Counterfactual Active Learning for Out-of-Distribution Generalization

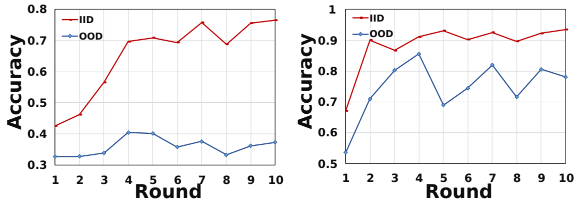
Xun Deng [1], **Wenjie Wang** [3][*], **Fuli Feng** [1,2], **Hanwang Zhang** [4], **Xiangnan He** [1], **Yong Liao** [5],
[1] University of Science and Technology of China, Hefei 230026, China
[2] Institute of Dataspace, Hefei, Anhui, China
[3] National University of Singapore
[4] Nanyang Technological University
[5] China Academic of Electronics and Information Technology
{dx981228,wenjiewang96,fulifeng93,xiangnanhe}@gmail.com
hanwangzhang@ntu.edu.sg, yliao@ustc.edu.cn

Abstract

We study the out-of-distribution generalization of active learning that adaptively selects samples for annotation in learning the decision boundary of classification. Our empirical study finds that increasingly annotating seen samples may hardly benefit the generalization. To address the problem, we propose Counterfactual Active Learning (CounterAL) that empowers active learning with counterfactual thinking to bridge the seen samples with unseen cases. In addition to annotating factual samples, CounterAL requires annotators to answer counterfactual questions to construct counterfactual samples for training. To achieve CounterAL, we design a new acquisition strategy that selects the informative factual-counterfactual pairs for annotation; and a new training strategy that pushes the model update to focus on the discrepancy between factual and counterfactual samples. We evaluate CounterAL on multiple public datasets of sentiment analysis and natural language inference. The experiment results show that CounterAL requires fewer acquisition rounds and outperforms existing active learning methods by a large margin in OOD tests with comparable IID performance.

1 Introduction

Active Learning (AL) is widely applied to alleviate the scarcity of labeled data in various machine learning applications (Ren et al., 2021) such as financial fraud detection (Carcillo et al., 2017) where the annotation cost is high. Existing research on AL mainly focuses on the design of an acquisition strategy that adaptively selects informative samples for annotation from an unlabeled pool (Tan et al., 2021; Kirsch et al., 2019). While the models learned by AL methods have comparable in-distribution performance with the ones learned from fully labeled data, they typically result in poor generalization on out-of-distribution samples (Krishnan et al., 2021). As OOD samples widely exist in practice (Wang et al., 2022), it is critical to enhance the OOD generalization of AL.



(a) Natural Language Inference (b) Sentiment Analysis

Figure 1: The performance on IID and OOD tests of sentiment analysis and natural language inference datasets (Kaushik et al., 2020) in the learning procedure of the AL method: Entropy.

We first investigate how the OOD generalization ability varies during the AL procedure. Figure 1 shows the empirical evidence on two text classification datasets (Kaushik et al., 2020) of sentiment analysis and natural language inference with both IID and OOD tests, where we evaluate a representative AL method: Entropy (Ren et al., 2021). The IID performance steadily increases, while the OOD performance fluctuates at a much lower range close to the initial status as the number of annotations increases. We thus hypothesize that the seen samples selected from the unlabeled pool are not informative for OOD generalization. The model will recognize some spurious correlations between input features and labels probably varying beyond the observed data. In other words, the model may over-emphasize some non-causal features to construct the decision boundary.

Counterfactual thinking (Roese, 1997; Pearl, 2009) is essential for bridging the gap between seen IID samples and unseen OOD ones. It answers counterfactual questions like *"what would the sentence be if its sentiment were negative?"*, indicating the causal features that change labels and

[*] Wenjie Wang is the corresponding author. This work is supported by the National Key Research and Development Program of China (2021YFC3300500-02), the National Natural Science Foundation of China (62272437 and U20B2047), and the CCCD Key Lab of Ministry of Culture and Tourism.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 11362–11377
July 9-14, 2023 ©2023 Association for Computational Linguistics

break the spurious correlations in seen IID samples. Along this line, counterfactual training (Teney et al., 2020) is effective for enhancing OOD generalization, which leverages factual and counterfactual samples to push the learning of decision boundary to focus on features causally affect the label (Sauer and Geiger, 2021; Teney et al., 2020). We thus believe that incorporating counterfactual samples into AL can enhance OOD generalization[1].

To embrace counterfactual samples, it is natural to consider combining AL and counterfactual sample construction in a pipelined manner. As a pre-stage of AL, we can first augment the entire unlabeled pool by pairing all samples with counterfactual samples, then perform AL over the augmented pool. As a post-stage of AL, with the factual samples acquired by an AL method, we can construct counterfactual samples, and perform counterfactual training to obtain the final model. However, the pre-stage approach is cost unfriendly due to the large size of the unlabeled pool for counterfactual construction. The post-stage approach cannot consider the potential gain from the counterfactual samples in the acquisition of AL.

In this work, we consider combining sample annotation and counterfactual sample construction in the procedure of AL. Towards this end, we propose *Counterfactual Active Learning* (CounterAL), which requires annotators to additionally perform counterfactual thinking on the selected samples. Given a selected sample (x), in addition to the annotation (y), the annotator further imagines a counterfactual class ($\bar{y}$) and edits the factual features to be coherent with the counterfactual class, *i.e.,* constructing the counterfactual feature ($x^*_{\bar{y}}$). CounterAL then updates model parameters over pairs of factual samples (x, y) and counterfactual samples $(x^*_{\bar{y}}, \bar{y})$ to enhance the OOD generalization.

The key to the success of CounterAL lies in: 1) an acquisition strategy that looks ahead the construction of counterfactual samples to select informative factual and counterfactual pairs; and 2) a training strategy that recognizes the discrepancy between each pair of factual and counterfactual samples. In the light that informative factual-counterfactual pairs are close to each other (label flip with fewer feature changes), we design a variability-based acquisition strategy to select factual samples with high variability to model updates (high probability of label flip). Besides, we incorporate a new dropout to model training which masks the common features of factual and counterfactual samples to push the model to focus more on the discrepancy that implies causal features. Lastly, we take two text classification problems of sentiment analysis and natural language inference as examples and validate the strong OOD generalization ability of CounterAL on three public datasets. Our main contributions are summarized as follows:

- We propose a *Counterfactual Active Learning* paradigm for OOD generalization, which extends the role of human annotators in active learning from simple annotation to also performing counterfactual thinking.
- We design a novel acquisition strategy and a new training strategy for CounterAL, which enables the acquisition of informative factual-counterfactual pairs for OOD generalization under affordable construction cost.
- We conduct extensive experiments on three public datasets of two text classification tasks, validating the effectiveness of the CounterAL framework in enhancing OOD generalization of AL.

2 Methodology

In this section, we first present the OOD generalization issue in active learning, and then detail the proposed CounterAL framework, followed by several discussions on CounterAL.

2.1 OOD Generalization in Active Learning

We focus on batch-mode active learning for classification problems. Given a huge unlabeled pool with samples following the distribution of $\mathcal{X}_U$, we need to learn a K-way classifier $\hat{y} = f(x; \theta)$ where x and θ denote sample features and model parameters, respectively. $\hat{y}$ denotes the prediction in the label space $\mathcal{Y} = \{1, \cdots, K\}$, and the label of sample x is y. The target of active learning is to adaptively selects informative samples from the unlabeled pool to construct a labeled set $\mathcal{X}_L$ for training the final model[2].

Active Learning assumes that the distribution of the unlabeled pool can represent the real distribution of samples, which is usually not satisfied in reality. Specifically, deep models tend to use the spurious correlations between non-causal features and label in $\mathcal{X}_L$ for prediction (Kaushik et al., 2021;

[1] Empirical evidence about the effectiveness of data augmentation in recent AL work (Ducoffe and Precioso, 2018) supports this hypothesis to some extent.

[2] In practice, we replay all labeled samples to train a new model instead of applying the final model of AL directly.

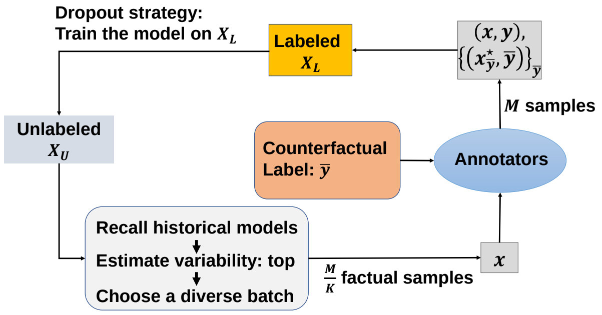

Dropout strategy:
Train the model on X_L
Labeled X_L
(x,y), $\{(x^{\star}_{\bar{y}},\bar{y})\}_{\bar{y}}$
M samples
Unlabeled X_U
Counterfactual Label: $\bar{y}$
Annotators
Recall historical models
Estimate variability: top
Choose a diverse batch
$\frac{M}{K}$ factual samples
x

Figure 2: Overview of CounterAL framework.

YU et al., 2022). Such spurious correlations often shift in the real test sets, causing the models' poor generalization. The common OOD testing samples call for a new AL objective: enhancing the OOD generalization of AL methods while maintaining the IID performance.

To boost OOD generalization, past literature in Natural Language Understanding has made some exploration such as instance reweighting (Utama et al., 2020; Ghaddar et al., 2021). However, these methods are not as effective when the size of the training set is small (See empirical evidence in Appendix B.2). Another promising line of work uses counterfactual training (Teney et al., 2020) to pursue strong OOD generalization. It learns model parameters by comparing pairs of factual samples (x, y) and counterfactual samples $(x^*_{\bar{y}}, \bar{y})$. The sample pair exhibits the relation how feature changes $(x \rightarrow x^*_{\bar{y}})$ cause label changes $(y \rightarrow \bar{y})$. The editions on causal features and the labels naturally break the spurious correlations, as the non-causal features appear in both the factual and counterfactual samples with different classes (Kaushik et al., 2020). Due to counterfactual training, the model cannot rely on such spurious correlations for prediction, improving the OOD generalization ability (Kaushik et al., 2020; Nie et al., 2019).

2.2 Counterfactual Active Learning

Considering the success of counterfactual training in enhancing the model's OOD generalization ability, we set the target of pursuing the OOD generalization of active learning as constructing a labeled set with both factual and counterfactual samples. Without loss of generality, we take binary classification to explain the method, and it is simple to extend it to general multi-class classification problems. It is natural to consider the incorporation of counterfactual construction into each round of active learning. We term this new paradigm as Counterfactual Active Learning (Figure 2). In the procedure of CounterAL, annotators play twofold roles: annotating factual samples x selected from the unlabeled pool and **imagining the counterfactual features $x^*_{\bar{y}}$ given a counterfactual label $\bar{y}$.**

Similar to conventional active learning, the core of CounterAL includes an acquisition strategy to fetch factual samples at each round and a training strategy to update model parameters to adjust the acquisition strategy. As counterfactual samples are invisible, the acquisition strategy needs to look ahead the counterfactual construction to select informative factual-counterfactual pairs. The training strategy then updates model parameters over all annotated factual and counterfactual samples. The update is expected to lead the acquisition strategy to emphasize more on the factual samples that can produce the informative pairs of (x, y) and $(x^*_{\bar{y}}, \bar{y})$.

2.2.1 Variability-based Acquisition Strategy

Distance between Factual and Counterfactual Samples. Our key consideration for the informative factual-counterfactual pair is that the factual and counterfactual samples are similar to each other. As factual and counterfactual samples locate at different sides of the decision boundary, similar pairs are closer to the decision boundary, which is more informative for the learning of decision boundary (Teney et al., 2020). What's more, similar pairs help to discover causal features that decide labels, boosting the OOD generalization ability (*cf.* Table 11 in Appendix B).

As the distance between similar factual and counterfactual samples is small, it will be difficult for the model to distinguish them. Intuitively, minor changes on the causal features of the factual sample result in the label flip $(y \rightarrow \bar{y})$, and the model is apt to ignore the minor changes and alter its prediction during the training process. Inspired by the dataset map construction in (Swayamdipta et al., 2020), we propose to use the variability of the model's historical predictions to measure the dynamics of label flip and approximately estimate the distance between factual and counterfactual samples.

Samples with High Variability. As lacking of labels, we propose a new variability-based acquisition strategy that measures the easiness of label flip. We define the variability of a sample x over the set of historical models with different parameters like (Liu et al., 2022). Formally,

$$\begin{aligned} v(x) &= \max\{v_i(x) | i \in [1, K]\}, \\ v_i(x) &= \text{Var}(P(y_i | x, \theta_s)). \end{aligned} \quad (1)$$

Table 1: The construction of a counterfactual sample on the NLI task is shown. The sample is composed of a premise and a hypothesis about it. We highlight causal feature in red and the edited feature in blue. The model is apt to learn the correlation between lexical overlap and the entailment label, and will always predict entailment for similar samples with high lexical overlap, neglecting the numerical inconsistency. The existence of counterfactual samples breaks such spurious correlation, and forces the model to capture the logic of “less than".

Label	Example
Entailment (y)	**Tim has 350 pounds of cement in 100, 50, and 25 pound bags; Tim has less than 750 pounds of cement in 100, 50, and 25 pound bags.**
Contradiction ($\bar{y}$)	**Tim has 350 pounds of cement in 100, 50, and 25 pound bags; Tim has less than 350 pounds of cement in 100, 50, and 25 pound bags.**

$P(y_i|x, \theta_s)$ denotes the prediction probability on the i-th class given by the model s with parameters θ_s. $v_i(x)$ denotes the variance of prediction probability on the i-th class over the model set at the current round r. Note that we omit the superscript r for briefness. Similar to (Swayamdipta et al., 2020), we adopt the checkpoints before round r as the model set. In this way, the high variability means that the factual sample might have varying predictions along the learning procedure,

Acquisition Strategy. As model checkpoints are not available at the initial round (*i.e.,* $r = 1$), we adopt a random strategy for acquisition. To acquire a diverse batch in the following rounds (*i.e.,* $r > 1$), we first select a batch of T samples with the highest variability according to Equation 1, these samples are then clustered with KMeans algorithm and the sample nearest to the centroid of each cluster is returned. The intuition is that samples in the same cluster tend to share similar linguistic properties, and keeping them will not improve the diversity of the acquired batch.

2.2.2 Discrepancy-aware Dropout

After the acquisition at each round, CounterAL updates the model over the factual and counterfactual pairs. However, the expert annotation budget of active learning is limited and practically less than 1000 (Tan et al., 2021). To efficiently utilize the annotated factual and counterfactual sample, we propose a dropout strategy that forces the model to focus on the causal features that are different between factual and counterfactual samples for prediction. In particular, applying discrepancy-aware dropout forms a three-step update procedure:

- **Substraction,** which calculate the difference between x and $x^*_{\bar{y}}$ through $\delta(x, x^*_{\bar{y}}) = |x - x_{\bar{y}^*}|$.
- **Masking**, which removes similar features through a feature-wise dropout mask m:

$$\tilde{x} = m \odot x,\ \tilde{x}^*_{\bar{y}} = m \odot x^*_{\bar{y}},\ m_i = \begin{cases} 1, & \delta_i > \tau, \\ 0, & \delta_i \le \tau, \end{cases} \tag{2}$$

Table 2: User study on Tweet Data: the average time cost of annotating a factual sample (t_1), a counterfactual sample (t_2), and the mean cost of annotating a factual-counterfactual pair (t_3). s stands for seconds.

t_1	t_2	t_3
54s	48s	44s

where $\odot$ means element-wise multiplication, m_i denotes the i-th entry of the dropout mask; τ is the threshold for masking. Note that only significantly different features between factual and counterfactual samples are used for the following parameter update.

- **Updating,** which updates model parameters with masked features $\tilde{x}$ and $\tilde{x}^*_{\bar{y}}$:

$$\underset{\theta}{min}\, \mathbb{E}_{(x,y)\sim\mathcal{X}_L}[l(\tilde{x}, y; \theta) + l(\tilde{x}^*_{\bar{y}}, \bar{y}; \theta)], \tag{3}$$

where $l(\cdot)$ is typically the cross-entropy loss.

2.3 Discussions

Counterfactual Construction. For each factual sample, the annotator is asked to give the label based on causal features. The annotator further seeks edits on causal features according to a given counterfactual label. They are not expected to alter other features unless necessary. We exemplify the annotation and show the effect of counterfactual samples with the cases in Table 1.

User Study on Annotation Cost. The time cost of variability calculation and KMeans clustering are relatively low. The major cost of CounterAL lies in the human annotation. To explore whether it is affordable to conduct CounterAL in practice, we conduct a user study on the Tweet data (Rosenthal et al., 2017) and require annotators to annotate both the factual and counterfactual samples. From the result in Table 2, we find that the average time spent on creating a counterfactual sample is less than the time on labeling a factual sample, showing that CounterAL is cost-effective. We provide more details about the settings of the user study, evaluation of sample quality, and the feedback from the annotators in the Appendix C.

Instructions for Annotators The instructions given to annotators for editing counterfactual samples are as follows: "The edition should satisfy (i) the counterfactual label applies; (ii) the document remains coherent; and (iii) no unnecessary modifications are made". We believe that these instructions won't lead to data leakage because they do not induce annotators to make any specific modifications related to the OOD test.

3 Experiments

We evaluate the proposed CounterAL framework over two text classification tasks about sentiment analysis (SA) and natural language inference (NLI) to answer three research questions. **RQ1:** How effective is the proposed CounterAL as compared to conventional active learning methods? **RQ2:** How do the proposed acquisition strategy and training strategy influence the effectiveness of CounterAL? **RQ3:** How do the counterfactual samples affect the learning procedure?

3.1 Experimental Settings

Datasets. We use three benchmark datasets (one for sentiment analysis (Kaushik et al., 2020) and two for natural language inference (Kaushik et al., 2020; Nie et al., 2019)) with both factual samples and manually constructed counterfactual samples, which are denoted as **SA**, **NLI**, and **ANLI**, respectively. All three datasets contain train and test (IID test) sets. Factual samples in the train set form the unlabeled pool for active learning. The corresponding counterfactual samples are treated as the response of annotators for counterfactual sample construction. SA contains textual movie reviews from IMDB for sentiment analysis, and we instead adopt the tweet data with different distributions from SemEval-2017 Task 4 subtask A (Rosenthal et al., 2017) as the OOD test. NLI has factual sentence pairs for natural language inference, while ANLI is a set of sentence pairs intentionally edited to exhibit distribution shifts with NLI. We thus take the test set of NLI as the OOD test for ANLI, and randomly select a subset from ANLI as the OOD test of NLI.

To further prove the OOD generalization of CounterAL across various OOD tests with different distributions, we add one OOD test Amazon (Kaushik et al., 2021) for SA, where the reviews are from different fields. We also introduce NLI stress tests (Naik et al., 2018) with five different OOD tests for NLI and ANLI. The data statistics of all datasets are shown in the appendix A.

Baselines. We select five representative baselines: **Random**, **Entropy**, **BERT-KM** (Arthur and Vassilvitskii, 2006; Margatina et al., 2021), **BADGE** (Ash et al., 2020), and **CAL** (Margatina et al., 2021). **Random** selects samples from the unlabeled pool uniformly. **Entropy** is the most common uncertainty-based strategy that selects samples with the highest model predictive entropy. **BERT-KM** targets to get a diverse batch, which performs clustering in the representation space and selects samples near the centroid of each cluster. **BADGE** estimates both sample uncertainty and diversity from the gradient of samples. **CAL** is an acquisition strategy for NLP tasks. It selects samples whose predictive probability differs most from their nearest k neighbors in the labeled set.

Note that all baselines have the same acquisition batch size per round as the proposed CounterAL framework, *i.e.,* the total number of both factual and counterfactual samples is the same.

Implementation. We use RoBERTa (Liu et al., 2019) as the backbone model for all experiments and set the maximum length of text as 512. Following the setup in CAL, we apply AdamW (Loshchilov and Hutter, 2017) with a learning rate of 1e-5 and a batch size of 4 for the training at each round. The threshold for dropout (τ) in our training strategy is set as 0.1 for SA and NLI, and 0.03 for ANLI. The acquisition size (M) is 20/48/96 for SA/NLI/ANLI, respectively. We set the start pool size as M for all baselines (an acquisition strategy like CAL requires a start pool to acquire a meaningful batch), and adopt the cold start setting for CounterAL.[3] For a fair comparison, the start pool is abandoned in the rest rounds of training. For each method, we report the average classification accuracy on both IID and OOD tests over five runs with different initialization.

3.2 Performance Comparison (RQ1)

We first investigate the effectiveness of the proposed CounterAL framework through the performance comparison with baselines. Table 3 shows the IID and OOD performance of different methods on the three datasets. From the table, we have the following observations:

[3] The code and data used in this paper are available at https://github.com/xiangtanshi/CounterAL

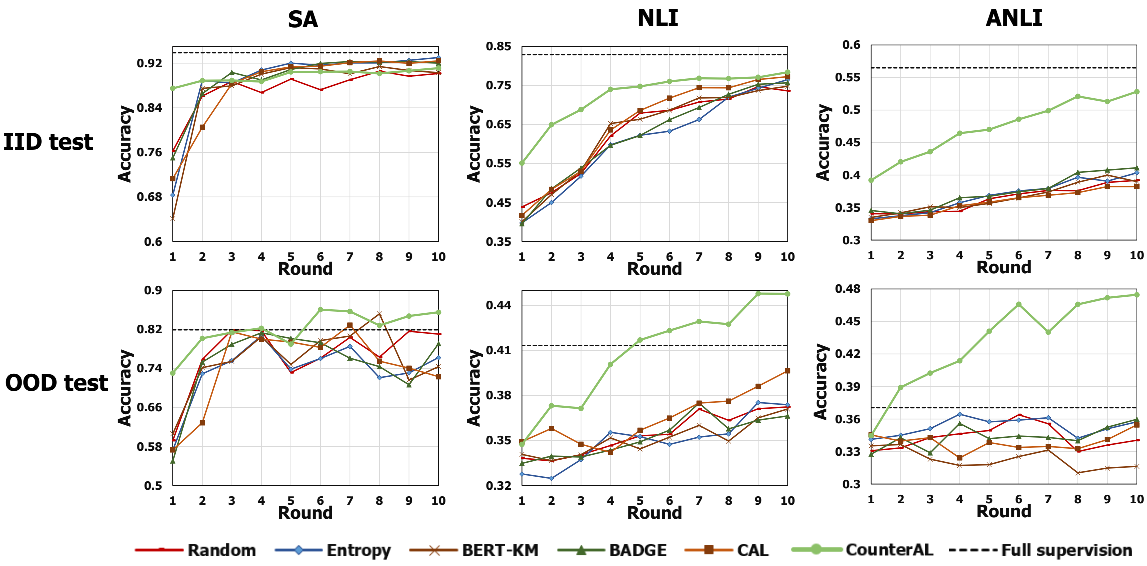


Figure 3: IID and OOD performance along the learning procedure of CounterAL and the baselines. Full supervision is the performance when the model is trained with the entire training set.

Table 3: Performance comparison between CounterAL framework and baselines on the IID and OOD tests of the three datasets *w.r.t.* classification accuracy (%). The second-best result is underlined in each column.

Method	SA			NLI			ANLI		
	IID	OOD	Ave	IID	OOD	Ave	IID	OOD	Ave
Random	92.18	82.75	87.46	76.33	36.66	56.50	39.92	33.07	36.49
Entropy	**93.33**	72.86	83.09	77.27	38.92	58.09	42.03	34.59	38.31
BERT-KM	91.78	76.68	84.23	76.61	37.46	57.03	38.46	33.05	35.75
BADGE	93.11	66.84	79.97	77.23	38.36	57.80	42.43	35.39	38.91
CAL	92.86	72.42	82.64	77.56	37.71	57.63	39.30	34.85	37.08
CounterAL	91.88	**86.21**	**89.04**	**78.87**	**45.02**	**61.95**	**52.35**	**51.25**	**51.80**

- Across the three datasets, the proposed CounterAL outperforms all baselines by a significant margin (absolute improvements of 4%∼15%) regarding the OOD test. More results on additional OOD tests (NLI stress tests for **NLI**, **ANLI** and Amazon for **SA**) in Table 4, 5 also show consistent improvements. The performance gain is attributed to the consideration of counterfactual samples in active learning, which validates the effectiveness and rationality of CounterAL for OOD generalization.
- As to the IID test, CounterAL also achieves the best performance on the NLI and ANLI dataset. This is because neural NLP models are apt to capture the easy-to-learn spurious correlations in the acquired training data (Kaushik et al., 2021; YU et al., 2022), and fail in the IID tests with correlation shifts. Instead, CounterAL includes counterfactual samples and discrepancy-aware dropout which force the model to focus on the features that causally affect the label, enhancing the OOD generalization ability. On the **SA**, CounterAL only achieves comparable IID performance with the baselines, which is consistent with previous work on counterfactual training (Teney et al., 2020). This might be because this IID test has similar correlations with the acquired training data.
- There is a clear gap between the IID and OOD performance of all baselines, which means that existing active learning methods face weak generalization problems on the two natural language processing tasks. This is consistent with previous findings on computer vision tasks (Krishnan et al., 2021), indicating that poor OOD generalization of active learning is a general problem.
- Remarkably, the OOD performance of Random is better than the other baselines in **SA**. It implies that delicately selected samples for training might hurt OOD generalization. We attribute the reason to overfitting, *i.e.,* complex acquisition strategies might make the model overfit some samples with strong correlations.

We further compare the learning process of CounterAL with the baselines. Figure 3 shows the round-wise performance of each acquisition function on the three datasets. As shown in Figure 3, the IID performance of CounterAL quickly achieves a relatively high level in early rounds and becomes stable with fewer rounds than baselines. It shows that counterfactual samples can accelerate the active learning process, reducing the acquisition amount. In addition, CounterAL shows consistent improvements in the OOD test during training, surpassing all baselines by a large margin. This validates the effectiveness of counterfactual samples

Table 4: Performance comparison between CounterAL and baselines on the NLI stress test.

	Methods	AT	LN	NG	SE	WO	Ave.
NLI	Random	15.10	47.33	40.98	48.34	49.01	40.15
	Entropy	13.92	52.49	43.55	50.83	51.52	42.46
	BERT-KM	11.60	48.08	41.06	49.79	48.73	39.85
	BADGE	18.01	52.93	45.08	52.30	50.70	43.80
	CAL	11.07	50.75	43.29	51.47	50.42	41.40
	CounterAL	**22.19**	**56.24**	**51.02**	**53.84**	**55.80**	**47.82**
ANLI	Random	85.00	33.96	35.50	33.95	34.69	44.62
	Entropy	**87.57**	37.59	35.43	33.36	35.42	45.87
	BERT-KM	71.31	33.00	35.02	33.03	34.28	41.33
	BADGE	67.61	36.33	35.27	36.50	35.21	42.18
	CAL	56.33	35.26	35.49	36.40	35.28	39.75
	CounterAL	79.14	**49.71**	**41.18**	**46.73**	**46.51**	**52.65**

Table 5: The OOD performance of baselines and CounterAL on Amazon for **SA**. We also report the standard error for each result.

Random	Entropy	BERT-KM	BADGE	CAL	CounterAL
88.67	88.35	86.68	84.01	86.04	**90.81**

in enhancing model's OOD generalization.

3.3 In-depth Analysis (RQ2)

Ablation Study. We then study the effectiveness of our proposed acquisition strategy, training strategy, and the KMeans clustering by comparing three variants of CounterAL: **1) CounterAL-KM**, which discards the KMeans sampling and directly selects top-$\frac{M}{K}$ samples *w.r.t.* variability. **2) CounterAL-A+BADGE, CounterAL-A+CAL**, which replace the proposed acquisition strategy in CounterAL with BADGE and CAL respectively. **3) CounterAL-T**, which discards the proposed training strategy, *i.e.,* updating model parameters normally during the iterations of CounterAL.

Table 6 shows the performance of these variants on **SA** and **NLI**. From the table, we have the following observations: 1) Across the two datasets, CounterAL outperforms its four variants in the OOD tests, which validates that all three components of CounterAL contribute to model's generalization ability. 2) CounterAL-KM performs worse than CounterAL, which shows that KMeans sampling can improve batch diversity that benefits CounterAL. This is consistent with the results in BEMPS (Tan et al., 2021). 3) CounterAL achieves better performance than CounterAL-A+BADGE and CounterAL-A+CAL, especially in the OOD test, hinting that the acquisition strategy plays a central role in acquiring informative samples to enhance models' OOD generalization ability.

We further explore how the acquisition strategy works by comparing four versions of variability for sample selection on **ANLI**: 1) Max-variability, which is defined in Equation 1; 2) Max-variability-opposite, which selects samples with the lowest value of Max-variability. 3) Ave-variability, which replaces the max operation in Equation 1 with average, *i.e.,* obtaining the mean value of variance across K classes. 4) Y-variability (Swayamdipta et al., 2020), which directly uses the variance of the ground-truth class as $v(x)$. We separately apply these strategies for sample selection on **ANLI**, and the detailed setups are provided in Appendix A.

Table 6: Performance comparison between CounterAL and its variants on the IID and OOD tests of the two NLI datasets *w.r.t.* classification accuracy (%).

Version	NLI			ANLI		
	IID	OOD	Ave	IID	OOD	Ave
CounterAL-KM	77.93	42.20	60.06	52.59	45.19	48.89
CounterAL-T	78.32	43.83	61.07	**52.79**	48.36	50.57
CounterAL-A+BADGE	**79.00**	42.34	60.67	51.6	40.90	46.25
CounterAL-A+CAL	77.88	42.67	60.27	51.44	40.16	45.80
CounterAL	78.87	**45.02**	**61.95**	52.35	**51.25**	**51.80**

Table 7: The performance of different sample selection strategies on ANLI. 300 factual-counterfactual pairs are selected for each strategy.

	Max-variability	Max-variability-opposite	Ave-variability	Y-variability
IID test	45.13	**47.21**	43.76	45.19
OOD test	**50.58**	34.56	44.98	48.11
Ave	**47.86**	40.88	44.37	46.65

Table 7 shows the IID and OOD performance of each strategy. We can find that: 1) max-variability outperforms the other three strategies in the OOD test, revealing that max-variability is more effective for OOD generalization. Besides, different from Y-variability, Max-variability does not utilize the ground-truth labels for sample selection, making it more suitable for CounterAL. 2) Max-variability-opposite has superior IID results but performs poorly in OOD tests (16% lower than Max-variability). This is possibly attributed to that Max-variability-opposite acquires many samples with spurious correlations that sacrifice OOD generalization for superior IID improvements.

3.4 Effect of Counterfactual Samples (RQ3)

We further investigate the effect of counterfactual samples by implementing two intuitive settings of combining counterfactual construction and active learning: pre-stage and post-stage methods introduced in Section 1. Table 8 shows their IID and OOD performance on **SA** and **NLI**. Comparing the results in Table 3 and 8, we find that: 1) under the post-stage setting, all three AL methods achieve better OOD performance than the vanilla versions, which further validates the rationality of incorporating counterfactual samples into active learning for OOD generalization. 2) Under the pre-stage setting, three AL methods achieve little OOD performance gain. The reason might be ignoring pairing information as they acquire and train factual and counterfactual samples indiscriminately. 3) CounterAL achieves better OOD performance than the three AL methods under two settings, validating the superiority of our proposed acquisition and training strategies over the intuitive methods.

We analyze the types of edits made by annotators when constructing counterfactual samples for the samples selected by CounterAL in the three datasets. In SA and NLI tasks, there are eight general types of editions that can be performed separately (Kaushik et al., 2020). We find that popular examples of modifications in sentiment analysis tasks include inserting or replacing modifiers, inserting phrases, and altering perspective. For natural language inference tasks, examples include modifying actions, substituting entities, and adding or removing negations and modifiers. We document the frequency of different types of modifications, along with specific examples and statistical results, in our project code.

We also explore whether asking humans to perform data augmentation (creating samples of the same class as factual samples) for the acquired batch can improve model's OOD generalization ability. We implement active learning on the **ANLI** datasets with BADGE and augment the acquired batch in each round. We evaluate the model after ten rounds of sample acquisition and training, and the OOD performance is 35.99%, which is only 0.6% higher than the normal BADGE. What's worse, its IID performance is 34.91%, which is 7.5% lower than the normal BADGE. These results support the superiority of counterfactual construction in text classification tasks.

Table 8: Performance of two settings of combining active learning and counterfactual construction.

		SA			NLI		
	Method	IID	OOD	Ave	IID	OOD	Ave
Pre-stage	Random	90.60	79.56	85.08	74.31	38.44	56.37
	Entropy	91.84	80.24	86.04	76.77	41.55	59.16
	BADGE	**92.14**	76.98	84.56	76.10	42.84	59.47
	CAL	91.70	76.72	84.21	77.24	40.15	58.69
Post-stage	Random	91.26	82.86	87.06	77.99	42.46	60.23
	Entropy	91.53	83.63	87.58	78.72	40.98	59.85
	BADGE	91.60	84.19	87.89	78.15	42.96	60.55
	CAL	91.43	84.72	88.07	78.52	43.20	60.86
CounterAL		91.88	**86.21**	**89.04**	**78.87**	**45.02**	**61.95**

4 Related Work

Uncertainty Measure of Active Learning. Active learning typically adopts uncertainty to estimate the informativeness of a sample (Ren et al., 2021), which mainly consists of two directions: estimating the uncertainty of a model's direct output (Wang et al., 2016; He et al., 2019) and the calibrated uncertainty with a group of models (Houlsby et al., 2011; Gal et al., 2017; Kirsch et al., 2019). Besides, some work estimates the uncertainty of a sample by comparing it with other related samples. (Gao et al., 2020) proposes to calculate the variance (*i.e.,* inconsistency) of predictions over a random set of data augmentation over the given sample. And (Margatina et al., 2021) proposes to select the contrastive samples by calculating the KL-divergence between a sample and its nearest neighbors in the labeled set. These variance-based methods only focus on the prediction of current model, which provides little information about whether model relies on spurious correlation to predict the sample. However, our acquisition strategy focus on the dynamic process of how the model changes its prediction along the training, and the variability recovers the existence of the spurious correlations that are informative for OOD generalization.

OOD Generalization in Active Learning. Recent studies on AL have demonstrated the performance drop in OOD tests (Krishnan et al., 2021). To alleviate this issue, SCAL (Krishnan et al., 2021) utilizes contrastive learning to improve models' robustness and JEPIG (Kirsch et al., 2021) detects test-time distribution shifts and uses the information gain for test–time prediction. Besides, on-line active learning (Lughofer, 2017) solves the OOD generalization problem by updating the unlabeled pool, hoping to make it more representative of the test samples. However, it is expensive to implement online active learning as it requires access to real-time samples. Hence, we consider the generalization of pool-based active learning.

Counterfactual Generation in NLP. In the field of text classification, recent studies have explored building new datasets by introducing counterfactual samples to improve model's generalization ability. (Nie et al., 2019) and (Kaushik et al., 2020) request the annotator to annotate counterfactual samples for existing samples, and the enhanced datasets are proven to significantly improve model's OOD generalization ability. Besides, (Gardner et al., 2020) proposes to construct contrast sets by annotating counterfactual samples for the test sets, and the contrast sets provide a better evaluation of model's decision boundary. Another line of work focuses on generating counterfactual samples with models. (Liu et al., 2022) resorts to GPT-3 (Brown et al., 2020) to create counterfactual samples with similar linguistic patterns to factual samples, and (Plyler et al., 2021) proposes a causal framework to create counterfactual samples for the sentiment analysis task. Overall, the quality of counterfactual samples created by humans is higher than the generated one (Kaushik et al., 2021), which reflects the value of human feedback.

Previous studies seldom consider transferring the counterfactual thinking ability from human to the model via active learning. By contrast, we propose counterfactual active learning, which improves model's OOD generalization ability with limited samples acquired in active learning.

5 Conclusion

We introduced a counterfactual active learning paradigm to improve the OOD generalization of active learning. Specifically, we developed novel acquisition and training strategies, which first acquire a diverse batch of informative factual-counterfactual pairs, and then capture the discrepancy between factual and counterfactual samples for model learning. Experiments on two classic NLP problems validate that the proposed strategies can significantly enhance the OOD performance.

In the future, we consider building models for automatic counterfactual sample generation to reduce the labor cost of annotators. In addition, a promising research direction is devising more effective training methods to leverage counterfactual samples. Furthermore, our method has higher potential in domains where human expertise and knowledge advantages are less captured by Large Language Models, and we will consider the application to more datasets in vertical domains

Limitations

The limitations of the paper are twofold. First, we need to train the annotators to be familiar with another annotation paradigm: creating counterfactual samples for the labeled factual samples. It is an additional cost for active learning although our user study has shown that annotating counterfactual samples has similar costs to labeling factual samples. Second, we require the annotators to manually find and edit the causal features, which is not effective enough. It can be improved by developing tools like generative models to automatically edit features for annotator judgment.

References

David Arthur and Sergei Vassilvitskii. 2006. k-means++: The advantages of careful seeding. Technical report, Stanford.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2020. Deep batch active learning by diverse, uncertain gradient lower bounds. *ICLR*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Fabrizio Carcillo, Yann-Aël Le Borgne, Olivier Caelen, and Gianluca Bontempi. 2017. An assessment of streaming active learning strategies for real-life credit card fraud detection. In *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 631–639. IEEE.

Melanie Ducoffe and Frederic Precioso. 2018. Adversarial active learning for deep networks: a margin based approach. *ICML*.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *ICML*, pages 1183–1192. PMLR.

Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan Ö Arık, Larry S Davis, and Tomas Pfister. 2020. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. In *ECCV*, pages 510–526. Springer.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. 2020. Evaluating models' local decision boundaries via contrast sets. *EMNLP*.

Abbas Ghaddar, Philippe Langlais, Mehdi Rezagholizadeh, and Ahmad Rashid. 2021. End-to-end self-debiasing framework for robust nlu training. *arXiv preprint arXiv:2109.02071*.

Tao He, Xiaoming Jin, Guiguang Ding, Lan Yi, and Chenggang Yan. 2019. Towards better uncertainty sampling: Active learning with multiple views for deep convolutional neural network. In *ICME*, pages 1360–1365. IEEE.

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

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2020. Learning the difference that makes a difference with counterfactually augmented data. *ICLR*.

Divyansh Kaushik, Amrith Setlur, Eduard Hovy, and Zachary C Lipton. 2021. Explaining the efficacy of counterfactually augmented data. *ICLR*.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30.

Andreas Kirsch, Tom Rainforth, and Yarin Gal. 2021. Test distribution-aware active learning: A principled approach against distribution shift and outliers. *arXiv preprint arXiv:2106.11719*.

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

Ranganath Krishnan, Alok Sinha, Nilesh Ahuja, Mahesh Subedar, Omesh Tickoo, and Ravi Iyer. 2021. Mitigating sampling bias and improving robustness in active learning. *arXiv preprint arXiv:2109.06321*.

Alisa Liu, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2022. Wanli: Worker and ai collaboration for natural language inference dataset creation. *arXiv preprint arXiv:2201.05955*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ICLR*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *ICLR*.

Edwin Lughofer. 2017. On-line active learning: A new paradigm to improve practical useability of data stream modeling methods. *Information Sciences*, 415:356–376.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active learning by acquiring contrastive examples. In *EMNLP*, pages 650–663.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *ACL*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. *ACL*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *ACL*.

Judea Pearl. 2009. *Causality*. Cambridge university press.

Mitchell Plyler, Michael Green, and Min Chi. 2021. Making a (counterfactual) difference one rationale at a time. *NeurIPS*, 34:28701–28713.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. 2021. A survey of deep active learning. *ACM Computing Surveys (CSUR)*, 54(9):1–40.

Neal J Roese. 1997. Counterfactual thinking. *Psychological bulletin*, 121(1):133.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the international workshop on semantic evaluation*, pages 502–518.

Axel Sauer and Andreas Geiger. 2021. Counterfactual generative networks. *ICLR*.

Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A Smith, and Yejin Choi. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. *EMNLP*.

Wei Tan, Lan Du, and Wray Buntine. 2021. Diversity enhanced active learning with strictly proper scoring rules. *NeurIPS*, 34:10906–10918.

Damien Teney, Ehsan Abbasnedjad, and Anton van den Hengel. 2020. Learning what makes a difference from counterfactual examples and gradient supervision. In *ECCV*, pages 580–599. Springer.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards debiasing nlu models from unknown biases. *EMNLP*.

Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. 2016. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600.

Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, Min Lin, and Tat-Seng Chua. 2022. Causal representation learning for out-of-distribution recommendation. In *Proceedings of the ACM Web Conference 2022*, pages 3562–3571.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *ACL*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *EMNLP*, pages 38–45.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet, Timothy J Hazen, and Alessandro Sordoni. 2019. Increasing robustness to spurious correlations using forgettable examples. *arXiv preprint arXiv:1911.03861*.

Sicheng YU, Jing JIANG, Hao ZHANG, Yulei NIU, Qianru SUN, and Lidong BING. 2022. Interventional training for out-of-distribution natural language understanding.

We provide more details and results about the datasets and experiments in the appendix. Section A provides information about the datasets and the experiment setup. Section B provides more results about model's OOD performance on **SA** and the standard error of results in 3. Section C describes how we implement the user-study on the twitter datasets to get the final counterfactual datasets.

A Detailed Experiment Settings

A.1 Dataset Information

The detailed statistics of the three datasets are presented in Table 9. Factual samples from different classes are balanced in the unlabeled pool for **SA** and **NLI**, but unbalanced for **ANLI**. Unbalanced class distribution in **ANLI** is because we randomly select a subset from the samples that have counterfactual samples in round 2 and 3 of the large adversarial NLI (Nie et al., 2019) (*i.e.,* A2 and A3 in adversarial NLI), which thus follows the unbalanced distribution in (Nie et al., 2019). Therefore, the results in the three datasets provide a straightforward comparison between baselines and CounterAL when dealing with balanced and unbalanced unlabeled pools.

As for the test sets, we randomly sample from the source datasets of **SA**, **NLI** and **ANLI** to get the IID test sets. We then give a detailed discussion of how we choose the OOD test sets for the three benchmark datasets. We adopt tweet and Amazon as the OOD test sets for **SA**, which follows the setting in (Kaushik et al., 2020). The difference between these two OOD datasets and **SA** is that they are reviews from different fields. The reason we choose tweets as the major OOD test set is that it additionally shows quite different linguistic properties (incomplete sentences and colloquial expression) from **SA** and is more challenging[4]. We choose the NLI stress tests as the additional OOD tests for **NLI** and **ANLI**. The NLI stress test is designed to test if the model captures the right linguistic pattern for prediction. It adds multiple different interferences such as spelling error, word overlap, and length control to exhibit distribution shifts. We randomly select 3000 samples from each of the five subtasks in NLI stress test as our additional OOD tests.

The scales of the IID and OOD test sets are comparable, and samples of different classes are balanced for each test set. This promises that the test results will not be influenced by the class bias or the size of the test sets.

Table 9: Dataset statistics.

Datasets	**SA**	**NLI**	**ANLI**
Class number	2	3	3
Unlabeled pool	1707	1666	4935
Class ratio	1:1	1:1:1	5:3:2
Counterfactual samples	1707	1666×2	4935×2
IID test set	2000	2000	2400
OOD test set	1400	2400	2000

A.2 Implementation

We now introduce how we implement the baselines and CounterAL to acquire a batch from $\mathcal{X}_U$. **Random** shuffles the samples in $\mathcal{X}_U$ and randomly selects M samples. **Entropy** calculates the predicted probability for each sample in $\mathcal{X}_U$ and selects M samples with the highest probability. **BERT-KM** (Arthur and Vassilvitskii, 2006; Ash et al., 2020) clusters the samples into M groups and selects one sample that is closest to the centroid from each group. **BADGE** (Ash et al., 2020) is a parameter-free method which acquires a random subset using the k-MEANS++ seeding algorithms (Arthur and Vassilvitskii, 2006) in the gradient space. **CAL** (Margatina et al., 2021) calclutes the average KL-divergence between each unlabeled sample and its K nearest samples in the labeled set, then selects M samples with the highest mean divergence. Follow the setup in **CAL**, we set K as 10. CounterAL requires to store historical models and recall them to calculate the variability for samples in $\mathcal{X}_U$. In order to calculate a meaningful variability after the first round, we save three checkpoints in the first round when model's predictive accuracy first reaches 70%, 90%, and 100% separately. To save the cost, we store the predicted probability for each sample once it is calculated. Thereafter, CounterAL first selects $c \times M$ samples with the highest variability from $\mathcal{X}_U$, then applies KMeans-

[4] For instance, samples from the tweet dataset follows the style of *“New on @Twitter . Big fan of @NICKIMINAJ and @ArianaGrande #ArianaGrande #NickiMinaj #Barbies #Barbz #Arianators. . . "*

Table 10: Performance comparison between models trained on the entire unlabeled pool. The results related to the model of baselines and CounterAL are put down below for easy comparison.

Method	SA			NLI			ANLI		
	IID	OOD	Ave	IID	OOD	Ave	IID	OOD	Ave
	93.90±0.12	81.94±2.05	87.92	82.94±0.17	41.33±1.32	62.13	56.51±0.73	37.05±1.14	46.78
Random	92.18±0.34	82.75±1.95	87.46	76.33±0.65	36.66±0.80	56.49	39.92±0.48	33.07±1.37	36.49
Entropy	**93.33±0.10**	72.86±2.83	83.10	77.27±1.15	38.93±1.41	58.10	42.02±1.37	34.59±2.47	38.31
BERT-KM	91.78±0.33	76.68±3.33	84.23	76.61±1.57	37.46±1.06	57.03	38.45±1.54	33.05±1.72	35.75
BADGE	93.11±0.41	66.84±1.90	79.97	77.23±1.18	38.36±1.26	57.79	42.43±1.09	35.39±1.11	38.91
CAL	92.86±0.24	72.42±3.27	82.64	77.56±0.61	37.70±0.92	57.63	39.30±1.13	34.85±2.37	37.07
CounterAL	91.88±0.31	**86.21±1.21**	**88.60**	**78.87±0.35**	**45.02±1.06**	**61.94**	**52.35±1.32**	**51.25±2.02**	**51.80**

clustering to select a diverse batch of size $\frac{M}{K}$. We recommend setting c as 4 for our experiments.

As for the training strategy, we tune the dropout threshold (τ) so that the ratio of masked features approximates 50%. For the initialized model in the first round, τ is 0 to make sure not all features are masked because the initialized model extracts similar features for both factual and counterfactual samples. We choose to apply the dropout after multiple training epochs until the model is able to correctly classify over 80% of the acquired samples.

The Setup of Table 7 The test consists of three steps: (i) we train a model on the training set of **ANLI** for multiple epochs; (ii) we calculate the value of the specific variability we want to test according to the historical models from step (i); (iii) we apply KMeans clustering to choose a diverse batch of 300 samples from the top-600 samples with the highest variability. Then we train a new model with the acquired 300 samples and their counterfactual counterparts for five times with different initialization. The OOD performance of the new model reflects the quality of the samples that are selected by the corresponding variability.

A.3 More Discussions

Under Multi-class. Given a factual sample of K-way classification, CounterAL constructs $K-1$ counterfactual samples. To reduce the cost of counterfactual construction, we can restrict the construction to informative counterfactual classes of the sample. In particular, we can sort candidate classes in the descending order of prediction probability given by the model at current round. In this way, we only consider the top-ranked classes with sufficient probability as counterfactual classes. As deep neural networks typically give highly skewed probability distributions (Kendall and Gal, 2017), the number of considered candidate classes will remain small when K is large.

Table 11: The contribution of different numbers of factual-counterfactual pairs to OOD generalization on NLI. 160 and 240 are the number of pairs for training.

		IID Test		OOD Test	
Number of training pairs		160	240	160	240
Distance	**Small**	76.28	78.58	45.90	47.15
	Large	79.33	**80.15**	41.04	40.99

Compute Resources. All the experiments were run on 3 GeForce RTX 3090 GPUs.

B Model Performance

B.1 Extra OOD Performance

We investigate how the distance between factual and counterfactual samples affects their out-of-distribution (OOD) generalization contribution to the model. The result is shown in Table 11, which indicates that factual-counterfactual pairs with small distances play a more significant role in improving the model's generalization performance as they locate near the decision boundary (The model is currently unable to distinguish them well), providing better constraints during training.

B.2 OOD Results of Reweighting Methods

We explore the relationship between the generalization ability of reweighting methods and the size of the training set. Following the setup in the original paper, We choose MNLI (Williams et al., 2017) as the training set and HANS (McCoy et al., 2019)) as the OOD test set. We randomly select a subset from MNLI for model training. The results are shown in Table 12, from which we find that all the methods are not effective for small training sets, hence do not fit active learning.

Table 12: The OOD performance of the vanilla model and several SOTA reweighting methods on **HANS**. They are trained with the randomly selected subset from MNLI.

subset size	1k	3k	10k	390.27k
BERT-base (Wolf et al., 2020)	50.35	50.00	49.79	61.50
Reweighting (UB) (Utama et al., 2020)	49.46	49.95	49.75	69.70
Self-Debiasing (Ghaddar et al., 2021)	50.10	49.95	49.85	71.20
Forgetabble Examples (Yaghoobzadeh et al., 2019)	49.26	49.55	49.65	70.50

B.3 Extra Results with the Standard Error

We train a RoBERTa model on the entire unlabeled pool (noted as Full supervision in Figure 3). The results are reported in Table 10, where we also report the standard error for results in Table 3. We have the following observations: all the baselines achieve similar IID performance on both **SA** and **NLI** as compared to the upper bound model, *i.e.*, the model trained by using the entire unlabeled pool and counterfactual samples. However, there is still a certain gap compared to the upper bound model on **ANLI**. This is because the redundancy of **ANLI** is relatively small and the distribution of **ANLI** is not balanced. Meanwhile, their OOD performance is much worse. By contrast, CounterAL achieves comparable IID and OOD performance on all three datasets compared to the upper bound.

C User Study

C.1 Implementation Setup

We aim to empirically explore whether the annotation cost of creating counterfactual samples is much more expensive compared to labeling factual samples. We choose tweet data from SemEval-2017 task 4 subtask A (Rosenthal et al., 2017) to conduct the user study for the following considerations: 1) It is a topic-free 3-way classification task which is challenging for deep models and a counterfactual dataset for it would be valuable [5]; 2) Sentiment analysis on tweet data is a meaningful task that has received much attention.

Initially, we used the variability-based acquisition strategy to select 300 samples from the original training set. We then assigned the task of labeling these samples to three volunteers, with each responsible for 100 samples[6]. The volunteers were paid based on a standard rate of 30 dollars per hour for their work. The entire process was carried out in four steps:

Table 13: User study on Tweet data: the average time cost and accuracy of labeling factual samples and annotating counterfactual samples. s stands for seconds.

Annotator index	T1	T2	T3	Accuracy
1	36s	36s	29s	82%
2	92s	66s	50s	88%
3	36s	42s	53s	85%
Ave	54s	48s	44s	-

- First, all annotators are instructed on the same labeling rules for Positive/Neutral/Negative. Following this, we provide an explanation and training on how to modify causal features to generate the corresponding counterfactual samples through a few examples.
- Second, to avoid potential ethical concerns during the labeling process, we explicitly instruct the annotators to discard any samples related to sensitive issues such as war and politics.
- Third, we ask each annotator to record three different timestamps: 1) **T1**, the time taken for labeling 50 factual samples; 2) **T2**, the time taken for annotating 50 × 2 counterfactual samples for the same set of samples annotated in step 1; 3) **T3**, while annotating the remaining 50 factual samples, we instruct the annotator to create counterfactual samples for each factual sample immediately after labeling it and record the total amount of time taken to annotate 50 factual-counterfactual pairs (150 samples in total).
- Fourth, we calculate the labeling accuracy for each annotator and remove any mislabeled samples. The results are presented in Table 13.

C.2 Conclusion

According to the result and the feedback from the annotators, we have the following conclusions:

- Our results indicate that the average time required to annotate a counterfactual sample is lower than that for labeling a factual sample, indicating that identifying causal features is more time-consuming than modifying them. In addition, our study shows that annotating counterfactual samples immediately after labeling their corresponding factual samples improves efficiency.
- Throughout the process, the most common questions posed by annotators pertained to labeling certain special factual samples, often involving complex metaphors in English. In addition, one annotator sought clarification on whether it was permissible to make significant modifications to such samples, and we confirmed that it was al-

[5] There is no such dataset available yet, and we will release our constructed dataset along with our code.

[6] All annotators had similar levels of English proficiency and academic qualifications.

lowed.

- We observed that most of the mislabeled samples were attributed to biases in the original labels. For instance, samples involving Cristiano Ronaldo were often labeled as positive, whereas replacing his name with other characters resulted in neutral labels for the same sentences. In such cases, we respected the decisions of the annotators to retain or discard these samples.
- The annotators reported having difficulty constructing counterfactual samples for a particular type of sample: those that were not clearly positive, making it challenging to determine whether to label them as positive or neutral. Moreover, it was difficult to create a neutral counterfactual sample for a positive factual sample, such as "Saint Valentine's Day.".
- Overall, the annotators reported that the most challenging aspect of labeling was the lack of clear guidelines for distinguishing between Positive and Neutral categories. This difficulty stems from the annotation setup used in the original paper, where the majority label among five different annotators was selected as the final label. As such, there was often no clear rule or guideline for selecting a definitive label.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Left blank.

☑ A2. Did you discuss any potential risks of your work?
Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

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

B ☑ Did you use or create scientific artifacts?

Left blank.

☑ B1. Did you cite the creators of artifacts you used?
Left blank.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☑ 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)?
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?

Left blank.

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

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?
Left blank.

☑ 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?
Left blank.

□ 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.?
Left blank.

□ 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)?
Not applicable. Left blank.

☑ 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?
Left blank.

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.