电子科技大学
University of Electronic Science and Technology of China

Representation-based Transfer Learning and Some Advances

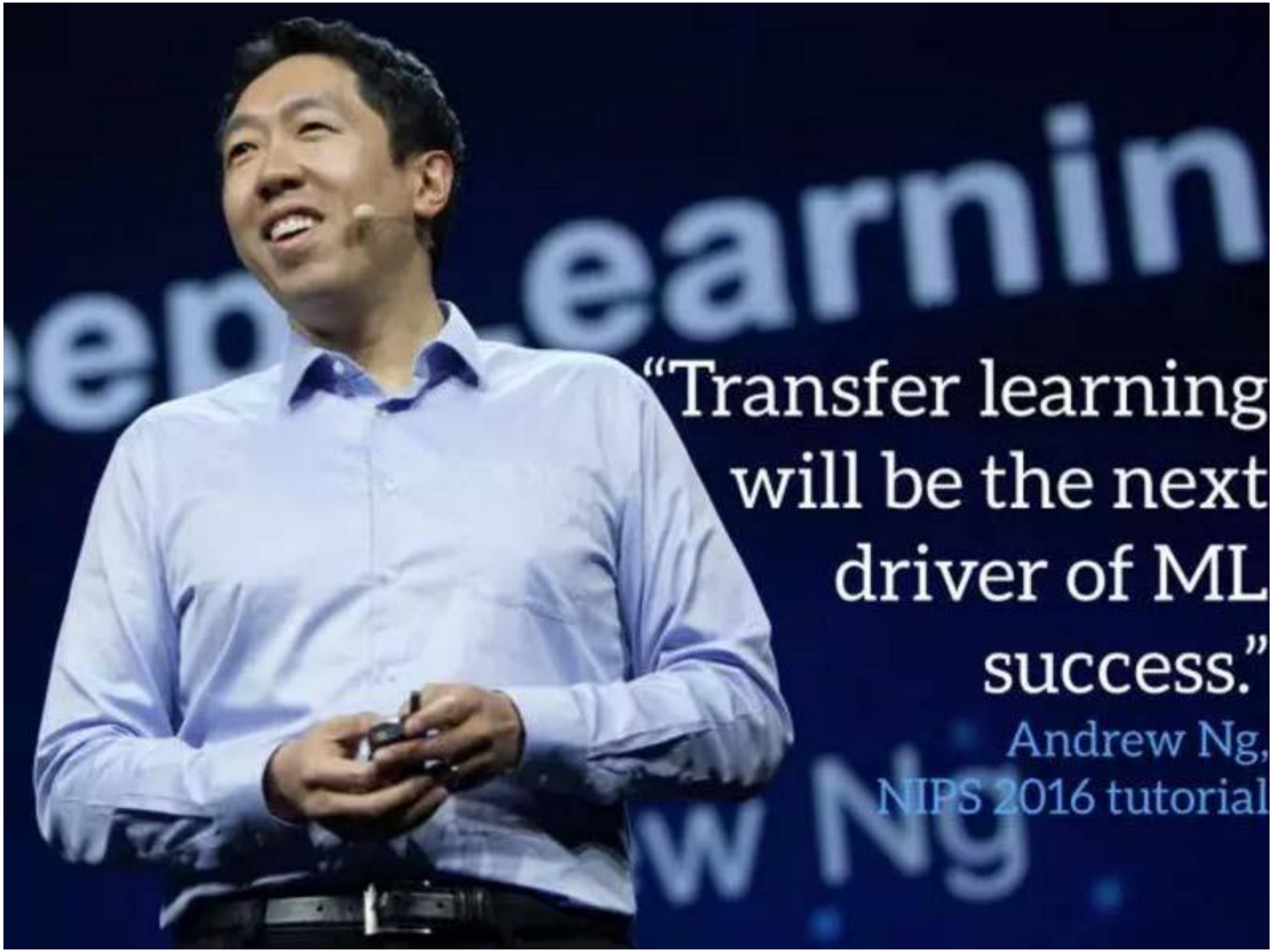
Wei Han

Data Mining Lab,
Big Data Research Center, UESTC
Email：wei.hb.han@gmail.com

Outline

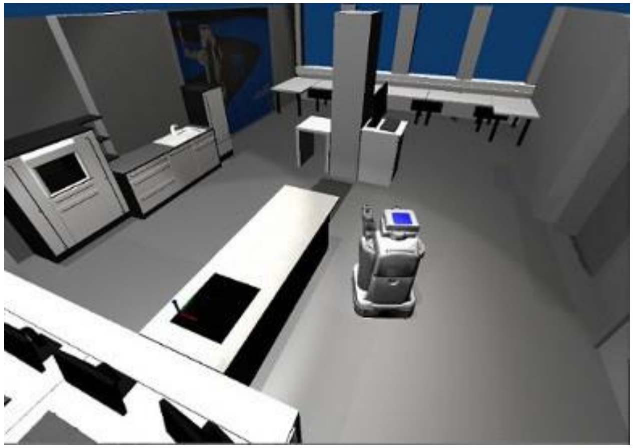
Motivation

1.1. Exciting foresight



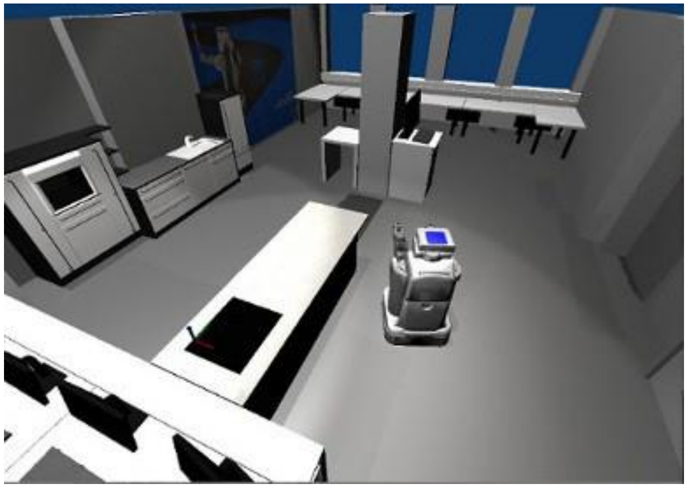
1.2. A Motivating Example

- Goal: to train a robot to accomplish Task $\boldsymbol{T}_1$ in an indoor environment $\boldsymbol{E}_1$ using machine learning techniques:
  - **Sufficient training data** required: sensor readings to measure the environment as well as **human supervision**, i.e. labels
  - A predictive model can be learned, and used in **the same** environment



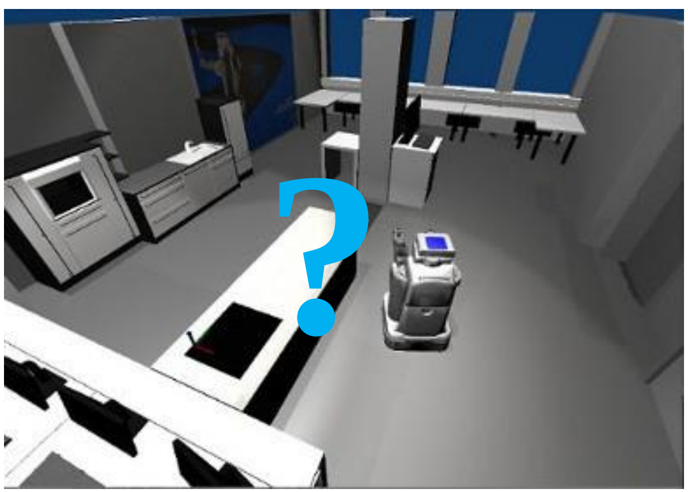
Task $\boldsymbol{T}_1$ in environment $\boldsymbol{E}_1$

1.2. A Motivating Example



Task $\boldsymbol{T}_1$

Environment changes $\boldsymbol{E}_2$



Task $\boldsymbol{T}_2$

New robot

1.3. Motivation

- People is of capacity to learn **quickly and precisely** with priori knowledge of target domain and the knowledge transferred from the similar but different (source) domain

- Performance of traditional machine learning techniques highly relies on whether **sufficient labeled data** is available to build a predictive model

- When **environment changes** (e.g., new domain or new task), the learned predictive model performs poorly

Definition

2.1.1. Intuitive View

- Inspired by human's transfer of learning ability

- The ability of a system to recognize and apply knowledge and skills learned in previous domains/tasks to novel tasks/domains, which share some commonality

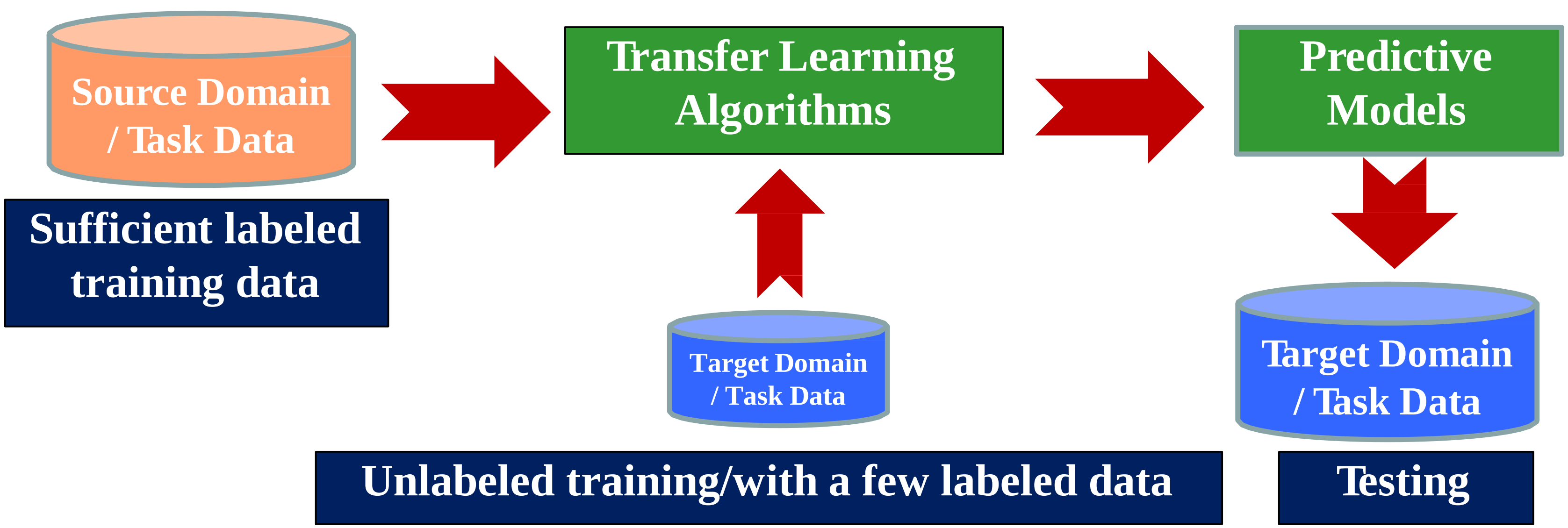
2.1.2. Formal Concepts

- Common concepts of TL
  - **Domain:** A domain D consists of two components: a feature space X and a marginal probability distribution $P(X)$, where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$
  - **Task:** A task consists of two components: a label space Y and an objective predictive function $f(\cdot)$ (denoted by $T = \{y, f(\cdot)\}$)

- Formal Definition of TL
  - **Condition:** Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T
  - **Goal:** Transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in D_T using the knowledge in D_S and T_S
  - **Limitation:** where $D_S \neq D_T$, or $T_S \neq T_T$

2.1.3. Further Understanding

- Given a target domain/task, transfer learning aims to
  1) identify the <u>**commonality**</u> between the target domain/task and previous domains/tasks
  2) transfer <u>**knowledge**</u> from the previous domains/tasks to the target one such that human supervision on the target domain/task can be dramatically reduced.



2.2.1. Category of Transfer Learning

- The category of transfer learning problems

Learning Setting	Source and Target Domain	Source and Target Task
Traditional Machine Learning	The same	The same
Inductive Transfer Learning/ Unsupervised Transfer Learning	The same	Different but related
	Different but related	Different but related
Transductive Transfer Learning	Different but related	The same

2.2.1. Category of Transfer Learning

- Basic ideas of transfer learning approaches

Transfer learning approaches	Description
Instance-transfer	To re-weight some labeled data in a source domain for use in the target domain
Feature-representation-transfer	Find a “good” feature representation that reduces difference between a source and a target domain or minimizes error of models
Model(Parameter)-transfer	Discover shared parameters or priors of models between a source domain and a target domain
Relational-knowledge-transfer	Build mapping of relational knowledge between a source domain and a target domain.

2.2.1. Category of Transfer Learning

数据挖掘实验室

Data Mining Lab

- Scope of approaches

	Inductive Transfer Learning	**Transductive Transfer Learning**	**Unsupervised Transfer Learning**
Instance-transfer	√	√	
Feature-representation-transfer	√	√	√
Model(Parameter)-transfer	√		
Relational-knowledge-transfer	√		

2.2.1. Other Categories

- Homogeneous/Heterogeneous transfer learning

$$X_S \neq X_T, P(X_S) \neq P(X_T), Y_S \neq Y_T$$

- Symmetric/Asymmetric feature-based transfer learning

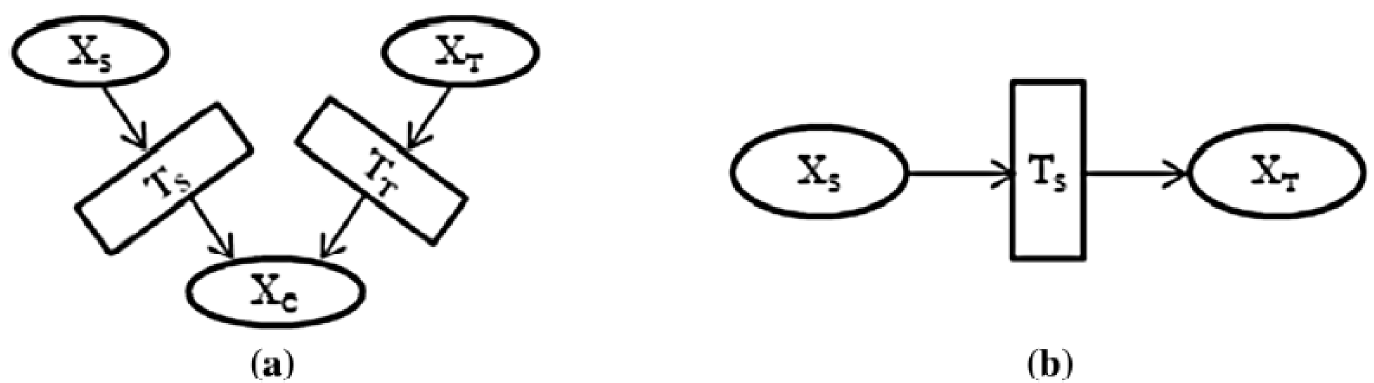


Fig. 1 **a** The symmetric transformation mapping (T_S and T_T) of the source (X_S) and target (X_T) domains into a common latent feature space. **b** The asymmetric transformation (T_T) of the source domain (X_S) to the target domain (X_T)

2.2.2. Confusing Related Areas

- <u>Directly Similar</u>:
  - Domain adaptation
  - Multi-view learning
  - Zero-shot/Few-shot learning
  - Multi-task learning

- <u>Indirectly Similar</u>:
  - Learning to learn
  - Label embedding/Attribute
  - Continuous learning
  - Lifelong learning

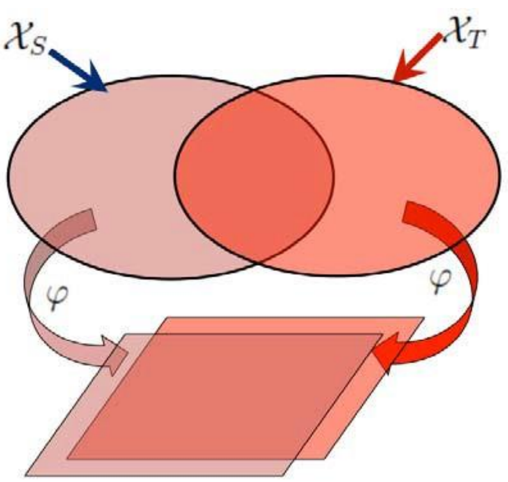
Representation-based Transfer Learning

3.1.1. Overview

- **<u>Assumption</u>**： Source and target domains only have some overlapping features

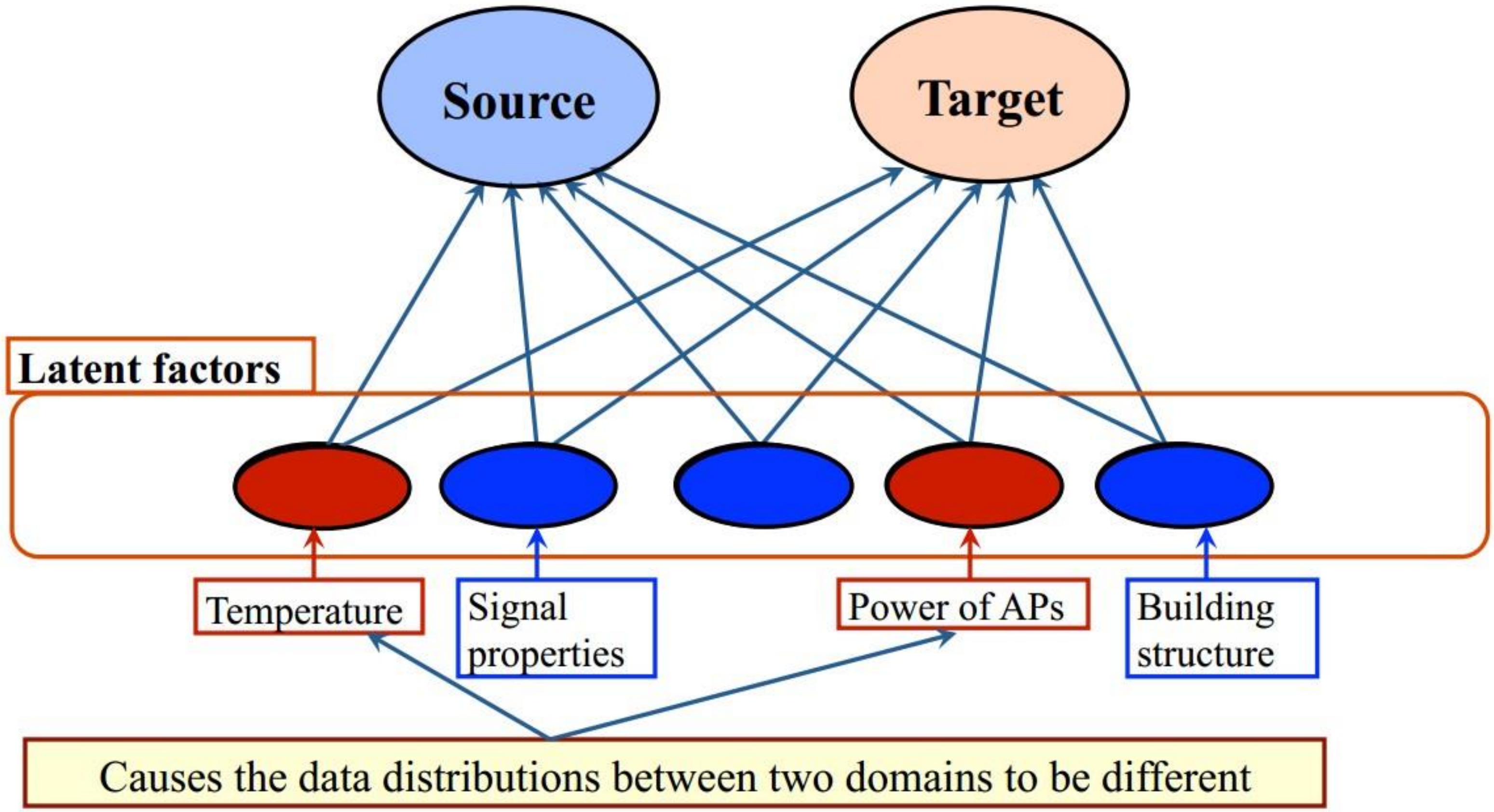
- **<u>Idea</u>**： Through feature transformation, the data in two domain are merged into one feature space

- **Classical Methods**：
  - ➢ Transfer component analysis (TCA) [Pan, TKDE-11]
  - ➢ Geodesic flow kernel (GFK) [Duan, CVPR-12]
  - ➢ Transfer kernel learning [Long, TKDE-15]
  - ➢ ……



3.1.2. Transfer Component Analysis

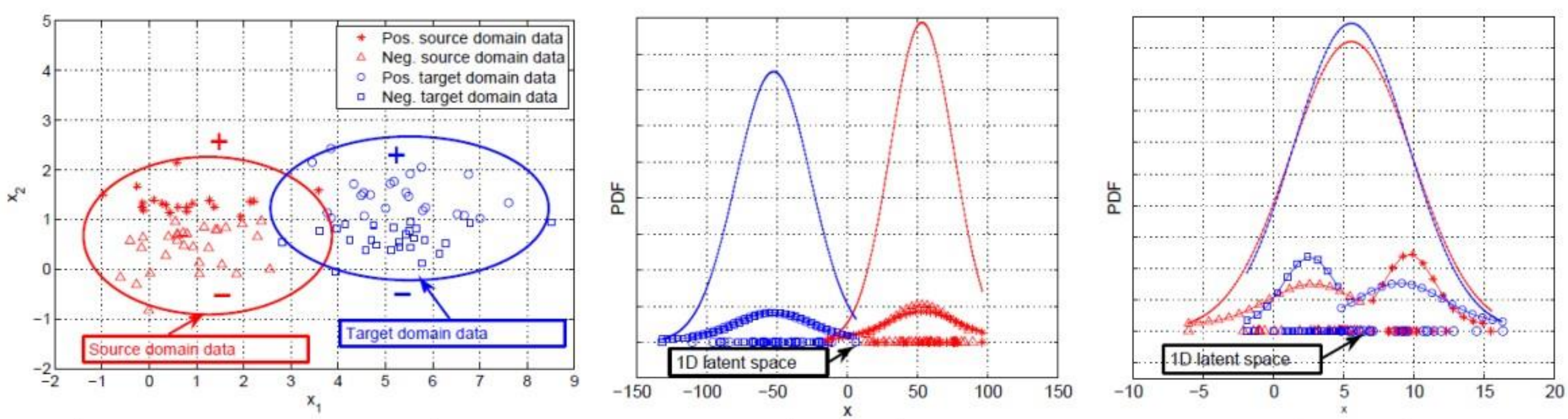
- **<u>Main idea</u>**: the learned φ should map the source domain and target domain data to a latent space spanned by the factors that reduce domain distance as well as preserve data structure



- **<u>High level optimization problem</u>**

$$\begin{aligned} \min_{\varphi}\ & \boxed{\mathrm{Dist}\big(\varphi(X_S),\varphi(X_T)\big)} + \lambda\Omega(\varphi) \\ \text{s.t.}\ & \text{constraints on } \varphi(X_S) \text{ and } \varphi(X_T) \end{aligned}$$

Maximum Mean Discrepancy (MMD)



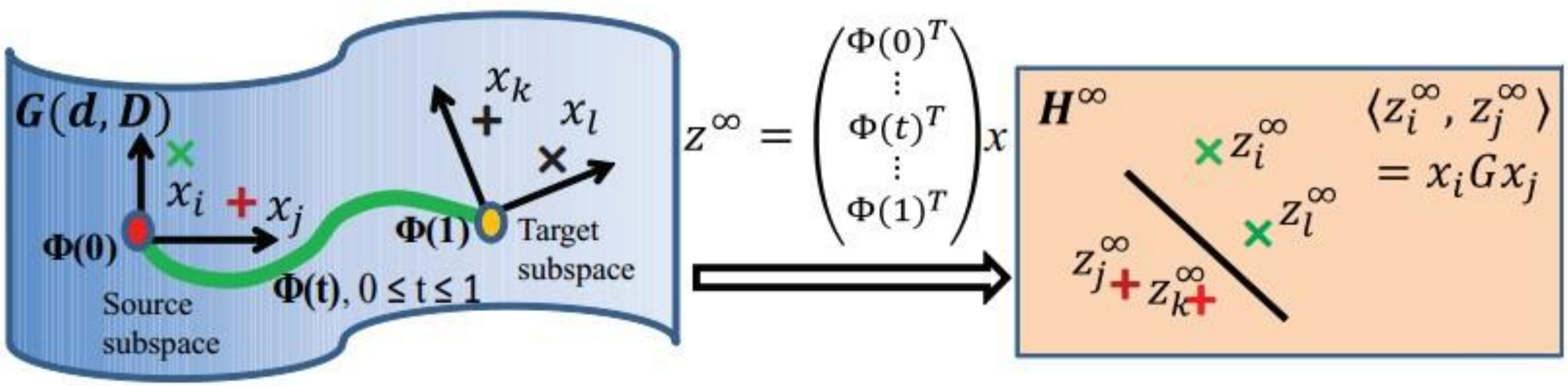
Original feature space

PCA

TCA

3.1.3. Geodesic Flow Kernel

<u>Idea</u>: Data are mapped into manifold space, and the distance between two domain is measured and minimized



3.2.1. Leverage Domain-specific Info

TL for Sentiment Classification

Electronics	Video Games
(1) **Compact**; easy to operate; very ***good*** picture quality; looks **sharp**!	(2) A very ***good*** game! It is action packed and full of ***excitement***. I am very much **hooked** on this game.
(3) I purchased this unit from Circuit City and I was very ***excited*** about the quality of the picture. It is really ***nice*** and **sharp**.	(4) Very **realistic** shooting action and ***good*** plots. We played this and were **hooked**.
(5) It is also quite **blurry** in very dark settings. I will ***never_buy*** HP again.	(6) The game is so **boring**. I am extremely ***unhappy*** and will probably ***never_buy*** UbiSoft again.

3.2.1. Leverage Domain-specific Info

- Three different types of features
  - Source domain (***Electronics***) specific features, e.g., ***compact, sharp, blurry***
  - Target domain (***Video Game***) specific features, e.g., ***hooked***, ***realistic***, ***boring***
  - Domain independent features (pivot features), e.g., ***good, excited***, ***nice***, ***never_buy***

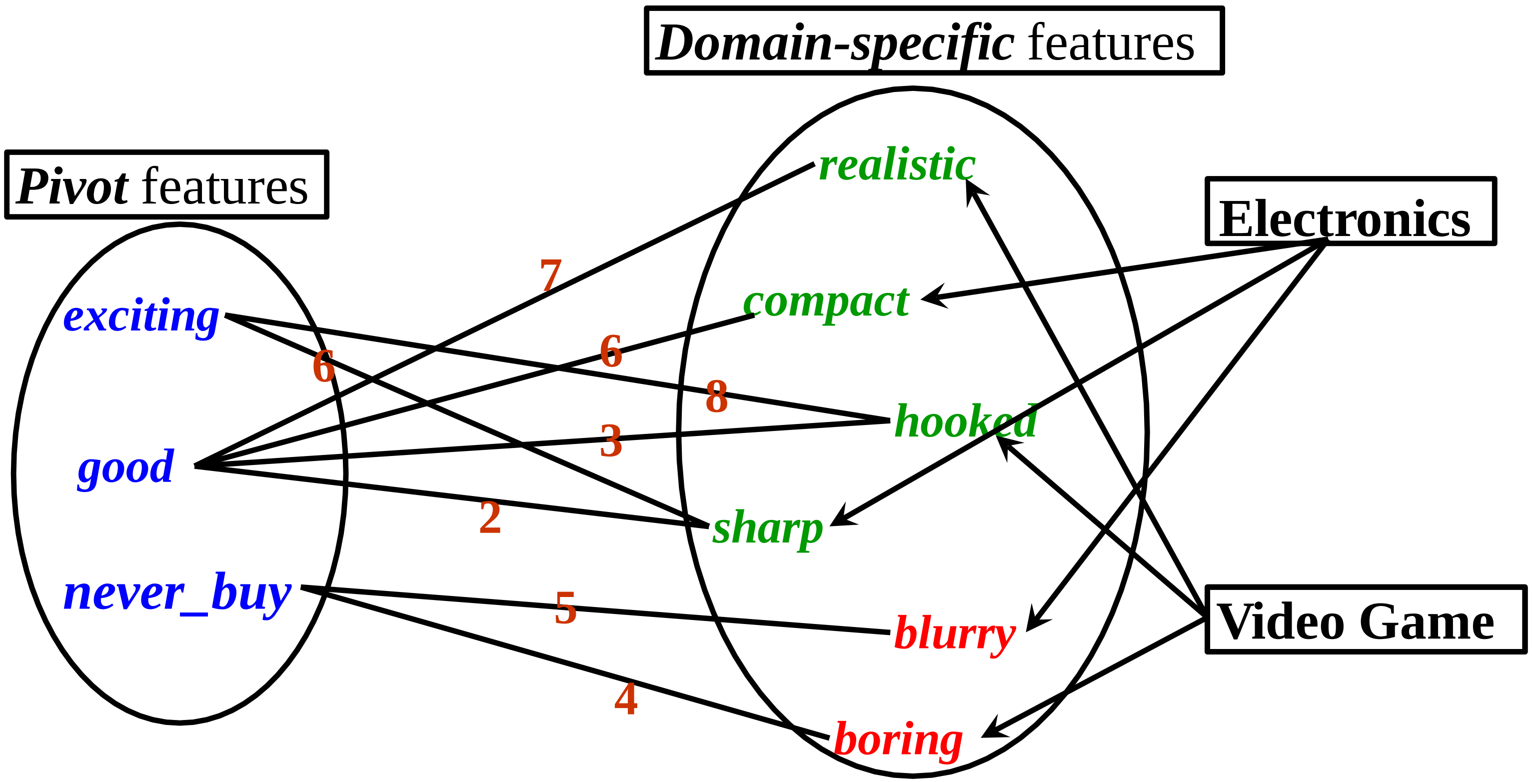
3.2.1. Leverage Domain-specific Info

- Intuition
  - Use **pivot** features as a bridge to connect domain- specific features
  - Model correlations between **pivot** features and domain-specific features
  - Discover new shared features by exploiting the feature correlations

- How to select **pivot** features?
  - Term frequency on both source and target domain data.
  - Mutual information between features and source domain labels
  - Mutual information on between features and domains

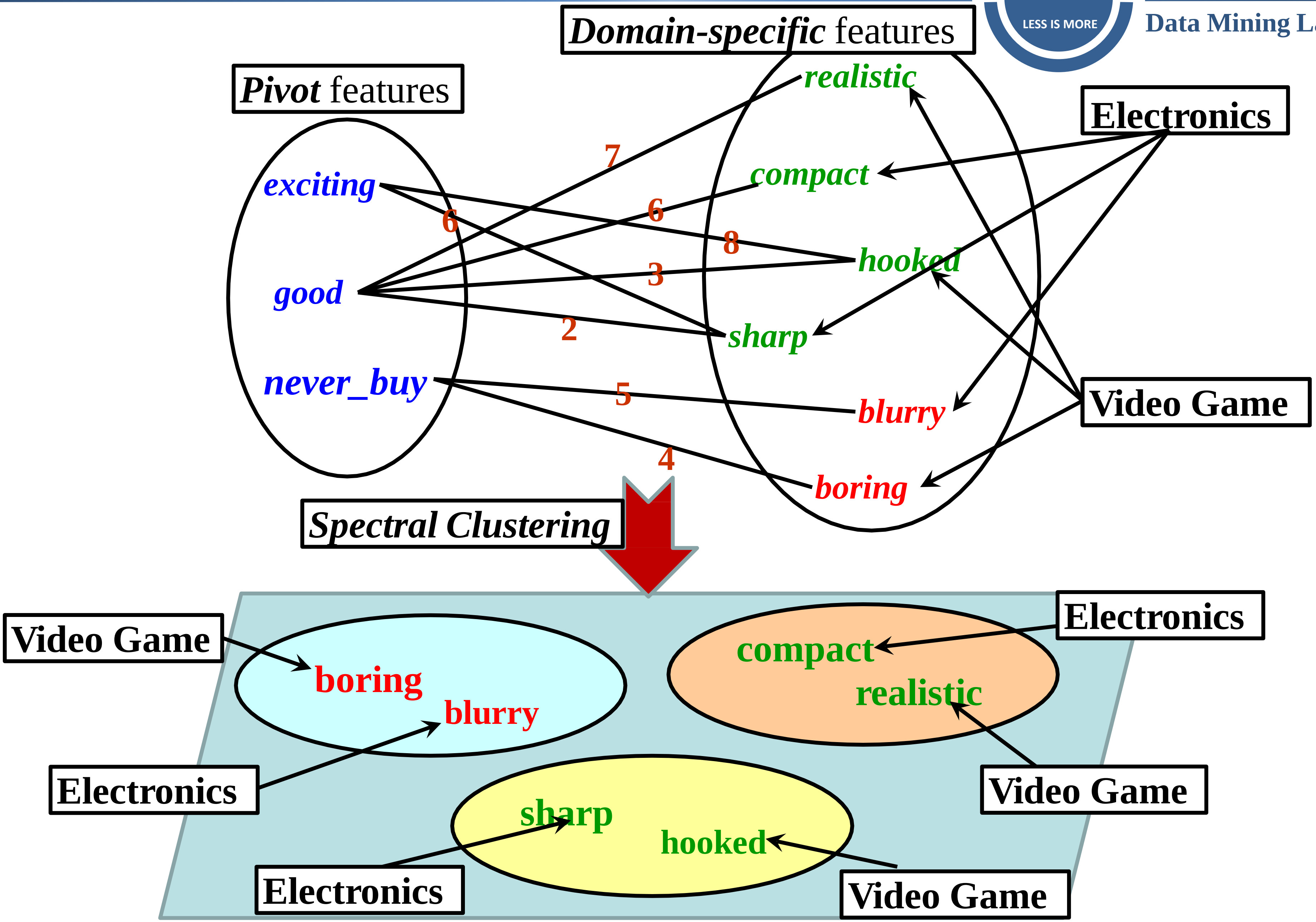
数据挖掘实验室
Data Mining Lab

Spectral Feature Alignment (SFA)



- If two ***domain-specific*** words have connections to more common ***pivot*** words in the graph, they tend to be aligned or clustered together with a higher probability.
- If two ***pivot*** words have connections to more common ***domain-specific*** words in the graph, they tend to be aligned together with a higher probability.

3.2.1. Leverage Domain-specific Info



3.2.1. Leverage Domain-specific Info

Electronics

	sharp/hooked	compact/realistic	blurry/boring
	1	1	0
	1	0	0
	0	0	1

Training

$$y = f(x) = \mathrm{sgn}(w \cdot x^T), \qquad w = [1, 1, -1]$$

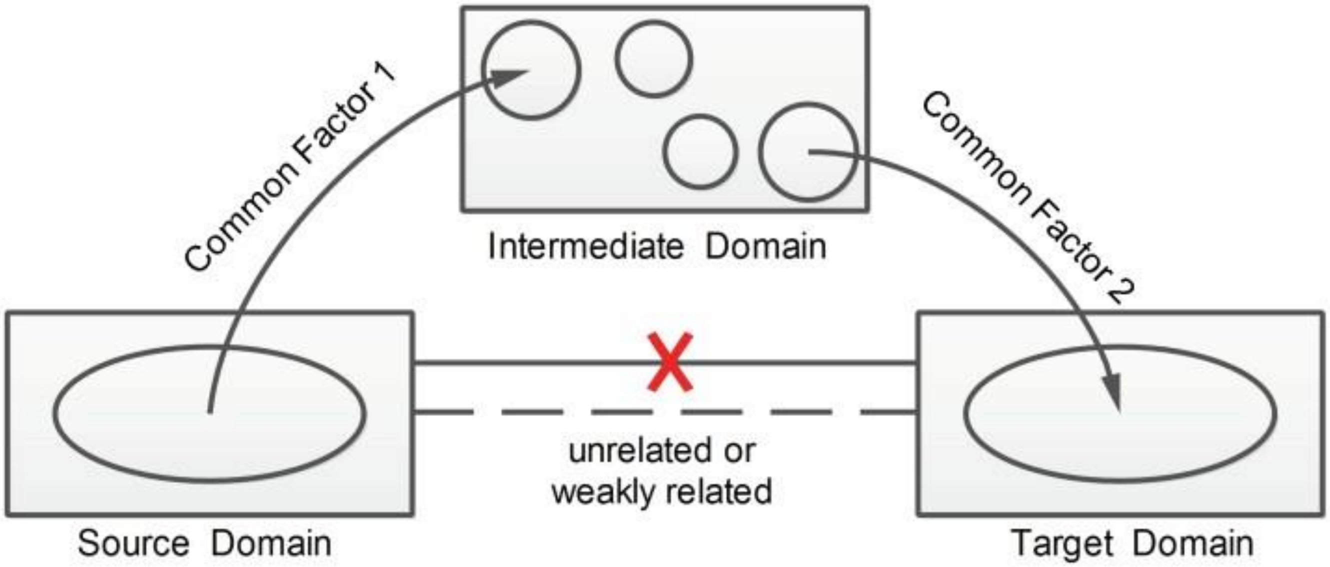
Prediction

Video Game

	sharp/hooked	compact/realistic	blurry/boring
	1	0	0
	1	1	0
	0	0	1

Transitive transfer learning [Tan, KDD-15]

In two dissimilar domains, intermediate domains are leveraged to help knowledge transform



How to select intermediate domain?

- Domain complexity

$$cplx(D) = \frac{|\{x|c(x) < t \times n\}|}{m}$$

The domain complexity is calculated as the percentage of long tail features that have low frequency.

- A-distance

$$dis_{\mathcal{A}}(\boldsymbol{D}_i, \boldsymbol{D}_j) = 2(1 - 2\min_{h \in \mathcal{H}} error(h|\boldsymbol{D}_i, \boldsymbol{D}_j))$$

The A-distance estimates the distribution difference of two sets of data samples that are drawn from two probability distributions

Intermediate Domain Selection

- Given a triple *tr* = {*S*, *D*, *T*}
- Take six measurements as features to construct a logistic regression model

feature	description
cplx_src (c_1)	source domain complexity
cplx_inter (c_2)	intermediate domain complexity
cplx_tar (c_3)	target domain complexity
$dis_{\mathcal{A}}^{si}$ (c_4)	a_distance between source and intermediate
$dis_{\mathcal{A}}^{st}$(c_5)	a_distance between source and target
$dis_{\mathcal{A}}^{it}$ (c_6)	a_distance between intermediate and target

- Estimate variables with MLE

Nonnegative Matrix tri-Factorization:

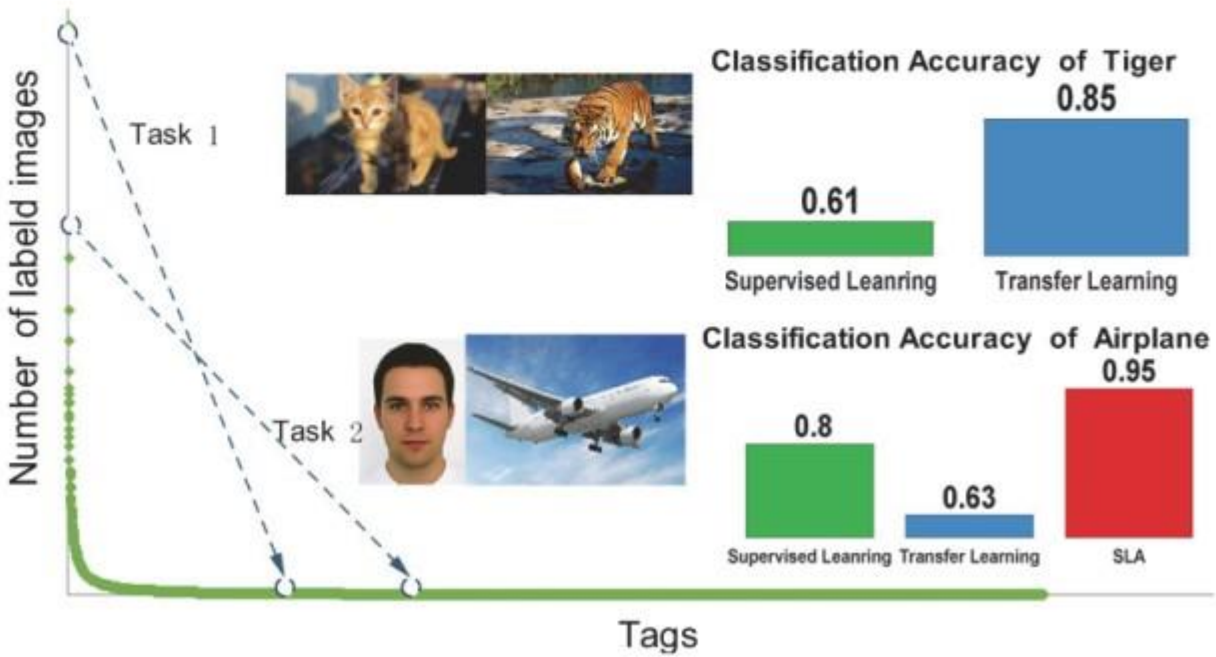
$$\mathcal{L}_{ST} = ||X_s - F_s A_s G_s|| + ||X_t - F_t A_t G_t||$$

$$= \left\| X_s - [F^1, F_s^2] \begin{bmatrix} A^1 \\ A_s^2 \end{bmatrix} G_s^T \right\| + \left\| X_t - [F^1, F_t^2] \begin{bmatrix} A^1 \\ A_t^2 \end{bmatrix} G_t^T \right\|$$

- The matrix $F \in \mathbb{R}^{m \times p}$ indicates the information of feature clusters and p is the number of hidden feature clusters
- The matrix G $\in \mathbb{R}^{c \times n}$ is the instance cluster assignment matrix and c is the number of instance clusters
- $A \in \mathbb{R}^{p \times c}$ is the association matrix. c is the number of instance clusters or label classes

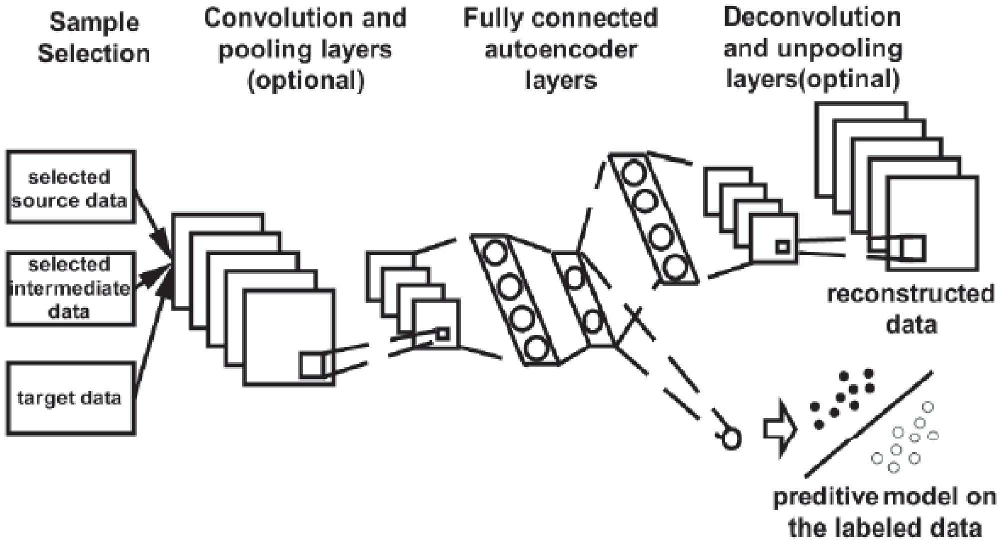
Distant domain TL [Tan, AAAI-17]

In the transferring between two highly dissimilar domains, the autoencoder is used to select unlabeled intermediate data from multiple auxiliary domain.



Idea: Reconstruction errors on the source domain data and the target domain data are both small

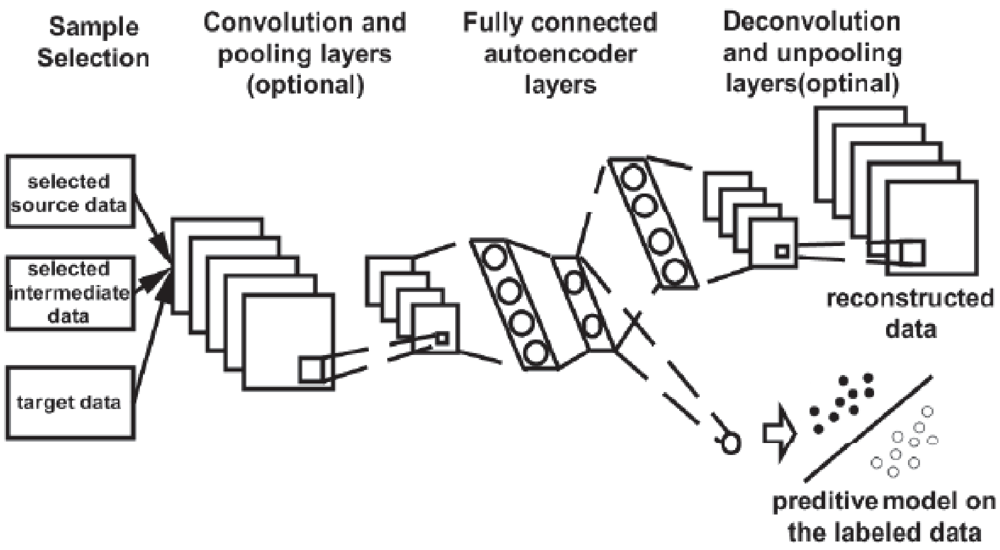
$$\mathcal{J}_1(f_e, f_d, v_S, v_T) = \frac{1}{n_S}\sum_{i=1}^{n_S} v_S^i \|\hat{x}_S^i - x_S^i\|_2^2 + \frac{1}{n_I}\sum_{i=1}^{n_I} v_I^i \|\hat{x}_I^i - x_I^i\|_2^2 + \frac{1}{n_T}\sum_{i=1}^{n_T} \|\hat{x}_T^i - x_T^i\|_2^2 + R(v_S, v_T), \quad (1)$$



3.2.3. Distant Transfer Learning

$$R(\boldsymbol{v}_S, \boldsymbol{v}_T) = -\frac{\lambda_S}{n_S}\sum_{i=1}^{n_S} v_S^i - \frac{\lambda_I}{n_I}\sum_{i=1}^{n_I} v_I^i$$

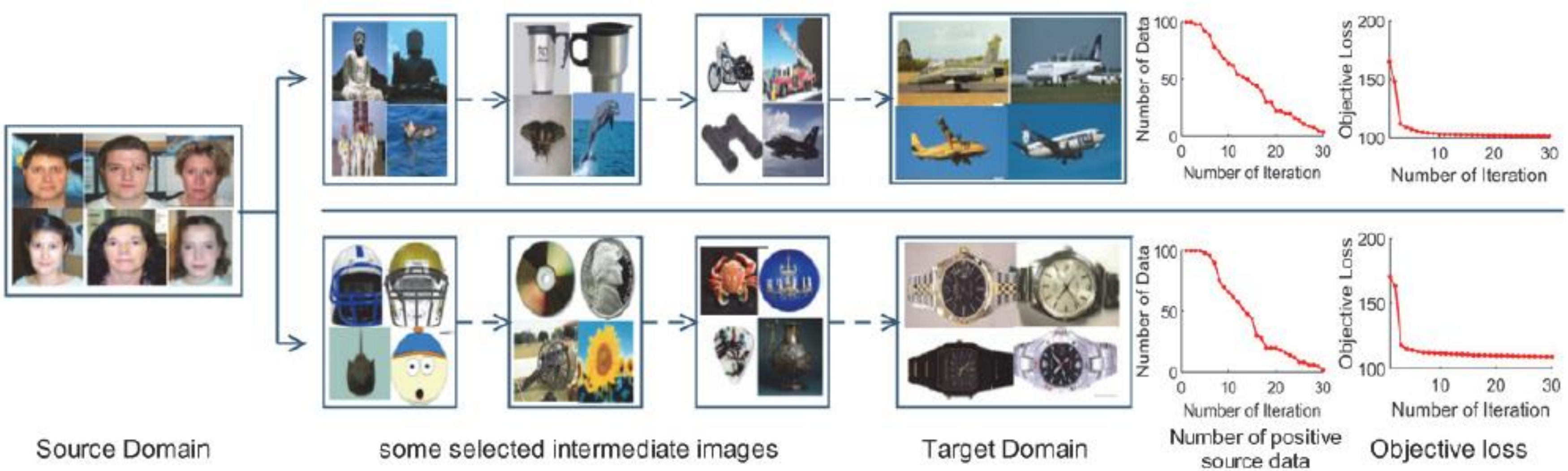
$$\mathcal{J}_2(f_c, f_e, f_d) = \frac{1}{n_S}\sum_{i=1}^{n_S} v_S^i \ell(y_S^i, f_c(\boldsymbol{h}_S^i)) + \frac{1}{n_T}\sum_{i=1}^{n_T} \ell(y_T^i, f_c(\boldsymbol{h}_T^i)) + \frac{1}{n_I}\sum_{i=1}^{n_I} v_I^i g(f_c(\boldsymbol{h}_I^i)), \quad (2)$$



3.2.3. Distant Transfer Learning

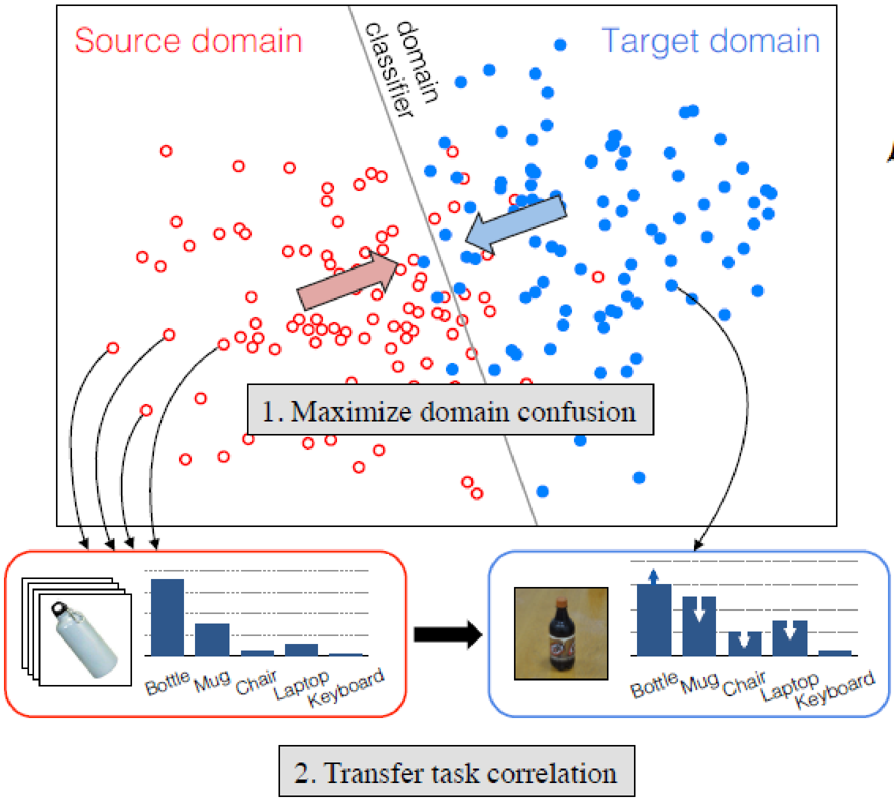
Results:

	SVM	DTL	GFK	LAN	ASVM	TTL	STL	SLA
'horse-to-face'	84 ± 2	88 ± 2	77 ± 3	79 ± 2	76 ± 4	78 ± 2	86 ± 3	$\mathbf{92} \pm 2$
'airplane-to-gorilla'	75 ± 1	62 ± 3	67 ± 5	66 ± 4	51 ± 2	65 ± 2	76 ± 3	$\mathbf{84} \pm 2$
'face-to-watch'	75 ± 7	68 ± 3	61 ± 4	63 ± 4	60 ± 5	67 ± 4	75 ± 5	$\mathbf{88} \pm 4$
'zebra-to-collie'	71 ± 3	69 ± 2	56 ± 2	57 ± 3	59 ± 2	70 ± 3	72 ± 3	$\mathbf{76} \pm 2$



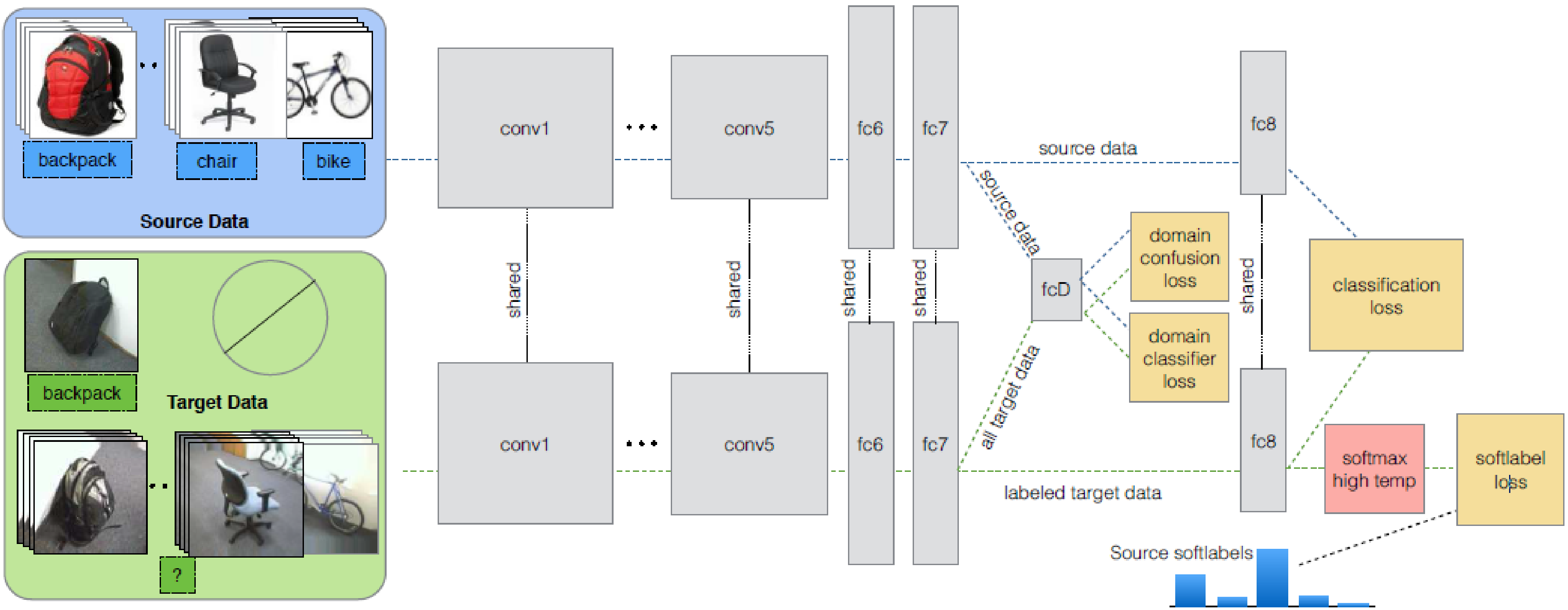
Simultaneous Deep Transfer Across Domains and Tasks [Tzeng, ICCV-15]

Simultaneously optimizes for **domain invariance** to facilitate domain transfer and uses a **soft label distribution** matching loss to transfer information between tasks



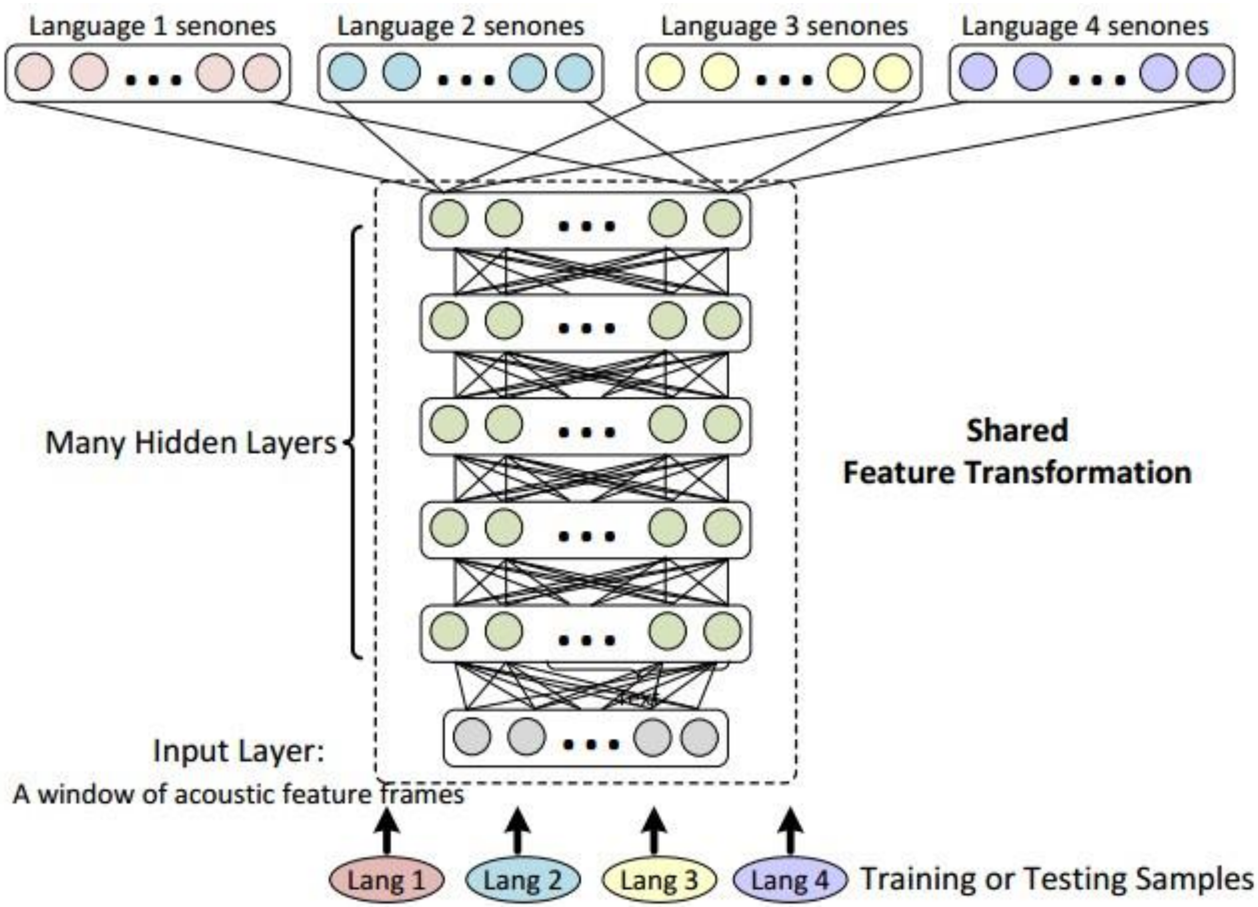
$$
\begin{aligned}
\mathcal{L}(x_S, y_S, x_T, y_T, \theta_D; \theta_{\text{repr}}, \theta_C) = \\
&\mathcal{L}_C(x_S, y_S, x_T, y_T; \theta_{\text{repr}}, \theta_C) \\
&+ \lambda \mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}) \\
&+ \nu \mathcal{L}_{\text{soft}}(x_T, y_T; \theta_{\text{repr}}, \theta_C).
\end{aligned}
$$

3.2.4. Label Embedding/Attribute for TL



SHL-MDNN [Huang, ICASSP-13]

Sharing hidden layers in DNN model, and learning different tasks by different softmax layers



Adversarial Discriminative Domain Adaptation [Tzeng, arXiv-17]

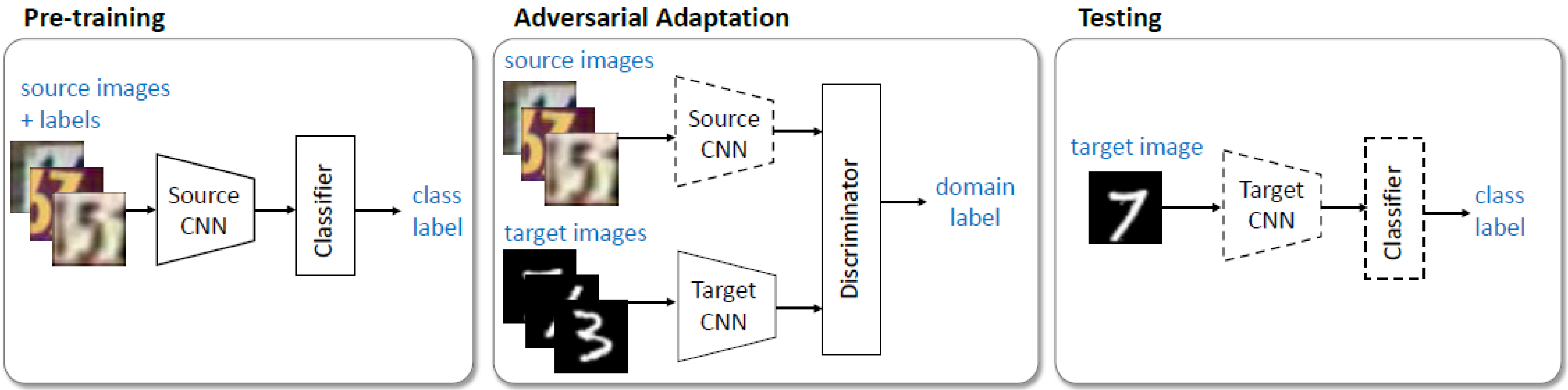


Figure 3: An overview of our proposed Adversarial Discriminative Domain Adaptation (ADDA) approach. We first pre-train a source encoder CNN using labeled source image examples. Next, we perform adversarial adaptation by learning a target encoder CNN such that a discriminator that sees encoded source and target examples cannot reliably predict their domain label. During testing, target images are mapped with the target encoder to the shared feature space and classified by the source classifier. Dashed lines indicate fixed network parameters.

GoGAN [Liu, NIPS-16]

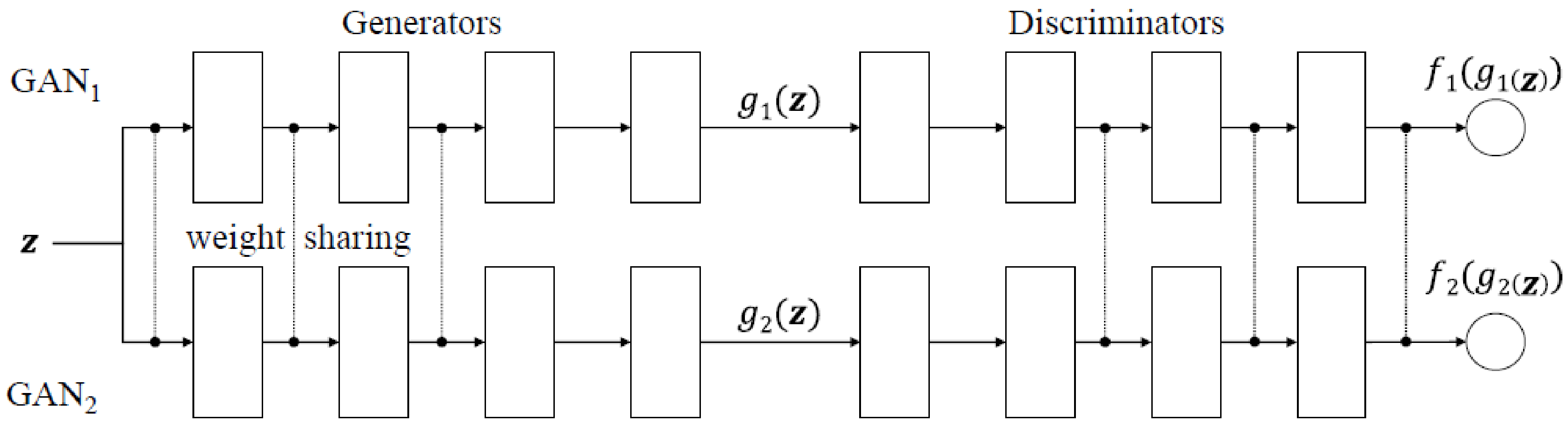


Figure 1: CoGAN consists of a pair of GANs: GAN_1 and GAN_2. Each has a generative model for synthesizing realistic images in one domain and a discriminative model for classifying whether an image is real or synthesized. We tie the weights of the first few layers (responsible for decoding high-level semantics) of the generative models, g_1 and g_2. We also tie the weights of the last few layers (responsible for encoding high-level semantics) of the discriminative models, f_1 and f_2. This weight-sharing constraint allows CoGAN to learn a joint distribution of images without correspondence supervision. A trained CoGAN can be used to synthesize pairs of corresponding images—pairs of images sharing the same high-level abstraction but having different low-level realizations.

CDD [Fu, arXiv-17]

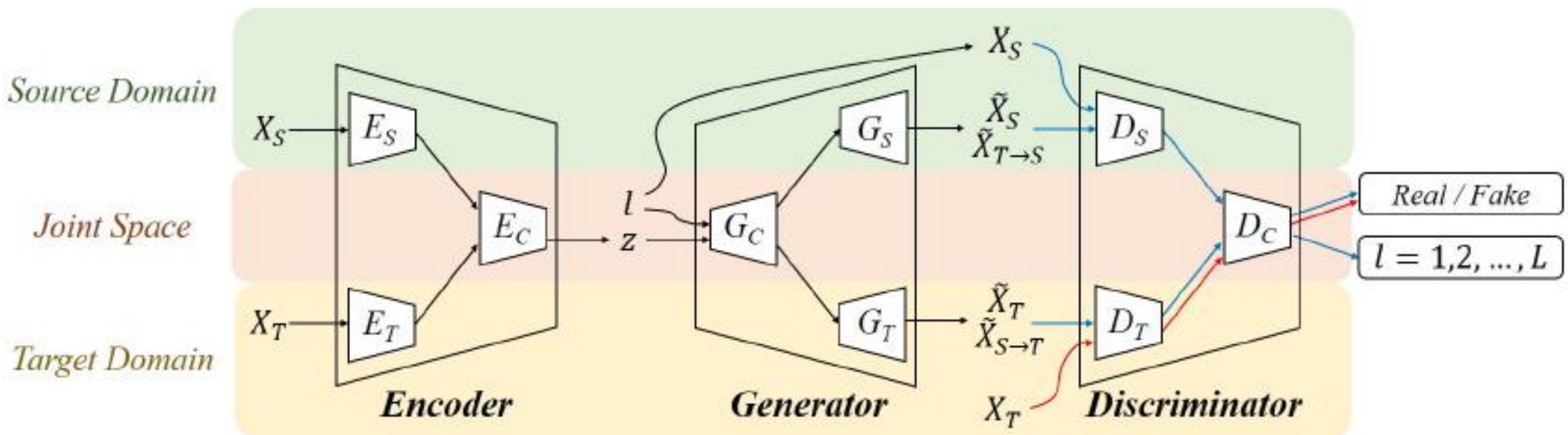
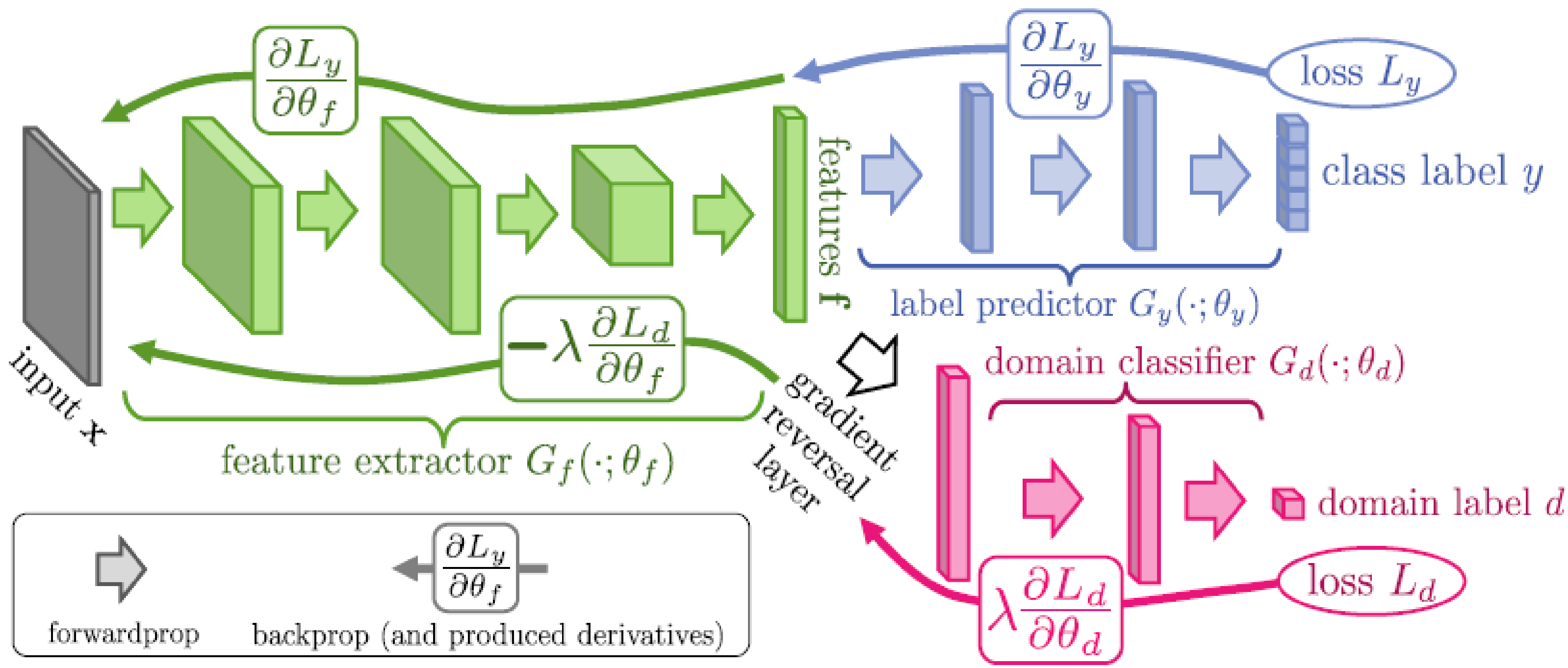


Figure 1: Our proposed architecture of Cross-Domain Disentanglement (CDD). The network components E_C, G_C, and D_C are shared by cross-domain data, while those with subscripts S and T are associated with data in the corresponding domain. Note that for X_T will be recognized as real/fake images due to the lack of ground truth labels l (shown in red).

Domain-Adversarial Training of Neural Networks [Ganin, JMLR-16]



SBAGA-GAN [Russo, arXiv-17]

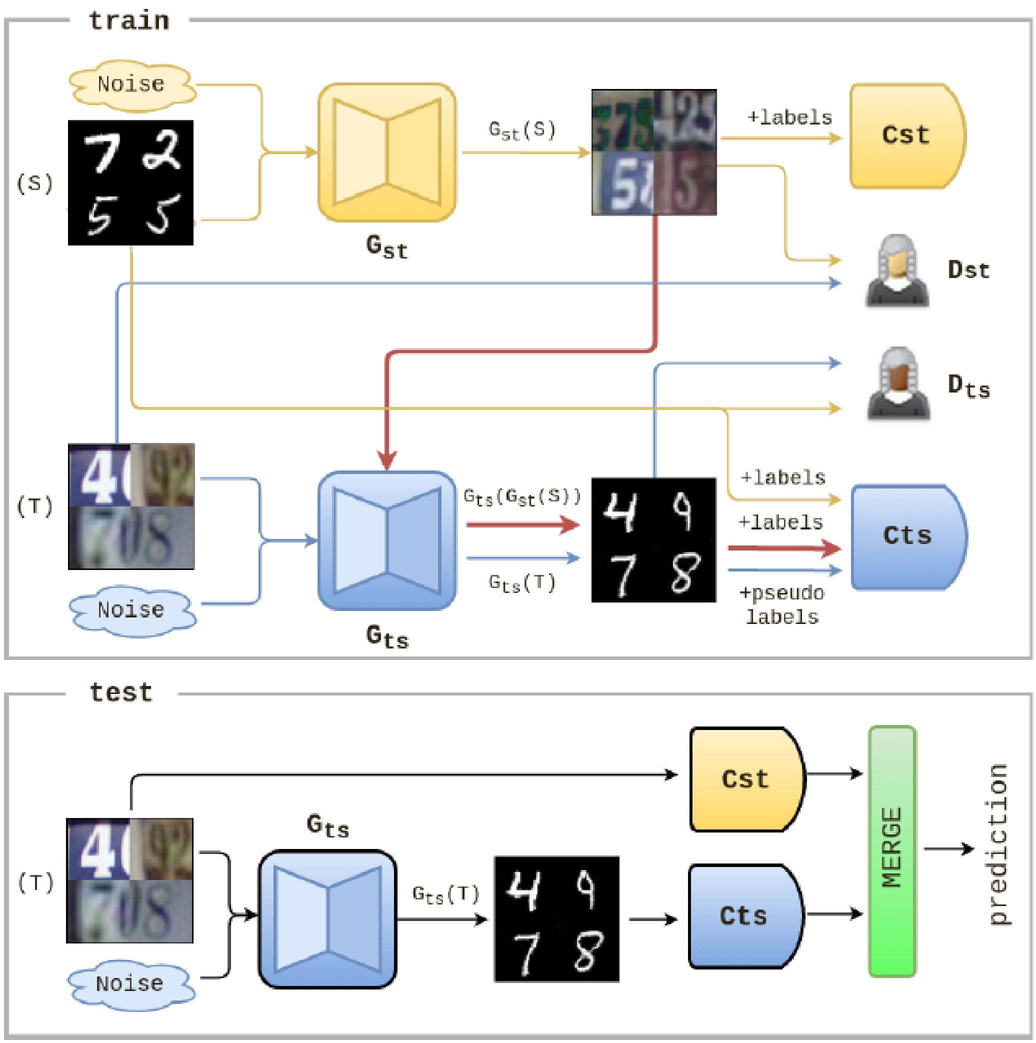


Figure 1: Schematic illustration of our SBADA-GAN. In the training phase, yellow lines represent data flow from source to target, while blue lines represent data flow from target to source. The red lines indicate the proposed *class consistency* condition that constraints a source image to keep its own label when passing sequentially through the two generators G_{st} and G_{ts} for domain transformations. During test phase the target samples are fed directly to C_{st} and transformed by G_{ts} before entering C_{ts}, to match the respectively classifiers trained data styles. The output of the two classifiers are merged by linear combination to get the final prediction.

Future Work

4.1. Open Questions

- Theoretical study beyond generalization error bound (Negative transfer learning, Domain similarity metric)
  - Given a source domain and a target domain, determine whether transfer learning should be performed
  - For a specific transfer learning method, given a source and a target domain, determine whether the method should be used for knowledge transfer
- Good (Interpretive) representation
- Transfer learning with plenty of source domains
- Online transfer learning
- Transfer learning for deep reinforcement learning
- Lifelong continuous learning

Thank you