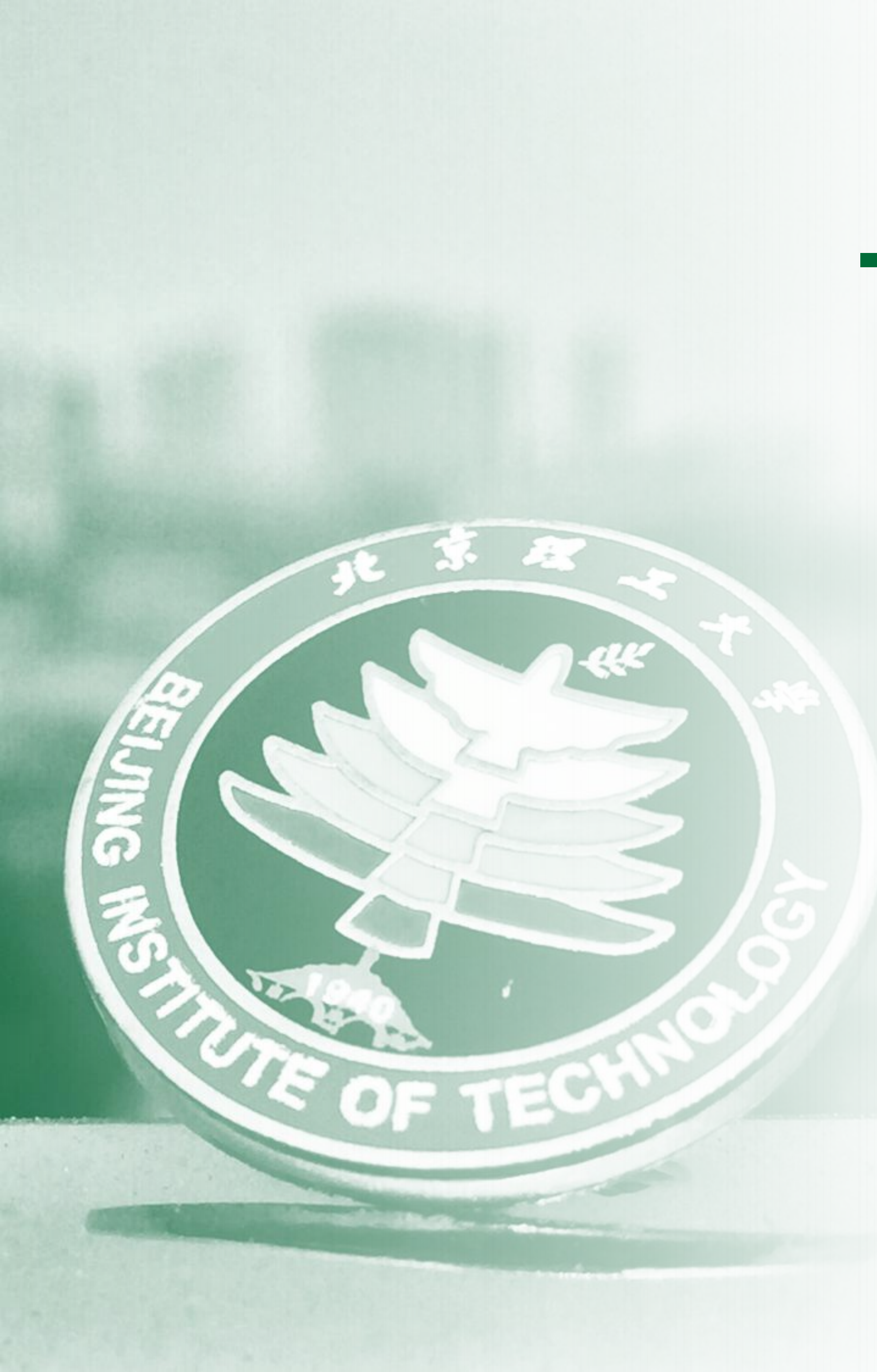
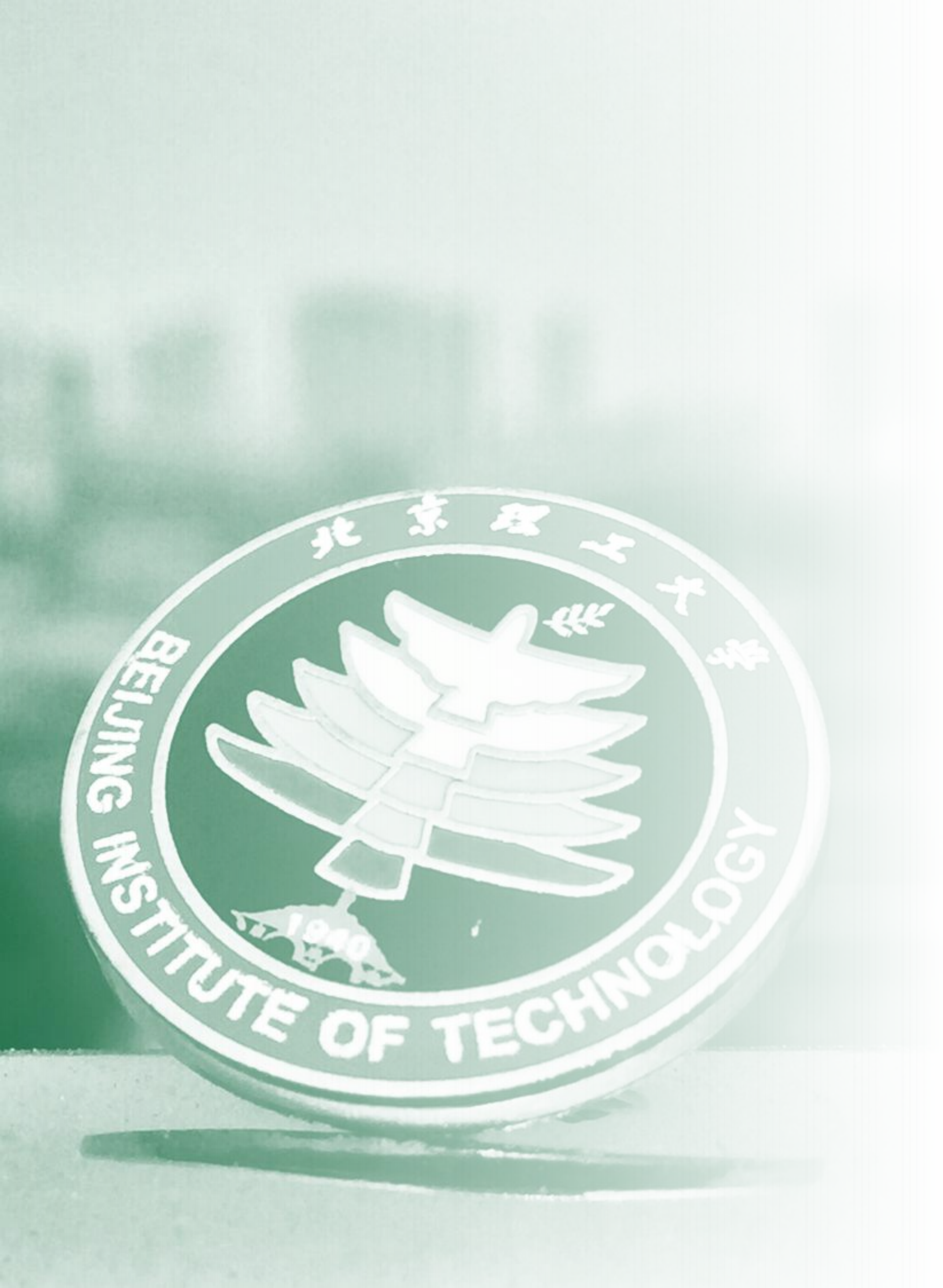
Progressive Feature Alignment for Unsupervised Domain Adaptation

苏丽敏

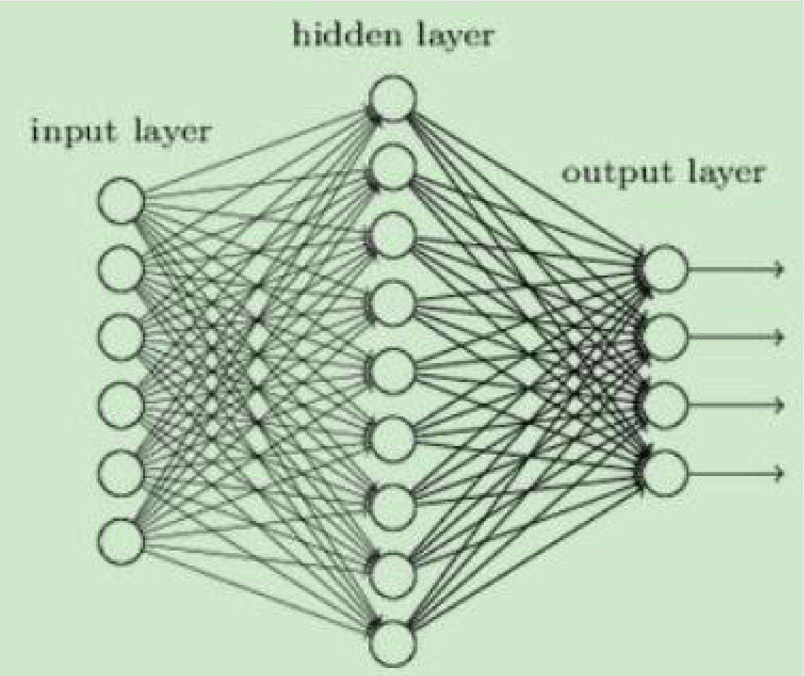


CONTENTS



1 BACKGROUND

BACKGROUND



Deep Neural Network

achieve impressive performance

need massive well-labeled training data

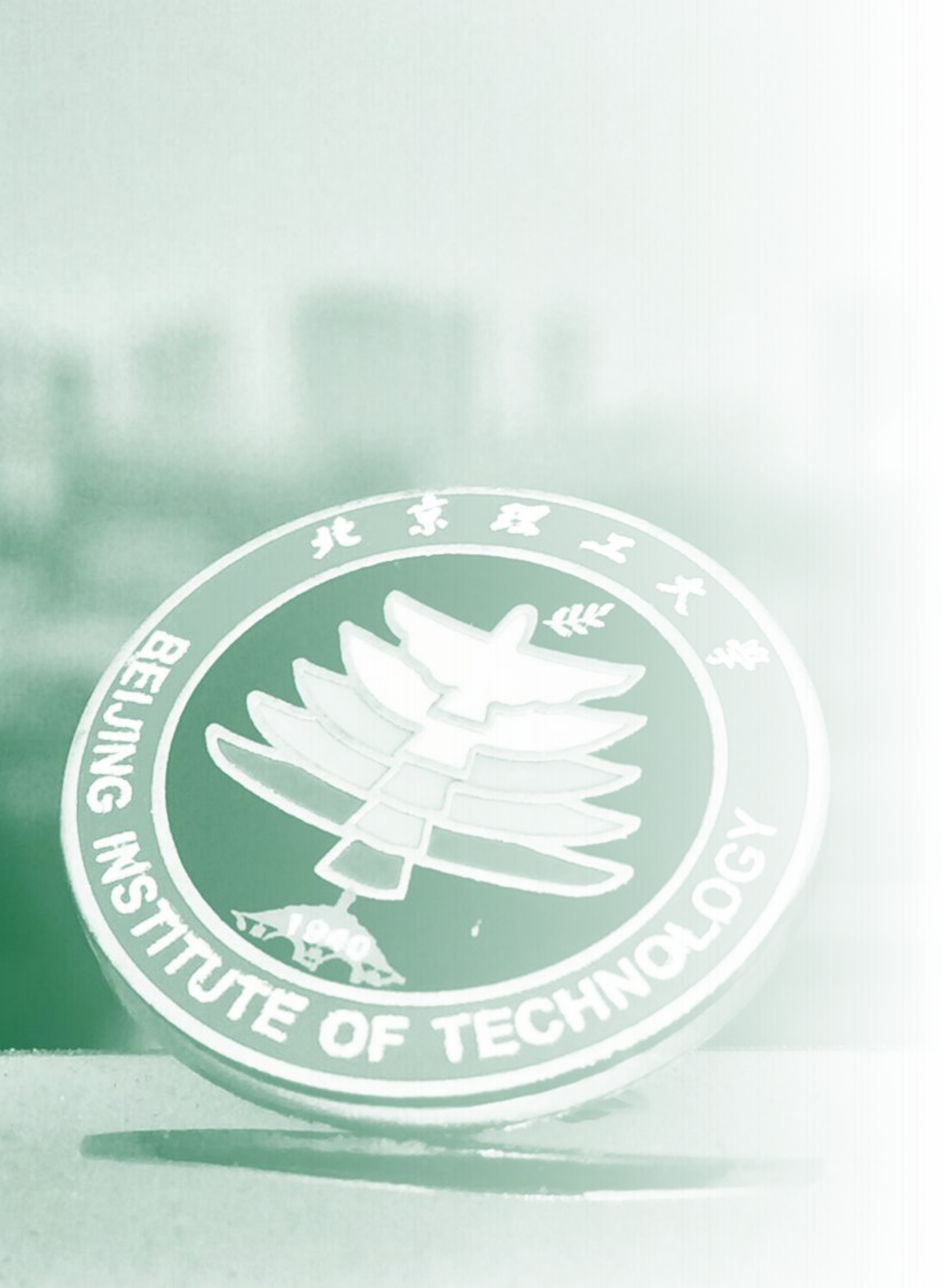
annotate sufficient label

time-consuming and expensive

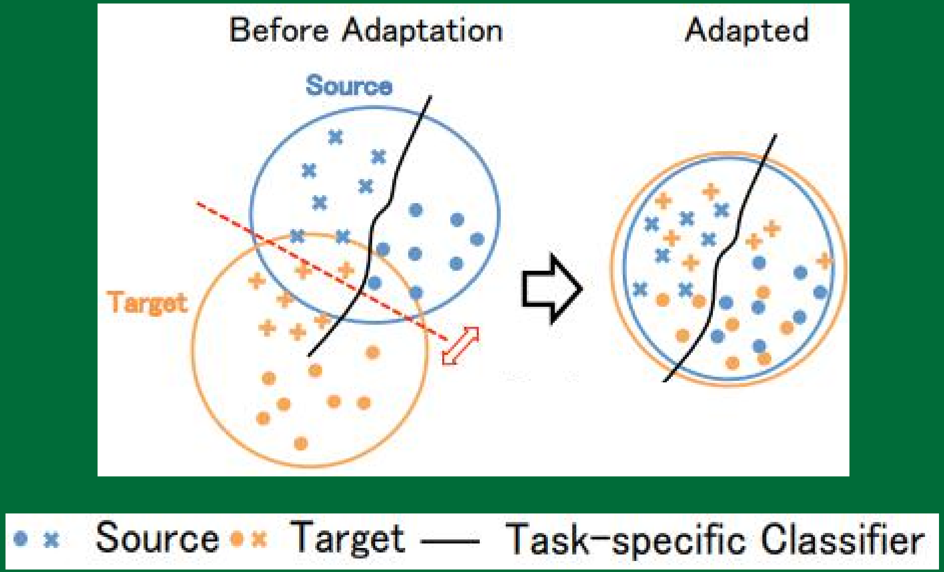
new datasets

may perform poorly

datasets bias or shift

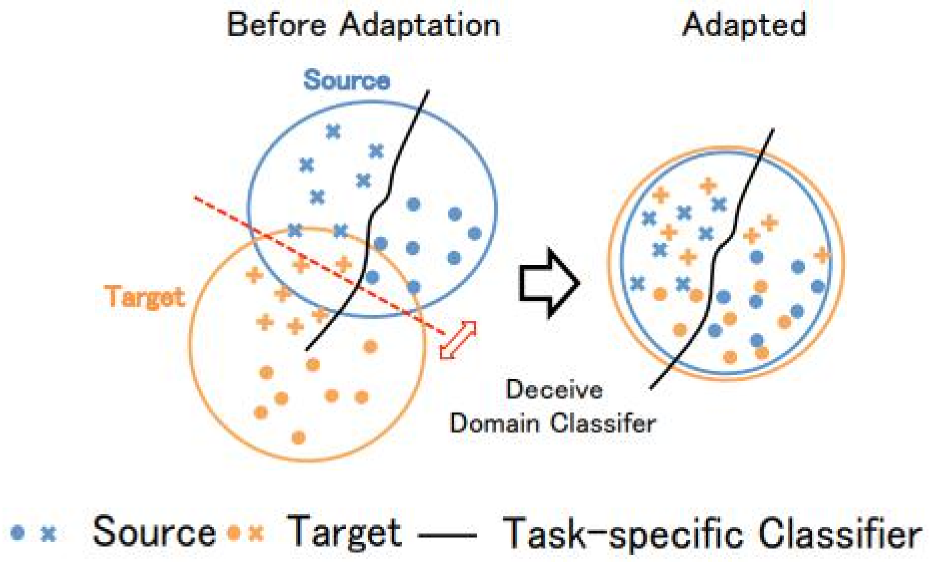


2 domain adaptation



transfer enriched knowledge
from a well-annotated domain
(source domain)
to a different label-scarce domain
(target domain)

domain adversarial methods



$$\mathcal{L}_d(\theta_g,\theta_d) = E_{x\sim D_s}[logD(G(x))] + E_{x\sim \hat{D}_t}[logD(1-G(x))].$$

$$\min_{\theta_g,\theta_f}\max_{\theta_d}\sum_{i=1}^{n_s}\mathcal{L}_c(F(G(x_i^s;\theta_g);\theta_f),y_i^s) + \lambda\mathcal{L}_d(\theta_g,\theta_d) + \gamma\mathcal{L}_{apa}(\theta_g)$$

- domain discriminator:min
- feature extractor:max

3 Progressive Feature Alignment

Progressive Feature Alignment

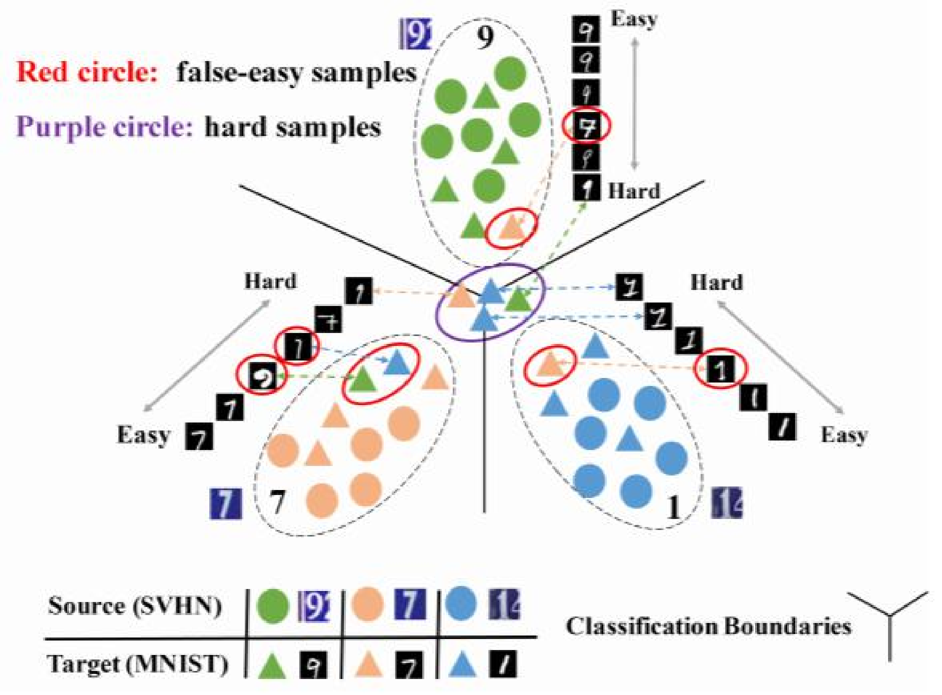
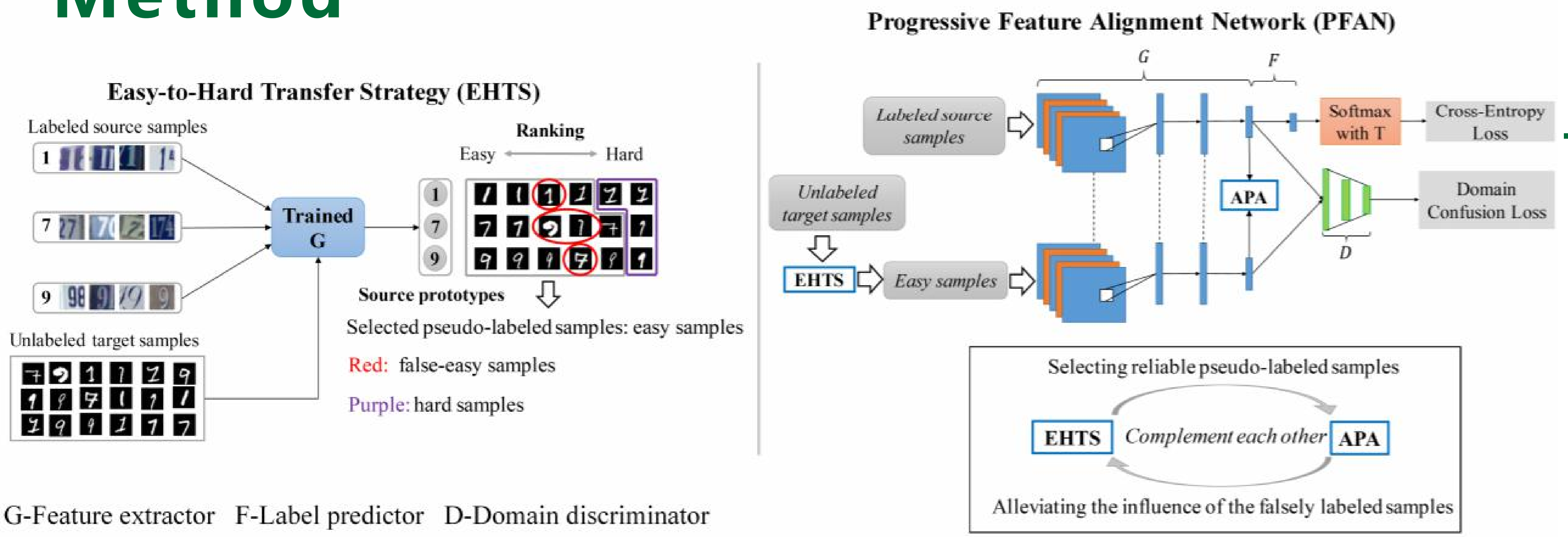


Figure 1: (Best viewed in color.) Motivations of the proposed work (SVHN→MNIST). The classification boundaries are first drawn by the fully labeled source domain. There is intra-class variation in the target domain.

Motivation

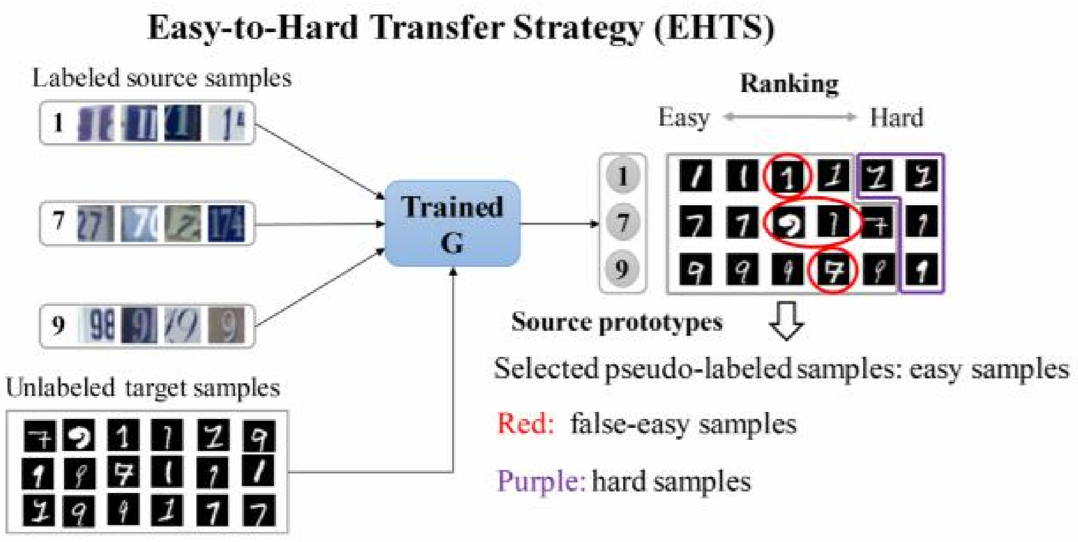
- 伪标签的错误积累：
  简单但被分错的样本
  难分类的样本

Method



G-Feature extractor F-Label predictor D-Domain discriminator

- EHTS:reliable pseudo labels from easy to hard by iterations
- APA:explicitly enforces the cross-domain category alignment
- Soft-max function with a temperature variate



G-Feature extractor F-Label predictor D-Domain discriminator

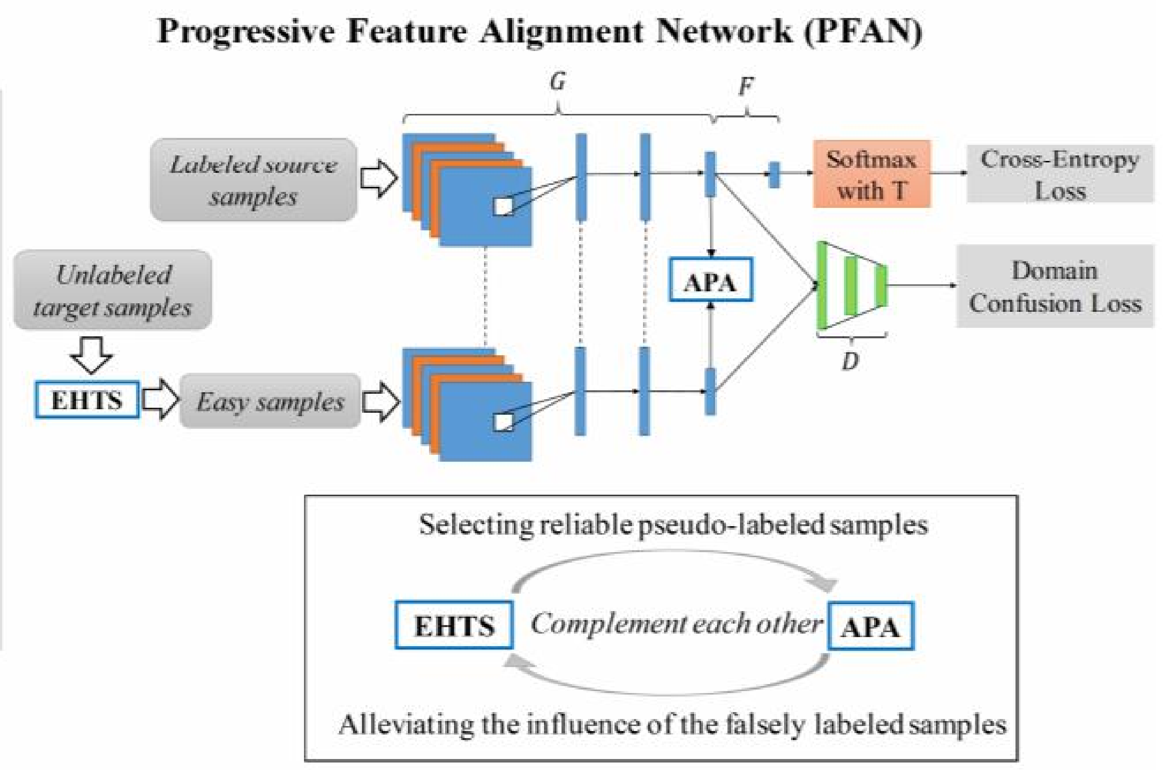
methods

- source prototype $c_k^S = \frac{1}{N_s^k} \sum_{(x_i^s, y_i^s) \in D_s^k} G(x_i^s)$
  - 相似度 $\psi(x_j^t) = CS(G(x_j^t), c_k^S), k = \{1, 2, ..., C\}$
  - 伪标签 $\arg\max_k \psi_k(x_j^t)$
- 所挑选样本的阈值 $\tau = \frac{1}{1 + e^{-\mu \cdot (m+1)}} - 0.01$

a selected pseudo-labeled target domain $\hat{D}_t = \{x_j^t, \hat{y}_j^t\}_{j=1}^{\hat{n}_t}$

Progressive Feature Alignment

北京理工大学 BEIJING INSTITUTE OF TECHNOLOGY



methods

$$\mathcal{L}_{apa}(\theta_g) = \sum_{k=1}^{C} d(c_{k(I)}^{S}, c_{k(I)}^{\mathcal{T}})$$

$$d(c_k^S, c_k^{\mathcal{T}}) = \left\| c_k^S - c_k^{\mathcal{T}} \right\|^2$$

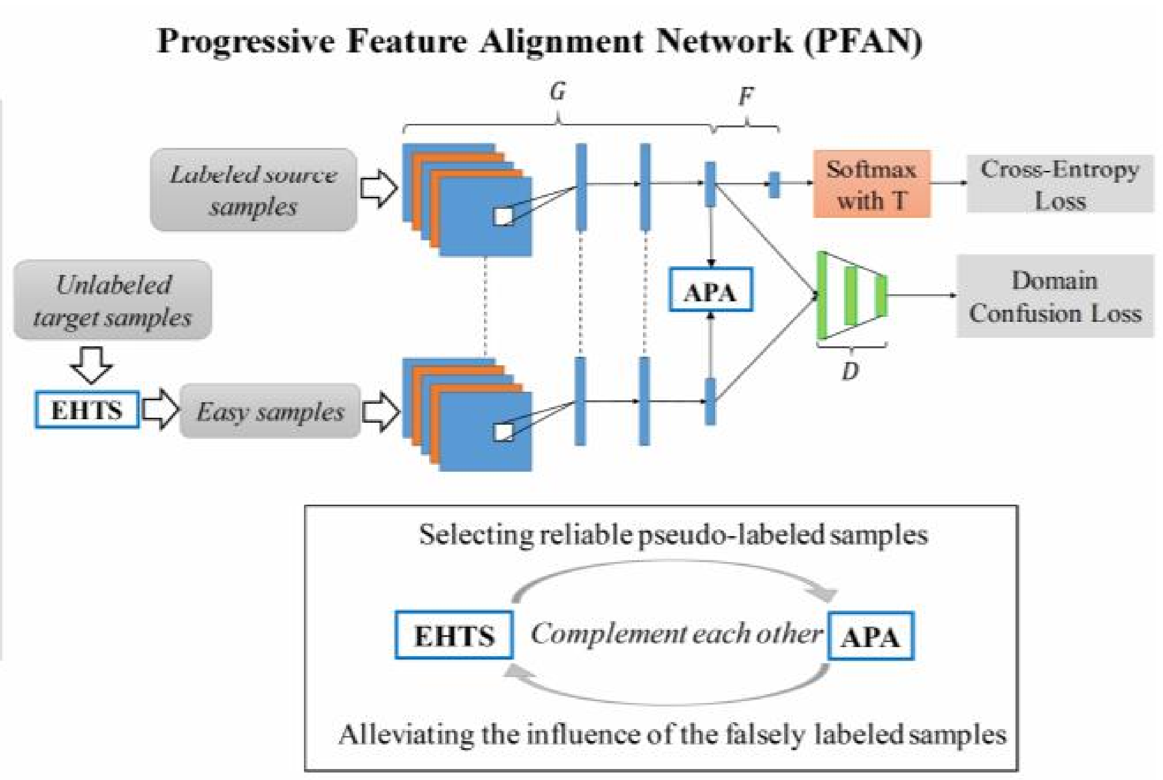
- target prototyes $c_{k(0)}^{\mathcal{T}} = \frac{1}{\hat{D}_t^k} \sum_{(x_j^t, y_j^t) \in \hat{D}_t^k} G(x_j^t)$

$$\bar{c}_{k(I)}^{t} = \frac{1}{I} \sum_{i=1}^{I} c_{k(i)}^{t}$$

$$\rho_t = CS(\bar{c}_{k(I)}^{t}, c_{k(I-1)}^{\mathcal{T}}),$$

$$c_{k(I)}^{\mathcal{T}} \leftarrow {\rho_t}^2 \bar{c}_{k(I)}^{t} + (1 - {\rho_t}^2) c_{k(I-1)}^{\mathcal{T}}$$

- source prototypes 类似



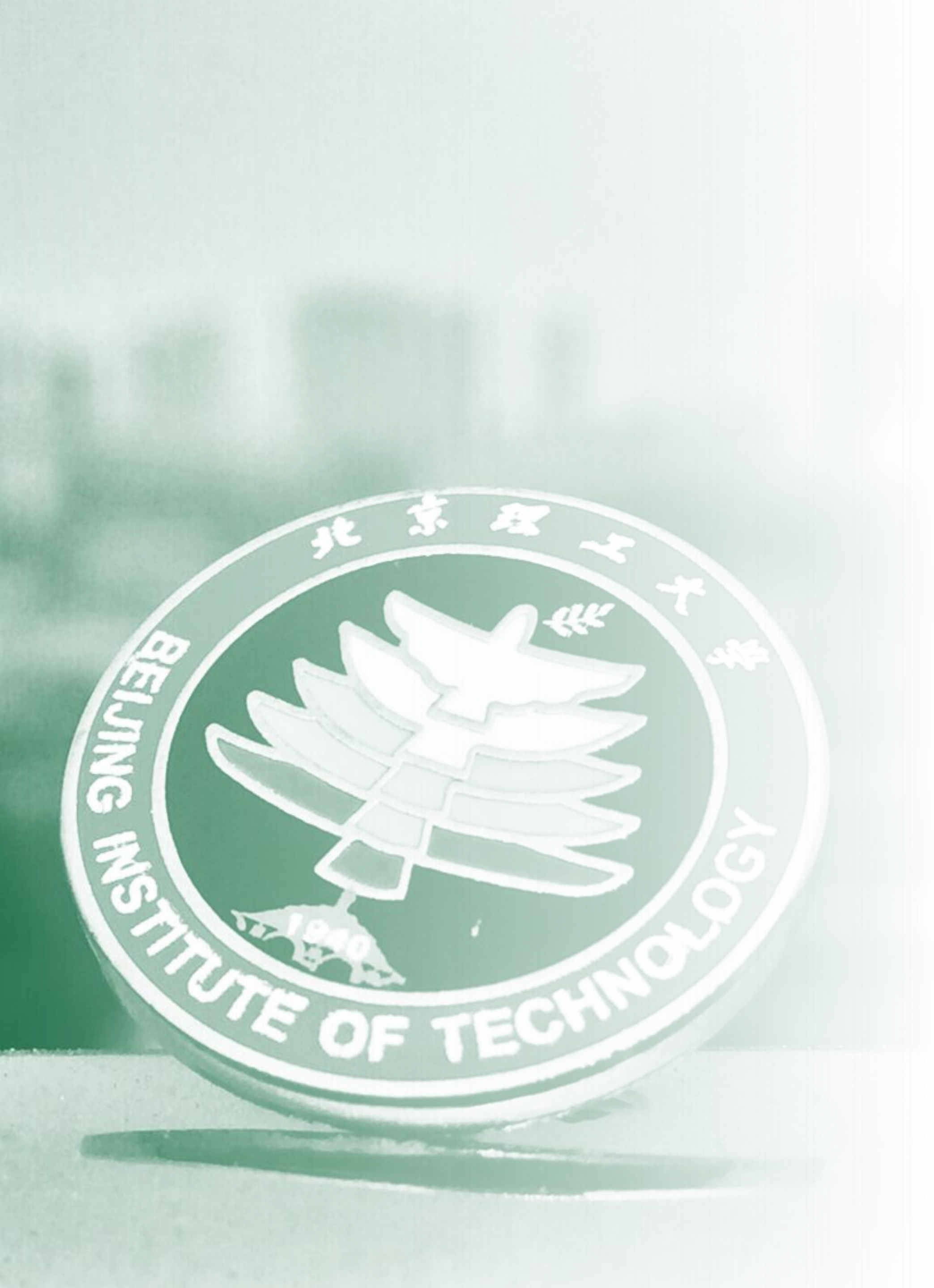
methods

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}$$

- T>1 防止对source sample过拟合
- 减缓收敛速度
- 域适应效果更好

methods

$$\min_{\theta_g,\theta_f} \max_{\theta_d} \sum_{i=1}^{n_s} \mathcal{L}_c(F(G(x_i^s;\theta_g);\theta_f), y_i^s)$$
$$+\lambda\mathcal{L}_d(\theta_g,\theta_d) + \gamma\mathcal{L}_{apa}(\theta_g)$$



4 Q&A

Thanks for Your Attention