Learning Transferable Features with Deep Adaptation Networks
Mingsheng Long\textsuperscript{12}, Yue Cao\textsuperscript{1}, Jianmin Wang\textsuperscript{2}, and Michael I. Jordan\textsuperscript{2}
\textsuperscript{1}School of Software, Tsinghua University, China \hspace{1cm} \textsuperscript{2}AMPLab, Department of EECS, UC Berkeley

Summary
- A deep adaptation network for learning transferable features
- Two important improvements:
  - Deep adaptation of multiple task-specific layers (including output)
  - Optimal adaptation using multiple kernel two-sample matching
- A brief analysis of learning bound for the proposed deep network
- Outlook
  - Principle way of deciding the boundary of generality and specificity
  - Deeper adaptation of convolutional layers to enhance transferability
  - Fine-grained adaptation using structural embedding of distributions

Deep Learning for Domain Adaptation
- None or very limited supervision in the target task (new domain)
  - Target classifier cannot be reliably trained due to over-fitting
  - Fine-tuning is impossible as it requires substantial supervision
- Leverage supervision (same categories) from related source task
  - Deep networks can learn more transferable features for adaptation
  - Transferability of features decreases as the task discrepancy increases
- Hard to find big source task for learning deep features from scratch
  - Fine-tune from deep networks pre-trained on unrelated big dataset
  - Transferring features from distant tasks is better than random features

Deep Adaptation Network (DAN)
Key Assumptions (AlexNet)
- Conv-layers learn general features: safely transferable
  - Safely freeze conv1-conv3 & fine-tune conv4-conv5
- FC-layers fit domain-specific variations: NOT transferable
  - Deeply adapt fc6-fc8 using optimized two-sample matching

Unrelated Big Data
Deep Neural Network
Source Task
Target Task
Pre-train Fine-tune

Multiple Kernel Maximum Mean Discrepancy (MK-MMD)
\[ d^2_p (p, q) \triangleq \| \mathbb{E}_p [\phi (x^t)] - \mathbb{E}_q [\phi (x')] \|^2_{H_k}. \] (2)
\[ k (x^t, x') = \langle \phi (x^t), \phi (x') \rangle \text{ is a convex combination of } m \text{ PSD kernels } m k_i. \]
\[ k^* \triangleq \left\{ \sum_{i=1}^m \beta_i k_i : \sum_{i=1}^m \beta_i = 1, \beta_i \geq 0, \forall u \right\}. \] (3)

Two-Sample Test (Gretton et al. 2012)
- \( p = q \) if and only if \( d^2_p (p, q) = 0 \) (In practice, \( d^2_p (p, q) < \varepsilon \))
- \( \max_{k \in K} d^2_p (p, q) \approx \varepsilon \) \iff \text{min Type II Error } \left( d^2_p (p, q) < \varepsilon \text{ when } p \neq q \right) \)

Learning CNN
Linear-Time Algorithm of MK-MMD (Streaming Algorithm)
\[ O(n^2): \quad d^2_p (p, q) = \mathbb{E}_x \mathbb{E}_y [k(x^t, x') - 2 k(x^t, y')] \]
\[ O(n): \quad d^2_p (p, q) = \frac{1}{n^2} \sum_{i=1}^n \mathbb{E}_{z_i \sim p} [g(z_i) \rightarrow \text{linear-time unbiased estimate}] \]
- Quad-tuple \( z_i \triangleq (x_{i-1}, x_i, x_{i+1}, x_i') \) is better than \( k(x_{i-1}, x_i) - k(x_{i-1}, x_{i+1}) - k(x_i, x_{i+1}) \)
- SGD: for each layer \( l \) and each quad-tuple \( z_i \)
  \[ \begin{equation}
  \nabla g_l = \frac{\partial J (z)}{\partial g_l} + \lambda \frac{\partial g_l (z)}{\partial g_l}
  \end{equation} \] (4)

Learning Kernel
Learning optimal kernel \( k \triangleq \sum_{i=1}^m \beta_i k_i \) by minimizing Type II error
\[ \max_{k \in K} d^2_p (D^t, D^1) \approx \varepsilon^2, \] (5)
where \( \varepsilon^2 = \mathbb{E}_x \mathbb{E}_y [g(z) - (\mathbb{E}_y [g(z)])]^2 \) is the estimation variance.

Quadratic Program (QP), scaling linearly to sample size: \( O(m^2 n + m^3) \)
\[ \min_{d: \beta_i \geq 1, \beta_0 = 0} \beta (Q + \varepsilon I) \beta, \] (6)
where \( q = (d_1, d_2, \ldots, d_m)^T \), and each \( d_i \) is MMD using base kernel \( k_i \).

Theorem (Adaptation Bound)
Let \( \theta \in \mathcal{H} \) be a hypothesis, \( \epsilon_i (\theta) \) and \( \epsilon (\theta) \) be the expected risks of source and target respectively, then
\[ \epsilon_i (\theta) \leq \epsilon (\theta) + 2 d^2_p (p, q) + C, \] (7)
where \( C \) is a constant for the complexity of hypothesis space and the risk of an ideal hypothesis for both domains.

VC-Dimension of neural nets with linear-threshold gates \( O (W \log W) \).

Results and Discussion
Learn transferable features by deep adaptation and optimal matching
- Deep adaptation of multiple domain-specific layers (DAN) vs. shallow adaptation of one hard-to-tweak layer (DDC)
- Two samples can be matched better by MK-MMD vs. SK-MMD
- Semi-supervised adaptation: source vs. target supervision?
  - Limited target supervision is prone to over-fitting the target task
  - Source supervision can provide strong but inaccurate inductive bias
  - Via source inductive bias, target supervision is much more powerful
  - Two-sample matching is more effective for bridging disimilar tasks

Empirical Analysis
How generalization performance relates to two-sample discrepancy?
- \( d^2 \) on CNN & DAN features is larger than \( d^2 \) on Raw features
  - Deep features are salient for both category & domain discrimination
- \( d^2 \) on DAN feature is much smaller than \( d^2 \) on CNN feature
  - Domain adaptation can be boosted by minimizing domain discrepancy

- CNN on Source
- (b) DDC on Target
- (c) DAN on Target
- (d) A-distance
- (e) Accuracy vs. \( \lambda \)