基于Fredholm多核学习的半监督知识迁移

Fredholm Multiple Kernel Learning for Semi-Supervised Domain Adaptation

The Thirty-First AAAI Conference on Artificial Intelligence 2017

Wei Wang, Hao Wang, Chen Zhang, Yang Gao
Institute of Software, Chinese Academy of Sciences, wangwei2014@iscas.ac.cn

Motivation

Semi-Supervised Domain Adaptation:

•Use **labeled source** and **target data**: adapting traditional models (e.g., SVM, logistic regression) to the target domain;
•Use **unlabeled target data**: coping with the inconsistency of two data distributions.

Challenges:

•Ignore **unlabeled target data** in the process of learning adaptive classifiers, while the unlabeled data is desirable for **robustness** and **noise resiliency**.

Semi-Supervised Kernel Prediction:

•Based on Fredholm integral, use **labeled** and **unlabeled data** samples for noise suppression.
•Learn a predictive function over the RKHS $\mathcal{H}$:

$$f = arg\min \frac{1}{n}\sum_{i=1}^{n} L(\mathcal{K}_P f(\mathbf{x}_i), y_i) + \beta\|f\|^2_{\mathcal{H}}$$

where $\mathcal{K}_P$ is the Fredholm integral $\mathcal{K}_P f(\mathbf{x}) = \int k^P(\mathbf{x},\mathbf{z}) f(\mathbf{z}) P(\mathbf{z}) d\mathbf{z}$ and $k^P(\mathbf{x},\mathbf{z})$ is a predefined outside kernel.

Challenges:

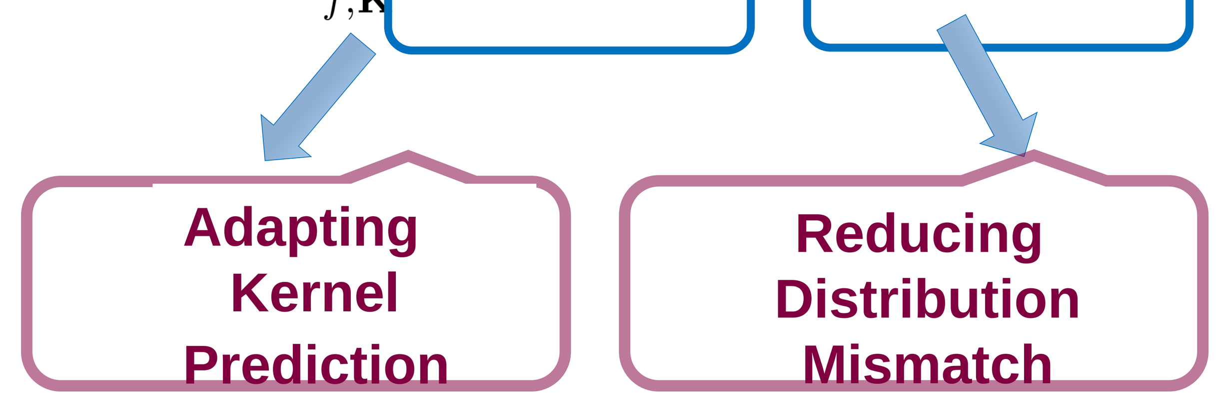
•Effective only when labeled and unlabeled data has **the same distribution**.
• Rely on **the single** $k^P(\mathbf{x},\mathbf{z})$ and the choice heavily influences the performance.

Contributions

•A novel domain adaptation method: **Transfer Fredholm Multiple Kernel Learning**.
•A predictive model for **noise resiliency, facilitating knowledge transfer** and **analyzing diverse data characteristics**.
•A simple but efficient procedure, guaranteeing rapid **convergence**.

Transfer Fredholm Multiple Kernel Learning

$$[f, \mathbf{K}] = arg\min_{f,\mathbf{K}} \ell(f, \mathbf{K}, D_s, D_t) + \theta\Omega(d_{\mathbf{K}}(D_s, D_t)),$$



Adapting Kernel Prediction:

•Construct two Fredholm integrals on the two domains respectively:

$$\ell(f,\mathbf{K},D_s,D_t) = arg\min_{f\in\mathcal{H}}\{\beta\|f\|^2_{\mathcal{H}} + \frac{\lambda_s}{n_s}\sum_{i=1}^{n_s} L(\mathcal{K}_{P_s} f(\mathbf{x}_i^s), y_i^s) + \frac{\lambda_t}{n_l}\sum_{i=1}^{n_l} L(\mathcal{K}_{P_t} f(\mathbf{x}_i^t), y_i^t)\},$$

Reducing Mismatch of Distributions:

•Consider the learnt kernel **K** as a convex combination of given (base) kernels:

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^{M} d_m \mathbf{K}_m(\mathbf{x}_i, \mathbf{x}_j)$$

•Maximum Mean Discrepancy (MMD):

$$d_{\mathbf{K}}(D_s, D_t) = \|\frac{1}{n_s}\sum_{i=1}^{n_s}\phi(\mathbf{x}_i^s) - \frac{1}{n_u+n_l}\sum_{i=1}^{n_u+n_l}\phi(\mathbf{x}_i^t)\|^2_{\mathcal{H}} = tr(\Phi^T\Phi\mathbf{S}) = tr(\mathbf{KS}),$$

•Reducing mismatch is translated to the choice of optimal weights d_m:

$$\Omega(d_{\mathbf{K}}(D_s, D_t)) = (tr((\sum_{m=1}^{M} d_m\mathbf{K}_m)\mathbf{S}))^2 = \mathbf{d}^T\mathbf{p}\mathbf{p}^T\mathbf{d},$$

Implementation:

•Cost function:

$$\min_{\mathbf{d}} J(\mathbf{d}) \quad \text{with} \quad d_m \geq 0, \sum_{m=1}^{M} d_m = 1,$$

$$\text{where } J(\mathbf{d}) = \min_{f\in\mathcal{H}}\{\ \beta\|f\|^2_{\mathcal{H}} + \frac{\lambda_s}{n_s}\sum_{i=1}^{n_s}(\mathcal{K}_{P_s} f(\mathbf{x}_i^s) - y_i^s)^2 + \frac{\lambda_t}{n_l}\sum_{i=1}^{n_l}(\mathcal{K}_{P_t} f(\mathbf{x}_i^t) - y_i^t)^2 + \theta\mathbf{d}^T\mathbf{p}\mathbf{p}^T\mathbf{d}\}.$$

Experiments

Synthetic Example

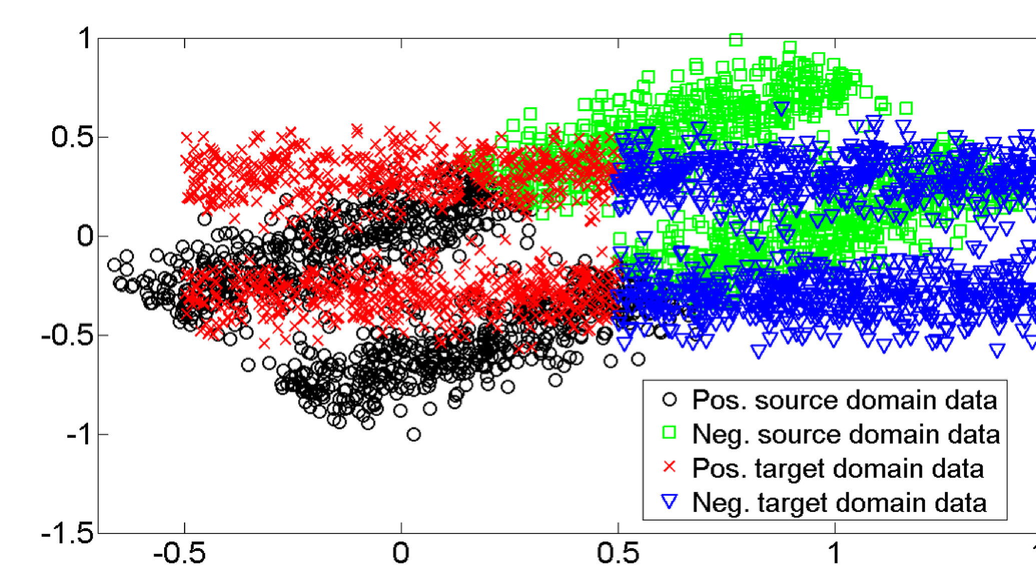


Table 1: Classification accuracy and standard error.

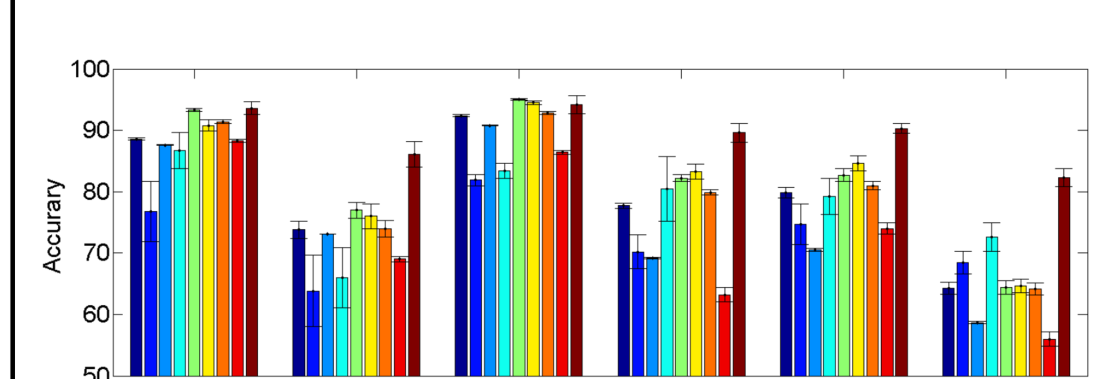
SVM-st	Fred-st	DTMKL-f	TFMKL
87.86±0.02	90.95±0.01	91.07±0.01	96.02 ±0.11

Object Recognition

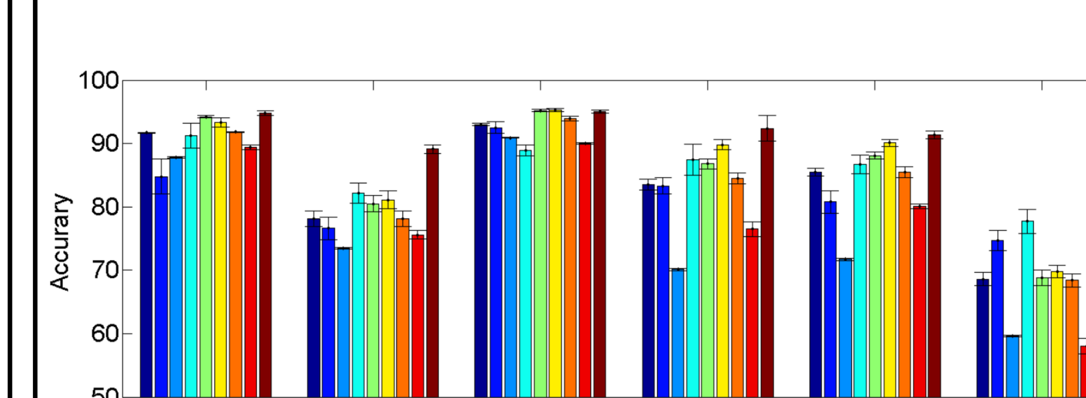
Table 2: Classification accuracy and standard error on the 12 cross-domain object categorization data sets. The best recognition rates are in red and bold font. The second best recognition rates are in blue and italics font.

Data Set	Standard Learning				Transfer Learning				
	SVM-st	SVM-t	Fred-st	Fred-t	DTMKL-f	MMDT	KMM	GFK	TFMKL
A→D	45.6±0.7	55.9±0.8	46.4±0.8	*58.2±0.9*	46.4±0.7	56.7±1.3	49.0±0.7	50.7±0.8	**62.1±0.9**
A→W	46.3±0.7	62.4±0.9	50.3±0.9	*69.2±1.1*	48.6±0.8	64.6±1.2	47.4±0.9	58.6±1.0	**69.8±0.9**
A→C	39.8±0.3	32.0±0.8	39.0±0.4	35.3±0.9	40.5±0.4	36.4±0.8	*40.8±0.3*	36.0±0.5	**43.6±0.6**
D→A	41.6±0.4	45.7±0.9	51.5±0.5	*51.7±0.8*	42.6±0.4	46.9±1.0	42.6±0.4	45.7±0.6	**52.9±0.7**
D→W	77.4±0.6	62.1±0.8	**82.4±0.3**	65.4±0.8	76.1±0.5	74.1±0.8	*78.5±0.6*	76.5±0.5	77.8±0.6
D→C	35.9±0.4	31.7±0.6	*37.7±0.4*	35.2±0.8	37.5±0.5	34.1±0.8	36.9±0.4	32.9±0.5	**37.9±0.4**
W→A	43.4±0.3	45.6±0.7	49.5±0.5	*52.3±0.7*	45.3±0.4	47.7±0.9	44.4±0.3	44.1±0.4	**54.4±0.4**
W→D	69.5±0.8	55.1±0.8	**73.2±0.6**	58.2±0.8	69.9±1.1	67.0±1.1	70.4±0.8	*70.5±0.7*	67.7±0.9
W→C	36.4±0.4	30.4±0.7	37.5±0.3	34.6±0.9	**37.8±0.4**	32.2±0.8	*37.6±0.4*	31.1±0.6	36.1±0.8
C→A	46.9±0.6	45.3±0.9	*52.1±0.6*	49.9±1.0	49.5±0.9	49.4±0.8	48.0±0.6	44.7±0.8	**54.2±0.9**
C→D	52.0±1.0	55.8±0.9	55.7±0.9	57.2±1.1	53.1±0.9	56.5±0.9	53.0±1.0	*57.7±1.1*	**59.2±1.1**
C→W	53.6±0.9	60.3±1.0	60.6±1.2	62.0±1.1	55.4±1.1	*63.8±1.1*	54.6±0.9	63.7±0.8	**68.2±0.9**
Mean	49.0±0.6	48.5±0.8	*52.9±0.6*	52.3±0.8	50.2±0.7	52.5±1.0	50.3±0.6	51.0±0.7	**57.0±0.8**

Text Classification



(a) 20-Newsgroups: s=5



(b) 20-Newsgroups: s=10