清華大學
Tsinghua University

Recent Advances in Transfer Learning

Mingsheng Long
Tsinghua University
Jan 25, 2018

https://github.com/thuml

Machine Learning

Tsinghua University

Learner: $f : \boldsymbol{x} \rightarrow y$ **Distribution:** $(\boldsymbol{x}, y) \sim P(\boldsymbol{x}, y)$

fish

bird

mammal

tree

flower

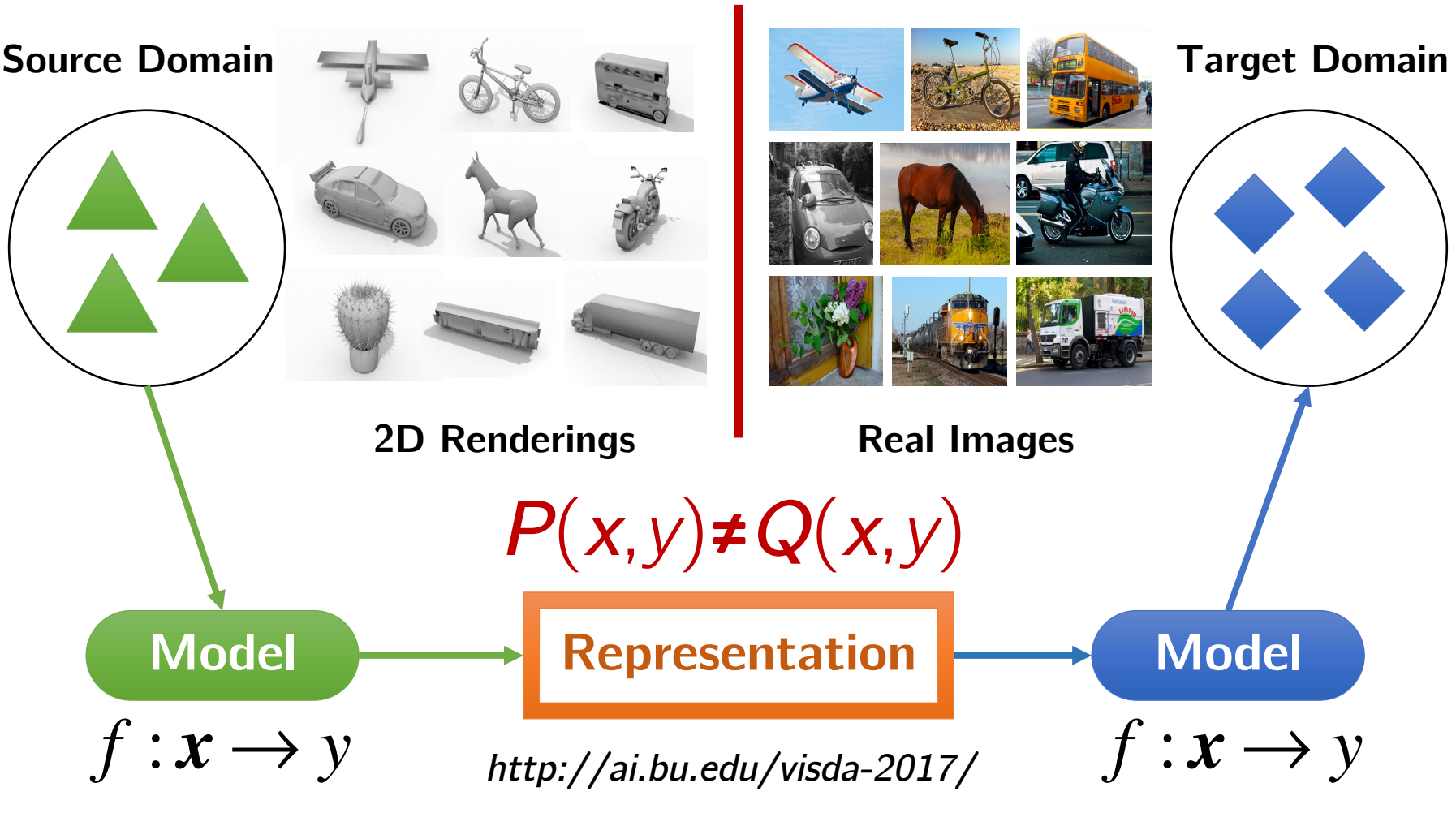
.....

Error Bound: $\epsilon_{\text{test}} \leq \hat{\epsilon}_{\text{train}} + \sqrt{\frac{\text{complexity}}{n}}$

Transfer Learning

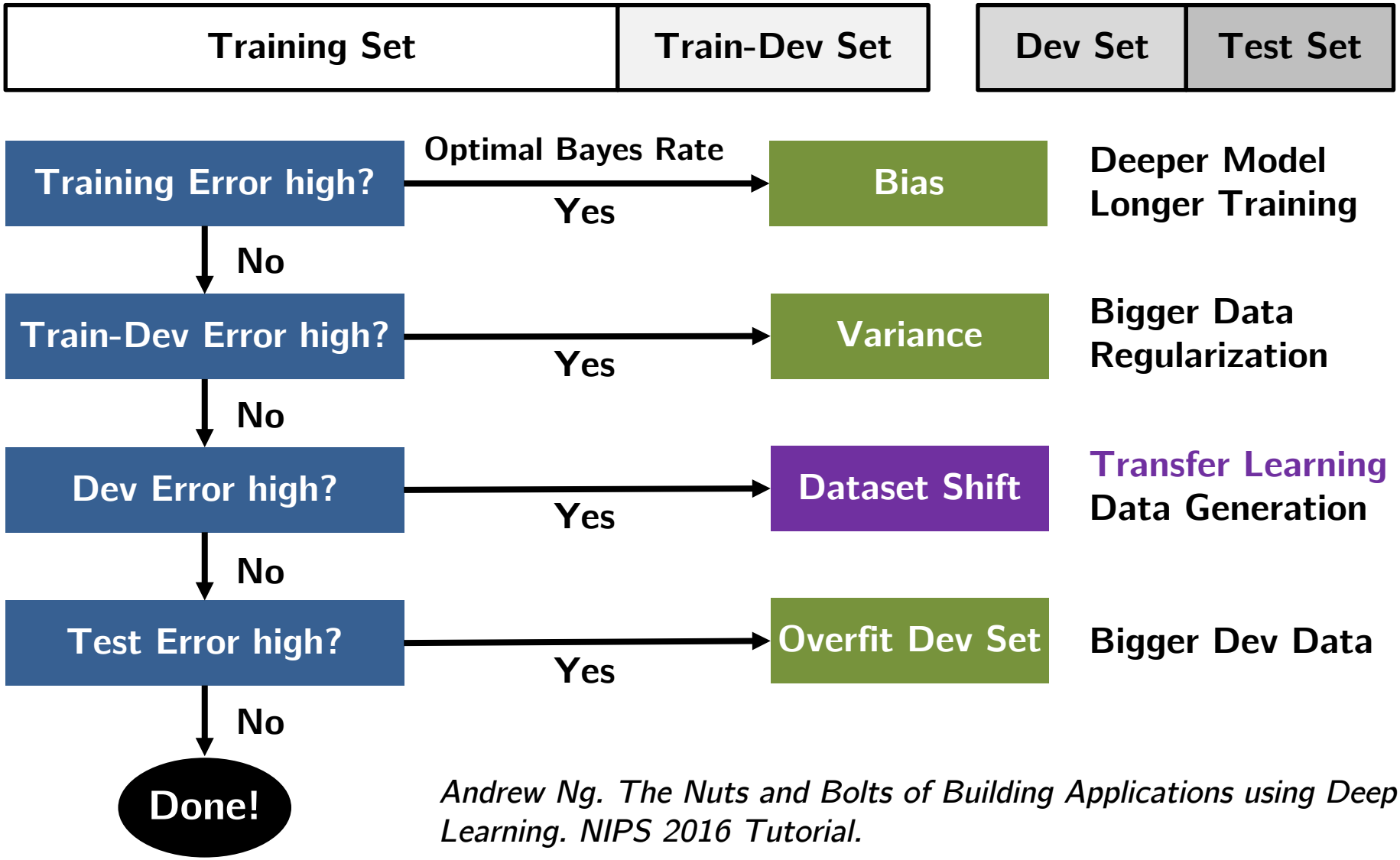
Tsinghua University

Learning across domains with **non-IID** distributions $P \neq Q$



Transfer Learning: Why?

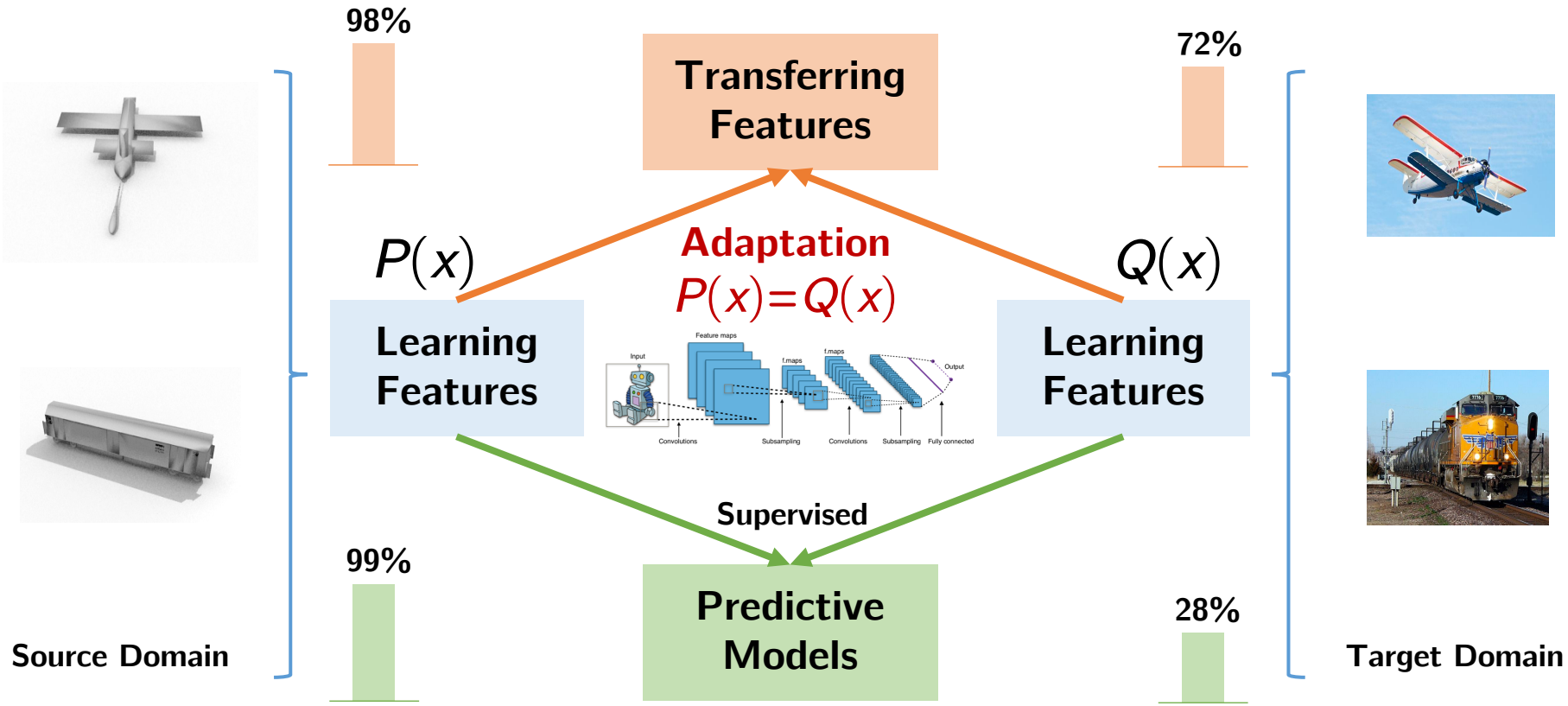
清華大學 Tsinghua University



Andrew Ng. The Nuts and Bolts of Building Applications using Deep Learning. NIPS 2016 Tutorial.

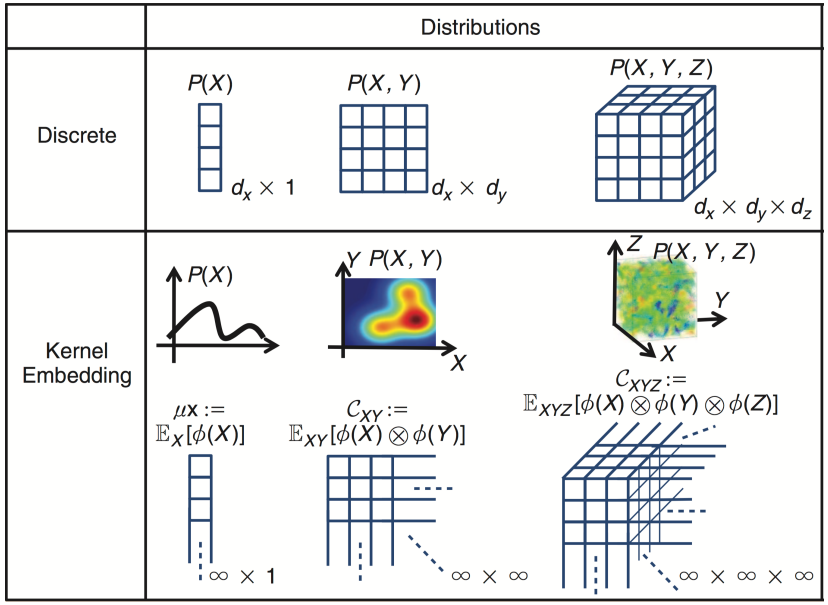
Transfer Learning: How?

- Learning predictive models on transferable features s.t. $P(\mathbf{x})=Q(\mathbf{x})$
- Distribution matching: **MMD** (ICML'15), **GAN** (ICML'15, JMLR'16)

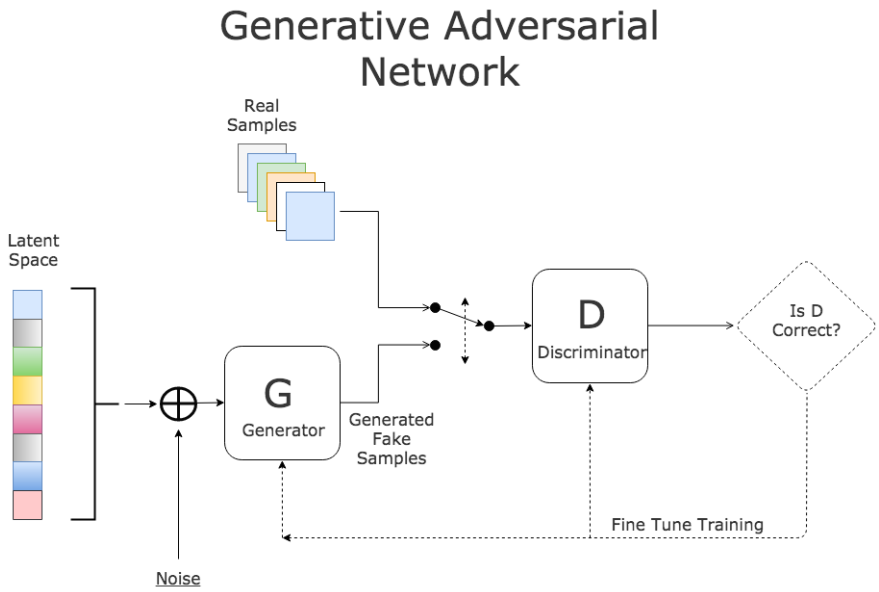


Distribution Matching

- Marginal distribution mismatch: $P(x) \neq Q(x)$
- Conditional distribution mismatch: $P(y|x) \neq Q(y|x)$



Kernel Embedding



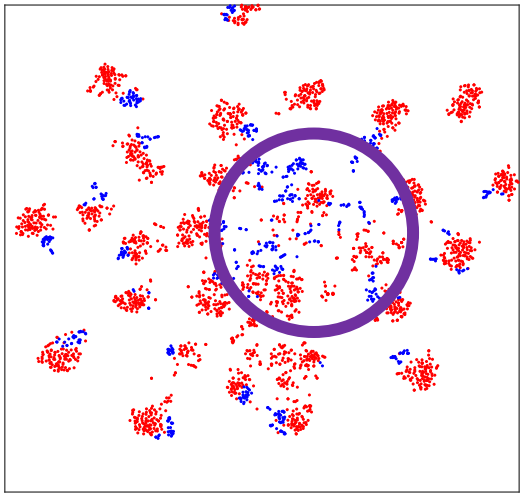
Adversarial Learning

Song et al. Kernel Embeddings of Conditional Distributions. ***IEEE****, 2013.*
Goodfellow et al. Generative Adversarial Networks. ***NIPS*** *2014.*

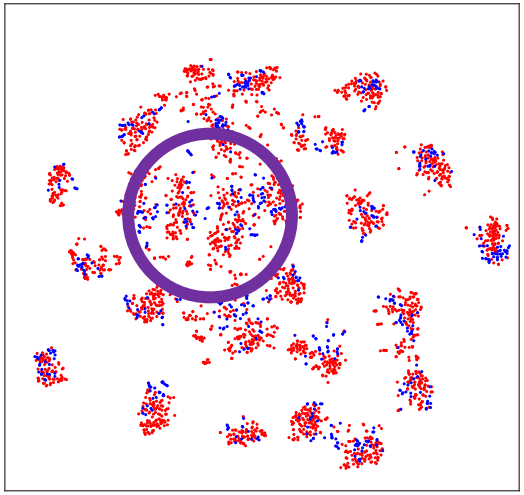
Distribution Matching

清華大學 Tsinghua University

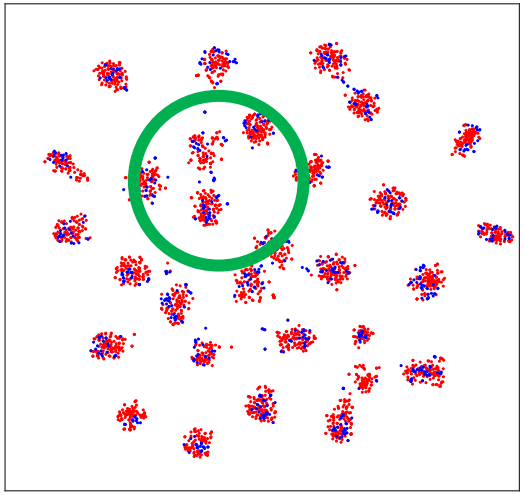
- Marginal distribution mismatch: $P(\mathbf{x}) \neq Q(\mathbf{x})$
- Conditional distribution mismatch: $P(\mathbf{y}|\mathbf{x}) \neq Q(\mathbf{y}|\mathbf{x})$



$P(\mathbf{x}) \neq Q(\mathbf{x})$
$P(\mathbf{y}|\mathbf{x}) \neq Q(\mathbf{y}|\mathbf{x})$



$P(\mathbf{x}) \approx Q(\mathbf{x})$
$P(\mathbf{y}|\mathbf{x}) \neq Q(\mathbf{y}|\mathbf{x})$



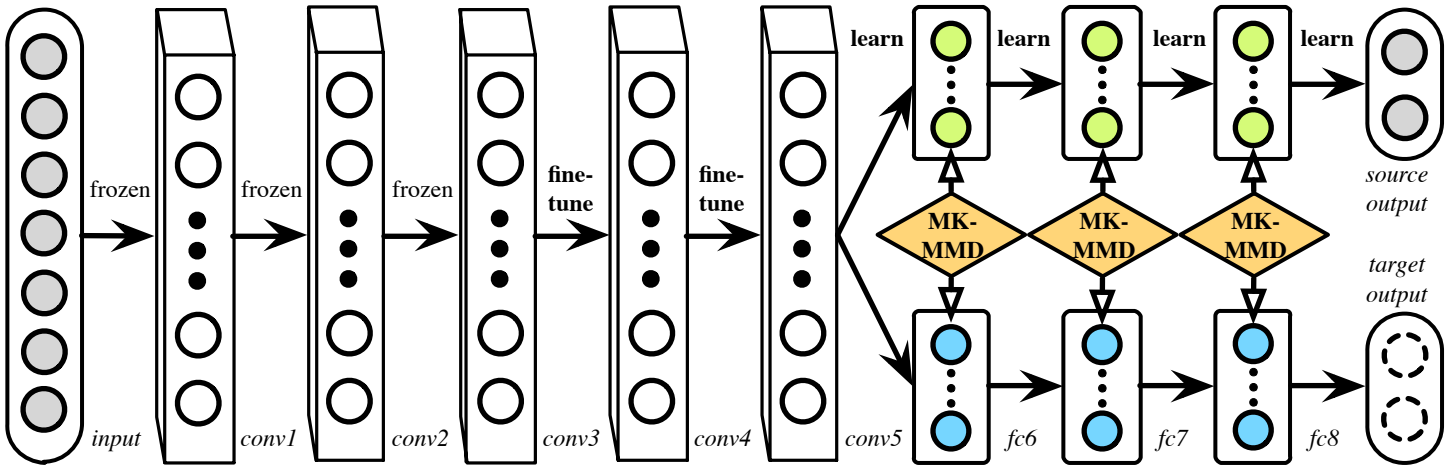
$P(\mathbf{x},\mathbf{y}) \approx Q(\mathbf{x},\mathbf{y})$
$P(\mathbf{y}|\mathbf{x}) \approx Q(\mathbf{y}|\mathbf{x})$

Problem 1

清華大學
Tsinghua University

$$P(x) \neq Q(x)$$

Deep Adaptation Network (DAN)



- Deep adaptation: match distributions in multiple domain-specific layers
- Optimal matching: maximize two-sample test power by multiple kernels

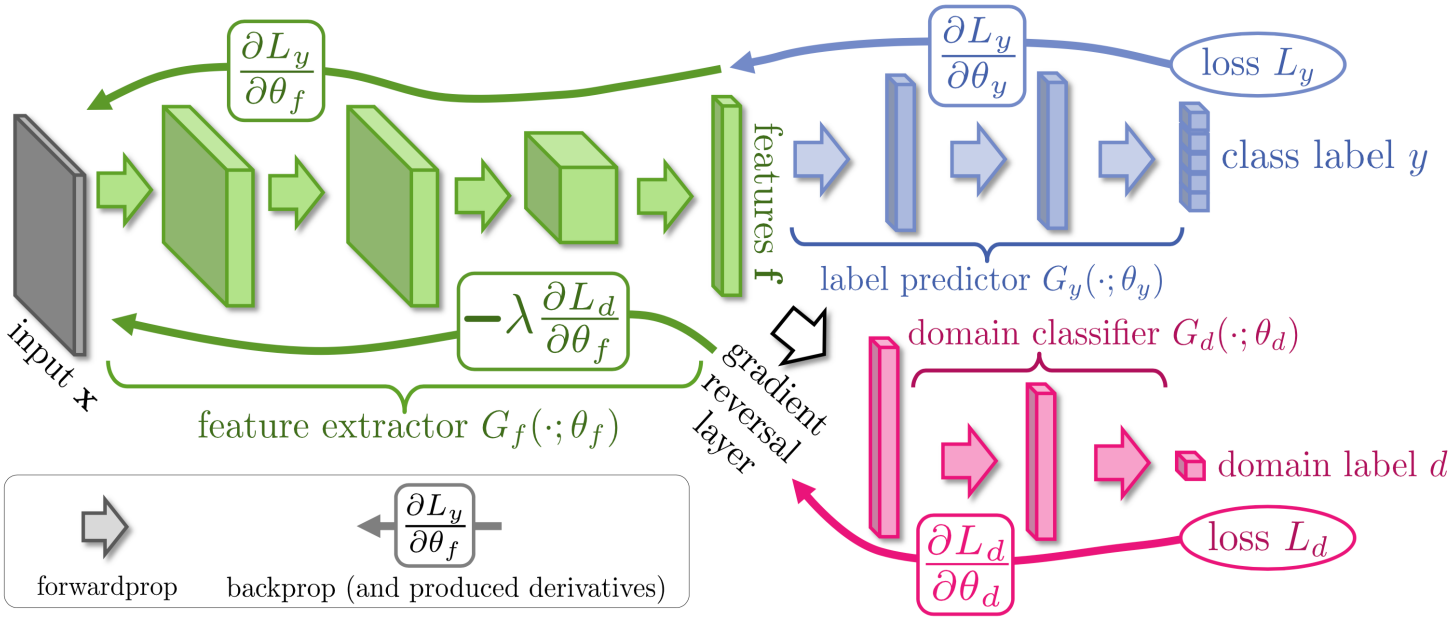
$$d_k^2\left(P,Q\right) \triangleq \left\| \mathbf{E}_P\left[\phi\left(\mathbf{x}^s\right)\right] - \mathbf{E}_Q\left[\phi\left(\mathbf{x}^t\right)\right] \right\|_{\mathcal{H}_k}^2$$

$$\min_{f\in\mathcal{F}} \max_{k\in\mathcal{K}} \frac{1}{n_a}\sum_{i=1}^{n_a} J\left(f\left(\mathbf{x}_i^a\right), y_i^a\right) + \lambda \sum_{\ell=l_1}^{l_2} d_k^2\left(\mathcal{D}_s^\ell, \mathcal{D}_t^\ell\right)$$

M. Long et al. Learning Transferable Features with Deep Adaptation Networks. ***ICML****, 2015.*

Domain Adversarial Training (DANN)

Tsinghua University



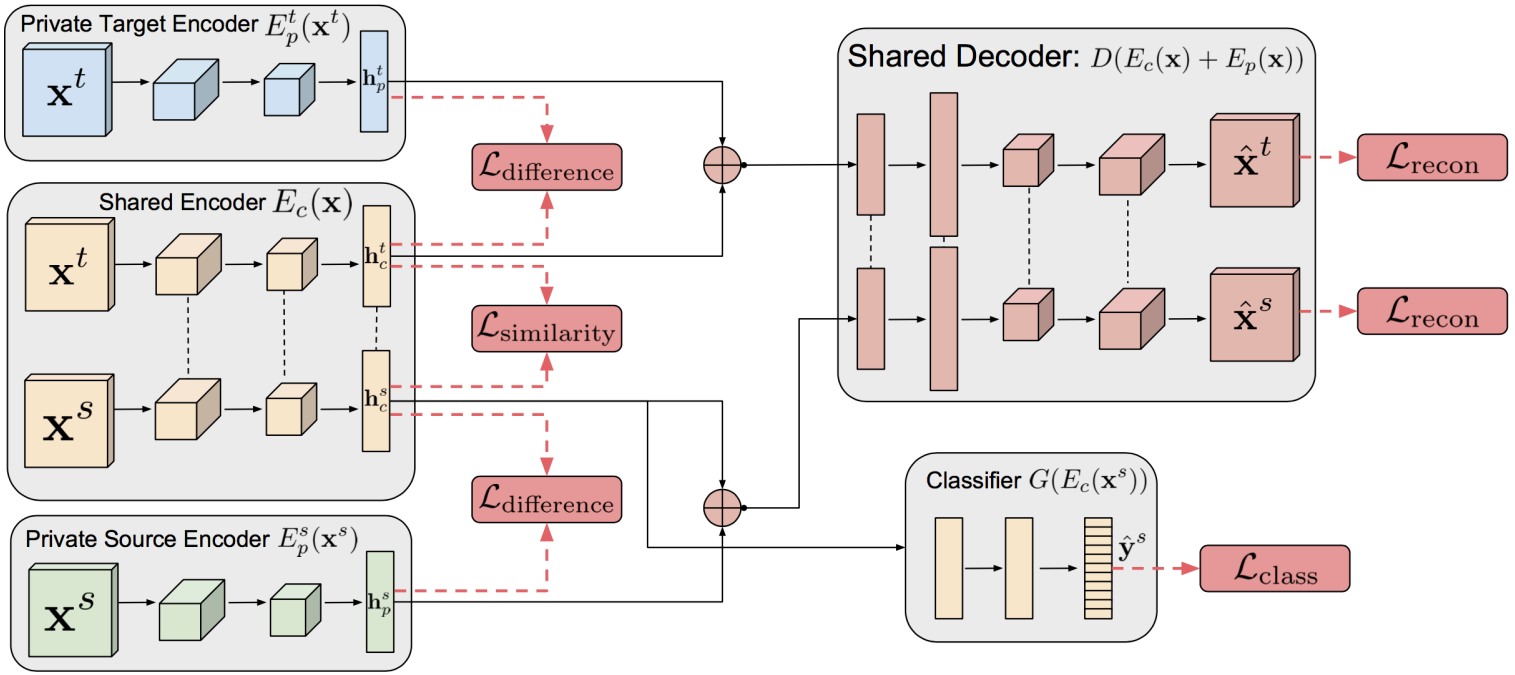
- Adversarial adaptation: learning features indistinguishable across domains

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

$$\left(\hat{\theta}_f,\hat{\theta}_y\right)=\arg\min_{\theta_f,\theta_y}E\left(\theta_f,\theta_y,\theta_d\right)\quad\left(\hat{\theta}_d\right)=\arg\max_{\theta_d}E\left(\theta_f,\theta_y,\theta_d\right)$$

Y Ganin et al. Unsupervised Domain Adaptation by Backpropagation. ***ICML****, 2015.*

Domain Separation Network (DSN)



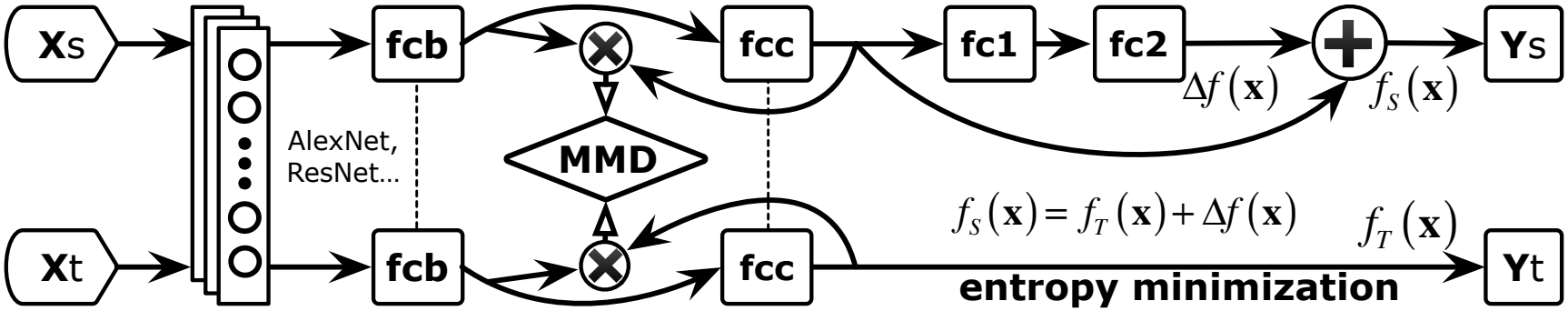
$$\hat{\mathbf{x}} = D\left(E_c\left(\mathbf{x}\right) + E_p\left(\mathbf{x}\right)\right) \quad \hat{\mathbf{y}} = G\left(E_c\left(\mathbf{x}\right)\right)$$

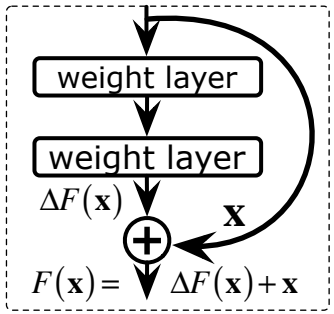
$$L = L_{task} + \alpha L_{recon} + \beta L_{diff} + \gamma L_{sim}$$

$$L_{diff} = \left\| \mathbf{H}_c^{s\mathsf{T}} \mathbf{H}_p^s \right\|_F^2 + \left\| \mathbf{H}_c^{t\mathsf{T}} \mathbf{H}_p^t \right\|_F^2$$

K. Bousmalis et al. Domain Separation Networks. ***NIPS****, 2016.*

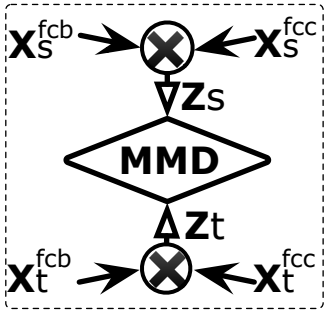
Residual Transfer (RTN)





Classifier Adaptation

$$\min_{f_S = f_T + \Delta f} \frac{1}{n_s} \sum_{i=1}^{n_s} L\left(f_s\left(\mathbf{x}_i^s\right), y_i^s\right)$$



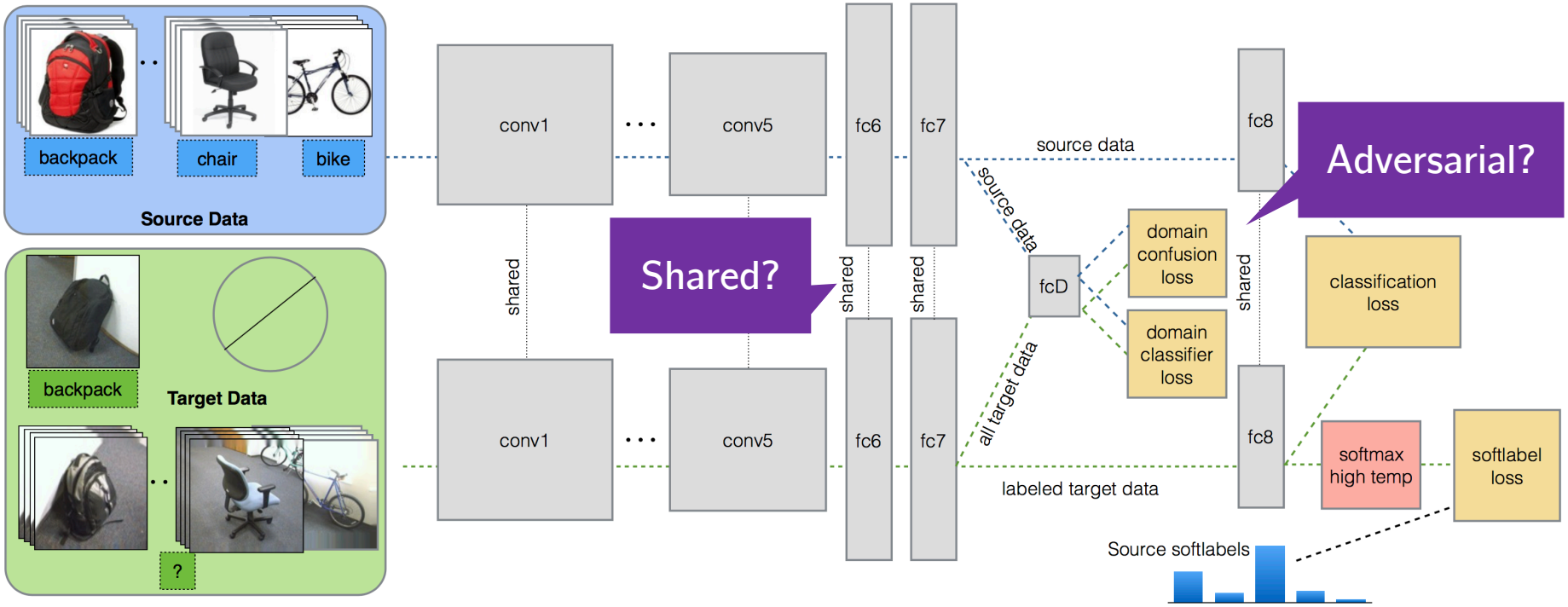
Feature Adaptation

$$+ \frac{\gamma}{n_t} \sum_{i=1}^{n_t} H\left(f_t\left(\mathbf{x}_i^t\right)\right)$$

$$+ \lambda\, D_{\mathcal{L}}\left(\mathcal{D}_s, \mathcal{D}_t\right),$$

M. Long et al. Unsupervised Domain Adaptation with Residual Transfer Networks. ***NIPS****, 2016.*

Asymmetric Transfer (ADDA)



$$\min_D L_{adv_D}(\mathbf{X}_s, \mathbf{X}_t, M_s, M_t)$$
$$= -\mathbf{E}_{\mathbf{x}_s}\left[\log D\left(M_s(\mathbf{x}_s)\right)\right]$$
$$-\mathbf{E}_{\mathbf{x}_t}\left[\log\left(1 - D\left(M_t(\mathbf{x}_t)\right)\right)\right]$$

$$\min_{M_s, M_t} L_{adv_M}(\mathbf{X}_s, \mathbf{X}_t, D)$$
$$= -\mathbf{E}_{\mathbf{x}_t}\left[\log D\left(M_t(\mathbf{x}_t)\right)\right]$$

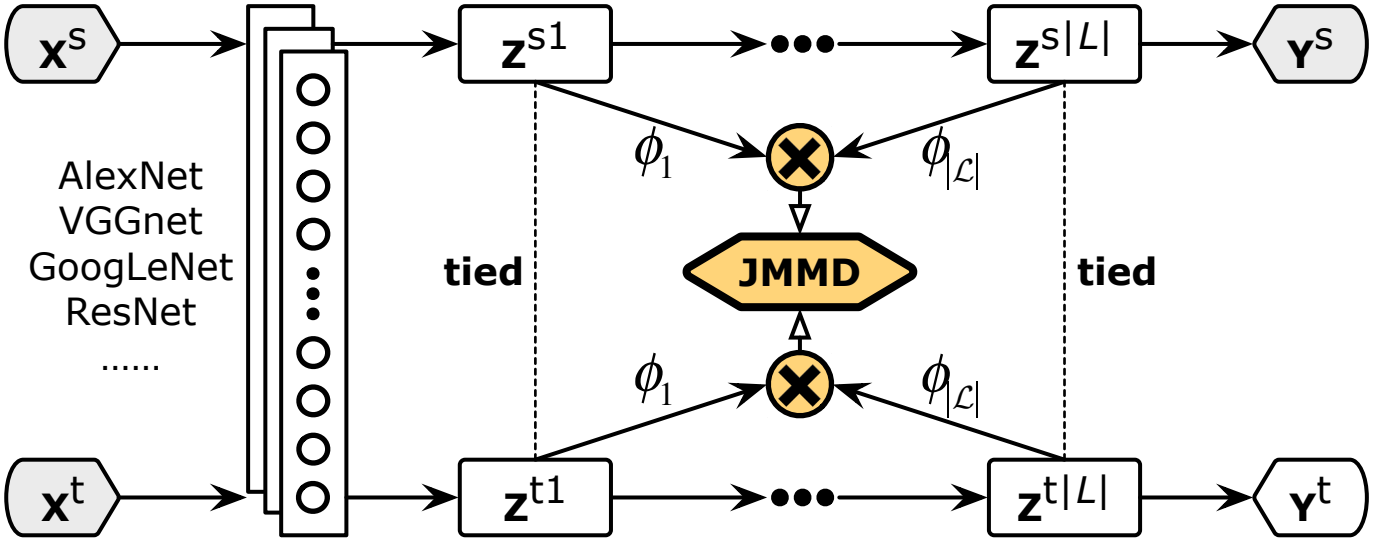
Asymmetric

E. Tzeng et al. Adversarial Discriminative Domain Adaptation. ***CVPR****, 2017.*

Problem 2

$$P(x,y) \neq Q(x,y)$$

Joint Adaptation Network (JAN)

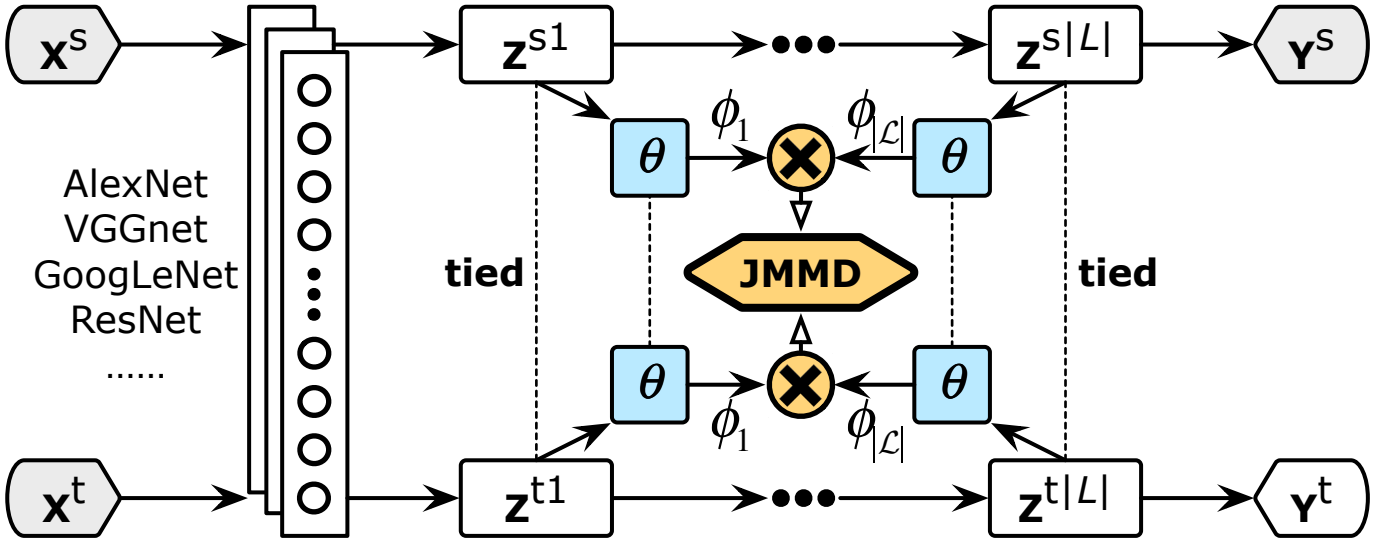


$$\min_f \frac{1}{n_s}\sum_{i=1}^{n_s} J\left(f\left(\mathbf{x}_i^s\right),\mathbf{y}_i^s\right)+\lambda\hat{D}_{\mathcal{L}}\left(P,Q\right)$$

$$D_{\mathcal{L}} \triangleq \left\|\mathbf{E}_P\left[\otimes_{\ell=1}^{|\mathcal{L}|}\phi^{\ell}\left(\mathbf{z}^{s\ell}\right)\right]-\mathbf{E}_Q\left[\otimes_{\ell=1}^{|\mathcal{L}|}\phi^{\ell}\left(\mathbf{z}^{t\ell}\right)\right]\right\|^2_{\otimes_{\ell=1}^{|\mathcal{L}|}\mathcal{H}^{\ell}}$$

M. Long et al. Deep Transfer Learning with Joint Adaptation Networks. ***ICML****, 2017.*

Adversarial JAN

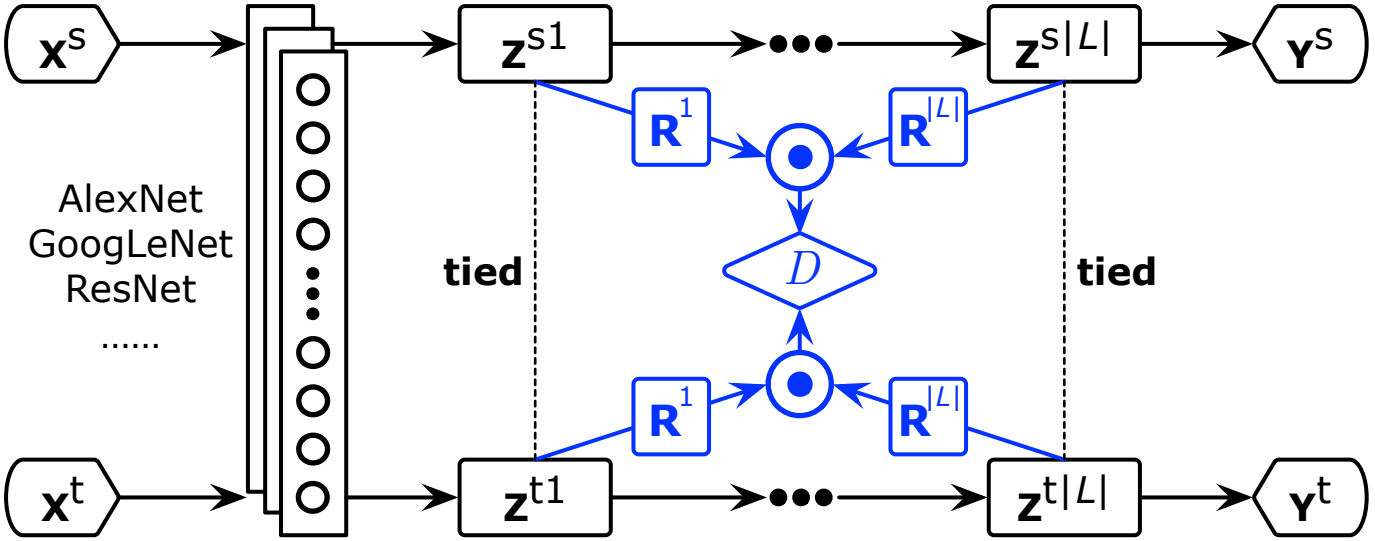


$$\min_f \frac{1}{n_s}\sum_{i=1}^{n_s} J\left(f\left(\mathbf{x}_i^s\right),\mathbf{y}_i^s\right)+\lambda \hat{D}_{\mathcal{L}}\left(P,Q;\theta\right)$$

$$D_{\mathcal{L}} \triangleq \left\| \mathbf{E}_P\left[\otimes_{\ell=1}^{|\mathcal{L}|}\phi^{\ell}\left(\theta^{\ell}\left(\mathbf{z}^{s\ell}\right)\right)\right]-\mathbf{E}_Q\left[\otimes_{\ell=1}^{|\mathcal{L}|}\phi^{\ell}\left(\theta^{\ell}\left(\mathbf{z}^{t\ell}\right)\right)\right]\right\|^2_{\otimes_{\ell=1}^{|\mathcal{L}|}\mathcal{H}^{\ell}}$$

M. Long et al. Deep Transfer Learning with Joint Adaptation Networks. ***ICML****, 2017.*

Multilinear Adversarial Network (MAN)



$$\phi_{\mathcal{L}}\left(\mathbf{z}_i^s\right)=\frac{1}{\sqrt{d}}\left(\odot_{\ell}^{|\mathcal{L}|}\mathbf{R}^{\ell}\mathbf{z}_i^{s\ell}\right),\phi_{\mathcal{L}}\left(\mathbf{z}_j^t\right)=\frac{1}{\sqrt{d}}\left(\odot_{\ell}^{|\mathcal{L}|}\mathbf{R}^{\ell}\mathbf{z}_j^{t\ell}\right)$$

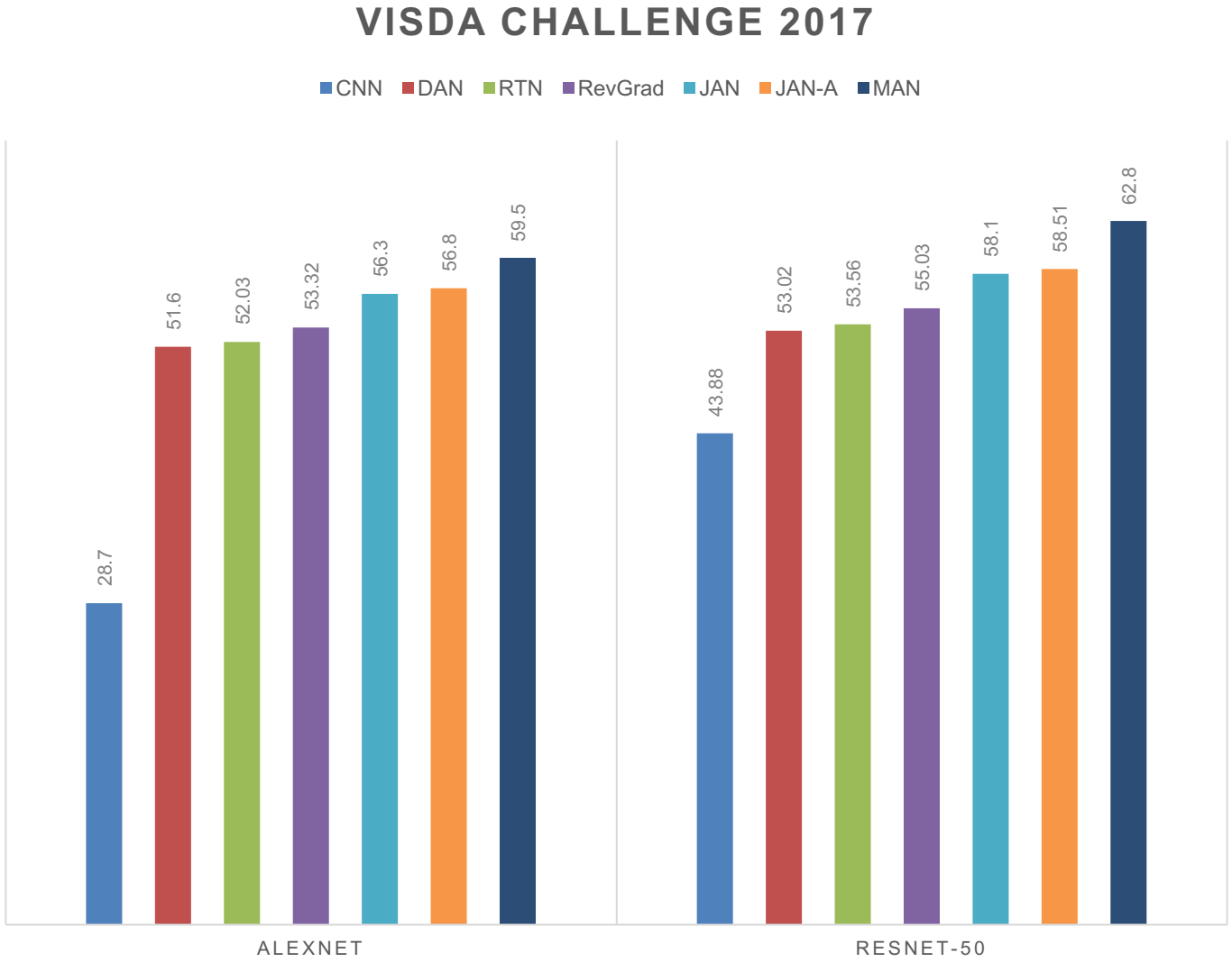
$$\min_F \frac{1}{n_s}\sum_{i=1}^{n_s}J\left(F\left(\mathbf{x}_i^s\right),\mathbf{y}_i^s\right)+\frac{\lambda}{n_s}\sum_{i=1}^{n_s}\log D\left(\phi_{\mathcal{L}}\left(\mathbf{z}_i^s\right)\right)+\frac{\lambda}{n_t}\sum_{j=1}^{n_t}\log\left(1-D\left(\phi_{\mathcal{L}}\left(\mathbf{z}_j^t\right)\right)\right)$$

$$\min_D -\frac{1}{n_s}\sum_{i=1}^{n_s}\log D\left(\phi_{\mathcal{L}}\left(\mathbf{z}_i^s\right)\right)-\frac{1}{n_t}\sum_{j=1}^{n_t}\log\left(1-D\left(\phi_{\mathcal{L}}\left(\mathbf{z}_j^t\right)\right)\right)$$

Empirical Benchmark

Source Domain

Target Domain



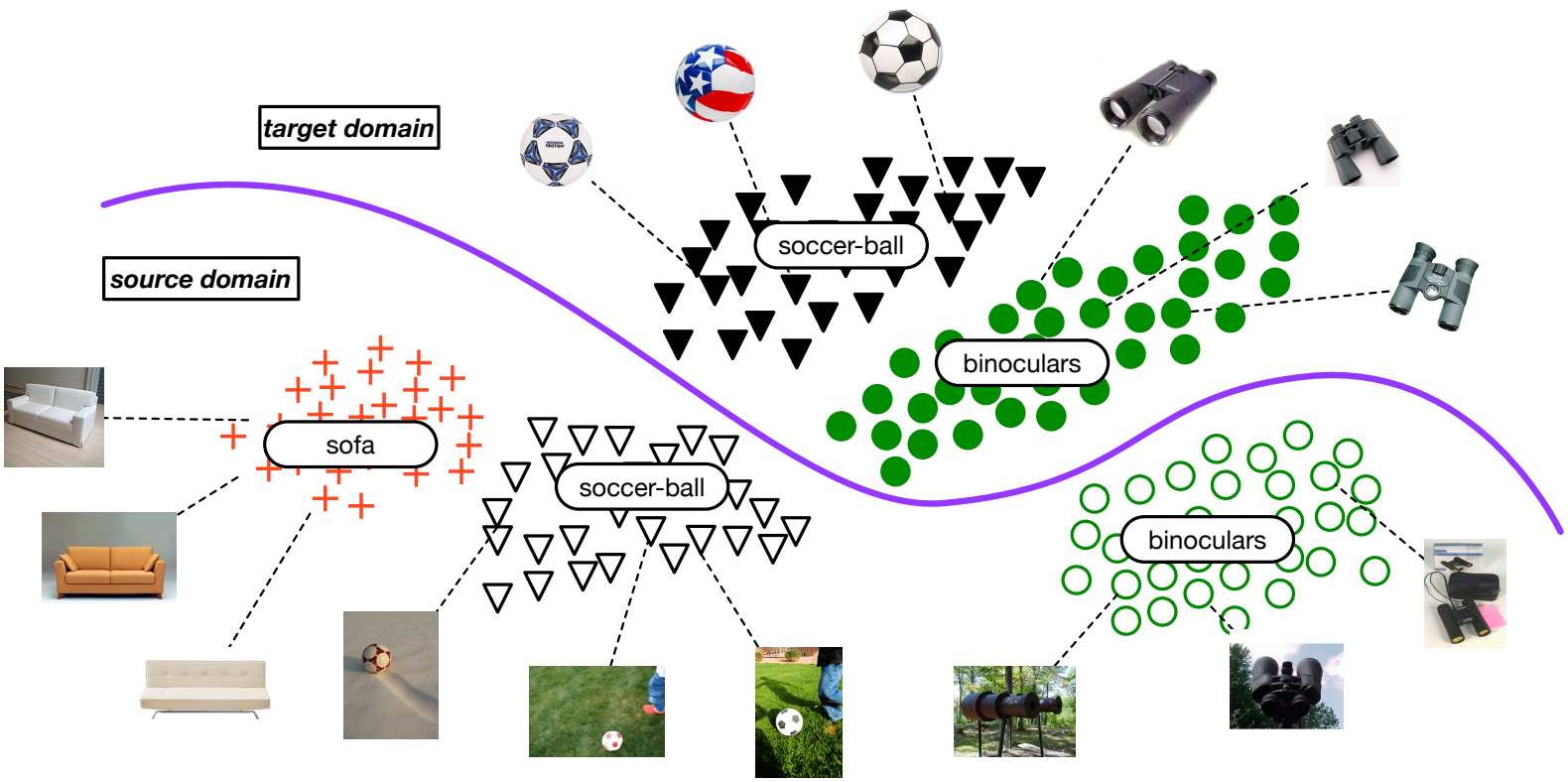
Problem 3

清華大學 Tsinghua University

$$\mathbf{Y}_s \neq \mathbf{Y}_t$$

Partial Transfer Learning

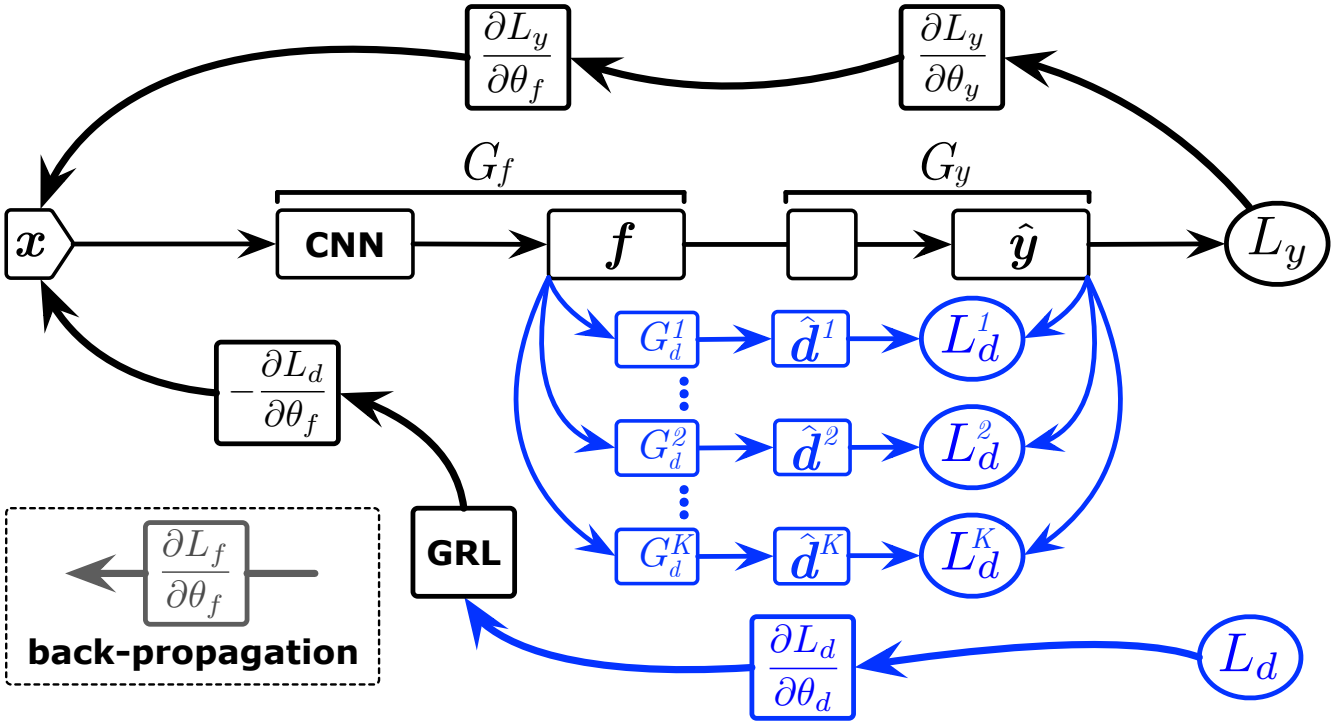
Tsinghua University



$$\mathbf{Y}_s \supset \mathbf{Y}_t$$

Selective Adversarial Network (SAN)

Tsinghua University



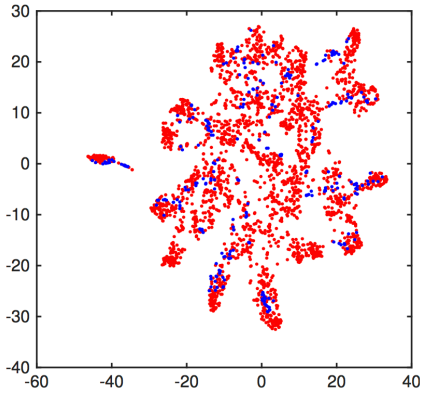
$$C\left(\theta_f,\theta_y,\theta_d^k\ |_{k=1}^{|\mathcal{C}_s|}\right)=\frac{1}{n_s}\sum_{\mathbf{x}_i\in\mathcal{D}_s}L_y\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right),y_i\right)+\frac{1}{n_t}\sum_{\mathbf{x}_i\in\mathcal{D}_t}H\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right)\right)$$

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

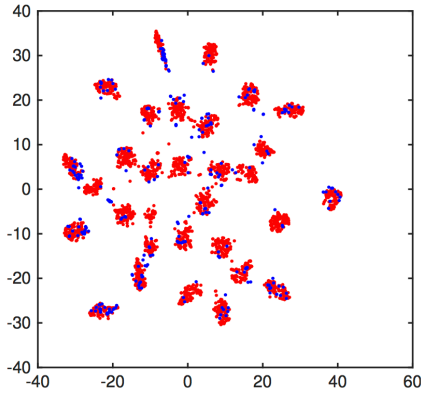
Selective Adversarial Network (SAN)

Tsinghua University

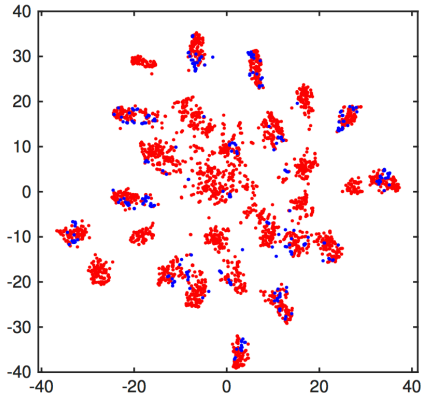
Method	Caltech-Office				ImageNet-Caltech		
	C 256 → W 10	C 256 → A 10	C 256 → D 10	Avg	I 1000 → C 84	C 256 → I 84	Avg
AlexNet [14]	58.44	76.64	65.86	66.98	52.37	47.35	49.86
DAN [15]	42.37	70.75	47.04	53.39	54.21	52.03	53.12
RevGrad [6]	54.57	72.86	57.96	61.80	51.34	47.02	49.18
RTN [17]	71.02	81.32	62.35	71.56	63.69	50.45	57.07
ADDA [26]	73.66	78.35	74.80	75.60	64.20	51.55	57.88
SAN-selective	76.44	81.63	80.25	79.44	66.78	51.25	59.02
SAN-entropy	72.54	78.95	76.43	75.97	55.27	52.31	53.79
SAN	**88.33**	**83.82**	**85.35**	**85.83**	**68.45**	**55.61**	**62.03**



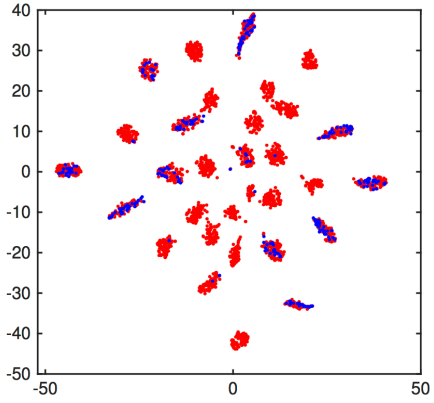
(a) DAN



(b) RevGrad



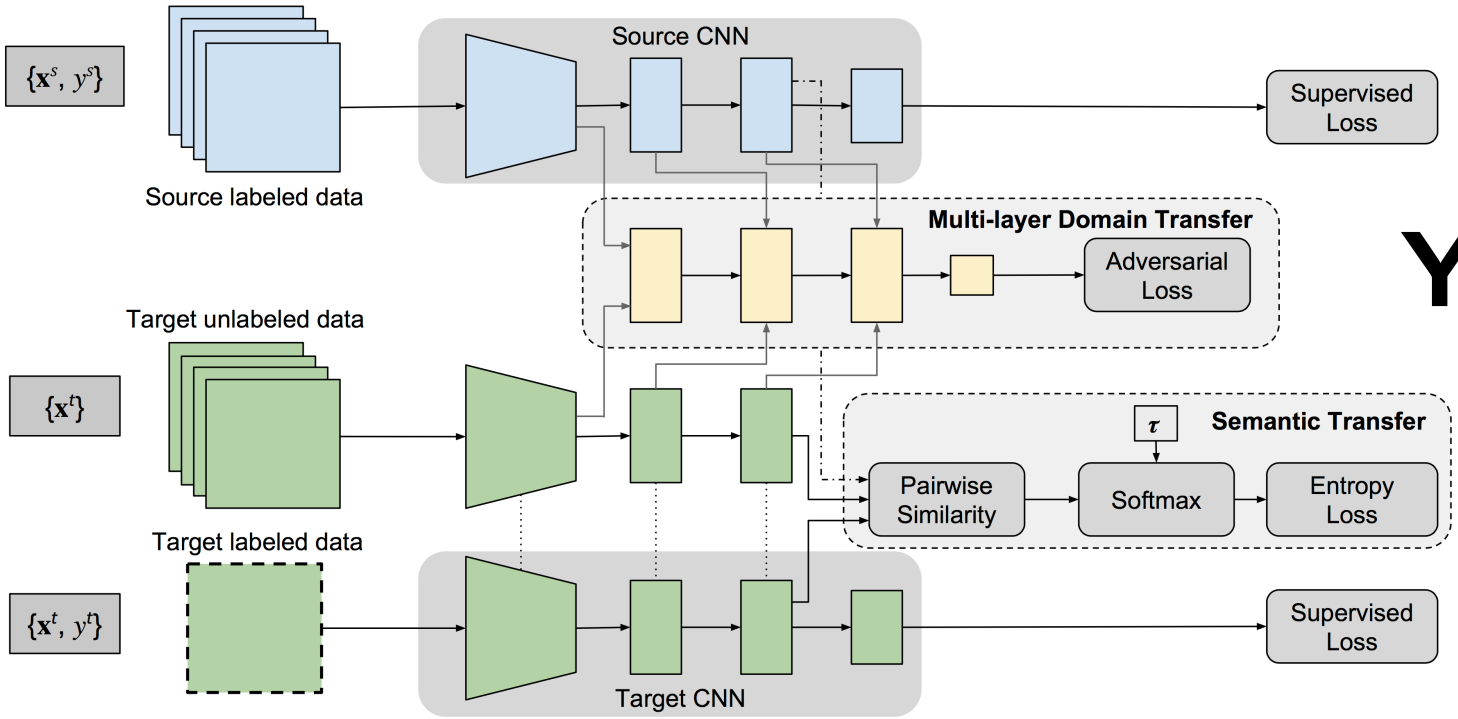
(c) RTN



(d) SAN

Z. Cao et al. Partial Transfer Learning with Selective Adversarial Networks. ***arXiv****, 2017.*

Joint Domain and Semantic Transfer



$$\mathbf{Y}_s \neq \mathbf{Y}_t$$

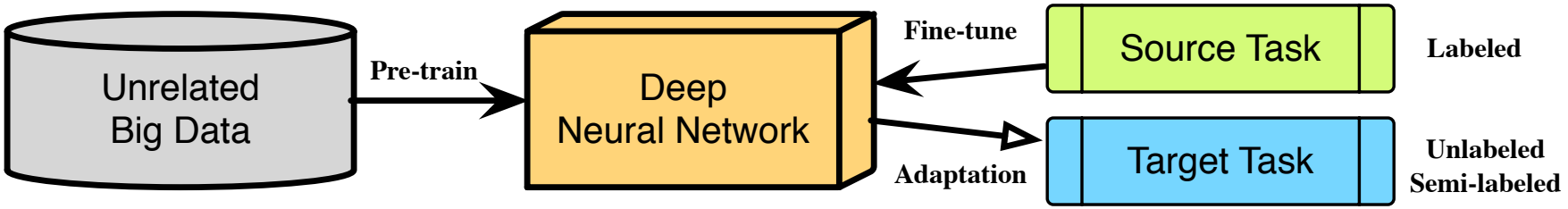
$$L_{ST}\left(\tilde{\mathbf{X}}_t, \mathbf{X}_s\right) = \sum_{\tilde{\mathbf{x}}_t \in \tilde{\mathbf{X}}_t} H\left(\sigma\left(v_s\left(\tilde{\mathbf{x}}_t\right)/\tau\right)\right) \qquad L_{ST,upsup}\left(\tilde{\mathbf{X}}_t, \mathbf{X}_t\right) = \sum_{\tilde{\mathbf{x}}_t \in \tilde{\mathbf{X}}_t} H\left(\sigma\left(v_t\left(\tilde{\mathbf{x}}_t\right)/\tau\right)\right)$$

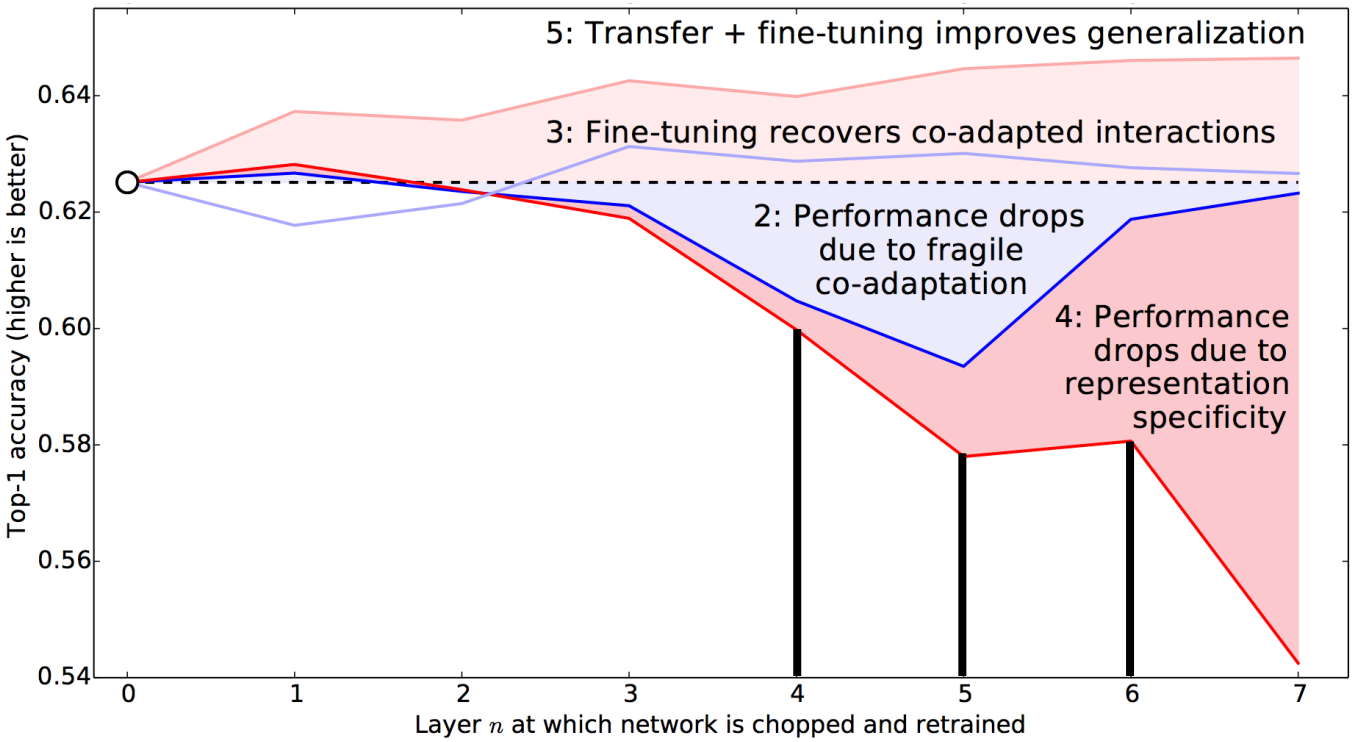
$$L_{ST,sup}\left(\mathbf{X}_t\right) = -\sum_{\{\mathbf{x}_s, \mathbf{x}_t\} \in \mathbf{X}_t} \log \frac{\exp\left(\left[v_t\left(\mathbf{x}_t\right)\right]_{\mathbf{y}_t}\right)}{\sum_{i=1}^{n} \exp\left(\left[v_t\left(\mathbf{x}_t\right)\right]_i\right)}$$

Z. Luo et al. Label Efficient Learning of Transferable Representations across Domains and Tasks. ***NIPS****, 2017.*

Transferable Architecture

Transferability

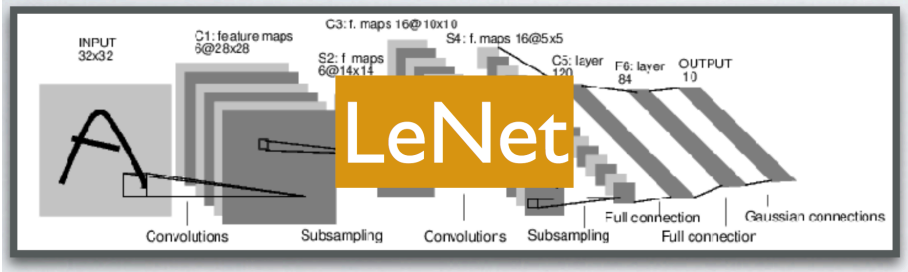




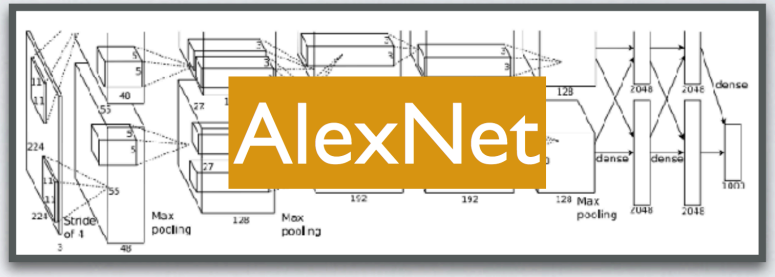
J. Yosinski et al. How Transferable Are Features in Deep Neural Networks?. ***NIPS****, 2014.*

Transferable Architecture

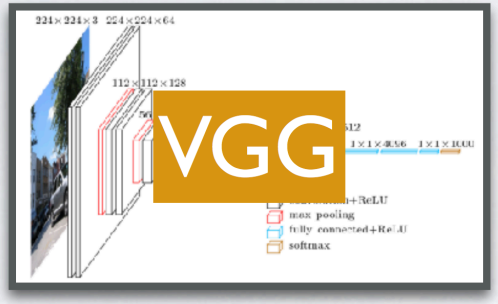
Tsinghua University



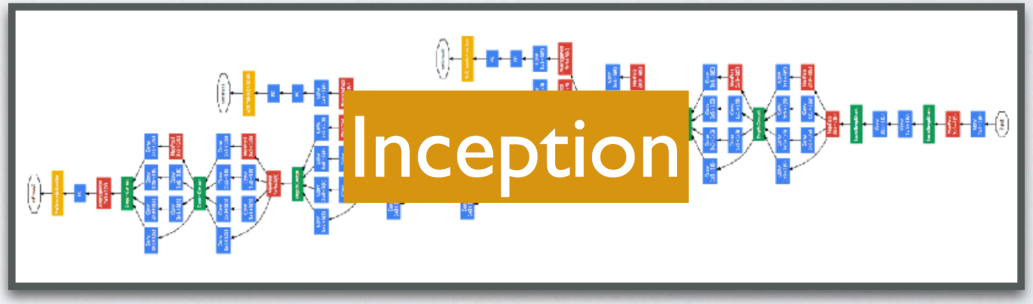
LeNet



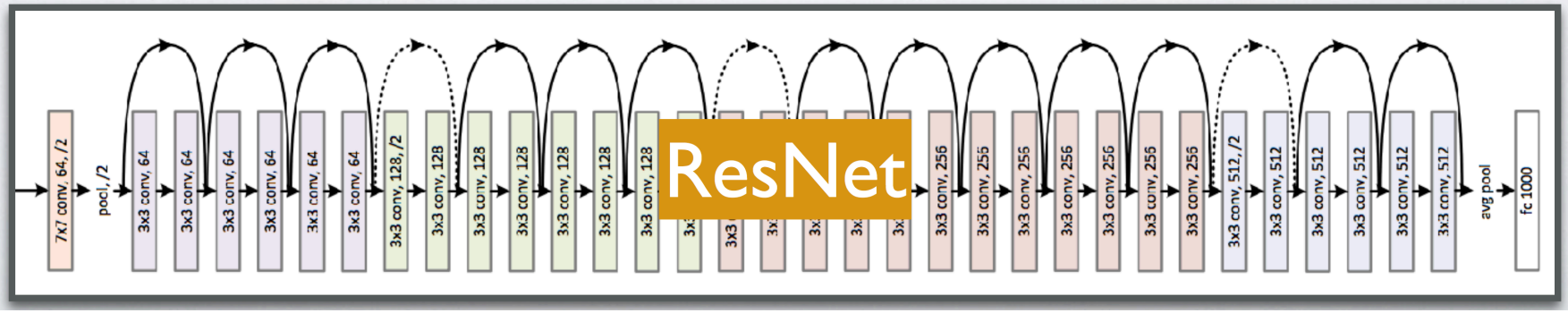
AlexNet



VGG



Inception



ResNet

Some modules may not influence in-domain accuracy but influence the transferability

Open Problems

Tsinghua University

- Heterogeneous Transfer Learning

$$\mathbf{X}_s \neq \mathbf{X}_t \ \wedge \ \mathbf{Y}_s \neq \mathbf{Y}_t$$

- Pixel-Level Transfer Learning

$$\mathbf{P(x) \neq Q(x)} \ \wedge \ \mathbf{P(z) \neq Q(z)}$$

- Learning Transferable Architectures

Thank You!