https://doi.org/10.2352/EI.2023.35.9.IPAS-288

Evaluating Active Learning for Blind Imbalanced Domains

Hiroshi Kuwajima; DENSO CORPORATION, Tokyo, Japan
Masayuki Tanaka, Masatoshi Okutomi; Tokyo Institute of Technology, Tokyo, Japan

Abstract

Deep learning, which has been very successful in recent years, requires a large amount of data. Active learning has been widely studied and used for decades to reduce annotation costs and now attracts lots of attention in deep learning. Many real-world deep learning applications use active learning to select the informative data to be annotated. In this paper, we first investigate laboratory settings for active learning. We show significant gaps between the results from different laboratory settings and describe our practical laboratory setting that reasonably reflects the active learning use cases in real-world applications. Then, we introduce a problem setting of blind imbalanced domains. Any data set includes multiple domains, e.g., individuals in handwritten character recognition with different social attributes. Major domains have many samples, and minor domains have few samples in the training set. However, we must accurately infer both major and minor domains in the test phase. We experimentally compare different methods of active learning for blind imbalanced domains in our practical laboratory setting. We show that a simple active learning method using softmax margin and a model training method using distance-based sampling with center loss, both working in the deep feature space, perform well.

Introduction

Deep learning [1] techniques are rapidly advanced recently and becoming necessary components for widespread systems. The performance of deep learning techniques is obtained with complex (”deep”) model structures with massive parameters, along with computational resources to enable optimization of such parameters. On the other hand, it also requires a large amount of data, and efficiently collecting and annotating data is one of the most important issues in deep learning. Annotation, which involves human workers, is time-consuming and costly. Thus we cannot annotate all collected samples because the budget is always limited. Active learning is a technical solution having been studied before the advent of deep learning to annotate data efficiently [2, 3, 4, 5, 6, 7]. It selects the most informative samples from collected data, and humans annotate them. Then, we use the cumulatively annotated data to train machine learning models to improve performance with a minimal annotation cost. Active learning has already been studied in a wide variety of applications such as automated driving [8, 9], medical imaging [10], medical diagnosis, microbiology, and manufacturing [4]. Experimental settings in such active learning studies (”laboratory setting”) and active learning use cases in real-world applications cannot be identical. We must carefully set up active learning experiments that reasonably reflect the active learning use cases in real-world applications. In this work, we focus on 1) investigating practical active learning experimental settings and 2) the problem of blind imbalanced domains in active learning.

Figure 1 illustrates the actual situation in real-world applications and the laboratory setting in academic studies. One of the essential points of laboratory settings is to create a huge and realistic data pool. In the actual situation, we collect massive data samples based on true data distribution, as shown at the top of Figure 1. On the other hand, we conduct laboratory experiments with the data generation process illustrated at the bottom of Figure 1. The size of collected data samples in the actual situation is much larger [11, 12] than standard experimental data sets [13, 14, 15, 16]. Collecting new data samples only for active learning experiments is not realistic. Therefore, we cannot evaluate active learning algorithms in the actual situation and must carefully design appropriate laboratory settings. Such laboratory settings are essential for experimental comparisons.

Any data set includes multiple domains, *e.g.*, different individuals in handwritten character recognition with different social attributes such as communities, ages, genders, etc. We refer to major and minor domains as the domains associated with many data samples and few data samples. Domains are imbalanced if data has major and minor domains and blind if the domain assignment of data samples is unknown. In many real-world applications, multiple domains are blind and imbalanced. The trained machine learning models usually perform best on the major domains, because the major domains have dominant samples in data pools and training data sets. However, particularly for safety-critical applications, the performance on minor domains is critical. For example, accidents, *i.e.*, minor domains, in automated driving systems and credit authorization systems have a small number of samples but bring grave consequences. Therefore, we need to improve the performance on the minor domains while maintaining that on the major domains.

Our contributions to this paper are twofold.

- We investigate laboratory settings for active learning and empirically show that laboratory settings greatly impact experimental results.
- We evaluate methods of active learning for blind imbalanced domains in our practical laboratory setting.

This paper is organized as follows. *Related works* section visits related works in active learning and machine learning with blind imbalance domains. *Laboratory settings for active learning* section investigates laboratory settings for active learning. *Active learning for blind imbalanced domains* section elaborates on active learning for blind imbalanced domains. Then, *Experiments* section shows the concrete settings we used in our experiments, the experimental results of active learning in different laboratory settings, and the experimental result of different active learning approaches to blind imbalanced domains. Finally, *Conclusion* section concludes our research in this paper.

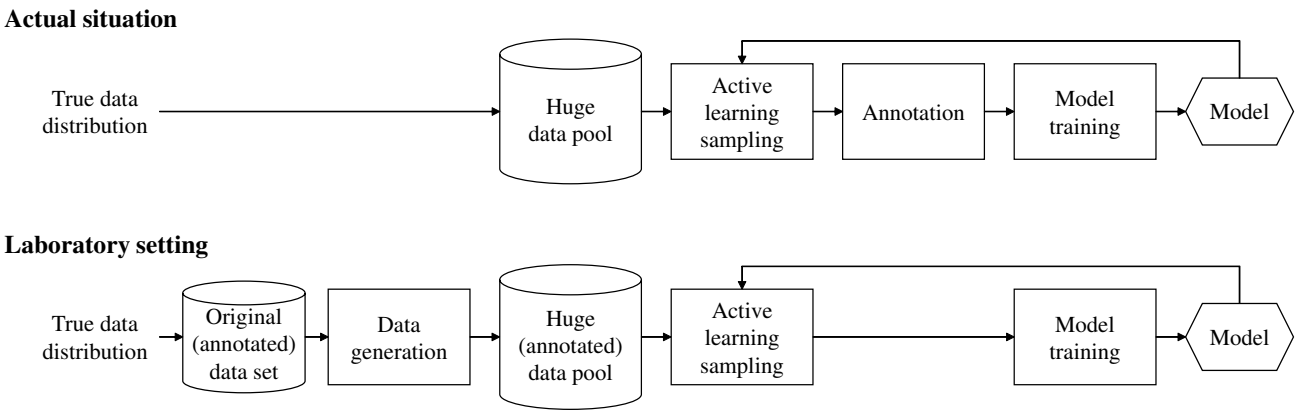


Figure 1: Laboratory setting for active learning.

Related works

Active learning is a set of techniques to select data samples from a data pool for annotation. Active learning tries to select the most informative samples, *i.e.*, the samples the trained model is not expected to know. We usually believe that the data samples with uncertain inference results are informative. Furthermore, the acquisition mode of active learning is important, especially for deep learning.

In the context of classification, the softmax value is the most convenient approach to evaluate inference uncertainty [17]. Softmax values represent the estimated categorical probabilities by applying the softmax function to the output logits from neural networks. As proxies of inference uncertainty, we have variations using softmax values such as maximum softmax value, *e.g.*, the probability of the most probable class, and the margin between the two most significant softmax values. Other tasks such as detection (bounding box regression) and segmentation can have other forms of uncertainty representation different from softmax values. A Bayesian neural network [18, 19, 20] is an approach to directly quantifying the uncertainty of inference results in the context of Bayesian inference [21]. BALD [22] is a method to use Bayesian neural networks for active learning.

The time complexity of deep learning training is very high, and we want to reduce the total count of training attempts in active learning. There are batch and sequential methods in active learning [23]. Sequential methods require one-by-one data acquisition and model training for each single data point acquired. On the other hand, batch active learning selects several samples at once. It runs model training for a batch of data points acquired. As a result, the total count of training attempts to reach a specific number of acquired samples is high in sequential active learning and low in batch active learning. Therefore, although batch active learning is not optimal, it is convenient for deep learning in terms of time [23, 24]. BatchBALD [25] is a batch version of BALD [22] which ensures the independence of the samples in an acquisition batch.

Latest active learning studies incorporate deep feature and model-specific extensions. As mentioned above, active learning is used along with deep learning whose important property is its deep feature space. Active learning can leverage such deep feature space to acquire high-dimensional data, *e.g.* images and videos [26]. Deep learning introduced new problem settings with high-dimensional labels. Active learning can acquire partial labels, *e.g.* specific regions in pixel labels of semantic segmentation to avoid annotating entire high-dimensional labels [27].

One of the latest research considers data augmentation in active learning. Data augmentation is a set of techniques for model training that increases the amount of training data by transforming existing data points [28, 29, 30, 31]. Both unlabeled data instances in data pools and their augmented data instances can be used for active learning to integrate active learning and data augmentation [32]. However, most of the latest active learning studies [27, 26] do not incorporate such training considerations yet.

Machine learning is good at statistically capturing the overall characteristics of the entire training data. However, training data sets generally include multiple domains in real-world machine learning applications, and some domains have higher importance or risks. Machine learning with blind imbalanced domains addressed such a problem by evaluating the performance of each domain in the test data under the condition of unknown and imbalanced domain assignment of training samples [33]. Although previous work focused only on model training, blind imbalanced domains apply to active learning, too.

Laboratory settings for active learning

One of the essential points of laboratory settings is how to create realistic huge annotated data pools, as shown in Figure 1. Another point is considering realistic model training configurations even in active learning, including training data augmentation and validation set size. We review general existing laboratory settings and describe our practical laboratory setting.

We consider the experimental settings of recent active learning studies [25, 32, 26, 27]. The first point is about the size of data pools. It is assumed that the size of data pools is huge in active learning. However, original annotated data sets are usually not large enough in active learning studies. In some studies, huge annotated data pools are generated by copying samples in the original annotated data sets, adding elementwise Gaussian noises. As a result, the generated data pools have very similar samples. This generation process changes the distribution in data pools from that in the original annotated data sets. Second, an active learning algorithm acquires informative samples from the

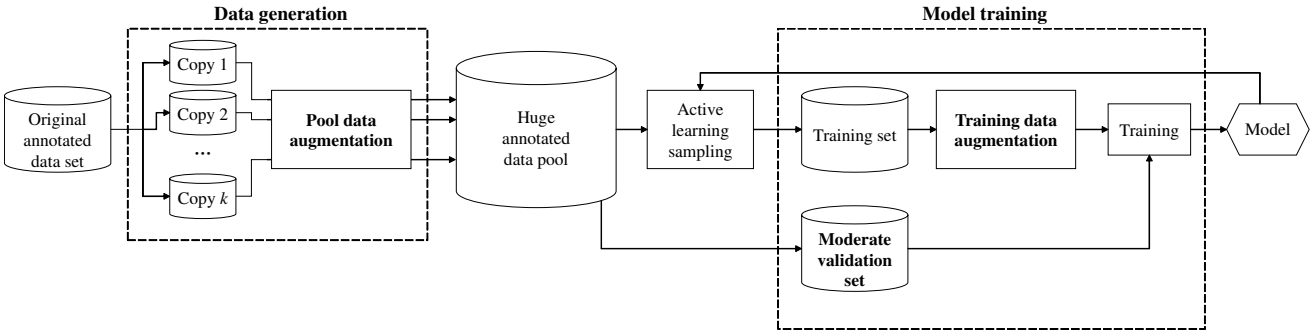


Figure 2: Our practical laboratory setting for active learning. Pool data augmentation in large data generation increases variation in data pools. Validation set size is appropriate, and training data augmentation is used in model training.

huge annotated data pool and puts them into a training set. Then, we train a machine learning model. The training strategy is important even in active learning. Data augmentation has not been used during training in active learning studies. However, LADA addressed that data augmentation is important for active learning [32]. In addition to that, some active learning research used a validation set that is tens of times the training set. In machine learning, validation sets are usually smaller than training data sets. If we have such a large validation set, it is natural to use it as an additional training set.

Next, we investigate our practical laboratory setting, as illustrated in Figure 2, closer to the actual situation in Figure 1. First, we generate huge annotated data pools by copying samples in the original annotated data sets and applying data augmentation. As a result, the generated data pools have a wide variety of samples, which is a natural condition for real-world applications. Second, active learning algorithms select samples from the generated data pools and put them into training sets. Then, we train a machine learning model. We use data augmentation during the model training, following basic machine learning practices for improving inference performance. The validation sets are similar to or smaller than training sets, following the actual data usage practices in machine learning development. In the rest of this section, we address the three points in Figure 2, pool data augmentation, training data augmentation, and train-validation split.

Pool data augmentation

Huge annotated data pools should have a wide variety of data samples. Just copying the original annotated data sets is not enough. Therefore, in our practical laboratory setting, we apply data augmentation [28, 29, 30, 31] to the copied samples after copying original annotated data sets as k times to make huge annotated data pools, as shown in Figure 2. Augmentation techniques should be realistic, such as random affine transformations [34] and random crops [35, 36]. This data generation process increases variation in data pools. We refer to this data augmentation in the data generation process as pool data augmentation.

Training data augmentation

We can also use data augmentation during the model training even in active learning [32]. We refer to the data augmentation in model training as training data augmentation. Training data augmentation considerably impacts model training, and we usually use it in real-world development. Therefore, in our practical laboratory setting, we apply training data augmentation, as shown in Figure 2.

Train-validation split

We use validation sets for model selection, hyperparameter tuning, early stopping, etc., as a part of model training. The parameters of the models are updated with the training data, while the validation data is not used for the parameter updating. Therefore, we usually split a given data into training and validation so that the training data size is comparable to or larger than that of validation data. In our practical laboratory setting, we use validation sets of similar size or smaller than training sets, as illustrated in Figure 2.

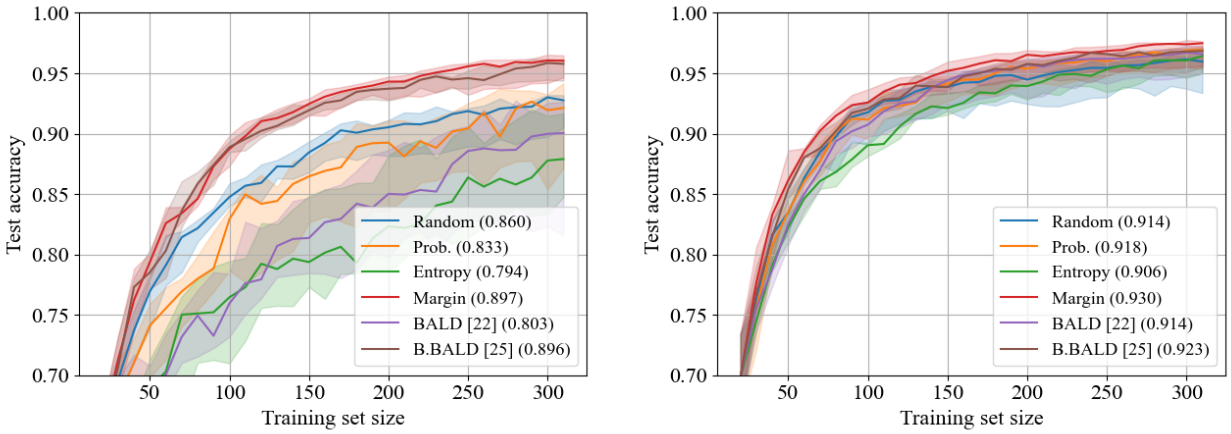
Active learning for blind imbalanced domains

We assume that a data pool contains samples from various domains. In industrial applications, small sample data are sometimes critical. For example, accidents in automated driving and frauds credit authorization are crucial but much smaller than standard samples. Therefore, improving the performance on minor domains is essential while maintaining that on major domains. Here, we refer to the minor domains as the domains associated with the small training samples. The major domains are the domains corresponding to the dominant training samples.

Let a data pool sample, a label, and a domain label of the sample be x, y, and z, respectively. y is unknown in active learning. The joint probability of the data pool sample and the label with multiple domains can be expressed by a mixture distribution:

$$p(x,y) = \sum_{z=0}^{N_z-1} p(z)p(x,y|z)\,, \tag{1}$$

where N_z is the number of domains. We say the non-blind domain if $p(z)$ is known. If $p(z)$ is unknown, then it is a blind domain problem. If the variance of $p(z)$ is small, then the distribution of domains is balanced. We say the imbalanced domains for the large variance of $p(z)$. In simple two-domain cases, *i.e.*, $N_z = 2$, $p(z = 0) \gg p(z = 1)$ is the imbalanced domain problem. If the domains



(a) Example laboratory setting in an existing active learning study [25].

(b) Our practical laboratory setting.

Figure 3: Test accuracy of each active learning algorithm in different laboratory settings. Figures 3a and 3b are the test accuracy in existing and our practical laboratory settings. Bald lines are the medians with lower and upper quartiles in shadow from 12 trials which consist of four trials for three data sets MNIST [13], EMNIST [14], and USPS [15].

are balanced, then $p(z=0) \simeq p(z=1)$. The data pool size should be large to keep the sampling distribution of imbalanced domains in data pools during active learning. To evaluate trained models in active learning for blind imbalanced domains, we use the domain-wise performance on a domain z.

Experiments

In this section, we conduct experiments of 1. different laboratory settings for active learning and 2. active learning for blind imbalanced domains.

We use the following common active learning and model training settings throughout experiments. We run batch active learning in all experiments with an acquisition size of 10. Acquisition size is the number of samples active learning algorithms select in an iteration. We use sequential active learning methods as naive batch methods to take the top samples with the highest acquisition scores from a data pool in an active learning iteration. The initial size of data pools is $300,000$. We start active learning with the initial training sets with 20 randomly selected samples and end with the maximum size of training sets being 320, *i.e.*, the number of active learning iterations is 30, and the maximum number of acquired samples is 300. In model training, we train a Bayesian neural network of LeNet [37] with ReLU [38, 39] on a classification task, handwritten digit recognition. We run MC dropout [40, 41, 42] and compute logit means before the softmax function [17]. We used the dropout probability of 0.5, and the number of inference samples using MC dropout is 10. Test accuracy of the trained models is measured after 40 epochs of training with batch size 128 using sampling with replacement [43].

The learning curve in active learning is the plot of test accuracy against training set size. Throughout experiments, we use a metric Area under Learning Curve (ALC) to evaluate active learning methods [44]. ALC is the square measure under such a learning curve in active learning, which is normalized to have values from 0.0 to 1.0. We conduct four trials with random seeds of 0 to 3 for all experiments because test accuracy in active learning and machine learning has variability.

Comparison of laboratory settings for active learning

We compare our practical laboratory setting with the example laboratory setting of an existing active learning study [25].

In the example laboratory setting of an existing active learning study, huge annotated data pools are generated by copying the original annotated data sets. Pool data augmentation is not used, but elementwise Gaussian noises with $\mu = 0.0$ and $\sigma = 0.1$ are added for avoiding identical samples. Deep learning models are trained without data augmentation. The size of the validation set is $3,072$.

On the other hand, in our practical laboratory setting, we copy the original annotated data sets, followed by pool data augmentation to generate huge annotated data pools. We use the same methods for pool data augmentation and training data augmentation illustrated in Figure 2. Data augmentation methods are random affine with the angle from $-10°$ to $10°$ and random crop. The validation set size is 100.

Results

Figure 3 shows the test accuracy of different active learning methods in the example laboratory setting of an existing study [25] (3a) and in our practical laboratory setting (3b).

In Figure 3a, BALD [22] performs worse than the random baseline. However, Figure 3b shows all active learning methods perform similarly with smaller variance. BALD [22] gets better as the training set size grows in both Figures 3a and 3b, but the curve is steep in Figure 3b. Even BatchBALD [25] performs worse than the random baseline at the beginning of active learning in Figure 3b. Softmax margin, one of the most naive active learning methods, performs best if we scale up Figure 3b. The test accuracy has a smaller variance in Figure 3b than that in Figure 3a.

We observe significant gaps between the results from different laboratory settings. We can conclude that evaluating the active

learning methods in a reasonable setting is important.

Comparison of methods of active learning for blind imbalanced domains

The experiments in *Comparison of laboratory settings for active learning* subsection showed the importance of laboratory settings to compare active learning methods. This section uses our practical laboratory setting in Figure 2 for the following experiments to investigate active learning and model training methods.

We simulate blind imbalanced domains in data pools. Data pools each have one of six pairwise domains in MNIST [13], EMNIST [14], and USPS [15] datasets abbreviated M, E, and U. An example pairwise domain M/E denotes a major domain MNIST with a minor domain EMNIST. We measure domain-wise test accuracy, so we have domain-wise learning curves and ALC scores. We define mean ALC scores as the mean of domain-wise ALC scores. In our experiments, 99% of the data pool consists of the samples from major domains and the rest does those from minor domains, *i.e.*, the minor ratio is 1%. Domain assignment is used only for the initial data pool generation and is blind to active learning algorithms.

Active learning methods

We evaluate six active learning methods, *i.e.*, random acquisition as the baseline, three softmax methods, probability, margin, and entropy, and two Bayesian methods, BALD [22] and BatchBALD [25]. Our softmax methods use 1. probability of the inference class, *i.e.*, the maximum softmax value, 2. margin between the two maximum softmax values, and 3. entropy of all softmax values to acquire samples from data pools.

Training methods

We evaluate three model training methods, random sampling as the baseline, random sampling with center loss, and distance-based sampling with center loss. These training methods were studied in previous work on machine learning with blind imbalanced domains [33]. Center loss and distance-based sampling work in the deep feature space at the F6 layer of LeNet.

Results

Table 1 shows the major, minor, and mean ALC scores of the six active learning and three model training methods introduced above, *i.e.*, 18 combinations. Results are indicated for each major/minor domain setting such as MNIST [13]/EMNIST [14] and the average of all six domain settings in Table 1.

First, we focus on the average ALC scores in the bottom rows of Table 1. We observe that the results in the active learning for blind imbalanced domains, *i.e.*, domain-wise view, look different from those in simple active learning, *i.e.*, the major domain view. For example, the combinations of an active learning method softmax entropy and model training methods with center loss perform better (0.913) than the random baseline (0.911) in the major ALC scores. However, these combinations perform worse (0.869 and 0.871) than the random baseline (0.872) in the mean ALC scores. We can conclude that incorporating blind imbalanced domains is essential for a fair evaluation of active learning algorithms. Based on the mean ALC scores, we can see that the active learning method of softmax margin and the training method of center loss and distance-based sampling outperforms others.

Next, we focus on the ALC scores for each major/minor domain setting in Table 1. On the mean ALC scores, the combination of softmax margin with center loss and distance-based sampling outperforms others in the three domain pairs out of six and performs comparably to the best methods in two of the rest three pairs. The domain pair U/M is the only setting in which the combination of softmax margin with center loss and distance-based sampling has the mean ALC score 0.01 lower than that of the best method. On the minor ALC scores, the combination of softmax margin with center loss and distance-based sampling is the best in two domain pairs out of six. Although one of the latest active learning methods, BatchBALD [25], does not explicitly assume imbalanced domains, its combinations perform better in minor domains for two other domain pairs. BatchBALD [25] combinations have minor ALC scores 0.01 higher than that of the combination of softmax margin with center loss and distance-based sampling in these two domain pairs U/M and U/E, but they are comparable in the rest two pairs, M/U and E/U. The combination of softmax margin with center loss and distance-based sampling consistently outperforms others for major domains except for U/E. However, even for U/E, the method performs second best, and the gap to the best method is the ALC score of 0.003.

In summary, the combination of softmax margin with center loss and distance-based sampling comparably performs in minor domains while maintaining that in the major domains. It achieves the best mean ALC scores for most domain pairs.

Conclusion

This paper introduced different laboratory settings for active learning and showed that active learning methods demonstrate significantly diverse behavior in each setting. We have demonstrated that appropriate laboratory settings for active learning experiments are important to selecting proper methods. Our practical laboratory settings have 1. pool data augmentation in the large data generation, 2. proper validation set size, and 3. training data augmentation to simulate realistic training practices. Then, we introduced a problem setting, active learning for blind imbalanced domains, which is important in specific real-world applications. Finally, we investigated the best active learning methods for blind imbalanced domains under our practical laboratory setting. As a result, active learning with softmax margin and model training with center loss along with distance-based sampling during training works for both major and minor domains on average and in most settings.

References

[1] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[2] Burr Settles. Active learning literature survey. *CS Technical Reports*, 2009.

[3] Burr Settles. Active learning. *su lectures on artificial intelligence and machine learning*, 6(1):1–114, 2012.

[4] Simon Tong. *ACTIVE LEARNING: THEORY AND APPLICATIONS*. PhD thesis, STANFORD UNIVERSITY, 2001.

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

[6] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active

Table 1: Domain-wise ALC scores [44] of combinations of active learning and model training methods of active learning for blind imbalanced domains. We abbreviate model training methods, random sampling with center loss, and distance-based sampling with center loss as R, R+C, and D+C, respectively. We indicate ALC scores for each of six major/minor domain pairs of MNIST [13], EMNIST [14], and USPS [15], as well as domain pair average. ALC scores for each domain pair are the mean of four trials with four seeds. Bold fonts represent the methods that achieve the highest major, minor, and mean ALC scores for each domain pair and domain pair average.

Active learning		Random acquisition			Softmax probability			Softmax entropy			Softmax margin			BALD [22]			BatchBALD [25]		
	Model training	R	R+C	D+C	R	R+C	D+C	R	R+C	D+C	R	R+C	D+C	R	R+C	D+C	R	R+C	D+C
M/E	Major	0.922	0.930	0.925	0.926	0.932	0.928	0.911	0.921	0.913	0.933	0.941	**0.943**	0.917	0.924	0.928	0.930	0.934	0.927
	Minor	0.890	0.897	0.880	0.883	0.886	0.876	0.841	0.847	0.853	0.897	0.909	**0.914**	0.884	0.889	0.889	0.892	0.898	0.897
	Mean	0.906	0.914	0.903	0.904	0.909	0.902	0.876	0.884	0.883	0.915	0.925	**0.928**	0.900	0.906	0.909	0.911	0.916	0.912
E/M	Major	0.929	0.937	0.935	0.928	0.932	0.937	0.916	0.927	0.925	0.943	0.948	**0.949**	0.922	0.928	0.920	0.936	0.928	0.935
	Minor	0.888	0.900	0.900	0.896	0.908	0.906	0.888	0.899	0.895	0.908	0.919	**0.920**	0.886	0.906	0.890	0.906	0.898	0.912
	Mean	0.908	0.919	0.918	0.912	0.920	0.921	0.902	0.913	0.910	0.926	**0.934**	**0.934**	0.904	0.917	0.905	0.921	0.913	0.923
M/U	Major	0.915	0.923	0.927	0.929	0.931	0.935	0.917	0.919	0.922	0.935	0.941	**0.944**	0.924	0.928	0.928	0.929	0.937	0.935
	Minor	0.787	0.761	0.755	**0.792**	0.761	0.754	0.759	0.730	0.740	0.772	0.771	0.780	0.776	0.750	0.750	0.785	0.758	0.758
	Mean	0.851	0.842	0.841	0.860	0.846	0.844	0.838	0.825	0.831	0.854	0.856	**0.862**	0.850	0.839	0.839	0.857	0.847	0.847
U/M	Major	0.895	0.903	0.903	0.910	0.910	0.913	0.895	0.897	0.899	0.916	**0.925**	**0.925**	0.902	0.897	0.896	0.903	0.903	0.905
	Minor	0.708	0.726	0.721	0.801	0.805	0.808	0.775	0.789	0.785	0.792	0.784	0.785	0.834	0.840	0.838	0.842	0.850	**0.862**
	Mean	0.802	0.815	0.812	0.855	0.858	0.861	0.835	0.843	0.842	0.854	0.854	0.855	0.868	0.868	0.867	0.872	0.876	**0.884**
E/U	Major	0.923	0.929	0.933	0.930	0.933	0.938	0.914	0.915	0.920	0.941	**0.945**	**0.945**	0.922	0.927	0.923	0.930	0.936	0.939
	Minor	0.874	0.861	0.850	0.872	0.855	0.857	0.834	0.826	0.837	**0.882**	0.881	0.873	0.852	0.848	0.833	0.880	0.860	0.843
	Mean	0.898	0.895	0.892	0.901	0.894	0.898	0.874	0.870	0.879	0.911	**0.913**	0.909	0.887	0.888	0.878	0.905	0.898	0.891
U/E	Major	0.885	0.895	0.901	0.905	0.911	0.906	0.892	0.898	0.898	0.916	**0.926**	0.923	0.894	0.903	0.898	0.905	0.915	0.914
	Minor	0.846	0.856	0.855	0.865	0.868	0.878	0.859	0.863	0.866	0.881	0.887	0.887	0.879	0.884	0.878	0.888	**0.903**	0.895
	Mean	0.865	0.876	0.878	0.885	0.889	0.892	0.875	0.881	0.882	0.898	0.906	0.905	0.887	0.894	0.888	0.896	**0.909**	0.904
Avg.	Major	0.911	0.920	0.921	0.921	0.925	0.926	0.908	0.913	0.913	0.931	**0.938**	**0.938**	0.914	0.918	0.916	0.922	0.925	0.926
	Minor	0.832	0.833	0.827	0.852	0.847	0.846	0.826	0.826	0.829	0.855	0.859	0.860	0.852	0.853	0.846	**0.865**	0.861	0.861
	Mean	0.872	0.877	0.874	0.886	0.886	0.886	0.867	0.869	0.871	0.893	0.898	**0.899**	0.883	0.885	0.881	0.894	0.893	0.894

learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[7] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian active learning with image data. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1183–1192. PMLR, 06–11 Aug 2017.

[8] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. In *Proceedings of Robotics: Science and Systems*, FreiburgimBreisgau, Germany, June 2019.

[9] Elmar Haussmann, Michele Fenzi, Kashyap Chitta, Jan Ivanecky, Hanson Xu, Donna Roy, Akshita Mittel, Nicolas Koumchatzky, Clement Farabet, and Jose M. Alvarez. Scalable active learning for object detection. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1430–1435, 2020.

[10] Steven CH Hoi, Rong Jin, Jianke Zhu, and Michael R Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd international conference on Machine learning*, pages 417–424, 2006.

[11] Dana Hull. The tesla advantage: 1.3 billion miles of data. *Bloomberg, December*, 20, 2016.

[12] Luca Pizzuto, Christopher Thomas, Arthur Wang, and Ting Wu. How china will help fuel the revolution in autonomous vehicles. *McKinsey & Company, January*, 2019.

[13] Yann LeCun. The mnist database of handwritten digits. *http://yann.lecun.com/exdb/mnist/*, 1998.

[14] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*,

pages 2921–2926. IEEE, 2017.

[15] Jonathan J. Hull. A database for handwritten text recognition research. *IEEE Transactions on pattern analysis and machine intelligence*, 16(5):550–554, 1994.

[16] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2(5):6, 2018.

[17] John Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Advances in neural information processing systems*, 2, 1989.

[18] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[19] David J. C. MacKay. A practical bayesian framework for backpropagation networks. *Neural Comput.*, 4(3):448–472, may 1992.

[20] David J. C. MacKay. Probable networks and plausible predictions-a review of practical bayesian methods for supervised neural networks. *Network: computation in neural systems*, 6(3):469, 1995.

[21] George EP Box and George C Tiao. *Bayesian inference in statistical analysis*. John Wiley & Sons, 2011.

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

[23] Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

[24] Javad Azimi, Alan Fern, Xiaoli Z. Fern, Glencora Borradaile, and Brent Heeringa. Batch active learning via coordinated matching. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, page 307–314, Madison, WI, USA, 2012. Omnipress.

[25] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. BatchBALD: Efficient and diverse batch acquisition for deep bayesian active learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[26] Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Gholamreza Reza Haffari, Anton van den Hengel, and Javen Qinfeng Shi. Active learning by feature mixing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12237–12246, 2022.

[27] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, and Xinjing Cheng. Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8068–8078, 2022.

[28] Sylvia Frühwirth-Schnatter. Data augmentation and dynamic linear models. *Journal of time series analysis*, 15(2):183–202, 1994.

[29] Sebastien C Wong, Adam Gatt, Victor Stamatescu, and Mark D McDonnell. Understanding data augmentation for classification: when to warp? In *2016 international conference on digital image computing: techniques and applications (DICTA)*, pages 1–6. IEEE, 2016.

[30] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[31] David A Van Dyk and Xiao-Li Meng. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10(1):1–50, 2001.

[32] Yoon-Yeong Kim, Kyungwoo Song, JoonHo Jang, and Il-chul Moon. LADA: Look-ahead data acquisition via augmentation for deep active learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 22919–22930. Curran Associates, Inc., 2021.

[33] Hiroshi Kuwajima, Masayuki Tanaka, and Masatoshi Okutomi. Machine learning with blind imbalanced domains. *Electronic Imaging*, 34:1–6, 2022.

[34] Alhussein Fawzi, Horst Samulowitz, Deepak Turaga, and Pascal Frossard. Adaptive data augmentation for image classification. In *2016 IEEE international conference on image processing (ICIP)*, pages 3688–3692. Ieee, 2016.

[35] Ryo Takahashi, Takashi Matsubara, and Kuniaki Uehara. Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931, 2019.

[36] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020.

[37] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[38] Kunihiko Fukushima. Visual feature extraction by a multilayered network of analog threshold elements. *IEEE Transactions on Systems Science and Cybernetics*, 5(4):322–333, 1969.

[39] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010.

[40] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016.

[41] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U Rajendra Acharya, et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021.

[42] Josiah Davis, Jason Zhu, Jeremy Oldfather, Samual MacDonald, and Maciej Trzaskowski. Quantifying uncertainty in deep learning systems. https://d1.awsstatic.com/APG/quantifying-uncertainty-in-deep-learning-systems.pdf, 2020.

[43] Yves Tillé. *Sampling algorithms*. Springer, 2006.

[44] Isabelle Guyon, Gavin C. Cawley, Gideon Dror, and Vincent Lemaire. Results of the active learning challenge. In Isabelle Guyon, Gavin Cawley, Gideon Dror, Vincent Lemaire, and Alexander Statnikov, editors, *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, volume 16 of *Proceedings of Machine Learning Research*, pages 19–45, Sardinia, Italy, 16 May 2011. PMLR.

Author Biography

Hiroshi Kuwajima received his master's degree from Osaka University, Osaka, Japan in 2008 and joined Microsoft Development, Tokyo, Japan. He was a visiting researcher at Stanford University, CA, USA from 2013 to 2015. He is currently a project manager at DENSO CORPORA-

TION, Aichi, Japan.

Masayuki Tanaka received his Ph.D. degree from Tokyo Institute of Technology, Tokyo, Japan in 2003 and joined Agilent Technology. He was a research scientist at Tokyo Institute of Technology from 2004 to 2008, a visiting scholar at Stanford University from 2013 to 2014. He is currently an associate professor at Tokyo Institute of Technology.

Masatoshi Okutomi received his master's degree from Tokyo Institute of Technology in 1983 and joined Canon Inc., Tokyo, Japan. He was a visiting research scientist at Carnegie Mellon University, PA, USA from 1987 to 1990. He received his Ph.D. degree from Tokyo Institute of Technology in 1993. He is currently a Professor at Tokyo Institute of Technology.