Adversarial Active Learning with Guided BERT Feature Encoding

Xiaolin Pang[1], Kexin Xie[1], Yuxi Zhang[1], Max Flemming[1], Damian Chen Xu[1], and Wei Liu[2]

[1] Salesforce Inc., San Francisco, CA 94105
{xpang,kexin.xie,yuxi.zhang,m.fleming,damian.xu}@salesforce.com
[2] School of Computer Science, University of Technology Sydney, NSW 2007, Australia
wei.liu@uts.edu.au

Abstract. Recent advances in BERT-based models has significantly improved the performance of many applications on text data, such as text classification, question answering, e-commerce search and recommendation system, etc. However, the labelling of text data is often complex and time-consuming. While active learning can interactively query and label the data, the effectiveness of existing active learning methods is mostly limited by static text embedding approaches and by the insufficiency of training data. To address this critical problem, in this research we propose a BERT-based adversarial semi-supervised active learning (B-ASAL) model. In our approach, we use generative adversarial modelling and semi-supervised learning to guide the fine-tuning of the BERT and to optimize its corresponding text embeddings and feature encodings. The adversarial generator paired with a semi-supervised classifier guided the BERT model to adjust its feature encoding to best fit the distribution of not only class labels but also the discrimination of labeled and unlabeled data. Moreover, our B-ASAL model selects data points with high uncertainty and high diversity to be labeled using minimax entropy regularization. To our best knowledge, this is the first work that uses adversarial semi-supervised learning joined with active learning to guide and optimize feature encoding. We evaluate our method on various real-world text classification datasets and show that our model outperforms state-of-the-art approaches.

1 Introduction

Advances in deep learning has transformed the field of natural language processing (NLP). BERT-based [4] models are widely used in various NLP tasks: from text classification to question answering to e-commerce search and recommendation etc. Meanwhile, data labelling, a fundamental bottleneck in machine learning, becomes a critical problem due to annotation cost and the need of large amount of labeled data for deep learning NLP tasks. For instance, to build a question answering (QA) model, a human annotator must first read a piece of text and then reason about the answer to the question from context. It is even harder for domain specific labeling task due to the cost of using domain expert.

In e-commerce, there are very few fine labeled data and professionals are needed to annotate fine labels to map items to fine-grained categories. Therefore, it is necessary to consider how to select more informative samples, so that a better model can be trained with limited labelling capabilities.

Active learning (AL) is one method to collect labeled data cost-efficiently. The goal is to choose the most relevant data points and then query labels from an oracle. Using AL, we can query labels for a small subset of the most relevant documents and immediately train a robust model. For instance, leveraging pre-trained BERT-based language models, task-specific models can be fined-tuned continuously by incorporating newly annotated samples in each iteration to boost the model performance. However, the effectiveness of AL learning methods on NLP tasks is mostly limited by static text embedding, the insufficiency of training data, and the similarity between labeled and unlabeled data distributions.

This research addresses the exact problems above. To address the static text embedding problem, we propose an active learning framework while BERT is fine-tuned in the training progress where the text embedding and feature encoding are both optimized for the training data. To address the problem of the insufficiency of labeled training data, we use adversarial semi-supervised learning to utilize unlabeled data for learning effective representations and for generating new synthetic samples [1][18]. To discriminate labeled data from unlabeled ones, we incorporate minimax entropy to measure and differentiate the distributions of labeled and unlabeled data. We name our method BERT-based adversarial semi-supervised active learning (B-ASAL). In summary, our contributions in this research are as follows:

- We propose B-ASAL for learning from partially labeled text data. Our B-ASAL model integrates active learning with the fine-tuning of BERT, which guides the BERT to optimize text embedding and feature encoding according to the distribution of the training data.
- We also introduce in the B-ASAL model a generative adversarial network joint with semi-supervised learning, a strategy that can utilize unlabeled data and generalize latent features to select samples for labelling.
- We employ minimax entropy optimization for the unlabeled data to reduce the distribution gap with labeled data while extracting discriminative features for selecting highly representative data samples. Moreover, we also employ conditional entropy maximization in the adversarial network to enhance the robustness and generate uniform-distributed samples.
- We conduct extensive experiments on public datasets and show that our model outperforms state-of-the-art approaches.

2 Related work

Deep Active Learning (DAL) DAL integrates data labeling and deep model training to improve model performance with minimal amount of labeled data.[1]

[1] In this work, we will only consider the most common pool-based deep active learning.

The scoring function for labeling can be entropy or confidence-score based. Coreset active learning [13] selects a small set of points that approximates the shape of a larger point set using concept of computational geometry. [19] combines clustering with a pre-trained language model (BERT) to select samples. Variational adversarial active learning (VAAL) [14] is proposed as a task-agnostic diversity-based algorithm that samples data points using a discriminator trained adversarially to discern labeled and unlabeled points.

GAN Semi-supervised Learning Semi-supervised models are able to improve the generalization capability by learning from fewer labeled data points with the help of a large number of unlabeled data points. Semi-Supervised GAN (SS-GAN)[12] extends standard GAN [7] where the labeled data is used to train the discriminator, while the unlabeled data (as well as the ones automatically generated) improve its inner representations. CatGAN [15] proposes categorical GAN for unsupervised and semi-supervised framework by utilizing unlabeled data to learn multi-class classifier. Besides, GAN-BERT [2], a semi-supervised learning model for natural language processing task, enriches the BERT fine-tuning process with a SS-GAN perspective.

Pre-trained BERT BERT [4] has been used in combination with AL to select representative samples to reduce labelling effort for text classification [5, 8]. In [5], it presents a large-scale an empirical study on AL techniques for BERT-based classification, covering a diverse set of AL strategies and datasets on binary text classification. [8] also conducts an empirical study by comparing different uncertainty-based acquisition strategies on two classical NLP multi-class classification datasets.

Entropy regularization Entropy regularization has been widely used in various deep learning models. In the field of domain adaptation, [11] uses entropy optimization for matching source data to target data distribution. The MAL framework [6] uses the similar idea and proposes a semi-supervised minimax entropy-based active learning algorithm in an adversarial manner for image related tasks. CatGAN [15] and the study of SS-GAN use entropy regularization [3] to improve generation of images conditioned on class assignment.

3 Learning Framework

In this section, we describe our proposed method, the B-ASAL (the BERT-based adversarial semi-supervised active learning) model.

3.1 Problem Formulation

We consider exploiting unlabeled data points and formulate semi-supervised generative adversarial active learning problem as: given an initial labeled data set $\mathcal{S}^l : (\mathcal{X}^l, \mathcal{Y}^l) = \{(x_l, y_l)\}$, where $l \in \{1, ..., m\}$ with size M, and a large unlabeled data pool $\mathcal{S}^u : \mathcal{X}^u = \{(x_u)\}$, where $u \in \{1, ..., n\}$ with size N ($M \ll N$) and $y_l \in \{0, 1\}$ is the class label of x_l for binary classification, or $y_l \in \{1, ..., K\}$ for multi-class classification. We also have a set of generated adversarial data points S^g: $\mathcal{X}^g = \{(x_g)\}$, pairing with true data points to enhance model learning, where x_g is transformed by noise input $\{z_1, ..., z_{m+n}\} \sim U\{0, 1\}$ (i.e. p_z)

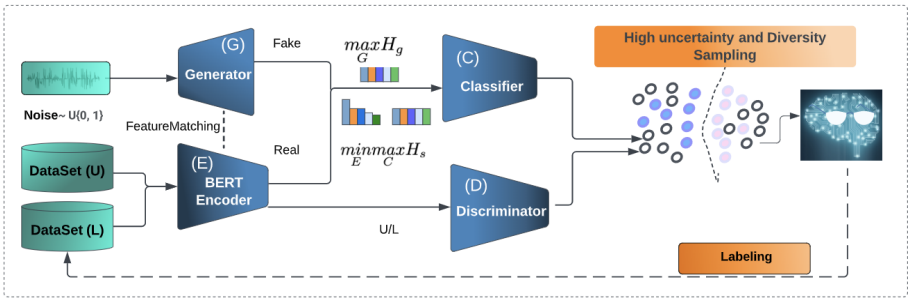


Fig. 1: Workflow of our B-ASAL model. There are Four components in our model: Generator (G), BERT Encoder (E), Classifier (C) and Discriminator (D). Each component and loss function has detailed explanations in Section 3.2 and 3.3.

and g $\in \{1, ..., m+n\}$. For all of feature inputs: $\mathcal{X}^l$, $\mathcal{X}^l$ and $\mathcal{X}^g$, we assume they denotes encoded input through encoder. The AL model $\mathcal{M}$ parameterized by $\theta \in \Theta$ is trained on labeled data with their labels, unlabeled data and adversarial data (i.e. $\mathcal{S}^l \cup \mathcal{S}^u \cup \mathcal{S}^g$). This training can be formalized by the optimization problem:

$$\underset{\theta}{\operatorname{argmin}} \mathcal{L}(\theta; y_i | x \in \mathcal{X}^u \cup \mathcal{X}^l \cup \mathcal{X}^g, y \in \mathcal{Y}^l), \tag{1}$$

where $\mathcal{L}$ is the loss function composed of supervised loss trained for labeled data, unsupervised loss trained for unlabeled data and generated fake data. In each AL cycle, trained model $\mathcal{M}$ selects top $k\%$ samples (denoted as $\mathcal{S}^q$ and $\mathcal{S}^q \in \mathcal{S}^u$) constrained by query budget limit and a designed acquisition function $f(x, \mathcal{M})$: $\text{argmax}_{x \in \mathcal{X}^u} f(x, \mathcal{M} | x \in \mathcal{X}^u)$ to obtain their labels from the oracle. $\mathcal{S}^l$ and $\mathcal{S}^u$ are then updated in next cycle, and $\mathcal{M}$ is retrained on $\mathcal{S}^l \cup \mathcal{S}^u \cup \mathcal{S}^g$.

3.2 Proposed Framework: B-ASAL

In this work, we propose a BERT-based adversarial semi-supervised active learning (B-ASAL) framework. We design each possible component to come up with a model learning objective and acquisition strategy. The components are: Generator (G), Classifier (C), Discriminator (D) and BERT Encoder (E) as shown in Figure 1.

To utilize unlabeled data, we introduce a semi-supervised GAN framework built with BERT fine-tuning across the entire training process. In an adversarial manner, the generator is used to fool the classifier by generating highly realistic data samples. It takes noise input[2] and transforms to map true data distribution. The transformed noise input is treated as $k+1th$ addition class for the semi-supervised learner. To enhance the robustness and reduce mode collapse, the generator is trained to apply feature matching between generated samples and

[2] Here we generate noise following a uniform distribution (which can be easily replaced by other distributions when needed). We denote noise as: $\{z_1, ..., z_n\} \sim U\{0, 1\}$

real data. Moreover, conditional entropy maximization over samples from the generator is employed as well.

The classifier is designed to pair with the generator and can be treated as a multi-class discriminator for K+1 classes. For labeled data, it is trained to differentiate k classes and $k+1th$ fake class. For unlabeled data, a minimax loss is optimized by performing entropy maximization with respect to the predicted class and entropy minimization with respect to fine-tuned feature encoder. It reduces the distribution gap while extracting discriminative features. We select samples having high entropy to be labeled, which indicates these samples are predicted by the model with high uncertainty.

The discriminator is a binary classifier, we use it to predict whether a sample is labeled or not based on a latent representation from our encoder. We select unlabeled data points with low discriminator scores, which indicates that these samples are sufficiently different from previously labeled ones.

BERT encoder is used as the feature encoder. It is fine-tuned, and the fine-tuning encoded features are through the logit activation layer of the classifier and the discriminator. For labeled data, It is trained to maximize the probability of class assignment from the classifier. It is also trained to differentiate label and unlabeled data from the discriminator. For unlabeled data, it is trained to minimize the entropy to have better discriminative features.

In each AL cycle, samples that have high uncertainty and diversity are selected from unlabeled data for labelling. Detailed steps of our method are shown in Algorithm 1.

3.3 Learning Objective

Now we discuss the overall cost function by incorporating each decomposed component, including generator loss (L_G), discriminator loss (L_D), and classifier loss (L_C). Each type of these losses has supervised loss for labeled data (L_L) and unsupervised loss for unlabeled data (L_U).

Labeled Data Learning BERT Encoder(E) and Classifier (C) are trained to classify labeled data points correctly into $\{1, ..., K\}$ class by both standard cross entropy loss and conditional entropy loss over samples uniformly distributed to K classes from the generator (G) to achieve optimal classification results. The generator (G) generates fake data points belonging to $K + 1th$ class. It tries to minimize the loss between generated fake data points with real data points, including the loss of feature matching and misclassification loss to K classes, while the classifier (C) tries to maximize it. This min-max loss is trained through an adversarial setting and can be denoted as:

$$\mathcal{L}_{\mathrm{L}} = -\underset{G}{min}\underset{C}{max}\mathcal{L}_{C^l} + \mathcal{L}_{G^l} \tag{2}$$

The loss function of Classifier (C) ($\mathcal{L}_{C^l}$):

$$\mathcal{L}_{C^l} = -\mathbb{E}_{(x,y)\in S^l} log[p(y \leq k|x)] - \mathbb{E}_{z\sim p_z} H_g[p(y \leq k|G(z), C)], \tag{3}$$

where conditional entropy $H_g = -\sum_1^m p(y = k|G(z), C)log[p(y = k|G(z^m), C)]$ and $k \in \{1, ..., K\}$ classes and $m \in M$.
The loss function related to G (L_G) includes feature matching loss to make generated data are very close to the real ones and also considers the error induced by fake data correctly identified by classifier.

$$\mathcal{L}_{G^l} = \|\mathbb{E}_{x \in S^l} f(x) - \mathbb{E}_{x \in S^g} f(x)\|_2^2 - \mathbb{E}_{x \in S^g} log[1 - p(y \leq k|x)], \tag{4}$$

where f is the layer with logits through the classifier and fine-tuning encoder.

Unlabeled Data Learning When training on the unlabeled data, the unsuprevised loss is $\mathcal{L}_{\mathrm{U}} = \mathcal{L}_{\mathrm{H}}^u + \mathcal{L}_G^u$, where $\mathcal{L}_{\mathrm{H}}^u$ denotes minimax entropy employed on classifier and feature encoder; L_G^u denotes feature matching loss for generated samples paired with unlabeled data, same as first term in L_G^l. They are computed as:

$$\mathcal{L}_{\mathrm{H}}^u = -\min_E \max_C H_s[p(y \leq k|x)], \tag{5}$$

where the minimax entropy $H_s = -\sum_1^K p(y = k|x)log(p(y = k|x)$ and $k \in \{1, ..., K\}$ classes; we first minimize the entropy in feature encoder to have more discriminative representation and then maximize entropy in classifier to have a more uniform feature representation.

$$\mathcal{L}_G^u = \|\mathbb{E}_{x \in S^u} f(x) - \mathbb{E}_{x \in S^g} f(x)\|_2^2, \tag{6}$$

where this part can be combined with first term of Eq. 4 as learning feature matching loss for all of generated samples coming from generator.

Discrimitive Learning for labeled and Unlabeled Data The diversity of the data is predicted by a binary classifier (i.e. discriminator denoted as D) that is trained to distinguish between the labeled and unlabeled encoded features. The loss function of D is:

$$\mathcal{L}_D = -\mathbb{E}_{(x,y) \in S^l} log[p(y^l|x^l)] - \mathbb{E}_{(x,y) \in S^u} log[p(y^u|x^u)]. \tag{7}$$

Acquisition Strategy In our acquisition strategy, we select data points with high diversity and high uncertainty to be labeled. The selection criteria are: (a) *high diversity*: we use the probability associated with the discriminator's (D) predictions as a score to rank samples. The lower the probability, the more confident D is that it comes from the unlabeled pool. (b) *high uncertainty*: the entropy obtained by the classifier on the unlabeled data is used to choose the data points. A higher entropy value is associated with a lower confidence score. The top-k% samples that meet both criteria are selected for labeling.

Algorithm 1 BERT-based Adversarial Semi-Supervised Active Learning (B-ASAL)

Input: labeled data S^l, unlabeled data S^u. Initialize parameters of generator ϕ_G, discriminator γ_D , classifier σ_C and BERT encoder β_E.
Output: Optimized ϕ_G, γ_D , σ_C and β_E

for i = 1 to epochs **do**
  Sample batch of size n from S^l labeled and S^u unlabeled data $|S^l| = |S^u| = n$
  Sample $\{z_1, \ldots, z_{m+n}\} \in S^g$ from the prior P_z
  Generate encode $E(S^l)$, $E(S^u)$ and $E(S^g)$
  For labeled data $(x, y \in S^l)$:
    Compute $\mathcal{L}_C^l$ from Eq. 3
    Update C by descending:
      $\sigma_C \leftarrow \sigma_C - \lambda_1 \mathcal{L}_C^l$
  For unlabeled data $(x \in S^u)$:
    Compute $\mathcal{L}_H^u$ from Eq. 5
    Update E and C by descending/ascending:
      $\beta_E \leftarrow \beta_E + \lambda_2 \mathcal{L}_H^u$
      $\sigma_C \leftarrow \sigma_C - \lambda_3 \mathcal{L}_H^u$
  For labeled data $(x, y \in S^l)$ and unlabeled data $(x \in S^u \cup S^g)$:
    Compute $\mathcal{L}_G$ from Eq. 4 and Eq. 6
      Update G by descending:
      $\phi_G \leftarrow \phi_G - \lambda_4 \mathcal{L}_G$
    Compute $\mathcal{L}_D$ from Eq. 7
      Update D by descending:
      $\gamma_D \leftarrow \gamma_D - \lambda_5 \mathcal{L}_D$
end for

4 Experiments

To study the effectiveness of our approach, we evaluate model performance on multiple open public data sets by comparing them with the different sampling strategy.

Datasets: Total of five data sets are used for evaluation: Fine and Coarse Question Classification (QC) [9], Match and Mismatched pair MNLI dataset [16] and Multi-label emotion data [10].

Experiments Settings: We fetch all of the above data from HuggingFace datasets library.We run 3 different seeds and 3 epochs for each experiment. We take the mean of the results. [17]

Performance Evaluation: The model performance is measured by the classification accuracy on balanced datasets and measuring micro-F1, macro-F1 and hamming score on imbalanced datasets/multi-label datasets by varying percentage of labeled data, ranging from $\{1\%, 2\%, ..., 10\%\}$ or $\{0.1\%, 2\%, ..., 5\%\}$ depending on data size.

Acquisition Strategies: To compare with our acquisition function (i.e. B-ASAL), we use three baselines: random sampling (Rdm), diversity sampling (Div) and entropy uncertainty sampling (En).

Table 1: Comparisons of Sampling Methods on Accuracy. The percentages shown in the table (and the same for all tables hereafter) refer to the percentages of training data labeled by active learning.

Method	QC-Coarse						QC-Fine					
	1%	2%	5%	10%	20%	30%	1%	2%	5%	10%	20%	30%
Rdm	21.2	36.3	58.3	84.4	92.8	94.4	8.1	11.6	33.9	56.3	72.0	77.8
En	22.1	36.5	81.1	89.1	93.6	94.5	7.2	13.5	38.7	54.9	67.9	75.9
Div	18.7	35.0	58.7	86.5	92.8	94.6	10.5	11.0	45.0	60.8	71.5	75.2
B-ASAL	**26.2**	**42.6**	**90.4**	**94.5**	**95.5**	**96.3**	**17.3**	**19.2**	**57.4**	**62.4**	**76.9**	**80.8**

Table 2: Comparisons of Sampling Methods on F1

Method	MNLI-mismatch							MNLI-match						
	0.1%	0.2%	0.5%	1%	2%	5%	10%	0.1%	0.2%	0.5%	1%	2%	5%	10%
Rdm	22.3	40.0	46.4	78.3	76.7	85.2	88.3	21.2	21.4	42.6	58.2	80.0	84.6	85.0
En	25.0	31.0	40.7	73.7	79.7	86.3	88.1	24.0	29.3	54.0	69.0	82.0	87.3	91.3
Div	25.0	27.0	42.3	73.7	81.3	86.7	87.0	22.7	37.3	43.0	65.7	71.0	88.0	87.7
B-ASAL	**29.0**	**42.3**	**55.7**	**77.0**	**86.7**	**89.7**	**91.7**	**29.3**	**39.7**	**74.3**	**77.7**	**83.3**	**91.0**	94.2

Implementation: For Classifier (C), Discriminator (D), Generator (G), we use the Multi-Layer Perceptron (MLP) neural network with one hidden layer activated by a leaky-relu function followed by a softmax layer for the multi-class prediction and sigmoid layer for multi-label prediction. The dropout is 0.1 after the hidden layer. The input noise vector of G is uniformly distributed. BERT Encoder (E) is loaded from the pre-trained BERT model and fine-tuned through C, D and G.

4.1 Question Answering Classification

Question Classification (QC) dataset [9] has both a six-class (QC-Coarse) and a fifty-class (QC-Fine) version. Both have 5,452 training data and 500 test data. Table 1 shows the experiment output. The accuracy performance of QC-Coarse data can achieve 90%+ when using only 5% labeled data and QC-Fine set achieves around 80% by using 20% labeled data. Our sampling strategy (i.e. B-ASAL) achieves much better performance.

4.2 Multi-Genre Natural Language Inference

The Multi-Genre Natural Language Inference (MultiNLI) corpus is a collection of 433k sentence pairs annotated with textual entailment information [16]. The task is to infer the relationship between the premise and hypothesis in binary classification. We evaluated matched and mismatched data sets. The results are shown in Table 2. The F1-score of mismatch data can achieve 90%+ when using only 10% labeled data, and match set achieves around 80% by using only 2% labeled data. Our sampling strategy (i.e. B-ASAL) consistently shows much better performance over the rest three baselines.

4.3 Multi-Label Emotion Classification

SemEval-2010 Task is for multi-label emotion classification (11 emotions) [10]. Hamming, F1-micro and F1-macro scores are used to evaluate model perfor-

Table 3: Comparisons of Sampling Methods on Multi-label Emotion Dataset

Method	Micro-F1					Macro-F1					Hamming				
	1%	2%	5%	10%	20%	1%	2%	5%	10%	20%	1%	2%	5%	10%	20%
Rdm	19.6	25.0	50.7	57.7	60.6	7.51	11.4	27.1	32.4	38.2	11.5	14.8	34.0	40.5	43.5
En	20.6	23.7	41.1	54.8	56.3	7.6	7.96	18.4	26.8	28.2	12.2	13.4	26.5	37.7	38.9
Div	19.6	23.8	41.0	54.3	55.9	7.9	7.98	25.9	26.5	27.2	11.5	13.6	26.4	37.3	38.1
B-ASAL	**23.9**	**33.9**	**54.6**	**58.2**	**60.9**	**8.51**	**13.1**	**27.3**	**32.7**	**38.5**	**13.7**	**20.6**	**37.6**	**40.8**	**43.8**

Table 4: Ablation Studies on Accuracy

Method	QC-Coarse						QC-Fine					
	1%	2%	5%	10%	20%	30%	1%	2%	5%	10%	20%	30%
L-only	20.4	36.0	79.5	91.1	94.2	95.1	8.4	13.4	39.1	59.8	72.9	76.7
No-GAN	25.1	37.3	72.8	79.6	93.2	95.0	13.2	14.1	24.6	43.6	61.7	67.8
BERT	20.1	21.4	50.0	81.6	93.4	95.1	0.8	3.5	17.0	30.5	54.7	64.9
B-ASAL	**26.2**	**42.6**	**90.4**	**94.5**	**95.5**	**96.3**	**17.3**	**19.2**	**57.4**	**62.4**	**76.9**	**80.8**

mance defined by the Task. Our method's performance outperforms the other sampling strategies and with only 20% labeled samples, our method can almost achieve the benchmark performance (as shown in Table 3).

4.4 Further Analysis

To investigate the contribution of each component and understand the benefit of utilizing the unlabeled data points, we designed several types of experiments to show the overall effectiveness of B-ASAL.

Labeled-only vs. (Labeled ∪ Unlabeled) We study the effectiveness of semi-supervised learning compared to supervised learning by having GAN component. The comparison outputs are shown on the row of L-only vs. the row of B-ASAL in Tables 4 and 5, where L-only denotes only labeled data is used for training with GAN and B-ASAL is our model utilizing semi-supervised learning with GAN. The results show that semi-supervised B-ASAL performs better than supervised GAN where there are only annotated data in training.

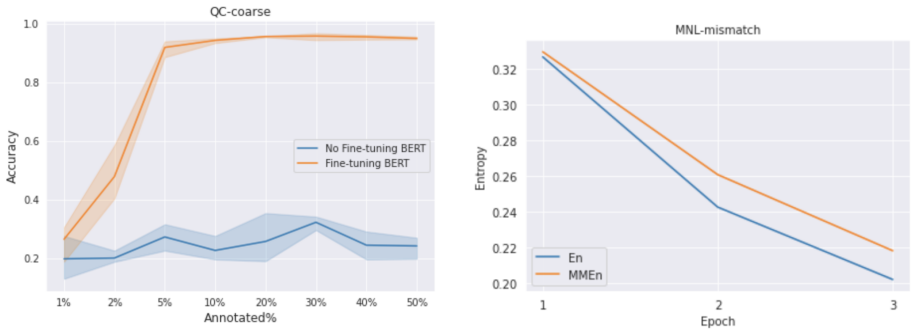
With GAN vs. Without GAN We study the performance of the model with Generator (G) compared to the model without G. The model without G is when B-ASAL only has components E, D and C. The output shows on the row of No-GAN vs. the row of B-ASAL in Table 4 and 5. The results show GAN generates better performance by utilizing unlabeled data and pairing with the classifier.

B-ASAL vs. BERT-only We compare our model performance with the fully supervised BERT classifier. Results are shown on the row of BERT vs. the row of B-ASAL in Table 4 and 5. Apparently, B-ASAL outperforms supervised classification without utilizing GAN and unlabeled data.

BERT Encoder Fine-tuning vs. No Fine-tuning To demonstrate fine-tuning BERT plays an important role throughout the entire B-ASAL training, we study the performance of fine-tuning vs. no fine-tuning (Figure 2a). It shows the encoder plays an important role not only as a representation encoder but also as a collaborator with component of D and C to achieve the optimal results.

Table 5: Ablation Studies on F1 Score

Method	MNLI-mismatch							MNLI-match						
	0.1%	0.2%	0.5%	1%	2%	5%	10%	0.1%	0.2%	0.5%	1%	2%	5%	10%
L-only	23.3	26.6	44.6	47.6	77.3	85.3	87.6	23.2	26.1	44.3	61.6	78.3	86.0	91.2
No-GAN	22.7	29.3	43.2	65.2	71.5	88.3	89.2	23.4	34.3	39.5	60.4	74.7	88.1	88.5
BERT	22.2	26.4	48.3	58.3	65.5	72.7	73.9	26.3	32.1	48.3	58.2	65.4	70.4	71.9
B-ASAL	**29.0**	**42.3**	**55.7**	**77.0**	**86.7**	**89.7**	**91.7**	**29.3**	**39.7**	**74.3**	**77.7**	**83.3**	**91.0**	**94.2**



(a) No fine-tuning vs. Fine-tuning BERT (b) MEn vs. MMEn

Fig. 2: Two of our Ablation Studies

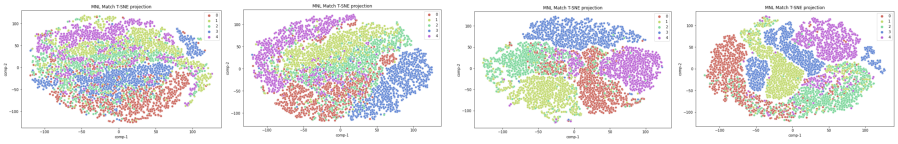
Table 6: Comparisons When Labels are Partially Available at Training

Method	20 out of 50 classes					40 out of 50 classes				
	2%	5%	10%	20%	30%	2%	5%	10%	20%	30%
Random	13.0	29.2	49.4	64.8	75.2	22.7	39.2	57.8	64.8	67.1
En	15.4	27.9	41.5	64.3	64.8	31.1	41.6	59.0	64.4	70.3
Div	15.5	35.0	46.3	65.7	65.9	31.2	48.0	59.6	64.7	71.5
B-ASAL	**16.5**	**38.1**	**50.5**	**66.9**	**75.6**	**36.1**	**40.1**	**60.0**	**66.6**	**75.9**

Entropy Optimization For unlabeled data, we perform entropy minimax optimization. Fig. 2(b) plots out the study when minimax optimization (MMEn) is used for our approach vs. when only entropy minimization (MEn) is used for extracting discriminative features. It shows entropy value decreases with the increase of epochs, and the entropy of MMEn is higher than that of MEn, which demonstrates a more effective optimization of the objective function.

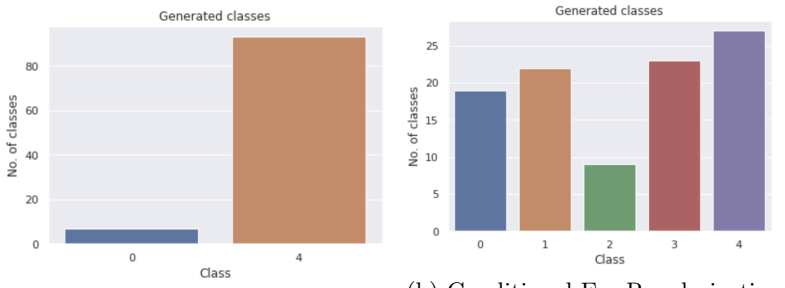
Robustness In our method, Classifier (C) chooses data with high uncertainty for labelling, while Discriminator (D) differentiates labeled data from unlabeled ones. To evaluate the effectiveness, we studied QC-Fine dataset and randomly use 20 classes and 40 classes (out of 50) in the initial training set as a labeled pool. The results are shown in Table 6, which demonstrates our model is less affected when initially labeled data don't well represent the entire data distribution.

Discriminative Feature Visualization We demonstrate the discriminative features learned from the model. Figure 3 shows the results using 10% labeled MNL-match data for training: (a) feature encoded from Pre-trained Bert before tuning, (b) feature learned at first epoch, (c) feature learned by BERT classifier,



(a) Pre-Train: Initialization (b) Model 1st epoch: acc=0.32 (c) Sup. classifier: acc=0.89 (d) B-ASAL: acc=0.94

Fig. 3: MNL-Match Feature Visualizaton



(a) No En. Regularization (b) Conditional En. Regularization

Fig. 4: MNL-mismatch: samples generation from G

and (d) feature learned by B-ASAL. It can be seen that B-ASAL generates more discriminative features.

Sample Generation We study the Generator capability and distribution coverage of generated samples. We take MNL-mismatch data (5 classes total) as an example and use 5% annotated data to train our model. The sample generation is compared by having conditional entropy regularization on the generator with not having an entropy regularizer. Figure 4(a) shows the histogram of generated classes without conditional entropy regularize, and Figure 4(b) shows the histogram of generated classes by imposing conditional entropy regularizer. These outputs illustrate that the conditional entropy regularizer helped the generator to generate effective samples to represent true data distribution.

5 Conclusions and Future Work

In this paper, we proposed a BERT encoder-based semi-supervised active learning algorithm, B-ASAL, which guides the fine-tuning of BERT to better fit the training data, creates synthetic data to address data insufficiency problems, and incorporates minimax entropy to differentiate the distribution of labeled data from that of unlabeled data. We introduced a hybrid sampling strategy that selects samples that are most diverse and have high uncertainty from class assignments learned by the multi-class classifier. Our experiments demonstrated significant improvements over the existing state-of-the-art methods. In future, we plan to extend this research to more NLP applications such as question-answering and recommendation systems.

References

1. Chivukula, A.S., Liu, W.: Adversarial deep learning models with multiple adversaries. IEEE Transactions on Knowledge and Data Engineering **31**(6), 1066–1079 (2018)
2. Croce, D., Castellucci, G., Basili, R.: Gan-bert: Generative adversarial learning for robust text classification with a bunch of labeled examples. In: Proceedings of the 58th annual meeting of the association for computational linguistics. pp. 2114–2119 (2020)
3. Dai, Z., Yang, Z., Yang, F., Cohen, W.W., Salakhutdinov, R.R.: Good semi-supervised learning that requires a bad gan. Advances in neural information processing systems **30** (2017)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 (2018)
5. Dor, L.E., Halfon, A., Gera, A., Shnarch, E., Dankin, L., Choshen, L., Danilevsky, M., Aharonov, R., Katz, Y., Slonim, N.: Active learning for bert: an empirical study. In: EMNLP. pp. 7949–7962 (2020)
6. Ebrahimi, S., Gan, W., Chen, D., Biamby, G., Salahi, K., Laielli, M., Zhu, S., Darrell, T.: Minimax active learning. arXiv preprint arXiv:2012.10467 (2020)
7. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)
8. Jacobs, P.F., Maillette de Buy Wenniger, G., Wiering, M., Schomaker, L.: Active learning for reducing labeling effort in text classification tasks. In: Benelux Conference on Artificial Intelligence. pp. 3–29. Springer (2021)
9. Li, X., Roth, D.: Learning question classifiers: the role of semantic information. Natural Language Engineering **12**(3), 229–249 (2006)
10. Mohammad, S.M., Bravo-Marquez, F., Salameh, M., Kiritchenko, S.: Semeval-2018 Task 1: Affect in tweets. In: SemEval-2018. New Orleans, LA, USA (2018)
11. Saito, K., Kim, D., Sclaroff, S., Darrell, T., Saenko, K.: Semi-supervised domain adaptation via minimax entropy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8050–8058 (2019)
12. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Advances in neural information processing systems **29** (2016)
13. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. arXiv preprint arXiv:1708.00489 (2017)
14. Sinha, S., Ebrahimi, S., Darrell, T.: Variational adversarial active learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5972–5981 (2019)
15. Springenberg, J.T.: Unsupervised and semi-supervised learning with categorical generative adversarial networks. arXiv preprint arXiv:1511.06390 (2015)
16. Williams, A., Nangia, N., Bowman, S.R.: A broad-coverage challenge corpus for sentence understanding through inference. arXiv preprint arXiv:1704.05426 (2017)
17. Wolf, T., Debut, L., et al.: Huggingface's transformers: State-of-the-art natural language processing. arXiv:1910.03771 (2019)
18. Yang, P., Liu, W., Yang, J.: Positive unlabeled learning via wrapper-based adaptive sampling. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 3273–3279 (2017)
19. Yuan, M., Lin, H.T., Boyd-Graber, J.: Cold-start active learning through self-supervised language modeling. arXiv:2010.09535 (2020)