Transfer Adversarial Training:
A General Approach to Adapting Deep Classifiers

Hong Liu, Mingsheng Long, Jianmin Wang, Michael I. Jordan

School of Software, Tsinghua University
National Engineering Lab for Big Data Software
University of California. Berkeley

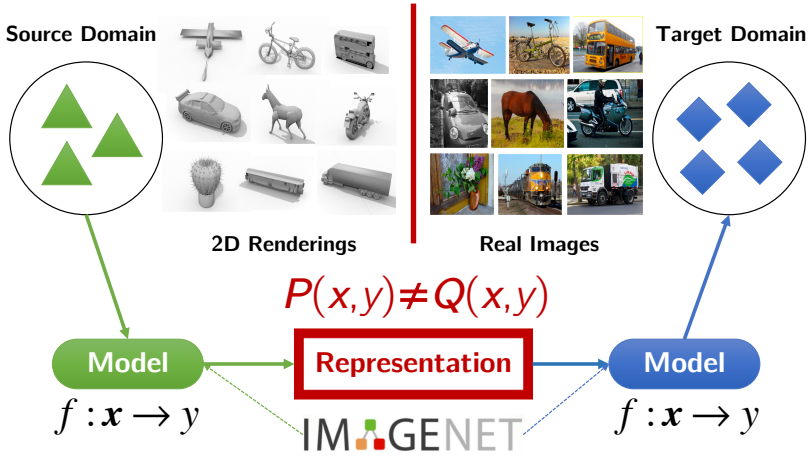
https://github.com/thuml

36th International Conference on Machine Learning

Outline

Transfer Learning

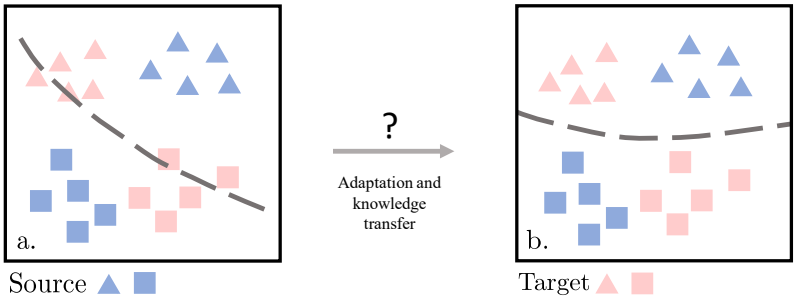
- In real-world applications, the IID assumption is frequently violated.
- How to generalize a learner across Non-IID distributions $P \neq Q$.



Domain Adaptation

Transfer knowledge across different domains:

- The learner is provided with n_s i.i.d. observations $\{\mathbf{x}_s^{(i)}, \mathbf{y}_s^{(i)}\}_{i=1}^{n_s}$ from a source domain of distribution $P(\mathbf{x}_s, \mathbf{y}_s)$, and n_t i.i.d. observations $\{\mathbf{x}_t^{(i)}\}_{i=1}^{n_t}$ from a target domain of distribution $Q(\mathbf{x}_t, \mathbf{y}_t)$.
- Learn an accurate model for the target domain
- Formally bound the target risk with the source risk



The $\mathcal{H}\Delta\mathcal{H}$-divergence

For any hypothesis $h \in \mathcal{H}$, with probability no less than $1 - \delta$,

$$\epsilon_Q(h, f_Q) \leq \epsilon_{\hat{P}}(h, f_P) + D_{\mathcal{H}\Delta\mathcal{H}}(\hat{P}, \hat{Q}) + \lambda + 10\hat{\mathcal{R}}_P(h) + 8\hat{\mathcal{R}}_Q(h) + 6\sqrt{\frac{\log\frac{6}{\delta}}{m}} + 3\sqrt{\frac{\log\frac{6}{\delta}}{n}}, \tag{1}$$

where $D_{\mathcal{H}\Delta\mathcal{H}}(P, Q) = \sup_{h,h' \in \mathcal{H}} |\epsilon_Q(h, h') - \epsilon_P(h, h')|$,

$$\lambda = \epsilon_P(h^*, f_P) + \epsilon_Q(h^*, f_Q), \tag{2}$$

$$h^* = \arg\min_{h \in \mathcal{H}} \epsilon_P(h, f_P) + \epsilon_Q(h, f_Q). \tag{3}$$

- Intuitively, the target risk can be bounded with the source risk + discrepancy between the source and the target + the best hypothesis risk we can expect.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. Machine Learning, 79(1-2):151–175, 2010.

Adversarial Feature Adaptation

Minimize the source risk

- Train the model with supervision from the source domain

Minimize the discrepancy term

- Learn a new feature representation where the discrepancy is minimized.

The two-player game

- A domain discriminator tries to discriminate the source and target domains, while the feature extractor tries to confuse it.
- Two classifier try to maximize their disagreement while the feature extractor tries to minimize it.

Ganin, Y., Ustinova, E.,Ajakan, H., Germain, P., Larochelle, H., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. Journal of Machine Learning Research, 17(1):2096–2030, 2016.

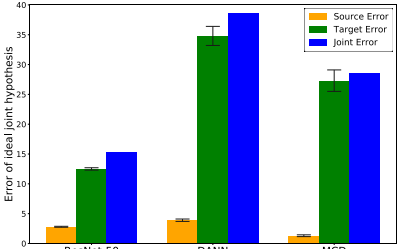
Saito, K., Watanabe, K., Ushiku, Y., and Harada, T. Maximum classifier discrepancy for unsupervised domain adaptation. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3723–3732, 2018.

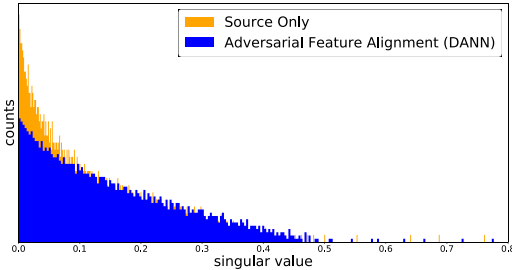
Outline

Hidden Limitations of Adversarial Feature Adaptation

Adaptability quantified by λ, is an essential prerequisite of domain adaptation.

- If λ is large, we can never expect to adapt a learner trained on the source domain to the target domain.
- Simply learning a new feature representation cannot guarantee that the ideal joint risk won't explode!





Diminishing domain-specific variations inevitably breaks the discriminative structures of the original representations.

Possible Solutions

Since we have no access to target labels, we cannot expect to minimize λ directly. Can we at least prevent the adaptability from going worse?

- **FIX** the feature representations and adapt classifiers instead.

With feature representations fixed, how can we adapt to the target domain?

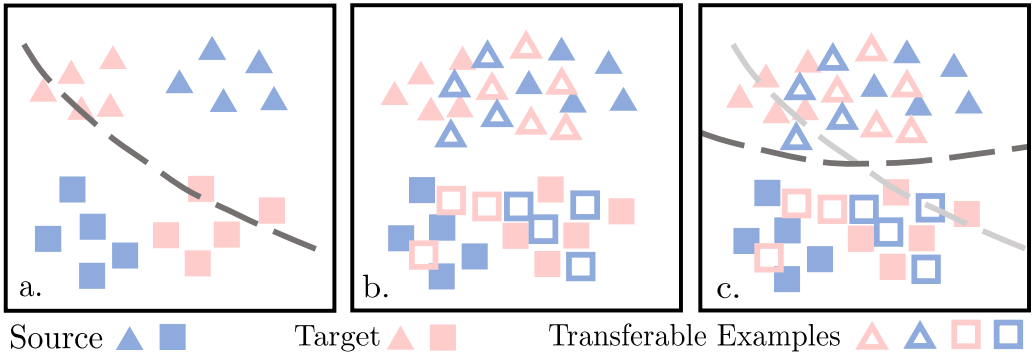
- Adapt deep classifiers instead.
- Extend adversarial training paradigm to domain adaptation.

Outline

Transferable Adversarial Training

Instead of feature adaptation, associate the source and target domain with transferable examples.

- Generate **transferable examples** at feature level.
- Adapt the classifier to the target domain by training on transferable examples.



Generating Transferable Examples

Generate Transferable Examples to bridge domain gap.

- Train a classifier and a domain discriminator.
- Transferable examples should confuse both the classifier and the domain discriminator.

$$\ell_d\left(\theta_D, \mathbf{f}\right) = -\frac{1}{n_s}\sum_{i=1}^{n_s}\log[D(\mathbf{f}_s^{(i)})] - \frac{1}{n_t}\sum_{i=1}^{n_t}\log[1 - D(\mathbf{f}_t^{(i)})]. \tag{4}$$

$$\ell_c(\theta_C, \mathbf{f}) = \frac{1}{n_s}\sum_{i=1}^{n_s}\ell_{ce}(C(\mathbf{f}_s^{(i)}), \mathbf{y}_s^{(i)}). \tag{5}$$

Concretely, we generate transferable examples from both domains in an iterative manner,

$$\mathbf{f}_{t^{k+1}} \leftarrow \mathbf{f}_{t^k} + \beta\nabla_{\mathbf{f}_{t^k}}\ell_d(\theta_D, \mathbf{f}_{t^k}) - \gamma\nabla_{\mathbf{f}_{t^k}}\ell_2(\mathbf{f}_{t^k}, \mathbf{f}_{t^0}), \tag{6}$$

$$\mathbf{f}_{s^{k+1}} \leftarrow \mathbf{f}_{s^k} + \beta\nabla_{\mathbf{f}_{s^k}}\ell_d(\theta_D, \mathbf{f}_{s^k}) - \gamma\nabla_{\mathbf{f}_{s^k}}\ell_2(\mathbf{f}_{s^k}, \mathbf{f}_{s^0}) + \beta\nabla_{\mathbf{f}_{s^k}}\ell_c(\theta_C, \mathbf{f}_{s^k}). \tag{7}$$

Training with Transferable Examples

Training the classifier and the domain discriminator on transferable examples.

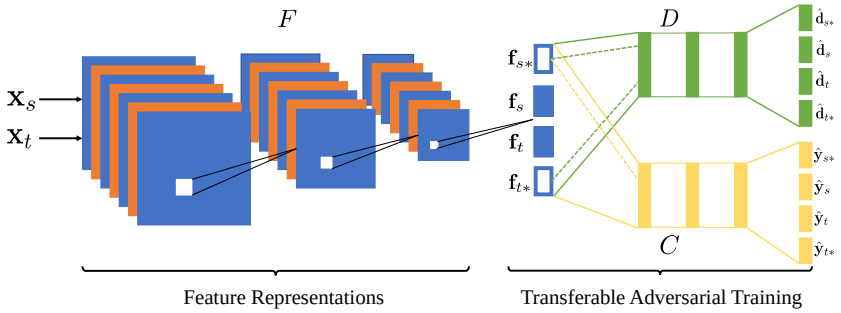
- We require the classifier to make consistent predictions for the transferable examples and their original counterparts.
- Train the domain discriminator to further distinguish transferable examples generated from the source and target.

$$\ell_{c,adv}(\theta_C, \mathbf{f}_*) = \frac{1}{n_s}\sum_{i=1}^{n_s} \ell_{ce}(C(\mathbf{f}_{s*}^{(i)}), \mathbf{y}_{s*}^{(i)}) + \frac{1}{n_t}\sum_{i=1}^{n_t} \left| C((\mathbf{f}_{t*}^{(i)})) - C((\mathbf{f}_{t}^{(i)})) \right|, \tag{8}$$

$$\ell_{d,adv}(\theta_D, \mathbf{f}_*) = -\frac{1}{n_s}\sum_{i=1}^{n_s} \log[D(\mathbf{f}_{s*}^{(i)})] - \frac{1}{n_t}\sum_{i=1}^{n_t} \log[1 - D(\mathbf{f}_{t*}^{(i)})]. \tag{9}$$

The Overall Optimization Problem

$$\min_{\theta_D,\theta_C} \ell_d(\theta_D, \mathbf{f}) + \ell_c(\theta_C, \mathbf{f}) + \ell_{d,adv}(\theta_D, \mathbf{f}_*) + \ell_{c,adv}(\theta_C, \mathbf{f}_*). \tag{10}$$

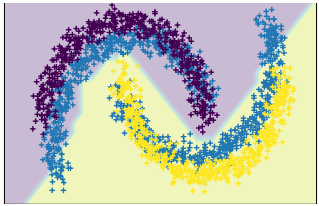


- **Fixed** feature representations – guaranteed adaptability
- No need of feature adaptation – light weight computation
- An order of magnitude faster than adversarial feature adaptation

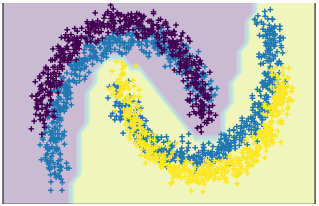
Outline

Analysis

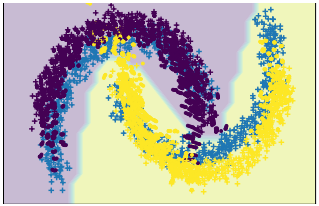
The rotating two moon problem: The target domain is rotated 30° from the source domain.



(a) Source Only Model



(b) TAT



(c) Transferable Examples

Behaviors on the *two moon* problem. Purple and yellow "+"s indicate source samples, blue "+"s are target samples, while dots are transferable examples. (a) The source only model. (b) The decision boundary of TAT. (c) The distribution of the transferable examples.

- As expected, transferable examples bridge domain gap effectively.

Experimental Setups

Datasets

- Office-31: Standard benchmark
- Image-CLEF: Balanced domains
- Office-home: Large domain gap
- VisDA: Large-scale synthetic-to-real
- Multi-domain sentiment: Sentiment polarity classification

Results

Table: Classification accuracies (%) on Office-31 with ResNet-50.

Method	A→W	D→W	W→D	A→D	D→A	W→A	Avg.
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
DANN	82.6±0.4	96.9±0.2	99.3±0.2	81.5±0.4	68.4±0.5	67.5±0.5	82.7
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
VADA	86.5±0.5	98.2±0.4	99.7±0.2	86.7±0.4	70.1±0.4	70.5±0.4	85.4
GTA	89.5±0.5	97.9±0.3	99.7±0.2	87.7±0.5	72.8±0.3	71.4±0.4	86.5
MCD	88.6±0.2	98.5±0.1	**100.0**±.0	92.2±0.2	69.5±0.1	69.7±0.3	86.5
CDAN	**93.1**±0.1	98.6±0.1	**100.0**±.0	92.9±0.2	71.0±0.3	69.3±0.3	87.5
TAT	92.5±0.3	**99.3**±0.1	**100.0**±.0	**93.2**±0.2	**73.1**±0.3	**72.1**±0.3	**88.4**

Table: Classification accuracies (%) on Office-Home with ResNet-50.

Method	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-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
TAT	**51.6**	**69.5**	**75.4**	**59.4**	**69.5**	**68.6**	**59.5**	**50.5**	**76.8**	**70.9**	**56.6**	**81.6**	**65.8**

Summary

- Adaptability quantified by λ is not guaranteed by feature adaptation and may be worsened.
- A new perspective: Adapt deep classifiers instead of feature representations.
- Associate the source and target domains with transferable examples: Extending adversarial training paradigm to transfer learning
- Free from adversarial feature learning: Lighter computation, faster speed, and even better performance.

Thanks!

Contact: h-l17@mails.tsinghua.edu.cn
mingsheng@tsinghua.edu.cn

Poster: Pacific Ballroom #255, Wed, June 12

Code: `github.com/thuml/Transferable-Adversarial-Training`