Improving Clustering Uncertainty-weighted Embeddings for Active Domain Adaptation

Sheng-Feng Wu
Dept. of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
r08922134@csie.ntu.edu.tw

Hsuan-Tien Lin
Dept. of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
htlin@csie.ntu.edu.tw

***Abstract*—Domain adaptation generalizes deep neural networks to new target domains under domain shift. Active domain adaptation (ADA) does so efficiently by allowing the learning model to strategically ask data annotation questions. The state-of-the-art active domain adaptation via clustering uncertainty-weighted embeddings (ADA-CLUE) uses uncertainty-weighted clustering to identify target instances for labeling. In this work, we carefully study how ADA-CLUE balances uncertainty and diversity during active learning. We compare the original ADA-CLUE with a variant that weights clusters by a constant instead of by the uncertainty, and confirm that constant-weighted clustering sampling outperforms ADA-CLUE at early stages due to its stability. We then merge constant-weighted sampling and uncertainty-weighted sampling with a threshold to get the best of the two worlds. The merged solution, called CLUE with a loop threshold, is shown to be an empirically better choice than the original ADA-CLUE.**

***Index Terms*—domain adaptation, active learning**

I. Introduction

When trained on large-scale datasets, deep convolutional neural networks learn representations that are generically useful across a variety of tasks and visual domains [1], [2]. However, collecting large amounts of labeled data from scratch is time-consuming and impractical. Therefore, our goal is to use a single large dataset to train models that generalize well to novel datasets and tasks.

Due to covariate shift [3], however, models generalize poorly to novel datasets and tasks. Therefore, we seek to use cheaper sources of labeled data to train models that generalize to real-world targets [4], [5]. Consider, for example, an image classifier trained on synthetic or semi-synthetic images, which are abundant and fully labeled but which inevitably follow distributions that do not reflect real images. For object recognition, we seek to adapt models trained on synthetic data to real-world datasets [6] such as the Cityscapes dataset [7].

To this end, researchers have proposed domain adaptation (DA), a new research area in machine learning [4], [5]. In this scenario, training and test sets are termed source and target domains, respectively. Domain adaptation generally involves using labeled source data to train a model that generalizes to a target domain by minimizing the difference between domain distributions.

Domain adaptation using labeled source data and unlabeled target data in the training phase is unsupervised domain adaptation (UDA). Although domain adaptation provides a good starting point, the performance of UDA methods often falls far behind their supervised counterparts [8], [9]. In such cases, adding labeled data from the target domain could translate to performance benefits. Active learning (AL) selects queries to retain the most useful data for training from a large unlabeled data pool. Whereas domain adaptation (DA) transfers knowledge from a label-rich source domain to an unlabeled target domain, AL queries labels to yield a small subset of the most relevant samples. To this end, a series of active domain adaptation (ADA) methods has been developed. Given labeled data in a source domain, unlabeled data in a target domain, and the ability to obtain labels for a fixed budget of target instances, we seek to select target instances for labeling and update the model's representations to maximize performance on the target test set.

The traditional AL setting typically focuses on techniques to select samples to efficiently learn a model from scratch, rather than adapting a model under a domain shift. In traditional AL, labels are acquired for samples by committee [10], uncertainty [11]–[14], or representativeness [15]–[17]. Although traditional AL methods dramatically lower human annotation costs, they are impractical when collecting out-of-distribution test data [18], [19]. As neural networks are poorly calibrated, model uncertainty is not reliable [20]. Given such model miscalibration, high confidence scores do not imply a high likelihood of correctness; thus the sampled examples are not the most uncertain ones. On the other hand, the main difference between ADA and traditional AL is that the calibration of these estimates in the target domain depends on the degree of the domain shift. These limitations complicate deployment of traditional AL. The major challenge of ADA is thus the selection criterion under domain shift.

Prior ADA work [21]–[23] has also underscored the importance of identifying instances that are both uncertain and diverse. Active adversarial domain adaptation (AADA) [23] combines uncertainty with diversity measured by *targetness* under a learned domain discriminator. However, such targetness does not ensure that the selected instances are representative of the entire target data distribution (i.e., not outliers), or dissimilar to one another. Ash et al. [15] instead propose clustering in a hallucinated "gradient embedding" space.

However, this depends on distance-based clustering in high-dimensional spaces, which often leads to suboptimal results. The state-of-the-art method for ADA is active domain adaptation via clustering uncertainty-weighted embedding (ADA-CLUE) [22], which selects the instance closest to each cluster centroid under the uncertainty-weighted embedding of target instances and then optimizes a semi-supervised adversarial entropy loss to induce domain alignment, jointly capturing uncertainty and diversity for active DA.

In this work, we systematically design the new algorithm and package to analyze whether the quality of the uncertainty is important. We propose active domain adaptation via density-weighted uncertainty sampling (ADA-DWUS), a label acquisition strategy for active DA. We select effective target samples on the basis of their diversity as well as the classifier uncertainty for the domain adaptation task. First, the unlabeled target pool is grouped into clusters using the k-means++ algorithm [24]. We leverage this to select the most effective target samples from the diverse target pool obtained in the first step, for annotation by the oracle. Second, we note that the target samples which differ from the source domain are difficult to classify correctly. Thus, to facilitate domain adaptation, those target samples which are farther away from the source samples and are likely to be incorrectly labeled should be given greater weight. The intuition is that instances for which the classifier is uncertain (small margin) provide the most information for learning. Hence we propose weighting the margin term for difficult samples without explicitly using their predicted labels, which may be incorrect. ADA-DWUS empirically reaches state-of-the-art performance on active DA benchmarks.

We compare the proposed method ADA-DWUS with CLUE [22] and observe that uncertainty as estimated by the classifier suffers from large variance, leading to unreliable queries and thus uninformative instances. We address this problem using CLUE with a loop threshold as a simple solution: in the initial stage, we query target samples from constant-weighted clustering; samples that exceed the threshold are additionally weighted according to their uncertainty. In active DA, this produces sampling instances from regions of the feature space that are well-aligned across domains in the early stage. After a given number of iterations, we query the target samples from uncertainty-weighted clusters to ensure that the target samples are not drawn from well-aligned feature space across domains. We conduct empirical experiments to determine a threshold that yields the desired value, with results that attest the usefulness of our study and our proposed approach.

In summary:

- We propose a loop threshold method to improve uncertainty-weighted embedding clustering.
- We show the importance of the uncertainty quality, propose an ADA framework, and compare different active learning methods. Our code is available at https://github.com/HUTTON9453/Active-DA.
- We demonstrate the effectiveness of our method on three datasets: SVHN→MNIST, DomainNet, and Office-31.
- We benchmark the performance of state-of-the-art active learning methods on three different domain shifts, and find that uncertainty-based methods are unreliable in the early stages for active DA.

II. Related Work

A. Notation and Problem Setup

Here we introduce the setup and notation used throughout the paper. We consider C-way classification, in which X and Y represent random variables for features and labels respectively, where $Y = \{0, \ldots, C-1\}$. In active domain adaptation, we have a labeled source domain $L_S = \{(X_S, Y_S)\}$, an unlabeled target domain $U_T = \{(X_{\mathrm{UT}}, Y_{rmUT})\}$ and a labeled target domain $L_T = \{(X_{\mathrm{LT}}, Y_{rmLT})\}$ of size B which is much smaller than the size of unlabeled target domain. The goal is to generalize a model trained on the source domain using a selection of labels from the target domain. The task is to learn a function $h : X \to Y$ (a convolutional neural network (CNN) parameterized by θ) that achieves good prediction performance on the target. The probability of class label y according the model weights θ is denoted by $p(y \mid x; \theta)$.

Classical machine learning techniques showcase the incorporation of active learning for the DA scenario. Chattopadhyay et al. [25] use importance weighting and select target samples with larger distances between features while training using MMD. Su et al. [23] propose AADA, an active learning method that uses H-divergence and importance sampling to query target instances. In AADA, a domain discriminator G_d is learned to distinguish between source and target features obtained from an extractor G_f, in addition to a task classifier G_y. For active sampling, points are scored via the following importance weighting-based acquisition function (H denotes model entropy): $s(x) = \frac{1-G_d(G_f(x))}{G_d^*(G_f(x))} H(G_y(G_f(x)))$; the top B instances are selected for labeling. However, their criteria do not include the diversity of the overall data. Prabhu et al. [22] instead propose ADA-CLUE, a state-of-the-art uncertainty-weighted clustering method that identifies target instances for labeling that are both uncertain under the model and diverse in feature space. The uncertainty term is measured by the predicted entropy under the model.

We propose active domain adaptation via density weighted uncertainty sampling (ADA-DWUS), a variant of CLUE that calculates the target samples as the centroid of the each uniform cluster and then weights the margin corresponding to the target sample to select informative target samples. In comparison with CLUE, ADA-DWUS yields state-of-the-art performance. However, we observe that early-stage model miscalibration causes the uncertainty term to be uninformative. Therefore, we propose improving ADA-CLUE using a loop threshold to retain diversity in the early stage and then combine uncertainty with diversity over the iteration threshold; we show that this outperforms prior work on diverse shifts across multiple learning strategies.

III. Main Approach

Here we introduce the overall framework of active domain adaptation, after which we analyze CLUE and compare it with variants including the proposed ADA-DWUS. Finally, we describe how to use the loop threshold method with CLUE.

a) Active Domain Adaptation Framework: Although active learning and domain adaptation have been well-studied individually, active domain adaptation presents new challenges. It is difficult to determine which selection criterion under domain shift is the most sample-efficient. Another difficulty is how exactly to perform adaptation given these labeled data from the target domain.

The proposed approach queries the most informative samples from this unlabeled pool during the DA process at regular intervals. These queried samples and their labels provided by the oracle are the labeled target data for semi-superviused DA. Our package follows this framework to analyze the state-of-the-art CLUE.

b) CLUE with Loop Threshold: Prior work on active learning identifies instances based primarily on uncertainty and diversity. Here we revisit uncertainty and diversity in the CLUE setting.

c) Uncertainty in CLUE: It is essential to identify instances that provide the model with new information. CLUE uses predictive entropy to measure the corresponding uncertainty of the informativeness. For C-way classification, entropy is defined as

$$H(Y|x) = -\sum_{c=1}^{C} p(c \mid x;\theta) \ln p(c \mid x;\theta). \tag{1}$$

d) Diversity in CLUE: A parallel line of work in active learning involves sampling instances that are representative of the unlabeled pool of data. CLUE clusters deep embeddings of target data points weighted by the corresponding uncertainty of the target model and selects the nearest neighbors of the inferred cluster centroids for labeling. First, CLUE selects an initial pool of diverse unlabeled target samples from the dataset U_T with weighted k-means++ [24]. The goal is to group target instances that are similar in the CNN feature space based on their informativeness ($H(Y|x)$) and to measure the density of a given data point x. Let $\{\mu_1, \mu_2, \ldots, \mu_B\}$ denote the corresponding centroid of each set. CLUE acquires labels for the nearest neighbors of each of the B centroids:

$$X_{\text{LT}} = \{\text{NN}(\mu_b); b = 1, 2, \ldots, B\}. \tag{2}$$

The problem here is that when learning from scratch, model uncertainty may be unreliable, leading to the sampling of less-informative points. We use variants of CLUE to demonstrate this problem in the following section.

e) Active Domain Adaptation via Density Weighted Uncertainty Sampling (ADA-DWUS): We motivate the sample selection criteria using the idea of density-weighted uncertainty sampling (DWUS) [26], which assumes that data points lying on the classification boundary are informative, and that higher-density samples lie close to the decision boundary. DWUS uses the following active selection criterion:

$$\underset{i \in I_u}{\arg\max} \quad E[(\hat{y}_i - y_i)^2 | x_i] p(x_i), \tag{3}$$

where $E[(\hat{y}_i - y_i)^2 | x_i]$ and $p(x_i)$ are the expected error and density of a given data point x_i, respectively, and I_u is the index for the unlabeled data. This formulation indicates that those points which have the largest contribution to the current classification error and lie close to the decision boundary are more informative. The expected error is the uncertainty-based selection criterion. Uncertainty is combined with the density of the underlying data to increase the diversity.

Unfortunately, applying this intuition to come up with a sample selection strategy is non-trivial, because the target data is mostly unlabeled and the empirical risk cannot be computed before annotation. We thus take advantage of the margin: To measure the uncertainty using the margin, the uncertainty criterion $Q_u(x)$ for each target sample x is defined as

$$Q_u(x) = 1 - (\max_i \hat{y}_i - \max_{j | j \neq \underset{k \in I_u}{\arg\max} \hat{y}_k} \hat{y}_j). \tag{4}$$

We use the negative margin since a smaller margin corresponds to higher uncertainty. We use the margin of the highest and second-highest probabilities in the predicted class distribution $\hat{y}$. The margin is preferred since it integrates the second-most probable class label in the uncertainty metric and thus reduces the error rate by defining a decision boundary.

Next, we measure the density of a given data point x based on feature space coverage. We select an initial pool of diverse unlabeled target samples from the dataset U_T with k-means++. In particular, given the unlabeled target data U_T, we initially group all the samples into $N_{\text{cluster}} = C$ clusters, where C is the number of classes. The goal is to group target instances that are similar in the CNN feature space and measure the density of a given data point x. Let $\phi(x)$ denote the feature embeddings extracted from the model and let $\{\mu_1, \mu_2, \ldots, \mu_C\}$ denote the corresponding centroid of each set. We use the L2 distance $\sigma(x)$ from the corresponding centroid to measure $p(x)$ as

$$\sigma(x) = ||\phi(x) - \mu_k||^2. \tag{5}$$

To jointly capture both diversity and uncertainty, the overall objective is

$$\underset{i \in I_u}{\arg\min} \quad Q_u(x_i) * \sigma(x_i). \tag{6}$$

Two components in this measure are interpreted as follows: diversity cue $\sigma(x_i)$ and uncertainty cue $Q_u(x_i)$. The diversity cue allows us to select unlabeled target data instances which are representative, while the uncertainty cue suggests data which the model cannot predict confidently and which is close to the decision boundary.

TABLE I

Test accuracy with MME on Office31 and DomainNet. Office31: 10 rounds with $B = 30$. Clipart (C)→Sketch (S): 10 rounds with $B = 500$. Best performance in bold.

AL method	**Clipart→Sketch**			**DSLR→Amazon**		
	1k	2k	5k	30	60	150
CLUE	46.56	**49.01**	**51.21**	58.23	**62.38**	**71.39**
Constant-weighted clustering	**46.95**	48.03	49.18	**59.13**	60.12	68.56
DWUS	46.91	48.54	51.08	58.67	61.42	71.31

A. Comparison to CLUE variants

In Table I we compare the three conditions. We observe that constant-weighted cluster sampling outperforms uncertainty-weighted CLUE in the early stages, because at this point the model is not yet well-aligned across the source and target domains, and thus evaluations of the uncertainty for the target instances are unreliable. In contrast, after a few iterations, the model becomes more reliable under the weak distribution shift, at which point we add the uncertainty term to capture informative target instances as opposed to redundant instances. To ensure diversity, after clustering, we select representative instances (i.e., non-outliers).

B. CLUE with Loop Threshold

Let t be a threshold value and U be a weighted-uncertainty matrix. We consider an weight matrix W based on the loop threshold:

$$W = \begin{cases} U & loop_{\mathrm{cur}} > t \\ constant & \text{otherwise.} \end{cases} \tag{7}$$

The core idea is to increase the credibility of the uncertainty so that we can identify informative instances for labeling. In general, we select t as the loop threshold when the empirical performance of constant-weighted clustering sampling is better than that in CLUE. For example, increasing t means more chances to query uninformative instances that are already well-aligned across domains; correspondingly, we expect diversity to play a bigger role. Similarly, at lower values of t we expect uncertainty to have greater influence.

After querying labels data for U_T, we proceed to the next step of active domain adaptation: we train the model with labeled source and target data and unlabeled target data. We conduct experiments with three methods: 1) Fine-tuning the model on L_t; 2) Adversarial training via DANN [27] using $(L_S \cup L_T, U_T)$; and 3) Semi-supervised domain adaptation via minimax entropy [28] using $(L_S \cup L_T, U_T)$. We compare each query strategy in the different domain adaptation methods.

IV. Experiments

For all experiments, we followed the standard flow of active domain adaptation, with multiple rounds of batch active sampling, label acquisition, and model updates. As our performance metric, we computed model accuracy on the target test split. We used a ResNet34 CNN and performed 10 rounds of active DA with a per-round budget of 500 instances (for a total of 5000 labels) for DomainNet and a budget of 30 instances (300 labels total) for Office31. In addition, we evaluated the performance on the SVHN → MNIST shift. We used a modified LeNet architecture, and performed 30 rounds of active adaptation with a per-round budget of 10. We evaluated the results on the standard benchmark datasets: DomainNet [29], DIGITS (SVHN [30] as the source domain and MNIST [31] as the target domain), and Office31 [4].

We compared with CLUE to show that the loop threshold yields improved performance. (1) CLUE [22] is a state-of-the-art active DA method for entropy-weighted clustering that selects diverse, informative target instances for labeling from dense regions of the feature space. (2) ADA-DWUS is our proposed variant of CLUE which clusters deep embeddings of target instances by uniform weighting and then selects the nearest neighbors of the inferred cluster centroids weighted by the corresponding uncertainty for labeling. By comparing these, we propose (3) CLUE with a loop threshold, which introduces threshold t to control when to perform uncertainty clustering. We additionally compared with an active domain adaptation method: (4) AADA [23] is a hybrid active learning method for domain adaptation that uses a combination of the entropy measure from a classifier and the outputs of a domain discriminator. Target instances are selected based on the predictive entropy and targetness measured by an adversarial domain discriminator followed by adversarial domain adaptation via DANN. Furthermore, we compared the proposed approach with traditional active learning methods. Uncertainty-based sampling [14] includes (5) Entropy sampling, which selects samples with the highest predictive entropy, and (6) Margin sampling, which selects samples for which the score difference between the top-2 predictions is the smallest. We experimentally illustrate that traditional active learning methods which are based purely on uncertainty are ineffective because the uncertainty term is unreliable under domain shift. In terms of diversity-based sampling, we used (7) Core-set [17], which uses greedy k-center clustering to select samples from unlabeled data such that the largest distance between the remaining unlabeled data and labeled data is minimized. We show that this solely diversity-based method generalizes poorly under domain shift. Hence we also considered methods that combine uncertainty and diversity. (8) BADGE [15] is a hybrid deep active learning method that optimizes both uncertainty and diversity. The main difference between CLUE and BADGE is the embeddings for clustering. BADGE clusters gradient embeddings via KMeans++, whereas CLUE clusters penultimate-layer embeddings via KMeans++. Finally, (9) Random sampling selects samples uniformly at random. We take this as our baseline; this allows us to compare the benefit of active learning over passive learning.

We evaluated all query methods across three DA paradigms: (1) Fine-tuning from source: we trained the model using L_S and then fine-tuned it on L_T, both in a supervised way. (2) MME from source: Minimax entropy (MME) [32] is a state-of-the-art semi-supervised DA method that starts from a model trained on L_S that minimizes adversarial entropy loss for unsupervised domain alignment in addition to finetuning

TABLE II

TEST ACCURACY WITH MME ON OFFICE31 AND DOMAINNET. OFFICE31: 10 ROUNDS WITH $B = 30$. CLIPART (C)→SKETCH (S): 10 ROUNDS WITH $B = 500$. BEST PERFORMANCE IN BOLD.

DA	Uncertainty	**Clipart→Sketch**			**DSLR→Amazon**		
method	weighting	1k	2k	5k	30	60	150
MME from source	Entropy	46.56	**49.05**	**51.21**	55.23	**62.38**	**71.39**
	Margin	46.49	48.82	50.95	54.12	61.82	70.50
	Random	44.12	46.69	47.85	53.05	58.24	63.12
	Constant	**46.95**	48.03	49.18	**56.13**	60.12	68.56

on $L_S \cup L_T$. (3) DANN from source: DANN [27] is an adversarial method. We trained the classifier using $(L_S \cup L_T, U_T)$. Regardless of the DA paradigm, we show that the proposed method selects informative instances.

In summary, the proposed method addresses two questions: how to select images to label from U_T to yield the greatest performance gain, and how to train a classifier given $\{L_S, L_T, U_T\}$. We conduct the following experiments to answer these questions.

V. RESULTS AND DISCUSSION

a) Comparison of Sampling Methods: We observe that CLUE with the loop threshold consistently outperforms the other methods. Pure uncertainty (margin, entropy) and diversity (core-set) methods work well with easy cases (Real→ Clipart and SVHN→MNIST) (Table IV), but all are outperformed by random sampling. In contrast, the proposed method and other hybrid approaches perform favorably against these. Overall, our method consistently performs best. Averaged over four DomainNet scenarios (Table III), CLUE with the loop threshold outperforms margin-based uncertainty sampling and core-set diversity sampling at $B = 2k$ by 1.4% and 2.7% with fine-tuning, and 2.1% and 3.0% with MME adaptation.

However, across learning strategies, shifts, benchmarks, and most rounds, DWUS's performance is within 1% of CLUE's performance. These two methods use the same clustering method (K-means++) and are hybrid methods that combine uncertainty and diversity sampling. CLUE clusters deep embeddings of target instances weighted by the corresponding predictive entropy of the target model and then selects the nearest neighbors of the inferred cluster centroids for labeling. However, DWUS clusters deep embeddings and then weights these by the corresponding margin of the target model. Clearly, the main difference between DWUS and CLUE is the uncertainty term and whether clustering is weighted. We discuss this below.

Table II shows that constant weighting is better than uncertainty weighting at the early stage. This shows that the uncertainty measure is uninformative at the early stage.

b) Comparison of DA paradigms: We now evaluate our method's compatibility with a few additional domain adaptation strategies from the literature. Across AL methods, we observe that MME adaptation consistently outperforms finetuning by 2.3%–4.4% accuracy on DomainNet.

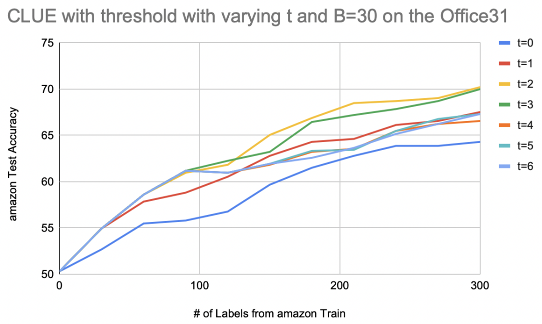

Fig. 1. Measuring sensitivity to t of CLUE with loop threshold on Office31

c) Sensitivity to threshold t: In Figure 1, we conduct a sweep over the threshold values for CLUE with the loop threshold on Office31. As seen, CLUE with the loop threshold improves performance, particularly at later rounds when the number of labels from target data is about 3%. For example, in Figure 1, the best performance is obtained when the total number of instances of the target is 1886 and total number of labeled target instances via diversity sampling is 60. Thus, the ratio of the labeled target instances with diversity to the total number of target data instances is 3.2%. We select $t * B/(\text{total number of target data instances}) \approx 3\%$ as the default value.

VI. CONCLUSION AND FUTURE WORK

We propose active domain adaptation via density weighted uncertainty sampling (ADA-DWUS), a unified framework for active domain adaptation. When few labeled targets are available, the domain adaptation model improves classification while combining uncertainty and diversity to select the most informative samples from the target domain. We also propose CLUE with a loop threshold to account for miscalibration of the model uncertainty in the early stage. We show that our methods outperform the state-of-the art method for active DA.

ACKNOWLEDGMENTS

This work roots from the M.S. thesis of the first author [33]. We thank Profs. Yun-Nun Chen and Hung-Yi Lee for their valuable feedback. The work is partially supported by the Ministry of Science and Technology of Taiwan via the grants MOST 110-2628-E-002-013 and 111-2628-E-002-018. We also thank the National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational resources.

REFERENCES

[1] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," *CoRR*, vol. abs/1310.1531, 2013.

[2] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *CoRR*, vol. abs/1411.1792, 2014.

[3] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schölkopf, *Covariate shift and local learning by distribution matching*. MIT Press, 2009, pp. 131–160.

[4] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *ECCV*, 2010.

TABLE III

Evaluation on DomainNet for active DA and active learning. We perform 10 rounds of active DA with $B = 500$ and $t = 3$ and show the accuracy with three budgets under four scenarios: Real (R)→ Clipart (C), Clipart→Sketch (S), Sketch→Painting (P), and Clipart→Quickdraw (Q), as well as the average (AVG). We compare our method against state-of-the-art methods for active learning and active DA. We use three DA methods. Best performance in bold.

DA method	AL method	**R→C** (easy)			**C→S** (moderate)			**S→P** (hard)			**C→Q** (very hard)			**AVG**		
		1k	2k	5k	1k	2k	5k	1k	2k	5k	1k	2k	5k	1k	2k	5k
Fine-tuning from source	Random	52.1	54.2	60.2	41.7	43.9	46.8	42.1	44.2	46.8	22.5	28.8	35.5	39.6	42.8	47.3
	Entropy	48.2	51.7	58.5	40.8	42.7	45.2	41.8	43.8	46.5	21.0	25.9	34.7	38.0	41.0	46.2
	Margin	51.0	55.2	60.5	41.8	43.5	45.4	42.5	44.3	46.8	23.5	28.2	35.4	39.7	42.8	47.0
	Core-set	49.5	54.5	59.3	41.2	42.5	44.9	40.5	42.9	44.1	22.1	25.9	31.1	38.3	41.5	44.9
	BADGE	52.1	54.2	60.9	42.1	44.8	47.2	42.5	44.7	47.2	23.2	28.3	34.8	40.0	43.0	47.5
	CLUE	52.6	55.8	61.8	42.0	45.2	47.8	42.8	45.1	47.5	23.5	28.9	35.6	40.2	43.8	48.2
	DWUS	53.0	56.0	61.5	**42.5**	45.2	47.3	43.0	45.3	47.0	24.1	28.5	35.5	40.7	43.8	47.8
	CLUE with loop threshold (Ours)	**53.1**	**56.2**	**62.2**	**42.5**	**45.8**	**48.1**	**43.2**	**45.5**	**47.6**	**24.3**	**29.2**	**35.9**	**40.8**	**44.2**	**48.5**
MME from source	Random	55.1	59.5	63.8	45.5	48.2	49.5	42.8	45.2	47.9	24.5	30.1	38.3	42.0	45.8	49.9
	Entropy	53.5	58.5	64.2	44.8	45.8	48.5	41.5	44.1	47.1	21.8	24.8	32.9	40.4	43.3	48.6
	Margin	55.7	60.5	**65.9**	46.2	48.2	49.3	43.8	45.5	48.1	23.1	28.2	37.8	42.2	45.6	50.3
	Core-set	54.1	59.2	64.8	45.5	47.0	49.2	42.7	44.5	47.3	24.0	27.9	34.2	41.6	44.7	48.9
	BADGE	56.2	60.5	65.3	45.9	49.1	50.9	43.3	45.9	48.5	24.8	29.2	38.5	42.6	46.2	50.8
	CLUE	56.5	60.7	65.7	46.5	49.8	51.4	43.7	46.4	49.4	25.5	30.8	39.0	43.1	47.0	51.4
	DWUS	56.6	60.5	65.5	46.0	49.1	50.5	43.9	46.3	48.7	25.3	31.1	38.4	43.0	47.0	51.0
	CLUE with loop threshold (Ours)	**56.8**	**61.5**	**66.8**	**46.8**	**50.8**	**52.2**	**44.1**	**46.8**	**49.9**	**25.6**	**31.5**	**39.1**	**43.3**	**47.7**	**52.0**
DANN from source	AADA	53.1	57.5	62.5	44.5	46.5	49.0	41.5	42.8	45.9	22.8	25.5	31.2	40.5	43.1	47.2
	CLUE	55.2	58.4	64.0	45.0	48.2	50.4	43.4	45.5	48.3	24.6	28.8	35.6	42.1	45.2	49.6
	DWUS	55.4	58.5	63.8	**45.2**	48.6	50.4	43.1	45.1	48.0	**25.1**	28.9	35.3	42.2	45.3	49.4
	CLUE with loop threshold (Ours)	**55.7**	**59.2**	**64.2**	**45.2**	**48.9**	**50.7**	**43.8**	**46.0**	**48.7**	**25.1**	**29.0**	**35.8**	**42.5**	**45.8**	**49.9**

TABLE IV

Office and digits results. Digits: 30 rounds with $B = 10$ and $t = 1$; Office31: 10 rounds with $B = 30$ and $t = 2$; Best performance in bold.

DA method	AL method	**SVHN→MNIST**			**DSLR→Amazon**		
		30	60	150	30	60	150
Fine-tuning from source	Random	79.2	87.8	95.1	51.8	55.4	66.4
	Entropy	65.4	74.5	91.8	51.2	53.5	59.1
	Margin	84.5	90.5	95.4	52.4	56.5	65.5
	Core-set	72.9	77.2	88.4	52.9	55.7	66.8
	BADGE	80.5	88.9	95.3	55.4	59.1	70.8
	CLUE	85.6	89.5	94.8	56.6	60.8	70.5
	DWUS	85.8	89.5	**95.5**	56.8	61.0	71.6
	CLUE-loop (Ours)	**86.1**	**89.8**	**95.5**	**57.0**	**61.5**	**72.9**
MME from source	Random	85.1	92.1	95.2	56.9	58.5	69.8
	Entropy	80.9	85.5	92.4	55.7	57.2	62.1
	Margin	87.1	92.4	96.1	55.9	59.1	70.7
	Core-set	85.0	89.5	94.2	56.4	59.2	70.5
	BADGE	89.6	92.6	**96.5**	57.5	62.6	71.1
	CLUE	91.2	93.6	95.9	59.5	63.2	71.5
	DWUS	**91.7**	93.9	95.5	60.1	63.3	70.9
	CLUE-loop (Ours)	**91.7**	**94.2**	**96.5**	**61.2**	**64.6**	**72.3**
DANN from source	AADA	88.6	90.5	95.1	53.1	56.0	63.2
	CLUE	90.5	92.5	94.8	59.2	63.1	71.3
	DWUS	91.0	92.8	95.3	59.5	62.5	69.5
	CLUE-loop (Ours)	**91.2**	**93.1**	**95.4**	**60.1**	**63.5**	**72.4**

[5] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR*, 2011, pp. 1521–1528.

[6] S. Sankaranarayanan, Y. Balaji, A. Jain, S. Lim, and R. Chellappa, "Unsupervised domain adaptation for semantic segmentation with gans," *CoRR*, vol. abs/1711.06969, 2017.

[7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," *CoRR*, vol. abs/1604.01685, 2016.

[8] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. V. Gool, "Domain adaptive faster R-CNN for object detection in the wild," *CoRR*, vol. abs/1803.03243, 2018.

[9] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *CVPR*, 2018.

[10] W.-N. Hsu and H.-T. Lin, "Active learning by learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[11] M. Ducoffe and F. Precioso, "Adversarial active learning for deep networks: a margin based approach," *CoRR*, vol. abs/1802.09841, 2018.

[12] D. Roth and K. Small, "Margin-based active learning for structured output spaces," in *ECML*, ser. Lecture Notes in Computer Science, vol. 4212. Springer, 2006, pp. 413–424.

[13] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," *CoRR*, vol. abs/1703.02910, 2017.

[14] D. Wang and Y. Shang, "A new active labeling method for deep learning," in *IJCNN*, 2014, pp. 112–119.

[15] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal, "Deep batch active learning by diverse, uncertain gradient lower bounds," *CoRR*, vol. abs/1906.03671, 2019.

[16] A. Kirsch, J. van Amersfoort, and Y. Gal, "Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning," *CoRR*, vol. abs/1906.08158, 2019.

[17] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," 2018.

[18] M. Prince, "Does active learning work? a review of the research," *Journal of Engineering Education*, vol. 93, pp. 223–231, 2004.

[19] S. Bickel, M. Brückner, and T. Scheffer, "Discriminative learning under covariate shift," *JMLR*, vol. 10, no. 75, pp. 2137–2155, 2009.

[20] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," *CoRR*, vol. abs/1706.04599, 2017.

[21] P. Rai, A. Saha, H. Daumé, and S. Venkatasubramanian, "Domain adaptation meets active learning," in *ALNLP*, 2010.

[22] V. Prabhu, A. Chandrasekaran, K. Saenko, and J. Hoffman, "Active domain adaptation via clustering uncertainty-weighted embeddings," *CoRR*, 2020.

[23] J. Su, Y. Tsai, K. Sohn, B. Liu, S. Maji, and M. Chandraker, "Active adversarial domain adaptation," *CoRR*, vol. abs/1904.07848, 2019.

[24] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *SODA*, 2007.

[25] R. Chattopadhyay, W. Fan, I. Davidson, S. Panchanathan, and J. Ye, "Joint transfer and batch-mode active learning," in *ICML*, 2013.

[26] P. Donmez, J. G. Carbonell, and P. N. Bennett, "Dual strategy active learning," in *ECML*, 2007.

[27] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," 2016.

[28] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization." in *CAP*, F. Denis, Ed. PUG, 2005, pp. 281–296.

[29] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," *CoRR*, vol. abs/1812.01754, 2018.

[30] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," 2011.

[31] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[32] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko, "Semi-supervised domain adaptation via minimax entropy," *CoRR*, vol. abs/1904.06487, 2019.

[33] S.-F. Wu, "Improving clustering uncertainty-weighted embeddings for active domain adaptation," Master's thesis, National Taiwan Univeristy, 2021.