Active Learning for Domain Adaptation: An Energy-based Approach

Binhui Xie[1] **Longhui Yuan**[1] **Shuang Li**[1,*] **Chi Harold Liu**[1] **Xinjing Cheng**[2,3] **Guoren Wang**[1]

[1]Beijing Institute of Technology [2]Tsinghua University [3]Inceptio Technology
{binhuixie, longhuiyuan, shuangli, chiliu, wanggr}@bit.edu.cn, cnorbot@gmail.com

Abstract

Unsupervised domain adaptation has recently emerged as an effective paradigm for generalizing deep neural networks to new target domains. However, there is still enormous potential to be tapped to reach the fully supervised performance. In this paper, we present a novel active learning strategy to assist knowledge transfer in the target domain, dubbed active domain adaptation. We start from an observation that energy-based models exhibit *free energy biases* when training (source) and test (target) data come from different distributions. Inspired by this inherent mechanism, we empirically reveal that a simple yet efficient energy-based sampling strategy sheds light on selecting the most valuable target samples than existing approaches requiring particular architectures or computation of the distances. Our algorithm, *Energy-based Active Domain Adaptation (EADA)*, queries groups of targe data that incorporate both domain characteristic and instance uncertainty into every selection round. Meanwhile, by aligning the free energy of target data compact around the source domain via a regularization term, domain gap can be implicitly diminished. Through extensive experiments, we show that EADA surpasses state-of-the-art methods on well-known challenging benchmarks with substantial improvements, making it a useful option in the open world. Code is available at https://github.com/BIT-DA/EADA.

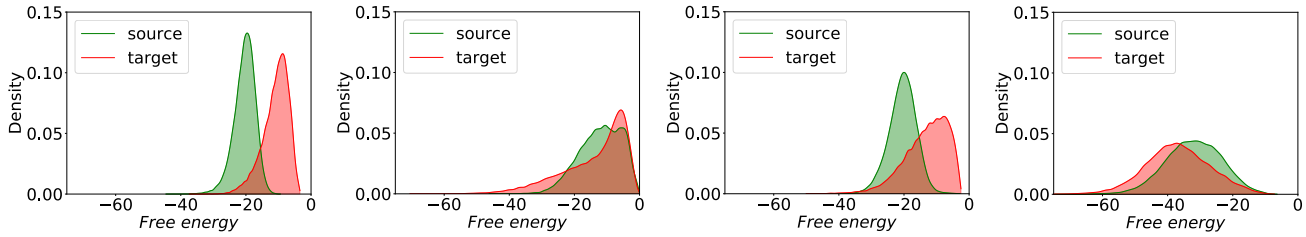
Introduction

In recent years, we have witnessed great strides in diverse machine learning problems with the success of deep neural networks (Krizhevsky, Sutskever, and Hinton 2012a). At the moment, however, these leaps in performance come only when massive labeled data are available. This limits their usage in many practical applications, such as autonomous driving with abundant unlabeled data (Yogamani et al. 2019) and medical diagnosis with high labeling cost (Ronneberger, Fischer, and Brox 2015). Moreover, even labeling all available data is not an excellent solution, as it's impossible to fully capture the way the world looks in a single dataset, let alone the fact that the test data rarely matches the data seen during training. Recognizing the challenges, extensive studies have been explored in domain adaptation (DA), which transfer the knowledge from a label-rich source domain to an unlabeled target domain (Pan and Yang 2010; Ganin and Lempitsky 2015; Tzeng et al. 2015, 2017; Long et al. 2019, 2018; Bousmalis et al. 2017; Saito et al. 2018; Li et al. 2021a,c). The performance of DA, in spite of great success, often falls far behind that of supervised learning. In practice, it may be feasible to obtain extra annotations for a small set of the target domain. But to be effective, it is critical to identify samples with high information only via active learning (Prince 2004; Hanneke 2014; Bickel, Brückner, and Scheffer 2009).

While previous active learning studies drastically lower human annotation costs, they are impractical when test data are collected from out-of-distribution. How can we design an efficient and practical sampling strategy for domain adaptation? For one thing, it is essential to determine which target samples will, once labeled, boost the accuracy and generalization considerably. For another, it remains the boundary to explore how to effectively utilize limited labeled data from the target domain to perform adaptation. Aware of this need, researchers have developed an array of active domain adaptation (Active DA) methods (Chattopadhyay et al. 2013; Rai et al. 2010; Su et al. 2020; Fu et al. 2021; Prabhu et al. 2021; Chan and Ng 2007). Prior works mainly focus on assessing how private each target data is according to the output of a domain discriminator or calculating its distance to the cluster centroids. However, these additional procedures either select target samples that are originally well aligned with the source domain or increase the computational overhead, which limits their capability. Therefore, a simple yet efficient solution is urgently desired.

In this paper, we advocate the use of energy-based models (EBMs) (LeCun et al. 2006) to help realize the potential of active learning under domain shift. For any given x (e.g., an image), an EBM approach gives the lowest energy to the correct answer y (e.g., a label). Grathwohl et al. (2020) and Liu et al. (2020) have demonstrated that energy-based training improves calibration and better distinguishes in- and out-of-distribution samples than the standard discriminative classifier. At this point, we begin with investigating the distributions of free energy on source and target domains using diverse methods and make several observa-

*Corresponding author.



(a) Source Only (b) Fully Supervised (c) Random (d) EADA (Ours)

Figure 1: (a & b) Free energy distribution biases between source and target domains on VisDA-2017 from "Source Only". We then contrast the distributions from (c) a native baseline of random selection and (d) our energy-based selection strategy. EADA exhibits better-aligned distribution than "Random" and is similar to "Full Supervised" i.e., all source and target data are labeled.

tions from Fig. 1. First, a model trained only on labeled source data will cause the free energy distribution of the supervised source data to be lower than that of the unlabeled target data, that is, *free energy biases* between the two domains (Fig. 1(a)). Then, an interesting finding is that these two distributions tend to be consistent using "Full Supervised" (Fig. 1(b)). Next, the biases eliminate slightly when a few unlabeled target data are randomly annotated in the training process (Fig. 1(c)). Lastly, using our algorithm to identify limited target instances for labeling, surprisingly, it well matches two distributions as same as the situation of "Full Supervised" (Fig. 1(d)). We conjecture that there exist redundant or trivial data in the target domain itself that do little to help the learning objective.

Intuitively, we provide both mathematical insights and empirical evidence that an energy-based active learning scheme is desirable for active domain adaptation. A central theme of this work is that we design an approach, Energy-based Active Domain Adaptation (EADA), which adequately ensures samples that are representative of the entire target domain to be selected by considering both domain characteristic and instance uncertainty. More precisely, as mentioned above, the free energies of most labeled source data are lower than that of unlabeled target data. Thus, we can treat the intrinsic free energy of an unlabeled target sample as a surrogate metric to reflect the domain characteristic. Naturally, the target samples with higher free energy are more dissimilar to source data, and thus be typical for target distribution. In addition, we assess the value of minimum energy versus second-minimum energy (MvSM) for each unlabeled target data to quantify its uncertainty under the current model. To this end, given the labeling budget in each round, we first maintain a candidate set from unlabeled target data with higher free energies and then select samples with significant MvSM values from candidates. Furthermore, free energy can also serve as a regularization signal in the form of an alignment loss to implicitly diminish domain shift, which is complementary to our active strategy.

In summary, our work makes the following contributions:

- We provide a new perspective to select a highly informative subset of unlabeled target data under domain shift via exploiting *free energy biases* between the two domains.
- We complement empirical results with theoretical investigations in the method section and establish an intuitive sufficient condition when it would help.
- Though simple, EADA attains excellent results with quite limited labeling expenses. Extensive experiments and in-depth analysis demonstrate its effectiveness.

Related Work

Active learning (AL) has been studied for decades in both theory and practice (Settles 2009; Dasgupta 2011; Bachman, Sordoni, and Trischler 2017; Gal, Islam, and Ghahramani 2017). A case in point is to search informative data for labeling in order to learn a satisfactory model at a low annotation cost. Most popular algorithms formulate and solve it by uncertainty sampling. They select samples about which the current model is uncertain (Schohn and Cohn 2000; Joshi, Porikli, and Papanikolopoulos 2009; Wang and Shang 2014). Another line of work turns to representative sampling (Sener and Savarese 2018; Sinha, Ebrahimi, and Darrell 2019; Gissin and Shalev-Shwartz 2019), which picks a set of typical samples via clustering or core-set selection.

Recently, several studies have leveraged a hybrid of the above active sampling objectives to achieve promising results, such as Ash et al. (2020). However, these conventional AL methods cannot deal with the domain shift issues for domain adaptation, whereas our method aims to overcome this challenge by leveraging a simple energy-based strategy.

Unsupervised domain adaptation (UDA) studies the problem of transferring knowledge gained from an abundant labeled source domain to a target domain where labeled data are scarce (Ganin and Lempitsky 2015; Long et al. 2019, 2017; Xu et al. 2019; Li et al. 2018, 2020, 2021b; Hoffman et al. 2018; Zou, Yang, and Wu 2021; Gong et al. 2012). A series of works minimizes the domain discrepancy at the uppermost layer of deep neural networks using maximum mean discrepancy (Gretton et al. 2007) or adversarial training (Goodfellow et al. 2014). Recently, some methods allow a few target data labeled, e.g., semi-supervised DA (Saito et al. 2019) and few-shot DA (Teshima, Sato, and Sugiyama 2020). Though impressive, they randomly select a few data to annotate, neglecting which target samples should be labeled given a fixed labeling budget. Consequently, some selected samples are originally well predicted by the current

model. In contrast, our work differentiates itself by allowing the model to acquire labels for valuable target samples via an oracle. As such, it would have the best potential performance gain compared with randomly picking labels.

Active domain adaptation (Active DA). The seminal work (Rai et al. 2010) has demonstrated the synergy between active learning and domain adaptation, which facilitates AL in a domain of interest with the aid of the knowledge from a related domain. Recently, Su et al. (2020) and Fu et al. (2021) incorporate Active DA with advanced tools, such as adversarial training, both of which identify domainness via a learned domain discriminator. However, it may give identically high scores to most target data, thus not adequately ensuring that selected samples are representative of the entire target distribution. A parallel line of work instead proposes to select active samples via clustering. For example, Prabhu et al. (2021) cluster deep embeddings of target data weighted by the uncertainty and select nearest neighbors to the inferred cluster centroids for labeling. However, clustering-based strategies have some drawbacks in nature. First, they encounter a computational burden and could hardly be applied on large data sets. Second, the clustering is sensitive to noise and easy to collapse.

Originating from energy-based models, our method adapts the concept of energy to identify limited target samples that are most unique to the target distribution and meanwhile complementary to labeled source data. It yields a new sampling protocol that accounts for domain characteristic and instance uncertainty together. Also, it has no extra parameters that need to be optimized and learning is efficient.

Method

In active domain adaptation (Active DA), we have access to a labeled source domain $\mathcal{S} = \{(x_s, y_s)\}$ and an unlabeled target domain $\mathcal{T} = \{x_t\}$ from different distributions. Following the standard Active DA setting (Fu et al. 2021; Prabhu et al. 2021), B active samples are selected in the target domain for annotation, which are much smaller than the amount of $\mathcal{T}$. Therefore, the entire target domain consists of a labeled pool $\mathcal{T}_l$ and an unlabeled pool $\mathcal{T}_u$, i.e., $\mathcal{T} = \mathcal{T}_l \cup \mathcal{T}_u$. The goal is to learn a neural network with parameter θ that brings good generalization on the target. In this work, we introduce an energy-based strategy to select the most valuable target samples to assist the knowledge transfer.

Energy-based Models Revisit

The essence of machine learning is to encode dependencies between variables. Let us consider an energy-based model (EBM) with two sets of variables x (a high-dimensional variable) and y (a discrete variable). Training this model consists in finding an energy function i.e., $E(x, y)$ that gives the lowest energy to correct answer and higher energy to all other (incorrect) answers.[1] Precisely, the model must produce the value y^* for which $E(x, y)$ is the smallest:

$$y^* = \arg\min_{y \in \mathcal{Y}} E(x, y) \,. \tag{1}$$

[1] See (LeCun et al. 2006) for a comprehensive tutorial.

Generally, the size of set $\mathcal{Y}$ is small for classification, hence the inference procedure can simply compute $E(x, y)$ for all possible values of $y \in \mathcal{Y}$ and pick the smallest.

With the energy function, the joint probability of input x and label y can be estimated through the Gibbs distribution:

$$p(x, y) = \exp\left(-E(x, y)\right)/Z \ , \tag{2}$$

where $Z = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \exp\left(-E(x, y)\right)$ is called the partition function that marginalizes over x and y. It should be noted that the above transformation of energy into probability is only possible if Z converges. By marginalizing out y, we obtain the probability density for x as well,

$$p(x) = \sum\nolimits_{y \in \mathcal{Y}} p(x, y) = \sum\nolimits_{y \in \mathcal{Y}} \exp\left(-E(x, y)\right) \Big/ Z \,. \tag{3}$$

Intuitively, in Active DA, to select the most representative target samples, one can directly estimate the probability of occurrence for each target sample from Eq. (3) and then those samples with lower probabilities should be selected.

Unfortunately, one cannot compute or even reliably estimate Z. Therefore, we turns to *free energy* i.e., $\mathcal{F}(x)$, a function hidden in EBMs that serves as the "rationality" of the occurrence of the variable x. Mathematically, the probability density for x can also be expressed as

$$p(x) = \frac{\exp\left(-\mathcal{F}(x)\right)}{\sum_{x \in \mathcal{X}} \exp(-\mathcal{F}(x))} \,. \tag{4}$$

This formulation indicates that $\mathcal{F}(x)$ could be substituted for $p(x)$ to select the target samples that have lower probabilities. By connecting Eq. (3) and Eq. (4), we have

$$\mathcal{F}(x) = -\log \sum_{y \in \mathcal{Y}} \exp\left(-E(x, y)\right) \,. \tag{5}$$

Energy-based Active Domain Adaptation

We now wish to take advantage of a new perspective of the energy-based model (EBM) to gain the benefits of active domain adaptation, where the biases of free energy between source- and target-domain data allow effective selection and adaptation. In the following, we first describe how to train an EBM with several loss functions. We then describe using an energy-based sampling strategy to identify the most informative unlabeled target data to annotate. At last, we provide an intuitive sufficient condition when it helps.

Training process Given a set of labeled source samples $\mathcal{S} = \{(x_s, y_s)\}$, we want to train a well-behaved EBM that gives the lowest energy to the correct answer and higher energy to all other (incorrect) answers. To this end, we utilize a commonly used loss in EBMs, i.e., the negative log-likelihood loss that comes from probabilistic modeling to train a model for classification, and it can be formulated as

$$\mathcal{L}_{nll}(x, y; \theta) = E(x, y; \theta) + \frac{1}{\tau} \log \sum_{c \in \mathcal{Y}} \exp\left(-\tau E(x, c; \theta)\right) , \tag{6}$$

where τ ($\tau > 0$) is the reverse temperature and a low value corresponds to smooth partition of energy over the space $\mathcal{Y}$. For simplicity, we fix τ=1, and then we have

$$\mathcal{L}_{nll}(x, y; \theta) = E(x, y; \theta) - \mathcal{F}(x; \theta) \,. \tag{7}$$

The second term in Eq. (7) will cause the energies of all answers to be pulled up. The energy of the correct answer is

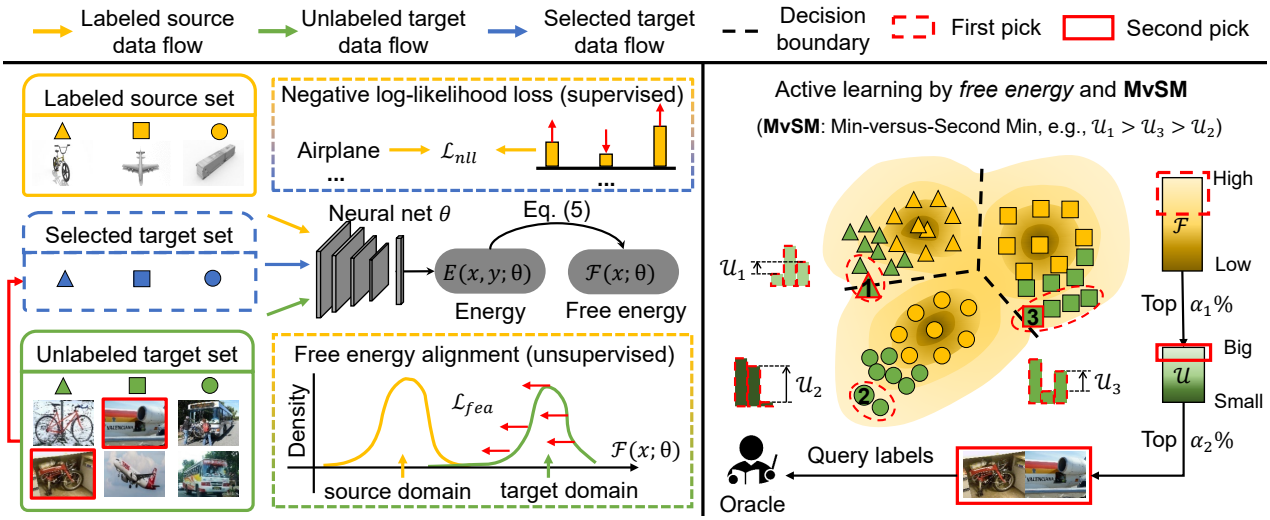


Figure 2: Overview of the EADA. **Left** (training process): we utilize the standard negative log-likelihood loss in conjunction with the proposed free energy alignment loss to train the network; **Right** (selection process): to iteratively build a labeled target set, 1% of target samples are quired to annotate in each selection round. We first select a set of $\alpha_1\%$ candidates with the highest free energy (domain characteristic). Then we sample $\alpha_2\%$ points from candidates with biggest MvSM (instance uncertainty).

also pulled up, but not as hard as it is pushed down by the first term. An analysis of gradient is presented in Appendix.

However, we observe that the values of free energy on target samples are considerably higher than those on source ones, called *free energy biases*. Naturally, one can treat it as a surrogate to reflect the domain divergence. By designing a simple regularization term, these biases can be reduced, which to some extent aligns the distribution across domains. And the free energy alignment loss $\mathcal{L}_{fea}$ is defined as:

$$\mathcal{L}_{fea}(x;\theta) = \max\left(0, \mathcal{F}(x;\theta) - \Delta\right), \tag{8}$$

where $\Delta = \mathbb{E}_{x\sim\mathcal{S}}\mathcal{F}(x;\theta)$ is the average value of the free energy over source data. During training, Δ is estimated via exponential moving average: $\Delta_t = \lambda\Delta_{t-1} + (1-\lambda)\Delta'_t$, where Δ_t is the estimation of average value in all t mini-batches and Δ'_t is the average value in t^{th} mini-batch and λ is a weight sampled from the uniform distribution, $\lambda \sim U(0,1)$. Additionally, we experimentally found that such way is comparable with calculating average value over the whole source domain data while improving the efficiency.

Overall, the full learning objective is given by:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{S}\cup\mathcal{T}_l}\mathcal{L}_{nll}(x,y;\theta) + \gamma\mathbb{E}_{x\sim\mathcal{T}_u}\mathcal{L}_{fea}(x;\theta), \tag{9}$$

where γ is a loss weight hyperparameter.

Selection process The goal in Active DA is to identify more valuable target samples that, once labeled and used for training, improve the model's accuracy and generalization performance significantly. In practice, we suggest a two-step sampling strategy to adequately ensure such samples by incorporating domain characteristic and instance uncertainty. To be clear, we summarize the training and selection processes based on the above discussion as Algorithm 1.

Step one: we observe that biases of free energy distribution between source and target domains exhibit. Thus, we can utilize this intrinsic free energy of an unlabeled target sample as a surrogate metric to reflect the domain characteristic. Certainly, the target samples with higher free energy are unique to the target distribution and meanwhile complementary to the labeled source data.

Step two: to measure instance uncertainty, existing methods rely primarily on the entropy score (Su et al. 2020; Prabhu et al. 2021). In contrast, we consider the difference between the energy values of the two answers with the lowest estimated energy value as a measure of uncertainty. Since it is a comparison of the minimum answer and the second minimum answer, we refer to it as the Min-versus-Second-Min (MvSM) strategy and it can be formulated as

$$\mathcal{U}(x) = E(x,y^*;\theta) - E(x,y';\theta), \tag{10}$$

where $y^* = \arg\min_{y\in\mathcal{Y}} E(x,y;\theta)$ is the lowest energy output and $y' = \arg\min_{y\in\mathcal{Y}\setminus\{y^*\}} E(x,y;\theta)$ is the second-

Algorithm 1: **EADA algorithm**

Input: Labeled source data $\mathcal{S}$, unlabeled target data $\mathcal{T}_u$ and labeled target set $\mathcal{T}_l = \emptyset$, maximum epoch M, selection rounds R, selection ratios α_1 , α_2
Output: Final model parameters θ_M
for $m = 1$ to M **do**
  Update model θ_m via Eq. (9)
  if m in R **then**
    $\forall x \in \mathcal{T}_u$, compute free energy $\mathcal{F}(x)$ (Eq. (5)) to serve as measure of domain characteristic
    $\mathcal{T}_l^r \leftarrow$ select $\alpha_1\%$ of $\mathcal{F}$ with the highest values
    $\forall x \in \mathcal{T}_l^r$, compute MvSM $\mathcal{U}(x)$ (Eq. (10)) to serve as measure of instance uncertainty
    $\mathcal{T}_l^r \leftarrow$ select $\alpha_2\%$ of $\mathcal{U}$ with the highest values as active samples for annotating, getting $\mathcal{T}_l = \mathcal{T}_l \cup \mathcal{T}_l^r$
  end if
end for

lowest energy output. Such a measure is a more direct way of estimating confusion about class membership from a classification standpoint. Using the MvSM measure, the instances around the decision boundaries in Fig. 2 during the selection procedure will be selected to query an oracle.

Theoretical Analysis

This section contains our preliminary study of why *free energy biases* exhibit between two different domains. For an energy-based model, we prove that positive gradient inner product between the negative log-likelihood loss function and *free energy* leads to a lower value of *free energy* on labeled source samples during the training process. Limited by space, all the proofs are left for the Appendix.

Before stating our main theoretical result, we first illustrate the general intuition with a toy problem. Considering a simple energy-based model on the classification task, where the network is a one layer linear network parameterized by $\mathbf{W} = (\omega_1 \quad \cdots \quad \omega_C)^\top \in \mathbb{R}^{C\times N}$, $x \in \mathbb{R}^N$ denotes a source sample, $y \in \{1, ..., C\}$ denotes the label, we have

$$\begin{aligned} E(x, j; \mathbf{W}) &= \omega_j^\top x, \ j = 1, ..., C, \\ \mathcal{F}(x; \mathbf{W}) &= -\log \sum_{c=1}^{C} \exp(-\omega_c^\top x), \\ \mathcal{L}_{nll}(x, y; \mathbf{W}) &= E(x, y; \mathbf{W}) - \mathcal{F}(x; \mathbf{W}). \end{aligned} \tag{11}$$

Now we update the the weight matrix $\mathbf{W}$ by one step of gradient descent on $\mathcal{L}_{nll}$ as follows:

$$\mathbf{W}' = \mathbf{W} - \eta \nabla \mathcal{L}_{nll}(x, y; \mathbf{W}), \tag{12}$$

where η is the learning rate and $\mathbf{W}'$ is the updated matrix.

Then we have two lemmas to show that the inner product between the gradients of negative log-likelihood loss function and free energy is positive, and the value of the free energy of a labeled source sample is descending with a step of gradient descent on negative log-likelihood loss function.

Lemma 1. *Assume that a toy model correctly predict a labeled source sample (x, y), we have*

$$\langle \nabla \mathcal{L}_{nll}(x, y; \mathbf{W}), \nabla \mathcal{F}(x; \mathbf{W}) \rangle > 0, \tag{13}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of gradients.

Lemma 2. *Assume that a toy model correctly predict a labeled source sample (x, y) with learning rate $\eta > 0$ we have*

$$\mathcal{F}(x; \mathbf{W}) > \mathcal{F}(x; \mathbf{W}'), \tag{14}$$

To summarize, if the positive gradient inner product between the negative log-likelihood loss function and *free energy*, free energy biases are exhibited. Our main theoretical results extend this to general deep neural networks.

Theorem 1. *Let $\mathcal{L}_{nll}(x, y; \theta)$ denote the negative log-likelihood loss on source domain (x, y) with parameters of deep network θ and $\mathcal{F}(x; \theta)$ denote the free energy of x. Assume that $\forall (x, y)$, $\mathcal{L}_{nll}(x, y; \theta)$ is differentiable, β-smooth in θ and $\forall \theta$, $\|\nabla \mathcal{L}_{nll}(x, y, \theta)\| < G$, $\|\nabla \mathcal{F}(x; \theta)\| < G$. With learning rate $\eta \in (0, \frac{2\varepsilon}{\beta G^2})$, and for every (x, y) such that*

$$\langle \nabla \mathcal{L}_{nll}(x, y; \theta), \nabla \mathcal{F}(x; \theta) \rangle > \varepsilon, \tag{15}$$

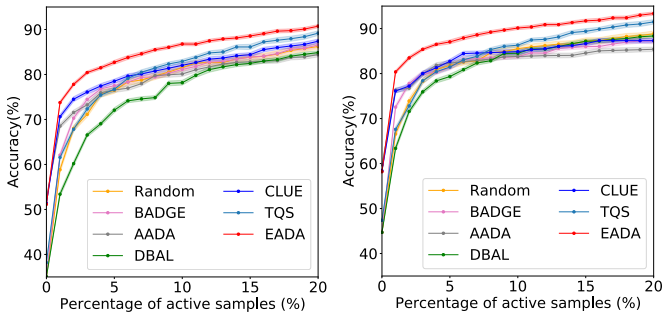
where $\varepsilon > 0$, we have

$$\mathcal{F}(x; \theta) > \mathcal{F}(x; \theta'), \tag{16}$$

where $\theta' = \theta - \eta \nabla \mathcal{L}_{nll}(x, y; \theta)$ i.e., supervised training with one step of gradient descent, and $\langle \cdot, \cdot \rangle$ denotes the inner product of gradients.

Experiments

We evaluate EADA against state-of-the-art approaches on various scenarios including a toy problem, three popular image classification datasets: **VisDA-2017** (Peng et al. 2017), **Office-Home** (Venkateswara et al. 2017) and **Office-31** (Saenko et al. 2010), as well as a challenging semantic segmentation task, i.e., **GTAV** (Richter et al. 2016) to **Cityscapes** (Cordts et al. 2016). All methods are implemented based on PyTorch, employing ResNet (He et al. 2016) models pre-trained on ImageNet (Krizhevsky, Sutskever, and Hinton 2012b). We follow the standard protocols for Active DA as (Su et al. 2020; Fu et al. 2021). Meanwhile, the various compared active learning, active domain adaptation and domain adaptation algorithms are **Source Only** (ResNet), **Random** (randomly select target samples to label), **BvSB** (Joshi, Porikli, and Papanikolopoulos 2009), **Entropy** (Wang and Shang 2014), **CoreSet** (Sener and Savarese 2018), **WAAL** (Shui et al. 2020), **BADGE** (Ash et al. 2020), **AADA** (Su et al. 2020), **DBAL** (de Mathelin, Mougeot, and Vayatis 2021), **TQS** (Fu et al. 2021), **CLUE** (Prabhu et al. 2021), **AdaptSegNet** (Tsai et al. 2018), and **PLCA** (Kang et al. 2020). Notably, we carry out experiments with five different random seeds and report the average accuracy. More details are presented in Appendix.



(a) ResNet-18 (b) ResNet-50

Figure 3: Comparison results of varying the percentage of labeled target samples on **VisDA-2017** with ResNet-18/50.

Main Results

VisDA-2017. The experimental results of different methods with 5% labeling budget on VisDA-2017 are shown in the first column in Table 1, proving that EADA is superior to all the baselines. Randomly selecting samples achieves better performance than ResNet, which implies that active learning is a promising and complementary solution for DA.

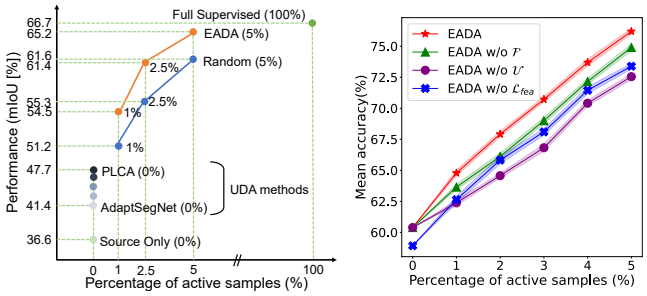
In addition, to further validate the effectiveness of EADA, we vary the target labeling budget from 0% to 20% with different backbones ResNet-18/50 and report the performance

Method	VisDA-2017	Office-Home												
		Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Mean
Source Only	44.7 ± 0.1	42.1	66.3	73.3	50.7	59.0	62.6	51.9	37.9	71.2	65.2	42.6	76.6	58.3
Random	78.1 ± 0.6	52.5	74.3	77.4	56.3	69.7	68.9	57.7	50.9	75.8	70.0	54.6	81.3	65.8
BvSB	81.3 ± 0.4	56.3	78.6	79.3	58.1	74.0	70.9	59.5	52.6	77.2	71.2	56.4	84.5	68.2
Entropy	82.7 ± 0.3	58.0	78.4	79.1	60.5	73.0	72.6	60.4	54.2	77.9	71.3	58.0	83.6	68.9
CoreSet	81.9 ± 0.3	51.8	72.6	75.9	58.3	68.5	70.1	58.8	48.8	75.2	69.0	52.7	80.0	65.1
WAAL	83.9 ± 0.4	55.7	77.1	79.3	61.1	74.7	72.6	60.1	52.1	78.1	70.1	56.6	82.5	68.3
BADGE	84.3 ± 0.3	58.2	79.7	79.9	61.5	74.6	72.9	61.5	56.0	78.3	71.4	60.9	84.2	69.9
AADA	80.8 ± 0.4	56.6	78.1	79.0	58.5	73.7	71.0	60.1	53.1	77.0	70.6	57.0	84.5	68.3
DBAL	82.6 ± 0.3	58.7	77.3	79.2	61.7	73.8	73.3	62.6	54.5	78.1	72.4	59.9	84.3	69.6
TQS	83.1 ± 0.4	58.6	81.1	81.5	61.1	76.1	73.3	61.2	54.7	79.7	73.4	58.9	86.1	70.5
CLUE	85.2 ± 0.4	58.0	79.3	80.9	68.8	77.5	76.7	66.3	57.9	81.4	75.6	60.8	86.3	72.5
EADA	**88.3 ± 0.1**	**63.6**	**84.4**	**83.5**	**70.7**	**83.7**	**80.5**	**73.0**	**63.5**	**85.2**	**78.4**	**65.4**	**88.6**	**76.7**

Table 1: Comparison results on **VisDA-2017** and **Office-Home** with **5%** target samples as the labeling budget.

Method	A→D	A→W	D→A	D→W	W→A	W→D	Mean
Source Only	81.5	75.0	63.1	95.2	65.7	99.4	80.0
Random	87.1	84.1	75.5	98.1	75.8	99.6	86.7
BvSB	89.8	87.9	78.2	99.0	78.6	**100.0**	88.9
Entropy	91.0	89.2	76.1	99.7	77.7	**100.0**	88.9
CoreSet	82.5	81.1	70.3	96.5	72.4	99.6	83.7
WAAL	88.4	89.6	76.4	**100.0**	76.0	**100.0**	88.4
BADGE	90.8	89.1	79.8	99.6	79.6	**100.0**	89.8
AADA	89.2	87.3	78.2	99.5	78.7	**100.0**	88.8
DBAL	88.2	88.9	75.2	99.4	77.0	**100.0**	88.1
TQS	92.8	92.2	80.6	**100.0**	80.4	**100.0**	91.1
CLUE	92.0	87.3	79.0	99.2	79.6	99.8	89.5
EADA	**97.7**	**96.6**	**82.1**	**100.0**	**82.8**	**100.0**	**93.2**

Table 2: Comparison results on **Office-31** with **5%** target samples as the labeling budget.



(a) GTAV → Cityscapes (b) Ablation study

Figure 4: (a) Experimental results on GTAV→Cityscapes.(b) Mean accuracy of EADA and its variants on Office-Home.

after each round in Fig. 3. We can observe that EADA consistently outperforms alternative methods across rounds. For instance, with shallower ResNet-18, we improve upon the state-of-the-art method, i.e., TQS by 2-6% over rounds, and obtain comparable results against other methods using deeper ResNet-50 at some rounds. This demonstrates that EADA can indeed select more representative and informative target data using our novel energy-based criterion. Additional comparison results with standard active learning methods are shown in Appendix.

Office-Home & Office-31. The results on Office-Home and Office-31 are reported in Table 1 & 2, respectively, which show the best performance across tasks. It can be observed that most Active DA methods outperform the traditional AL methods since the latter does not take the domain shift into account. EADA performs much better than all the baselines with a large margin, especially for very hard shifts e.g., Ar→Cl, Pr→Cl, D→A and W→A, which emphasizes the benefit of jointly capturing domain characteristic and instance uncertainty for active sampling in combination with free energy alignment.

GTAV → Cityscapes. While prior works restrict their task to image classification, it is important to also study Active DA in the context of related tasks. Now we focus on task of semantic segmentation adapting from GTAV to Cityscapes and we use the same setting as (Tsai et al. 2018; Kang et al. 2020), which adopts DeepLab-v2 (Chen et al. 2018) with ResNet-101 as backbone network. We select 5% target images to query for pixel-level labels of the whole image. The experimental results are shown in Fig. 4(a). There is a large performance gap between the UDA methods and the full supervised version, such as a popular adversarial approach, AdaptSegNet, lags behind 25.2% mIoU. Surprisingly, our EADA brings a significant boost and shows performance comparable to that of fully supervised at the final round.

Insight Analysis

Ablation study. To investigate the efficacy of key components of the proposed EADA, we conduct a thorough ablation study with the following variants on all 12 tasks of Office-Home: (i) EADA w/o $\mathcal{F}$: removing the free energy sampling from selection process; (ii) EADA w/o $\mathcal{U}$: removing the instance uncertainty sampling from the selection process; (iii) EADA w/o $\mathcal{L}_{fea}$: removing $\mathcal{L}_{fea}$ from Eq. (9).

The results are shown in Fig. 4(b), it is clear that the full method outperforms other variants and achieves large improvements. We also observe that EADA surpasses EADA (w/o $\mathcal{F}$ and w/o $\mathcal{U}$), manifesting that domain characteristic sampling and instance uncertain sampling are both necessary to select representative and informative data. Further, the consistent and notable increases from EADA w/o $\mathcal{L}_{fea}$ to EADA justify our decision to use a regularization term to align free energy distributions between both domains, which is beneficial to reducing the domain shift implicitly.

Toy example. To better explain why the energy-based label acquisition strategy works well and what kind of sample is more representative and informative, we perform a toy

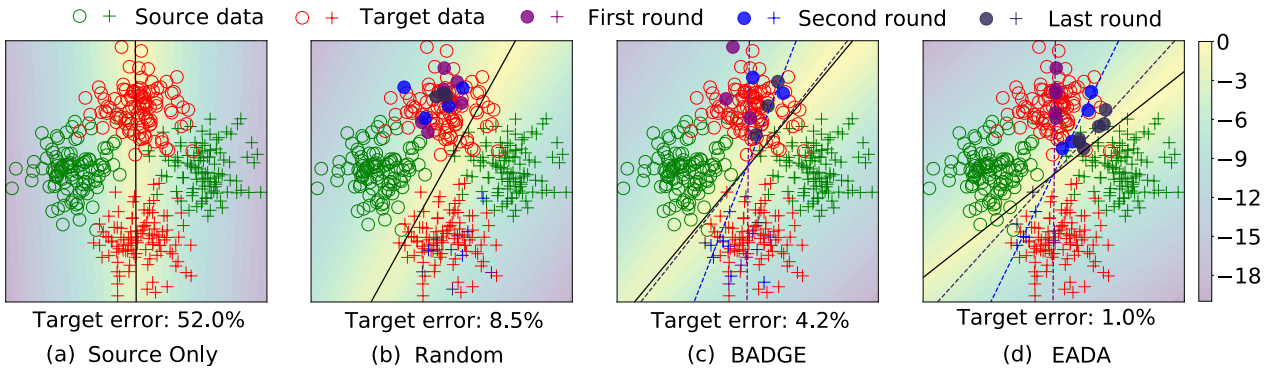


Figure 5: (Best viewed in color.) Illustrative comparison of sampling strategies on a toy example. Red points and green points denote unlabeled target data and labeled source data, respectively. Source data are drawn from two different Gaussian distributions denoted as circle class and plus class and target data are generated by rotating source data directly. We train a single layer fully-connected network and perform 3 rounds of active learning with a per-round budget is 2% of target samples. In (c) & (d), we draw the decision boundary before each selection round with a dash line, and the final decision boundary with a solid line.

α_1 / α_2 (%)	10 / 10	25 / 4	50 / 2	75 / 1.3	100 / 1	1 / 100
Office-31	91.9	92.6	**93.2**	91.5	90.8	90.4
Office-Home	76.2	76.3	**76.7**	76.0	74.7	72.6
VisDA-2017	87.2	87.6	**88.3**	87.1	86.6	85.7

Table 3: Effect of selection ratios.

example, a binary classification task with domain shift. As shown in Fig. 5, from the left to the right: Source Only, Random, BADGE, and our EADA are shown one by one and the target errors are 52.0%, 8.5%, 4.2%, 1.0%, respectively.

From the experimental results, we can make several insightful findings. (i) *Free energy biases*: the values of free energy on target samples are considerably higher than those on source samples in Fig. 5(a). Motivated by this, we design a free energy sampling as a surrogate measure to describe domain characteristic. (ii) *Redundant/Trivial selection*: in Fig. 5(b), we can observe that a large portion of samples selected by "Source Only" resides in an area where the target data density is high, leading to many redundant instances. BADGE (a state-of-the-art active learning method) runs a clustering scheme on "gradient embedding" to incorporate both uncertainty and diversity, which slightly mitigates the dilemma of redundancy. However, when we deeply study the relationship between decision boundary and the selected samples in each round, we find that BADGE still selects a few well-aligned samples and the selected samples are not the most uncertain samples of the current classifier. (iii) *Free energy versus decision boundary*: the final decision boundary is the area with the highest free energy. Accordingly, we explore a MvSM metric to precisely quantify the uncertainty of a target sample under the current model. The results in Fig. 5(d) validate the effectiveness of our method. In short, we define that a target sample with the highest free energy and located around the decision boundary serves as the most valuable, both representative and informative, sample.

Effect of selection ratios. In Table 3, we show the accuracy on three image classification benchmarks with varying α_1 (α_2). Our EADA can achieve consistent performance within a wide range. It is worth noting that excluding any step (α_1 or $\alpha_2 = 100$) will lead to a performance drop. We leave it as future work to explore other more complex combinations like self-adaptive α_1 and weighted calculation.

	AL Strategy	Query Complexity	Query Time (Ar→Cl, VisDA-2017)
cluster	CoreSet	$\mathcal{O}(DN^2)$	(0.1s, 1.3m)
	BADGE	$\mathcal{O}(CDN^2)$	(4.7s, 3.5m)
	DBAL	$\mathcal{O}(DN(M+N))$	(0.4s, 5.3m)
	CLUE	$\mathcal{O}(DN(N+TB))$	(0.5s, 2.9m)
rank	Entropy	$\mathcal{O}(N \log N)$	(0.04s, 2.1s)
	AADA	$\mathcal{O}(N \log N)$	(0.03s, 2.2s)
	TQS	$\mathcal{O}(N \log N)$	(0.04s, 1.7s)
	EADA	$\mathcal{O}(N \log N)$	**(0.02s, 0.9s)**

Table 4: Comparison results on query complexity and query time. C, M, N denote number of classes, source instances and target instances respectively. D denotes feature dimension, B is labeling budget, T denotes clustering rounds.

Time complexity. Table 4 lists the query complexity and query time for EADA and comparable baseline methods. BADGE and CLUE achieve better mean accuracy (see Table 1 and Table 2) but are slower due to a clustering step. Our EADA obtains the best accuracy and is significantly more efficient than the competitive baselines as well.

Conclusion

In this paper, we present Energy-based Active Domain Adaptation (EADA), an algorithm to tackle performance limitations of domain adaptation at minimal label cost. We propose a novel energy-based sampling strategy into domain adaptation, for the selection of limited target samples that are representative and informative. On top of that, we further explore a regularization term to implicitly diminish the domain gap. In addition, theoretical results about when and why EADA is expected to work are elaborated. Through our experiments, we demonstrate its effectiveness in various

transfer scenarios. More generally, our work is but a small step toward alleviating the intensive workload of annotation. This offers encouraging evidence that there remains value to be explored to go beyond the fully supervised method.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61902028).

References

Arthur, D.; and Vassilvitskii, S. 2007. k-means++: the advantages of careful seeding. In *SODA*, 1027–1035.

Ash, J. T.; Zhang, C.; Krishnamurthy, A.; Langford, J.; and Agarwal, A. 2020. Deep Batch Active Learning by Diverse, Uncertain Gradient Lower Bounds. In *ICLR*.

Bachman, P.; Sordoni, A.; and Trischler, A. 2017. Learning Algorithms for Active Learning. In Precup, D.; and Teh, Y. W., eds., *ICML*, 301–310.

Bickel, S.; Brückner, M.; and Scheffer, T. 2009. Discriminative Learning Under Covariate Shift. *J. Mach. Learn. Res.*, 10: 2137–2155.

Bousmalis, K.; Silberman, N.; Dohan, D.; Erhan, D.; and Krishnan, D. 2017. Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks. In *CVPR*, 95–104.

Bubeck, S. 2015. Convex Optimization: Algorithms and Complexity. *Found. Trends Mach. Learn.*, 8(3-4): 231–357.

Chan, Y. S.; and Ng, H. T. 2007. Domain Adaptation with Active Learning for Word Sense Disambiguation. In *ACL*.

Chattopadhyay, R.; Fan, W.; Davidson, I.; Panchanathan, S.; and Ye, J. 2013. Joint Transfer and Batch-mode Active Learning. In *ICML*, 253–261.

Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4): 834–848.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 3213–3223.

Dasgupta, S. 2011. Two faces of active learning. *Theor. Comput. Sci.*, 412(19): 1767–1781.

de Mathelin, A.; Mougeot, M.; and Vayatis, N. 2021. Discrepancy-Based Active Learning for Domain Adaptation. *CoRR*, abs/2103.03757.

Fu, B.; Cao, Z.; Wang, J.; and Long, M. 2021. Transferable Query Selection for Active Domain Adaptation. In *CVPR*, 7272–7281.

Gal, Y.; Islam, R.; and Ghahramani, Z. 2017. Deep Bayesian Active Learning with Image Data. In *ICML*, 1183–1192.

Ganin, Y.; and Lempitsky, V. 2015. Unsupervised Domain Adaptation by Backpropagation. In *ICML*, 1180–1189.

Gissin, D.; and Shalev-Shwartz, S. 2019. Discriminative Active Learning. *CoRR*, abs/1907.06347.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2066–2073.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NeurIPS*, 2672–2680.

Grathwohl, W.; Wang, K.; Jacobsen, J.; Duvenaud, D.; Norouzi, M.; and Swersky, K. 2020. Your classifier is secretly an energy based model and you should treat it like one. In *ICLR*.

Gretton, A.; Borgwardt, K. M.; Rasch, M.; Schölkopf, B.; and Smola, A. J. 2007. A kernel method for the two-sample-problem. In *NeurIPS*, 513–520.

Hanneke, S. 2014. Theory of disagreement-based active learning. *Found. Trends Mach. Learn.*, 7(2-3): 131–309.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.; Isola, P.; Saenko, K.; Efros, A. A.; and Darrell, T. 2018. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In *ICML*, 1994–2003.

Joshi, A. J.; Porikli, F.; and Papanikolopoulos, N. 2009. Multi-class active learning for image classification. In *CVPR*, 2372–2379.

Kang, G.; Wei, Y.; Yang, Y.; Zhuang, Y.; and Hauptmann, A. G. 2020. Pixel-Level Cycle Association: A New Perspective for Domain Adaptive Semantic Segmentation. In *NeurIPS*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012a. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*, 1097–1105.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012b. Imagenet Classification with Deep Convolutional Neural Networks. In *NeurIPS*, 1097–1105.

LeCun, Y.; Chopra, S.; Hadsell, R.; Ranzato, M.; and Huang, F. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Li, S.; Liu, C. H.; Lin, Q.; Wen, Q.; Su, L.; Huang, G.; and Ding, Z. 2021a. Deep Residual Correction Network for Partial Domain Adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(7): 2329–2344.

Li, S.; Liu, H. C.; Lin, Q.; Xie, B.; Ding, Z.; Huang, G.; and Tang, J. 2020. Domain Conditioned Adaptation Network. In *AAAI*, 11386–11393.

Li, S.; Lv, F.; Xie, B.; Liu, C. H.; Liang, J.; and Qin, C. 2021b. Bi-Classifier Determinacy Maximization for Unsupervised Domain Adaptation. In *AAAI*, 8455–8464.

Li, S.; Song, S.; Huang, G.; Ding, Z.; and Wu, C. 2018. Domain Invariant and Class Discriminative Feature Learning for Visual Domain Adaptation. *IEEE Trans. Image Process.*, 27(9): 4260–4273.

Li, S.; Xie, M.; Gong, K.; Liu, C. H.; Wang, Y.; and Li, W. 2021c. Transferable Semantic Augmentation for Domain Adaptation. In *CVPR*, 11516–11525.

Liu, W.; Wang, X.; Owens, J. D.; and Li, Y. 2020. Energy-based Out-of-distribution Detection. In *NeurIPS*.

Long, M.; Cao, Y.; Cao, Z.; Wang, J.; and Jordan, M. I. 2019. Transferable Representation Learning with Deep Adaptation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(12): 3071–3085.

Long, M.; Cao, Z.; Wang, J.; and Jordan, M. I. 2018. Conditional adversarial domain adaptation. In *NeurIPS*, 1647–1657.

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2017. Deep Transfer Learning with Joint Adaptation Networks. In *ICML*, 2208–2217.

Pan, S. J.; and Yang, Q. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10): 1345–1359.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 8024–8035.

Peng, X.; Usman, B.; Kaushik, N.; Hoffman, J.; Wang, D.; and Saenko, K. 2017. VisDA: The Visual Domain Adaptation Challenge. *CoRR*, abs/1710.06924.

Prabhu, V.; Chandrasekaran, A.; Saenko, K.; and Hoffman, J. 2021. Active Domain Adaptation via Clustering Uncertainty-weighted Embeddings. In *ICCV*, 8505–8514.

Prince, M. 2004. Does active learning work? A review of the research. *Journal of engineering education*, 93(3): 223–231.

Rai, P.; Saha, A.; Daumé III, H.; and Venkatasubramanian, S. 2010. Domain adaptation meets active learning. In *Proceedings of the NAACL HLT 2010 Workshop on Active Learning for Natural Language Processing*, 27–32.

Richter, S. R.; Vineet, V.; Roth, S.; and Koltun, V. 2016. Playing for data: Ground truth from computer games. In *ECCV*, 102–118.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*, 234–241.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. In *ECCV*, 213–226.

Saito, K.; Kim, D.; Sclaroff, S.; Darrell, T.; and Saenko, K. 2019. Semi-Supervised Domain Adaptation via Minimax Entropy. In *ICCV*, 8050–8058.

Saito, K.; Watanabe, K.; Ushiku, Y.; and Harada, T. 2018. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 3723–3732.

Schohn, G.; and Cohn, D. 2000. Less is More: Active Learning with Support Vector Machines. In *ICML*, 839–846.

Sener, O.; and Savarese, S. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *ICLR*.

Settles, B. 2009. Active learning literature survey.

Shui, C.; Zhou, F.; Gagné, C.; and Wang, B. 2020. Deep Active Learning: Unified and Principled Method for Query and Training. In *AISTATS*, volume 108, 1308–1318.

Sinha, S.; Ebrahimi, S.; and Darrell, T. 2019. Variational Adversarial Active Learning. In *ICCV*, 5971–5980.

Su, J.; Tsai, Y.; Sohn, K.; Liu, B.; Maji, S.; and Chandraker, M. 2020. Active Adversarial Domain Adaptation. In *WACV*, 728–737.

Teshima, T.; Sato, I.; and Sugiyama, M. 2020. Few-shot Domain Adaptation by Causal Mechanism Transfer. In *ICML*, 9458–9469.

Tsai, Y.; Hung, W.; Schulter, S.; Sohn, K.; Yang, M.; and Chandraker, M. 2018. Learning to Adapt Structured Output Space for Semantic Segmentation. In *CVPR*, 7472–7481.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous Deep Transfer Across Domains and Tasks. In *ICCV*, 4068–4076.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *CVPR*, 7167–7176.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 5018–5027.

Wang, D.; and Shang, Y. 2014. A new active labeling method for deep learning. In *IJCNN*, 112–119.

Xu, R.; Li, G.; Yang, J.; and Lin, L. 2019. Larger Norm More Transferable: An Adaptive Feature Norm Approach for Unsupervised Domain Adaptation. In *ICCV*, 1426–1435.

Yogamani, S. K.; Witt, C.; Rashed, H.; Nayak, S.; Mansoor, S.; Varley, P.; Perrotton, X.; O'Dea, D.; Pérez, P.; Hughes, C.; Horgan, J.; Sistu, G.; Chennupati, S.; Uricár, M.; Milz, S.; Simon, M.; and Amende, K. 2019. WoodScape: A Multi-Task, Multi-Camera Fisheye Dataset for Autonomous Driving. In *ICCV*, 9307–9317.

Zou, H.; Yang, J.; and Wu, X. 2021. Unsupervised Energy-based Adversarial Domain Adaptation for Cross-domain Text Classification. In *ACL*, 1208–1218.

Appendix

Contents

Dataset Details

VisDA-2017 (Peng et al. 2017) is a large-scale synthetic-2-real dataset for image classification competition. In total there are over 280k images from 12 categories. The images are split into three sets, i.e., a training set with 152,397 synthetic 2D renderings of 3D models, a validation set with 55,388 real images, and a test set with 72,372 real images. In this paper, we utilize the training and validation images as the source domain and the target domain, respectively.

Office-Home (Venkateswara et al. 2017) is a challenging benchmark, consisting of 15,500 images in 65 object classes. There are 4 extremely distinct domains: Artistic images (**Ar**), Clip Art (**Cl**), Product images (**Pr**), and Real-World

images (**Rw**). And we build twelve transfer tasks: Ar→Cl, Ar→Pr, . . . , Rw→Cl, Rw→Pr.

Office-31 (Saenko et al. 2010) is widely adopted by domain adaptation methods, comprising 31 categories of 4,110 images. It involves 3 different domains: **A**mazon (images downloaded from Amazon website), **D**SLR (images collected from digital SLR camera) and **W**ebcam (images recorded by web camera). We evaluate our method on 6 transfer tasks: A→D, A→W, . . . , W→D.

GTAV (Richter et al. 2016) contains 24,966 synthetic images with the resolution of 1914×1052, which are rendered using the open-world video game "Grand Theft Auto V". There are 19 semantic categories that are compatible with the semantic categories in the Cityscapes dataset.

Cityscapes (Cordts et al. 2016) includes 5,000 urban scene images of resolution 2048×1024. They are split into a training set with 2,975 images and a validation set with 500 images. Similar to (Tsai et al. 2018; Kang et al. 2020), we evaluate our model on the validation set and report the mIoU of the common 19 classes.

Implementation Details

Image classification. We implement our experiments on the widely-used PyTorch (Paszke et al. 2019) platform. For a fair comparison, our backbone network is identical to the competitive methods and is also pre-trained on ImageNet (Krizhevsky, Sutskever, and Hinton 2012b). For optimizer, we use the AdaDelta with a learning rate of 0.1 and train for 50 epochs. The batch size is 32. We adopt a unified set of hyper-parameters throughout the VisDA-2017, Office-Home, Office-31 datasets, where γ=0.01, α_1=50, and α_2=2. We use random-crop images for data augmentation during training and use center-crop images for testing. We follow the standard protocol (Fu et al. 2021; Su et al. 2020; Prabhu et al. 2021) to use the whole target domain as testing data. We carry out experiments with five different random seeds and report the average classification accuracy.

Semantic segmentation. We employ the DeepLab-v2 (Chen et al. 2018) as the feature extractor which is composed of the backbone ResNet-101 (He et al. 2016) pre-trained on ImageNet (Krizhevsky, Sutskever, and Hinton 2012b) and the Atrous Spatial Pyramid Pooling (ASPP) module. Following (Tsai et al. 2018), sampling rates of ASPP module are fixed as $\{6, 12, 18, 24\}$. To train the segmentation network, we adopt the SGD optimizer where the momentum is 0.9 and the weight decay is 10^{-4}. The learning rate is initially set to 2.5×10^{-4} and is decreased following a 'poly' learning rate policy with a power of 0.9. γ is constantly set to 0.001 and α_1, α_2 are set to 50 and 2, respectively. The source input image is resized to 1280×720 and the target input image is resized to 1024×512. We respectively select 1%, 2.5%, and 5% Cityscapes training samples as total labeling budget and query for pixel-level semantic annotation of the whole image.

Labeling budget. Following the previous active domain adaptation work (Fu et al. 2021), the labeling budge in each selection round is **1% of all target samples**. In Table 1, Table 2 of the main paper, we perform 5 rounds (in total 5% target samples) for VisDA-2017, Office-Home, Office-31 and report the classification accuracy of final model. Similarly, we perform 20 rounds (in total 20% target samples) for VisDA-2017 and provide the accuracy after each round in Fig. 3 of the main paper. Unless specified otherwise, we perform 5 round selections throughout the analysis below.

Baseline implementation. Note that partial reported results are copied from (Fu et al. 2021) if the experimental setup is the same. We now elaborate on our implementation of other baseline algorithms:

- Random: The naive baseline of randomly selecting 1% target samples to query at each round.
- BvSB (Joshi, Porikli, and Papanikolopoulos 2009): We compute the difference between the largest and second-largest predicted probability as an uncertainty metric and then select the bottom 1% samples sorted according to the value of this metric at each round.
- Entropy (Wang and Shang 2014): We utilize the entropy of the model outputs as a confidence measurement for each unlabeled target sample and then select the top 1% samples sorted according to the value of this measurement at each round.
- CoreSet (Sener and Savarese 2018): A representative sampling algorithm using core-set selection. We implement CoreSet using the released code: https://github.com/ozansener/active_learning_coreset.
- WAAL (Shui et al. 2020): A hybrid sampling algorithm which models the interactive procedure in active learning as distribution matching by adopting Wasserstein distance. We implement WAAL using the released code: https://github.com/cjshui/WAAL
- BADGE (Ash et al. 2020): We compute "gradient embeddings" through taking the gradient of model loss w.r.t. classifier weights. Next, the k-MEANS++ scheme (Arthur and Vassilvitskii 2007) is run on these embeddings to yield a bach of samples. We implement BADGE using the released code: https://github.com/JordanAsh/badge
- AADA (Su et al. 2020): In AADA, a domain discriminator G_d is learned to distinguish the features obtained from the feature extractor G_f whether come from the source domain or the target domain. For active sampling strategy, all unlabeled target data are scored via the selection criterion $s(x)$ defined in the original paper: $s(x) = \frac{1-G_d^*(G_f(x))}{G_d^*(G_f(x))}\mathcal{H}(G_y(G_f(x)))$, where G_y is the class predictor and $\mathcal{H}$ denotes the model entropy. Then, we select top 1% samples for labeling at each round.
- DBAL (de Mathelin, Mougeot, and Vayatis 2021): A discrepancy-based active learning for domain adaptation. We implement DBAL using the released code: https://github.com/antoinedemathelin/dbal
- TQS (Fu et al. 2021) A state-of-the-art active domain adaptation method, which selects the most informative target samples by an ensemble of transferable committee, transferable uncertainty, and transferable domain-

ness. We implement TQS using the released code https://github.com/thuml/Transferable-Query-Selection

- CLUE (Prabhu et al. 2021): A more recent example that jointly captures uncertainty and diversity for active domain adaptation. We implement CLUE according to Algorithm 1 provided in (Prabhu et al. 2021). Consider with the original work, we also optimize a semi-supervised adversarial entropy loss and perform cross-validation on source data to tune the hyperparameters.

Further Analysis for EADA

More details about free energy biases. As a matter of fact, free energy biases exhibit when two domains have distinct distributions, i.e., $p_s(x) \neq p_t(x)$. Statistically, Eq. (4) indicates that $\mathcal{F}(x)$ could be substituted for $p(x)$ where $\mathcal{F}(x)$ is the free energy of x from our model. $\mathcal{P}_s(\mathcal{F})$ and $\mathcal{P}_t(\mathcal{F})$ are free energy distributions of source and target, respectively. Thus, if $p_s(x) \neq p_t(x)$, $\mathcal{P}_s(\mathcal{F}) \neq \mathcal{P}_t(\mathcal{F})$. Empirically, we randomly divide source domain into two data sets ($\mathcal{S}_1$, $\mathcal{S}_2$), where $\mathcal{S}_1$ is used for supervised training while both $\mathcal{S}_2$ and $\mathcal{T}$ are used for testing. Then, we plot the free energy distributions of these three sets in Fig. 7. The results show that $\mathcal{S}_1$ (green, circle) and $\mathcal{T}$ (red, star) still exhibit biases while $\mathcal{S}_1$ (green, circle) and $\mathcal{S}_2$ (blue, cross) do not.

Effect of free energy alignment. Our key observation is that the values of free energy on target samples are considerably higher than those on source ones, called *free energy biases*. Mathematically, we show that these biases exhibit when two domains have different distributions. Naturally, one can treat this as a surrogate to reflect the divergence across domains. By designing a simple regularization term, we are allowed to minimize the biases, which to some extent aligns the distribution of source and target data. Experimentally, by calculating two common measurements, $\mathcal{A}$-distance and MMD, on task A $\rightarrow$ D (see Fig. 8), we observe that EADA achieves a lower $\mathcal{A}$-distance/MMD, implying lower domain divergence.

Effect of regularization weight γ. We perform parameter sensitivity analysis to evaluate the sensitivity of EADA on task Rw $\rightarrow$ Pr of Office-Home and A $\rightarrow$ D of Office-31. As shown in Fig. 9, we select regularization weight from $\gamma \in \{0.001, 0.005, 0.01, 0.05, 0.1\}$. It can be seen that the performance of EADA increases first and then decreases slightly, which is a slow bell-shaped curve. Overall, these results indicate that the robustness of EADA under a wide range of parameter choices.

Visualizing selected sampels. To understand the behavior of EADA intuitively, we visualize the top 10 samples selected by EADA at Round 1 and Round 2 on VisDA-2017 in Fig. 6. As seen, EADA manages to sample instances from every class, and more representative and informative target samples are selected.

Error rate of the selected samples. To explore the relationship between sampling strategy and predictions by the current model, we compare the error rate of the selected samples with two popular sampling strategies in Active DA, i.e., adopting the output of a domain discriminator (AADA) or using the distance to the cluster centroids (CLUE), in Table 5. Interestingly, the error rate of selected samples decreases as the number of rounds increases. We conjecture that the accuracy and generalization of the model will be significantly improved when a limited number of target samples are labeled and used to train the model. However, contrary to intuition, we find that selecting more incorrect target samples may not lead to performance gains in the target domain. These results suggest that a good active selection strategy should identify those samples that are representative and informative. We see it as future work.

Method / Round	1	2	3	4	5
AADA	83.6	79.2	75.5	75.3	77.3
CLUE	36.8	26.5	26.2	24.0	22.2
EADA	70.2	67.3	67.1	64.6	66.5

Table 5: Error rate of the selected samples in each round on VisDA-2017 with **5%** target samples as the labeling budget.

$\delta_s = 1.0$ (default)	$\delta_s = 0.5$	$\delta_s = 0.1$
88.0	88.3	**88.7**

Table 6: Analysis of the trade-offs on $\mathcal{L}_{nll}$.

Analysis of the trade-offs on $\mathcal{L}_{nll}$. Given numerous labeled data from the source domain and a small quota of the target domain, we compute the negative log-likelihood loss over all available labeled data. For simplicity, in the main paper, we directly add the two supervised losses in Eq. (9) without trade-offs. However, we argue that the trade-offs may be needed when we identify and annotate a few target samples [2]. Here, we employ a scalar weight to re-define the overall objective:

$$\min_{\theta} \delta_s \mathbb{E}_{(x,y)\sim\mathcal{S}} \mathcal{L}_{nll}(x, y; \theta) + \delta_t \mathbb{E}_{(x,y)\sim\mathcal{T}_l} \mathcal{L}_{nll}(x, y; \theta) + \gamma \mathbb{E}_{x\sim\mathcal{T}_u} \mathcal{L}_{fea}(x; \theta),$$

where δ_s and δ_t are scalar weights. We fix loss weight $\delta_t = 1$ and study three ways for δ_s: $\delta_s = 1.0$ (default), $\delta_s = 0.5$ and $\delta_s = 0.1$. Looking at Table 6, it is apparent that $\delta_s = 0.1$ achieves the best accuracy, verifying the weight of source supervised loss should indeed be small when having access to a few labeled target samples.

Additional Results

Varying labeling budget on VisDA-2017. In Fig. 3 of the main paper, we vary the target labeling budget from 0% to 20% with existing Active DA methods. For completeness, we present the performance results of other active learning methods in Fig. 10. EADA consistently beats all previous active learning methods. In addition, due to the existence of domain shift, traditional active learning methods perform poorly on the target domain even few target samples are labeled. We can see that EADA performs best even with a

[2] See CLUE (Prabhu et al. 2021), the loss weights for source and target supervised loss are set to 0.1 and 1.0, respectively.

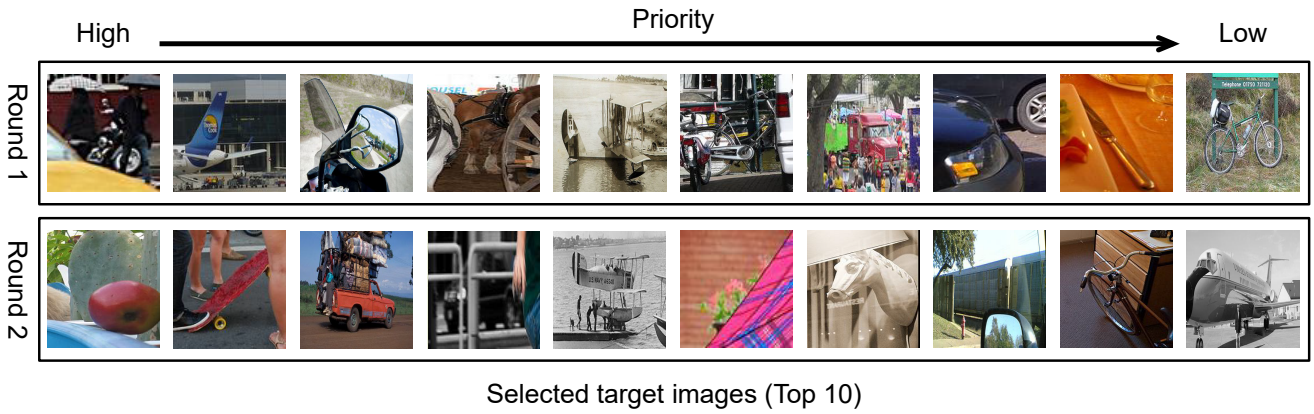


Figure 6: Visualization of instances selected by our method at Round 1 and Round 2 on VisDA-2017.

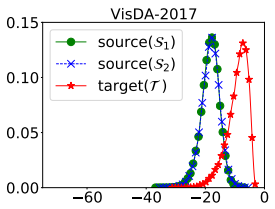


Figure 7: Free energy biases

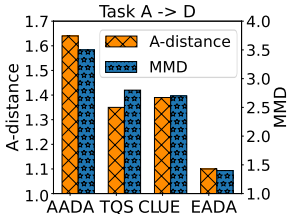
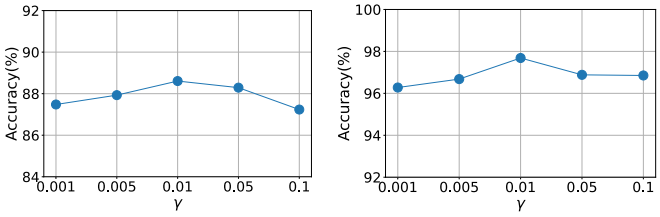
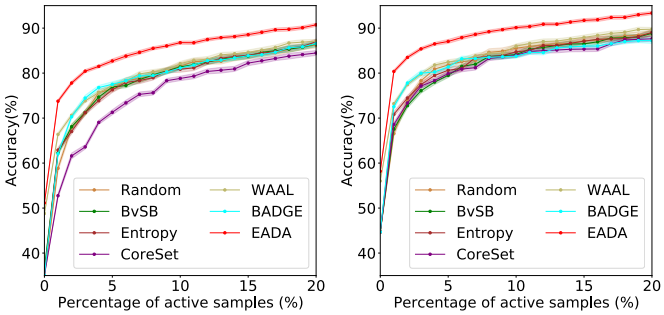


Figure 8: $\mathcal{A}$-distance



(a) Office-Home: Rw → Pr (b) Office-31: A → D

Figure 9: Effect of regularization weight γ.



(a) ResNet-18 (b) ResNet-50

Figure 10: Comparison results of varying the percentage of labeled target samples on **VisDA-2017** with ResNet-18/50.

very limited labeling budget (<5%), comparable to other active learning approaches that allow for larger budgets (about 12%), suggesting that EADA requires very little budget, yet significantly improves adaptation performance.

Detailed Theoretical Proofs

Gradient analysis about $\mathcal{L}_{nll}$. In the method part, we mention that the second term in $\mathcal{L}_{nll}$ (Eq. (7) in the main paper) will cause the energies of all answers to be pulled up. The energy of the correct answer is also pulled up, but not as hard as it is pushed down by the first term. In other words, the energy of the correct answer will be pulled down and other energies of incorrect answers will be pulled up when training with the $\mathcal{L}_{nll}$. Here, we analyze the gradient of $\mathcal{L}_{nll}$ to corroborate this statement.

Firstly, we emphasize the equation of $\mathcal{L}_{nll}$ with τ=1 again:

$$\begin{aligned}\mathcal{L}_{nll}(x,y;\theta) &= E(x,y;\theta) + \log \sum_{c\in\mathcal{Y}} \exp\left(-E(x,c;\theta)\right) \\ &= E(x,y;\theta) - \mathcal{F}(x;\theta)\,. \end{aligned} \tag{17}$$

And with the energy function $E(x,y;\theta)$, the conditional probability of label y give the input x can be estimated through the Gibbs distribution:

$$p(y|x;\theta) = \frac{\exp(-E(x,y;\theta))}{\sum_{c\in\mathcal{Y}} \exp(-E(x,c;\theta))}\,. \tag{18}$$

In backward propagation, gradient of $\mathcal{L}_{nll}(x,y;\theta)$ is given by:

$$\frac{\partial \mathcal{L}_{nll}(x,y;\theta)}{\partial\theta} = \frac{\partial E(x,y;\theta)}{\partial\theta} - \frac{\partial \mathcal{F}(x;\theta)}{\partial\theta}\,. \tag{19}$$

For the second term in Eq. (19), we have

$$\begin{aligned}\frac{\partial \mathcal{F}(x;\theta)}{\partial\theta} &= \frac{\partial\left(-\log\sum_{c\in\mathcal{Y}}\left(\exp\left(-E(x,c;\theta)\right)\right)\right)}{\partial\theta} \\ &= \sum_{c\in\mathcal{Y}} p(c|x;\theta)\frac{\partial(E(x,c;\theta))}{\partial\theta}\,. \end{aligned} \tag{20}$$

From above, we can get

$$\begin{aligned}\frac{\partial \mathcal{L}_{nll}(x,y;\theta)}{\partial \theta} &= \frac{\partial E(x,y;\theta)}{\partial \theta} - \sum_{c\in\mathcal{Y}} p(c|x;\theta)\frac{\partial (E(x,c;\theta))}{\partial \theta} \\ &= (1-p(y|x;\theta))\frac{\partial E(x,y;\theta)}{\partial \theta} \\ &\quad - \sum_{c\in\mathcal{Y}\setminus\{y\}} p(c|x;\theta)\frac{\partial (E(x,c;\theta))}{\partial \theta}.\end{aligned} \tag{21}$$

From Eq. (21), we can clearly see that the first term will pull down the energy of the correct answer and the second term will pull down the energies of all the incorrect answers.

Toy example Consider a C-class classification problem in a simple energy-based model (EBM) where $x \in \mathbb{R}^D$ denotes the input, $y \in \{1, \cdots, C\}$ denotes the label. And the network is a single layer fully-connected network with parameter $\mathbf{W} \in \mathbb{R}^{C\times D}$. The output for x is

$$O(x) = \mathbf{W}x,$$

where

$$\mathbf{W} = \begin{pmatrix} \omega_{11} & \cdots & \omega_{1D} \\ \vdots & \ddots & \vdots \\ \omega_{C1} & \cdots & \omega_{CD} \end{pmatrix} = \begin{pmatrix} \omega_1 & \cdots & \omega_C \end{pmatrix}^\top.$$

In this discrete prediction task, the negative log-likelihood loss is

$$\mathcal{L}_{nll}(x,y;\mathbf{W}) = E(x,y;\mathbf{W}) - \mathcal{F}(x;\mathbf{W}), \tag{22}$$

where $E(x,y;\mathbf{W})$ denotes the energy of correct answer and $\mathcal{F}(x;\mathbf{W})$ denotes free energy of x. And they can be calculated as:

$$E(x,j;\mathbf{W}) = O(x)[j] = \omega_j^\top x, \forall j \in \{1,\cdots,C\}, \tag{23}$$

$$\begin{aligned}\mathcal{F}(x;\mathbf{W}) &= -\log \sum_{c=1}^{C} \exp\left(-E(x,c;\mathbf{W})\right) \\ &= -\log \sum_{c=1}^{C} \exp\left(-\omega_c^\top x\right).\end{aligned} \tag{24}$$

Following, we calculate the gradient of the negative log-likelihood loss $\mathcal{L}_{nll}(x,y;\mathbf{W})$ step by step.

First of all, for $E(x,j;\mathbf{W})$ in Eq. (23), we have

$$\nabla E(x,j;\mathbf{W}) = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ x_1 & \cdots & x_D \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{pmatrix} = x(e^j)^\top, \tag{25}$$

where $e^j \in \mathbb{R}^D$, and $e_i^j = 1$ when $i = j$, 0 otherwise. Then, according to Eq. (20), we have

$$\begin{aligned}\nabla \mathcal{F}(x;\mathbf{W}) &= \sum_{c=1}^{C} p(c|x)\nabla E(x,c;\mathbf{W}) \\ &= x\begin{pmatrix} p(1|x) \\ \vdots \\ p(C|x) \end{pmatrix}.\end{aligned} \tag{26}$$

where

$$p(j|x) = \frac{\exp\left(-E(x,j)\right)}{\exp\left(-\mathcal{F}(x)\right)}, \forall j \in \{1,\cdots,C\}. \tag{27}$$

Lastly, combining Eq. (22), Eq. (25) and Eq. (26), we have the gradient of $\mathcal{L}_{nll}(x,y;\mathbf{W})$ as follows:

$$\begin{aligned}\nabla \mathcal{L}_{nll}(x,y;\mathbf{W}) &= \nabla E(x,y;\mathbf{W}) - \nabla \mathcal{F}(x;\mathbf{W}) \\ &= x\begin{pmatrix} -p(1|x) \\ \vdots \\ 1-p(y|x) \\ \vdots \\ -p(C|x) \end{pmatrix}.\end{aligned} \tag{28}$$

Proof of Lemma 1

Proof. A correct toy model prediction on the labeled source sample (x,y) means that the inequality

$$p(y|x) > p(j|x), \forall j \in \{1,\cdots,C\}, j \neq y \tag{29}$$

holds. And from Eq. (25), it is easy to see that

$$\langle \nabla E(x,i;\mathbf{W}), \nabla E(x,j;\mathbf{W}) \rangle = \begin{cases} \|x\|^2, & i = j, \\ 0, & \text{otherwise}, \end{cases} \tag{30}$$

where $\langle \cdot,\cdot \rangle$ denotes the inner product. Then with Eq. (25), Eq. (26), and Eq. (28), we explicitly calculate the inner product between $\nabla \mathcal{L}_{nll}(x,y;\mathbf{W})$ and $\nabla E(x,y;\mathbf{W})$

$$\begin{aligned}&\langle \nabla \mathcal{L}_{nll}(x,y;\mathbf{W}), \nabla \mathcal{F}(x;\mathbf{W}) \rangle \\ &= \langle \nabla E(x,y;\mathbf{W}) - \nabla \mathcal{F}(x;\mathbf{W}), \nabla \mathcal{F}(x;\mathbf{W}) \rangle \\ &= \langle \nabla E(x,y;\mathbf{W}), \nabla \mathcal{F}(x;\mathbf{W}) \rangle - \langle \nabla \mathcal{F}(x;\mathbf{W}), \nabla \mathcal{F}(x;\mathbf{W}) \rangle \\ &= \left\langle \nabla E(x,y;\mathbf{W}), \sum_{c=1}^{C} p(c|x)\nabla E(x,c;\mathbf{W}) \right\rangle \\ &\quad - \left\langle \sum_{c=1}^{C} p(c|x)\nabla E(x,c;\mathbf{W}), \sum_{c=1}^{C} p(c|x)\nabla E(x,c;\mathbf{W}) \right\rangle \\ &= p(y|x)\|x\|^2 - \sum_{c=1}^{C} p(c|x)^2 \|x\|^2 \\ &= p(y|x)(1-p(y|x))\|x\|^2 - \sum_{c=1,c\neq y}^{C} p(c|x)^2\|x\|^2 \\ &= p(y|x)\left(\sum_{c=1,c\neq y}^{C} p(c|x)\right)\|x\|^2 - \sum_{c=1,c\neq y}^{C} p(c|x)^2\|x\|^2 \\ &= \sum_{c=1,c\neq y}^{C} p(c|x)(p(y|x) - p(c|x))\|x\|^2.\end{aligned} \tag{31}$$

Next with the Eq. (29) it is easy to know

$$\langle \nabla \mathcal{L}_{nll}, \nabla \mathcal{F} \rangle > 0. \tag{32}$$

It should be pointed out that the sample x with norm $\|x\| = 0$ is out of consideration because it is meaningless in our toy example. Consequently, each term of the sum in the last term of Eq. (31) is greater than zero, resulting in the sum of them is greater than zero.

□

Proof of Lemma 2

Proof. Before embarking on the proof we make some preparation include some conventions of some symbols. In order to show our proof more conveniently, all the symbols with a prime superscript denotes the changed values of the same one without superscript after one step of gradient descent, e.g. $E(x,y) = E(x,y;\mathbf{W})$ and $E'(x,y) = E(x,y;\mathbf{W}')$.

Firstly, a correct toy model prediction on the labeled source sample (x,y) indicates that $\forall j \in \{1,\cdots,C\}\backslash\{y\}$

$$E(x,y) < E(x,j). \tag{33}$$

Then we have a serial of equivalent propositions

$$\begin{aligned}
&\mathcal{F}(x,\mathbf{W}) > \mathcal{F}(x,\mathbf{W}') \\
\Leftrightarrow &-\log\sum_{c=1}^{C}\exp\left(-E(x,c)\right) > -\log\sum_{c=1}^{C}\exp\left(-E'(x,c)\right) \\
\Leftrightarrow &\sum_{c=1}^{C}\exp\left(-E(x,c)\right) < \sum_{c=1}^{C}\exp\left(-E'(x,c)\right).
\end{aligned} \tag{34}$$

From Eq. (23) and Eq. (28) and one step of gradient descent, we can explicitly express E' as follows

$$\begin{aligned}
E'(x,y) &= \omega_y^\top x - \eta\left(1-p\left(y|x\right)\right)\|x\|^2 \qquad (35)\\
&= E(x,y) - \eta\left(1-p\left(y|x\right)\right)\|x\|^2, \\
E'(x,j) &= \omega_j^\top x + \eta p\left(j|x\right)\|x\|^2 \qquad (36)\\
&= E(x,j) + \eta p\left(j|x\right)\|x\|^2, j \in \{1,\cdots,C\}\backslash\{y\}.
\end{aligned}$$

With the Eq. (35) and Eq. (36), we can continue our deduction of equivalent propositions of Eq. (34) as follows

$$\begin{aligned}
&\sum_{c=1}^{C}\exp\left(-E(x,c)\right) < \sum_{c=1}^{C}\exp\left(-E'(x,c)\right) \\
&\Leftrightarrow \sum_{c=1}^{C}\exp\left(-E(x,c)\right) < \\
&\sum_{c=1,c\neq y}^{C}\exp\left(-E(x,c) - \eta p\left(c|x\right)\|x\|^2\right) \\
&+ \exp\left(-E(x,y) + \eta\left(1-p\left(y|x\right)\right)\|x\|^2\right) \\
&\Leftrightarrow \sum_{c=1}^{C}\exp\left(E(x,y) - E(x,c)\right) < \\
&\sum_{c=1,c\neq y}^{C}\exp\left(E(x,y) - E(x,c) - \eta p\left(c|x\right)\|x\|^2\right) \\
&+ \exp\left(\eta\left(1-p\left(y|x\right)\right)\|x\|^2\right) \\
&\Leftrightarrow \sum_{c=1,c\neq y}^{C}\exp\left(E(x,y) - E(x,c)\right) + e^0 < \\
&\sum_{c=1,c\neq y}^{C}\exp\left(E(x,y) - E(x,c) - \eta p\left(c|x\right)\|x\|^2\right) \\
&+ \exp\left(\eta\sum_{c=1,c\neq y}^{C} p\left(y|x\right)\|x\|^2\right).
\end{aligned} \tag{37}$$

Equipped with Lemma 3, with $\eta > 0$ and Eq. (33), we can clearly see that in the final inequality of Eq. (37),

$$\begin{aligned}
0 &> E(x,y) - E(x,c), \\
\eta p\left(c|x\right)\|x\|^2 &> 0, \\
c &= 1,\cdots,C, c \neq y,
\end{aligned}$$

which makes it be an example of Lemma 3, so the original inequality holds for our assumption. □

Lemma 3. *Let there be* $2n+1$ *real numbers* $a, a_1, \cdots, a_n$ *and* $b_1, \cdots, b_n$*, if*

$$a > a_i, b_i > 0, i = 1, \cdots, n,$$

we have

$$e^{a+\sum_{i=1}^{n} b_i} + \sum_{i=1}^{n} e^{a_i - b_i} > e^a + \sum_{i=1}^{n} e^{a_i},$$

Proof. Because

$$a > a_i, b_i > 0, i = 1, \cdots, n,$$

so

$$e^{a+\sum_{i=j}^{n} b_i} > e^{a_j}, 1 - e^{-b_j} > 0, j = 1, \cdots, n.$$

Then we obtain

$$e^{a+\sum_{i=j}^{n} b_i}\left(1-e^{-b_j}\right) > e^{a_j}\left(1-e^{-b_j}\right), j = 1, \cdots, n. \tag{38}$$

With some simple algebra, from Eq. (38) we can find that

$$e^{a+\sum_{i=j}^{n} b_i} + e^{a_j - b_j} > e^{a+\sum_{i=j+1}^{n} b_i} + e^{a_j}, j = 1, \cdots, n. \tag{39}$$

Equipped with Eq. (39), we can start our deduction as follows:

$$\begin{aligned}
&e^{a+\sum_{i=1}^{n} b_i} + \sum_{i=1}^{n} e^{a_i - b_i} \\
&= e^{a+\sum_{i=1}^{n} b_i} + e^{a_1 - b_1} + \sum_{i=2}^{n} e^{a_i - b_i} \\
&> e^{a+\sum_{i=2}^{n} b_i} + \sum_{i=1}^{1} e^{a_i} + \sum_{i=2}^{n} e^{a_i - b_i} \\
&= e^{a+\sum_{i=2}^{n} b_i} + e^{a_2 - b_2} + \sum_{i=1}^{1} e^{a_i} + \sum_{i=3}^{n} e^{a_i - b_i} \\
&> e^{a+\sum_{i=3}^{n} b_i} + \sum_{i=1}^{2} e^{a_i} + \sum_{i=3}^{n} e^{a_i - b_i} \\
&= e^{a+\sum_{i=3}^{n} b_i} + e^{a_3 - b_3} + \sum_{i=1}^{2} e^{a_i} + \sum_{i=4}^{n} e^{a_i - b_i} \\
&> e^{a+\sum_{i=4}^{n} b_i} + \sum_{i=1}^{3} e^{a_i} + \sum_{i=4}^{n} e^{a_i - b_i} \\
&\vdots \\
&> e^{a+\sum_{i=j}^{n} b_i} + \sum_{i=1}^{j-1} e^{a_i} + \sum_{i=j}^{n} e^{a_i - b_i} \\
&\vdots \\
&> e^a + \sum_{i=1}^{n} e^{a_i}.
\end{aligned} \tag{40}$$

□

Proof of Theorem 1

In this part, we provide the detailed proof of Theorem 1. Before embarking on the proof we make some preparation, here, we define β-smooth (Bubeck 2015):

Definition 1. *A function f is β-smooth, if the gradient ∇f is β-Lipschitz (Bubeck 2015) that is*

$$\|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\|. \tag{41}$$

Proof. For convenient, we abbreviate $\mathcal{L}_{nll}(x, y; \theta)$ to $\mathcal{L}_{nll}(\theta)$ and $\mathcal{F}(x; \theta)$ to $\mathcal{F}(\theta)$. Then with the smooth assumption and Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&\mathcal{F}(\theta - \eta\nabla\mathcal{L}_{nll}(\theta)) - \mathcal{F}(\theta) - \langle\nabla\mathcal{F}(\theta), -\eta\nabla\mathcal{L}_{nll}(\theta)\rangle \\
&= \int_0^1 \langle\nabla\mathcal{F}(\theta - t\eta\nabla\mathcal{L}_{nll}(\theta)), -\eta\nabla\mathcal{L}_{nll}(\theta)\rangle \,\mathrm{d}t \\
&- \langle\nabla\mathcal{F}(\theta), -\eta\nabla\mathcal{L}_{nll}(\theta)\rangle \\
&= \int_0^1 \langle\nabla\mathcal{F}(\theta - t\eta\nabla\mathcal{L}_{nll}(\theta)) - \nabla\mathcal{F}(\theta), -\eta\nabla\mathcal{L}_{nll}(\theta)\rangle \,\mathrm{d}t \\
&\leq \left| \int_0^1 \langle\nabla\mathcal{F}(\theta - t\eta\nabla\mathcal{L}_{nll}(\theta)) - \nabla\mathcal{F}(\theta), -\eta\nabla\mathcal{L}_{nll}(\theta)\rangle \,\mathrm{d}t \right| \\
&\leq \int_0^1 \|\nabla\mathcal{F}(\theta - t\eta\nabla\mathcal{L}_{nll}(\theta) - \nabla\mathcal{F}(\theta))\| \, \|-\eta\nabla\mathcal{L}_{nll}(\theta)\| \,\mathrm{d}t \\
&\leq \int_0^1 t\beta\eta^2 \|\nabla\mathcal{L}_{nll}(\theta)\|^2 \,\mathrm{d}t \\
&= \frac{\beta\eta^2}{2} \|\nabla\mathcal{L}_{nll}(\theta)\|^2.
\end{aligned} \tag{42}$$

From Eq. (42), it is easy to see that

$$\begin{aligned}
&\mathcal{F}(\theta - \eta\nabla\mathcal{L}_{nll}(\theta)) - \mathcal{F}(\theta) \\
&\leq \frac{\beta\eta^2}{2} \|\nabla\mathcal{L}_{nll}(\theta)\|^2 - \eta\langle\nabla\mathcal{F}(\theta), \nabla\mathcal{L}_{nll}(\theta)\rangle.
\end{aligned} \tag{43}$$

And by our assumption on the gradient norm and inner product, the right part of Eq. (43) becomes

$$\begin{aligned}
&\frac{\beta\eta^2}{2} \|\nabla\mathcal{L}_{nll}(\theta)\|^2 - \eta\langle\nabla\mathcal{F}(\theta), \nabla\mathcal{L}_{nll}(\theta)\rangle \\
&< \frac{\beta G^2\eta^2}{2} - \eta\langle\nabla\mathcal{F}(\theta), \nabla\mathcal{L}_{nll}(\theta)\rangle \\
&< \frac{\beta G^2\eta^2}{2} - \eta\varepsilon.
\end{aligned} \tag{44}$$

Let $g(\eta) = \frac{\beta G^2\eta^2}{2} - \eta\varepsilon$. Because $\frac{\beta G^2}{2} > 0$, so $g(\eta)$ is convex for $\forall\eta \in \mathbb{R}$. Then by convexity of g, for our fixed range of learning rate $\eta \in \left(0, \frac{2\varepsilon}{\beta G^2}\right)$,

$$\begin{aligned}
g(\eta) &= g\left(0\left(1 - \frac{\eta}{\frac{2\varepsilon}{\beta G^2}}\right) + \frac{\eta}{\frac{2\varepsilon}{\beta G^2}}\frac{2\varepsilon}{\beta G^2}\right) \\
&< \left(1 - \frac{\eta}{\frac{2\varepsilon}{\beta G^2}}\right) g(0) + \frac{\eta}{\frac{2\varepsilon}{\beta G^2}} g\left(\frac{2\varepsilon}{\beta G^2}\right) \\
&= 0.
\end{aligned} \tag{45}$$

Eventually, cooperate Eq. (43), Eq. (44) and Eq. (45), we proof that

$$\mathcal{F}(\theta - \eta\nabla\mathcal{L}_{nll}(\theta)) - \mathcal{F}(\theta) < 0. \tag{46}$$

□