

Supplementary Material: Divergence Optimization for Noisy Universal Domain Adaptation

Qing Yu^{1,2} Atsushi Hashimoto² Yoshitaka Ushiku²

¹The University of Tokyo ²OMRON SINIC X Corporation

yu@hal.t.u-tokyo.ac.jp {atsushi.hashimoto, yoshitaka.ushiku}@sinicx.com

1. Experiments on Toy Datasets.

We observed the behavior of the proposed method on a toy dataset, where we used scikit-learn to generate isotropic Gaussian blobs as the source samples and the target samples. The goal of the experiment was to observe the learned classifiers' boundaries. For the source samples, we generated three clusters of blobs and labeled them as red, blue and orange, respectively. Some label noise is also introduced in source samples. Target common samples were generated around the distribution of the red and blue source samples, where target private samples were generated far from the source samples. We generated 300 source and target samples per class as the training samples. The model consists of a 3-layered fully-connected network for a feature generator and 3-layered fully-connected networks for classifiers. In Fig. I, we visualized the learned decision boundary of the proposed method and its variants as follows:

Source Only The method that training with source samples only with Eq. (7), which means the final objectives are as follows:

$$\min_{G, F_1, F_2} \mathcal{L}_{sup}(D_s). \quad (16)$$

Ours w/o select The method that training without using the small-loss selection by Eq. (9), which means the objectives of Step A-1 are as follows:

$$\min_{G, F_1, F_2} \mathcal{L}_s(D_s). \quad (17)$$

Ours w/o sep The method that training without separating the divergence between the classifiers as Eq. (11), which means the following equation is used instead of Eq. (11):

$$\mathcal{L}_t(D_t) = 0. \quad (18)$$

Ours w/ KL The method that training with using general symmetric KL-divergence in Eq. (11), which means the following equation is used instead of Eq. (11):

$$\mathcal{L}_t(D_t) = \tilde{\mathcal{L}}_{SKLD}(D_t) = \frac{1}{N} \sum_{i=1}^N \tilde{\mathcal{L}}_{crs}(D_t) - \frac{1}{N} \sum_{i=1}^N \tilde{\mathcal{L}}_{ent}(D_t) \quad (19)$$

Regarding Source Only and Ours w/o select shown in Fig. Ia and Fig. Ib, a large region is detected as the target private class having large \mathcal{L}_{crs} due to the label noise of the source samples. As for Ours w/o sep and Ours w/ KL shown in Fig. Ic and Fig. Id, though they are not effected by the label noise, they cannot detect target private samples well because their loss function is not suitable for separate the target common and target private classes. Compared to other variants, the proposed method shown in Fig. Ie achieves the best performance. In our proposed method, the classifiers are not effected by the noisy source samples, and target private samples exist in the area having large divergence, where our proposed method attempted to increase the divergence on these target private samples and decrease that on target common samples. The code of the toy problem is provided in the supplementary.

2. Sensitivity to hyper-parameters.

In Fig. II and Fig. III, we show the sensitivity to hyper-parameters on task A→D with P20 noise and D→W with S45 noise in the Office dataset, respectively. Regarding α , which controls the number of source samples ignored in each mini-batch, we can see that when no samples are dropped ($\alpha = 0$), the performance could deteriorate due to the label noise. Though the true noise rate is 0.2 and 0.45, dropping more samples with larger α still could achieve good performance. As for λ , which controls the weight of divergence loss on source samples, $\lambda = 0.1$ shows the best results in both settings. Regarding δ and m , which control divergence separation on target samples, deciding δ as $\log |C_s| \approx 3$ works well, and setting m around 1 achieves best performances. As for n , which is the time of repeating Step C during the training process, $n > 1$ shows similar results in the setting of P20 A→D, and $n = 4$ shows the best results in the setting of S45 D→W.

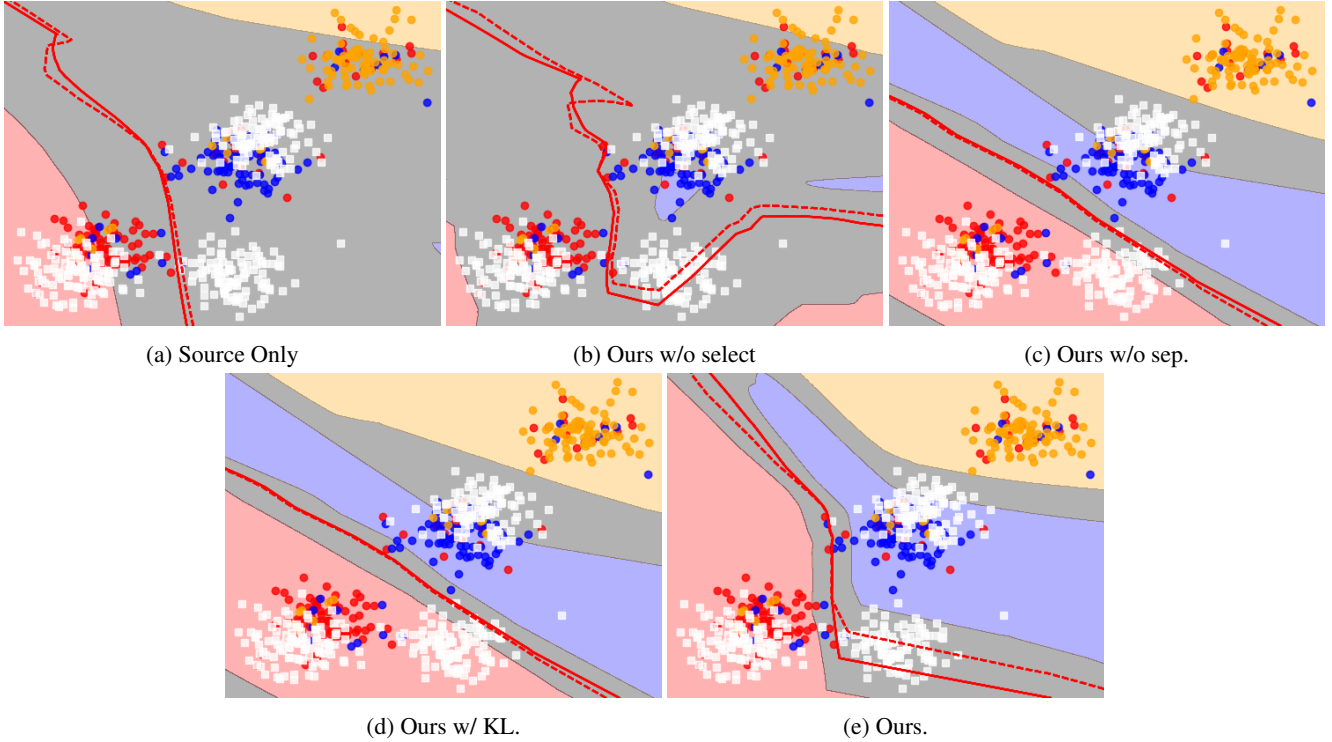
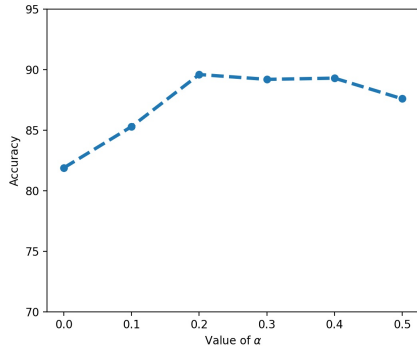
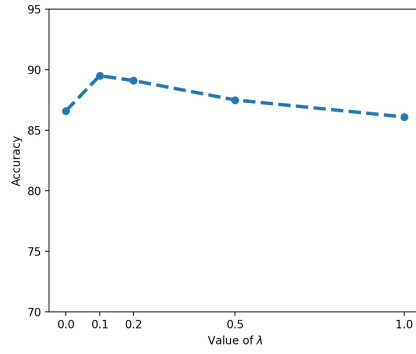


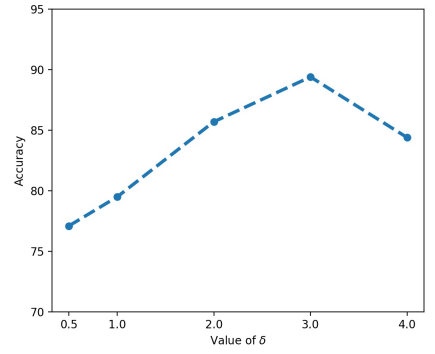
Figure I: (Best viewed in color.) Visualization of the toy problem. Red, blue and orange points indicate the three classes of the source samples, where the orange class is the source private class. The labels of source samples are corrupted with 20% symmetric noise. White points represent target samples, and the target samples at the right bottom of each figure are target private samples. The dashed and normal lines are two decision boundaries in our method for identifying the red class. The pink, light blue and light yellow regions are where the results of both classifiers are class red, blue and orange, respectively. The gray regions are detected as the target private class having large divergence by Eq. (15).



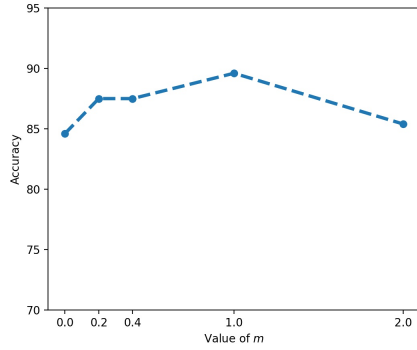
(a) Accuracy w.r.t. value of α in Eq. (9).



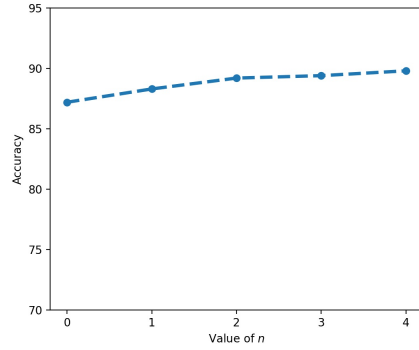
(b) Accuracy w.r.t. value of λ in Eq. (8).



(c) Accuracy w.r.t. value of δ in Eq. (11).

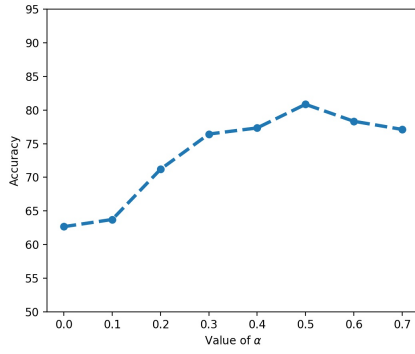


(d) Accuracy w.r.t. value of m in Eq. (11).

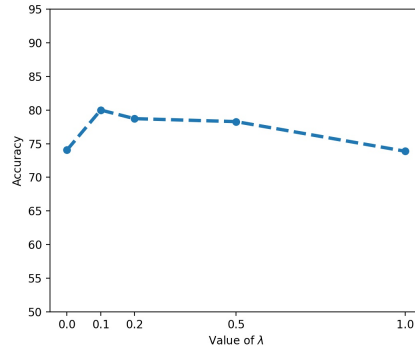


(e) Accuracy w.r.t. value of n in Step C.

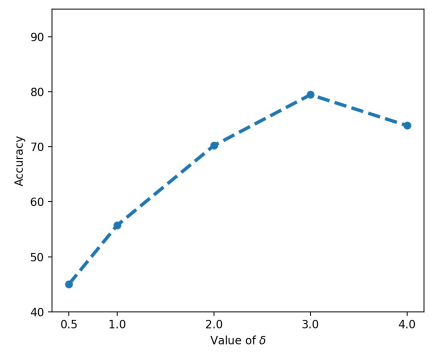
Figure II: Analysis of the sensitivity to hyper-parameters in task A→D with P20 noise.



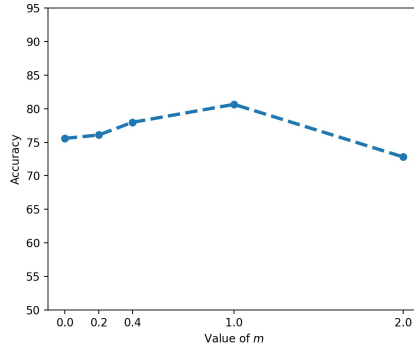
(a) Accuracy w.r.t. value of α in Eq. (9).



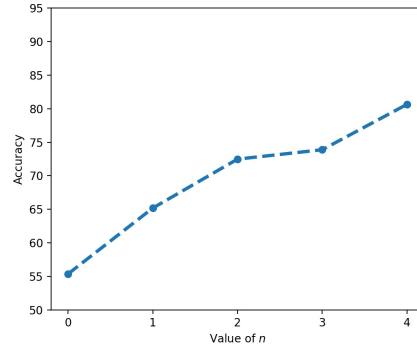
(b) Accuracy w.r.t. value of λ in Eq. (8).



(c) Accuracy w.r.t. value of δ in Eq. (11).



(d) Accuracy w.r.t. value of m in Eq. (11).



(e) Accuracy w.r.t. value of n in Step C.

Figure III: Analysis of the sensitivity to hyper-parameters in task $D \rightarrow W$ with S45 noise.