Partial Transfer Learning with Selective Adversarial Networks

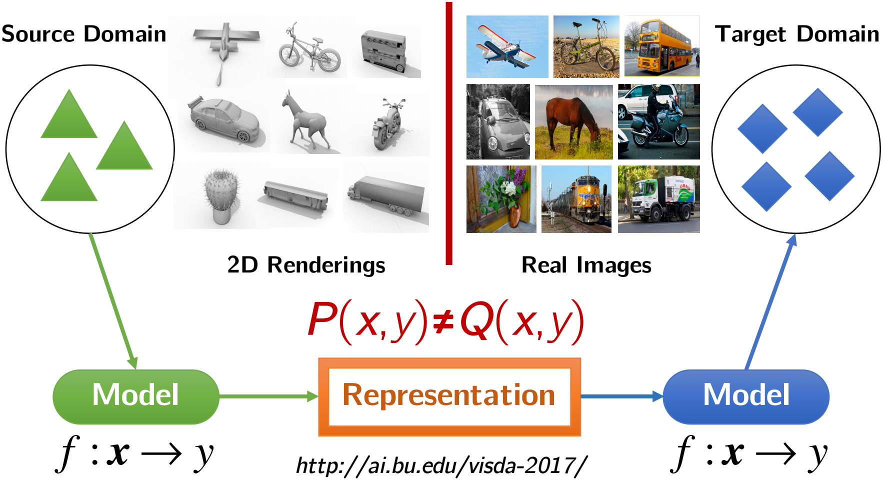
Zhangjie Cao[1], Mingsheng Long[1], Jianmin Wang[1], and Michael I. Jordan[2]

[1]KLiss, MOE; School of Software, Tsinghua University, China
[1]National Engineering Laboratory for Big Data Software
[2]University of California, Berkeley, Berkeley, CA, USA

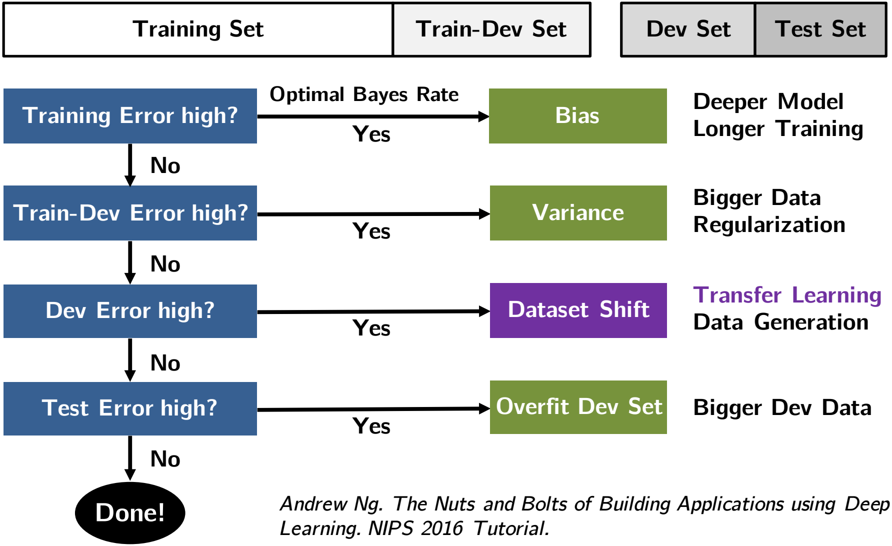
IEEE Conference on Computer Vision and Pattern Recognition
CVPR 2018 (Spotlight)

Deep Transfer Learning

- Deep learning across domains of different distributions $P \neq Q$



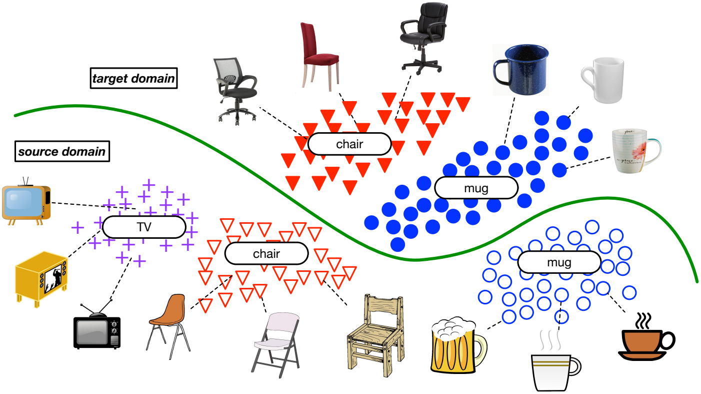
Deep Transfer Learning: Why?



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

Partial Transfer Learning

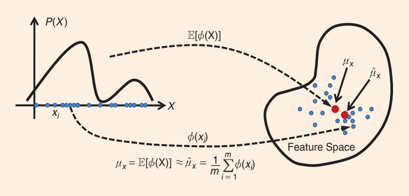
- Deep learning across domains with different label spaces $\mathcal{C}_s \supset \mathcal{C}_t$
- Positive transfer across domains in shared label space $P_{\mathcal{C}_t} \neq Q_{\mathcal{C}_t}$
- Negative transfer across domains in outlier label space $P_{\mathcal{C}_s \backslash \mathcal{C}_t} \neq Q_{\mathcal{C}_t}$



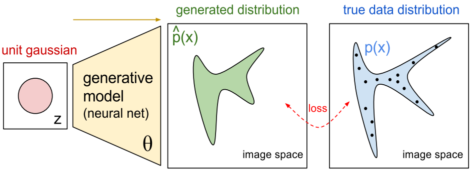
Partial Transfer Learning: How?

Matching distributions across the source and target domains s.t. $P \approx Q$

- Reduce marginal distribution mismatch: $P(\mathbf{X}) \neq Q(\mathbf{X})$
- Reduce conditional distribution mismatch: $P(Y|\mathbf{X}) \neq Q(Y|\mathbf{X})$



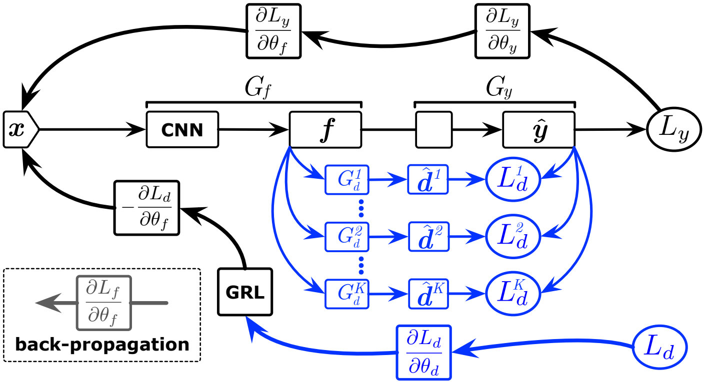
Kernel Embedding



Adversarial Learning

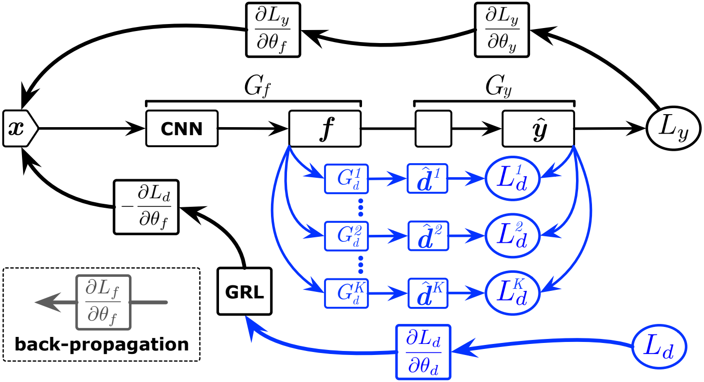
Song et al. Kernel Embeddings of Conditional Distributions. ***IEEE****, 2013.*
Goodfellow et al. Generative Adversarial Networks. ***NIPS*** *2014.*

Selective Adversarial Networks



- $\mathbf{f} = G_f(\mathbf{x})$: feature extractor
- $\hat{\mathbf{y}}$: predicted data label
- $\hat{\mathbf{d}}$: predicted domain label
- G_y, L_y: label predictor and loss
- G_d^k, L_d^k: domain discriminator
- GRL: gradient reversal layer

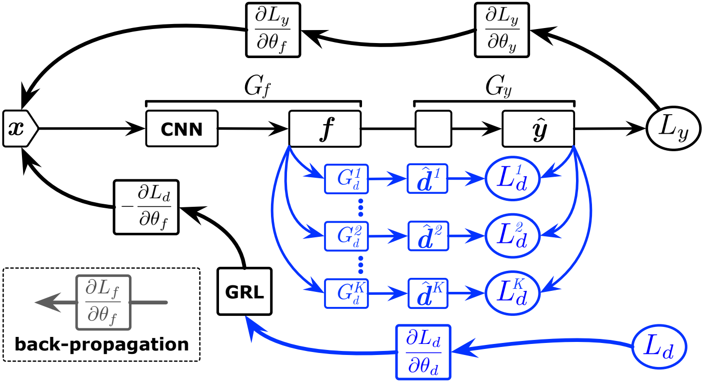
Selective Adversarial Networks



Instance Weighting (IW): probability-weighted loss for $G_d^k, k = 1, \ldots, |\mathcal{C}_s|$

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

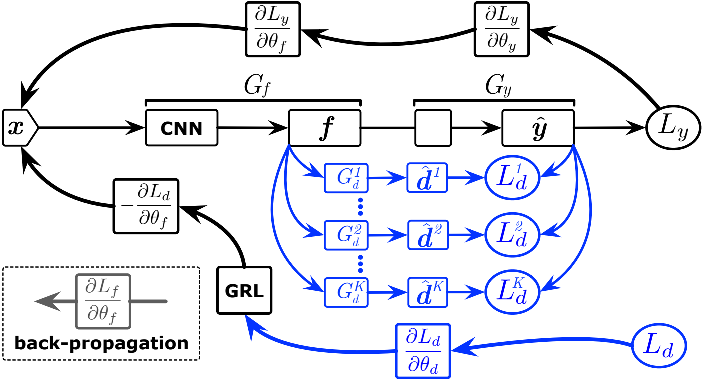
Selective Adversarial Networks



Class Weighting (CW): down-weigh $G_d^k, k = 1, \ldots, |\mathcal{C}_s|$ for outlier classes

$$L_d = \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \left\{ \left(\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k \right) \times \left(\sum_{\mathbf{x}_i \in (\mathcal{D}_s \cup \mathcal{D}_t)} \hat{y}_i^k L_d^k \left(G_d^k \left(G_f \left(\mathbf{x}_i \right) \right), d_i \right) \right) \right\} \quad (2)$$

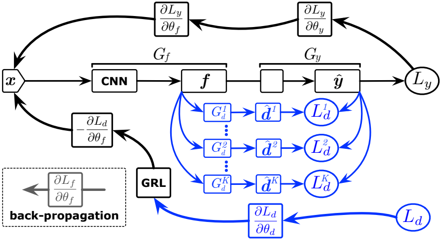
Selective Adversarial Networks



Entropy (uncertainty) minimization: $H\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right)\right) = -\sum_{k=1}^{|\mathcal{C}_s|} \hat{y}_i^k \log \hat{y}_i^k$

$$E = \frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} H\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right)\right) \tag{3}$$

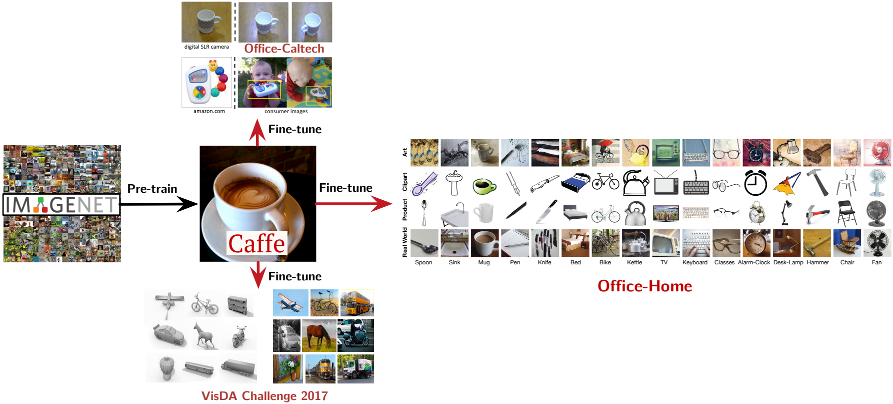
Selective Adversarial Networks



$$C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right) = \frac{1}{n_s}\sum_{\mathbf{x}_i \in \mathcal{D}_s} L_y\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right), y_i\right) + \frac{1}{n_t}\sum_{\mathbf{x}_i \in \mathcal{D}_t} H\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right)\right)$$
$$- \frac{1}{n_s + n_t}\sum_{k=1}^{|\mathcal{C}_s|}\left\{\left(\frac{1}{n_t}\sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k\right) \times \left(\sum_{\mathbf{x}_i \in (\mathcal{D}_s \cup \mathcal{D}_t)} \hat{y}_i^k L_d^k\left(G_d^k\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right)\right)\right\} \tag{4}$$

$$(\hat{\theta}_f, \hat{\theta}_y) = \arg\min_{\theta_f, \theta_y} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right)$$
$$(\hat{\theta}_d^1, ..., \hat{\theta}_d^{|\mathcal{C}_s|}) = \arg\max_{\theta_d^1, ..., \theta_d^{|\mathcal{C}_s|}} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right) \tag{5}$$

Setup



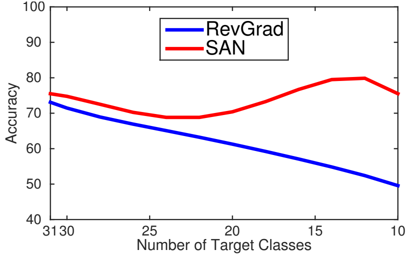
- **Transfer Tasks:** Office-31 ($31 \to 10$), Caltech-Office ($256 \to 10$) and ImageNet-Caltech ($I1000 \to C84$ and $C256 \to I84$)

Results

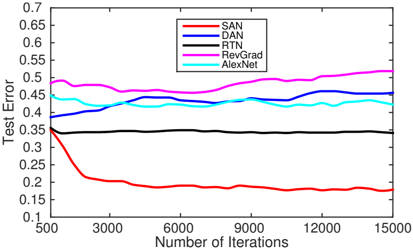
Method	Office-31						
	A 31 → W 10	D 31 → W 10	W 31 → D 10	A 31 → D 10	D 31 → A 10	W 31 → A 10	Avg
AlexNet [2]	58.51	95.05	98.08	71.23	70.6	67.74	76.87
DAN [3]	56.52	71.86	86.78	51.86	50.42	52.29	61.62
RevGrad [1]	49.49	93.55	90.44	49.68	46.72	48.81	63.11
RTN [4]	66.78	86.77	99.36	70.06	73.52	76.41	78.82
ADDA [5]	70.68	96.44	98.65	72.90	74.26	75.56	81.42
SAN-selective	71.51	98.31	100.00	78.34	77.87	76.32	83.73
SAN-entropy	74.61	98.31	100.00	80.29	78.39	82.25	85.64
SAN	**80.02**	**98.64**	**100.00**	**81.28**	**80.58**	**83.09**	**87.27**

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 [2]	58.44	76.64	65.86	66.98	52.37	47.35	49.86
DAN [3]	42.37	70.75	47.04	53.39	54.21	52.03	53.12
RevGrad [1]	54.57	72.86	57.96	61.80	51.34	47.02	49.18
RTN [4]	71.02	81.32	62.35	71.56	63.69	50.45	57.07
ADDA [5]	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**

Analysis



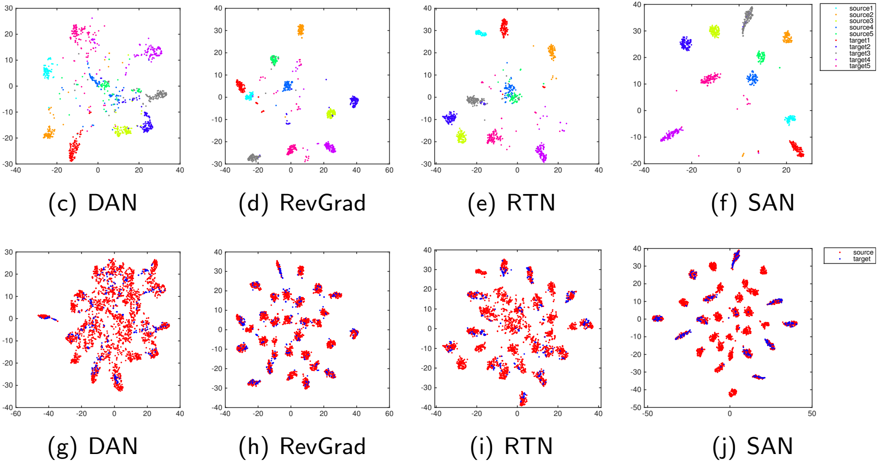
(a) Accuracy w.r.t #Target Classes



(b) Test Error

- SAN outperforms RevGrad even more for larger class-space difference
- SAN converges more stably and fast to lower test error than RevGrad

Visualization



(c) DAN (d) RevGrad (e) RTN (f) SAN

(g) DAN (h) RevGrad (i) RTN (j) SAN

Figure: t-SNE with class information (top) and domain information (bottom).

References

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky.
Domain-adversarial training of neural networks.
Journal of Machine Learning Research, 17:59:1–59:35, 2016.

A. Krizhevsky, I. Sutskever, and G. E. Hinton.
Imagenet classification with deep convolutional neural networks.
In *NIPS*, 2012.

M. Long, Y. Cao, J. Wang, and M. I. Jordan.
Learning transferable features with deep adaptation networks.
In *ICML*, 2015.

M. Long, H. Zhu, J. Wang, and M. I. Jordan.
Unsupervised domain adaptation with residual transfer networks.
In *NIPS*, pages 136–144, 2016.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell.
Adversarial discriminative domain adaptation.
In *CVPR*, 2017.

Summary

- A novel selective adversarial network for partial transfer learning
  - Circumvent negative transfer by selecting out outlier source classes
  - Promote positive transfer by matching shared-class-space distributions
- Code will be available soon at: `https://github.com/thuml/`
- A work at CVPR 2018 follows our arXiv version: how fast they are!