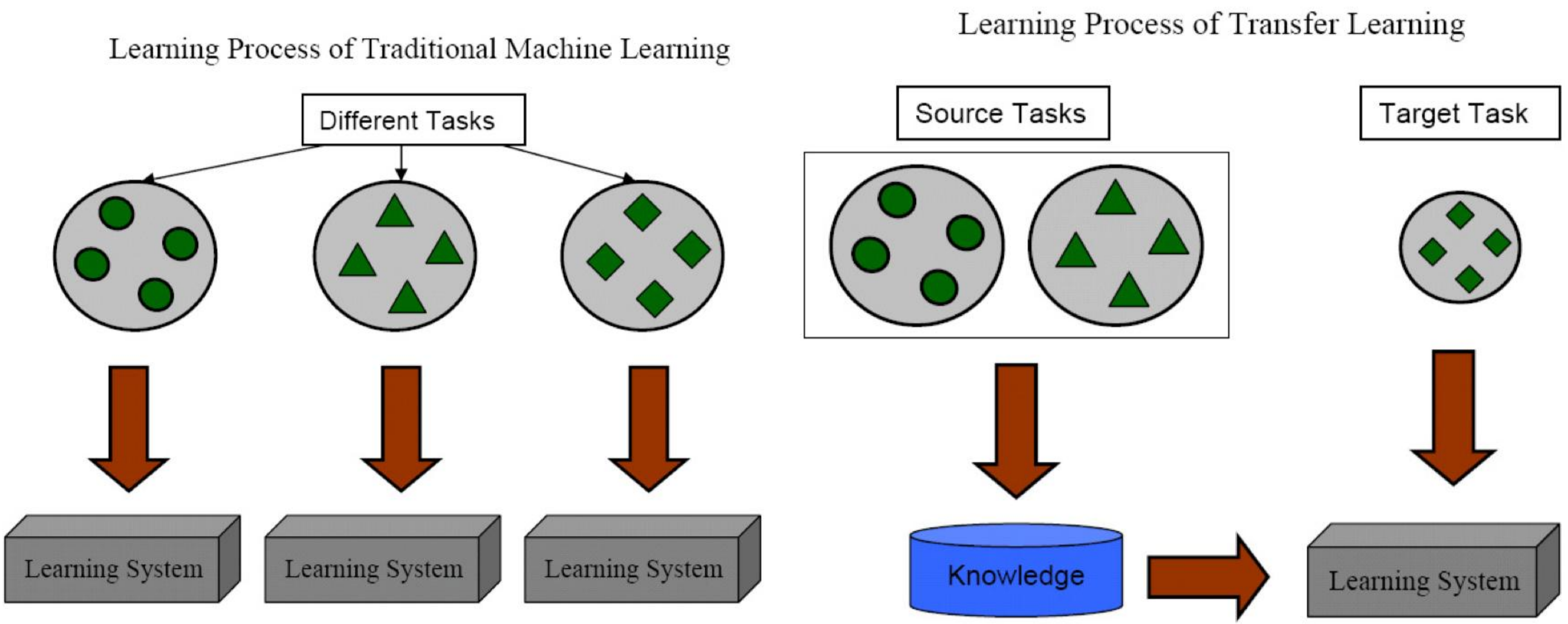
# Active Learning with Transfer Learning in Deep Neural Networks

by Jiawei

# Transfer Learning



# Transfer Learning

Instance Transfer

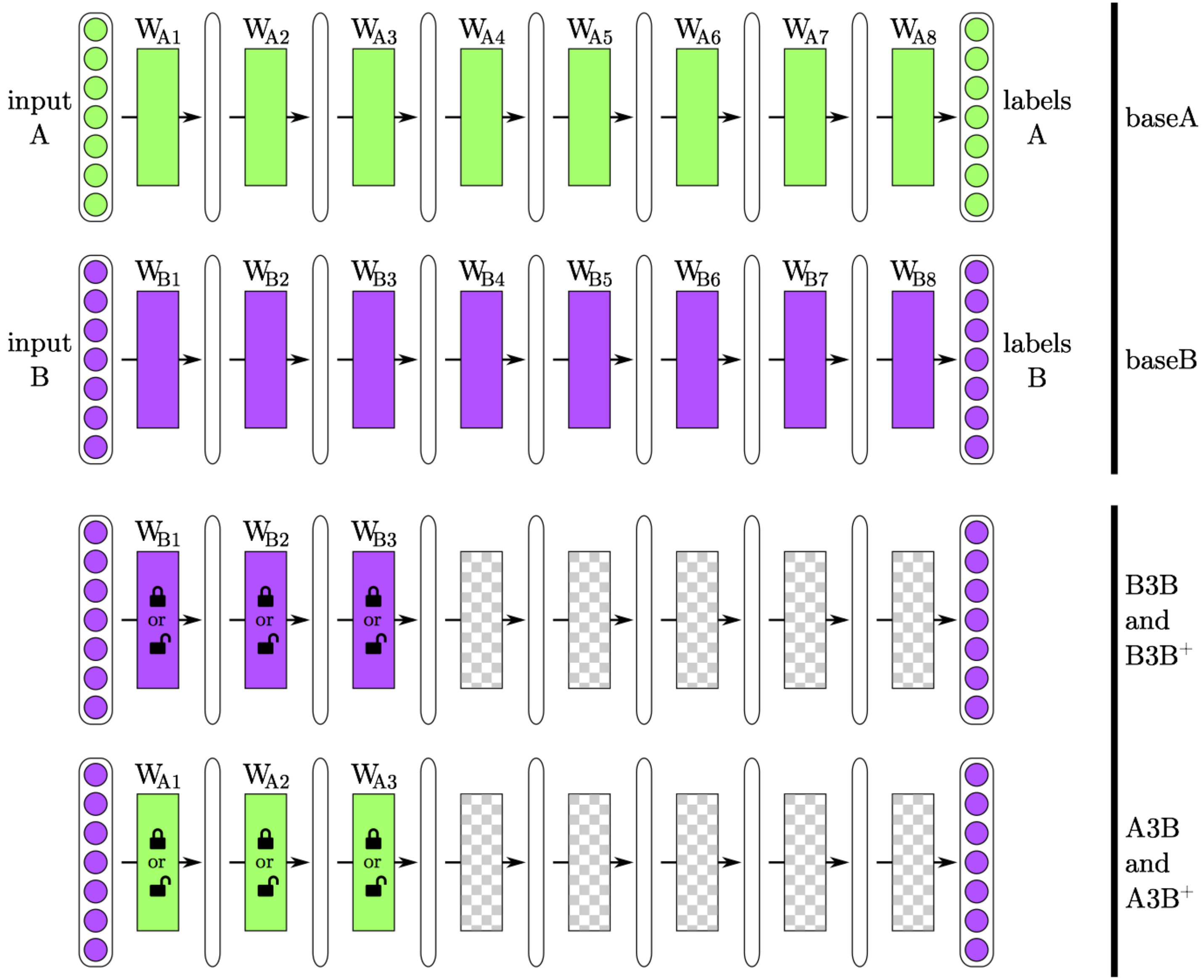
Feature representation Transfer

Parameter Transfer

# Transfer Learning in Deep Neural Networks

- Many deep neural networks trained on natural images exhibit a curious phenomenon in common: on the first layer they learn features similar to Gabor filters and color blobs. Such first-layer features appear not to be specific to a particular dataset or task, but general in that they are applicable to many datasets and tasks.

- general to specific
- fixed transfer layers
- domain discrepancy



input A
$W_{A1}$ $W_{A2}$ $W_{A3}$ $W_{A4}$ $W_{A5}$ $W_{A6}$ $W_{A7}$ $W_{A8}$
labels A
baseA
input B
$W_{B1}$ $W_{B2}$ $W_{B3}$ $W_{B4}$ $W_{B5}$ $W_{B6}$ $W_{B7}$ $W_{B8}$
labels B
baseB
$W_{B1}$ $W_{B2}$ $W_{B3}$
or
B3B and B3B$^+$
$W_{A1}$ $W_{A2}$ $W_{A3}$
or
A3B and A3B$^+$

# Deep Adaptation Networks

- The main idea of this work is to enhance the feature transferability in the task-specific layers of the deep neural network by explicitly reducing the domain discrepancy

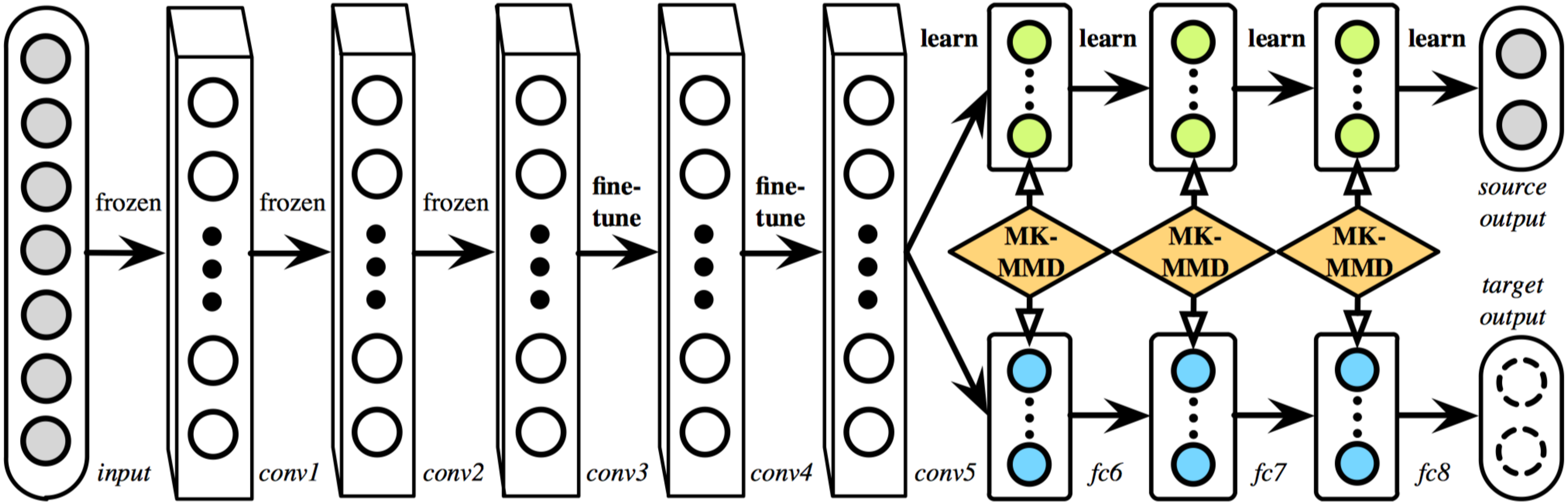
- to find the domain-invariant representation

# MK-MMD

- maximum mean discrepancies

- 

$$d_k^2(p,q) \triangleq \left\| \mathbf{E}_p\left[\phi\left(\mathbf{x}^s\right)\right] - \mathbf{E}_q\left[\phi\left(\mathbf{x}^t\right)\right] \right\|_{\mathcal{H}_k}^2 .$$



frozen
frozen
frozen
fine-tune
fine-tune
learn
learn
learn
learn
MK-MMD
MK-MMD
MK-MMD
source output
target output
input
conv1
conv2
conv3
conv4
conv5
fc6
fc7
fc8

change from

$$\min_{\Theta} \frac{1}{n_a} \sum_{i=1}^{n_a} J\left(\theta\left(\mathbf{x}_i^a\right), y_i^a\right),$$

to

$$\min_{\Theta} \frac{1}{n_a} \sum_{i=1}^{n_a} J\left(\theta\left(\mathbf{x}_i^a\right), y_i^a\right) + \lambda \sum_{\ell=l_1}^{l_2} d_k^2\left(\mathcal{D}_s^\ell, \mathcal{D}_t^\ell\right),$$

# Problem definition

- Given a pre-trained deep neural networks and its source training data (or not) , and target unlabeled data, transfer the deep neural networks from source data to target data by actively select and label the useful samples in target unlabeled data.

# Related Work

- Fine-tuning Convolutional Neural Networks for Biomedical Image Analysis Actively and Incrementally

- All patches generated from the same candidate share the same label; they are expected to have similar predictions by the current C- NN. As a result, their entropy and diversity provide a use- ful indicator to the “power” of a candidate in elevating the performance of the current CNN.

$$e_i^j = -\sum_{k=1}^{|Y|} p_i^{j,k} \log p_i^{j,k}$$

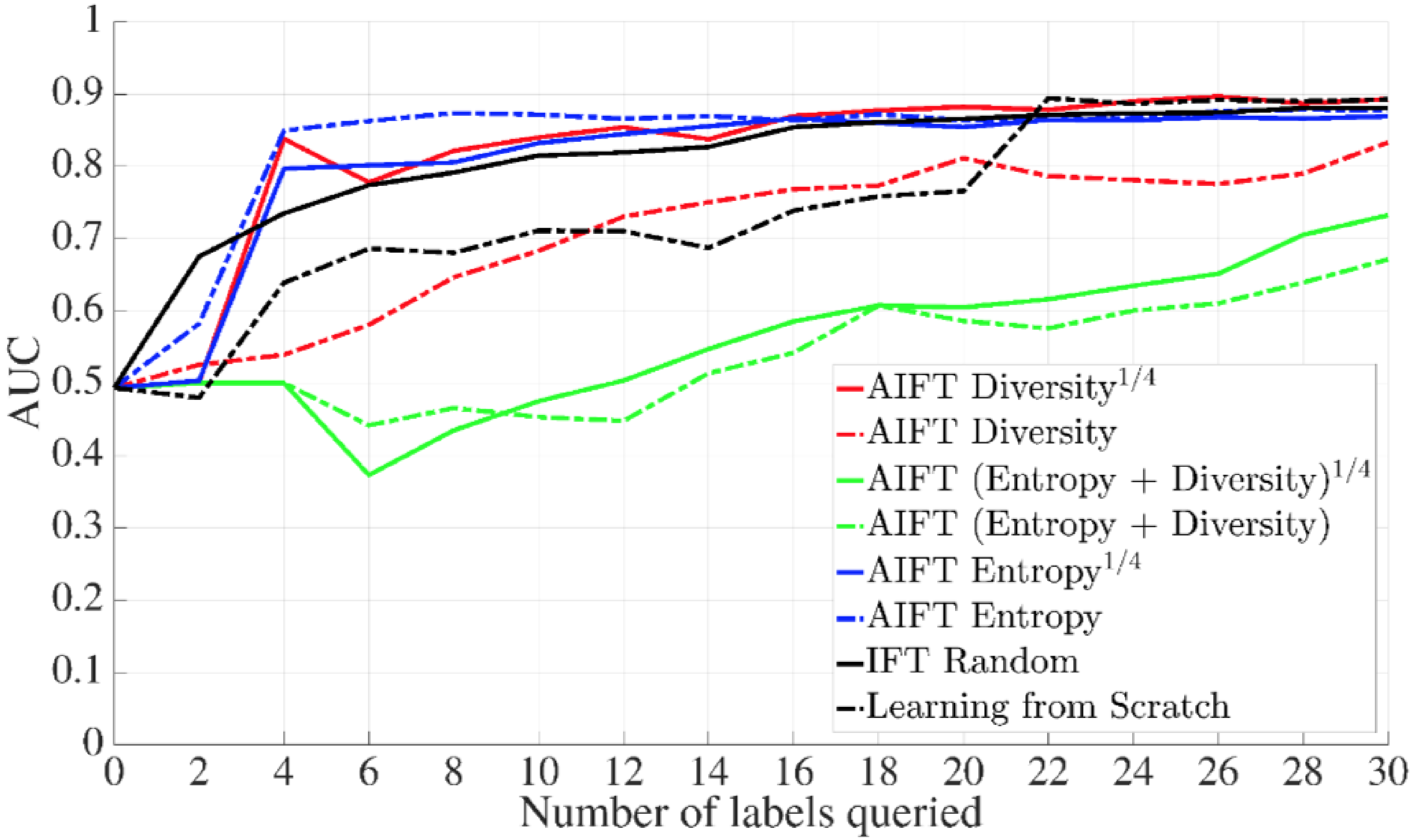
$$d_i(j,l) = \sum_{k=1}^{|Y|} (p_i^{j,k} - p_i^{l,k}) log \frac{p_i^{j,k}}{p_i^{l,k}}$$

$$R_i(j,l) = \begin{cases} \lambda_1 e_i^j & \text{if } j = l, \\ \lambda_2 d_i(j,l) & \text{otherwise} \end{cases}$$

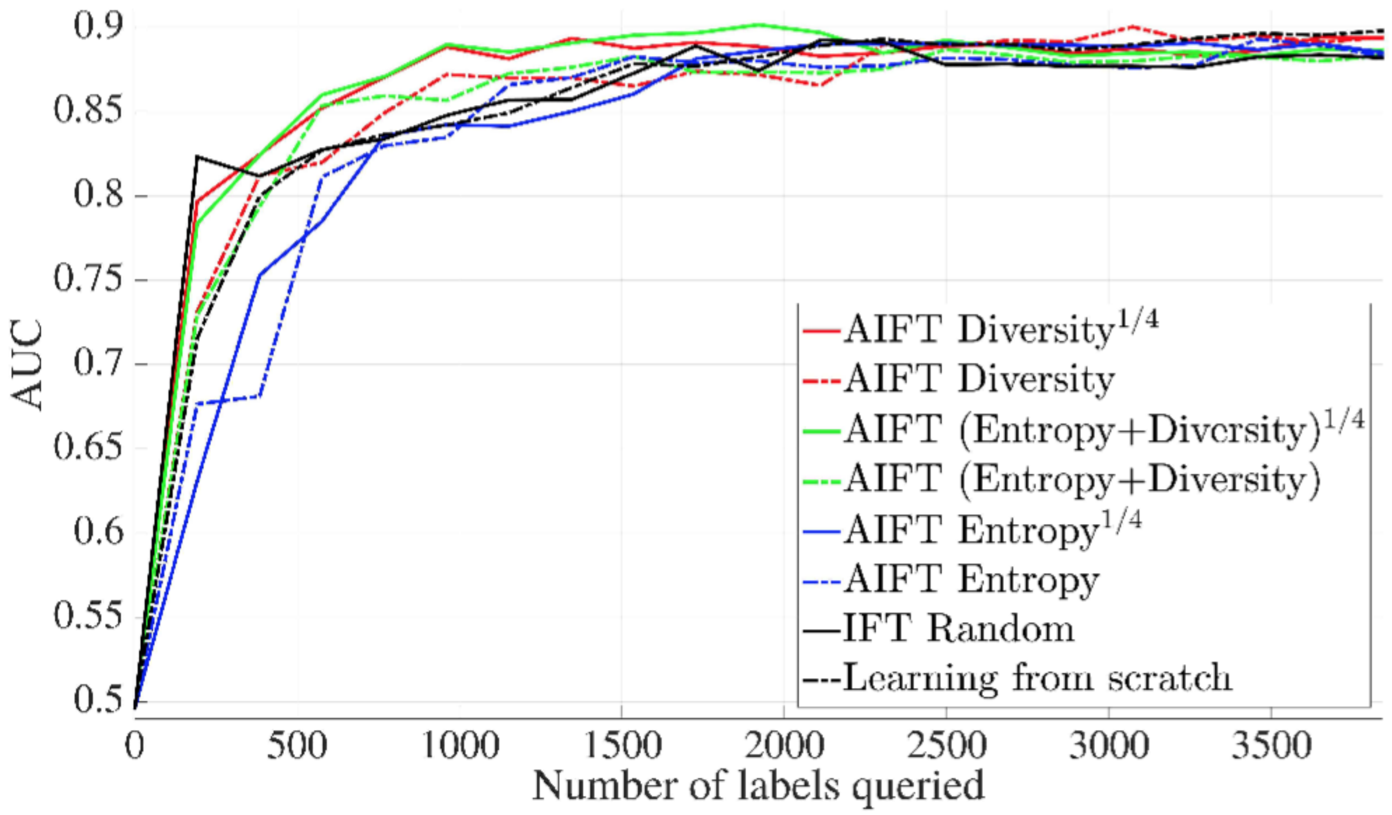
**Algorithm 1:** Active incremental fine-tuning method.

**Input**:
$\mathcal{U} = \{\mathcal{C}_i\}, i \in [1, n]$ $\{\mathcal{U}$ contains $n$ candidates$\}$
$\mathcal{C}_i = \{x_i^j\}, j \in [1, m]$ $\{\mathcal{C}_i$ has $m$ patches$\}$
$\mathcal{M}_0$: pre-trained CNN
$b$: batch size
$\alpha$: patch selection ratio
**Output**:
$\mathcal{L}$: labeled candidates
$\mathcal{M}_t$: fine-tuned CNN model at Iteration $t$
**Functions**:
$p \leftarrow P(\mathcal{C}, \mathcal{M})$ $\{$outputs of $\mathcal{M}$ given $\forall x \in \mathcal{C}\}$
$\mathcal{M}_t \leftarrow F(\mathcal{L}, \mathcal{M}_{t-1})$ $\{$fine-tune $\mathcal{M}_{t-1}$ with $\mathcal{L}\}$
$a \leftarrow mean(p_i)$ $\{a = \frac{1}{m}\sum_{j=1}^{m} p_i^j\}$
**Initialize**:
$\mathcal{L} \leftarrow \varnothing$, t$\leftarrow 1$

**repeat**
  **for** *each* $\mathcal{C}_i \in \mathcal{U}$ **do**
    $p_i \leftarrow P(\mathcal{C}_i, \mathcal{M}_{t-1})$
    **if** $mean(p_i) > 0.5$ **then**
      $\mathcal{S}_i' \leftarrow$ top $\alpha$ percent of the patches of $\mathcal{C}_i$
    **else**
      $\mathcal{S}_i' \leftarrow$ bottom $\alpha$ percent of the patches of $\mathcal{C}_i$
    **end**
    Build matrix $R_i$ using Eq. 3 for $\mathcal{S}_i'$
  **end**
  Sort $\mathcal{U}$ according to the numerical sum of $R_i$
  Query labels for top $b$ candidates, yielding $\mathcal{Q}$
  $\mathcal{L} \leftarrow \mathcal{L} \bigcup \mathcal{Q}$; $\mathcal{U} \leftarrow \mathcal{U} \setminus \mathcal{Q}$
  $\mathcal{M}_t \leftarrow F(\mathcal{L}, \mathcal{M}_{t-1})$; $t \leftarrow t + 1$
**until** *classification performance is satisfactory*;



AUC
Number of labels queried
AIFT Diversity$^{1/4}$
AIFT Diversity
AIFT (Entropy + Diversity)$^{1/4}$
AIFT (Entropy + Diversity)
AIFT Entropy$^{1/4}$
AIFT Entropy
IFT Random
Learning from Scratch



0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
AUC
0
500
1000
1500
2000
2500
3000
3500
Number of labels queried
AIFT Diversity$^{1/4}$
AIFT Diversity
AIFT (Entropy+Diversity)$^{1/4}$
AIFT (Entropy+Diversity)
AIFT Entropy$^{1/4}$
AIFT Entropy
IFT Random
Learning from scratch

# Thoughts

- Training
- 采用DAN模型对源数据集进行训练，训练出对于源数据和目标数据空间都比较general的特征。以此可以有效的描述目标数据的特征空间。

# Thoughts

- Transfer
- 策略分为两个方面 uncertainty 和 diversity

- diversity  利用最后一层的特征空间使用core-set 的方式进行衡量

- uncertainty 去寻找一组样本使之与源数据空间的分布差异性最大
- 方法： 使用MMD 描述目标数据空间与源数据空间的差异，在目标数据集中寻找一组样本，去掉这组样本后，源与目标空间的差异性会变小。标记将差异性减小最大的一组。