# Deep Active Learning: Unified and Principled Method for Query and Training

**Changjian Shui** [1] **Fan Zhou** [1] **Christian Gagné** [1 2] **Boyu Wang** [3]

## Abstract

In this paper, we are proposing a unified and principled method for both the querying and training processes in deep batch active learning. We are providing theoretical insights from the intuition of modeling the interactive procedure in active learning as distribution matching, by adopting the Wasserstein distance. As a consequence, we derived a new training loss from the theoretical analysis, which is decomposed into optimizing deep neural network parameters and batch query selection through alternative optimization. In addition, the loss for training a deep neural network is naturally formulated as a min-max optimization problem through leveraging the unlabeled data information. Moreover, the proposed principles also indicate an *explicit* uncertainty-diversity trade-off in the query batch selection. Finally, we evaluate our proposed method on different benchmarks, consistently showing better empirical performances and a better time-efficient query strategy compared to the baselines.

## 1. Introduction

Deep neural networks (DNNs) achieved unprecedented success for many supervised learning tasks such as image classification and object detection. Although DNNs are successful in many scenarios, there still exists an obvious limitation: the requirement for a large set of labeled data. To address this issue, *Active Learning* (AL) appears as a compelling solution by searching the most informative data points (batch) to label from a pool of unlabeled samples in order to maximize prediction performance.

How to search the most informative samples in the context of DNN? A common solution is to apply DNN's output confidence score as an uncertainty acquisition function to conduct the query (Settles, 2012; Gal et al., 2017; Haussmann et al., 2019). However, a well-known issue for uncertainty-based sampling in AL is the so-called *sampling bias* (Dasgupta, 2011): the current labeled points are not representative of the underlying distribution. For example, as shown in Fig. 1, let us assume that the very few initial samples we obtain lie in the two extreme regions. Then based on these initial observations, the queried samples nearest to the currently estimated decision boundary will lead to a final sub-optimal risk of $10\%$ instead of the true optimal risk of $5\%$. This will be even more severe in high dimensional and complex datasets, which are common when DNNs are employed.

Recent works have considered obtaining a diverse set of samples for training deep learning with a reduced sampling bias. For example, (Sener and Savarese, 2018) constructed the core-sets through solving the $K$-center problem. But the search procedure itself is still computationally expensive as it requires constructing a large distance matrix from unlabeled samples. More importantly, it might not be a proper choice particularly for a large-scale unlabeled pool and a *small* query batch, where it is hard to cover the entire data (Ash et al., 2019).

Instead of focusing exclusively on either uncertainty or diversity instances when determining the query, following a hybrid strategy can be more appropriate. For example, (Yin et al., 2017) heuristically selected a portion of samples according to the uncertainty score for exploitation and the remaining portion used random sampling for exploration. (Ash et al., 2019) collected samples whose gradients span a diverse set of directions for implicitly considering these two. Since such hybrid strategies empirically showed improved performance, a goal of our paper is to derive the query strategy that explicitly considers the uncertainty-diversity trade-off in a principled way.

Moreover, in the context of deep AL, the available large set of unlabeled samples may be helpful to construct a good feature representation that would potentially allow to improve performance. In order to further promote better results, the question to answer is how we can additionally design a loss for optimizing DNN's weights that would leverage from the unlabeled samples during the training.

To address this question, a promising line of work is

[1]Université Laval, Canada [2]Mila, Canada CIFAR AI Chair [3]University of Western Ontario, Vector Institute. Correspondence to: Changjian Shui <changjian.shui.1@ulaval.ca>.

*2nd ICML Workshop on Human in the Loop Learning*, 2020. 

to integrate the training with a *deep generative model* which naturally focuses on the unlabeled data information (Goodfellow et al., 2014; Kingma and Welling, 2013). Only a few works strode in this direction, notably (Sinha et al., 2019) who empirically adopted a $\beta$-VAE to construct the latent variables. Then they adopted the intuition from (Gissin and Shalev-Shwartz, 2019) of searching the diverse unlabeled batch for samples that do not look like the labeled samples, through an adversarial training based on the $\mathcal{H}$-divergence (Ben-David et al., 2010). In spite of some good performance, this approach still concentrated on empirically designing the training loss, simply adopting the $\mathcal{H}$-divergence based query strategy. In particular, *the formal justifications still remain elusive*, and *the $\mathcal{H}$-divergence may not be a proper metric for measuring the diversity of the query batch* (see Fig. 2), which will be verified in our paper.

In this paper, we are proposing a *unified and principled* approach for *both* a fast querying and a better training procedure in deep AL, relying on the use of labeled and unlabeled examples. We derived the theoretical analysis through modeling the interactive procedure in AL as *distribution matching* by adopting the Wasserstein distance. We also analytically reveal that the Wasserstein distance is better at capturing the diversity, compared with the most common $\mathcal{H}$-divergence. From the theoretical result, we derived the loss from the distribution matching, which is naturally decomposed into two stages: optimization of DNN parameters and query batch selection, through alternative optimization.

For the stage of training DNN, the derived loss indicates a min-max optimization problem by leveraging the unlabeled data. More precisely, this involves a maximization of the critic function so as to distinguish the labeled and unlabeled empirical distributions based on the Wasserstein distance, while the feature extractor function aims, on the contrary, to confound the distributions (minimization of empirical distribution divergence). In the query stage, the loss for batch selection *explicitly* indicates the uncertainty-diversity trade-off. For the uncertainty, we want to find the samples with low prediction confidence over two different interpretations: the highest least prediction confidence score and the uniform prediction score (Section 3.4). As for the diversity, we want to find the unlabeled batch holding a larger transport cost w.r.t. the labeled set under Wasserstein distance (i.e. not looking like the current labelled ones), which has been shown as a good metric for measuring diversity.

Finally, we tested our proposed method on different benchmarks, showing a consistently improved performance, particularly in the initial training, and a much faster query strategy compared with the baselines. The results reaffirmed the benefits and potential of deriving unified principles for Deep AL. We also hope it will open up a new avenue for rethinking and designing query efficient and principled Deep AL algorithms in the future.

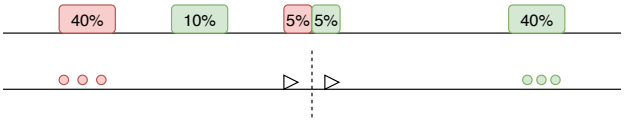



*Figure 1.* Sampling bias in AL (Dasgupta, 2011). In the one dimensional binary classification problem (prediction red/green), the data generation distribution consists of four uniform intervals. Red/Green dots: the initial observations; dotted line: estimated decision boundary from the initial samples; triangles: querying samples according to the uncertainty based strategies w.r.t. current decision boundary.

## 2. Active Learning as Distribution Matching

In supervised learning, observations $\hat{\mathcal{D}}$ are i.i.d. generated by the underlying distribution $\mathcal{D}$ and a labeling function $h^\star$, i.e. $\{(x_i, h^\star(x_i))\}_{i=1}^N$ with $x_i \sim \mathcal{D}$. While in AL, the querying sample is not an i.i.d. procedure w.r.t. $\mathcal{D}$ — otherwise it will be simple random sampling. Thus we assume in AL that the query procedure is an i.i.d. empirical process following a distribution $\mathcal{Q} \neq \mathcal{D}$. For example, in the disagreement based approach, $\mathcal{Q}$ can be somehow regarded as a uniform distribution over the disagreement region. Then the interactive procedure can be viewed as estimating a proper $\mathcal{Q}$ to control the generalization error w.r.t. $(\mathcal{D}, h^\star)$.

### 2.1. Preliminaries

We define the hypothesis $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ over $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \in [0, 1]$, and loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$. The expected risk w.r.t. $\mathcal{D}$ is $R_{\mathcal{D}}(h) = \mathbb{E}_{x\sim\mathcal{D}}\ell(h(x), h^\star(x))$ and empirical risk $\hat{R}_{\mathcal{D}}(h) = \frac{1}{N}\sum_{i=1}^N \ell(h(x_i), y_i)$. $\mathbb{P}(\mathcal{X})$ is the set of all probability measures over $\mathcal{X}$. We assume that the loss $\ell$ is symmetric, $L$-Lipschitz and $M$-upper bounded and $\forall h \in \mathcal{H}$ is at most $H$-Lipschitz function.

**Wasserstein Distance** Given two probability measures $\mathcal{D} \in \mathbb{P}(\mathcal{X})$ and $\mathcal{Q} \in \mathbb{P}(\mathcal{X})$, the *optimal transport* (or Monge-Kantorovich) problem can be defined as searching for a probabilistic coupling (joint probability distribution) $\gamma \in \mathbb{P}(\Omega \times \Omega)$ for $x_{\mathcal{D}} \sim \mathcal{D}$ and $x_{\mathcal{Q}} \sim \mathcal{Q}$ that are minimizing the cost of transport w.r.t. some cost function $c$:

$$\operatorname*{argmin}_{\gamma} \int_{\mathcal{X}\times\mathcal{X}} c(x_{\mathcal{D}}, x_{\mathcal{Q}})^p d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}),$$
$$\text{s.t.} \quad \mathbf{P}^+\#\gamma = \mathcal{D}; \quad \mathbf{P}^-\#\gamma = \mathcal{Q},$$

where $\mathbf{P}^+$ and $\mathbf{P}^-$ is the marginal projection over $\Omega\times\Omega$ and $\#$ denotes the push-forward measure. The $p$-Wasserstein

distance between $\mathcal{D}$ and $\mathcal{Q}$ for any $p \geq 1$ is defined as:

$$W_p^p(\mathcal{D}, \mathcal{Q}) = \inf_{\gamma \in \Pi(\mathcal{D}, \mathcal{Q})} \int_{\mathcal{X} \times \mathcal{X}} c(x_\mathcal{D}, x_\mathcal{Q})^p d\gamma(x_\mathcal{D}, x_\mathcal{Q}),$$

where $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is the cost function of transportation of one unit of mass $x$ to $y$ and $\Pi(\mathcal{D}, \mathcal{Q})$ is the collection of all joint probability measures on $\mathcal{X} \times \mathcal{X}$ with marginals $\mathcal{D}$ and $\mathcal{Q}$. Throughout this paper, we only consider the case of $p = 1$, i.e. the Wasserstein-1 distance and the cost function as Euclidean ($\ell_2$) distance.

**Labeling Function Assumption** Some theoretical works show that AL cannot improve the sample complexity in the worst case, thus identifying properties of the AL paradigm is beneficial (Urner and Ben-David, 2013). For example, (Urner et al., 2013) defined a formal *Probabilistic Lipschitz* condition, in which the Lipschitzness condition is relaxed and formalizes the intuition that *under suitable feature representation, the probability of two close points having different labels is small* (Urner and Ben-David, 2013). We adopt the Joint Probabilistic Lipschitz property, which can be viewed as an extension of (Pentina and Ben-David, 2018) and is also coherent with (Courty et al., 2017).

**Definition 1.** *Let $\phi : \mathbb{R} \to [0, 1]$. We say labeling function $h^\star$ is $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz if $\mathrm{supp}(\mathcal{Q}) \subseteq \mathrm{supp}(\mathcal{D})$ and for all $\lambda > 0$ and all distribution coupling $\gamma \in \Pi(\mathcal{D}, \mathcal{Q})$:*

$$\mathbb{P}_{(x_\mathcal{D}, x_\mathcal{Q}) \sim \gamma}[|h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| > \lambda \|x_\mathcal{D} - x_\mathcal{Q}\|_2] \leq \phi(\lambda), \quad (1)$$

*where $\phi(\lambda)$ reflects the decay property. (Urner et al., 2013) showed that the faster the decay of $\phi(\lambda)$ with $\lambda \to 0$, the better the labeling function and the easier it is to learn the task.*

### 2.2. Bound related Querying Distribution

In this part, we will derive the relation between the querying and the data generation distribution.

**Theorem 1.** *Supposing $\mathcal{D}$ is the data generation distribution and $\mathcal{Q}$ is the querying distribution. If the loss $\ell$ is symmetric, $L$-Lipschitz; $\forall h \in \mathcal{H}$ is at most $H$-Lipschitz function and the underlying labeling function $h^\star$ is $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz; then the expected risk w.r.t. $\mathcal{D}$ can be upper bounded by:*

$$R_\mathcal{D}(h) \leq R_\mathcal{Q}(h) + L(H + \lambda) W_1(\mathcal{D}, \mathcal{Q}) + L\phi(\lambda). \quad (2)$$

See the proof in the supplementary material. From Eq. (2), the expected risk of $\mathcal{D}$ is upper bounded by the expected risk w.r.t. the query distribution $\mathcal{Q}$, the Wasserstein distance $W_1(\mathcal{D}, \mathcal{Q})$, and the labeling function property $\phi(\lambda)$. That means a desirable query should hold a small expected risk with a better matching the original distribution $\mathcal{D}$ (diversity).

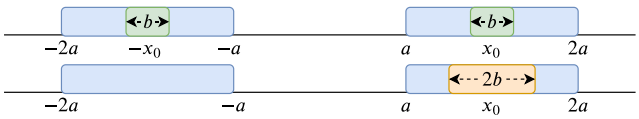



*Figure 2.* $\mathcal{H}$-divergence vs. Wasserstein distance for $\mathcal{D}$-$\mathcal{Q}$ distribution matching. The desirable query distribution should be more diverse (first row) for avoiding *sampling bias* (second row). The computational result shows that $\mathcal{H}$-divergence is not a proper metric to measure query diversity while Wasserstein is.

**Non-Asymptotic Analysis** Moreover, we can extend the non-asymptotic analysis of Theorem 1 since we generally have finite observations. The proof is also provided in the supplementary material.

**Corollary 1.** *Supposing we have the finite observations which are i.i.d. generated from $\mathcal{D}$ and $\mathcal{Q}$: $\hat{D} = \frac{1}{N}\sum_{i=1}^{N} \delta\{x_\mathcal{D}^i\}$ and $\hat{Q} = \frac{1}{N_q}\sum_{i=1}^{N_q} \delta\{x_\mathcal{Q}^i\}$ with $N_q \leq N$. Then with probability $\geq 1 - \delta$, the expected risk w.r.t. $\mathcal{D}$ can be further upper bounded by:*

$$R_\mathcal{D}(h) \leq \hat{R}_\mathcal{Q}(h) + L(H + \lambda) W_1(\hat{\mathcal{D}}, \hat{\mathcal{Q}}) + L\phi(\lambda) + 2L\mathrm{Rad}_{N_q}(h) + \kappa(\delta, N, N_q),$$

*where $\kappa(\delta, N, N_q) = \mathcal{O}(N^{-1/s_d} + N_q^{-1/s_q} + \sqrt{\frac{\log(1/\delta)}{N}} + \sqrt{\frac{\log(1/\delta)}{N_q}})$ with some positive constants $s_d$ and $s_q$. $\mathrm{Rad}_{N_q}(h) = \mathbb{E}_{S \sim \mathcal{Q}^{N_q}} \mathbb{E}_{\sigma_1^{N_q}}[\sup_h \frac{1}{N_q}\sum_{i=1}^{N_q} \sigma_i h(x_i)]$ is the expected Rademacher complexity generally with $\mathrm{Rad}_{N_q}(h) = \mathcal{O}(\sqrt{\frac{1}{N_q}})$ (e.g (Mohri et al., 2018)).*

### 2.3. Why Wasserstein Distance

In the context of deep active learning, current work such as (Gissin and Shalev-Shwartz, 2019; Sinha et al., 2019) generally explicitly or implicitly adopted the idea of $\mathcal{H}$-divergence (Ben-David et al., 2010): $d_\mathcal{H}(\mathcal{D}, \mathcal{Q}) = 1 - 2\epsilon$, with $\epsilon$ the prediction error when training a binary classifier to *discriminate* the observations sampling from the query and original distribution. Thus a smaller error facilitates the separation of the two distributions with larger $\mathcal{H}$-divergence and vice versa.

However, we should notice that in AL, $\mathrm{supp}(\mathcal{Q}) \subseteq \mathrm{supp}(\mathcal{D})$, thus $\mathcal{H}$-divergence may not be a good metric for indicating the diversity property of the querying distribution. On the contrary, Wasserstein distance reflects an optimal transport cost for moving one distribution to another. A smaller transport cost means a better coverage of the distribution $\mathcal{D}$.

For a better understanding of this problem, we give an illustrative example by computing the exact $\mathcal{H}$-divergence and Wasserstein-1 distance in one-dimension, shown

in Fig. 2. More specifically, we have three uniform distributions: $\mathcal{D}_1$ the original data distribution, $\mathcal{D}_2$, $\mathcal{D}_3$ two different query distributions:

$$\mathcal{D}_1 \sim \mathcal{U}\big([-2a,-a]\cup[a,2a]\big),$$
$$\mathcal{D}_2 \sim \mathcal{U}\big([-x_0-\frac{b}{2},-x_0+\frac{b}{2}]\cup[x_0-\frac{b}{2},x_0+\frac{b}{2}]\big),$$
$$\mathcal{D}_3 \sim \mathcal{U}\big([x_0-b,x_0+b]\big).$$

In AL, we can further assume $\text{supp}(\mathcal{D}_2)\subseteq\text{supp}(\mathcal{D}_1)$, $\text{supp}(\mathcal{D}_3)\subseteq\text{supp}(\mathcal{D}_1)$ and $a>b>0$. For $\mathcal{H}$-divergence, we set the classifier as a threshold function $f(x)=\mathbf{1}\{x\geq p\}$. Then we can compute the exact $d_{\mathcal{H}}(\cdot,\cdot)$ and $W_1(\cdot,\cdot)$:

$$\begin{aligned} d_{\mathcal{H}}(\mathcal{D}_1,\mathcal{D}_2) &= d_{\mathcal{H}}(\mathcal{D}_1,\mathcal{D}_3) \\ \min_{x_0} W_1(\mathcal{D}_1,\mathcal{D}_3) &> \max_{x_0} W_1(\mathcal{D}_1,\mathcal{D}_2) \end{aligned} \tag{3}$$

From Eq. (3), the $\mathcal{H}$-divergence indicates the same divergence result where Wasserstein-1 distance exactly captures the property of diversity: *more diverse query distribution $\mathcal{Q}$ means smaller Wasserstein-1 distance* $W_1(\mathcal{D},\mathcal{Q})$.

## 3. Practical Deep Batch Active Learning

We have discussed the interactive procedure as the distribution matching and also showed that Wasserstein distance is a *proper* metric for measuring the diversity during distribution matching. Based on the aforementioned analysis, in the batch active learning problem, we have labelled data $\hat{L}=\frac{1}{L}\sum_{i=1}^{L}\delta\{x_i^l\}$ and its labels $\{y_i^l\}_{i=1}^{L}$, unlabelled data $\hat{U}=\frac{1}{U}\sum_{i=1}^{U}\delta\{x_i^u\}$ and total distribution $\hat{\mathcal{D}}=\hat{L}\cup\hat{U}$ with partial labels $\{y_i^l\}_{i=1}^{L}$. The goal of AL at each interaction is: 1) find a batch $\hat{B}=\frac{1}{B}\sum_{i=1}^{B}\delta\{x_i^b\}$ with $x_i^b\in\hat{U}$ during the query; 2) find a hypothesis $h\in\mathcal{H}$ such that:

$$\min_{\hat{B},h}\hat{R}_{\hat{L}\cup\hat{B}}(h)+\mu W_1(\hat{\mathcal{D}},\hat{L}\cup\hat{B}). \tag{4}$$

Eq.(4) follows the principles (upper bound) from Theorem 1 and Corollary 1. Moreover, if we fix the hypothesis $h$, the sampled batch holds the following two requirements simultaneously:

1. Minimize the empirical error. We will show later it is related to uncertainty based sampling.
2. Minimize the Wasserstein-1 distance w.r.t. original distribution, which encourages a better distribution matching of $\hat{\mathcal{D}}$.

### 3.1. Min-Max Problem in DNN

Based on Eq.(4), we can extend the loss to the deep representation learning scenario, since directly estimating the Wasserstein-1 distance through solving optimal transport for complex and large-scale data is still a challenging and open problem. Then inspired by (Arjovsky et al., 2017), we then adopt the min-max optimizing through training the DNN. Namely, according to Kantorovich-Rubinstein duality, Eq. (4) can be reformulated as:

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h,\hat{B}}\max_{\boldsymbol{\theta}^d}\ \hat{R}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h)+\mu\hat{E}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d), \tag{5}$$

where $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^h$, $\boldsymbol{\theta}^d$ are parameters corresponding to the *feature extractor*, *task predictor* and *distribution critic*; $\hat{R}$ is the predictor loss and $\hat{E}$ is the adversarial (min-max) loss.

We further denote the *parametric task prediction* function $h(x,y,(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h))\equiv h(x,y):\mathcal{X}\times\mathcal{Y}\to(0,1]$ with $\sum_y h(x,y)=1$ and the *parametric critic* function $g(x,(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d))\equiv g(x):\mathcal{X}\to[0,1]$ with restricting $g(x)$ to the 1-Lipschitz function (Kantorovich-Rubinstein theorem). Then each term in Eq. (5) can be expressed as:

$$\hat{R}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h)=\mathbb{E}_{(x,y)\sim\hat{L}\cup\hat{B}}\ell(h(x,y)),$$
$$\hat{E}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d)=\mathbb{E}_{x\sim\hat{\mathcal{D}}}[g(x)]-\mathbb{E}_{x\sim\hat{L}\cup\hat{B}}[g(x)].$$

### 3.2. Two-stage Optimization

Then through some computation, we can decompose Eq. (5) into three terms:

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h,\hat{B}}\max_{\boldsymbol{\theta}^d}\Bigg[\underbrace{\frac{1}{L+B}\sum_{(x,y)\in\hat{L}}\ell(h(x,y))}_{\text{Training: Prediction Loss}}$$
$$\underbrace{+\mu\left(\frac{1}{L+U}\sum_{x\in\hat{U}}g(x)-\left(\frac{1}{L+B}-\frac{1}{L+U}\right)\sum_{x\in\hat{L}}g(x)\right)}_{\text{Training: Min-max Loss}}$$
$$\underbrace{+\frac{1}{L+B}\left(\sum_{(x,y^?)\in\hat{B}}\ell(h(x,y^?))-\mu\sum_{x\in\hat{B}}g(x)\right)}_{\text{Query}}\Bigg], \tag{6}$$

where the critic function $g(x)$ is 1-Lipschitz and $L,U,B$ are the size of labeled, unlabeled, and query data. $y^?$ is called the *agnostic-label*, since it is not available during the query stage. Then from Eq. (6), each interaction of AL can be naturally decomposed into two stages (optimizing DNN and batch selection), through alternative optimization.

### 3.3. Training DNN

In the training stage, we used all of the observed data to optimize the neural network parameters:

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h}\max_{\boldsymbol{\theta}^d}\left[\frac{1}{L+B}\sum_{(x,y)\in\hat{L}}\ell(h(x,y))+\frac{\mu}{L+U}\sum_{x\in\hat{U}}g(x)-\mu\left(\frac{1}{L+B}-\frac{1}{L+U}\right)\sum_{x\in\hat{L}}g(x)\right], \quad (7)$$

with restricting $g(x)$ to the 1-Lipschitz function. Instead of only minimizing the prediction error, the proposed approach naturally leveraged the unlabeled data information through a min-max training. More intuitively, the critic function $g$ aims to evaluate how probable it is that the sample comes from the labeled or unlabeled parts[1]. According to the loss, given a fixed $g$, when $g(x) \to 1$ meaning that it is highly probable that the samples come from the unlabeled set $x \in \hat{U}$ and vice versa. Since $B < U$, thus $\frac{1}{L+B} - \frac{1}{L+U} > 0$, means the proposed adversarial loss being always valid.

Based on Eq. (7), we call this framework the Wasserstein Adversarial Active Learning (WAAL) in our deep batch AL. The labeled $\hat{L}$ and unlabeled data $\hat{U}$ pass a common feature extractor, then $\hat{L}$ will be used in the prediction and $\hat{L}$, $\hat{U}$ together will be used in the min-max (adversarial) training. In the practical deep learning, we apply the cross entropy loss: $\ell(x, y) = -\log(h(x, y))$.

**Redundancy Trick** One can directly apply gradient descent to optimize Eq. (7) on the whole dataset. Actually, we generally apply the mini-batch based SGD approach in training the DNN[2]. While a practical concern during the adversarial training procedure is the unbalanced label and unlabeled data during the training procedure. Thus we propose the *redundancy trick* to solve this concern. For abuse of notation, we denote the unbalanced ratio $\gamma = \frac{U}{L}$ and the query ratio $\alpha = \frac{B}{L}$, with the adversarial loss simplified as:

$$\mu'\left(\frac{1}{U}\sum_{x\in\hat{U}}g(x)-\frac{1}{\gamma}\frac{\gamma-\alpha}{1+\alpha}\frac{1}{L}\sum_{x\in\hat{L}}g(x)\right),$$

with $\mu' = \frac{\gamma}{1+\gamma}\mu$.

Then following the *redundancy trick* for optimizing the adversarial loss, we keep the same mini-batch size $S$ for labelled and unlabeled observations. Due to the existence of the unbalanced data, we simply conduct a sampling with replacement to construct the training batch for the labeled data, then divided by the unbalanced ratio $\gamma$. For each training batch, the adversarial loss can be rewritten as:

$$\min_{\boldsymbol{\theta}^f}\max_{\boldsymbol{\theta}^d}\mu'\left(\frac{1}{S}\sum_{x\in\hat{U}_S}g(x)-C_0\frac{1}{S}\sum_{x\in\hat{L}_S}g(x)\right), \quad (8)$$

where $\hat{U}_s$ $\hat{L}_s$ are unlabeled and labeled training batch and $C_0 = \frac{1}{\gamma^2}\frac{\gamma-\alpha}{1+\alpha}$ is the "bias coefficient" in deep active adversarial training. For example, if there exist 1K labeled samples, 9K unlabeled samples and a current query batch budget of 1K, then we can compute $C_0 \approx 0.05$ so as to control excessive reusing of the labelled dataset.

### 3.4. Query Strategy

The second stage over the unlabeled data aims to find a querying batch such that:

$$\operatorname*{argmin}_{\hat{B}\subset\hat{U}}\frac{1}{L+B}\left[\sum_{(x,y^?)\in\hat{B}}\ell(h(x,y^?))-\mu\sum_{x\in\hat{B}}g(x)\right], \quad (9)$$

where $y^?$ is the agnostic-label.

**Agnostic-label upper bound loss indicates uncertainty** Since we do not known $y^?$ during the query, we can instead optimize an *upper bound* of Eq. (9). In the classification problem with cross entropy loss, suppose that we have $\{1, \ldots, K\}$ possible outputs with $\sum_{y\in\{1,\ldots,K\}} h(x, y) = 1$, then we have upper bounds Eq. (10) and (11), which both reflect the uncertainty measures with different interpretations.

1. Minimizing over the single worst case upper bound indicates the sample with the *highest least prediction confidence score*:

$$\min_x \ell(h(x,y^?)) \le \min_x \max_{y\in\{1,\ldots,K\}} -\log(h(x,y)). \quad (10)$$

For example, we have two samples with a binary decision score $h(x_1, \cdot) = [0.4, 0.6]$ and $h(x_2, \cdot) = [0.3, 0.7]$. Since $\max_y -\log(h(x_1, y)) < \max_y -\log(h(x_2, y))$, we will choose $x_1$ as the query since the least prediction label confidence $0.4$ is higher. Intuitively such a sample seems uncertain since the least label prediction confidence is high[3].

[1] We should point out that this is the high-level intuition. More specifically, the critic parameter $\boldsymbol{\theta}^d$ of $g$ tries to maximize and the feature parameter $\boldsymbol{\theta}^f$ of $g$ tries to minimize the adversarial loss according to the Wasserstein metric. Moreover the proposed min-max loss differs from the standard Wasserstein min-max loss since they hold different weights ("bias coefficient").

[2] We have referred to this as the *training/mini batch* to avoid any confusion with the querying batch mentioned before.

[3] In the binary classification problem, it recovers the least prediction confidence score approach (Baseline 2), which is a common strategy in AL.

2. Minimizing over $\ell_1$ norm upper bound indicates the sample with a *uniformly of prediction confidence score*:

$$\min_x \ell(h(x, y^?)) \leq \min_x \sum_{y\in\{1,\dots,K\}} -\log(h(x, y)). \quad (11)$$

Intuitively if the sample's prediction confidence trend is more uniform, the more uncertain the sample will be. We can also show the min arrives when the output score is uniform, as shown in the supplementary material.

We would like to point out that the upper bounds proposed in Eq. (10,11) are additive, i.e. we can apply any convex combination for these two losses as the hybrid uncertain query strategy.

**Critic output indicates diversity** As for the critic function $g(x) : \mathcal{X} \to [0, 1]$ from the adversarial loss, if the critic function output trends to $g(x) \to 1$, it means $x \in \hat{U}$ and vice versa. Then according to the query loss, we want to select the batch with higher critic values $g(x)$, meaning they look more different than the labelled samples under the Wasserstein metric.

If the unlabeled samples look like the labeled ones (small $g(x)$ with $x \in \hat{U}$), then under some proper conditions (such as *Probabilistic Lipschitz Condition* in def. 1), such examples can be more easily predicted because we can infer them from their very near neighbours' information.

On the contrary, the unlabeled samples with high $g(x)$ under the current assumption cannot be effectively predicted by the current labeled data (far away data). Moreover, the $g(x)$ is trained through Wasserstein distance based loss, shown as a proper metric for measuring the diversity. Therefore the query batch with higher critic value ($g(x)$) means a larger transport cost from the labeled samples, indicating that it is more informative and represents diversity.

**Remark** The aforementioned two terms in the query strategy indicate an explicit *uncertainty* and *diversity* trade-off. Uncertainty criteria can reduce the empirical risk but leading to a potential sampling bias. While the diversity criteria can improve the exploration of the distribution while might be inefficient for a small query batch. Our query approach naturally combines these two, for choosing the samples with prediction uncertainty and diversity. Moreover, since Eq. (9) is additive, we can easily estimate the query batch through the greedy algorithm.

### 3.5. Proposed Algorithm

Based on the previous analysis, our proposed algorithm includes a training stage [Eq. (7) and (8)] and a query stage [Eq. (9), (10) and (11)] for solving Eq. (5) or (6). We only show the learning algorithm for one interaction in Algorithm 1, then the remaining interactions will be repeated accordingly.

**Algorithm 1** WAAL: one interaction

**Require:** Labeled samples $\hat{L}$, unlabeled samples $\hat{U}$, query budget $B$ and hyper-parameters (learning rate $\eta$, trade-off rate $\mu$, $\mu'$)
**Ensure:** Neural network parameters $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^h$, $\boldsymbol{\theta}^d$

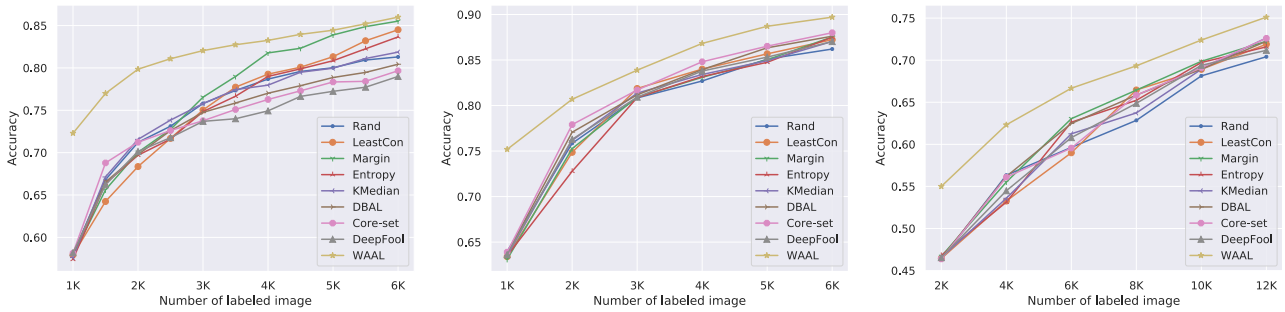
▷ ▷ ▷ **DNN Parameter Training Stage** ◁ ◁ ◁
**for** mini-batch of samples $\{(x^u)\}_{i=1}^S$ from $\hat{U}$ **do**
  Constructing mini-batch $\{(x^l, y^l)\}_{i=1}^S$ from $\hat{L}$ through sampling with replacement (redundancy trick).
  Updating $\boldsymbol{\theta}^h$: $\boldsymbol{\theta}^h = \boldsymbol{\theta}^h - \frac{\eta}{S}\sum_{(x^l,y^l)} \frac{\partial \ell(h((x^l,y^l))}{\partial \boldsymbol{\theta}^h}$
  Updating $\boldsymbol{\theta}^f$: $\boldsymbol{\theta}^f = \boldsymbol{\theta}^f - \frac{\eta}{S}\big(\sum_{(x^l,y^l)} \frac{\partial \ell(h((x^l,y^l))}{\partial \boldsymbol{\theta}^f} + \mu'\{\sum_{x^u} \frac{\partial g(x)}{\partial \boldsymbol{\theta}^f} - C_0 \sum_{x^l} \frac{\partial g(x)}{\partial \boldsymbol{\theta}^f}\}\big)$
  Updating $\boldsymbol{\theta}^d$: $\boldsymbol{\theta}^d = \boldsymbol{\theta}^d + \frac{\eta\mu'}{S}\{\sum_{x^u} \frac{\partial g(x)}{\partial \boldsymbol{\theta}^d} - C_0 \sum_{x^l} \frac{\partial g(x)}{\partial \boldsymbol{\theta}^d}\}$
**end for**
▷ ▷ ▷ **Querying Stage** ◁ ◁ ◁
Applying the convex combination of Eq. (10) and (11) to compute uncertainly score $\mathcal{U}(x^u)$;
Computing diversity score $g(x^u)$;
Ranking the score $\mathcal{U}(x^u) - \mu g(x^u)$ with $x^u \in \hat{U}$, choosing the smallest $B$ samples, forming querying batch $\hat{B}$
▷ ▷ ▷ **Updating** ◁ ◁ ◁
$\hat{L} = \hat{L} \cup \hat{B}$, $\hat{U} = \hat{U} \setminus \hat{B}$

Since the discriminator function $g$ should be restricted in 1-Lipschitz, we add a gradient penalty term to $g$, such as (Gulrajani et al., 2017), to restrict the Lipschitz property.

## 4. Experiments

We start our experiments with a small initial labeled pool of the training set. The initial observation size and the budget size range from $1\% - 5\%$ of the training dataset, depending on the task. Following Alg. 1, the selected batch will be annotated and added into the training set. Then the training process for the next iteration will be repeated on the new formed labeled and unlabeled set *from scratch*.

We evaluate our proposed approach on three object recognition tasks, namely Fashion MNIST (image size: $28 \times 28$) (Xiao et al., 2017), SVHN ($32 \times 32$) (Netzer et al., 2011), CIFAR-10 ($32 \times 32$) (Krizhevsky et al., 2009). For each task we split the whole data into training, validation and testing parts. We evaluate the performance of the proposed algorithm for image classification task by computing the prediction accuracy. We repeat all of the experiments 5 times and report the average value. The details of the experimental




(a) Fashion MNIST (b) SVHN (c) CIFAR-10

*Figure 3.* Empirical performance on Fashion MNIST, SVHN and CIFAR-10 over five repetitions.

| **Method** | LeastCon | Margin | Entropy | $K$-Median | DBAL | Core-set | DeepFool | WAAL |
|---|---|---|---|---|---|---|---|---|
| **Time** | 0.94 | 0.95 | 0.95 | 33.98 | 9.25 | 45.88 | 124.46 | 1 |

*Table 1.* Relative Average querying time, assuming the query time of WAAL as the unit.

settings (dataset description, train/validation/test splitting, detailed implementations, hyper-parameter settings and choices) and additional experimental results are provided in the supplementary material.

**Baselines** We compare the proposed approach with the following baselines: 1) Random sampling; 2) Least confidence (Culotta and McCallum, 2005); 3) Smallest Margin (Scheffer and Wrobel, 2001); 4) Maximum-Entropy sampling (Settles, 2012); 5) $K$-Median approach (Sener and Savarese, 2018): choosing the points to be labelled as the cluster centers of $K$-Median algorithm ; 6) Core-set approach (Sener and Savarese, 2018); 7) Deep Bayesian AL (DBAL) (Gal et al., 2017); and 8) DeepFoolAL (Mayer and Timofte, 2018).

**Implementations** For the proposed approach, differing from baselines, we train the DNN from labeled and unlabeled data without data-augmentation. For the tasks on SVHN, CIFAR-10 we implement VGG16 (Simonyan and Zisserman, 2014) and for task on Fashion MNIST we implement LeNet5 (LeCun et al., 1998) as the feature extractor. On top of the feature extractor, we implement a two-layer multi-layer perceptron (MLP) as the classifier and critic function. For all tasks, at each interaction we set the maximum training epoch as 80. For each epoch, we feed the network with mini-batch of 64 samples and adopt SGD with momentum (Sutskever et al., 2013) to optimize the network. We tune the hyper-parameter through grid search. In addition, in order to avoid over-training, we also adopt early stopping (Caruana et al., 2001) techniques during training.

### 4.1. Results

We demonstrate the empirical results in Fig. 3. Exact numerical values and standard deviation are reported in the supplementary material. The proposed approach (WAAL) consistently outperforms all of the baselines during the interactions. We noticed that WAAL shows a large improvement ($> 5\%$) in the initial training procedure since it efficiently constructs a good representation through leveraging the unlabeled data information. For the relatively simple input task Fashion MNIST, the simplest uncertainty query (Smallest Margin/Least Confidence) finally achieved almost the same level performance with WAAL under 6K labeled samples. Moreover, we observed that for the small or middle sized queried batch (0.5K-2K) in the relatively complex dataset (SVHN, CIFAR-10), the baselines show similar results in deep AL, which is coherent with previous observations (Gissin and Shalev-Shwartz, 2019; Ash et al., 2019). On the contrary, our proposed approach still shows a good improved empirical result, emphasizing the benefits of properly designing loss for considering the unlabeled data in the context of deep AL.

We also report the average query time for the baselines and proposed approach on SVHN dataset in Tab. 1. The results indicates that WAAL holds the same querying time level with the standard uncertainty based strategies since they are all *end-to-end* strategies without knowing the internal information of the DNN. However some diversity based approaches such as Core-set and $K$-Median require the computation of the distance in the feature space and finally induce a much longer query time.

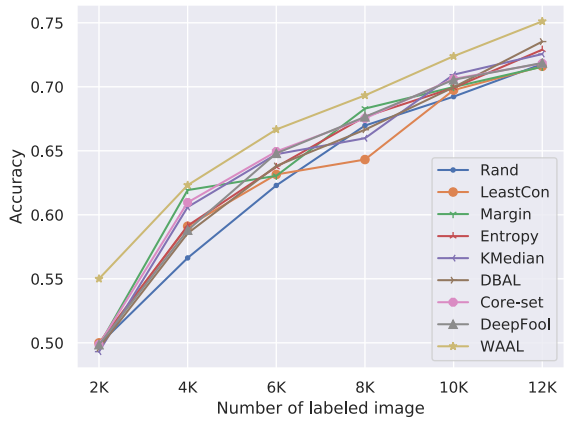



*Figure 4.* Ablation study in CIFAR-10: the baselines are all trained by leveraging the unlabeled information through $\mathcal{H}$-divergence.

### 4.2. Ablation Study: Advantage of Wasserstein Metric

In this part, we empirically show the advantage of considering the Wasserstein distance by the ablation study. Specifically, for the whole baselines we adopt $\mathcal{H}$-divergence based adversarial loss for training DNN. That is, we set a discriminator and we used the binary cross entropy (BCE) adversarial loss to discriminate the labeled and unlabeled data (Gissin and Shalev-Shwartz, 2019). Then in the query we still apply the different baselines strategies to obtain the labels. We tested in the CIFAR-10 dataset and report the performances in Fig. 4. Due to space limit, we present the brief introduction, exact numerical values, and more results in the supplementary material.

From the results, we observed that the gap between the initial training procedure has been reduced from about $8\%$ to $5\%$ because of introducing the adversarial based training. However, our proposed approach (WAAL) still consistently outperforms the baselines. The reason might be that the $\mathcal{H}$-divergence based adversarial loss is not a good metric for the Deep AL as we formally analyzed before. The results indicate the practical potential of adopting the Wasserstein distance for the Deep AL problem.

## 5. Conclusion

In this paper, we proposed a unified and principled method for both querying and training in the deep AL. We analyzed the theoretical insights from the intuition of modeling the interactive procedure in AL as distribution matching. Then we derived a new training loss for jointly learning hypothesis and query batch searching. We formulated the loss for DNN as a min-max optimization problem by leveraging the unlabeled data. As for the query for batch selection, it explicitly indicates the uncertainty-diversity trade-off. The results on different benchmarks showed a consistent better accuracy and faster efficient query strategy. The analytical and empirical results reaffirmed the benefits and potentials for reflecting on the unified principles for deep active learning. In the future, we want to 1) understand more general learning scenarios such as different distribution divergence metrics and its corresponding influences; 2) exploring other types of practical principles such as the auto-encoder based approach instead of adversarial training.

## References

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein gan. *arXiv preprint arXiv:1701.07875*.

Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. (2019). Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. (2010). A theory of learning from different domains. *Machine learning*, 79(1-2):151–175.

Caruana, R., Lawrence, S., and Giles, C. L. (2001). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. (2017). Joint distribution optimal transportation for domain adaptation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 3730–3739. Curran Associates, Inc.

Culotta, A. and McCallum, A. (2005). Reducing labeling effort for structured prediction tasks. In *AAAI conference on artificial intelligence*.

Dasgupta, S. (2011). Two faces of active learning. *Theoretical computer science*, 412(19):1767–1781.

Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1183–1192. JMLR. org.

Gissin, D. and Shalev-Shwartz, S. (2019). Discriminative active learning. *CoRR*, abs/1907.06347.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein

gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777.

Haussmann, M., Hamprecht, F., and Kandemir, M. (2019). Deep active learning with adaptive acquisition. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 2470–2476. International Joint Conferences on Artificial Intelligence Organization.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Krizhevsky, A. et al. (2009). Learning multiple layers of features from tiny images. Technical report, Citeseer.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Mayer, C. and Timofte, R. (2018). Adversarial sampling for active learning. *arXiv preprint arXiv:1808.06671*.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT Press.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

Pentina, A. and Ben-David, S. (2018). Multi-task kernel learning based on probabilistic lipschitzness. In *Algorithmic Learning Theory*, pages 682–701.

Scheffer, T. and Wrobel, S. (2001). Active learning of partially hidden markov models. In *In Proceedings of the ECML/PKDD Workshop on Instance Selection*. Citeseer.

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*.

Settles, B. (2012). Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sinha, S., Ebrahimi, S., and Darrell, T. (2019). Variational adversarial active learning. *arXiv preprint arXiv:1904.00370*.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147.

Urner, R. and Ben-David, S. (2013). Probabilistic lipschitzness a niceness assumption for deterministic labels. In *NIPS 2013*.

Urner, R., Wulff, S., and Ben-David, S. (2013). Plal: Cluster-based active learning. In *Conference on Learning Theory*, pages 376–397.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

Yin, C., Qian, B., Cao, S., Li, X., Wei, J., Zheng, Q., and Davidson, I. (2017). Deep similarity-based batch mode active learning with exploration-exploitation. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 575–584. IEEE.