ALVIN: Active Learning Via INterpolation

Michalis Korakakis[1,3] **Andreas Vlachos**[1] **Adrian Weller**[2,3]
[1]Department of Computer Science and Technology, University of Cambridge
[2]Department of Engineering, University of Cambridge
[3]The Alan Turing Institute
{mk2008,av308,aw665}@cam.ac.uk

Abstract

Active Learning aims to minimize annotation effort by selecting the most useful instances from a pool of unlabeled data. However, typical active learning methods overlook the presence of distinct example groups within a class, whose prevalence may vary, e.g., in occupation classification datasets certain demographics are disproportionately represented in specific classes. This oversight causes models to rely on shortcuts for predictions, i.e., spurious correlations between input attributes and labels occurring in well-represented groups. To address this issue, we propose Active Learning Via INterpolation (ALVIN), which conducts intra-class interpolations between examples from under-represented and well-represented groups to create anchors, i.e., artificial points situated between the example groups in the representation space. By selecting instances close to the anchors for annotation, ALVIN identifies informative examples exposing the model to regions of the representation space that counteract the influence of shortcuts. Crucially, since the model considers these examples to be of high certainty, they are likely to be ignored by typical active learning methods. Experimental results on six datasets encompassing sentiment analysis, natural language inference, and paraphrase detection demonstrate that ALVIN outperforms state-of-the-art active learning methods in both in-distribution and out-of-distribution generalization.

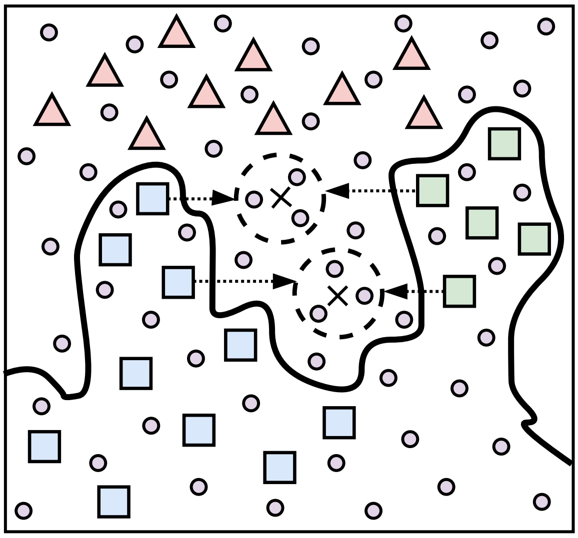


Figure 1: Illustration of ALVIN applied to a binary classification task. □ indicates well-represented, labeled examples in Class A, □ indicates under-represented, labeled examples in Class A, △ indicates labeled examples in Class B, ○ indicates unlabeled instances, and × indicates the anchors created via intra-class interpolations between under-represented and well-represented examples. Unlike typical active learning methods, ALVIN prioritizes high-certainty instances that integrate representations from different example groups at varied proportions. This approach enables ALVIN to adjust the model's decision boundary and mitigate its reliance on shortcuts.

1 Introduction

Despite the remarkable zero-shot and few-shot learning capabilities of large language models (LLMs) (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023, *inter alia*), supervised fine-tuning remains a critical component of model development (Yuan et al., 2023; Mosbach et al., 2023; Bai et al., 2023). Collecting high-quality labeled data is, nonetheless, time-consuming and labor-intensive (Tan et al., 2024). To address this annotation bottleneck, active learning (AL) seeks to select the most useful instances from a pool of unlabeled data, thereby maximizing model performance subject to an annotation budget (Settles, 2009).

However, datasets commonly used for model fine-tuning often contain shortcuts (Gururangan et al., 2018; McCoy et al., 2019; Wang and Culotta, 2020), i.e., spurious correlations between input attributes and labels present in a large number of examples (Geirhos et al., 2020). For example, in

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 22715–22728
November 12-16, 2024 ©2024 Association for Computational Linguistics

occupation classification datasets, many examples exhibit patterns that incorrectly associate certain demographics, such as race and gender, with specific occupations (Borkan et al., 2019). Consequently, models exploiting shortcuts achieve high performance on well-represented example groups, but fail on under-represented groups where shortcuts do not apply (Tu et al., 2020). This issue is particularly prominent in out-of-distribution settings, where under-represented groups can become more prevalent due to distribution shifts (Koh et al., 2021). By neglecting the presence of these distinct example groups in the training data, AL methods amplify the prevalence of well-represented groups, thereby exacerbating shortcut learning (Gudovskiy et al., 2020; Deng et al., 2023).

Motivated by these shortcomings, we introduce Active Learning Via INterpolation (ALVIN). The key idea behind ALVIN is to leverage interpolations between example groups to explore the representation space. Specifically, we identify unlabeled instances for annotation by assessing their proximity to anchors, i.e., artificial points in the representation space created through intra-class interpolations between under-represented and well-represented examples. Intuitively, ALVIN selects informative instances with features distinct from those prevalent in well-represented groups, helping the model avoid reliance on shortcuts. Importantly, because these instances are deemed high certainty by the model, they are often overlooked by typical AL methods.

We conduct experiments on six datasets spanning sentiment analysis, natural language inference, and paraphrase detection. Our results demonstrate that ALVIN consistently improves out-of-distribution generalization compared to several state-of-the-art AL methods, across different dataset acquisition sizes, while also maintaining high in-distribution performance.

We analyze ALVIN to gain deeper insights into its performance improvements. First, we examine the unlabeled examples identified by ALVIN, showcasing its ability to select diverse, high-certainty instances while avoiding outliers that could negatively impact performance. Next, through several ablation studies, we demonstrate the advantages of our interpolation strategy compared to other interpolation-based AL methods. Finally, we explore the impact of hyper-parameters on performance and assess the computational runtime required to select instances for annotation.

2 Active Learning Via INterpolation

2.1 Preliminaries

We consider the typical pool-based active learning (AL) scenario (Lewis and Gale, 1994), in which an initial set of labeled instances $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathcal{X}$ is the input and $y_i \in \{1, 2, \ldots, C\}$ is the corresponding label, along with a pool of unlabeled instances $\mathcal{U} = \{x_j\}_{j=1}^{M}$, where $N \ll M$. In each AL round, we query an annotation batch B comprised of b instances from $\mathcal{U}$ to be annotated and added to $\mathcal{L}$. Then $\mathcal{L}$ is used to train a model $f_\theta : \mathcal{X} \to \mathcal{Y}$ parameterized by θ. The model f_θ consists of an encoder $f_{\text{enc}} : \mathcal{X} \to \mathcal{Z}$ mapping an input x_i to a representation z_i, and a classifier $f_{\text{cls}} : \mathcal{Z} \to \mathcal{Y}$ which outputs a softmax probability over the labels based on z_i. The AL process continues until the annotation budget is exhausted or a satisfactory model performance level is reached.

Following Sagawa et al. (2019), we further assume that the training dataset contains distinct groups of instances within some classes. Some of these groups are well-represented and strongly associated with labels, e.g., high word overlap and "entailment" in natural language inference (NLI) datasets (McCoy et al., 2019), while others are under-represented, e.g., negation in the hypothesis and "entailment" (Gururangan et al., 2018). We refer to the instances belonging to the well-represented groups associated with a particular class as majority instances g_{maj} of said class, and the rest as minority instances g_{min}.[1]

Models often rely on shortcuts found in majority instances to make predictions (Puli et al., 2023), a dependency that becomes problematic when distribution shifts at test time increase the prevalence of minority examples, resulting in poor out-of-distribution generalization (Koh et al., 2021). This issue is further exacerbated in AL, where typical methods like uncertainty sampling (Lewis and Gale, 1994), select repetitive high uncertainty majority instances (Deng et al., 2023). To counter shortcut learning, it is crucial for the model to be exposed to instances whose patterns deviate from those prevalent in majority examples (Korakakis and Vlachos, 2023).

[1]Note that some instances can be majority for a particular class, and other instances exhibit the same patterns can be minority for a different class e.g., NLI instances containing negation in the hypothesis are majority for the "contradiction" class, but minority for the "entailment" class.

Algorithm 1 Active Learning Via INterpolation (ALVIN)

Input: Training dataset $\mathcal{L}$, unlabeled pool $\mathcal{U}$, model $f_\theta = \{f_{\text{enc}}, f_{\text{cls}}\}$, annotation batch size b, number of anchors K, shape parameter α of Beta distribution

1: $\mathcal{I} = \emptyset$
2: $g_{\text{min}}, g_{\text{maj}} = \textsc{InferMinMaj}(f_\theta, \mathcal{L})$
3: **for** $c \in \mathcal{C}$ **do**
4:     Sample $\mathcal{L}_c^{\text{min}}, \mathcal{L}_c^{\text{maj}} \sim \mathcal{L}$
5:     **for** $(x_i, y_i) \in \mathcal{L}_c^{\text{min}}$ **do**
6:         Sample $(x_j, y_j) \sim g_c^{\text{maj}}$
7:         **for** k in K **do** ▷ generate multiple anchors
8:             Sample $\lambda \sim \text{Beta}(\alpha, \alpha)$
9:             $a_{i,j}^k = \lambda f_{\text{enc}}(x_i) + (1-\lambda) f_{\text{enc}}(x_j)$
10:            $\mathcal{I} \leftarrow \mathcal{I} \cup \underset{x \in \mathcal{U}}{\text{Top-k KNN}}\big(a_{i,j}^k, \mathcal{U}\big)$ ▷ select k nearest neighbors of anchor from $\mathcal{U}$
11: $B = \underset{x \in \mathcal{I}}{\text{argmax}} - \sum_{i=1}^{C} f_{\text{cls}}(f_{\text{enc}}(x))_i \log f_{\text{cls}}(f_{\text{enc}}(x))_i, |B| = b$ ▷ select top-b instances via uncertainty

2.2 Algorithm

We hypothesize that the properties of the representation space are crucial for identifying unlabeled instances capable of mitigating shortcut learning. Specifically, the reliance on shortcuts for predictions creates a spurious decision boundary, incorrectly separating minority and majority examples within the same class. Thus, our goal is to select informative instances that will prompt the model to adjust its decision boundary, thereby correcting its reliance on shortcut features. To achieve this, ALVIN employs intra-class interpolations between minority and majority instances to create anchors. These anchors facilitate the exploration of diverse feature combinations within the representation space, enabling the identification of unlabeled instances that integrate representations from different example groups at varied proportions. However, because these instances exhibit high certainty, they are typically overlooked by existing AL methods, e.g., a model will confidently label an "entailment" instance with negation in NLI as "contradiction." The overall procedure of ALVIN is detailed in Algorithm 1 for an AL round.

Inferring Minority/Majority Examples At the beginning of each AL round, we first identify the minority and majority examples within each class in the training dataset (line 2). We are motivated by the observation that the existence of shortcuts within the majority examples causes a discrepancy in training dynamics, leading the model to fit majority examples faster than minority ones, and resulting in a spurious decision boundary (Shah et al., 2020; Tu et al., 2020; Pezeshki et al., 2021). Thus, we infer the example groups by monitoring the frequency with which the model incorrectly predicts an example (Toneva et al., 2019; Swayamdipta et al., 2020; Yaghoobzadeh et al., 2021). Specifically, we classify an example x_i as minority if (1) the model's predictions switch between correct to incorrect at least once during training, i.e., $\text{acc}_{x_i}^t > \text{acc}_{x_i}^{t+1}$, where $\text{acc}_{x_i}^t = \mathbb{1}_{\hat{y}_i^t = y_i}$ indicates that the example x_i is correctly classified at time step t, or (2) the example is consistently misclassified by the model throughout training, i.e., $\forall t \in \{1, 2, \ldots, T\}, \quad \text{acc}_{x_i}^t = 0$ where T is the total number of training epochs. Conversely, all other examples that do not meet these criteria are classified as majority examples.

Anchor Creation After identifying the minority and majority examples within each class, we then proceed to create anchors to explore the representation space between these example groups. In particular, for each class c in $\mathcal{C}$, we initially sample $\mathcal{L}_c^{\text{min}}$ and $\mathcal{L}_c^{\text{maj}}$ (line 4), where $|\mathcal{L}_c^{\text{min}}| = |\mathcal{L}_c^{\text{maj}}| \ll N$. Next, for every minority instance in $\mathcal{L}_c^{\text{min}}$ (line 5) we randomly sample a majority instance from $\mathcal{L}_c^{\text{maj}}$ (line 6), and interpolate their representations to create the anchor $a_{i,j}$ (line 9):

$$a_{i,j} = \lambda f_{\text{enc}}(x_i) + (1-\lambda) f_{\text{enc}}(x_j), \tag{1}$$

where the interpolation ratio $\lambda \in [0, 1]$ is sampled from a Beta distribution $\text{Beta}(\alpha, \alpha)$. By adjusting the parameter α of this distribution, we can control where the anchors lie in the representation space relative to minority or majority instances. Intuitively, when λ is closer to 0, the anchor $a_{i,j}$ is predominantly influenced by the minority instance x_i; conversely, as λ approaches 1, $a_{i,j}$ increasingly resembles the representation of majority instance x_j.

We generate K anchors for each minority-majority pair (line 7). This process enables us to create anchors that incorporate varied feature combinations, thus allowing for a comprehensive exploration of the representation space between minority and majority examples.

Example Selection After constructing the anchors, we use K-Nearest-Neighbors (KNN) to identify similar unlabeled examples $x_u \in \mathcal{U}$ to an anchor in the representation space (line 10).[2] We repeat this process for each anchor across all classes. Finally, we select for annotation the top-b unlabeled instances with the highest uncertainty (Lewis and Gale, 1994) (line 11). This approach maintains the advantages of uncertainty-based instance selection, while counteracting its tendency to facilitate shortcut learning by selecting a subset of unlabeled instances that mitigate this phenomenon.

3 Experimental Setup

Datasets We conduct experiments on six datasets across sentiment analysis, natural language inference, and paraphrase detection. In line with previous works in AL (Yuan et al., 2020; Margatina et al., 2021; Deng et al., 2023), we use SA (Kaushik et al., 2020), NLI (Kaushik et al., 2020), ANLI (Nie et al., 2020), SST-2 (Socher et al., 2013), IMDB (Maas et al., 2011), and QQP (Chen et al., 2017). To assess out-of-distribution (OOD) generalization we use SemEval-2017 Task 4 (Rosenthal et al., 2017) for SA, ANLI for NLI, and NLI for ANLI, IMDB for SST-2, SST-2 for IMDB, and TwitterPPDB (Lan et al., 2017) for QQP. Validation and test splits are used as described in Margatina et al. (2021) for IMDB, SST-2, and QQP, and Deng et al. (2023) for SA, ANLI, and NLI.

Comparisons We compare ALVIN with several baseline and state-of-the-art AL methods:

- **Random** samples instances uniformly at random.
- **Uncertainty** (Lewis and Gale, 1994) acquires annotations for unlabeled instances with the highest predictive entropy according to the model.
- **Batch Active learning by Diverse Gradient Embeddings (BADGE)** (Ash et al., 2020) selects unlabeled instances by applying the K-means++ (Arthur and Vassilvitskii, 2007) clustering algorithm on the gradients of the predicted class with respect to the model's last layer.
- **BERT-KM** (Yuan et al., 2020) clusters unlabeled instances within the representation space of a BERT (Devlin et al., 2019) model using k-means, then selects for annotation those instances that are closest to the center of each cluster.
- **Contrastive Active Learning (CAL)** (Margatina et al., 2021) selects unlabeled instances that, according to the model, diverge maximally from their nearest labeled neighbors.
- **Active Learning by Feature Mixing (ALFA-Mix)** (Parvaneh et al., 2022) conducts interpolations between unlabeled instances and anchors, i.e., the average embeddings of the labeled examples for each class, and then selects unlabeled instances whose interpolations have different predictions compared to the anchors.

Implementation Details We use the HuggingFace (Wolf et al., 2020) implementation of BERT-base (Devlin et al., 2019) for our experiments. Following Margatina et al. (2021), we set the annotation budget at 10% of the unlabeled pool $\mathcal{U}$, initialize the labeled set at 0.1% of $\mathcal{U}$, and the annotation batch size b at 50. We train BERT-base models with a batch size of 16, learning rate of $2e-5$, using the AdamW (Loshchilov and Hutter, 2019) optimizer with epsilon set to $1e-8$. For ALVIN, we set K to 15, α to 2, and use the CLS token from the final layer to obtain representations and conduct interpolations. For other AL methods, we follow the same hyper-parameter tuning methods mentioned in their original papers. Each experiment is repeated three times with different random seeds, and we report the mean accuracy scores and standard deviations.

4 Results

4.1 Main Results

Table 1 presents the main experimental results across the six datasets. Overall, we observe a considerable decline in OOD performance across all AL methods. ALVIN consistently outperforms all other AL methods in both in-distribution and out-of-distribution generalization. ALFA-Mix, CAL, and Uncertainty also show competitive perfor-

[2] Our distance metric is the Euclidean distance.

Data	Acq. Data	Random ID	Random OOD	Uncertainty ID	Uncertainty OOD	BADGE ID	BADGE OOD	BERT-KM ID	BERT-KM OOD	CAL ID	CAL OOD	ALFA-Mix ID	ALFA-Mix OOD	ALVIN ID	ALVIN OOD
SA	1%	$\mathbf{78.9}_{\pm0.2}$	$59.4_{\pm1.8}$	$69.7_{\pm0.2}$	$57.9_{\pm2.7}$	$74.6_{\pm0.5}$	$56.2_{\pm1.9}$	$66.4_{\pm0.4}$	$60.5_{\pm3.5}$	$72.4_{\pm0.2}$	$57.8_{\pm3.5}$	$73.9_{\pm0.5}$	$58.0_{\pm2.5}$	$77.9_{\pm0.7}$	$\underline{61.5}_{\pm0.5}$
	5%	$86.9_{\pm0.1}$	$73.9_{\pm2.2}$	$\mathbf{90.8}_{\pm0.3}$	$74.4_{\pm3.2}$	$88.9_{\pm0.6}$	$79.7_{\pm2.1}$	$90.2_{\pm0.3}$	$75.6_{\pm3.4}$	$89.4_{\pm0.3}$	$79.3_{\pm3.1}$	$89.7_{\pm0.9}$	$79.8_{\pm3.2}$	$\mathbf{90.8}_{\pm1.0}$	$\underline{82.2}_{\pm1.2}$
	10%	$88.3_{\pm0.2}$	$81.1_{\pm1.9}$	$91.1_{\pm0.3}$	$78.2_{\pm3.4}$	$90.2_{\pm0.4}$	$78.3_{\pm1.8}$	$88.3_{\pm0.5}$	$75.9_{\pm2.8}$	$90.5_{\pm0.2}$	$73.0_{\pm2.3}$	$90.5_{\pm0.7}$	$78.4_{\pm2.9}$	$\mathbf{91.8}_{\pm1.3}$	$\underline{84.1}_{\pm0.9}$
NLI	1%	$\mathbf{44.7}_{\pm0.6}$	$34.2_{\pm0.9}$	$41.2_{\pm1.2}$	$33.2_{\pm1.7}$	$41.3_{\pm1.3}$	$33.8_{\pm1.2}$	$42.4_{\pm1.6}$	$34.7_{\pm1.0}$	$43.3_{\pm0.4}$	$35.3_{\pm0.7}$	$42.8_{\pm1.4}$	$34.6_{\pm2.2}$	$43.4_{\pm0.8}$	$\underline{35.7}_{\pm1.5}$
	5%	$67.1_{\pm0.9}$	$35.8_{\pm1.1}$	$63.9_{\pm1.4}$	$35.7_{\pm1.9}$	$63.7_{\pm1.2}$	$35.0_{\pm1.4}$	$65.8_{\pm1.8}$	$34.6_{\pm1.2}$	$67.8_{\pm0.4}$	$36.0_{\pm1.2}$	$67.8_{\pm1.7}$	$36.3_{\pm1.9}$	$\mathbf{69.7}_{\pm1.1}$	$\underline{38.9}_{\pm0.7}$
	10%	$72.9_{\pm0.6}$	$37.9_{\pm0.8}$	$76.2_{\pm1.0}$	$37.9_{\pm1.3}$	$76.1_{\pm1.4}$	$37.0_{\pm1.4}$	$73.1_{\pm1.5}$	$37.6_{\pm1.2}$	$77.6_{\pm0.6}$	$39.9_{\pm0.8}$	$77.7_{\pm2.1}$	$40.1_{\pm3.1}$	$\mathbf{78.1}_{\pm1.1}$	$\underline{42.9}_{\pm1.5}$
ANLI	1%	$34.1_{\pm0.4}$	$33.1_{\pm1.3}$	$33.1_{\pm1.4}$	$34.1_{\pm2.4}$	$\mathbf{34.8}_{\pm1.4}$	$32.8_{\pm1.7}$	$33.4_{\pm1.2}$	$33.3_{\pm1.3}$	$33.0_{\pm1.1}$	$\underline{34.5}_{\pm2.4}$	$33.3_{\pm1.2}$	$33.7_{\pm1.7}$	$34.2_{\pm0.5}$	$33.8_{\pm0.9}$
	5%	$36.4_{\pm0.3}$	$35.1_{\pm0.9}$	$37.3_{\pm1.4}$	$35.9_{\pm1.9}$	$37.3_{\pm1.5}$	$34.6_{\pm1.7}$	$36.6_{\pm1.2}$	$32.4_{\pm1.2}$	$36.2_{\pm1.3}$	$34.1_{\pm1.9}$	$\mathbf{37.8}_{\pm1.8}$	$34.7_{\pm2.4}$	$37.4_{\pm0.9}$	$\underline{37.9}_{\pm0.6}$
	10%	$38.9_{\pm0.4}$	$33.5_{\pm1.2}$	$39.9_{\pm1.7}$	$35.9_{\pm2.7}$	$41.0_{\pm1.2}$	$36.0_{\pm1.5}$	$40.1_{\pm1.3}$	$31.5_{\pm1.1}$	$38.3_{\pm1.2}$	$35.2_{\pm2.2}$	$38.3_{\pm1.8}$	$36.1_{\pm2.3}$	$\mathbf{42.6}_{\pm1.0}$	$\underline{39.2}_{\pm1.3}$
SST-2	1%	$84.0_{\pm0.5}$	$69.3_{\pm0.7}$	$84.6_{\pm0.8}$	$68.6_{\pm1.5}$	$84.6_{\pm0.6}$	$68.6_{\pm1.1}$	$84.7_{\pm0.9}$	$68.6_{\pm1.4}$	$85.0_{\pm0.6}$	$69.8_{\pm0.7}$	$85.9_{\pm0.7}$	$70.6_{\pm0.6}$	$\mathbf{86.8}_{\pm0.3}$	$\underline{71.9}_{\pm0.9}$
	5%	$86.4_{\pm0.7}$	$71.8_{\pm0.6}$	$87.9_{\pm0.7}$	$70.3_{\pm1.3}$	$87.3_{\pm0.8}$	$70.9_{\pm1.2}$	$88.8_{\pm0.5}$	$70.9_{\pm0.7}$	$87.7_{\pm0.6}$	$73.6_{\pm1.2}$	$87.9_{\pm0.6}$	$74.2_{\pm0.8}$	$\mathbf{90.0}_{\pm0.3}$	$\underline{77.6}_{\pm0.9}$
	10%	$88.1_{\pm0.7}$	$73.1_{\pm0.9}$	$89.3_{\pm0.5}$	$72.1_{\pm1.1}$	$88.7_{\pm0.6}$	$71.2_{\pm1.4}$	$89.3_{\pm1.8}$	$71.4_{\pm0.9}$	$89.4_{\pm0.4}$	$75.4_{\pm0.8}$	$89.0_{\pm0.5}$	$76.3_{\pm1.4}$	$\mathbf{90.1}_{\pm0.5}$	$\underline{78.9}_{\pm0.8}$
IMDB	1%	$66.1_{\pm0.6}$	$59.4_{\pm1.8}$	$68.4_{\pm0.6}$	$60.6_{\pm1.0}$	$68.1_{\pm0.5}$	$60.3_{\pm2.7}$	$68.3_{\pm1.6}$	$60.1_{\pm1.5}$	$73.7_{\pm0.5}$	$60.6_{\pm1.2}$	$73.6_{\pm0.5}$	$61.4_{\pm1.8}$	$\mathbf{74.2}_{\pm1.5}$	$\underline{63.7}_{\pm0.6}$
	5%	$84.4_{\pm0.7}$	$77.3_{\pm1.6}$	$84.8_{\pm0.6}$	$80.3_{\pm0.9}$	$84.6_{\pm0.5}$	$79.6_{\pm3.3}$	$84.8_{\pm0.8}$	$79.1_{\pm2.3}$	$84.9_{\pm0.4}$	$79.4_{\pm0.7}$	$84.5_{\pm0.5}$	$80.3_{\pm2.0}$	$\mathbf{86.5}_{\pm1.2}$	$\underline{84.0}_{\pm0.3}$
	10%	$86.3_{\pm0.6}$	$79.6_{\pm2.9}$	$87.1_{\pm0.6}$	$82.4_{\pm1.2}$	$87.2_{\pm0.4}$	$81.7_{\pm3.1}$	$87.4_{\pm1.5}$	$81.2_{\pm1.5}$	$87.4_{\pm0.5}$	$81.3_{\pm0.6}$	$87.4_{\pm0.6}$	$82.2_{\pm2.1}$	$\mathbf{88.8}_{\pm0.9}$	$\underline{84.8}_{\pm0.7}$
QQP	1%	$77.5_{\pm0.6}$	$71.3_{\pm0.3}$	$78.6_{\pm0.6}$	$70.1_{\pm1.7}$	$78.2_{\pm0.7}$	$70.2_{\pm1.7}$	$78.0_{\pm0.7}$	$69.9_{\pm0.8}$	$78.3_{\pm0.6}$	$71.3_{\pm0.3}$	$77.9_{\pm0.6}$	$70.4_{\pm1.4}$	$\mathbf{78.9}_{\pm0.5}$	$\underline{72.8}_{\pm0.9}$
	5%	$81.7_{\pm0.7}$	$81.0_{\pm0.2}$	$82.2_{\pm0.6}$	$80.1_{\pm2.2}$	$81.8_{\pm0.6}$	$79.8_{\pm2.1}$	$80.9_{\pm0.5}$	$78.8_{\pm1.0}$	$82.4_{\pm0.5}$	$81.8_{\pm0.6}$	$81.9_{\pm0.5}$	$81.1_{\pm0.9}$	$\mathbf{84.0}_{\pm1.4}$	$\underline{83.9}_{\pm0.9}$
	10%	$84.6_{\pm0.7}$	$83.2_{\pm0.3}$	$85.6_{\pm0.4}$	$82.9_{\pm1.7}$	$84.2_{\pm0.6}$	$82.0_{\pm2.4}$	$84.3_{\pm0.8}$	$81.2_{\pm1.3}$	$84.2_{\pm0.5}$	$83.6_{\pm0.4}$	$84.4_{\pm0.6}$	$83.1_{\pm0.7}$	$\mathbf{86.7}_{\pm1.5}$	$\underline{86.4}_{\pm1.3}$
Avg.	1%	64.2	54.4	62.6	54.1	63.6	53.6	62.2	54.5	64.3	54.9	64.6	54.8	**65.9**↑1.3	$\underline{56.6}$↑1.7
	5%	73.8	62.5	74.5	62.8	73.9	63.3	74.5	61.9	74.7	64.0	74.8	64.4	**76.4**↑1.5	$\underline{67.3}$↑3.0
	10%	76.5	64.7	78.2	64.9	77.9	64.4	77.1	63.1	77.9	64.7	77.9	66.0	**79.7**↑1.5	$\underline{69.4}$↑3.4

Table 1: In-distribution (ID) and out-of-distribution (OOD) accuracy of active learning methods across six datasets, evaluated at different percentages of the entire dataset size. Results are averaged over three runs with different random seeds. Bold indicates the best ID values, underlining marks the best OOD values, and values highlighted in blue show an improvement over the next best result.

mance, but do not surpass that of ALVIN. Notably, ALVIN enhances the effectiveness of Uncertainty, considerably improving performance compared to using Uncertainty alone. Finally, BADGE and BERT-KM demonstrate improvements only over Random sampling.

	Method	AT	LN	NG	SE	WO	Avg.
NLI	Random	13.8	43.7	37.5	44.4	45.4	37.0
	Uncertainty	12.2	49.9	39.6	47.6	48.1	39.5
	BADGE	16.2	50.5	43.3	49.2	48.0	41.4
	BERT-KM	10.6	46.6	39.1	47.0	47.4	38.1
	CAL	11.8	50.1	42.5	49.8	48.7	40.6
	ALFA-Mix	13.6	47.9	41.3	49.3	47.7	40.0
	ALVIN	**18.2**	**54.1**	**48.3**	**52.8**	**53.6**	**45.4**↑**4.0**
ANLI	Random	83.2	29.9	31.4	29.7	41.7	43.2
	Uncertainty	85.0	32.5	30.7	29.8	41.8	44.0
	BADGE	62.4	30.2	33.3	30.2	39.5	39.1
	BERT-KM	74.3	28.6	30.2	29.4	37.4	40.0
	CAL	60.1	31.8	33.5	30.7	39.1	39.0
	ALFA-Mix	79.4	33.6	32.9	29.9	43.2	43.8
	ALVIN	**85.8**	**42.2**	**40.2**	**39.8**	**50.5**	**51.7**↑**7.7**

Table 2: Out-of-distribution performance of active learning methods trained on NLI and ANLI datasets, evaluated using the NLI stress test. Values highlighted in blue indicate an improvement over the next best result.

4.2 Additional OOD Generalization Results

Following Deng et al. (2023), we further evaluate the OOD generalization capabilities of models trained with various AL methods. Table 2 presents the results on the NLI Stress Test (Naik et al., 2018) for models trained on NLI and ANLI. We observe that ALVIN consistently outperforms all other AL methods in all stress tests, achieving an average performance improvement of 4.0 over BADGE, the next best performing method for models trained on NLI and, 7.7 over ALFA-Mix, the second best performing method for models trained on ANLI. Table 7 in the Appendix shows additional OOD results on Amazon reviews (Ni et al., 2019).

5 Analysis

5.1 Characteristics of Selected Instances

We analyze the characteristics of unlabeled instances identified through various active learning methods using uncertainty, diversity, and representativeness.

Uncertainty Following Yuan et al. (2020), we measure uncertainty with a model trained on the entire dataset to ensure that it provides reliable estimates. Specifically, we compute the average

Method	Unc.	Div.	Repr.
Random	0.121	0.641	0.584
Uncertainty	0.239	0.613	0.732
BADGE	0.117	0.635	0.681
BERT-KM	0.134	0.686	0.745
CAL	0.225	0.608	0.607
ALFA-Mix	0.136	0.645	0.783
ALVIN	0.123	0.672	0.823

Table 3: Uncertainty (Unc.), diversity (Div.), and representativeness (Repr.) of unlabeled instances selected for annotation by active learning methods. Results are averaged across all datasets.

predictive entropy of the annotation batch B via $-\frac{1}{|B|}\sum_{x \in B}\sum_{c=1}^{C} p(y = c|x) \log p(y = c|x)$, where C is the number of classes.

Diversity We assess diversity in the representation space as proposed by Ein-Dor et al. (2020). For each instance x_i, diversity within the batch B is calculated using $D(B) = \left(\frac{1}{|\mathcal{U}|}\sum_{x_i \in \mathcal{U}} \min_{x_j \in B} d(x_i, x_j)\right)^{-1}$, where $d(x_i, x_j)$ represents the Euclidean distance between x_i and x_j.

Representativeness We measure the representativeness of instances in the annotation batch B, to ensure that the generated anchors do not attract outliers, which can negatively affect both in-distribution and out-of-distribution performance (Karamcheti et al., 2021). To achieve this, we calculate the average Euclidean distance in the representation space between an example and its 10 most similar examples in $\mathcal{U}$, i.e., $R(x) = \frac{\sum_{x_i \in \text{KNN}(x)} \cos(x, x_i)}{K}$, where $\cos(x, x_i)$ is the cosine similarity between x and its k-nearest neighbors, and K is the number of nearest neighbors considered. Intuitively, a higher density degree within this neighborhood suggests that an instance is less likely to be an outlier (Zhu et al., 2008; Ein-Dor et al., 2020).

Results Table 3 presents the uncertainty, diversity, and representativeness metrics for unlabeled instances selected by different active learning methods. Uncertainty and CAL acquire the most uncertain examples, as indicated by their higher average entropy compared to other AL methods. Conversely, BADGE shows the lowest uncertainty, similar to ALFA-Mix and ALVIN. BERT-KM scores highest in diversity, while Uncertainty exhibits the lowest score, suggesting that uncertainty sampling often selects similar examples near the decision boundary. Compared with other AL methods, ALVIN overall has a considerably better diversity. ALVIN achieves the highest representativeness score, indicating that its anchors are effectively positioned in the representation space to attract meaningful unlabeled instances without including outliers that could degrade model performance.

	1%	5%	10%
NLI	94.5	94.8	96.1
ANLI	93.6	94.2	95.4

Table 4: Minority recall at different percentages of the dataset size.

5.2 Effectiveness of Minority Identification

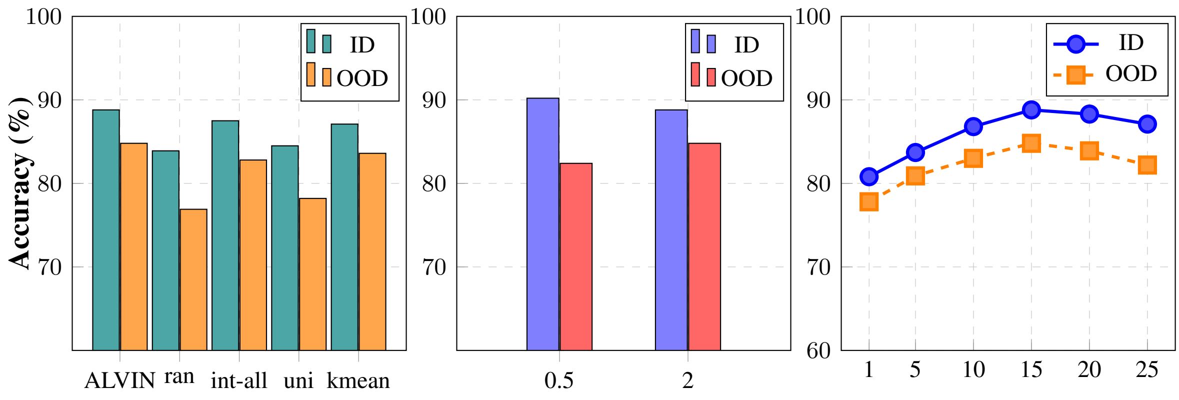
To verify the reliability of using training dynamics for identifying minority examples, we validate the approach across different AL rounds. We calculate recall, defined as the fraction of ground-truth minority examples identified by our strategy. We conduct experiments on the NLI and ANLI datasets, where minority and majority examples are predefined. As shown in Table 4, relying on training dynamics provides consistent results, as the identified minority instances align with the ground-truth annotations.

Method	Overlap Compr. ↓	Overlap Acc. ↓	Negation Compr. ↓	Negation Acc. ↓
Random	3.6±0.5	85.8±0.5	3.8±0.6	87.2±1.7
Uncertainty	3.3±0.4	85.2±1.2	4.3±0.2	93.7±1.8
BADGE	3.5±0.2	86.2±0.9	4.1±0.5	93.2±1.6
BERT-KM	3.1±0.6	84.5±0.5	3.9±0.3	91.5±2.2
CAL	3.8±0.2	88.2±0.7	3.5±0.2	86.5±1.9
Alfa-Mix	3.5±0.4	86.3±1.3	3.1±0.7	85.9±1.5
ALVIN	**2.4±0.5**	**80.7±0.8**	**2.2±0.4**	**82.6±1.8**

Table 5: Probing results for Overlap and Negation shortcut categories on the NLI dataset. Higher values in both compression (Compr.) and accuracy (Acc.) metrics indicate greater extractability of shortcut features from the model's representations.

5.3 Shortcut Extractability

We evaluate the extractability of shortcut features from model representations using minimum description length probing (Voita and Titov, 2020). Our evaluation focuses on two common shortcuts: high-word overlap between the premise and hy-



(a) ALVIN variants (b) Effect of α (c) Effect of K

Figure 2: Effects of different components of ALVIN and hyperparameter adjustments on both in-distribution (ID) and out-of-distribution (OOD) performance. Experiments are conducted on the IMDB dataset using 10% of the acquired data.

pothesis being labeled as "entailment," and the presence of negation being labeled as "contradiction." Higher probing accuracy and compression values suggest greater shortcut extractability. Table 5 presents the probing results on the NLI dataset for models trained with various AL methods over 10 rounds. We observe that prior AL methods increase shortcut extractability, as indicated by higher compression values and probing accuracies. In contrast, ALVIN exhibits the lowest compression values and probing accuracies.

5.4 Hyperparameter Study

We investigate the effect of the shape parameter α of Beta distribution on the overall performance of our proposed AL method. In Figure 2b we present the performance of ALVIN when the Beta distribution (1) is U-shaped, i.e., $\alpha = 0.5$, and (2) is bell-shaped, i.e., $\alpha = 2$. When the distribution is U-shaped, this leads to higher in-distribution accuracy but lower out-of-distribution generalization. This is due to the generated anchors being predominantly concentrated in two regions of the representation space, namely, those representing under-represented and well-represented groups. Due to the scarcity of examples in the under-represented group, anchors in this region fail to attract a sufficient number of instances, resulting in a tendency to attract examples from well-represented groups instead. Conversely, a bell-shaped distribution leads to anchors being dispersed across a wider range of the representation space, due to the broader variety of feature combinations it facilitates. Overall, adjusting the shape of the Beta distribution via α provides a means to balance the trade-off between in-distribution and out-of-distribution accuracy, potentially providing flexibility in the deployment of ALVIN depending on specific use-case requirements. Table 8 in the Appendix presents additional results where the Beta distribution is asymmetric.

We also investigate the impact of the hyperparameter K, which determines the number of anchors generated between under-represented and well-represented example pairs. As illustrated in Figure 2c, performance tends to be low with smaller K values due to inadequate exploration of the representation space. However, as K increases considerably, ALVIN's performance begins to align more closely with that of Uncertainty. This occurs because a larger number of anchors can cover a broader section of the representation space, thereby attracting high-uncertainty instances near the decision boundary.

5.5 Ablations

To better understand the effects of different components of ALVIN on both in-distribution and out-of-distribution performance, we conduct experiments with four ALVIN variants: (1) **ran** interpolates random pairs of labeled examples. The goal is to determine whether interpolations between under-represented and well-represented instances lead to the formation of meaningful anchors around unlabeled instances in the representation space, (2) **int-all** interpolates each minority example with every majority example, differing from the standard ALVIN practice which involves random pairings between under-represented and well-represented examples,

Method	SST-2	IMDB
Random	0	0
Uncertainty	173	107
BADGE	25640	3816
BERT-KM	4265	431
CAL	708	273
AlfaMix	915	428
ALVIN	781	357
$\rightarrow$ *Anchor Creation*	468	232
$\rightarrow$ *Example Selection*	311	125

Table 6: Time taken (in seconds) by active learning methods to select 100 instances from the unlabeled pool.

(3) **uni** uniformly samples from $\mathcal{I}$ (line 10) instead of using uncertainty to rank the unlabeled instances. It allows us to directly assess the impact of removing uncertainty-based selection, (4) **k-mean** clusters the samples from $\mathcal{I}$ (line 10) via k-means, and then selects the unlabeled instances closest to the centroids of these clusters.

The results from Figure 2a demonstrate the performance of standard ALVIN is superior to that of its variants. Notably, interpolations between under-represented and well-represented examples considerably enhance performance, as evidenced by the drastic drop in performance observed with the **ran** variant. Interpolating between an under-represented example and all well-represented examples also leads to a slight reduction in performance. We hypothesize that this is due to the anchors being spread across a large area of the representation space, thus attracting repetitive high-uncertainty instances from well-represented groups. Additionally, integrating uncertainty into ALVIN helps refine the selection of unlabeled instances, providing a more informative subset for annotation. Finally, the **kmean** variant does not show improvement over standard ALVIN.

5.6 Runtime

We assess the computational runtime required for selecting instances for annotation, following the methodology of Margatina et al. (2021). Specifically, we set the annotation batch size to 100, and conduct experiments using a Tesla V100 GPU. From Table 6, we see that Uncertainty is the most time-efficient AL method. Conversely, BADGE is the most computationally demanding AL method, as it involves clustering high-dimensional gradients. CAL ranks as the second most time-efficient method, followed by ALVIN, and ALFA-Mix. Overall, our approach demonstrates competitive speed compared to the fastest AL methods.

6 Related Work

Active Learning AL methods can be categorized into three groups, informativeness-based, representativeness-based, and hybrid AL approaches (Zhang et al., 2022b). Informativeness-based AL approaches typically measure the usefulness of unlabeled instances via uncertainty sampling (Lewis and Gale, 1994), expected gradient length (Settles et al., 2007), and Bayesian methods (Siddhant and Lipton, 2018). Recent AL works examine informativeness from the perspective of contrastive examples (Margatina et al., 2021), model training dynamics (Zhang and Plank, 2021), and adversarial perturbations (Zhang et al., 2022a). Representativeness-based AL approaches like core-sets (Sener and Savarese, 2018), discriminative active learning (Gissin and Shalev-Shwartz, 2019), and clustering-based methods (Zhdanov, 2019; Yu et al., 2022) aim to select diverse instances such that the underlying task is better specified by the labeled set. Finally, hybrid AL approaches combine these two paradigms either by switching between informativeness and representantivess (Hsu and Lin, 2015; Fang et al., 2017), or by first creating informativeness-based representations of the unlabeled instances and then clustering them (Ash et al., 2020; Ru et al., 2020). Compared to prior work using interpolations for AL (Parvaneh et al., 2022), ALVIN differs in two key ways: (1) we opt for interpolations between specific labeled instance pairs, rather than randomly interpolating labeled and unlabeled instances, and (2) we sample λ from a Beta distribution Beta(α, α) instead of optimizing it for each pair individually. This approach grants us greater control over the placement of the anchors in the representation space, ensuring they are positioned nearer to either under-represented or well-represented example groups as required.

Mixup ALVIN is inspired by mixup (Zhang et al., 2018), a popular data augmentation method originally explored in the field of computer vision. Mixup generates synthetic examples by interpolating random pairs of training examples and their labels. Recent mixup variants conduct interpolations using model representations (Verma et al., 2019), dynamically compute the interpolation ratio (Guo et al., 2019b; Mai et al., 2022), explore different interpolation strategies (Yin et al., 2021), and combine mixup with regularization techniques (Jeong et al., 2022; Kong et al., 2022). In the context of NLP, Guo et al. (2019a) apply mixup on word and sentence embeddings using convolutional and recurrent neural networks. Conversely, Yoon et al. (2021) propose a mixup variant that conducts interpolations on the input text. Park and Caragea (2022) apply mixup to calibrate BERT and RoBERTa models, while Chen et al. (2020) propose TMix, a mixup-inspired semi-supervised objective for text classification.

7 Conclusion

In this work, we propose ALVIN, an active learning method that uses intra-class interpolations between under-represented and well-represented examples to select instances for annotation. By doing so, ALVIN identifies informative unlabeled examples that expose the model to regions in the representation space which mitigate the effects of shortcut learning. Our experiments across six datasets, encompassing a broad range of NLP tasks, demonstrate that ALVIN consistently improves both in-distribution and out-of-distribution accuracy, outperforming other state-of-the-art active learning methods.

Limitations

While we have demonstrated that ALVIN mitigates shortcut learning, we have not explored its ability to address fairness issues. ALVIN may inadvertently amplify biases present in the model's representations, as these are used to generate the anchors. Additionally, our experiments are limited to models trained with the masked language modeling pre-training objective, excluding other pre-training methods and model sizes. Finally, we acknowledge that active learning simulations are not always representative of real-world setups and annotation costs.

Acknowledgements

Michalis Korakakis is supported by the Cambridge Commonwealth, European and International Trust, the ESRC Doctoral Training Partnership, and the Alan Turing Institute. Andreas Vlachos is supported by the ERC grant AVeriTeC (GA 865958). Adrian Weller acknowledges support from a Turing AI Fellowship under grant EP/V025279/1, and the Leverhulme Trust via CFI.

References

David Arthur and Sergei Vassilvitskii. 2007. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 1027–1035. SIAM.

Jordan T. Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. 2020. Deep batch active learning by diverse, uncertain gradient lower bounds. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Xuefeng Bai, Jialong Wu, Yulong Chen, Zhongqing Wang, and Yue Zhang. 2023. Constituency parsing using llms. *CoRR*, abs/2310.19462.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 491–500. ACM.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Zihang Chen, Hongbo Zhang, Xiaoji Zhang, and Leqi Zhao. 2017. Quora question pairs.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben

Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113.

Xun Deng, Wenjie Wang, Fuli Feng, Hanwang Zhang, Xiangnan He, and Yong Liao. 2023. Counterfactual active learning for out-of-distribution generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11362–11377, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Meng Fang, Yuan Li, and Trevor Cohn. 2017. Learning how to active learn: A deep reinforcement learning approach. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 595–605, Copenhagen, Denmark. Association for Computational Linguistics.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nat. Mach. Intell.*, 2(11):665–673.

Daniel Gissin and Shai Shalev-Shwartz. 2019. Discriminative active learning. *CoRR*, abs/1907.06347.

Denis A. Gudovskiy, Alec Hodgkinson, Takuya Yamaguchi, and Sotaro Tsukizawa. 2020. Deep active learning for biased datasets via fisher kernel self-supervision. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9038–9046. Computer Vision Foundation / IEEE.

Hongyu Guo, Yongyi Mao, and Richong Zhang. 2019a. Augmenting data with mixup for sentence classification: An empirical study. *CoRR*, abs/1905.08941.

Hongyu Guo, Yongyi Mao, and Richong Zhang. 2019b. Mixup as locally linear out-of-manifold regularization. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 3714–3722. AAAI Press.

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

Wei-Ning Hsu and Hsuan-Tien Lin. 2015. Active learning by learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 2659–2665. AAAI Press.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2022. Augmenting document representations for dense retrieval with interpolation and perturbation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 442–452, Dublin, Ireland. Association for Computational Linguistics.

Siddharth Karamcheti, Ranjay Krishna, Li Fei-Fei, and Christopher Manning. 2021. Mind your outliers! investigating the negative impact of outliers on active learning for visual question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7265–7281, Online. Association for Computational Linguistics.

Divyansh Kaushik, Eduard H. Hovy, and Zachary Chase Lipton. 2020. Learning the difference that makes A difference with counterfactually-augmented data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran S. Haque, Sara M. Beery, Jure Leskovec, Anshul

Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR.

Fanshuang Kong, Richong Zhang, Xiaohui Guo, Samuel Mensah, and Yongyi Mao. 2022. DropMix: A textual data augmentation combining dropout with mixup. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 890–899, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Michalis Korakakis and Andreas Vlachos. 2023. Improving the robustness of NLI models with minimax training. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14322–14339, Toronto, Canada. Association for Computational Linguistics.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234, Copenhagen, Denmark. Association for Computational Linguistics.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 3–12. ACM/Springer.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Zhijun Mai, Guosheng Hu, Dexiong Chen, Fumin Shen, and Heng Tao Shen. 2022. Metamixup: Learning adaptive interpolation policy of mixup with metalearning. *IEEE Trans. Neural Networks Learn. Syst.*, 33(7):3050–3064.

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

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12284–12314, Toronto, Canada. Association for Computational Linguistics.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Seo Yeon Park and Cornelia Caragea. 2022. On the calibration of pre-trained language models using mixup guided by area under the margin and saliency. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5364–5374, Dublin, Ireland. Association for Computational Linguistics.

Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Reza Haffari, Anton van den Hengel, and Javen Qinfeng Shi. 2022. Active learning by feature mixing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 12227–12236. IEEE.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron C. Courville, Doina Precup, and Guillaume Lajoie. 2021. Gradient starvation: A learning proclivity in neural networks. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 1256–1272.

Aahlad Manas Puli, Lily H. Zhang, Yoav Wald, and Rajesh Ranganath. 2023. Don't blame dataset shift! shortcut learning due to gradients and cross entropy. *CoRR*, abs/2308.12553.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

Dongyu Ru, Jiangtao Feng, Lin Qiu, Hao Zhou, Mingxuan Wang, Weinan Zhang, Yong Yu, and Lei Li. 2020. Active sentence learning by adversarial uncertainty sampling in discrete space. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4908–4917, Online. Association for Computational Linguistics.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *CoRR*, abs/1911.08731.

Ozan Sener and Silvio Savarese. 2018. Active learning for convolutional neural networks: A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Burr Settles. 2009. Active learning literature survey.

Burr Settles, Mark Craven, and Soumya Ray. 2007. Multiple-instance active learning. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pages 1289–1296. Curran Associates, Inc.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. 2020. The pitfalls of simplicity bias in neural networks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Aditya Siddhant and Zachary C. Lipton. 2018. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2904–2909, Brussels, Belgium. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

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. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, Online. Association for Computational Linguistics.

Zhen Tan, Alimohammad Beigi, Song Wang, Ruocheng Guo, Amrita Bhattacharjee, Bohan Jiang, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation: A survey. *CoRR*, abs/2402.13446.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. 2019. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics*, 8:621–633.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. 2019. Manifold mixup: Better representations by interpolating hidden states. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6438–6447. PMLR.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online. Association for Computational Linguistics.

Zhao Wang and Aron Culotta. 2020. Identifying spurious correlations for robust text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3431–3440, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yadollah Yaghoobzadeh, Soroush Mehri, Remi Tachet des Combes, T. J. Hazen, and Alessandro Sordoni. 2021. Increasing robustness to spurious correlations using forgettable examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3319–3332, Online. Association for Computational Linguistics.

Wenpeng Yin, Huan Wang, Jin Qu, and Caiming Xiong. 2021. BatchMixup: Improving training by interpolating hidden states of the entire mini-batch. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4908–4912, Online. Association for Computational Linguistics.

Soyoung Yoon, Gyuwan Kim, and Kyumin Park. 2021. SSMix: Saliency-based span mixup for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3225–3234, Online. Association for Computational Linguistics.

Yue Yu, Lingkai Kong, Jieyu Zhang, Rongzhi Zhang, and Chao Zhang. 2022. 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*, pages 1422–1436, Seattle, United States. Association for Computational Linguistics.

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

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *CoRR*, abs/2308.01825.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Mike Zhang and Barbara Plank. 2021. Cartography active learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 395–406, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shujian Zhang, Chengyue Gong, Xingchao Liu, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. 2022a. ALLSH: Active learning guided by local sensitivity and hardness. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1328–1342, Seattle, United States. Association for Computational Linguistics.

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

Fedor Zhdanov. 2019. Diverse mini-batch active learning. *CoRR*, abs/1901.05954.

Jingbo Zhu, Huizhen Wang, Tianshun Yao, and Benjamin K Tsou. 2008. Active learning with sampling by uncertainty and density for word sense disambiguation and text classification. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 1137–1144, Manchester, UK. Coling 2008 Organizing Committee.

A Appendix

A.1 Additional Results

Method	Accuracy (%)
Random	86.56
Uncertainty	85.89
BADGE	83.23
BERT-KM	84.98
CAL	86.22
ALFA-Mix	86.18
ALVIN	**89.75**↑3.19

Table 7: Out-of-distribution performance of active learning methods trained on the SA dataset and evaluated on Amazon reviews. The value highlighted in blue indicates an improvement over the next best result.

Beta	**ID**	**OOD**
$\alpha = 2, \beta = 5$	80.5	82.4
$\alpha = 5, \beta = 2$	87.5	78.2
$\alpha = 2, \beta = 2$	88.8	84.8

Table 8: Comparison of ALVIN ID and OOD performance when Beta is asymmetric.