DETECT: A Deep Discriminative Clustering Baseline for Unsupervised and Universal Domain Adaptation

Bin Deng, Yabin Zhang, Kui Jia

2020-06-15

Content

- Universal Domain adaptation: Background
- Existing work
- A proposed solution: DETECT
- Discussion and future work

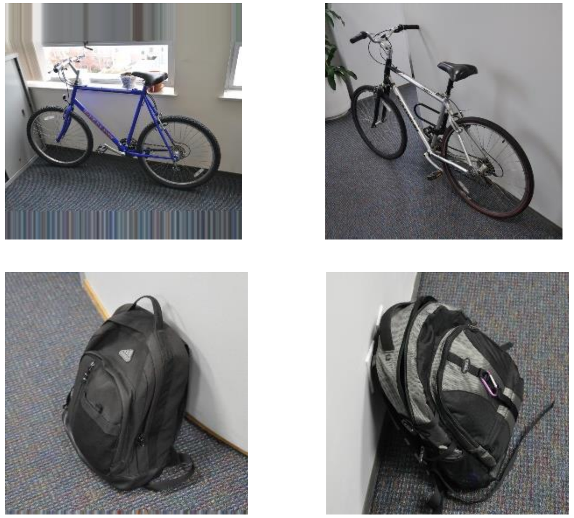
Background

Domain Adaptation

Source Domain $\sim P_S(X_S, Y_S)$
lots of **labeled** data

$\neq$

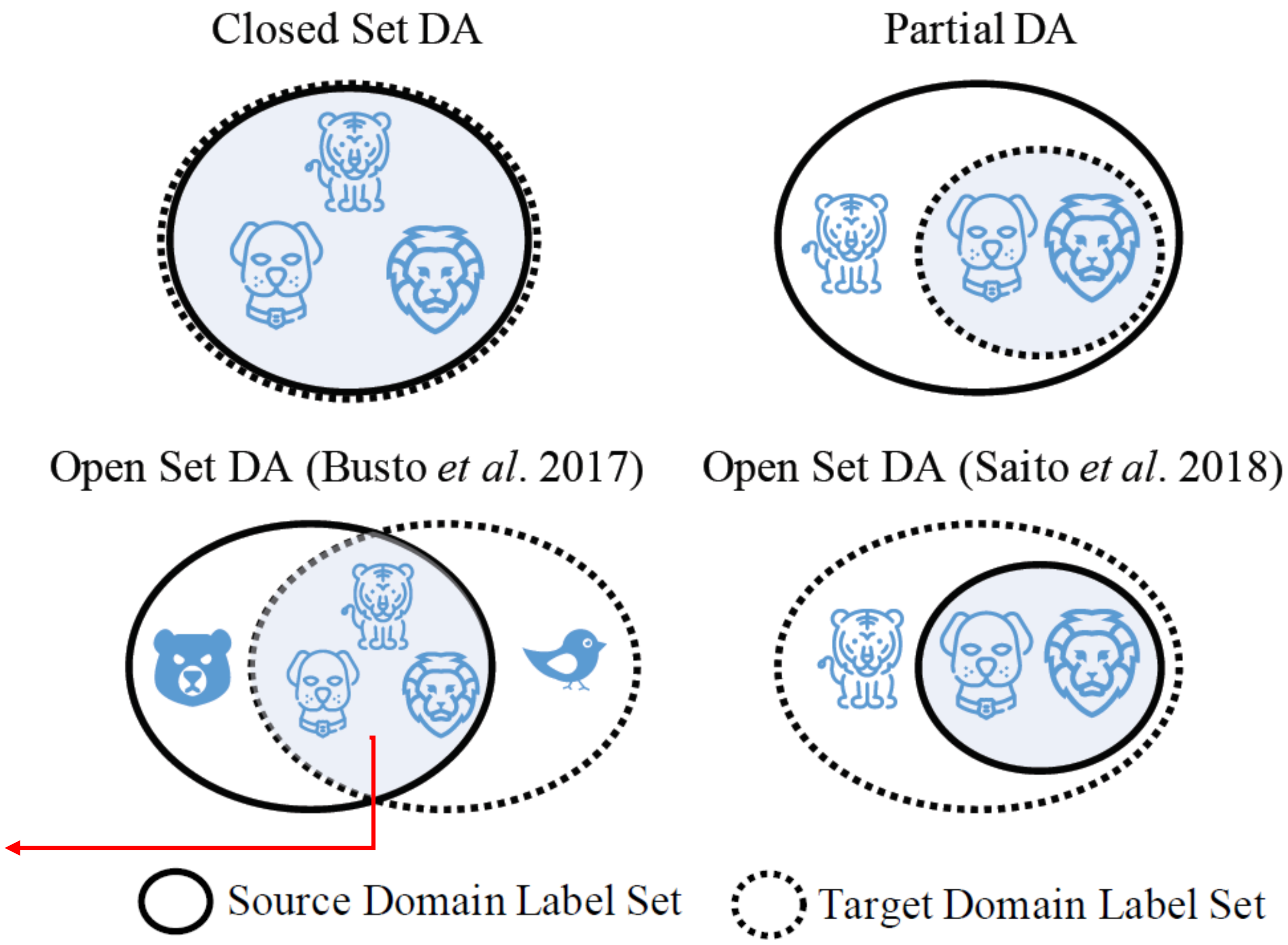
Target Domain $\sim P_T(X_T, Y_T)$
unlabeled ~~or limited labels~~



Unsupervised domain adaptation (UDA)

Train a model for Target Domain

Different Variants of UDA

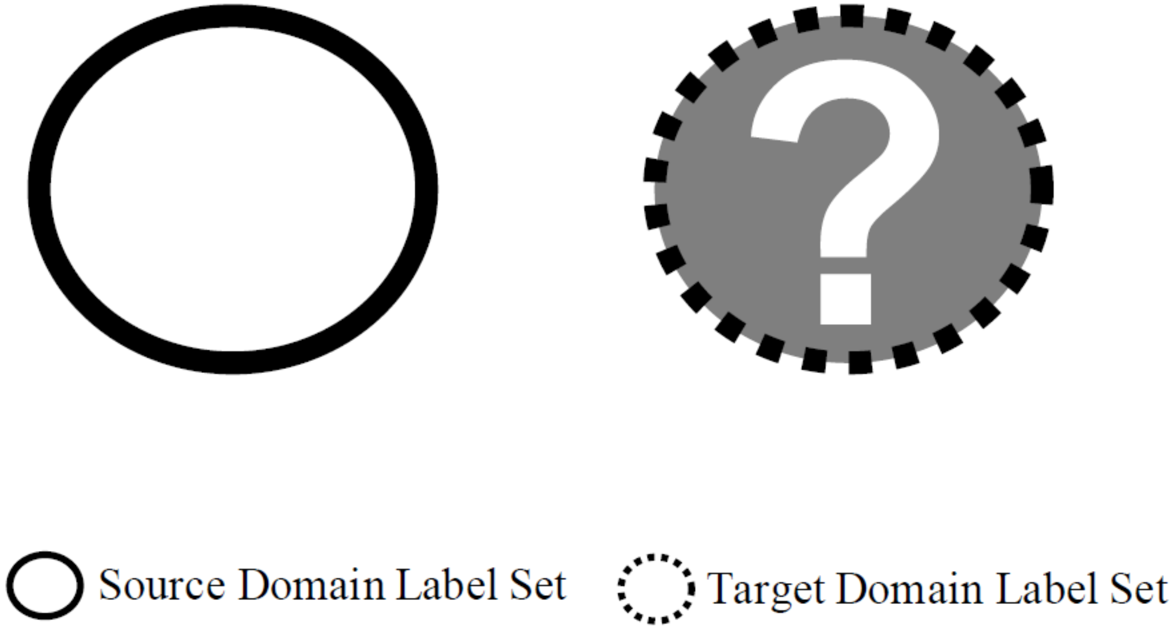


Shared label set is known in advance

[1] Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In ICCV, 2017.
[2] Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. Open set domain adaptation by backpropagation. In ECCV, 2018.

Universal Domain Adaptation

$$predictFunction: x^t \rightarrow \{sourceLabelSet + \text{unknow } class\}$$

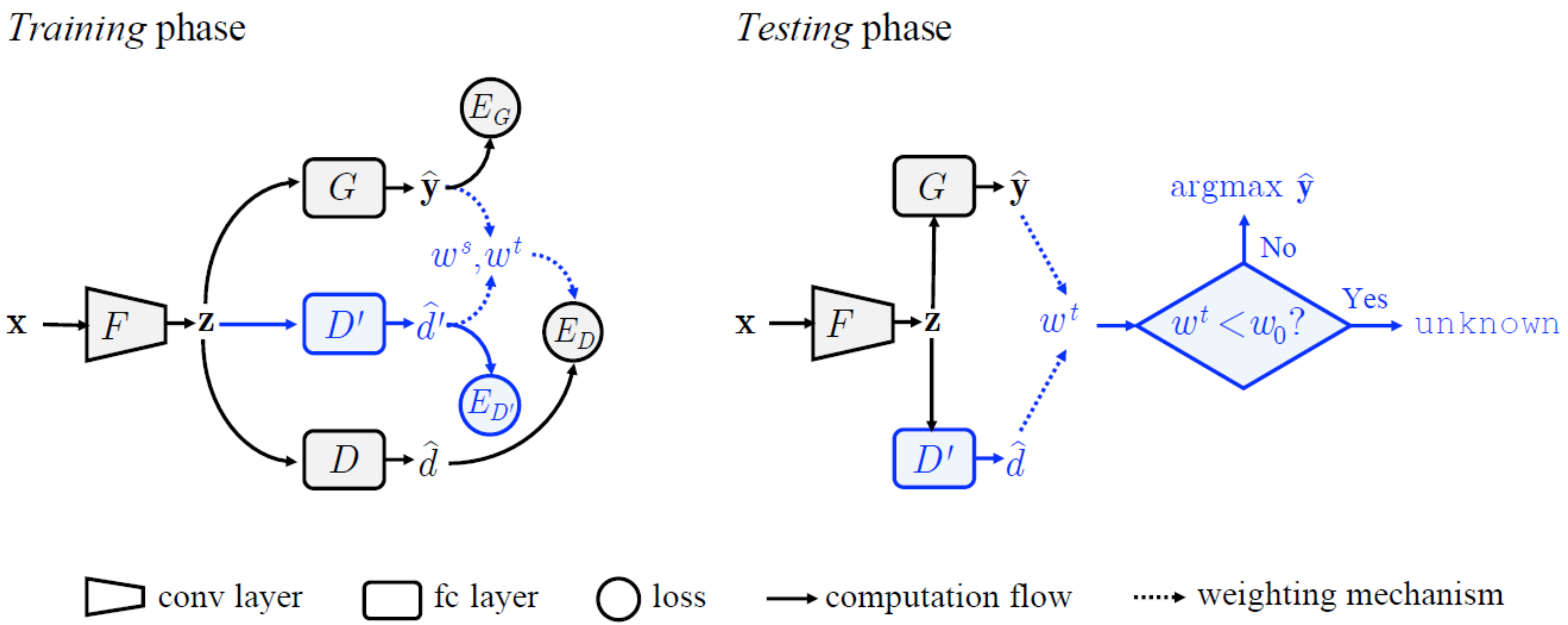


You, K., Long, M., Cao, Z., Wang, J., and Jordan, M. I. Universal domain adaptation. In CVPR, 2019.

Existing work

--- A review of existing algorithms for universal domain adaptation

Universal Adaptation Network (UAN)



$$E_G = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim p} L\left(\mathbf{y}, G(F(\mathbf{x}))\right)$$

$$\begin{aligned} E_{D'} = & - \mathbb{E}_{\mathbf{x}\sim p} \log D'\left(F\left(\mathbf{x}\right)\right) \\ & - \mathbb{E}_{\mathbf{x}\sim q} \log\left(1 - D'\left(F\left(\mathbf{x}\right)\right)\right) \end{aligned}$$

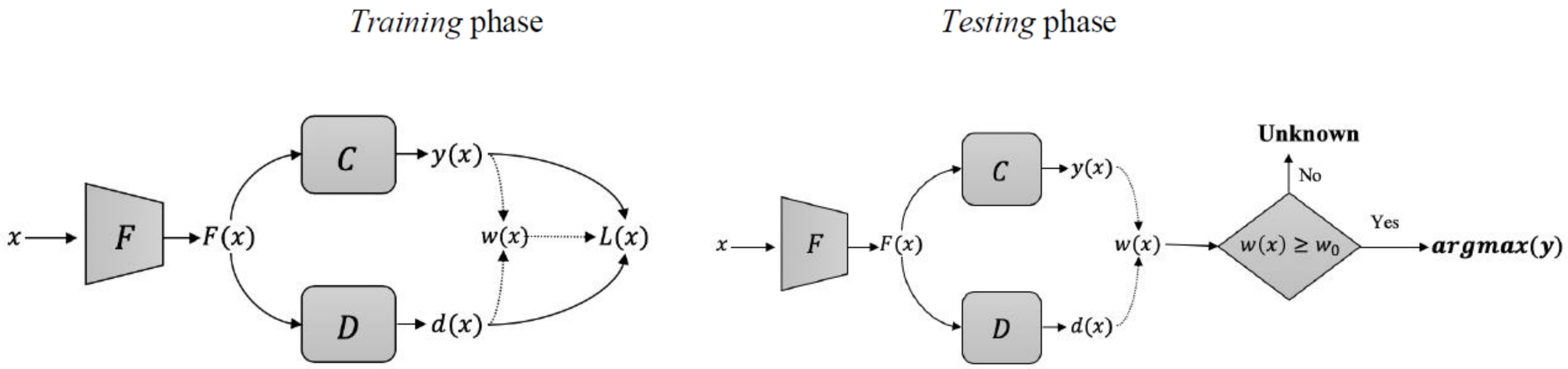
$$\begin{aligned} E_D = & - \mathbb{E}_{\mathbf{x}\sim p} w^s(\mathbf{x}) \log D\left(F\left(\mathbf{x}\right)\right) \\ & - \mathbb{E}_{\mathbf{x}\sim q} w^t(\mathbf{x}) \log\left(1 - D\left(F\left(\mathbf{x}\right)\right)\right) \end{aligned}$$

$$w^s(\mathbf{x}) = \frac{H(\hat{\mathbf{y}})}{\log|\mathcal{C}_s|} - \hat{d}'(\mathbf{x})$$

$$w^t(\mathbf{x}) = \hat{d}'(\mathbf{x}) - \frac{H(\hat{\mathbf{y}})}{\log|\mathcal{C}_s|}$$

You, K., Long, M., Cao, Z., Wang, J., and Jordan, M. I. Universal domain adaptation. In CVPR, 2019.

A Sample Selection Approach for Universal Domain Adaptation



$L(x)$
$= classification\ loss\ of\ pseudo\ label$
$+\ batch\ diversity\ loss$
$+\ domain\ adversarial\ loss$

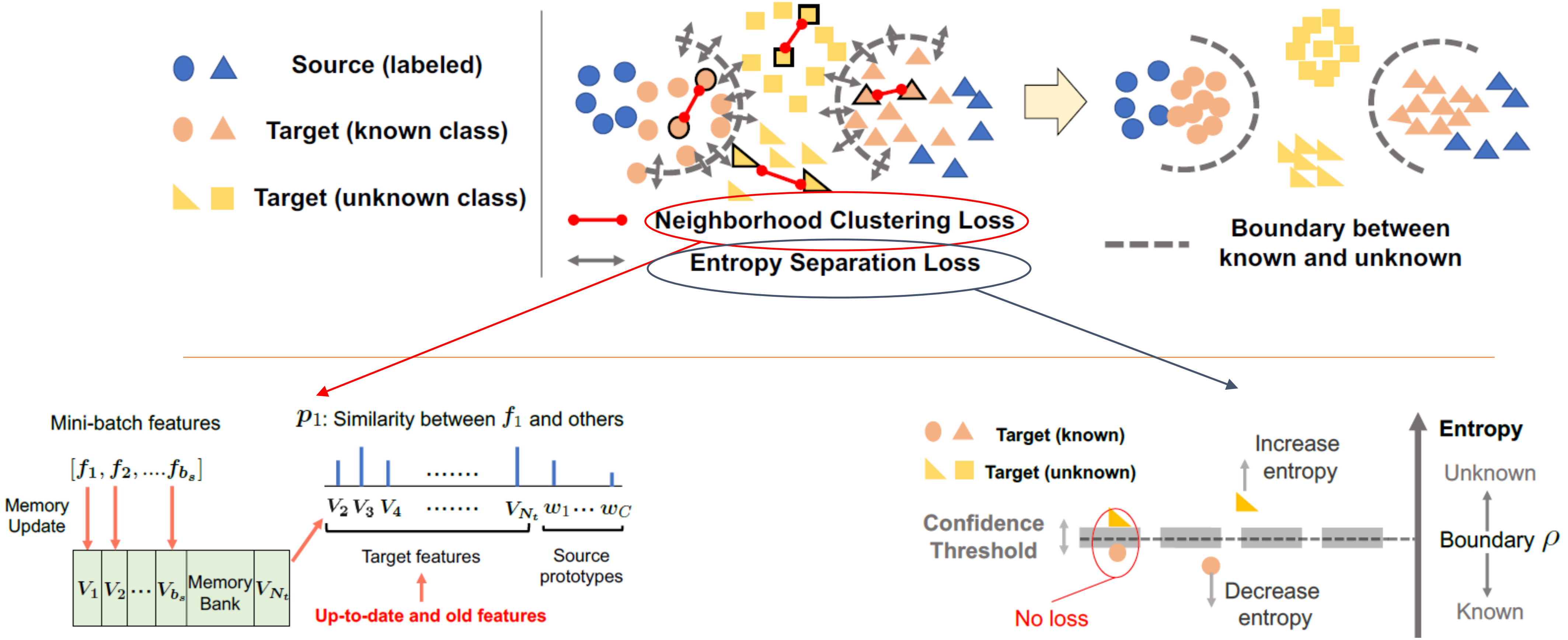
$$w(x) = d(x) + \max \bar{y}(x)$$

Only target instances are weighted

Omri Lifshitz, Lior Wolf. A Sample Selection Approach for Universal Domain Adaptation. arXiv:2001.05071, 2020.

DANCE

Domain specific batch normalization is also used to enhance alignment between source and target domain.



$$\mathcal{L}_{\text{nc}} = -\frac{1}{|B_t|} \sum_{i \in B_t} \sum_{j=1, j \neq i}^{N_t+K} p_{i,j} \log(p_{i,j}).$$

$$\mathcal{L}_{\text{es}}(\boldsymbol{p}_i) = \begin{cases} -|H(\boldsymbol{p}_i) - \rho| & (|H(\boldsymbol{p}_i) - \rho| > m), \\ 0 & otherwise. \end{cases} \qquad \mathcal{L}_{\text{es}} = \frac{1}{|B_t|} \sum_{i \in B_t} \mathcal{L}_{\text{es}}(\boldsymbol{p}_i).$$

Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Kate Saenko. Universal Domain Adaptation through Self Supervision. arXiv:2002.07953, 2020.

A proposed solution: DETECT

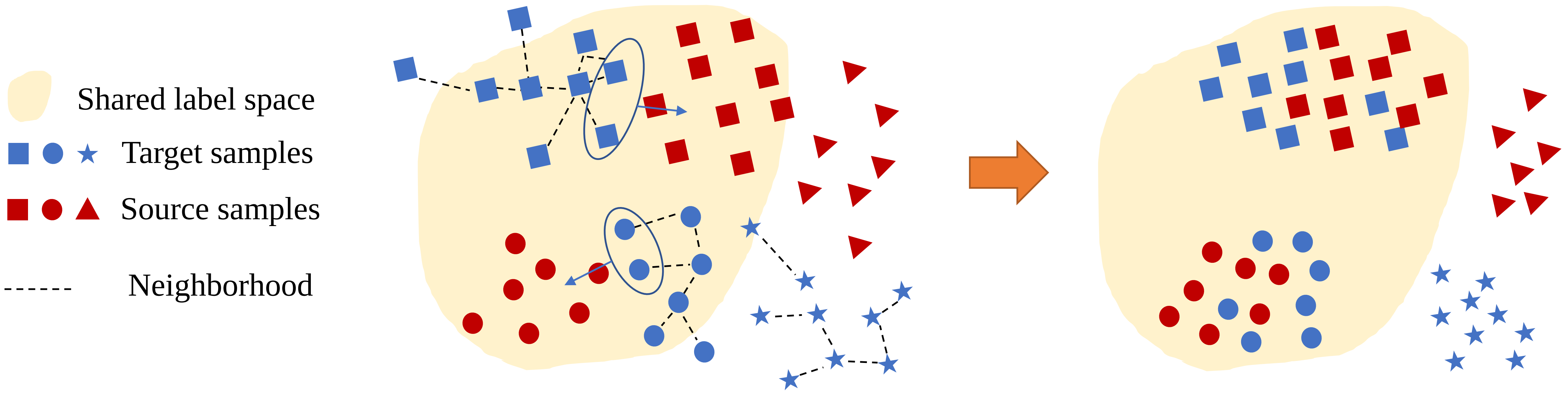
--- A Deep Discriminative Clustering Baseline for Unsupervised and Universal Domain Adaptation

Our basic idea

- To learn the intrinsic discrimination of target data in an unsupervised manner, regularized by the labeled discrimination of source data in an unknown but shared label space.

- To reduce contamination of the intrinsic target discrimination during the source regularization.

Framework



- Applying deep discriminative clustering to those target instances with high probabilities in the shared label space, regularized by the source labeled data.

- A neighborhood-preserved module is introduced to reduce contamination of the intrinsic target discrimination.

Deep discriminative clustering

- Introducing an auxiliary distribution Q^t, and then minimizing:

$$KL(Q^t||P^t)$$

A self-training method to build decision boundaries

- The Q^t is computed by minimizing:

$$KL(Q^t||P^t) + KL(g^t||u^t)$$

P^t: a prior distribution of selected target data over K clusters.
g^t: the empirical probability of target assignments over the K clusters with $g^t = Mean_i Q_i^t$.
u^t: a uniform distribution.

[1] Ryan Gomes, Andreas Krause, and Pietro Perona. Discriminative clustering by regularized information maximization. In NIPS, 2010.
[2] Junyuan Xie, Ross B. Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In ICML, 2016.
[3] Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In ICCV, 2017.

Neighborhood-preserved embedding

Target instances

Feature Extractor

Feature Extractor

Building the similarity of neighboring instances from the initial training step.

$$\mathcal{L}_{DE}(\vartheta, \theta; \{\boldsymbol{x}_i^t\}_{i=1}^{n_t}) = \frac{1}{2}\sum_{i,j=1}^{n_t} w_{ij}^t \cdot \|\boldsymbol{z}_i^t - \boldsymbol{z}_j^t\|^2 = \mathrm{Tr}(\boldsymbol{Z}^{t\top}\boldsymbol{L}^t\boldsymbol{Z}^t),$$

Neighborhood-preserved embedding in the latter training steps.

Cai, Xiaofei He, Jiawei Han, and Thomas S. Huang. Graph regularized nonnegative matrix factorization for data representation. IEEE Trans. Pattern Anal. Mach. Intell., 33(8):1548–1560, 2011.

An OOD technique by using auxiliary data

Minimizing $R_{OOD} = KL(U||P')$

Auxiliary data

Feature Extractor

Trainable Classifier

U : uniform distributions

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting **out-of-distribution** samples. In ICLR, 2018.

Learning and inference

- Learning:

Deep discriminative clustering

Neighborhood-preserved embedding

$$\mathcal{L}_{DETECT}(\vartheta, \theta) = \alpha \mathcal{L}_{DE} + \mathcal{R}_{TE} + \beta \mathcal{L}_{CT}$$

Regularization by source and auxiliary data

- Inference:

$$\hat{y}^t = \begin{cases} \arg\max_k [\boldsymbol{f}(\varphi(\boldsymbol{x}^t))]_k, & H(\boldsymbol{x}^t) < \tau \log(K), \\ y^{t/s}, & \text{otherwise.} \end{cases}$$

Unknown label

Threshold

Experiment

- Ablation study

$\mathcal{L}_{CT}$	$\mathcal{L}_{DE}$	$\mathcal{R}_{OOD}$	A2W	D2W	W2D	A2D	D2A	W2A	Avg.
-	-	-	80.34	95.38	97.02	85.11	82.92	82.25	87.17
✓	-	-	87.89	94.98	94.64	92.80	91.01	90.07	91.90
✓	✓	-	93.00	95.29	96.24	93.13	91.33	90.88	93.31
✓	✓	✓	94.43	96.87	96.31	94.69	91.08	92.06	94.24

Table 1: Ablation experiments on the Office-31 dataset with the evaluation metric of **AA** (%).

$\mathcal{L}_{CT}$	$\mathcal{L}_{DE}$	$\mathcal{R}_{OOD}$	A2W	D2W	W2D	A2D	D2A	W2A	Avg.
-	-	-	74.41	87.94	84.04	72.19	80.66	76.43	79.28
✓	-	-	72.28	82.74	79.82	75.20	76.29	72.21	76.42
✓	✓	-	82.56	86.76	85.14	89.76	77.68	75.33	82.87
✓	✓	✓	86.29	92.5	83.43	90.96	83.88	81.31	86.40

Table 2: Ablation experiments on the Office-31 dataset with the evaluation metric of **OA** (%).

Experiment

- Convergence performance and threshold sensitivity

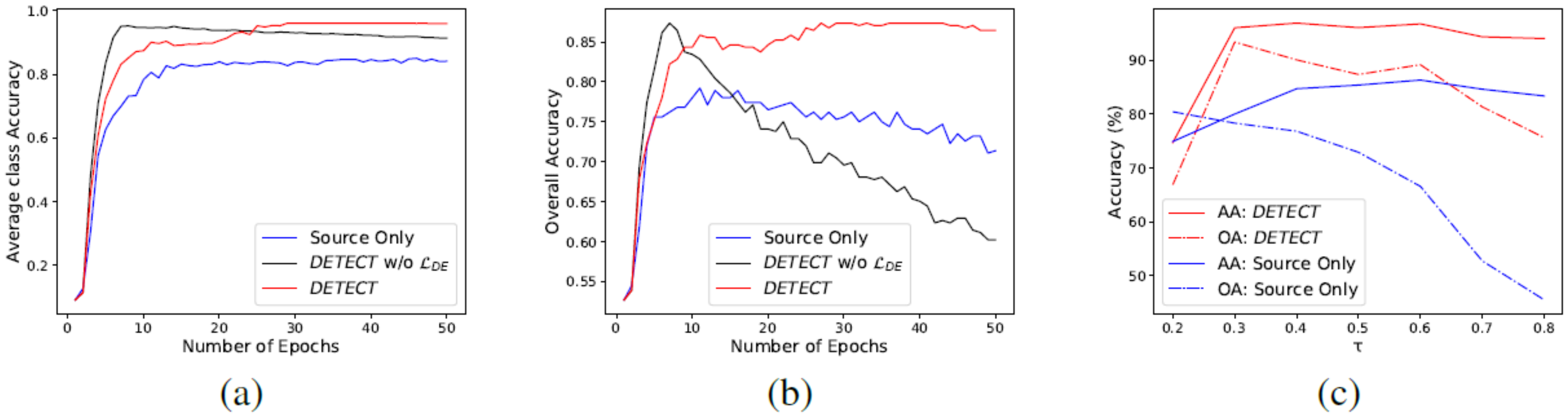
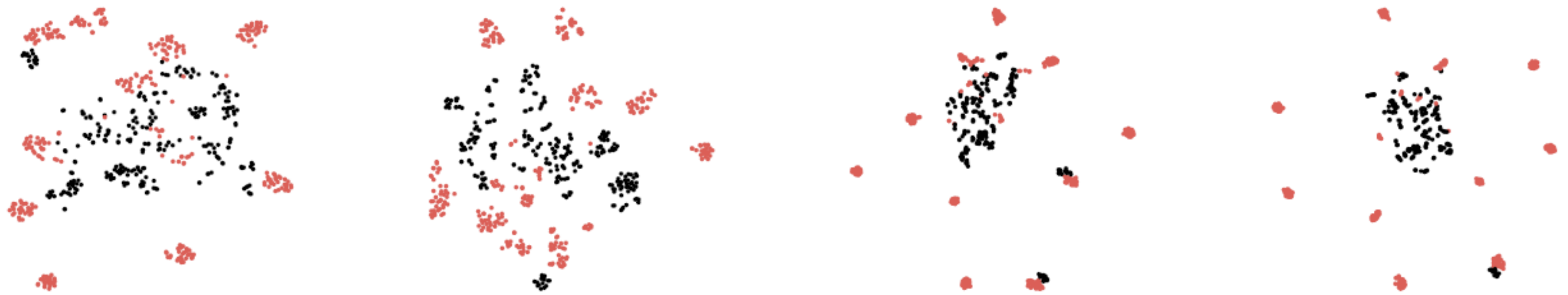


Figure 1: (a)-(b): Convergence performance on the task of A to D according to the evaluation metrics of AA and OA, respectively. (c): Performance w.r.t. the threshold τ in the task of A to D.

Experiment

- Feature Visualization

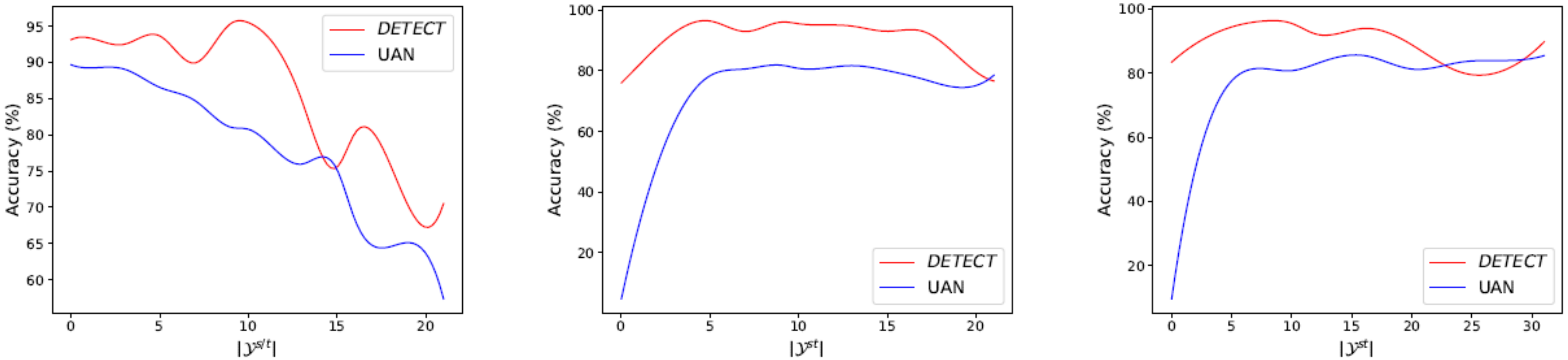


(a) Source Only (b) UAN [47] (c) $DETECT$ w/o $\mathcal{L}_{DE}$ (d) $DETECT$

Figure 2: t-SNE visualization of target features in the task of A to W. Red dots represent target samples that enjoy shared labels with the source domain (i.e. $\mathcal{Y}^{st}$) while black dots are samples from unknown classes (i.e. $\mathcal{Y}^{t/s}$). (Best viewed in color)

Experiment

- Analysis on different settings



(a) $|\mathcal{Y}^{st}| = 10$ (b) $|\mathcal{Y}^{s/t}| = 10$ (c) $|\mathcal{Y}^{t}| = |\mathcal{Y}^{s}| + 1$

Figure 3: Average class accuracy (%) of different ξ-UDA settings on the task of A to W. The results of UAN [47] are reproduced with its public code.

Experiment

- Results comparing to the existing methods

Method	Office-31							ImageNet-Caltech		VisDA
	A2W	D2W	W2D	A2D	D2A	W2A	Avg.	Im2Cal	Cal2Im	
Source Only [47]	75.94	89.60	90.91	80.45	78.83	81.42	82.86	70.28	65.14	52.80
DANN[11]	80.65	80.94	88.07	82.67	74.82	83.54	81.78	71.37	66.54	52.94
RTN[29]	85.70	87.80	88.91	82.69	74.64	83.26	84.18	71.94	66.15	53.92
IWAN[48]	85.25	90.09	90.00	84.27	84.22	86.25	86.68	72.19	66.48	58.72
PADA[7]	85.37	79.26	90.91	81.68	55.32	82.61	79.19	65.47	58.73	44.98
ATI[4]	79.38	92.60	90.08	84.40	78.85	81.57	84.48	71.59	67.36	54.81
OSBP[38]	66.13	73.57	85.62	72.92	47.35	60.48	67.68	62.08	55.48	30.26
UAN[47]	85.62	94.77	**97.99**	86.50	85.45	85.12	89.24	75.28	70.17	60.83
Method in [26]	90.25	95.25	96.96	88.84	90.19	89.30	91.80	76.13	74.67	64.31
DANCE [35]	92.8	**97.8**	97.7	91.6	**92.2**	91.4	93.9	-	-	69.2
$DETECT$	**94.43**	96.87	96.31	**94.69**	91.08	**92.06**	**94.24**	**78.52**	**76.81**	**71.38**

Table 3: Average class accuracy (%) on datasets of Office-31, ImageNet-Caltech, and VisDA2017.

	C2I	C2P	C2B	I2C	I2P	I2B	P2C	P2I	P2B	B2C	B2I	B2P	Avg.
Source Only	81.43	72.21	58.86	87.90	76.90	58.38	85.33	80.19	55.05	87.90	77.14	67.72	74.08
UAN [47]	79.81	70.38	57.33	84.00	72.97	58.00	81.33	77.52	54.10	80.77	72.57	64.27	71.09
$DETECT$	**89.05**	**75.34**	**61.24**	**92.00**	**78.20**	**60.57**	**90.38**	**86.10**	**60.67**	**94.00**	**88.00**	**74.21**	**79.15**

Table 4: Average class accuracy (%) on the ImageCLEF-DA dataset. The results of UAN [47] are reproduced with its public code.

Discussion

- Relationship between the detection of instances of unknown categories in Universal domain adaptation & the Out-of-Distribution (OOD) detection?
- Is really Universal domain adaptation solvable, and in what conditions does it can work?
- Is the evaluation metric of AA enough?
- Tune the hyper-parameters on a test dataset is really well?

Future work

- Focusing on semantic OOD detection;

- Discovering a new dataset for Universal domain adaptation, where the test data is may never be used to tune the parameters of any model;

- A better evaluation metric is need to better balance between the classification accuracy and the detection accuracy, which should depend on different applications;

- Discovering the conditions when Universal domain adaptation can work and can not work?

Thank you for listening