Universal Domain Adaptation

Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan

Paper Reading

Jingge Wang

2020/3/27

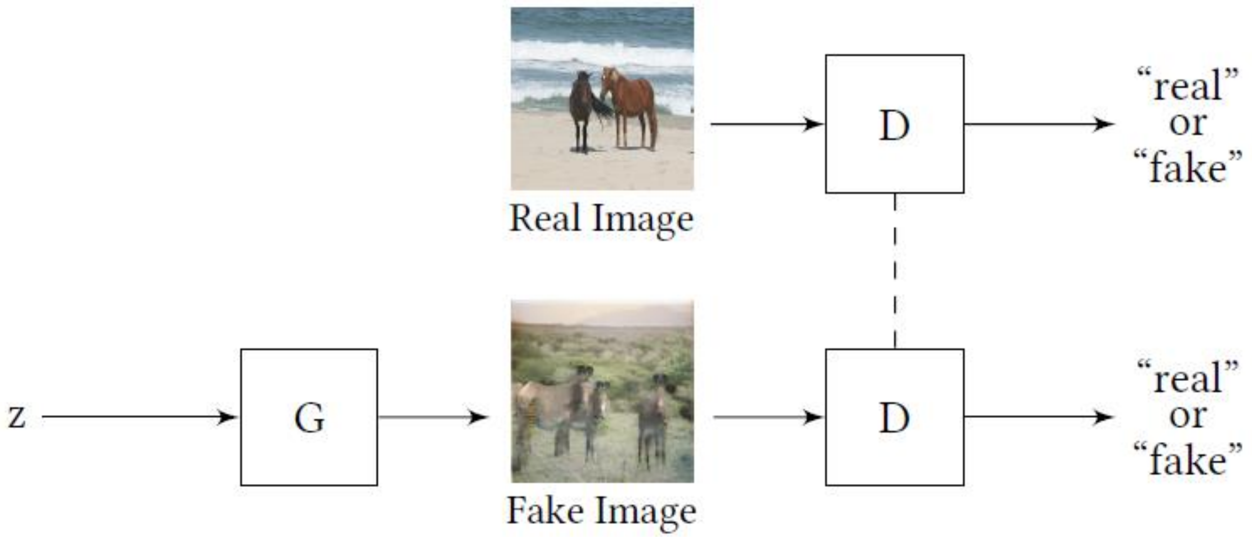
Preliminary

- GAN

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

  - Fix generator G: maximize the probability of assigning the correct label to both training examples and samples generated.

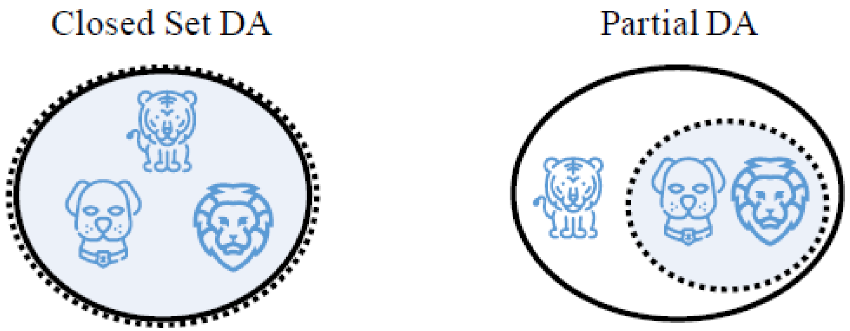
  - Fix discriminator D: minimize the probability of D making correct decision.



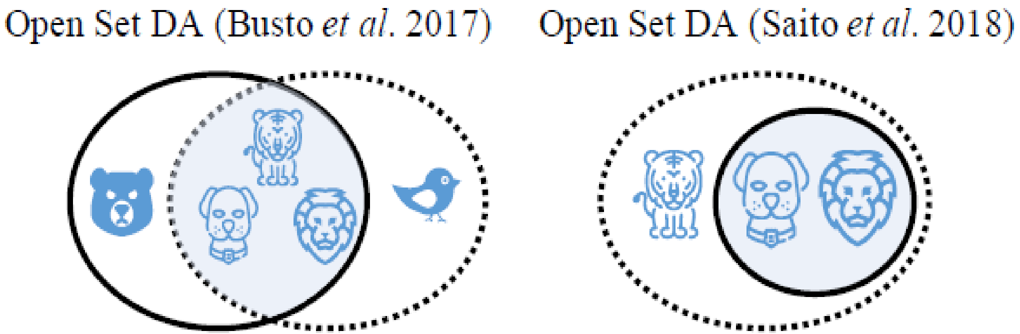
Related Work

Commonness between two domain label space $\mathcal{C}_s$ and $\mathcal{C}_t$ $\quad \xi = \frac{|\mathcal{C}_s \cap \mathcal{C}_t|}{|\mathcal{C}_s \cup \mathcal{C}_t|}$

- closed set domain adaptation $\mathcal{C}_t = \mathcal{C}_s$
- partial domain adaptation $\mathcal{C}_t \subset \mathcal{C}_s$



- open set domain adaptation



Related Work: closed set DA $\mathcal{C}_t = \mathcal{C}_s$

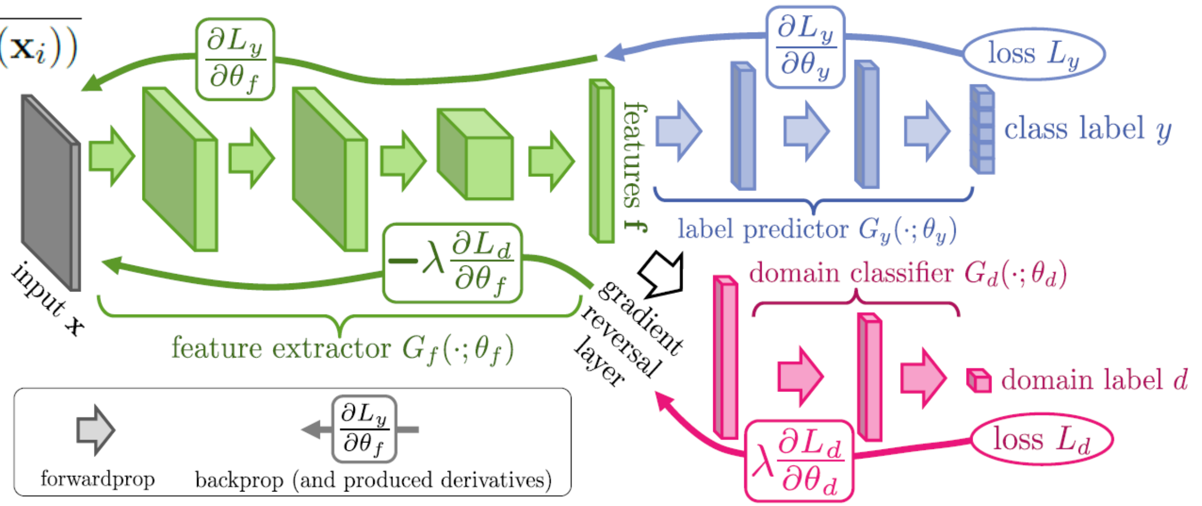
- **DANN** (Domain-Adversarial Training of Neural Networks)
  - Label Classifier G_y: **minimize** classification loss on source domain
  - Feature extractor G_f:
    - Discriminativeness: **minimize** classification loss on source domain
    - Domain invariance: **maximize** source / target domain classification loss
  - Domain classifier G_d: **minimize** source / target domain classification loss

$$E(\theta_f, \theta_y, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_y^i(\theta_f, \theta_y) - \lambda \left(\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_d^i(\theta_f, \theta_d) + \frac{1}{n'}\sum_{i=n+1}^{N} \mathcal{L}_d^i(\theta_f, \theta_d) \right)$$

$$\mathcal{L}_d\big(G_d(G_f(\mathbf{x}_i)), d_i\big) = d_i \log \frac{1}{G_d(G_f(\mathbf{x}_i))} + (1-d_i) \log \frac{1}{1-G_d(G_f(\mathbf{x}_i))}$$

$$(\hat{\theta}_f, \hat{\theta}_y) = \underset{\theta_f, \theta_y}{\operatorname{argmin}}\ E(\theta_f, \theta_y, \hat{\theta}_d)\,,$$

$$\hat{\theta}_d = \underset{\theta_d}{\operatorname{argmax}}\ E(\hat{\theta}_f, \hat{\theta}_y, \theta_d)\,.$$



Related Work: partial DA $\mathcal{C}_t \subset \mathcal{C}_s$

- **SAN**(Selective Adversarial Networks)
  - Negative transfer ↓ : Decrease influence of $\mathcal{C}_s \backslash \mathcal{C}_t$
  - Positive transfer ↑ : Reduce distribution discrepancy between $p_{\mathcal{C}_t} \neq q$

- Original DANN： Single discriminator

$$C_0\left(\theta_f, \theta_y, \theta_d\right) = \frac{1}{n_s} \sum_{\mathbf{x}_i \in \mathcal{D}_s} L_y\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right), y_i\right) - \frac{\lambda}{n_s + n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_s \cup \mathcal{D}_t} L_d\left(G_d\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right)$$

- ① Instance-level weighting
  - Multi-discriminator

$$L_d' = \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \sum_{\mathbf{x}_i \in \mathcal{D}_s \cup \mathcal{D}_t} \hat{y}_i^k L_d^k\left(G_d^k\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right)$$

Related Work: partial DA $\mathcal{C}_t \subset \mathcal{C}_s$

- **SAN**(Selective Adversarial Networks)
  - ② Class-level weighting

$$L_d = \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \left[\left(\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k \right) \times \left(\sum_{\mathbf{x}_i \in (\mathcal{D}_s \cup \mathcal{D}_t)} \hat{y}_i^k L_d^k \left(G_d^k \left(G_f \left(\mathbf{x}_i \right) \right), d_i \right) \right) \right]$$

  - ③ entropy minimization

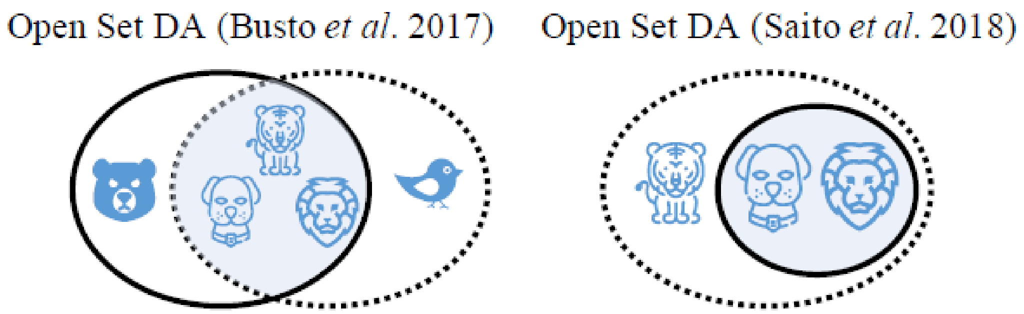
$$E = \frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} H \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right) \right) \qquad H \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right) \right) = - \sum_{k=1}^{|\mathcal{C}_s|} \hat{y}_i^k \log \hat{y}_i^k$$

  - Integrating all

$$\begin{aligned} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right) &= \frac{1}{n_s} \sum_{\mathbf{x}_i \in \mathcal{D}_s} L_y \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right), y_i \right) + \frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} H \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right) \right) \\ &- \frac{\lambda}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \left(\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k \right) \sum_{\mathbf{x}_i \in \mathcal{D}_s \cup \mathcal{D}_t} \hat{y}_i^k L_d^k \left(G_d^k \left(G_f \left(\mathbf{x}_i \right) \right), d_i \right) \end{aligned}$$

Related Work: open set DA

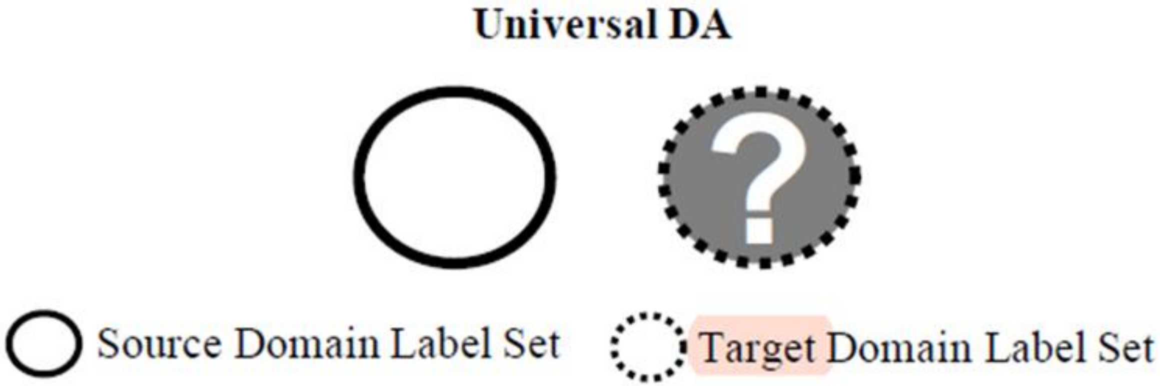
- **Assign-and-Transform-Iteratively (ATI)**
- **OSBP**

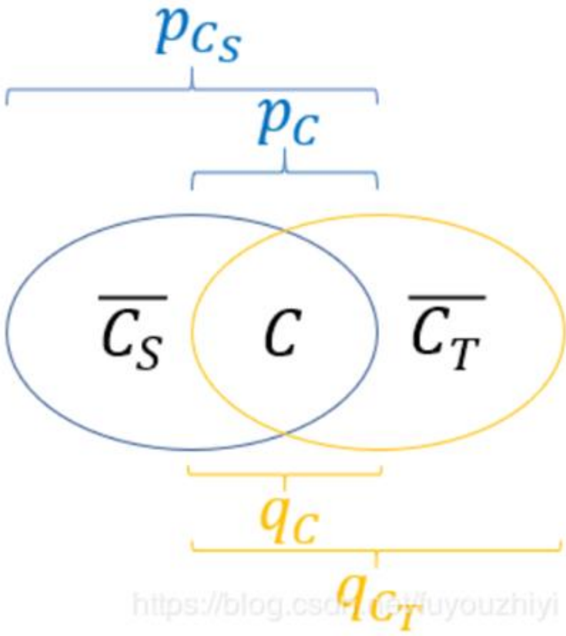


Introduction

- Settings: $\mathcal{C} = \mathcal{C}_s \cap \mathcal{C}_t$ $\quad \overline{\mathcal{C}}_s = \mathcal{C}_s \setminus \mathcal{C}$ and $\overline{\mathcal{C}}_t = \mathcal{C}_t \setminus \mathcal{C}$

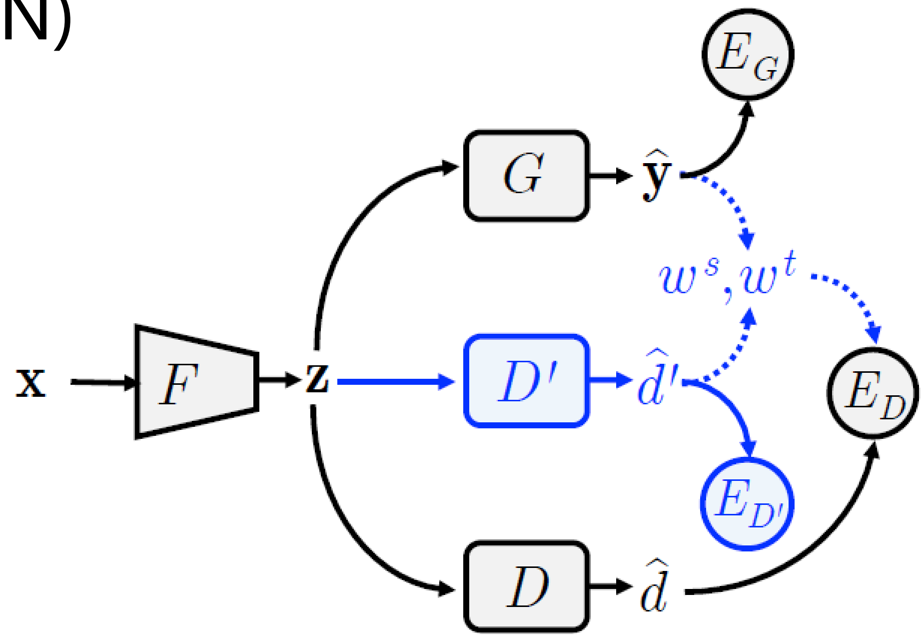
- No information about the target label set
- Negative transfer: Should not match whole source set with target set
- How to mark target samples from $\overline{\mathcal{C}}_t$ as "unknown"
- Learn model $\min \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim q_{\mathcal{C}}} [f(\mathbf{x}) \neq \mathbf{y}]$





Method：Universal Adaptation Network (UAN)

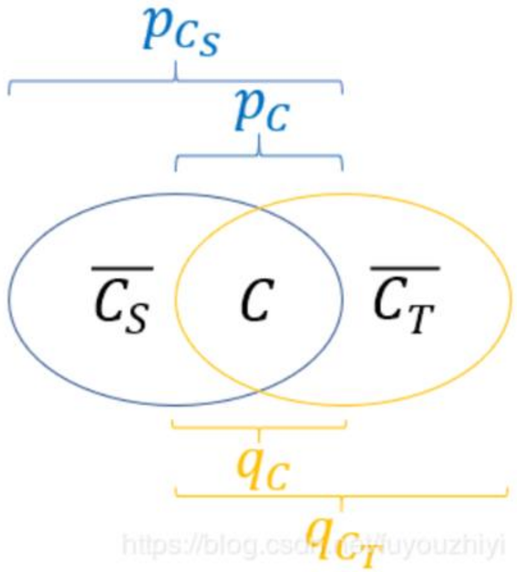
Training phase



- feature extractor F
- label classifier G
  - Probability $\hat{\mathbf{y}} = G(\mathbf{z})$ of x over $\mathcal{C}_s$

$$E_G = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim p} L\left(\mathbf{y}, G(F(\mathbf{x}))\right) \qquad (1)$$

- **non-adversarial** domain discriminator D'
  - $\hat{d}' = D'(\mathbf{z})$ similarity of x to the source domain
  - $\mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} \hat{d}'$
  - $$\begin{aligned} E_{D'} = & -\mathbb{E}_{\mathbf{x}\sim p} \log D'\left(F\left(\mathbf{x}\right)\right) \\ & -\mathbb{E}_{\mathbf{x}\sim q} \log\left(1 - D'\left(F\left(\mathbf{x}\right)\right)\right) \end{aligned} \qquad (2)$$



Method：Universal Adaptation Network (UAN)

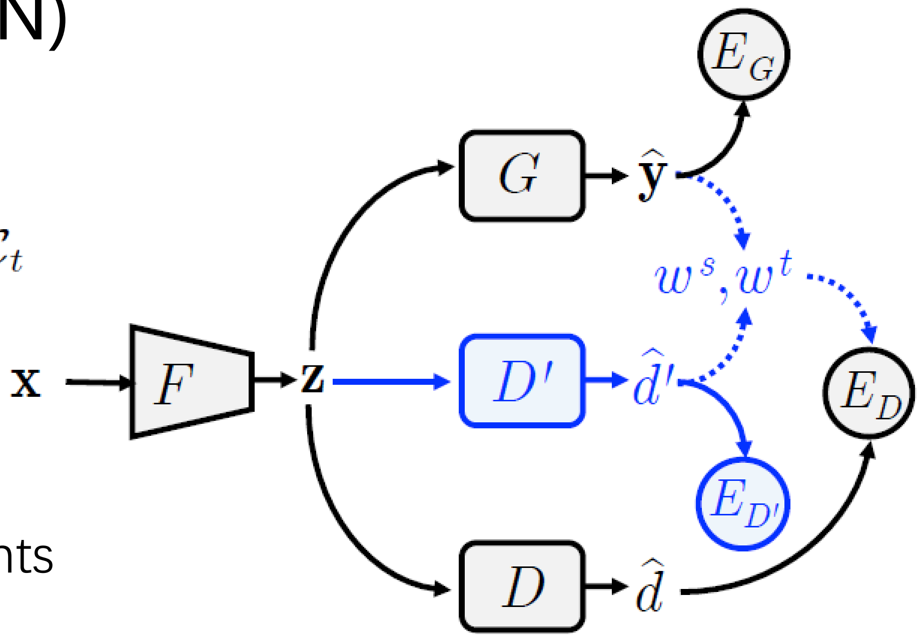
- **Adversarial** domain discriminator D
  - D distinguishes the source and target data in $\mathcal{C} = \mathcal{C}_s \cap \mathcal{C}_t$

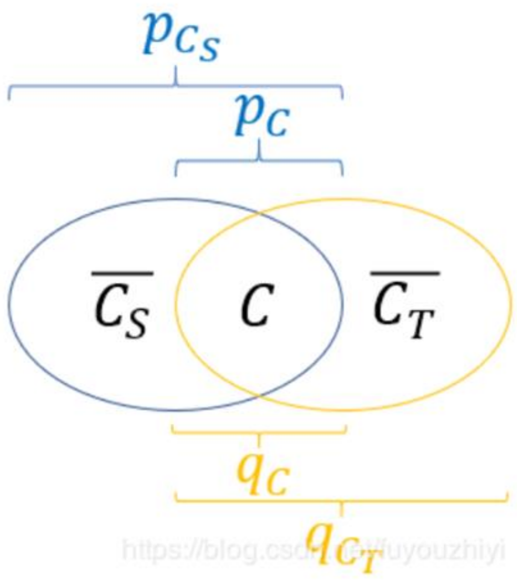
$$E_D = -\mathbb{E}_{\mathbf{x}\sim p} w^s(\mathbf{x}) \log D(F(\mathbf{x})) - \mathbb{E}_{\mathbf{x}\sim q} w^t(\mathbf{x}) \log(1 - D(F(\mathbf{x}))) \quad (3)$$

  - sample-level transferability criterion for source data points and target data points

$$\mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} w^s(\mathbf{x}) > \mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} w^s(\mathbf{x}) \\ \mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} w^t(\mathbf{x}) > \mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} w^t(\mathbf{x}) \quad (6)$$

Training phase





Method：Universal Adaptation Network (UAN)

- **Adversarial** domain discriminator D

$$
\begin{aligned}
\mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} w^s(\mathbf{x}) &> \mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} w^s(\mathbf{x}) \\
\mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} w^t(\mathbf{x}) &> \mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} w^t(\mathbf{x})
\end{aligned} \tag{6}
$$

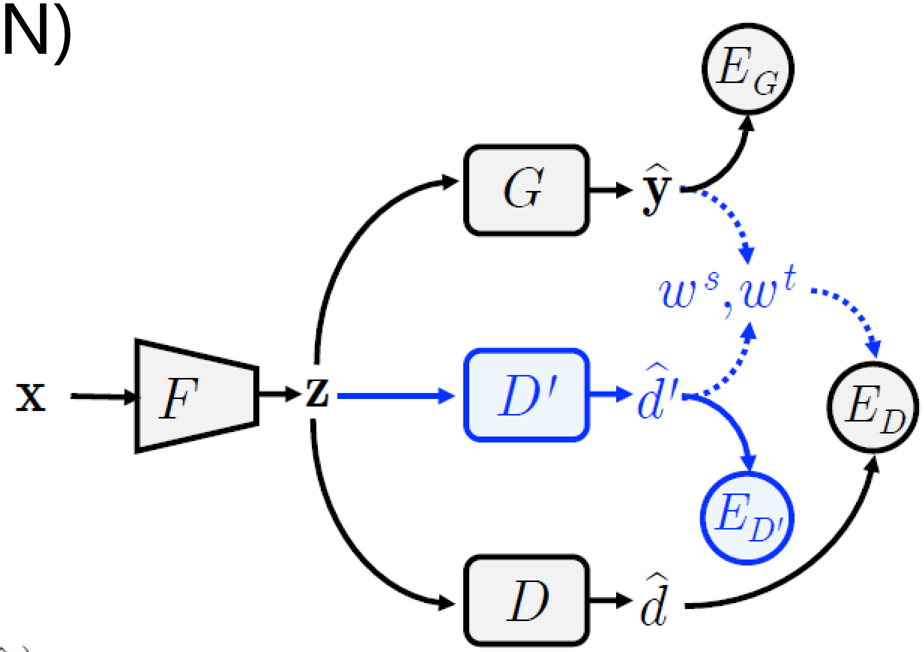
$$
\mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} \hat{d}' > \mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} \hat{d}'
$$

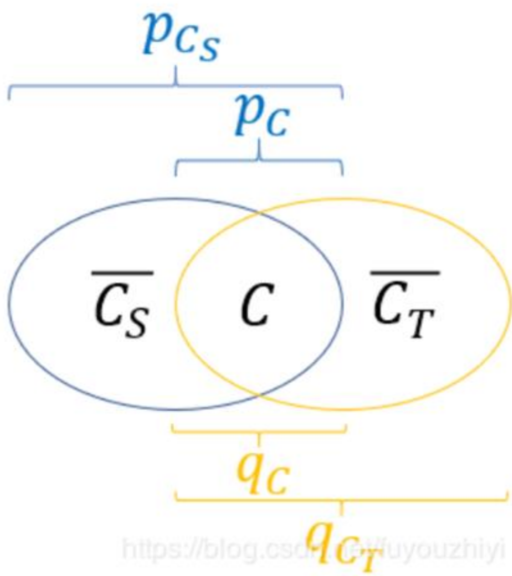
$$
\mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} H(\hat{\mathbf{y}}) > \mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} H(\hat{\mathbf{y}}) > \mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} H(\hat{\mathbf{y}}) > \mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} H(\hat{\mathbf{y}})
$$

$$
w^s(\mathbf{x}) = \frac{H(\hat{\mathbf{y}})}{\log|\mathcal{C}_s|} - \hat{d}'(\mathbf{x}) \tag{7}
$$

$$
w^t(\mathbf{x}) = \hat{d}'(\mathbf{x}) - \frac{H(\hat{\mathbf{y}})}{\log|\mathcal{C}_s|} \tag{8}
$$

Training phase





Method：Universal Adaptation Network (UAN)

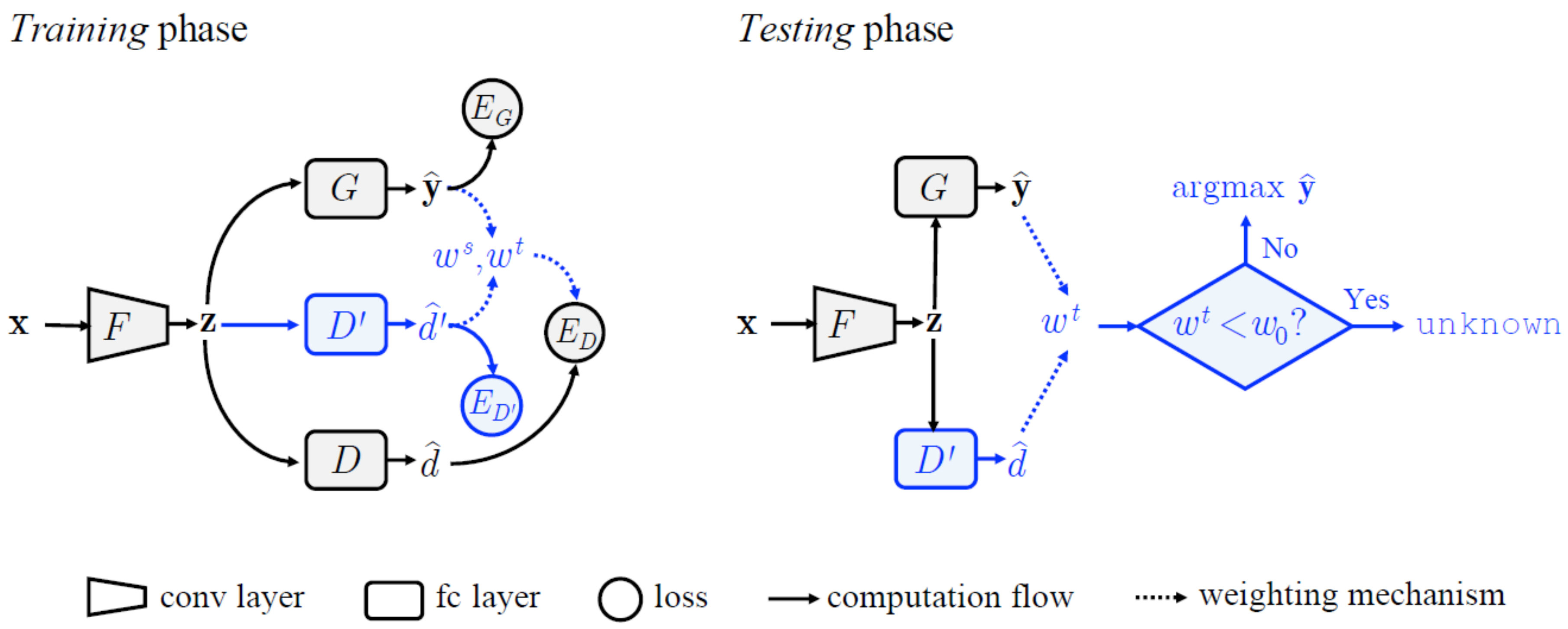


Figure 2. The training and testing phases of the Universal Adaptation Network (UAN) designed for Universal Domain Adaptation (UDA).

$$\max_{D} \min_{F,G} E_G - \lambda E_D$$

$$\min_{D'} E_{D'} \tag{4}$$

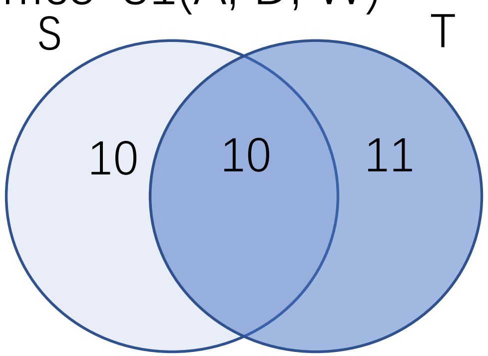
$$y(\mathbf{x}) = \begin{cases} \text{unknown} & w^t < w_0 \\ \operatorname{argmax}(\hat{\mathbf{y}}) & w^t \geq w_0 \end{cases} \tag{5}$$

Experiments

- Datasets

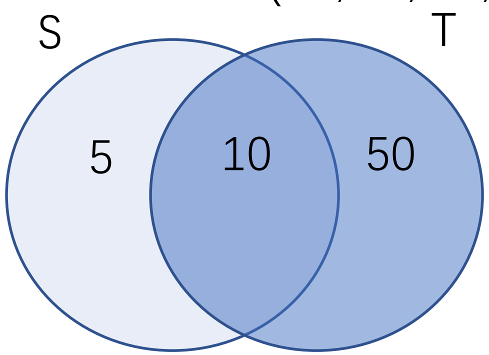
$$\xi = \frac{|\mathcal{C}_s \cap \mathcal{C}_t|}{|\mathcal{C}_s \cup \mathcal{C}_t|}$$

- Office-31(A, D, W)



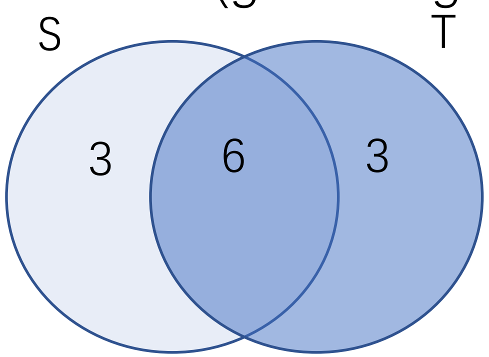
$\xi = 0.32$

- Office-Home(Ar, Cl, Pr,Rw)



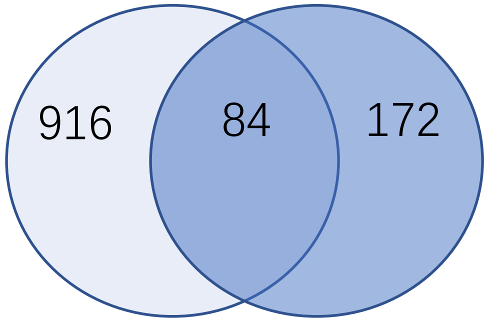
$\xi = 0.15$

- VisDA2017(game engines, real-world)



$\xi = 0.50$

- ImageNet-Caltech(ImageNet-1K, Caltech-256)



$\xi = 0.07$

Classification Results

- Results

Table 1. Average class accuracy (%) of universal domain adaptation tasks on **Office-Home** ($\xi = 0.15$) dataset (ResNet)

Method	Office-Home												
	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Avg
ResNet [13]	59.37	76.58	87.48	69.86	71.11	81.66	73.72	56.30	86.07	78.68	59.22	78.59	73.22
DANN [6]	56.17	81.72	86.87	68.67	73.38	83.76	69.92	56.84	85.80	79.41	57.26	78.26	73.17
RTN [23]	50.46	77.80	86.90	65.12	73.40	85.07	67.86	45.23	85.50	79.20	55.55	78.79	70.91
IWAN [45]	52.55	81.40	86.51	70.58	70.99	85.29	74.88	57.33	85.07	77.48	59.65	78.91	73.39
PADA [45]	39.58	69.37	76.26	62.57	67.39	77.47	48.39	35.79	79.60	75.94	44.50	78.10	62.91
ATI [28]	52.90	80.37	85.91	71.08	72.41	84.39	74.28	57.84	85.61	76.06	60.17	78.42	73.29
OSBP [35]	47.75	60.90	76.78	59.23	61.58	74.33	61.67	44.50	79.31	70.59	54.95	75.18	63.90
UAN	**63.00**	**82.83**	**87.85**	**76.88**	**78.70**	**85.36**	**78.22**	**58.59**	**86.80**	**83.37**	**63.17**	**79.43**	**77.02**

Classification Results

- Results

Table 2. Average class accuracy (%) on **Office-31** ($\xi = 0.32$), **ImageNet-Caltech** ($\xi = 0.07$) and **VisDA2017** ($\xi = 0.50$) (ResNet)

Method	Office-31							ImageNet-Caltech		VisDA
	A → W	D → W	W → D	A → D	D → A	W → A	Avg	I → C	C → I	
ResNet [13]	75.94	89.60	90.91	80.45	78.83	81.42	82.86	70.28	65.14	52.80
DANN [6]	80.65	80.94	88.07	82.67	74.82	83.54	81.78	71.37	66.54	52.94
RTN [23]	85.70	87.80	88.91	82.69	74.64	83.26	84.18	71.94	66.15	53.92
IWAN [45]	85.25	90.09	90.00	84.27	84.22	86.25	86.68	72.19	66.48	58.72
PADA [45]	85.37	79.26	90.91	81.68	55.32	82.61	79.19	65.47	58.73	44.98
ATI [28]	79.38	92.60	90.08	84.40	78.85	81.57	84.48	71.59	67.36	54.81
OSBP [35]	66.13	73.57	85.62	72.92	47.35	60.48	67.68	62.08	55.48	30.26
UAN	**85.62**	**94.77**	**97.99**	**86.50**	**85.45**	**85.12**	**89.24**	**75.28**	**70.17**	**60.83**

Classification Results

- Results

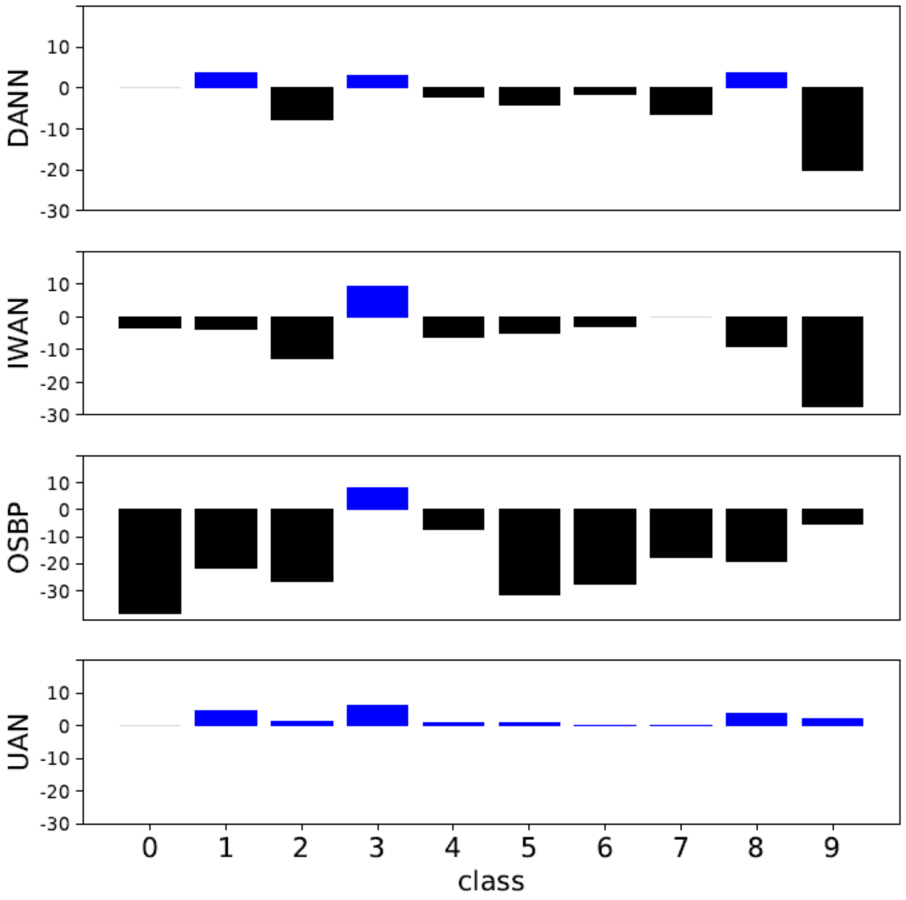
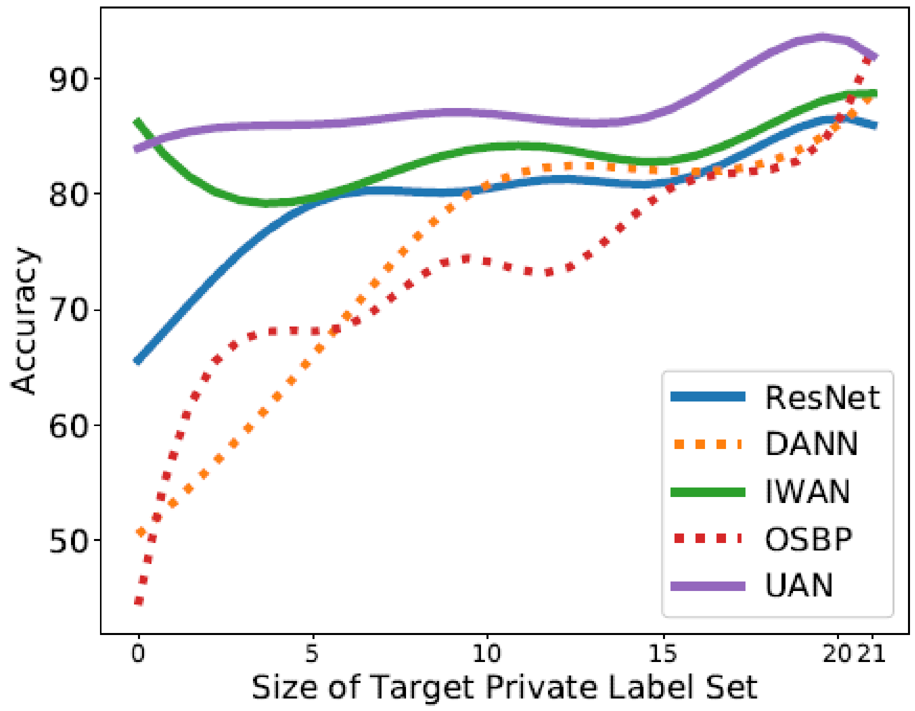


Figure 4. (a) The negative transfer influence in UDA (task **Ar** → **Cl**)

Analysis on Different UDA Settings

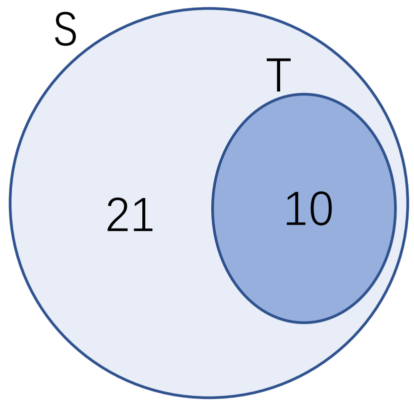
- Varying Size of $|\overline{\mathcal{C}}_t|$

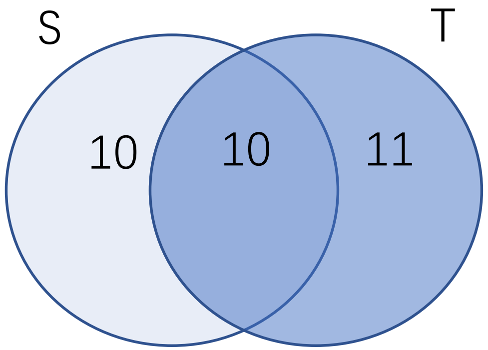


(a) Accuracy w.r.t. $|\overline{\mathcal{C}}_t|$

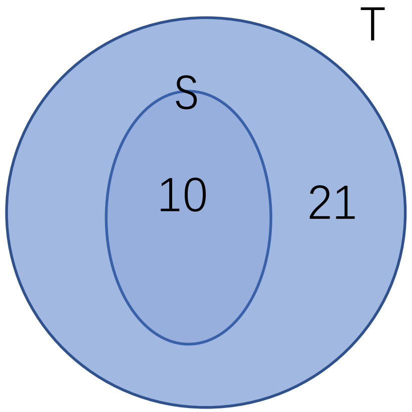
(a) Accuracy w.r.t. $|\overline{\mathcal{C}}_t|$ in task $\mathbf{A} \rightarrow \mathbf{D}$, $\xi = 0.32$.

$|\overline{\mathcal{C}}_t| = 0$



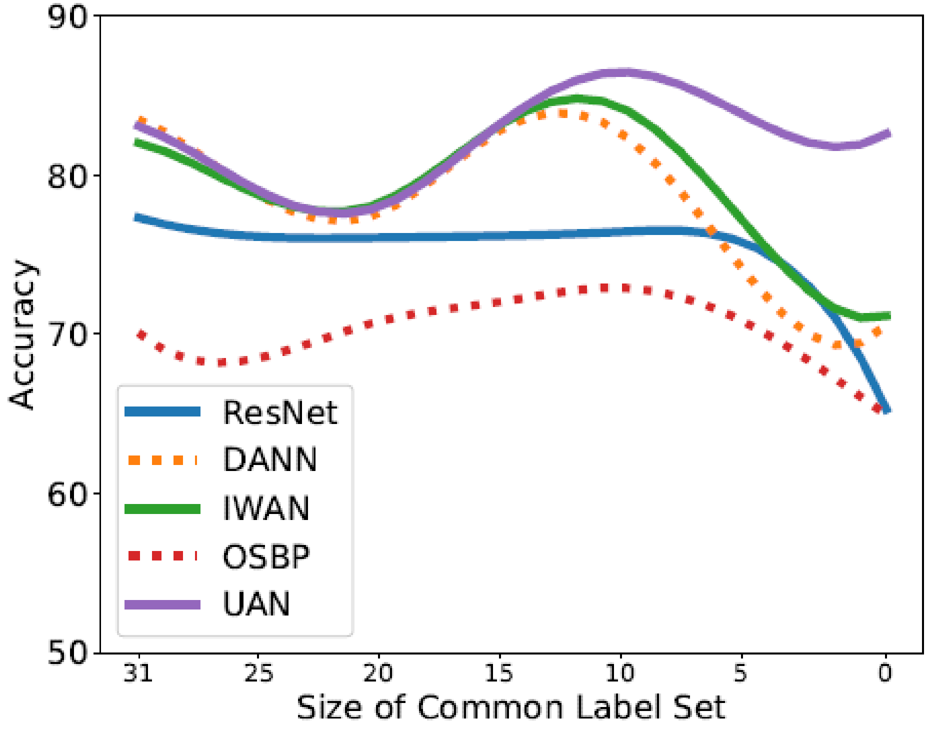


$|\overline{\mathcal{C}}_t| = 21$



Analysis on Different UDA Settings

- Varying Size of Common Label Set $\mathcal{C}$

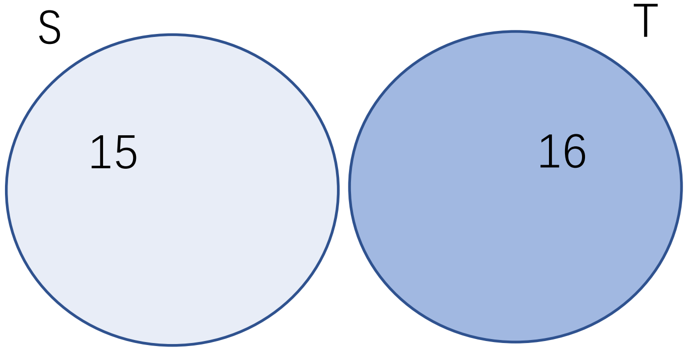


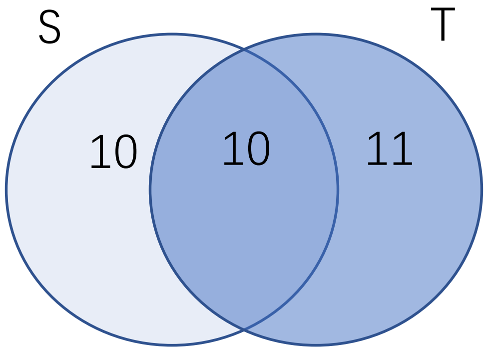
(b) Accuracy w.r.t. $|\mathcal{C}|$ in task $\mathbf{A} \rightarrow \mathbf{D}$.

$$|\mathcal{C}_t| = |\mathcal{C}_s| + 1$$

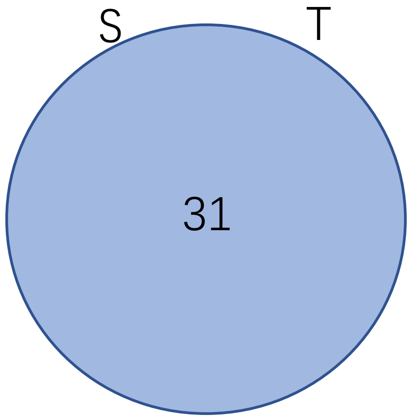
$$|\mathcal{C}| + |\overline{\mathcal{C}}_t| + |\overline{\mathcal{C}}_s| = 31$$

$|\mathcal{C}| = 0$





$|\mathcal{C}| = 31$



Analysis of Universal Adaptation Network

- Ablation Study

$$w^s(\mathbf{x}) = \frac{H(\hat{\mathbf{y}})}{\log |\mathcal{C}_s|} - \hat{d}'(\mathbf{x}) \quad (7)$$

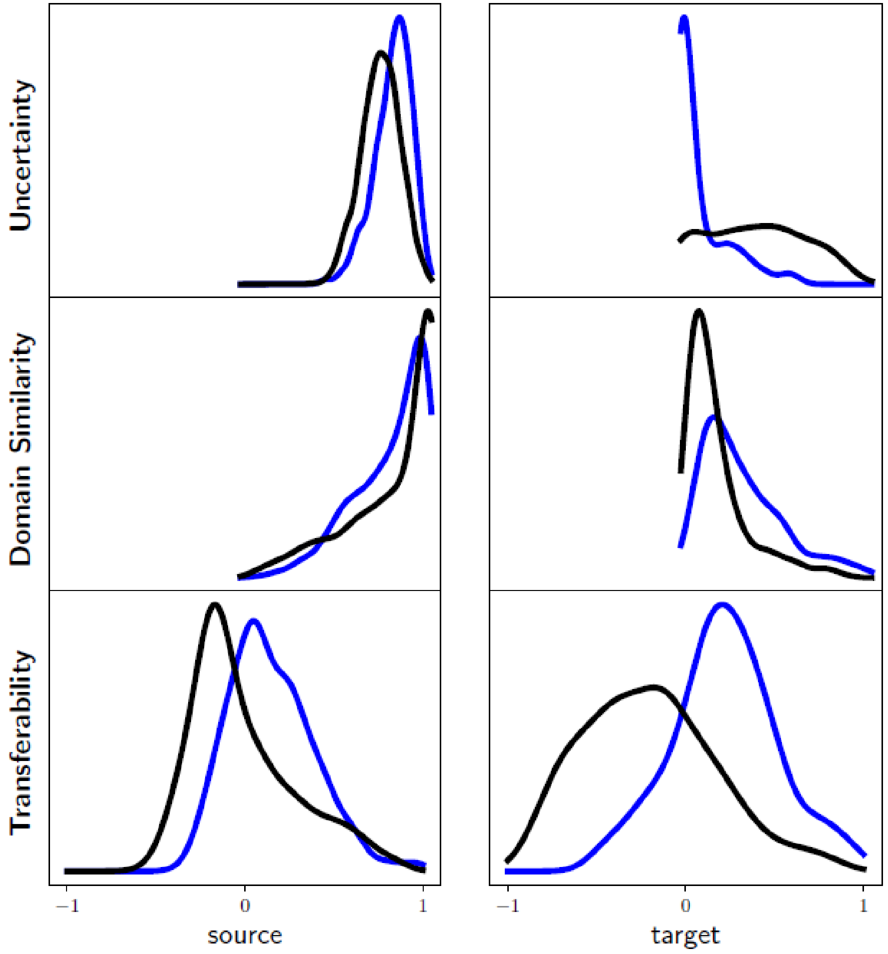
$$w^t(\mathbf{x}) = \hat{d}'(\mathbf{x}) - \frac{H(\hat{\mathbf{y}})}{\log |\mathcal{C}_s|} \quad (8)$$

Table 1. Average class accuracy (%) of universal domain adaptation tasks on **Office-Home** ($\xi = 0.15$) dataset (ResNet)

Method	Office-Home												
	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Avg
UAN w/o d	61.60	81.86	87.67	74.52	73.59	84.88	73.65	57.37	86.61	81.58	62.15	79.14	75.39
UAN w/o y	56.63	77.51	87.61	71.96	69.08	83.18	71.40	56.10	84.24	79.27	60.59	78.35	72.91
UAN	**63.00**	**82.83**	**87.85**	**76.88**	**78.70**	**85.36**	**78.22**	**58.59**	**86.80**	**83.37**	**63.17**	**79.43**	**77.02**

Analysis of Universal Adaptation Network

- Hypotheses Justification

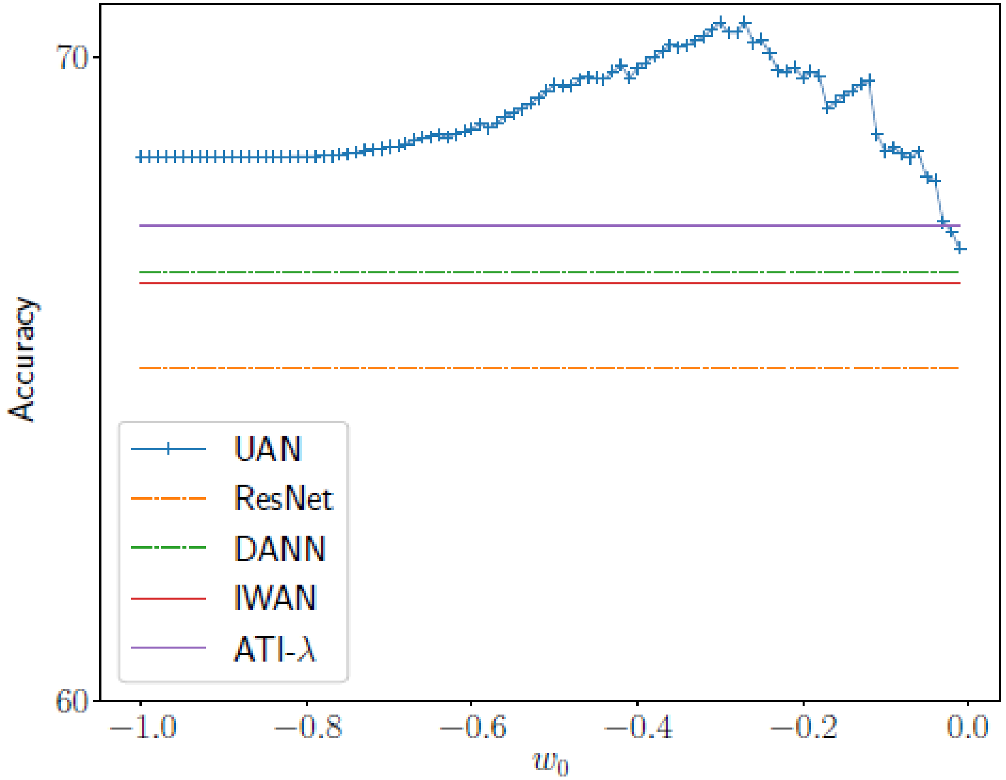


(b) Hypotheses Quality (blue for *common* and black for *private*)

$$\mathbb{E}_{\mathbf{x}\sim p_{\mathcal{C}}} w^s(\mathbf{x}) > \mathbb{E}_{\mathbf{x}\sim p_{\overline{\mathcal{C}}_s}} w^s(\mathbf{x})$$
$$\mathbb{E}_{\mathbf{x}\sim q_{\mathcal{C}}} w^t(\mathbf{x}) > \mathbb{E}_{\mathbf{x}\sim q_{\overline{\mathcal{C}}_t}} w^t(\mathbf{x}) \qquad (6)$$

Analysis of Universal Adaptation Network

- Threshold Sensitivity



(c) Sensitivity to w_0

Discussion

- UDA for not having access to target labels in unsupervised domain adaptation
- end-to-end solution
- exploits both the domain similarity and the prediction uncertainty of each sample to develop a weighting mechanism for discovering label sets shared by both domains and promote common-class adaptation
- serve as a pilot study when we encounter a new domain adaptation scenario.