Task-Aware Variational Adversarial Active Learning

Kwanyoung Kim[1,4], Dongwon Park[1], Kwang In Kim[2,3], Se Young Chun[1,2,5,†]
[1]Dept of EE, UNIST, [2]AIGS, UNIST, [3]Dept of CSE, UNIST, [4]Dept of Bio & Brain Eng, KAIST,
[5]Dept of ECE, INMC, Seoul National University, South Korea
cubeyoung@kaist.ac.kr, dong1@unist.ac.kr, kimki@unist.ac.kr, sychun@snu.ac.kr

Abstract

Often, labeling large amount of data is challenging due to high labeling cost limiting the application domain of deep learning techniques. Active learning (AL) tackles this by querying the most informative samples to be annotated among unlabeled pool. Two promising directions for AL that have been recently explored are task-agnostic approach to select data points that are far from the current labeled pool and task-aware approach that relies on the perspective of task model. Unfortunately, the former does not exploit structures from tasks and the latter does not seem to well-utilize overall data distribution. Here, we propose task-aware variational adversarial AL (TA-VAAL) that modifies task-agnostic VAAL, that considered data distribution of both label and unlabeled pools, by relaxing task learning loss prediction to ranking loss prediction and by using ranking conditional generative adversarial network to embed normalized ranking loss information on VAAL. Our proposed TA-VAAL outperforms state-of-the-arts on various benchmark datasets for classifications with balanced / imbalanced labels as well as semantic segmentation and its task-aware and task-agnostic AL properties were confirmed with our in-depth analyses.

1. Introduction

Deep learning has achieved remarkable performance in various computer vision tasks such as classification [18, 13], object detection [27, 26], and semantic segmentation [19, 4] due to massive datasets with annotations such as ImageNet for image classification [7] and PASCAL VOC for classification, detection, segmentation [9]. Obtaining good annotations is challenging and has often been a large-scale project. Moreover, there are often cases where labeling massive amount of data is even more challenging or infeasible due to high labeling cost such as labeling by experts [8] or long labeling time per large-scale sample such as videos [1] or pathology images [3]. Labeling cost seems to be a factor to limit the scope of applicability of deep learning to more research areas and more institutes with less labeling budget.

Active learning (AL) is one of the approaches to overcoming limited labeling budget by selecting data to label for the best possible performance [30, 11]. AL has been widely investigated in relatively traditional machine learning settings [5, 34, 2, 21, 23, 30, 14, 20, 36, 32, 25] and recently in deep learning settings [11, 29, 38, 40, 35, 31, 16].

Existing AL approaches can be categorized into two groups: Task-agnostic (or distribution-based) and task-aware methods. Suppose that our goal is to learn a functional model f that maps from the input domain $\mathcal{X}$ to the corresponding output domain $\mathcal{Y}$, each equipped with the corresponding probability distributions $P(x)$ and $P(y)$. Task-agnostic approaches select data instances to label by exploiting the input distribution $P(x)$. These are especially effective in identifying *influential* points, *e.g.* these lying in high-density regions such that once labeled, large numbers of *neighboring* samples can benefit from propagating these labels [20, 39, 29, 31]. A major drawback of these approaches is that they do not take account how outputs y depend on inputs x: For example, for classifications, it would be more effective to label data instances that lie in the vicinity of decision boundaries than these lying in high-density regions where most data points belong to the same class.

Task-aware approaches explicitly address this limitation by modeling such dependence, *e.g.* via estimating the conditional distribution $P(y|x)$. These are effective in identifying *difficult* data points (*e.g.* these close to decision boundaries) [36, 14, 11, 35, 40]. However, they do not directly consider how the labeled samples make influence on the entire dataset. Further, as $P(y|x)$ is unknown a priori, the label selection process has to rely on the learner f as a surrogate to $P(y|x)$ but such a learner might be inaccurate at the early stage of AL, thereby providing a poor estimate of $P(y|x)$.

Recently, there was an attempt (SRAAL) to combine the task-aware and task-agnostic approaches with a uncertainty indicator and with a unified representation for both labeled and unlabeled data [42]. Even though SRAAL achieved state-of-the-art performance, it did not use the information about the loss that is directly related to the given task [40]

[†]Corresponding author.

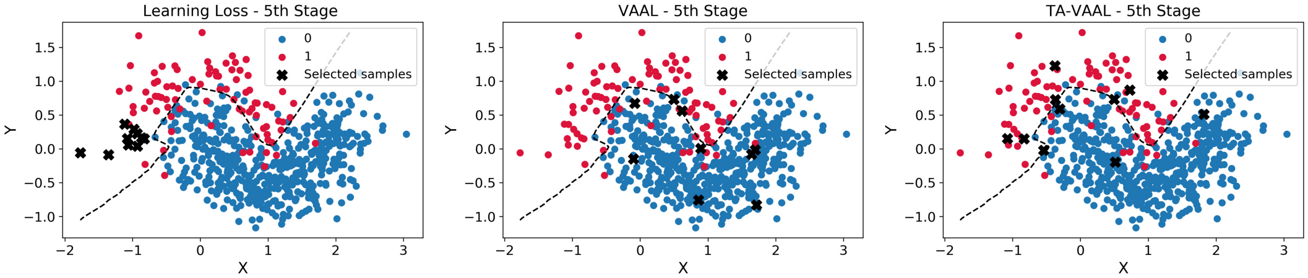


Figure 1: Visual results of active learning methods (Learning loss [40], VAAL [31], our TA-VAAL) on imbalanced toy example at the 5th stage. *Red* and *blue* dots indicate samples assigned to class 0 and 1, respectively. Ten samples at that stage (denoted by *black* cross) were selected using each method. The oracle decision boundary of the model is shown as a black dash line. Learning loss identified difficult samples near the decision boundary and VAAL found influential samples over the entire set. Our TA-VAAL selected samples that are both difficult (near decision boundary) and influential (over the entire set).

and its task learner seems to be limited only to VAE-type networks with a latent space for its unified representation. Moreover, its implementation is not yet available online.

In this paper, we propose a novel alternative AL scheme that combines the benefits of these two groups of approaches. Specifically, our algorithm builds upon two recent state-of-the-art approaches: *Variational adversarial active learning* (VAAL) [31] models how adding labels to selected data points make influence on the entire set. As a model-agnostic approach, this method does not exploit the structure $P(y|x)$ of the problem at hand. We address this by combining it with the recent *learning loss* approach [40]. This algorithm learns to estimate the errors of the predictions (loss) made by the learner and therefore helps identify difficult data points.

Here is the summary of our contributions:

- Proposing to relax the goal of loss prediction module [40] from accurate loss prediction to loss *ranking* prediction, which is still directly connected to the task. This relaxation leads to altering the loss for learning prediction module to remove margins for ranking and to add ranking loss in [28].
- Proposing *Task-Aware Variational Adversarial Active Learning* (TA-VAAL) to embed the normalized ranking loss information from any given task learner (with or without latent space) on the latent space of VAAL [31] via ranking conditional generative adversarial network (RankCGAN) [28] to reshape the latent space of it. This approach is significantly more robust than the original learning loss approach, especially at the early stage. By combining these two algorithms with our embedding strategy, our method offers the capability of identifying *difficult* and *influential* data points (see Figure 1; see Section 4 for details).
- Demonstrating the superior performance of our proposed TA-VAAL over state-of-the-art works (Learning loss [40], VAAL [31], Coreset [29], Monte-Carlo dropout [11]) by evaluating on various classification benchmark datasets: CIFAR10, CIFAR100 that have the same number of images per class (balanced) as well as Caltech101, modified CIFAR10 that has different numbers of images for classes (imbalanced), and on Cityscapes semantic segmentation benchmark dataset and by in-depth empirical analyses to confirm our proposed approach. Our codes are available at https://github.com/cubeyoung/TA-VAAL.

2. Related Works

There have been a number of AL works to select the most informative samples and we categorized them into two approaches: task-aware (or model uncertainty-based) and task-agnostic approaches. The former is using unlabeled data in a passive way while the latter is using unlabeled data in an active way. In other words, the former has sample selection rules that are not affected by unlabeled data, but simply are applied to it, while the latter exploits both labeled and unlabeled data to build up sample selection rules (or train deep neural networks (DNNs) for them).

Task-aware approach defined and used metrics for sample selection with labeled data. For example, the minimum distance from decision boundaries (or classification hyperplanes) can be used to select samples with the most ambiguous classification results [34, 2]. Empirical risk is used to minimize an upper bound of true risk so that one can query the most informative samples that are the most uncertain and representative [36]. Bayesian active learning by disagreement (BALD) maximizes the mutual information between model predictions and model parameters [14]. Then, BALD was extended to accommodate DNNs with Bayesian neural network and Monte-Carlo dropout [11]. Bayesian generative active deep learning was proposed to utilize both labeled data and labeled fake data to train a classifier (or a task-learner) as well as a discriminator for real / fake images [35]. Even though [35] uses deep generative models or VAEs, it does not use unlabeled data for training unlike our proposed TA-VAAL. Yoo and Kweon [40] proposed an AL loss method that attaches "loss prediction module" to a task-learner. The loss prediction module was trained to estimate target losses of unlabeled samples that were used as surrogates for model uncertainty based on feature information in mid-layers.

Task-agnostic approach exploits both labeled and unlabeled data to form sample selection rules so that selected samples are far from the distribution of labeled data and have the most well-representative information of unlabeled pool. Clustering unlabeled data could help to choose samples from diverse clusters, not from one or small number of clusters [23]. Expected error reduction using hierarchical clustering was developed for active sampling in a semi-supervised framework [20]. An objective function with diversity constrain was proposed to impose diversity on the subset of data pool for multi-class AL [39]. Recently, there have also been works on task-agnostic AL with DNNs. Core-set approach was proposed that minimizes the distance between labeled data and unlabeled data pool with intermediate feature information of trained convolutional DNN models [29]. Gudovskiy *et al.* [12] proposed to minimize distribution shift between unlabeled training set and weakly-labeled validation set for semi-supervised AL. Sinha *et al.* [31] proposed VAAL to train VAE that captures the representing information of both labeled and unlabeled data with adversarial learning to discriminate unlabeled samples from labeled data using the latent space information in the VAE.

An extended version of VAAL (SRAAL) [42] was proposed to combine task-aware and task-agnostic approaches with a uncertainty indicator and with a unified representation for both labeled and unlabeled data. However, SRAAL did not use the final information on task ($e.g.$, loss [40]), but used intermediate task information such as the latent space information from the task learner. Moreover, its task learner seems to be limited only to VAE-type networks with a latent space for its unified representation. In the meanwhile, our proposed TA-VAAL is a novel alternative to combine both task-aware and task-agnostic approaches that is another extension of task-agnostic VAAL to incorporate direct task related information (ranking loss) into the VAE framework. In addition, our TA-VAAL does not have any structural restriction for the task learner. We demonstrated that our proposed framework can accommodate both local task-related information and global data distribution structure so that high performance and reliability can be jointly achieved.

3. Method

Let us denote the pool of labeled data and annotations by (X_L, Y_L) and the pool of unlabeled data by X_U. The goal of AL is to select samples from X_U with limited label budget, to annotate them to yield pairs of sample / annotation (x^*, y^*), and to add them to (X_L, Y_L) for the best possible performance of a given task learner T (DNN parametrized by θ_T). (X_L, Y_L) will grow in size every stages. The task learner T will be trained by minimizing the loss $\sum_{(x_L,y_L)\in(X_L,Y_L)} l_L$ at each stage where $l_L = L_T(\hat{y}_L, y_L)$ is a loss value at (x_L, y_L) and $\hat{y}_L = T(x_L)$ is a predicted label.

3.1. Task loss prediction module as "Ranker"

Yoo and Kweon [40] proposed loss prediction module (LPM), denoted by Θ_{loss}, to predict the loss value $\hat{l}_U = \Theta_{loss}(x_U)$ for $x_U \in X_U$ without ground truth labels. LPM consists of global average pooling, fully connected layer and ReLU to predict unknown $l_U = L_T(T(x_U), y_U)$ where y_U is unknown ground truth label (indicated as "?" in Figure 2). Since the task loss is usually decreasing over epochs, using the mean squared error (MSE) caused scaling issue. To avoid that, [40] considered the difference between two losses and thus (X_L, Y_L) was re-grouped into a set of pairs $(x_P, l_P) = \{(x_i, x_j), (l_i, l_j)\}$. Then, the loss for LPM is $-(2/B)\sum_{i=1}^{B/2} \max(0, -I_i \cdot (\hat{l}_i - \hat{l}_j) + \epsilon))$ where $I_i = +1$ if $l_i > l_j$ and -1 otherwise, and ϵ is a positive scalar that was set to 1. If there were B elements in the original (X_L, Y_L), this re-grouped set had $B/2$ elements of (x_P, l_P). This enabled LPM to be trained to yield accurate loss values.

In this work, we relax the goal of LPM from predicting accurate loss values to estimating accurate ranking loss information. In other words, our proposed LPM will less care loss value itself, but more care relative loss rankings. For this purpose, we propose to exploit RankCGAN [28] to connect between task-agnostic VAAL [31] and task-aware learning loss [40]. While the LPM in [40] and the concept of "Ranker" in RankCGAN [28] both utilized the difference between two predicted loss values in training losses, the former aimed to predict accurate losses and the latter focused on predicting "ranking" of the loss values. Thus, our ranking loss is:

$$L_R(\hat{l}_P, l_P) = -(2/B)\sum_{i=1}^{B/2}\{I_i \log[\sigma(\hat{l}_i - \hat{l}_j)] \\ +(1 - I_i)\log[1 - \sigma(\hat{l}_i - \hat{l}_j)]\} \quad (1)$$

where $I_i = +1$ if $l_i > l_j$ and 0 otherwise, $\hat{l}_i = R(x_i)$ with R being the Ranker (DNN parametrized by θ_R) that predicts loss, and σ is the Sigmoid function. Rather than directly predicting *loss* itself [40] (thus, the loss includes a margin ϵ to trust more on loss value itself and to emphasize less on preserving rankings), Ranker in our TA-VAAL is predicting relative *rankings* of losses that can be embedded into the latent space of VAAL with the conditional latent variable r with normalization via the Sigmoid function (thus, the rankings are strictly preserved with (1)). This choice has been motivated by 1) the observation that the relative comparisons of the target attributes are often easier to learn and predict than the absolute attribute values [28], and 2) for AL, ranking the data points to label is often sufficient [40]. Moreover, while the learning loss in [40] is non-differentiable piecewise linear, our ranking loss (1) is a smooth differentiable function that potentially has nice convergence properties for gradient based optimizations (see supplemental).

Finally, the total loss function of task learner with our

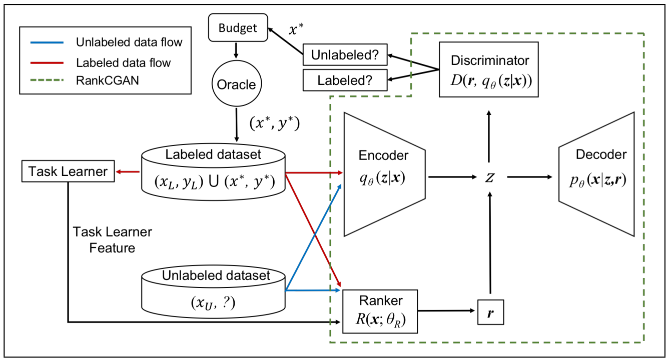


Figure 2: A schematic diagram of our TA-VAAL: VAAL is effective at capturing the overall influence of labels propagated to the *entire* distribution, but is agnostic to the nature of task at hand, *i.e.*, VAAL is independent of the task labels predicted or provided as ground-truths. By injecting the capability of capturing fine-grained task label rank information, TA-VAAL helps focus on both influential and informative (or difficult) labels and adjust how they are propagated.

proposed Ranker R is expressed as

$$L_{total} = L_T(\hat{y}_L, y_L) + \eta L_R(R(x_P), l_P) \tag{2}$$

where η is a scaling parameter. We empirically found that training a task learner with this ranking loss (1) was more stable and yielded better performance than the original learning loss [40] (see the ablation study in Section 5.1).

3.2. Proposed task-aware VAAL (TA-VAAL)

Figure 2 illustrates our proposed TA-VAAL that introduces Ranker, modifies the latent space of the original VAAL by incorporating a rank variable r from Ranker, and inputs the normalized loss ranking information r to both the decoder of VAAL and the discriminator of VAAL to select samples from unlabeled data pool. Our proposed framework allows us to control the latent subspace with loss ranking predictions so that the overall latent space can be reshaped.

TA-VAAL is obtained with the following optimization:

$$\begin{aligned}\min_{q_\theta} \max_{D} \mathbb{E}_{z_L \sim p_{x_L}} [\log(D(r_L, q_\theta(z_L|x_L)))] \\ + \mathbb{E}_{z_U \sim p_{x_U}} [\log(1 - D(r_U, q_\theta(z_U|x_U)))]\end{aligned} \tag{3}$$

where q_θ is an encoder of the VAE, and z_L and z_U belong to the latent spaces for labeled and unlabeled data, respectively, and r_L and r_U are the normalized outputs of the Ranker from labeled and unlabeled data, respectively. $z_L \sim p_{x_L}$ implies $z_L = q_\theta(z_L|x_L)$ with $x_L \sim p_{data}$ and $z_U \sim p_{x_U}$ is similar to $z_L \sim p_{x_L}$. By removing the rank information r_L and r_U from (3), TA-VAAL boils down to VAAL that offers the capability of modeling the *global* data distribution, but does not exploit the information gained from the task. Using the loss of the learner as a surrogate to such task information can help [40], but assessing the *individual* losses does not explicitly model how data instances make influence to each other and thus, they can be prone to noise and outliers.

```
Algorithm 1: Training pipeline of our TA-VAAL
Given: learning rates $\zeta_1, \zeta_2, \zeta_3$, # of epochs $N$;
Input   : labeled data $(x_L, y_L)$, unlabeled data $x_U$;
Initialize: network parameters $\theta_T, \theta_R, \theta, \theta_D$;
for $n = 1$ to $N$ do
    if $n = 1$ then $r_L, r_U \sim \mathcal{U}(0, 1)$;
    else $r_L \leftarrow R(x_L; \theta_R)$ , $r_U \leftarrow R(x_U; \theta_R)$;
    $L_{total} \leftarrow L_T + \eta L_R$
    if $n \leq 0.8N$ then $\theta_T \leftarrow \theta_T - \zeta_1 \nabla_{\theta_T} L_{total}$,
     $\theta_R \leftarrow \theta_R - \zeta_1 \nabla_{\theta_R} L_{total}$;
    else $\theta_T \leftarrow \theta_T - \zeta_1 \nabla_{\theta_T} L_{total}$;
    $L_{VAE} \leftarrow L_{VAE}^{trans} + \lambda L_{VAE}^{adv}, \theta \leftarrow \theta - \zeta_2 \nabla_\theta L_{VAE}$
    $L_D$ using (6),   $\theta_D \leftarrow \theta_D - \zeta_3 \nabla_{\theta_D} L_D$
end
```

Here, we conjecture that task-related information further improves the overall performance of AL. Our TA-VAAL bridges between model uncertainty-based and data distribution-based approaches in a *tight* way by using conditional GAN (RankCGAN) so that the information about data distribution accounts for model uncertainty information (predicted loss ranking). Our TA-VAAL will have an advantage to use more data (unlabeled data) over typical task-aware approach trained without unlabeled data and can offer significant improvements (see Section 4).

3.3. Training details for TA-VAAL

The objective function L_{VAE}^{trans} of the conditional VAE with ranking for learning features of both labeled and unlabeled pools can be formulated as

$$\begin{aligned}\mathbb{E}[\log p_\theta(x_L|z_L, r_L)] - \beta \mathrm{KL}(q_\theta(z_L|x_L)||p_z) \\ + \mathbb{E}[\log p_\theta(x_U|z_U, r_U)] - \beta \mathrm{KL}(q_\theta(z_U|x_U)||p_z)\end{aligned} \tag{4}$$

where q_θ and p_θ are the encoder and decoder of the VAE, p_z is Gaussian distribution, β is a hyper-parameter, and $\mathrm{KL}(\cdot)$ is Kullback-Leibler distance. Another function for training is the conditional adversarial loss to represent both $q_\theta(z_L|x_L)$ and $q_\theta(z_U|x_U)$ with the same distribution from labeled and unlabeled pools. The objective function L_{VAE}^{adv} is

$$-\mathbb{E}[\log D(r_L, q_\theta(z_L|x_L)) + \log D(r_U, q_\theta(z_U|x_U))] \tag{5}$$

where z_L, z_U belong to the latent spaces for labeled, unlabeled data. The final training loss is $L_{VAE}^{trans} + \lambda L_{VAE}^{adv}$.

The discriminator D with ranking is learned to distinguish if latent space variable belongs to labeled pool. The loss is

$$\begin{aligned}L_D = & -\mathbb{E}[\log D(r_L, q_\theta(z_L|x_L))] \\ & - \mathbb{E}[\log(1 - D(r_U, q_\theta(z_U|x_U)))].\end{aligned} \tag{6}$$

The smaller the output D is, the more likely unlabeled sample is selected. The overall training pipeline is in Algorithm 1 where ∇_θ denotes the gradient with respect to θ.

After training TA-VAAL, the data points $(x_1^*, ..., x_b^*)$ to be labeled at each stage are selected by

$$(x_1^*, ..., x_b^*) = \underset{(x_1, ..., x_b) \subset X_U}{\arg\min} \; D(R(x_U), q_\theta(z_U|x_U)). \quad (7)$$

The detail of selecting samples is described in supplemental. A subset method, replacing X_U in (7) with a random subset of X_U, was used to reduce outliers as suggested in [40].

4. Experimental Results

4.1. An illustrative example: binary classification

Dataset. The dataset with 2-dimensional features for binary classification was generated using scikit-learns makemoons library [24] as illustrated in Figure 1: The noise option was set to 0.2 and the dataset size was 500 samples for one class and 50 samples for the other class, eventually constituting a dataset of size 550 (imbalance ratio of class is ×10).

Implementation details. For the task learner, a 3-layer multi-layer perceptron (MLP) was used and Adam optimizer with learning rate 0.1 was used. For Ranker, a single layer perceptron was attached to the mid-layer of the task learner. For VAE, a 2-layer MLP with ReLU was used for encoder and decoder, respectively, and the discriminator comprised of a 2-layer MLP. For both the VAE and the discriminator, the Adam optimizer with learning rate 0.01 was used. All epochs were set to 100 and active sampling was performed starting from 20 random samples with 10 sample increment.

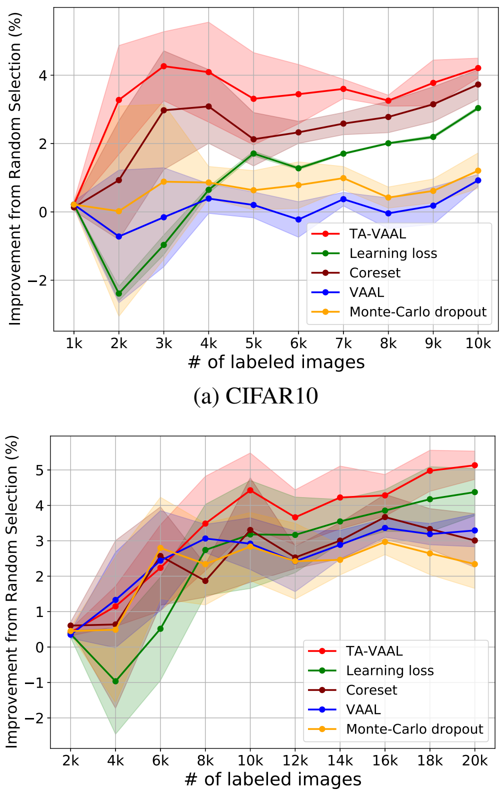
Results. Task-aware learning loss method tends to select difficult and informative samples that are all close to decision boundary, but are often clustered due to no information about global distribution even after performing a random subset method (left subfigure of Figure 1). In contrast, task-agnostic VAAL tends to select influential samples that are spread spatially, but that are often far from decision boundary due to no task related information (middle of Figure 1). Our proposed TA-VAAL tends to select difficult (close to decision boundary) and influential (over the entire distribution) samples due to task-aware ranking information and data distribution-based VAAL, respectively (right of Figure 1).

4.2. Image classification on balanced datasets

(Balanced) benchmark datasets. We evaluated our proposed TA-VAAL method on various (balanced) benchmark datasets: CIFAR10 [17], CIFAR100 [17] that consist of 50,000 / 10,000 32 × 32 images, for training / testing with 10, 100 classes, respectively. Each class includes the same number of images (5,000 / 1,000 images per class for CIFAR10, 500 / 100 images per class for CIFAR100). The numbers of initial random samples were 1,000 / 2,000 with the query sizes 1,000, 2,000 at each stage on CIFAR10 / CIFAR100, respectively. The results of the evaluations for our TA-VAAL on SVHN [22] and Fashion-MNIST [37] datasets are available in supplemental. The subset method was used to avoid overlaps and to introduce diversity in samples: the subset size was set to 10 times larger than the query size.

Implementation details. For training, 32 × 32 random crop from 36 × 36 zero-padded images, normalization with mean and standard deviation of training set, and horizontal flip / flop augmentation were used. ResNet18 [13] was used for all task learners and stochastic gradient descent (SGD) was used with momentum 0.9 and weight decay 0.005. Learning rate was 0.1 for the first 160 epochs and then 0.01 for the last 40 epochs. For VAE, a modified Wasserstein autoencoder [33] for taking ranking information was used and the discriminator was constructed as a 5-layer MLP. For both the VAE and the discriminator, Adam optimizer [15] with learning rate 5×10^{-4} was used. Mini-batch size was 128 and the total epochs were 200 for all datasets.

Results. Six AL methods were evaluated: random sam-



(a) CIFAR10

(b) CIFAR100

Figure 3: Mean accuracy improvements with standard deviation (shaded) of AL methods from random sampling baseline over the number of labeled samples. The absolute accuracy values are provided in the supplemental material. Our TA-VAAL outperformed others on (balanced) CIFAR10 in all stages and on (balanced) CIFAR100 after a few stages.

pling (baseline), Monte-Carlo dropout [11], Core-set [29], Learning loss [40], VAAL [31] and our TA-VAAL. Figure 3 presents the number of labeled images (active samples) versus the mean (line) and standard deviation (shaded region) for accuracy improvements from the baseline with 5 trials.

In Figure 3a for CIFAR10, learning loss method yielded even lower accuracy than baseline at early stages possibly due to insufficient labeled samples to capture the uncertainty of model and yielded good performance at later stages once sufficient labeled data was used to train learning loss. VAAL achieved better performance than learning loss possibly due to massive unlabeled data. Our proposed TA-VAAL outperformed other state-of-the-art methods at all stages.

In Figure 3b for CIFAR100, all active learning methods outperformed baseline (random sampling) in most stages. Learning loss method exhibited similar tendency (low performance at early stages, then high performance at later stages) on both CIFAR10 and CIFAR100. After 3k labeled samples, our TA-VAAL outperformed all compared state-of-the-art active learning methods substantially.

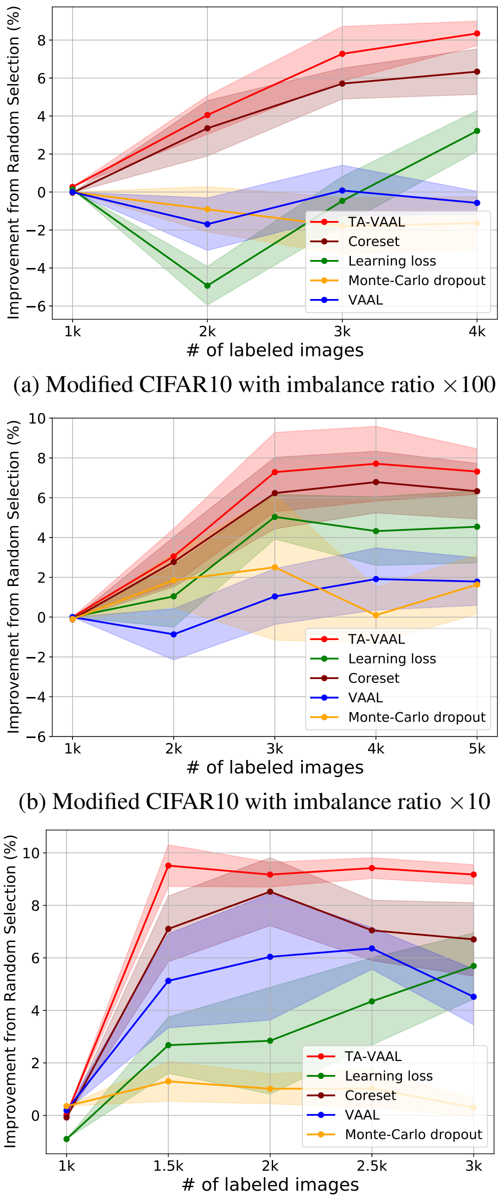
Discussions. 1) Core-set yielded comparable performance to our TA-VAAL on CIFAR10. However, Core-set is computationally demanding as compared to ours since core-set required 7.5 times more selection time per sample than our TA-VAAL. 2) We used much smaller initial data size / budget (1,000/1,000) than the original VAAL setting (5,000/2,500) on CIFAR10 [31] and VAAL yielded similar performance as random sampling for all cases in our setting (see supplemental for detail). 3) The performances of learning loss and ours yielded slightly higher or lower mean accuracies at the first stage due to additional LPM attached to the task learner. We performed additional study to show that this additional loss is not the most important factor for the overall performance improvements of our proposed method (see supplemental).

4.3. Image classification on imbalanced datasets

Datasets. We performed experiments on imbalanced datasets whose sizes are different for classes. Modified CIFAR10's were constructed by randomly reducing the number of samples that were associated with the first 5 classes. Imbalance ratio was defined as the ratio of the number of samples for the first 5 classes to the number of samples for the last 5 classes. Imbalance ratios of 10 and 100 were used.

Further evaluation was performed on Caltech101 [10] that consists of 9,144 images with about 300 ×200 and 101 categories with imbalanced labels (40 - 800 images per class, mostly 50). We set 8,125 images for training and 1,019 images for testing. Initially, 1,000 images were randomly selected and AL budget was 500 images per stage.

Implementation detail. For modified CIFAR10, we used the same implementation for CIFAR10. For Caltech101, we performed random horizontal flips for data augmentation and resized images to 224×224 for training. ResNet18 was used as the task learner and SGD was used with learning rate of 0.01. Modified Wasserstein autoencoder and 5-layer MLP were used for conditional VAE and discriminator, respectively. Adam optimizer with learning rate 1 $\times 10^{-4}$ was used with minibatch size 16 and 200 epochs. The details on hyper-parameters are described in supplement material.



(a) Modified CIFAR10 with imbalance ratio ×100

(b) Modified CIFAR10 with imbalance ratio ×10

(c) Caltech101

Figure 4: Mean accuracy improvements with standard deviation (shaded) of AL methods from random sampling baseline over the number of labeled samples on imbalanced datasets. Our TA-VAAL outperformed others on modified CIFAR10 with different imbalance ratios and Caltech101 in all stages.

Results. Figures 4a and 4b illustrate the mean and standard deviation accuracy improvements from random sampling baseline (5 trials) over the number of labeled images on two modified CIFAR10 with imbalance ratios of ×10 (less imbalanced) and ×100 (more imbalanced). Note that reduced data size in imbalanced datasets limited the maximum number of experiments up to 4k-5k labeled samples. Our proposed TA-VAAL outperformed all other state-of-the-art methods over all stages with more improvement margins for more imbalanced dataset (×100 imbalance ratio). Note that even though the final dataset sizes were 4k and 5k, there were some classes with 50 total images per class for imbalance ratio ×100. In this challenging case, our TA-VAAL still yielded improvements over other methods including random sampling baseline. See supplemental for absolute accuracy over the number of labeled images for all methods.

Figure 4c presents the number of labeled images vs. the mean and standard deviation for accuracy improvements from random sampling (5 trials) on (naturally imbalanced) Caltech101. Our TA-VAAL outperformed other state-of-the-art methods over all stages substantially. These results show the capability of our TA-VAAL in a more realistic setting with more classes with label imbalance and larger images.

4.4. Semantic segmentation on Cityscapes

Dataset. AL was performed for semantic segmentation on Cityscapes [6], a large-scale video dataset of street scenes, including 3,475 frames with instance segmentation annotations. Following [41], we converted labels into 19 classes. Initial label pool size was 200 with budget size 200 per stage.
Implementation detail. For training, we performed random horizontal flips for data augmentation similar to classification tasks. We adopt DRN [41] as task learner for image segmentation and SGD was used with learning rate 1×10^{-3}. Modified Wasserstein autoencoder and 5-layer MLP were used for conditional VAE and discriminator, respectively. Adam with learning rate 1 $\times 10^{-4}$ was used with mini-batch size 4 and total epoch 100. See supplemental for details.
Results. Six AL methods were evaluated including random sampling (baseline), Monte-Carlo dropout [11], Coreset [29], Learning loss [40], VAAL [31] and our TA-VAAL. Figure 5 shows the number of labeled images versus the mean IoU (Intersection Over Union) of 3 trials. We observed that our TA-VAAL outperformed all compared methods at all sampling stages. VAAL yielded better performance than random sampling and learning loss at all stages, but our TA-VAAL outperformed other methods with substantial margins at all sampling stages. This demonstrated the benefits from the ranking loss information of task learner to select the most informative samples from unlabeled pool.

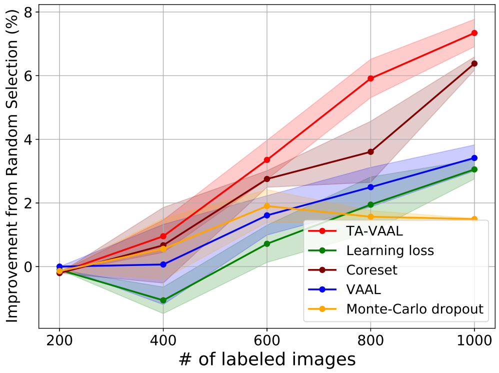


Figure 5: Relative accuracy improvements from random selection for semantic segmentation on Cityscape dataset.

5. Empirical Analyses

5.1. Ablation studies

Figure 6 shows the performance results of our proposed methods with and without proposed components / structures along with other state-of-the-art methods. The means and standard deviations of 5 trials were reported. Firstly, learning loss method with proposed ranking loss (1), called learning loss_v2, yielded substantially higher performances at later AL stages and comparable performances at early stages to the original learning loss. Thus, it seems that using our proposed loss (1) for accurate loss ranking prediction seems advantageous over using the original LPM loss for accurate loss prediction. Another study is to incorporate ranking information into VAAL by using the original learning loss architecture, rather than our proposed Ranker (1). This combination of VAAL+learning loss still yielded substantially better performances than VAAL over all stages. However, that was not able to yield better performance than the original learning loss method at later stages.

5.2. On selected samples of active learning

Figure 7a shows the bar graphs for the number of labeled images (selected samples) versus the entropies of the num-

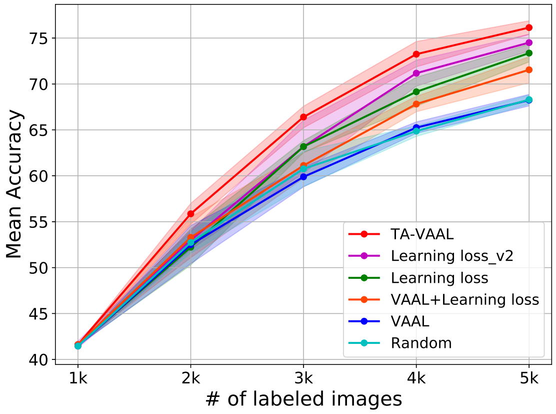
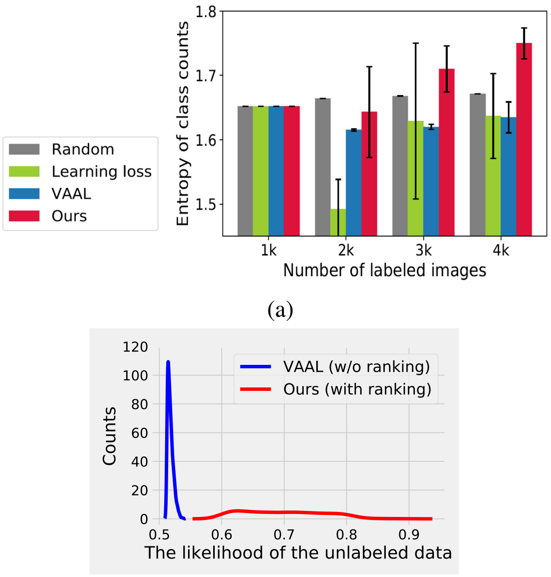


Figure 6: The results of ablation study by selectively removing core components (modified CIFAR10 with imbalanced ratio ×10): Learning loss_v2 is ours without VAAL. VAAL+learning loss is ours with the original learning loss.



(a)

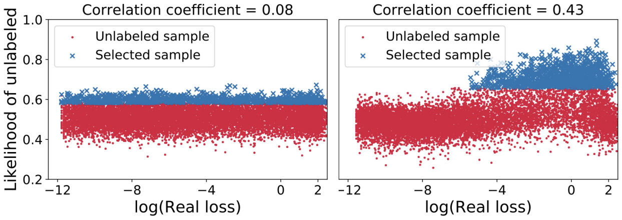
(b)

Figure 7: For the modified CIFAR10 with imbalance ratio ×100, (a) Bar graphs of number of labeled images vs. data class count entropy. (b) Likelihood of unlabeled data vs. number of samples at the last stage.

ber of selected samples over 10 classes (class counts). The higher the entropy is, the more uniform samples over classes are selected. Figure 7a shows that our proposed method selected samples with high data class count entropy on a severely imbalanced dataset. These results can provide insights to explain the performance results in Figure 4a. For example, learning loss method yielded substantially low performance at 2k stage in Figure 4a due to its data selection with low data count entropy over classes at that stage as illustrated in Figure 7a. This is possibly due to limited number of data for certain classes in the case of imbalance ratio ×100 so that task learner in learning loss method was not well-trained. However, our TA-VAAL was able to select good samples at the same stage due to the structure from VAAL to exploit overall data distribution so that good performance and high data class count entropy were able to be achieved.

Figure 7b shows the likelihood of unlabeled data from the discriminator D to select data points at the last stage. VAAL that takes latent space values as discriminator input yielded concentrated count distribution of the likelihood (from the output of D) at the last stage so that active learning selection became almost random, while our proposed method that takes latent space values along with ranking information for D yielded a wide range of likelihood distribution so that sample selection was more reliable and yielded good performances as illustrated in Figures 7a and 4a, 4b.

Figure 8 shows the graphs for (true) real loss (representing task-aware model uncertainty) vs. likelihood of data remaining unlabeled (representing task-agnostic data distribution) for VAAL and our proposed TA-VAAL. Note that both VAAL and our TA-VAAL methods select data points that have the highest estimated likelihood of unlabeled data. While task-agnostic VAAL selected samples with a wide range of real losses as illustrated in Figure 8(a), our proposed TA-VAAL was able to choose samples with relatively high real loss values thanks to the reshaped latent space by ranking information on real losses as shown in Figure 8(b). Thus, TA-VAAL seems to result in higher performances in various tasks over VAAL as in Figure 3.



(a) VAAL

(b) Our TA-VAAL

Figure 8: Relationships between real loss (task model uncertainty) and likelihood of data remaining unlabeled (task-agnostic data distribution) in (a) VAAL and (b) our TA-VAAL. We use the model from the last AL stage on imbalanced CIFAR-10. While task-agnostic VAAL selected samples with a wide range of real loss values, our TA-VAAL chose samples with relatively high real loss values.

6. Conclusion

We proposed TA-VAAL, a novel AL framework that simultaneously takes advantage of both task-agnostic data distribution-based AL and task-aware model uncertainty-based approach that exploits any generic task learner (with or without latent space). Our TA-VAAL exploits VAAL that considered data distribution of both label and unlabeled pools by incorporating LPM and RankCGAN concepts into VAAL by relaxing loss prediction with a ranker for ranking loss information. We demonstrate that our TA-VAAL outperforms state-of-the-art AL methods on various classification benchmark datasets such as CIFAR-10, CIFAR-100 and Caltech-101 for balanced and imbalanced cases and on Cityscapes semantic segmentation dataset. Our in-depth analyses also confirm that our TA-VAAL effectively takes advantage of both task-aware and task-agnostic AL approaches.

Acknowledgment

This work was supported in part by a grant of Korea Health Technology R&D Project through Korea Health Industry Development Institute (KHIDI), funded by Ministry of Health & Welfare, Republic of Korea (HI18C0316) and in part by the 2020 Research Fund(1.200093.01) and the U-K BRAND Research Fund(2.210040.01) of UNIST(Ulsan National Institute of Science & Technology).

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A large-scale video classification benchmark. *arXiv:1609.08675*, 2016. 1

[2] Klaus Brinker. Incorporating diversity in active learning with support vector machines. In *ICML*, pages 59–66, 2003. 1, 2

[3] Gabriele Campanella, Matthew G Hanna, Luke Geneslaw, Allen Miraflor, Vitor Werneck Krauss Silva, Klaus J Busam, Edi Brogi, Victor E Reuter, David S Klimstra, and Thomas J Fuchs. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 25(8):1301–1309, 2019. 1

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018. 1

[5] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996. 1

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016. 7

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009. 1

[8] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017. 1

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 1

[10] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006. 6

[11] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *ICML*, pages 1183–1192, 2017. 1, 2, 6, 7

[12] Denis Gudovskiy, Alec Hodgkinson, Takuya Yamaguchi, and Sotaro Tsukizawa. Deep active learning for biased datasets via fisher kernel self-supervision. In *CVPR*, pages 9041–9049, 2020. 3

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 5

[14] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv:1112.5745*, 2011. 1, 2

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[16] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *NeurIPS*, pages 7024–7035, 2019. 1

[17] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. 5

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 1

[19] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 1

[20] Oisin Mac Aodha, Neill DF Campbell, Jan Kautz, and Gabriel J Brostow. Hierarchical subquery evaluation for active learning on a graph. In *CVPR*, pages 564–571, 2014. 1, 3

[21] Prem Melville and Raymond J Mooney. Diverse ensembles for active learning. In *ICML*, pages 584–591, 2004. 1

[22] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, pages 1–9, 2011. 5

[23] Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *ICML*, pages 623–630, 2004. 1, 3

[24] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 5

[25] Robert Pinsler, Jonathan Gordon, Eric Nalisnick, and José Miguel Hernández-Lobato. Bayesian batch active learning as sparse subset approximation. In *NeurIPS*, pages 6356–6367, 2019. 1

[26] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, pages 779–788, 2016. 1

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 1

[28] Yassir Saquil, Kwang In Kim, and Peter Hall. Ranking CGANs: subjective control over semantic image attributes. In *BMVC*, 2018. 2, 3

[29] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, 2018. 1, 2, 3, 6, 7

[30] Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009. 1

[31] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *ICCV*, pages 5972–5981, 2019. 1, 2, 3, 6, 7

[32] Iiris Sundin, Peter Schulam, Eero Siivola, Aki Vehtari, Suchi Saria, and Samuel Kaski. Active learning for decision-making from imbalanced observational data. In *ICML*, pages 6046–6055, 2019. 1

[33] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. In *ICLR*, 2018. 5

[34] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2(Nov):45–66, 2001. 1, 2

[35] Toan Tran, Thanh-Toan Do, Ian Reid, and Gustavo Carneiro. Bayesian generative active deep learning. In *ICML*, pages 6295–6304, 2019. 1, 2

[36] Zheng Wang and Jieping Ye. Querying discriminative and representative samples for batch mode active learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–23, 2015. 1, 2

[37] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017. 5

[38] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *MICCAI*, pages 399–407, 2017. 1

[39] Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113(2):113–127, 2015. 1, 3

[40] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *CVPR*, pages 93–102, 2019. 1, 2, 3, 4, 5, 6, 7

[41] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *CVPR*, pages 472–480, 2017. 7

[42] Beichen Zhang, Liang Li, Shijie Yang, Shuhui Wang, Zheng-Jun Zha, and Qingming Huang. State-relabeling adversarial active learning. In *CVPR*, pages 8756–8765, 2020. 1, 3