Transfer Learning
From Algorithms to Theories and Back

Mingsheng Long

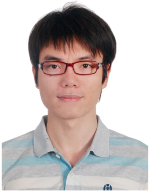
School of Software
National Engineering Lab for Big Data Software
Research Center for Big Data, Tsinghua University

`https://github.com/thuml`

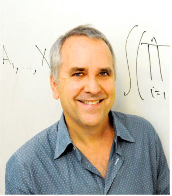
Vision And Learning SEminar, VALSE 2019

Joint Work With:

Jianmin Wang
Professor
Tsinghua University
jimwang@tsinghua.edu.cn

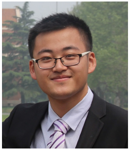


Mingsheng Long
Associate Professor
Tsinghua University
mingsheng@tsinghua.edu.cn



Michael I. Jordan
Professor
UC Berkeley
jordan@cs.berkeley.edu

Yuchen Zhang



Yue Cao

Han Zhu

Zhangjie Cao

Outline

1. Transfer Learning
2. Problem I: $P(\mathbf{X}) \neq Q(\mathbf{X})$
   - DAN: Deep Adaptation Network
3. Problem II: $P(Y|\mathbf{X}) \neq Q(Y|\mathbf{X})$
   - CDAN: Conditional Domain Adversarial Network
4. Bridging Algorithms and Theories
   - MDD: Margin Disparity Discrepancy
5. Benchmarking

Machine Learning

Learner: $f : x \to y$ **Distribution:** $(x, y) \sim P(x, y)$

fish

bird

mammal

tree

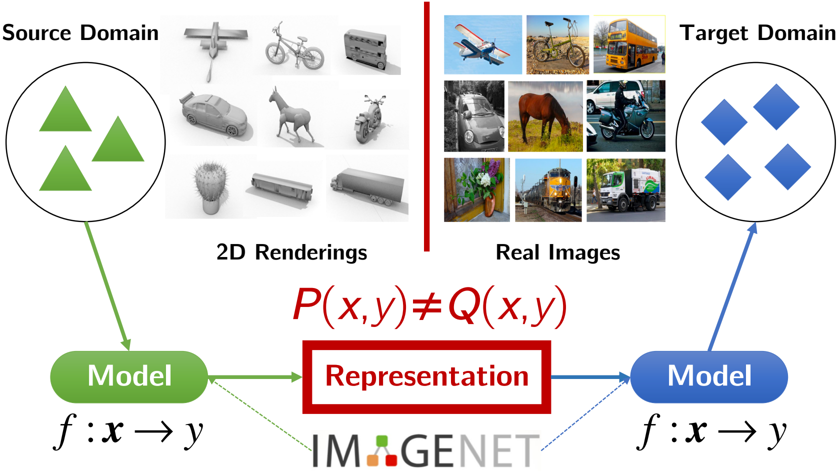
flower

......

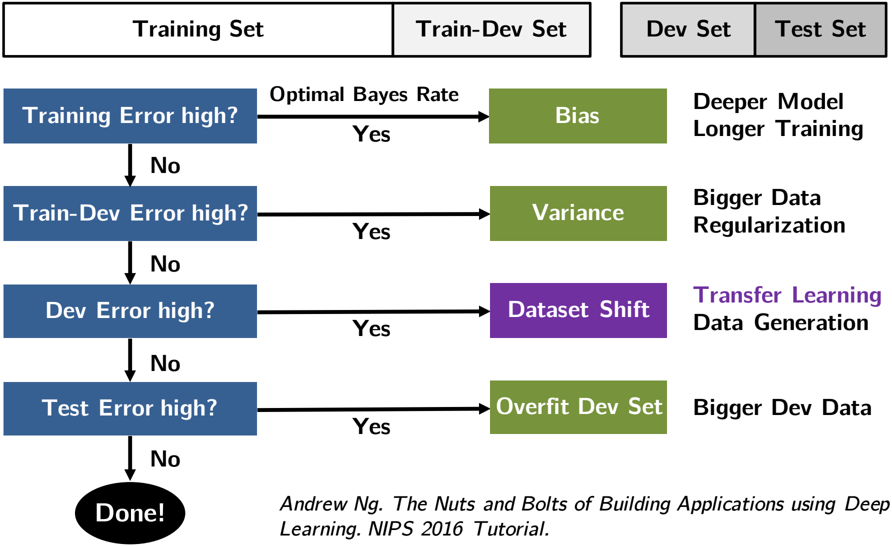
Error Bound: $\epsilon_{\text{test}} \leq \hat{\epsilon}_{\text{train}} + \sqrt{\frac{\text{complexity}}{n}}$

Transfer Learning

- Machine learning across domains of IDD distributions $P \neq Q$
- How to design models that effectively bound the generalization error?



Bias-Variance-Shift Tradeoff

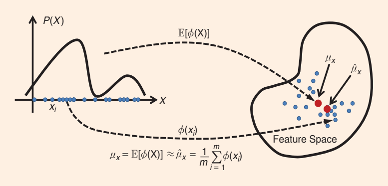


Andrew Ng. The Nuts and Bolts of Building Applications using Deep Learning. NIPS 2016 Tutorial.

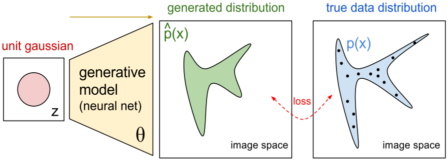
Basic Approaches to Transfer Learning

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

- Reduce marginal distribution mismatch: $P(\mathbf{X}) \neq Q(\mathbf{X})$
- Reduce conditional distribution mismatch: $P(Y|\mathbf{X}) \neq Q(Y|\mathbf{X})$
- Challenge: how to align different domains of multimodal distributions



Kernel Embedding

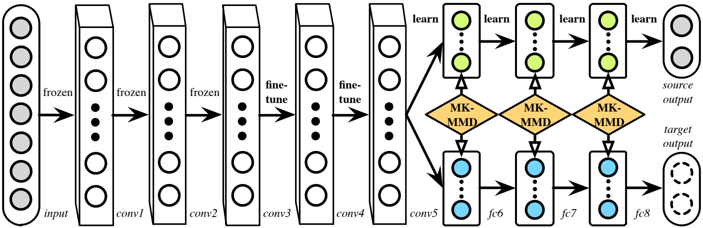


Adversarial Learning

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

Outline

DAN: Deep Adaptation Network[1]



Deep adaptation: match distributions in multiple domain-specific layers

Optimal matching: maximize two-sample test power by multiple kernels

$$d_k^2(P, Q) \triangleq \left\| \mathbf{E}_P[\phi(\mathbf{x}^s)] - \mathbf{E}_Q\left[\phi\left(\mathbf{x}^t\right)\right] \right\|_{\mathcal{H}_k}^2 \tag{1}$$

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

[1] *Mingsheng Long, Yue Cao, Jianmin Wang, Michael I. Jordan. Learning Transferable Features with Deep Adaptation Networks. ICML '15.*

DAN: MK-MMD

Multiple Kernel Maximum Mean Discrepancy (MK-MMD)

RKHS distance between *kernel embeddings* of distributions P_X and Q_X

$$d_k^2(P,Q) \triangleq \left\| \mathbf{E}_P[\phi(\mathbf{x}^s)] - \mathbf{E}_Q\left[\phi\left(\mathbf{x}^t\right)\right] \right\|_{\mathcal{H}_k}^2, \tag{3}$$

$k(\mathbf{x}^s, \mathbf{x}^t) = \langle \phi(\mathbf{x}^s), \phi(\mathbf{x}^t) \rangle$ is a convex combination of m PSD kernels

$$\mathcal{K} \triangleq \left\{ k = \sum_{u=1}^{m} \beta_u k_u : \sum_{u=1}^{m} \beta_u = 1, \beta_u \geqslant 0, \forall u \right\}. \tag{4}$$

Theorem (Kernel Two-Sample Test (Gretton et al. 2012))

- *$P = Q$ if and only if $d_k^2(P,Q) = 0$ (In practice, $d_k^2(P,Q) < \epsilon$)*
- $\max\limits_{k \in \mathcal{K}} d_k^2(P,Q)\,\sigma_k^{-2} \Leftrightarrow$ min *Type II Error ($d_k^2(P,Q) < \epsilon$ when $P \neq Q$)*

DAN: Feature Learning

Linear-Time Algorithm of MK-MMD (Streaming Algorithm)

$O(n^2)$: $d_k^2(p,q) = \mathbf{E}_{\mathbf{x}^s\mathbf{x}'^s} k(\mathbf{x}^s, \mathbf{x}'^s) + \mathbf{E}_{\mathbf{x}^t\mathbf{x}'^t} k(\mathbf{x}^t, \mathbf{x}'^t) - 2\mathbf{E}_{\mathbf{x}^s\mathbf{x}^t} k(\mathbf{x}^s, \mathbf{x}^t)$

$O(n)$: $d_k^2(p,q) = \frac{2}{n_s}\sum_{i=1}^{n_s/2} g_k(\mathbf{z}_i) \to$ linear-time unbiased estimate

- Quad-tuple $\mathbf{z}_i \triangleq (\mathbf{x}_{2i-1}^s, \mathbf{x}_{2i}^s, \mathbf{x}_{2i-1}^t, \mathbf{x}_{2i}^t)$
- $g_k(\mathbf{z}_i) \triangleq k(\mathbf{x}_{2i-1}^s, \mathbf{x}_{2i}^s) + k(\mathbf{x}_{2i-1}^t, \mathbf{x}_{2i}^t) - k(\mathbf{x}_{2i-1}^s, \mathbf{x}_{2i}^t) - k(\mathbf{x}_{2i}^s, \mathbf{x}_{2i-1}^t)$

Stochastic Gradient Descent (SGD)

For each layer ℓ and for each quad-tuple $\mathbf{z}_i^\ell = \left(\mathbf{h}_{2i-1}^{s\ell}, \mathbf{h}_{2i}^{s\ell}, \mathbf{h}_{2i-1}^{t\ell}, \mathbf{h}_{2i}^{t\ell}\right)$

$$\nabla_{\Theta^\ell} = \frac{\partial J(\mathbf{z}_i)}{\partial \Theta^\ell} + \lambda \frac{\partial g_k(\mathbf{z}_i^\ell)}{\partial \Theta^\ell} \tag{5}$$

DAN: Kernel Learning

Learning optimal kernel $k = \sum_{u=1}^{m} \beta_u k_u$

Maximizing test power $\triangleq$ minimizing Type II error (Gretton et al. 2012)

$$\max_{k \in \mathcal{K}} d_k^2 \left(\mathcal{D}_s^\ell, \mathcal{D}_t^\ell \right) \sigma_k^{-2}, \tag{6}$$

where $\sigma_k^2 = \mathbf{E_z} g_k^2(\mathbf{z}) - [\mathbf{E_z} g_k(\mathbf{z})]^2$ is the estimation variance.

Quadratic Program (QP), scaling linearly to sample size: $O(m^2 n + m^3)$

$$\min_{\mathbf{d}^\mathsf{T}\boldsymbol{\beta}=1, \boldsymbol{\beta} \geqslant \mathbf{0}} \boldsymbol{\beta}^\mathsf{T} (\mathbf{Q} + \epsilon \mathbf{I}) \boldsymbol{\beta}, \tag{7}$$

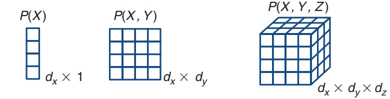
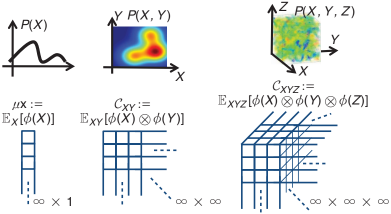
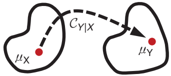
where $\mathbf{d} = (d_1, d_2, \ldots, d_m)^\mathsf{T}$, and each d_u is MMD using base kernel k_u.

Outline

CDAN: Conditional Domain Adversarial Network[2]

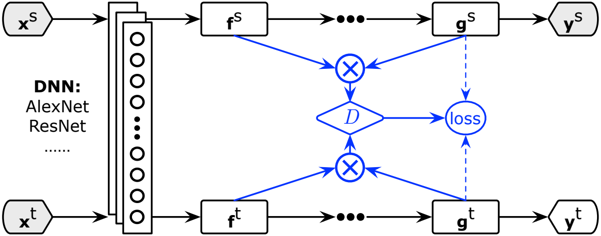
Main Idea of This Work: Distribution Embeddings with Statistics

- Capture cross-covariance statistics across multiple random vectors
  - Concatenation: $\mathbb{E}_{\mathbf{XY}}[\mathbf{X} \oplus \mathbf{Y}] = \mathbb{E}_{\mathbf{X}}[\mathbf{X}] \oplus \mathbb{E}_{\mathbf{Y}}[\mathbf{Y}]$
  - Multilinear: $\mathbb{E}_{\mathbf{XY}}[\mathbf{X} \otimes \mathbf{Y}] = \mathbb{E}_{\mathbf{X}}[\mathbf{X}|Y=1] \oplus \ldots \oplus \mathbb{E}_{\mathbf{X}}[\mathbf{X}|Y=C]$

	Distributions	Probabilistic Operations
Discrete	 $P(X)$, $d_x \times 1$; $P(X,Y)$, $d_x \times d_y$; $P(X,Y,Z)$, $d_x \times d_y \times d_z$	Sum Rule: $Q(X) = \sum_Y P(X\|Y)\pi(Y)$ Product Rule: $Q(X,Y) = P(X/Y)\pi(Y)$ Bayes' Rule: $Q(Y\|x) = \frac{P(x/Y)\pi(Y)}{Q(X)}$
Kernel Embedding	 $P(X)$; $P(X,Y)$; $P(X,Y,Z)$; $\mu_X := \mathbb{E}_X[\phi(X)]$, $\infty \times 1$; $\mathcal{C}_{XY} := \mathbb{E}_{XY}[\phi(X) \otimes \phi(Y)]$, $\infty \times \infty$; $\mathcal{C}_{XYZ} := \mathbb{E}_{XYZ}[\phi(X) \otimes \phi(Y) \otimes \phi(Z)]$, $\infty \times \infty \times \infty$	 $\mathcal{C}_{Y\|X}$, μ_X, μ_Y Sum Rule: $\mu_X^\pi = \mathcal{C}_{Y\|X}\mu_Y^\pi$ Product Rule: $\mathcal{C}_{XY}^\pi = \mathcal{C}_{Y\|X}\mathcal{C}_{YY}^\pi$ Bayes' Rule: $\mu_{Y\|x}^\pi = \mathcal{C}_{Y\|X}^\pi\phi(x)$

[2] *Mingsheng Long, Zhangjie Cao, Jianmin Wang, Michael I. Jordan. Conditional Adversarial Domain Adaptation. NIPS '18.*

CDAN: Multilinear Conditioning

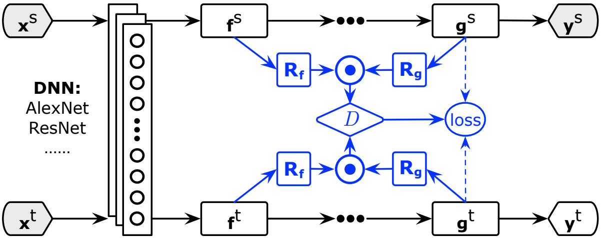


Conditional adaptation of distributions over representation & prediction

$$\begin{aligned} &\min_{G} \ \mathcal{E}(G) - \lambda\mathcal{E}(D, G) \\ &\min_{D} \ \mathcal{E}(D, G), \end{aligned} \tag{8}$$

$$\mathcal{E}(D, G) = -\mathbb{E}_{\mathbf{x}_i^s \sim \mathcal{D}_s} \log\left[D\left(\mathbf{f}_i^s \otimes \mathbf{g}_i^s\right)\right] - \mathbb{E}_{\mathbf{x}_j^t \sim \mathcal{D}_t} \log\left[1 - D\left(\mathbf{f}_j^t \otimes \mathbf{g}_j^t\right)\right] \tag{9}$$

CDAN: Randomized Multilinear Conditioning



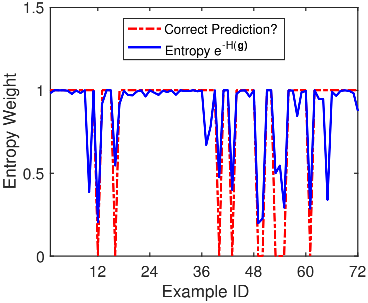
Conditional adaptation of distributions over representation & prediction

$$T_{\otimes}(\mathbf{f}, \mathbf{g}) = \mathbf{f} \otimes \mathbf{g} \tag{10}$$

$$T_{\odot}(\mathbf{f}, \mathbf{g}) = \frac{1}{\sqrt{d}} (\mathbf{R_f f}) \odot (\mathbf{R_g g}) \tag{11}$$

$$T(\mathbf{h}) = \begin{cases} T_{\otimes}(\mathbf{f}, \mathbf{g}) & \text{if } d_f \times d_g \leqslant 4096 \\ T_{\odot}(\mathbf{f}, \mathbf{g}) & \text{otherwise} \end{cases} \tag{12}$$

CDAN: Entropy Conditioning



Control the uncertainty of classifier prediction to guarantee transferability

$$w\left(H(\mathbf{g})\right) = 1 + e^{-H(\mathbf{g})}$$

$$\max_{D} \ \mathbb{E}_{\mathbf{x}_i^s \sim \mathcal{D}_s} w\left(H\left(\mathbf{g}_i^s\right)\right) \log\left[D\left(T\left(\mathbf{h}_i^s\right)\right)\right] + \mathbb{E}_{\mathbf{x}_j^t \sim \mathcal{D}_t} w\left(H\left(\mathbf{g}_j^t\right)\right) \log\left[1 - D\left(T\left(\mathbf{h}_j^t\right)\right)\right] \tag{13}$$

CDAN: Minimax Game

Conditional Domain Adversarial Networks (CDAN)

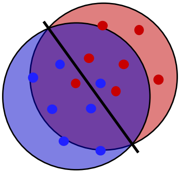
- Multilinear Conditioning: capture the cross-covariance between feature representation & classifier prediction to boost discriminability
- Entropy Conditioning: control the uncertainty of classifier prediction to guarantee transferability (entropy minimization principle)

$$\begin{aligned}&\min_{G} \mathbb{E}_{(\mathbf{x}_i^s, \mathbf{y}_i^s) \sim \mathcal{D}_s} L\left(G\left(\mathbf{x}_i^s\right), \mathbf{y}_i^s\right) \\ &\quad + \lambda \left(\mathbb{E}_{\mathbf{x}_i^s \sim \mathcal{D}_s} w\left(H\left(\mathbf{g}_i^s\right)\right) \log\left[D\left(T\left(\mathbf{h}_i^s\right)\right)\right] + \mathbb{E}_{\mathbf{x}_j^t \sim \mathcal{D}_t} w\left(H\left(\mathbf{g}_j^t\right)\right) \log\left[1 - D\left(T\left(\mathbf{h}_j^t\right)\right)\right]\right) \\ &\max_{D} \mathbb{E}_{\mathbf{x}_i^s \sim \mathcal{D}_s} w\left(H\left(\mathbf{g}_i^s\right)\right) \log\left[D\left(T\left(\mathbf{h}_i^s\right)\right)\right] + \mathbb{E}_{\mathbf{x}_j^t \sim \mathcal{D}_t} w\left(H\left(\mathbf{g}_j^t\right)\right) \log\left[1 - D\left(T\left(\mathbf{h}_j^t\right)\right)\right]\end{aligned} \tag{14}$$

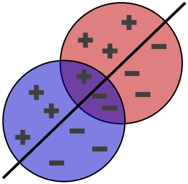
Outline

Notations and Assumptions

- Source risk: $\epsilon_P(G) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P}[G(\mathbf{x}) \neq \mathbf{y}]$
- Target risk: $\epsilon_Q(G) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim Q}[G(\mathbf{x}) \neq \mathbf{y}]$
- Source disparity: $\epsilon_P(G, G') = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P}[G(\mathbf{x}) \neq G'(\mathbf{x})]$
- Target disparity: $\epsilon_Q(G, G') = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim Q}[G(\mathbf{x}) \neq G'(\mathbf{x})]$
- Ideal hypothesis: $G^* = \arg\min_G \epsilon_P(G) + \epsilon_Q(G)$
- Assumption: ideal hypothesis has small risk $\epsilon_{ideal} = \epsilon_P(G^*) + \epsilon_Q(G^*)$



Distribution discrepancy



Ideal hypothesis with small error

Relating Target Risk to Source Risk

Theorem

The probabilistic bound of the target risk $\epsilon_Q(G)$ of (source) hypothesis G is given by the source risk $\epsilon_P(G)$ plus the distribution discrepancy:

$$\epsilon_Q(G) \leqslant \epsilon_P(G) + [\epsilon_P(G^*) + \epsilon_Q(G^*)] + |\epsilon_P(G, G^*) - \epsilon_Q(G, G^*)| \tag{15}$$

Proof.

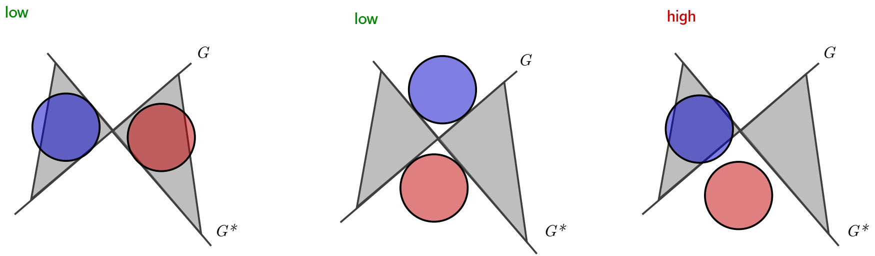
By using the triangle inequalities, we have

$$\begin{aligned} \epsilon_Q(G) &\leqslant \epsilon_Q(G^*) + \epsilon_Q(G, G^*) \\ &\leqslant \epsilon_Q(G^*) + \epsilon_P(G, G^*) + \epsilon_Q(G, G^*) - \epsilon_P(G, G^*) \\ &\leqslant \epsilon_Q(G^*) + \epsilon_P(G, G^*) + |\epsilon_Q(G, G^*) - \epsilon_P(G, G^*)| \\ &\leqslant \epsilon_P(G) + [\epsilon_P(G^*) + \epsilon_Q(G^*)] + |\epsilon_P(G, G^*) - \epsilon_Q(G, G^*)| \end{aligned} \tag{16}$$

□

Bounding the Distribution Discrepancy

Then how to bound the distribution discrepancy $\left|\epsilon_P(G, G^*) - \epsilon_Q(G, G^*)\right|$



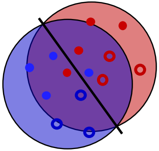
- $\mathcal{H}\Delta\mathcal{H}$-Divergence (Classic): $\sup_{G, G' \in \mathcal{H}} \left|\epsilon_P(G, G') - \epsilon_Q(G, G')\right|$
- Disparity Discrepancy (Ours): $\sup_{G' \in \mathcal{H}} \left|\epsilon_P(G, G') - \epsilon_Q(G, G')\right|$

Bounding the Distribution Discrepancy

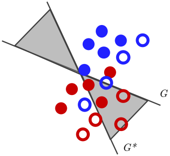
Let $\delta(\mathbf{x}) = |\mathbf{g} - G'(\mathbf{x})|$. The distribution discrepancy (DD) is bounded by

$$
\begin{aligned}
\left|\epsilon_P(G, G^*) - \epsilon_Q(G, G^*)\right| &= \left|\mathbb{E}_{(\mathbf{f},\mathbf{g})\sim P_G}\left[\mathbf{g} \neq G^*(\mathbf{f})\right] - \mathbb{E}_{(\mathbf{f},\mathbf{g})\sim Q_G}\left[\mathbf{g} \neq G^*(\mathbf{f})\right]\right| \\
&\leqslant \sup_{G' \in \mathcal{H}} \left|\mathbb{E}_{(\mathbf{f},\mathbf{g})\sim P_G}\left[\left|\mathbf{g} - G'(\mathbf{f})\right| \neq 0\right] - \mathbb{E}_{(\mathbf{f},\mathbf{g})\sim Q_G}\left[\left|\mathbf{g} - G'(\mathbf{f})\right| \neq 0\right]\right| \\
&\leqslant \sup_{\delta \in \Delta} \left|\mathbb{E}_{(\mathbf{f},\mathbf{g})\sim P_G}\left[\delta(\mathbf{f},\mathbf{g}) \neq 0\right] - \mathbb{E}_{(\mathbf{f},\mathbf{g})\sim Q_G}\left[\delta(\mathbf{f},\mathbf{g}) \neq 0\right]\right| \\
&\leqslant \sup_{D \in \mathcal{H}_D} \left|\mathbb{E}_{(\mathbf{f},\mathbf{g})\sim P_G}\left[D(\mathbf{f},\mathbf{g}) \neq 0\right] - \mathbb{E}_{(\mathbf{f},\mathbf{g})\sim Q_G}\left[D(\mathbf{f},\mathbf{g}) \neq 0\right]\right|
\end{aligned}
$$

This upper-bound can be evaluated by training a domain discriminator D.



Distribution discrepancy



Hypothesis-based distribution discrepancy

MDD: Towards an Informative Margin Theory[3]

- Multi-class Classification with Scoring Function and Margin Loss
- Scoring Function:

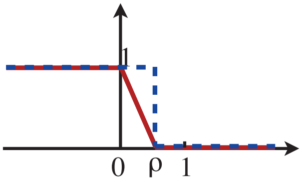
$$G \in \mathcal{F} : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$$

- Margin of a Hypothesis:

$$\rho_G(x, y) = \frac{1}{2}\left(G(x, y) - \max_{y' \neq y} G(x, y')\right)$$

- Margin Loss:

$$\Phi_\rho(x) = \begin{cases} 0 & \rho \leqslant x \\ 1 - x/\rho & 0 \leqslant x \leqslant \rho \\ 1 & x \leqslant 0 \end{cases}$$



[3] *Yuchen Zhang, Tianle Liu, Mingsheng Long*, Michael I. Jordan. Bridging Theory and Algorithm for Domain Adaptation. Preprint, 2019.*

MDD: Margin Disparity Discrepancy

- Source margin risk: $\epsilon_P^{(\rho)}(G) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P}\left[\Phi_\rho(\rho_G(x,y))\right]$
- Target margin risk: $\epsilon_Q^{(\rho)}(G) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim Q}\left[\Phi_\rho(\rho_G(x,y))\right]$
- Source margin disparity:
$$\epsilon_P^{(\rho)}(G_1, G_2) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim P}\left[\Phi_\rho(\rho_{G_2}(x, G_1^{labeling}(x)))\right]$$
- Target margin disparity:
$$\epsilon_Q^{(\rho)}(G_1, G_2) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim Q}\left[\Phi_\rho(\rho_{G_2}(x, G_1^{labeling}(x)))\right]$$
- Ideal hypothesis: $G^* = \arg\min_G \epsilon_P^{(\rho)}(G) + \epsilon_Q^{(\rho)}(G)$
- Margin Disparity Discrepancy (MDD):

$$d_{G,\mathcal{F}}^{(\rho)}(P,Q) = \sup_{G' \in \mathcal{H}} \left[\epsilon_Q^{(\rho)}(G,G') - \epsilon_P^{(\rho)}(G,G')\right]$$

MDD: Generalization Bound with Rademacher Complexity

Theorem

Let $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ be a hypothesis set with $\mathcal{Y} = \{1, \cdots, k\}$ and $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ be the corresponding $\mathcal{Y}$-valued classifier class. Fix $\rho > 0$. For all $\delta > 0$, with probability $1 - 3\delta$ the following inequality holds for all hypothesis $G \in \mathcal{F}$:

$$\begin{aligned} \epsilon_Q(G) \leq & \epsilon_{\widehat{P}}^{(\rho)}(f) + d_{G,\mathcal{F}}^{(\rho)}(\widehat{P}, \widehat{Q}) + \lambda \\ & + \frac{2k^2}{\rho}\mathfrak{R}_{n,P}(\Pi_1\mathcal{F}) + \frac{k}{\rho}\mathfrak{R}_{n,P}(\Pi_{\mathcal{H}}\mathcal{F}) + 2\sqrt{\frac{\log\frac{2}{\delta}}{2n}} \\ & + \frac{k}{\rho}\mathfrak{R}_{m,Q}(\Pi_{\mathcal{H}}\mathcal{F}) + \sqrt{\frac{\log\frac{2}{\delta}}{2m}}. \end{aligned} \tag{17}$$

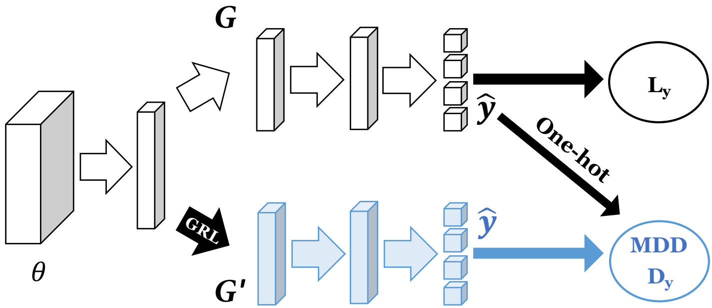
MDD: Generalization Bound with Covering Numbers

Theorem

Let $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}\times\mathcal{Y}}$ be a hypothesis set with $\mathcal{Y} = \{1, \cdots, k\}$ and $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ be the corresponding $\mathcal{Y}$-valued classifier class. Suppose $\Pi_1\mathcal{F}$ is bounded in $\mathcal{L}_2$ by L. Fix $\rho > 0$. For all $\delta > 0$, with probability $1 - 3\delta$ the following inequality holds for all hypothesis $G \in \mathcal{F}$:

$$
\begin{aligned}
\epsilon_Q(G) \leq & \epsilon_{\widehat{P}}^{(\rho)}(f) + d_{G,\mathcal{F}}^{(\rho)}(\widehat{P}, \widehat{Q}) + \lambda + 2\sqrt{\frac{\log\frac{2}{\delta}}{2n}} \\
& + \sqrt{\frac{\log\frac{2}{\delta}}{2m}} + \frac{16k^2\sqrt{k}}{\rho} \inf_{\epsilon\geq 0}\Big\{\epsilon + 3\big(\frac{1}{\sqrt{n}} + \frac{1}{\sqrt{m}}\big) \\
& \big(\int_{\epsilon}^{L}\sqrt{\log\mathcal{N}_2(\tau, \Pi_1\mathcal{F})}d\tau + L\int_{\epsilon/L}^{1}\sqrt{\log\mathcal{N}_2(\tau, \Pi_1\mathcal{H})}\mathrm{d}\tau\big)\Big\}.
\end{aligned}
\tag{18}
$$

MDD: Theory-Induced Algorithm



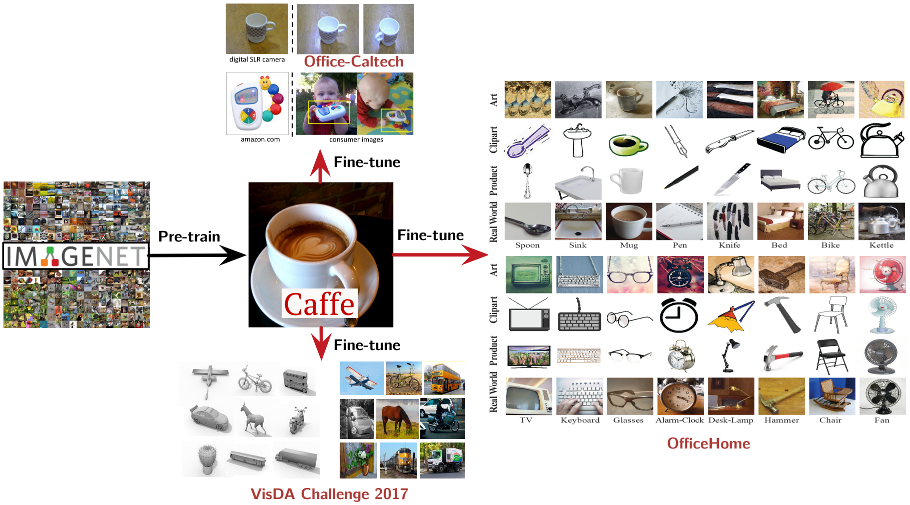
Minimax Optimization: Adversarial learning induced by the MDD Theory

$$\begin{aligned} &\min_{G} \; \epsilon_{\widehat{P}}^{(\rho)}(G) + \left(\epsilon_{\widehat{Q}}^{(\rho)}(G, G^*) - \epsilon_{\widehat{P}}^{(\rho)}(G, G^*)\right) \\ &G^* = \arg\max_{G'} \left(\epsilon_{\widehat{Q}}^{(\rho)}(G, G') - \epsilon_{\widehat{P}}^{(\rho)}(G, G')\right) \end{aligned} \tag{19}$$

Outline

1. Transfer Learning
2. Problem I: $P(\mathbf{X}) \neq Q(\mathbf{X})$
   - DAN: Deep Adaptation Network
3. Problem II: $P(Y|\mathbf{X}) \neq Q(Y|\mathbf{X})$
   - CDAN: Conditional Domain Adversarial Network
4. Bridging Algorithms and Theories
   - MDD: Margin Disparity Discrepancy
5. Benchmarking

Datasets

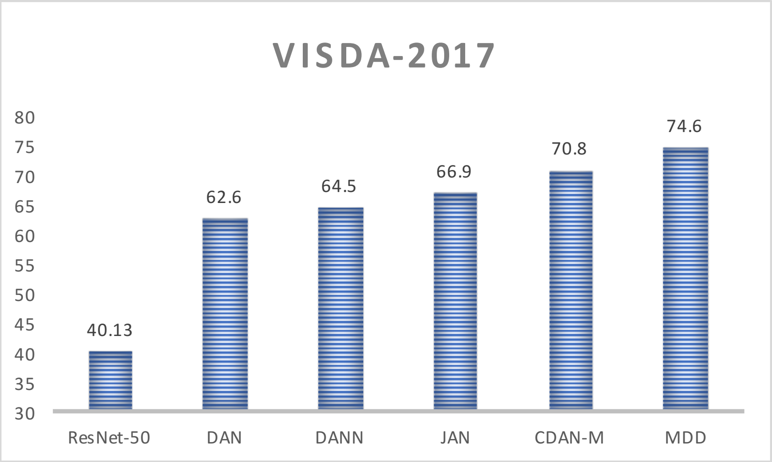

digital SLR camera
Office-Caltech
amazon.com
consumer images
Fine-tune
IMAGENET
Pre-train
Caffe
Fine-tune
Fine-tune
VisDA Challenge 2017
Art
Clipart
Product
Real World
Spoon
Sink
Mug
Pen
Knife
Bed
Bike
Kettle
TV
Keyboard
Glasses
Alarm-Clock
Desk-Lamp
Hammer
Chair
Fan
OfficeHome

Results

Table: Accuracy (%) on Office-31 for unsupervised domain adaptation

Method	A → W	D → W	W → D	A → D	D → A	W → A	Avg
AlexNet	61.6±0.5	95.4±0.3	99.0±0.2	63.8±0.5	51.1±0.6	49.8±0.4	70.1
DAN	68.5±0.5	96.0±0.3	99.0±0.3	67.0±0.4	54.0±0.5	53.1±0.5	72.9
RTN	73.3±0.3	96.8±0.2	99.6±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
DANN	73.0±0.5	96.4±0.3	99.2±0.3	72.3±0.3	53.4±0.4	51.2±0.5	74.3
ADDA	73.5±0.6	96.2±0.4	98.8±0.4	71.6±0.4	54.6±0.5	53.5±0.6	74.7
JAN	74.9±0.3	96.6±0.2	99.5±0.2	71.8±0.2	**58.3**±0.3	55.0±0.4	76.0
CDAN	**77.9**±0.3	96.9±0.2	**100.0**±.0	**74.6**±0.2	55.1±0.3	**57.5**±0.4	**77.0**
CDAN+E	77.6±0.2	**97.2**±0.1	**100.0**±.0	73.0±0.1	57.3±0.2	56.1±0.3	76.9
ResNet-50	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
DAN	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
RTN	84.5±0.2	96.8±0.1	99.4±0.1	77.5±0.3	66.2±0.2	64.8±0.3	81.6
DANN	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
ADDA	86.2±0.5	96.2±0.3	98.4±0.3	77.8±0.3	69.5±0.4	68.9±0.5	82.9
JAN	85.4±0.3	97.4±0.2	99.8±0.2	84.7±0.3	68.6±0.3	70.0±0.4	84.3
CDAN	93.0±0.2	98.4±0.2	**100.0**±.0	89.2±0.3	70.2±0.4	69.4±0.4	86.7
CDAN+E	93.1±0.1	**98.6**±0.1	**100.0**±.0	93.4±0.2	71.0±0.3	70.3±0.3	87.7
MDD	**94.5**±0.3	98.4±0.1	**100.0**±.0	**93.5**±0.2	**74.6**±0.3	**72.2**±0.1	**88.9**

Results: Simulation2Real



Summary & Thank You

- Domain adaptation theories inherently imply minimax games
- Connect to domain adaptation methods based on adversarial learning

- Disconnections between theory and algorithm:
  - Scoring functions and margin loss are standard choices for classifiers
  - Minimax game in large hypothesis space is hard to reach equilibrium

- More convincing advances can be made by bridging the gap between theories and algorithms

- Xlearn library is available: `https://github.com/thuml/Xlearn`