal{D}_u)$	0.0268	0.0337	0.0308	0.0280	0.0265	0.0393	**0.0231**
	`lift`	15.3183	3.6410	3.6338	15.1568	11.1895	0.7023	**17.2902**
	ℓ_0	**1.0176**	0.5209	0.5080	1.0470	0.9491	0.1842	0.9356
STOV	$\varepsilon(Q)$	**0.7328**	0.2536	0.3506	0.6904	0.6659	0.1307	0.7162
	$\varepsilon(\mathcal{D}_u)$	0.1048	0.1263	0.1209	0.1094	0.1101	0.1386	**0.1045**
	`lift`	**6.9934**	2.0079	2.8994	6.3114	6.0509	0.9435	6.8548
	ℓ_0	**2.1434**	1.0260	1.3874	2.0255	2.0062	0.6331	2.1131

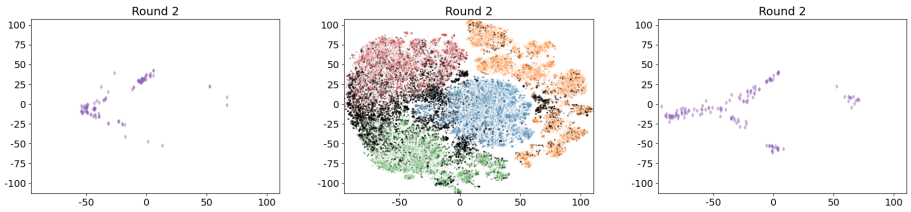
beddings of $\Phi(\mathcal{D}_u)$ and project the original token embeddings of 768 dimensions to 2D plane, as shown in Fig. 3. Then, we split the bounding box of $\mathcal{D}_u$ on 2-D plane into 50×50 uniform grids, where g_i denotes the number of instances that fall into the i-th grid.

We keep only the ground-truth errors within the decision boundaries. Our intuition of decision boundary is where instances have similar representations but different predictions [4]. We hypothesize that the pseudo errors selected by REAL is near to the model's decision boundary because 1) instances in the same cluster have similar representations; and 2) pseudo errors have different predictions than the majority in a cluster. To verify this hypothesis, we compute the distribution entropy of ground-truth labels for each grid, and reserve only the top 0.15 grids with high entropy values. For example, in binary classification, a ground-truth label distribution $[98, 100]$ for grid g_1 means there are 98 instances belonging to class-0 and 100 instances belonging to class-1 in g_1. Suppose another grid g_2 has ground-truth label distribution $[180, 10]$. Then g_1 will be closer to the decision boundary than g_2 because g_1 has a more even distribution, thus a larger entropy. We call grids with top 15% high entropy values as boundary grids. g_i^{ε} denotes the number of ground-truth errors fall into i-th boundary grid.

We compare the Jensen-Shannon divergence (JSD) between boundary grids set $\{g_i^{\varepsilon}\}_{i=1}^{m}$ (m is the number of boundary grids) and $\{s_i\}_{i=1}^{m}$, where s_i is the number of sampled instances in the i-th grid. JSD extends KL divergence (KLD)



(a) ENTROPY (b) Errors (in black) (c) REAL



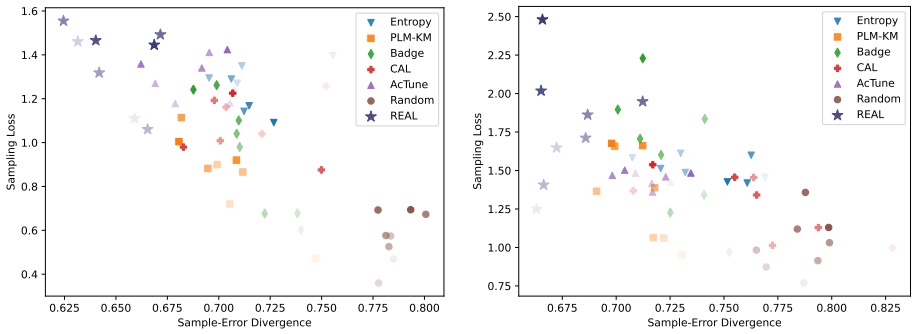
(d) ENTROPY (e) Errors (in black) (f) REAL

Fig. 3: The t-SNE based visualization for the sample/error distribution. Visualizations in the first row are from the first round active selection on AGNEWS and the second row from the second round. Two subfigures in the middle (3b and 3e) are all instances in $\mathcal{D}_u$. The four colors in 3b and 3e indicate different correctly predicted categories, except black. Black dots indicate the ground truth errors. Purple dots in the left side two visualizations are ENTROPY's selections and the right side are REAL's selections.

to derive a symmetric distance measure between two probability distributions P_1 and P_2: $\text{JSD}(P_1 \| P_2) = \frac{1}{2}\text{KLD}(P_1 \| M) + \frac{1}{2}\text{KLD}(P_2 \| M)$, where $M = \frac{1}{2}(P_1 + P_2)$. Together with previously introduced sampling loss, we plot the divergence in Fig. 4 for the two largest datasets AGNEWS and PUBMED. REAL clearly has the lowest divergence, which means our samples' distribution aligns well with the ground-truth errors on the boundary.

Besides the lowest divergence, REAL also has the largest sampling loss. As shown in Fig. 4, REAL's samples clearly distribute in the upper left corner. In contrast, RANDOM's samples lie in the lower right corner. The least sampling loss and largest divergence from boundary errors may be the reason why RANDOM fails. To our surprise, darker dots usually appear in higher positions, in Fig. 4, which means samples in later AL rounds provide larger loss. The reason may be that the model in later AL rounds is stable and confident in its predictions, thus introducing new samples will cause a larger loss.

Fig. 3 provides the case studies of our samples against ENTROPY on AGNEWS. We can see that most errors distribute near the decision boundaries. ENTROPY tends to miss some decision boundary areas and is lack of diversity. REAL matches the boundary errors better.



(a) AGNEWS (b) PUBMED

Fig. 4: Loss v.s. sample-error divergence. Each dot represents a sample set Q from one AL round. The dot shape and color hue indicate AL strategy. The dot transparency indicates different AL rounds. More transparent dot comes from earlier AL round, and darker dot comes from later AL round. REAL clearly shows a lower divergence to the ground truth errors on the decision boundary, and a larger sampling loss.

5.3 Ablation and Hyperparameter Study

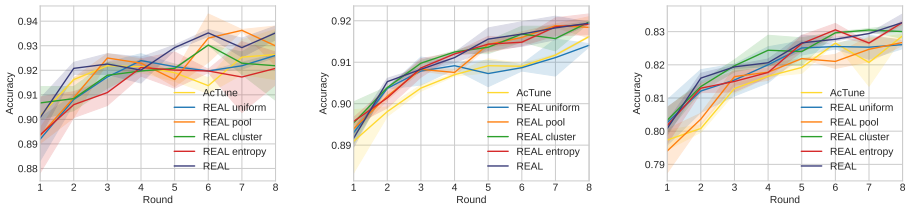
In this section, we address **RQ3** by extensive ablation (Fig. 5) and hyperparameter studies (Fig. 6) to understand the important components in REAL.

A. Ablation Study. (1) We test REAL with different budget allocation strategies (Eq. 6) per cluster. (1.1) Ignore the idea of "allocation by cluster". Specifically, we rank all the instances in $\mathcal{D}_u$ based on its erroneous probability (Eq. 5), and select top-b instances per round (`REAL pool`). (1.2) For each cluster, uniformly sample B/K pseudo errors, i.e., ignore cluster error weights in Eq. 6. (`REAL uniform`)

(2) REAL randomly samples within each cluster's pseudo errors (line 15 in Algorithm 1) based on the adaptive budget. Given the adaptive budget in each cluster, we also try to sample: (2.1) Instances with the largest erroneous probabilities in Eq. 5 (`REAL cluster`); (2.2) Pseudo errors with the largest prediction entropy (`REAL entropy`).

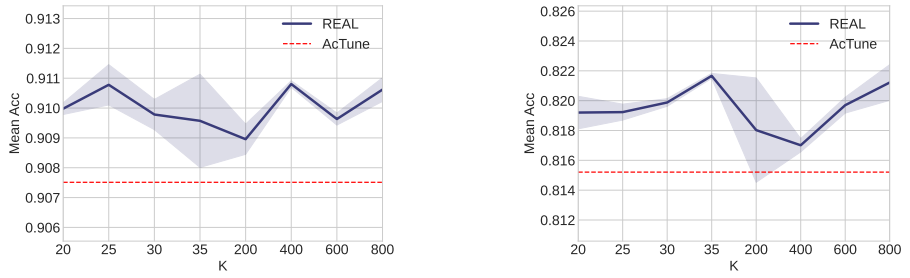
Fig. 5 show that most of REAL's variants still perform better than the best baseline ACTUNE on large datasets AGNEWS and PUBMED. However, the results on SST-2 are unstable. The reason may be that the decision boundary of binary classification is too simple so that dedicated methods are not necessary. `REAL cluster` fails only on SST-2. `REAL uniform` is slightly worse than ACTUNE in later rounds, which indicates the importance of weighted budget allocation for REAL. `REAL pool` is very close to ACTUNE on PUBMED, possibly because lacking diversity hurts it on the most difficult dataset.

B. Hyperparameter Study We study the impact of varying the number of clusters K. Experiment results in Fig. 6 shows our method stably beats the best baseline across a wide range of K (on a scale of tens to hundreds).



(a) SST-2 (b) AGNEWS (c) PUBMED

Fig. 5: Ablation study on different variants of REAL.



(a) AGNEWS (b) PUBMED

Fig. 6: Mean acc under a wide range of #clusters K for REAL against the best baseline ACTUNE on our largest two datasets.

6 Conclusion and Future Work

We present REAL, a novel AL sampling strategy that selects *representative pseudo errors* for efficient model training. We define pseudo errors as minority predictions within each cluster. The sampling budget per cluster is adaptive to the cluster's total estimated error density. Experiments on five datasets demonstrate that REAL performs better than other AL sampling strategies consistently. By analyzing the actively sampled instances, we find that REAL improves over all the best-performing baselines by guiding uncertainty sampling in errors near the decision boundary. The ablation study shows most alternative designs of REAL still beat the state-of-the-art baseline.

Future work will investigate the theoretical effectiveness of selecting errors near decision boundary for AL and the diversity within pseudo errors. Currently we only take text classification as an example to illustrate the effectiveness of REAL. But the framework of REAL can be easily adapted to other tasks such as image classification implemented in neural classification architectures.

Ethical Statement

All the datasets are widely-used benchmark text classification datasets and are publicly-available online, which do not have any privacy issues. Also, our approach can benefit data labeling workers and bring welfare to them. Data labeling is very costly and labour-intensive. For example, labeling toxic content is reported to be a "mental torture" [35]. Our approach aims to make active learning more label-efficient and can reduce the workload of data labeling workers, which is beneficial to the mental health of data labeling workers.

Acknowledgments

This work was done during Cheng Chen's internship at Singapore Management University (SMU) under the supervision of of Dr. Yong Wang. This work was supported by the National Key Research and Development Program of China (2020YFB1710004), Lee Kong Chian Fellowship awarded to Dr. Yong Wang by SMU, and the National Science Foundation of China under the grant 62272466. We would like to thank all the anonymous reviewers for their valuable feedback.

References

1. Aharoni, R., Goldberg, Y.: Unsupervised domain clusters in pretrained language models. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7747–7763 (2020)
2. Arthur, D., Vassilvitskii, S.: K-means++: The advantages of careful seeding. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. p. 1027–1035 (2007)
3. Ash, J.T., Zhang, C., Krishnamurthy, A., Langford, J., Agarwal, A.: Deep batch active learning by diverse, uncertain gradient lower bounds. In: Proceedings of the International Conference on Learning Representations (2020)
4. Balcan, M.F., Broder, A., Zhang, T.: Margin based active learning. In: 20th Annual Conference on Learning Theory. pp. 35–50 (2007)
5. Baram, Y., Yaniv, R.E., Luz, K.: Online choice of active learning algorithms. Journal of Machine Learning Research **5**(Mar), 255–291 (2004)
6. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning. pp. 1597–1607 (2020)
7. Choi, J., Yi, K.M., Kim, J., Choo, J., Kim, B., Chang, J., Gwon, Y., Chang, H.J.: Vab-al: Incorporating class imbalance and difficulty with variational bayes for active learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6749–6758 (2021)
8. Chung, C., Lee, J., Park, K., Lee, J., Kim, M., Song, M., Kim, Y., Choo, J., Hong, S.R.: Understanding human-side impact of sampling image batches in subjective attribute labeling. Proceedings of the ACM on Human-Computer Interaction **5**, 1–26 (2021)
9. Citovsky, G., DeSalvo, G., Gentile, C., Karydas, L., Rajagopalan, A., Rostamizadeh, A., Kumar, S.: Batch active learning at scale. Advances in Neural Information Processing Systems **34**, 11933–11944 (2021)

10. Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., et al.: Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. arXiv preprint arXiv:1805.10190 (2018)
11. Dernoncourt, F., Lee, J.Y.: PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers). pp. 308–313 (2017)
12. Desai, S., Durrett, G.: Calibration of pre-trained transformers. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. pp. 295–302 (2020)
13. Ducoffe, M., Precioso, F.: Adversarial active learning for deep networks: a margin based approach. arXiv preprint arXiv:1802.09841 (2018)
14. Fang, M., Li, Y., Cohn, T.: Learning how to active learn: a deep reinforcement learning approach. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 595–605 (2017)
15. Gal, Y., Ghahramani, Z.: Bayesian convolutional neural networks with bernoulli approximate variational inference. arXiv preprint arXiv:1506.02158 (2015)
16. Gal, Y., Islam, R., Ghahramani, Z.: Deep Bayesian active learning with image data. In: Proceedings of the 34th International Conference on Machine Learning. vol. 70, pp. 1183–1192 (2017)
17. Gissin, D., Shalev-Shwartz, S.: Discriminative active learning. arXiv preprint arXiv:1907.06347 (2019)
18. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020)
19. Hsu, W.N., Lin, H.T.: Active learning by learning. In: Proceedings of the AAAI Conference on Artificial Intelligence (2015)
20. Huang, S., Wang, T., Xiong, H., Huan, J., Dou, D.: Semi-supervised active learning with temporal output discrepancy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3447–3456 (2021)
21. Huijser, M., van Gemert, J.C.: Active decision boundary annotation with deep generative models. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5286–5295 (2017)
22. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with GPUs. IEEE Transactions on Big Data **7**(3), 535–547 (2019)
23. Kim, Y., Shin, B.: In defense of core-set: A density-aware core-set selection for active learning. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. p. 804–812 (2022)
24. Konyushkova, K., Sznitman, R., Fua, P.: Learning active learning from data. Advances in Neural Information Processing Systems **30** (2017)
25. Krempl, G., Kottke, D., Lemaire, V.: Optimised probabilistic active learning (opal) for fast, non-myopic, cost-sensitive active classification. Machine Learning **100**, 449–476 (2015)
26. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence (2015)
27. Lewis, D.D.: A sequential algorithm for training text classifiers. In: Acm Sigir Forum. vol. 29, pp. 13–19 (1995)
28. Lewis, D.D., Catlett, J.: Heterogeneous uncertainty sampling for supervised learning. In: Machine Learning Proceedings, pp. 148–156 (1994)

29. Li, M., Sethi, I.K.: Confidence-based active learning. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(8), 1251–1261 (2006)
30. Liu, M., Buntine, W., Haffari, G.: Learning how to actively learn: A deep imitation learning approach. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1874–1883 (2018)
31. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
32. Luo, J., Wang, J., Cheng, N., Xiao, J.: Loss prediction: End-to-end active learning approach for speech recognition. In: 2021 International Joint Conference on Neural Networks. pp. 1–7 (2021)
33. Margatina, K., Vernikos, G., Barrault, L., Aletras, N.: Active learning by acquiring contrastive examples. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 650–663 (2021)
34. Muller, M., Wolf, C.T., Andres, J., Desmond, M., Joshi, N.N., Ashktorab, Z., Sharma, A., Brimijoin, K., Pan, Q., Duesterwald, E., et al.: Designing ground truth and the social life of labels. In: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. pp. 1–16 (2021)
35. Perrigo, B.: Inside facebook's african sweatshop. Time `https://time.com/6147458/facebook-africa-content-moderation-employee-treatment/`, accessed on Mar 28, 2023
36. Roth, D., Small, K.: Margin-based active learning for structured output spaces. In: Proceedings of the 17th European Conference on Machine Learning. pp. 413–424 (2006)
37. Ru, D., Feng, J., Qiu, L., Zhou, H., Wang, M., Zhang, W., Yu, Y., Li, L.: Active sentence learning by adversarial uncertainty sampling in discrete space. In: Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 4908–4917 (2020)
38. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. In: International Conference on Learning Representations (2018)
39. Sia, S., Dalmia, A., Mielke, S.J.: Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. pp. 1728–1736 (2020)
40. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1631–1642 (2013)
41. Wan, C., Jin, F., Qiao, Z., Zhang, W., Yuan, Y.: Unsupervised active learning with loss prediction. Neural Computing and Applications pp. 1–9 (2021)
42. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45 (2020)
43. Xu, J., Wang, P., Tian, G., Xu, B., Zhao, J., Wang, F., Hao, H.: Short text clustering via convolutional neural networks. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing. pp. 62–69 (2015)

44. Yoo, D., Kweon, I.S.: Learning loss for active learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 93–102 (2019)
45. Yu, Y., Kong, L., Zhang, J., Zhang, R., Zhang, C.: Actune: Uncertainty-based active self-training for active fine-tuning of pretrained language models. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1422–1436 (2022)
46. Yuan, M., Lin, H.T., Boyd-Graber, J.: Cold-start active learning through self-supervised language modeling. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. pp. 7935–7948 (2020)
47. Yuan, T., Wan, F., Fu, M., Liu, J., Xu, S., Ji, X., Ye, Q.: Multiple instance active learning for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5330–5339 (2021)
48. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. Advances in Neural Information Processing Systems **28** (2015)
49. Zhang, Z., Fang, M., Chen, L., Namazi Rad, M.R.: Is neural topic modelling better than clustering? an empirical study on clustering with contextual embeddings for topics. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 3886–3893 (2022)
50. Zhu, J.J., Bento, J.: Generative adversarial active learning. arXiv preprint arXiv:1702.07956 (2017)