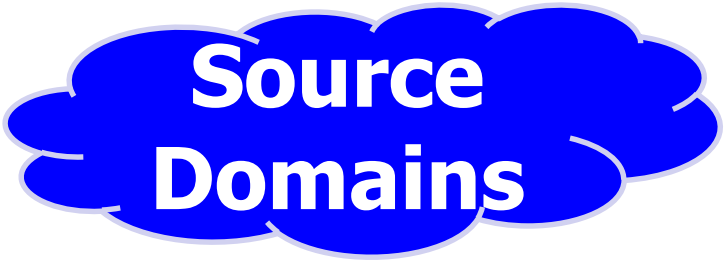
New Advances in Transfer Learning

Qiang Yang

Hong Kong University of Science and Technology
http://www.cse.ust.hk/~qyang

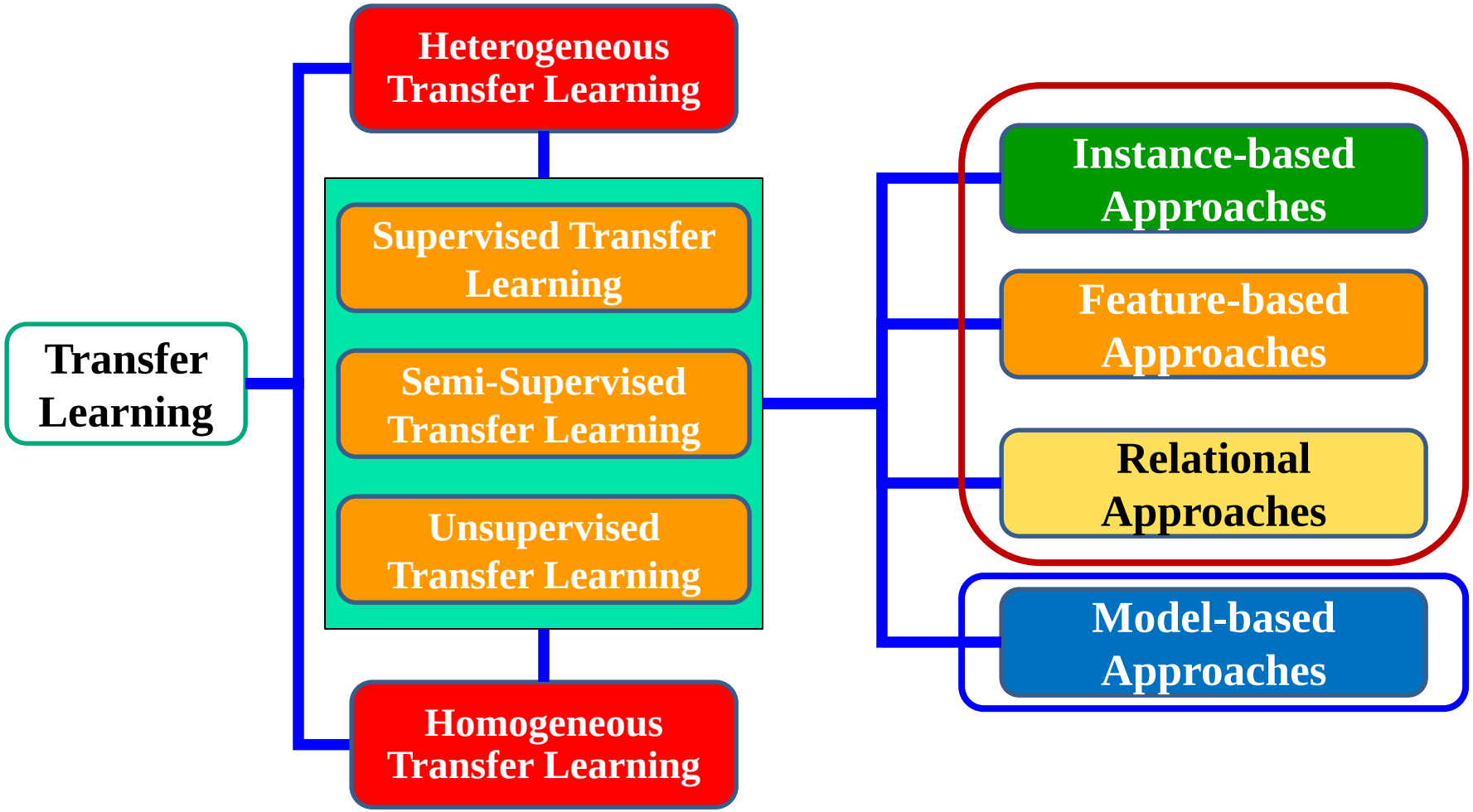
Learning: A Major Assumption

- Training and future (test) data:
  - follow the same distribution, and are in same feature space

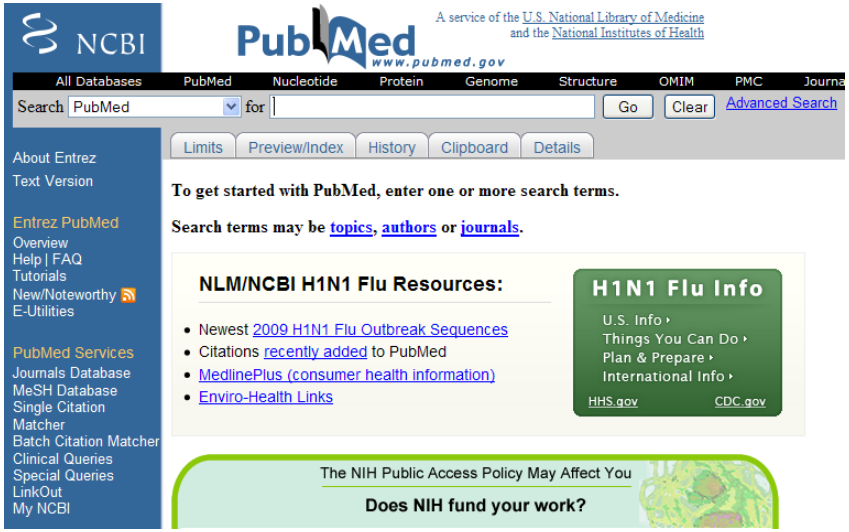


	Source Domain	Target Domain
Training Data	Labeled/Unlabeled	Labeled/Unlabeled
Test Data		Unlabeled

Overview


Transfer Learning
Heterogeneous Transfer Learning
Supervised Transfer Learning
Semi-Supervised Transfer Learning
Unsupervised Transfer Learning
Homogeneous Transfer Learning
Instance-based Approaches
Feature-based Approaches
Relational Approaches
Model-based Approaches

When distributions are different



- Part-of-Speech tagging
- Named-Entity Recognition
- Classification

Structural Correspondence Learning [Blitzer et al. ACL 2007]

- SCL: [Ando and Zhang, JMLR 2005]
- Method
  - Define pivot features: common in two domains (not buy)
  - Find non-pivot features in each domain (repetitive)
  - Build classifiers through the non-pivot Features

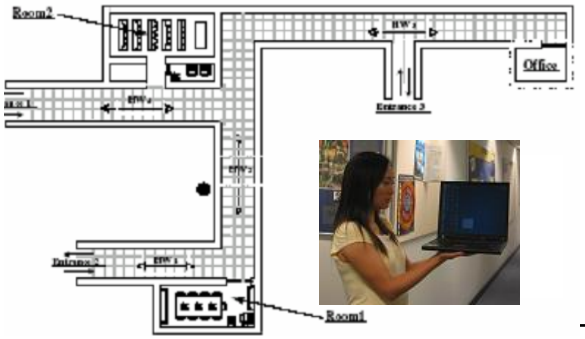
(1) The book is so repetitive that I found myself yelling …. I will definitely not buy another.

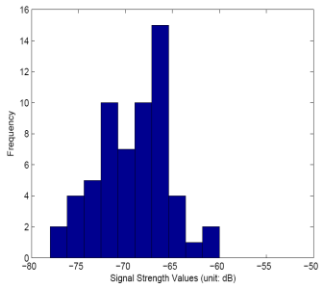
(2) Do not buy the Shark portable steamer …. Trigger mechanism is defective.

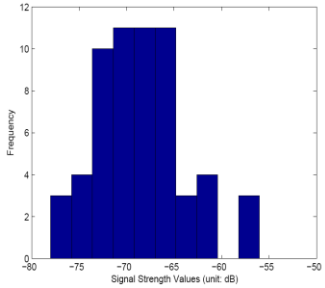
Book Domain ⟷ **Kitchen Domain**

Distribution Changes

- The mapping function *f* learned in the offline phase can be out of date.
- **Recollecting the WiFi data is very expensive.**
- **How to adapt the model ?**







Time

Night time period t_0

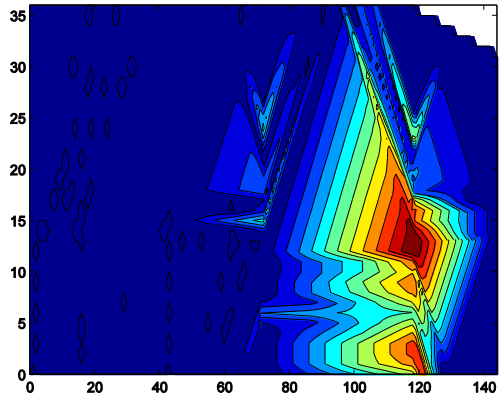
Day time period t_1

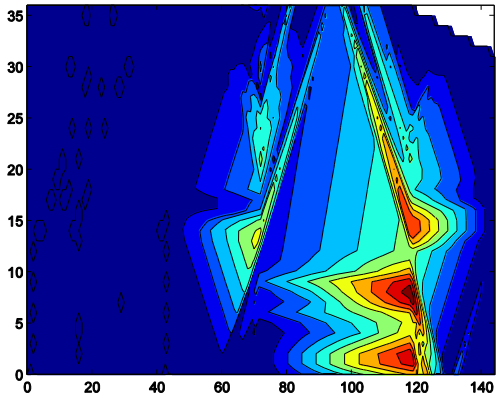
Differences between Domains

Time Period A

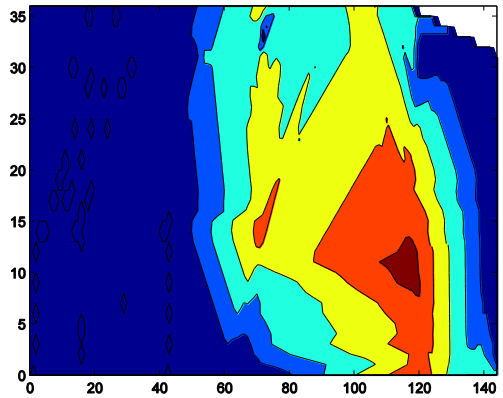
Time Period B

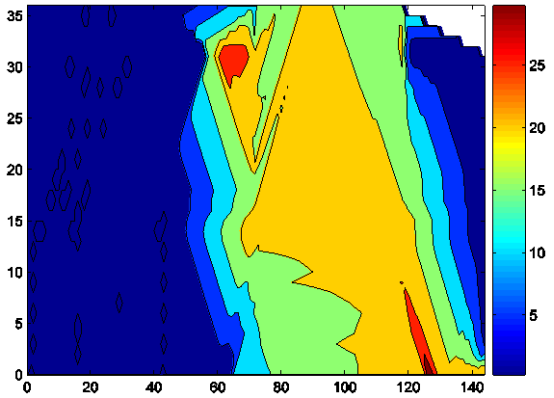
Device A





Device B





HTL Setting: Text to Images

- Source: labeled or unlabeled
- Target: few labeled

Training: Text

Testing: Images

Apple

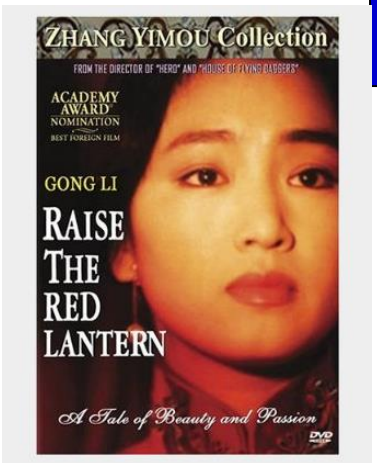
The apple is the pomaceous fruit of the apple tree, species Malus domestica in the rose family Rosaceae ...

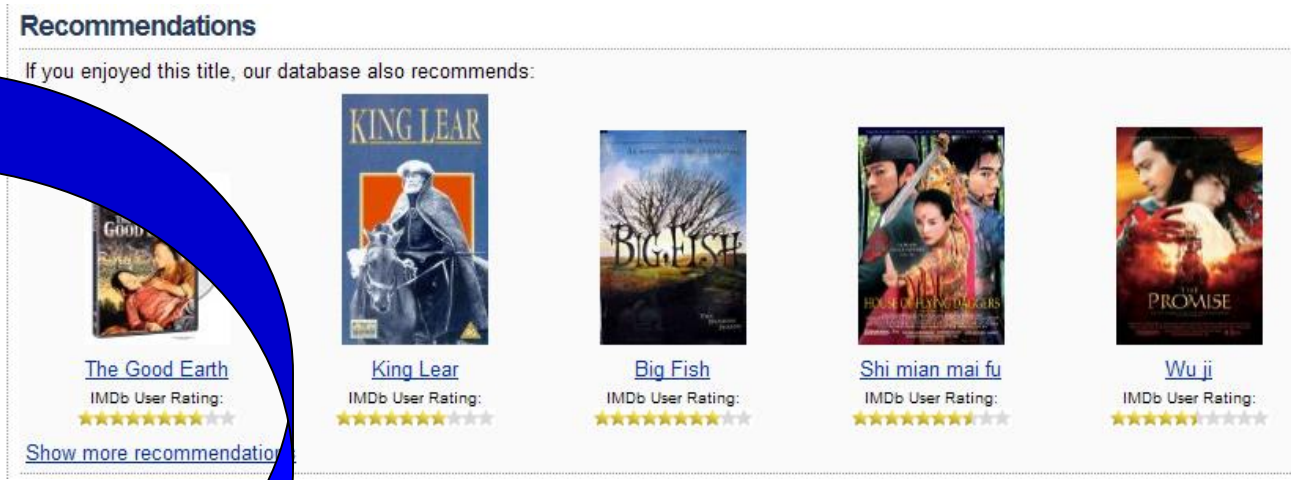
Banana

Banana is the common name for a type of fruit and also the herbaceous plants of the genus Musa which produce this commonly eaten fruit ...

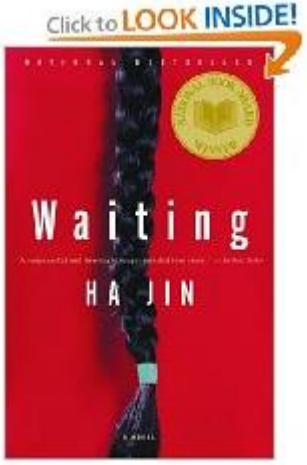
Transfer Learning for Collaborative Filtering

IMDB Database





Amazon.com



Activity Recognition

- Healthcare at home and in hospitals

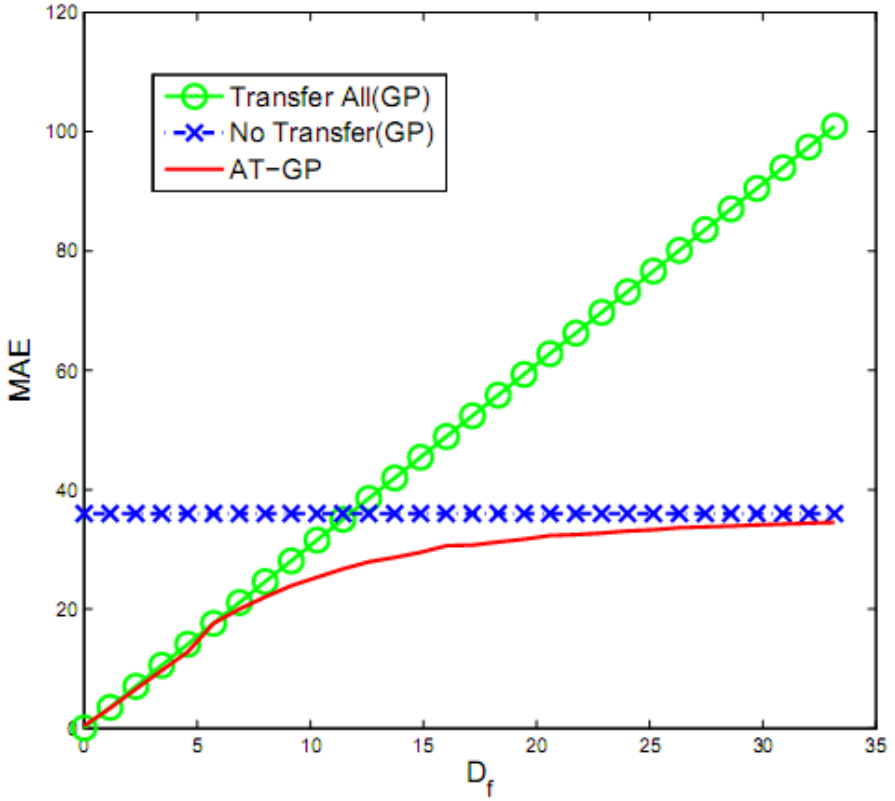
- Logistics, Shopping

Cross Domain Activity Recognition

[Zheng, Hu, Yang, ACM Ubicomp 2009]

- Challenges:
  - A new domain of activities without labeled data
- Cross-domain activity recognition
  - Transfer some available labeled data from source activities to help training the recognizer for the target activities.

Adaptive: transfer-all or none



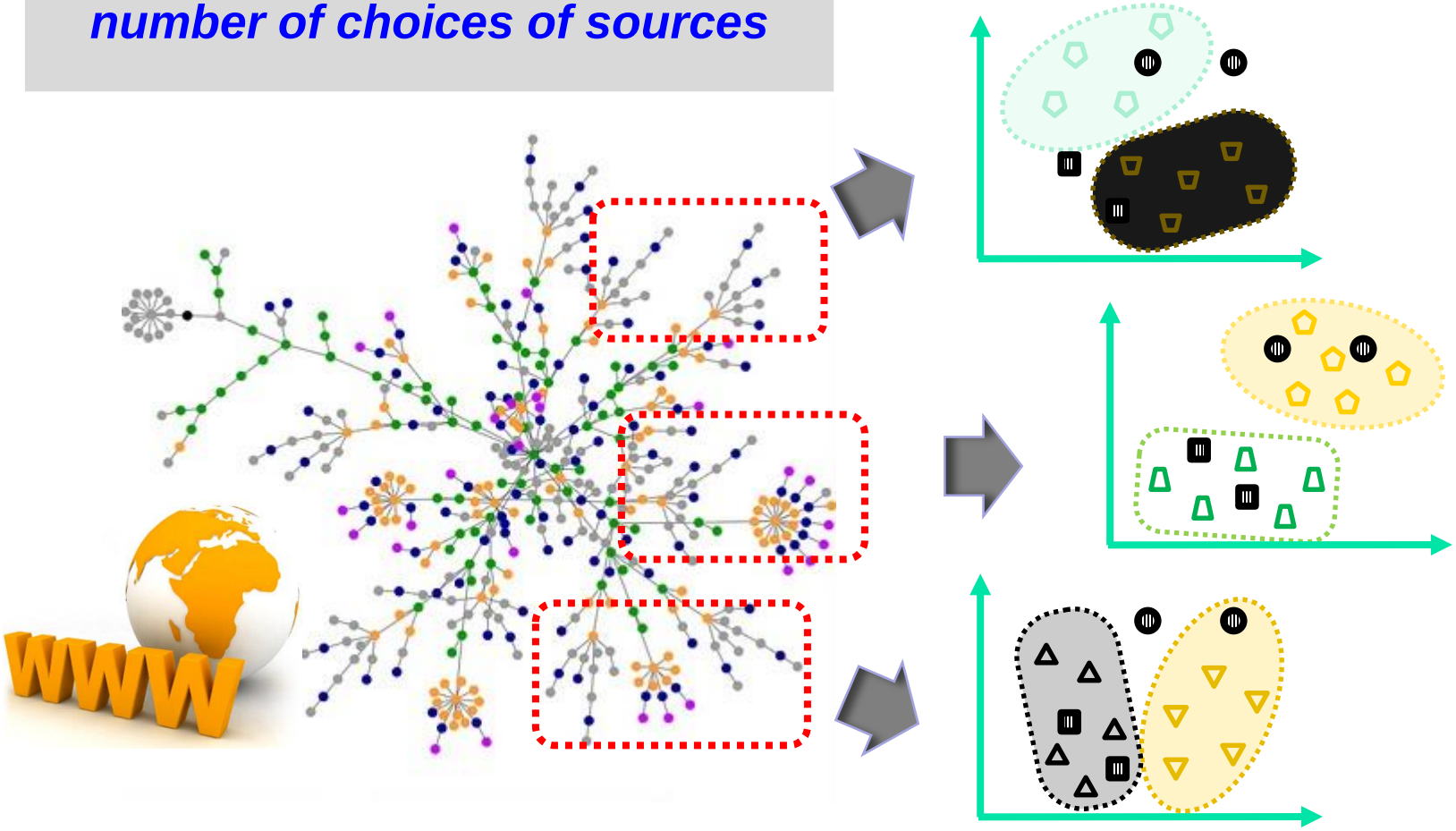
Distance between the source and target tasks

- As good as Transfer All when the source and target tasks are very similar.
- Not worse than No Transfer when the source and target tasks are not related at all.

Bin Cao, Sinno Jialin Pan, Yu Zhang, Dit-Yan Yeung and Qiang Yang. Adaptive Transfer Learning. In Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI-10). Atlanta, Georgia, USA. July 11-15, 2010.

Source-Free Transfer Learning (IJCAI11)

We may have an extremely large number of choices of sources



Source Free Transfer Learning – Evan Xiang, Sinno Pan, Q. Yang et al. IJCAI 2011

Lifelong Machine Learning

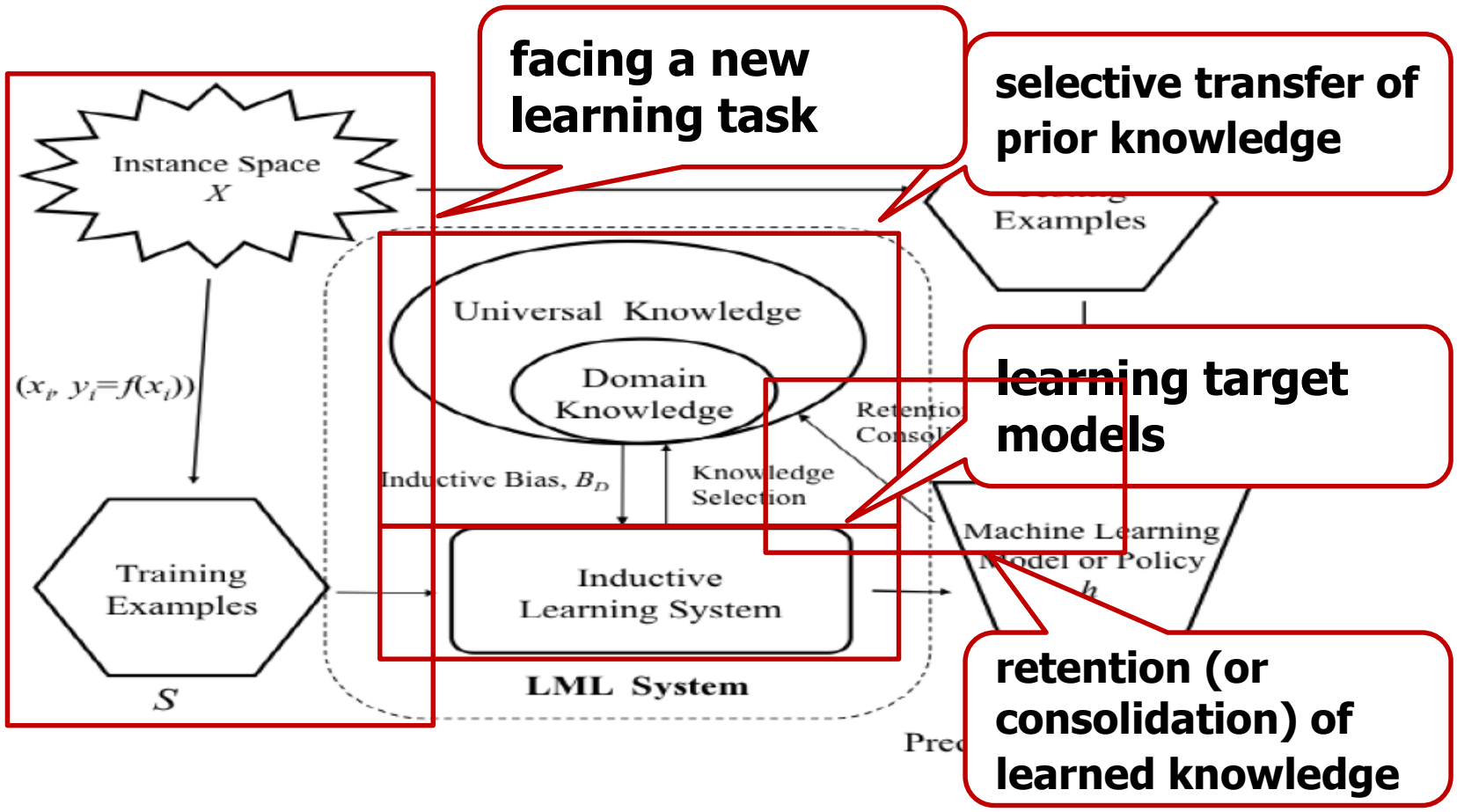


Figure 1: A framework for lifelong machine learning.

LML Examples

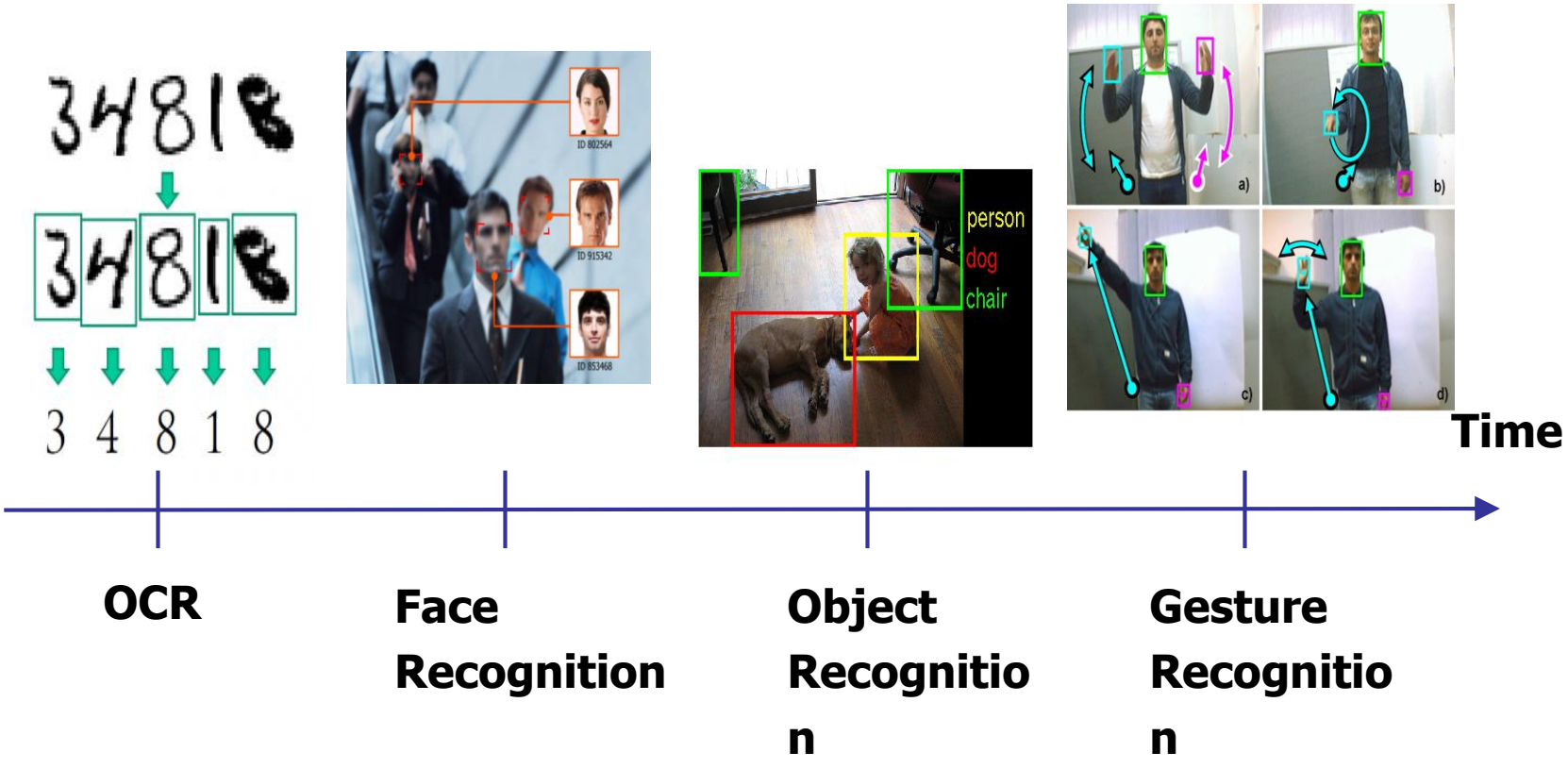
Never-Ending Language Learner [Tom Mitchell et al. 2010]

- Goal:
  - attempt to create a computer system that learns over time to read the Web (24x7, forever)
  - each day:
    - extract more facts from the web to populate the initial ontology, e.g.,
      - Brazil is a country
      - Poza_Rica is a city located in the country Mexico
    - learn to read (perform #1) better than yesterday

Lifelong Learning Test

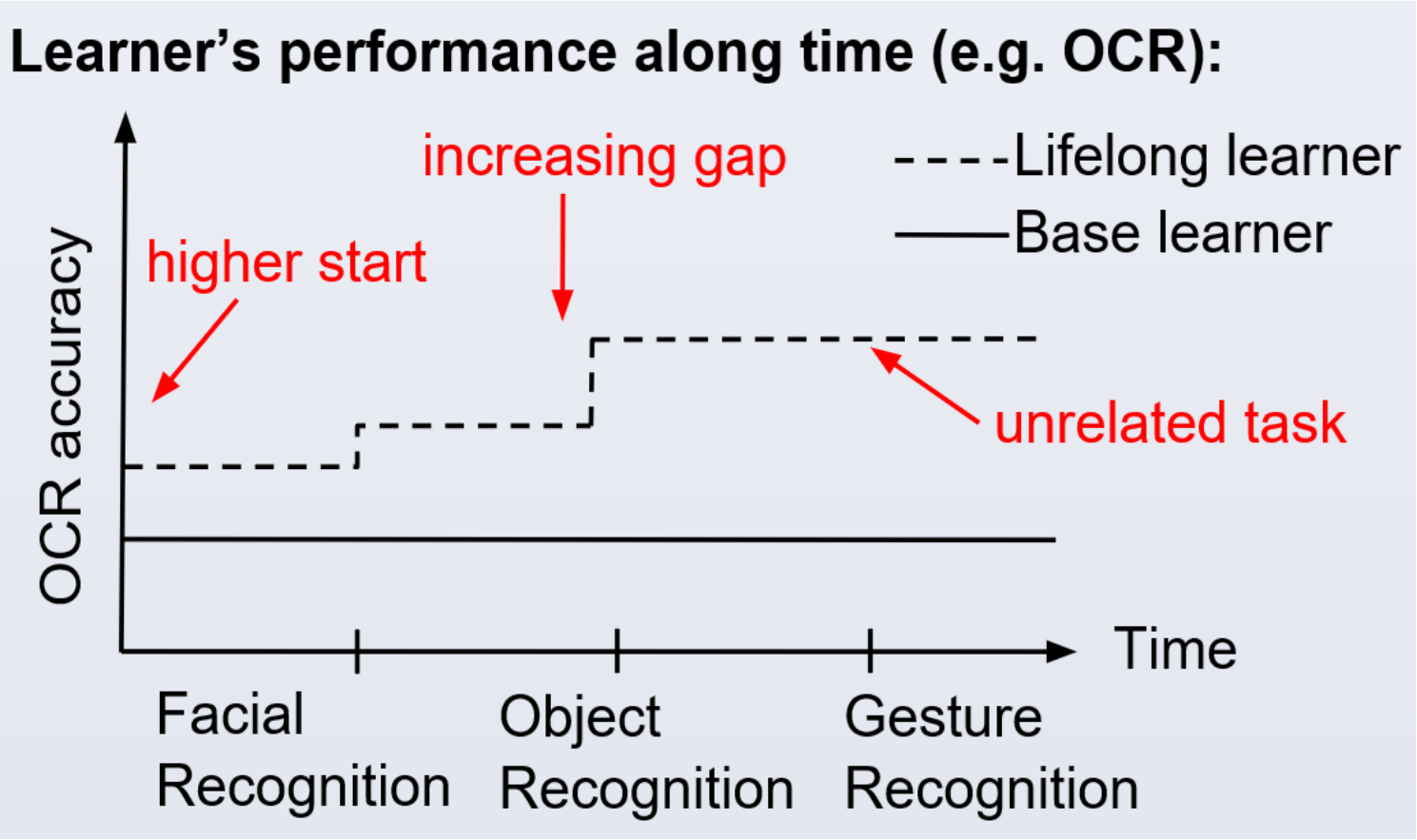
Two steps:

Step 1: learn **unrestricted** number of tasks over time, and



Lifelong Learning Test (Cont.)

Step 2: perform better and better than a base learner.



Theoretical Guarantee

Theoretical Requirement

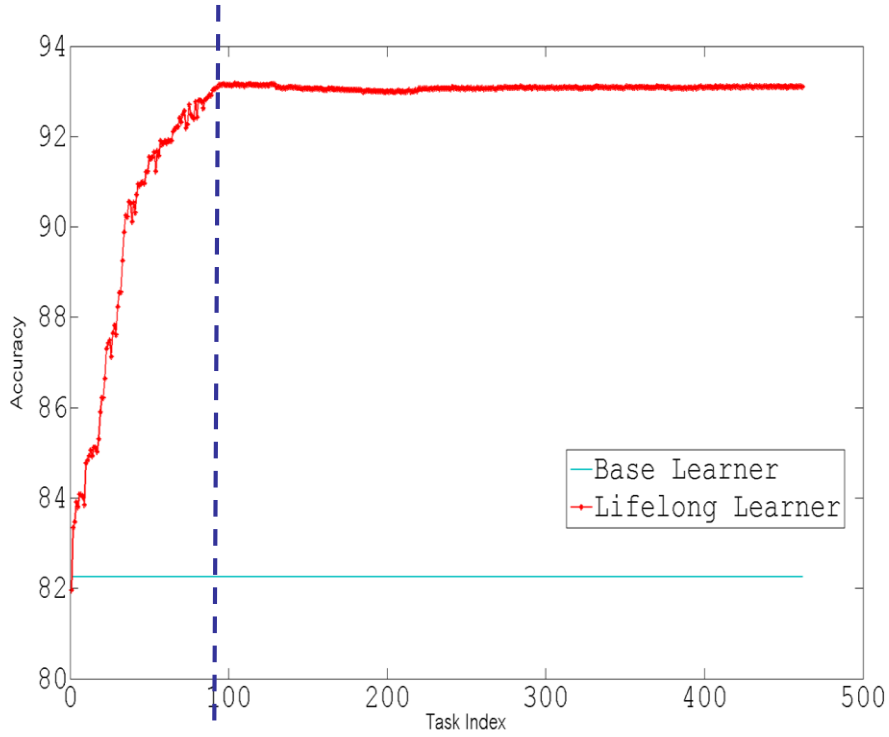
Let $\mathcal{H}$ be hypothesis space, *K* be the total number of tasks seen so far, $m^{(\ell)}$ is the number of training data in task *l*. Let δ be fixed. Then with probability at least $1-\delta$, a lifelong learner should hold the following generalization error (i.e. $\varepsilon(\hat{h}^{(l)})$) bound for all tasks

$$\varepsilon(\hat{h^{(\ell)}}) \leq \min_{h^{(\ell)} \in \mathcal{H}} \varepsilon(h^{(\ell)}) + 2\sqrt{\frac{1}{2m^{(\ell)}} \log \frac{2k}{\delta}} - \boxed{\alpha\left(K, m^{(1)}, m^{(2)}, \cdots, m^{(K)}\right)}$$

Lifelong learning credit: a monotonically increasing function w.r.t *K* and *m*.

$$\alpha(\cdot) \in \left(0, 2\sqrt{\frac{1}{2m^{(\ell)}} \log \frac{2k}{\delta}}\right]$$

LML on 500 Topic Classification Tasks in a Row



Topic classification

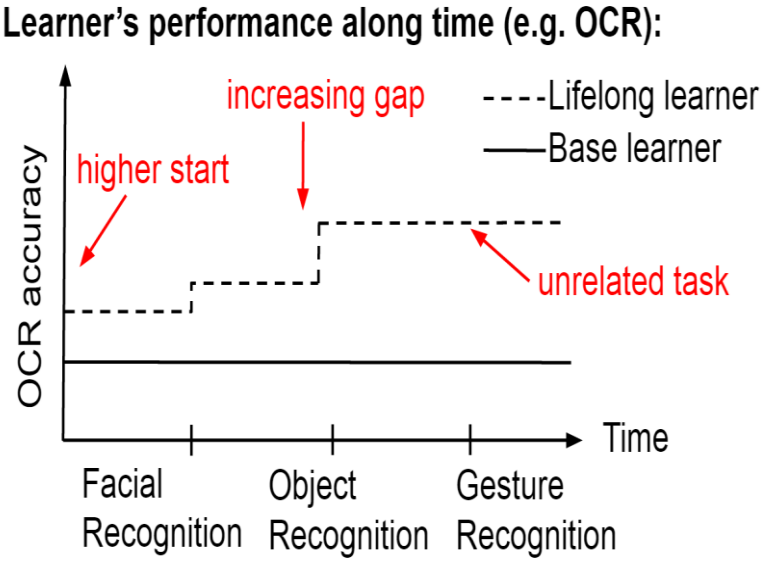


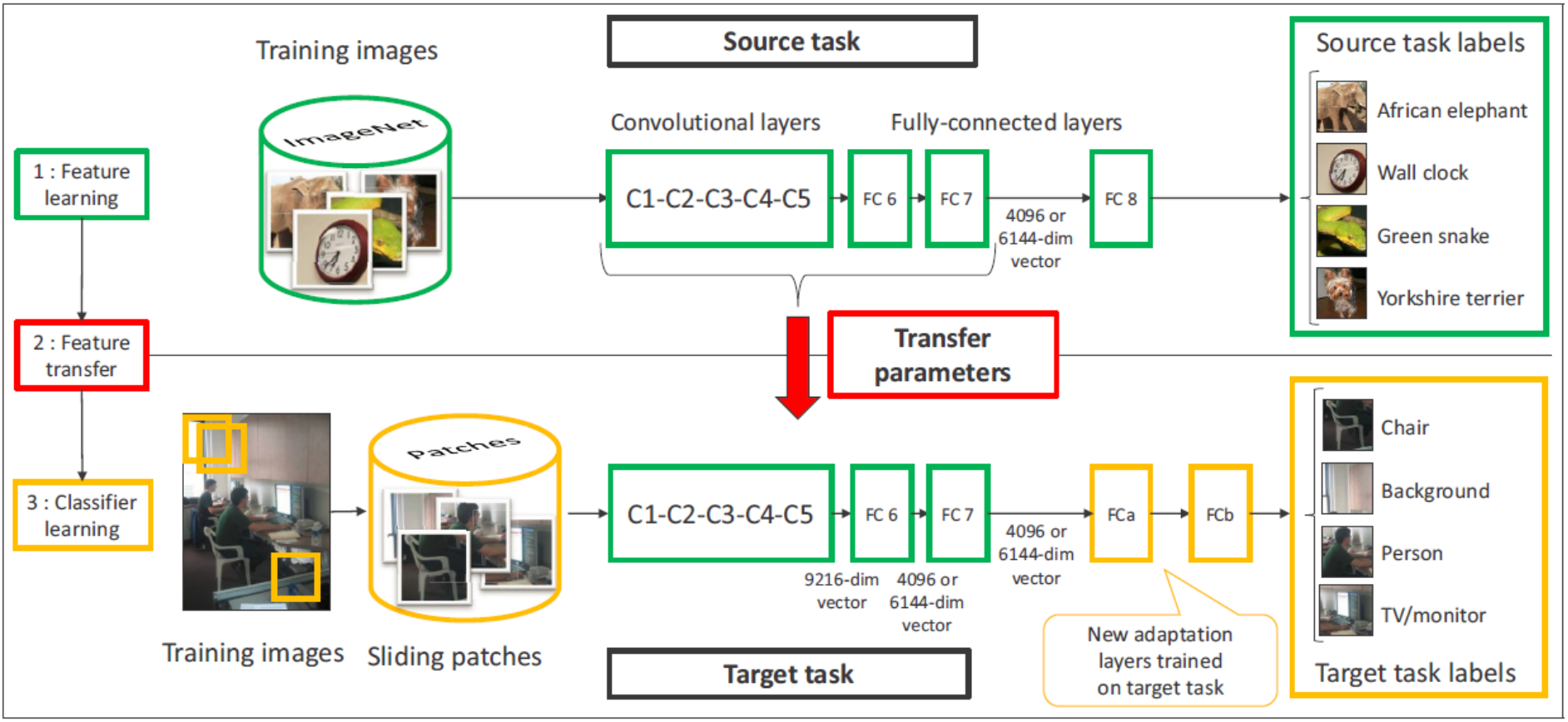
Figure 1. An example of Lifelong Machine Learning test (upper) and an illustration of the empirical performance requirement (bottom).

Transfer Learning in Convolutional Neural Networks

- Convolutional neural networks (CNN): outstanding image-classification.
- Learning CNNs requires a very large number of annotated image samples
  - Millions of parameters, to many that prevents application of CNNs to problems with limited training data.
- Key Idea:
  - the internal layers of the CNN can act as a generic extractor of mid-level image representation
  - Model-based Transfer Learning

The Transferring Framework

Oquab, Bottou, Laptev, Sivic: Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. **CVPR 2014**.



Transfer Learning in Convolutional Neural Networks

- Source Domain: ImageNet
  - 1000 classes, 1.2 million images
- Target Domain: Pascal VOC 2007 object classification
  - 20 classes, about 5000 images
- PRE-1000C: the proposed method

	plane	bike	bird	boat	btl	bus	car	cat	chair	cow
INRIA [33]	77.5	63.6	56.1	71.9	33.1	60.6	78.0	58.8	53.5	42.6
NUS-PSL [46]	82.5	79.6	64.8	73.4	**54.2**	75.0	77.5	79.2	46.2	62.7
PRE-1000C	**88.5**	**81.5**	**87.9**	**82.0**	47.5	**75.5**	**90.1**	**87.2**	**61.6**	**75.7**

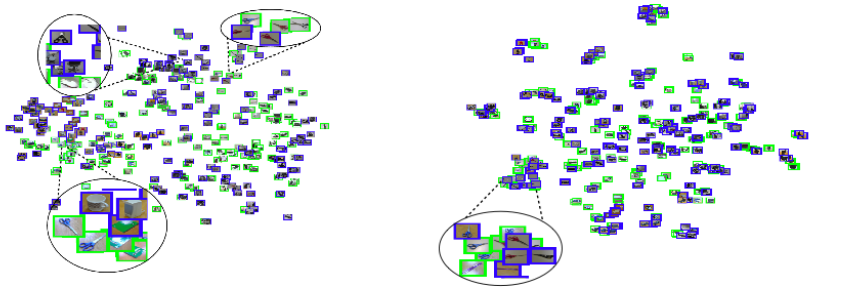
table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	**mAP**
54.9	45.8	77.5	64.0	85.9	36.3	44.7	50.6	79.2	53.2	59.4
41.4	74.6	**85.0**	76.8	91.1	53.9	61.0	**67.5**	83.6	70.6	70.5
67.3	**85.5**	83.5	**80.0**	**95.6**	**60.8**	**76.8**	58.0	**90.4**	**77.9**	**77.7**

Per-class results for object classification on the VOC2007 test set (average precision %)

DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition

- Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, Trevor Darrell
- ICML2014
- Questions: transferring features to tasks with different labels
  - Do features extracted from the CNN generalize to other datasets?
  - How does performance vary with network depth?
- Algorithm:
- A deep convolutional model is first trained in a fully supervised setting using a state-of-the-art method Krizhevsky et al. (2012).
- We then extract various features from this network, and evaluate the efficacy of these features on generic vision tasks.

Comparison: DECAF to others



(a) SURF features (b) DeCAF$_6$

ure 5. Visualization of the webcam (green) and dslr (blue) domains using the original released SURF features (a) and DeCAF$_6$ (
e figure is best viewed by zooming in to see the images in local regions. All images from the scissor class are shown enlarged. Th
well clustered and overlapping in both domains with our representation, while SURF only clusters a subset and places the others
joint parts of the space, closest to distinctly different categories such as chairs and mugs.

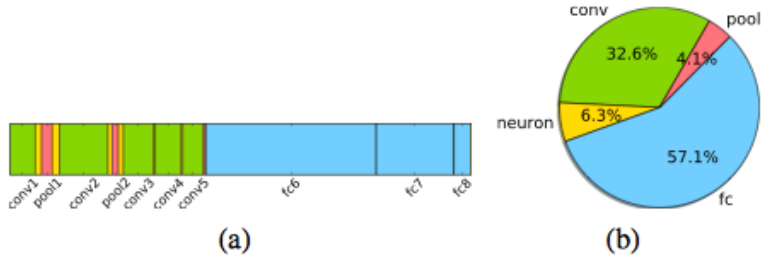
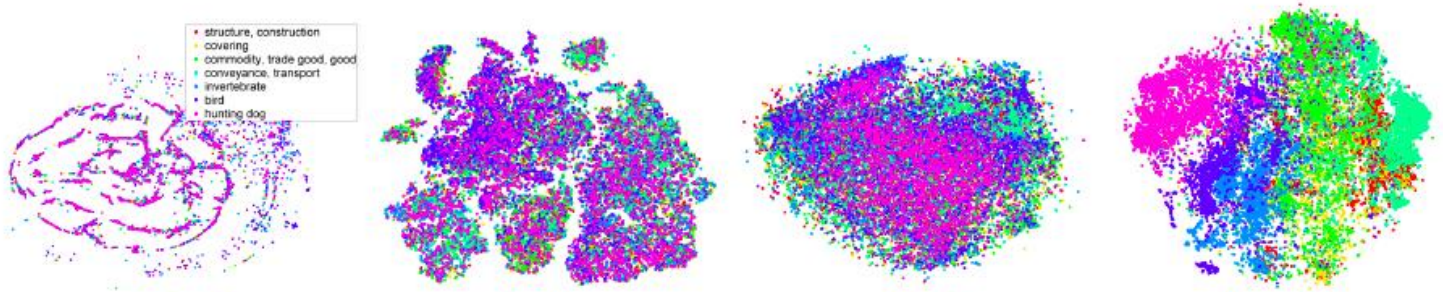


Figure 3. (a) The computation time on each layer when running classification on one single input image. The layers with the most time consumption are labeled. (b) The distribution of computation

DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition



(a) LLC (b) GIST (c) DeCAF$_1$ (d) DeCAF$_6$

Figure 1. This figure shows several t-SNE feature visualizations on the ILSVRC-2012 validation set. (a) LLC , (b) GIST, and features derived from our CNN: (c) DeCAF$_1$, the first pooling layer, and (d) DeCAF$_6$, the second to last hidden layer (best viewed in color).

Reinforcement Transfer Learning via Sparse Coding

- Slow learning speed remains a fundamental problem for reinforcement learning in complex environments.
- Main problem: the numbers of states and actions in the source and target domains are different.
  - Existing works: hand-coded inter-task mapping between state-action pairs
- Tool: new transfer learning based on sparse coding

Ammar, Tuyls, Taylor, Driessens, Weiss: Reinforcement Learning Transfer via Sparse Coding. AAMAS, 2012.

Reinforcement Transfer Learning via Sparse Coding

- Given State-Action-State Triplets in the source task, learn dictionary as

$$\min_{\{\mathbf{b}_j\},\{a_j^{(i)}\}} \sum_{i=1}^{m} \frac{1}{2\sigma^2} ||\langle s_0, a_0, s_0^{'}\rangle^{(i)} - \sum_{j=1}^{d_1} \mathbf{b}_j a_j^{(i)}||_2^2 + \beta \sum_{i=1}^{m}\sum_{j=1}^{d_1} ||a_j^{(i)}||_1 \;\; s.t. \;\; ||\mathbf{b}_j||_2^2 \leq c, \forall j = \{1, 2, \ldots, d_1\}$$

- Using the coefficient matrix in the first step, we can learn the dictionary in the target task as

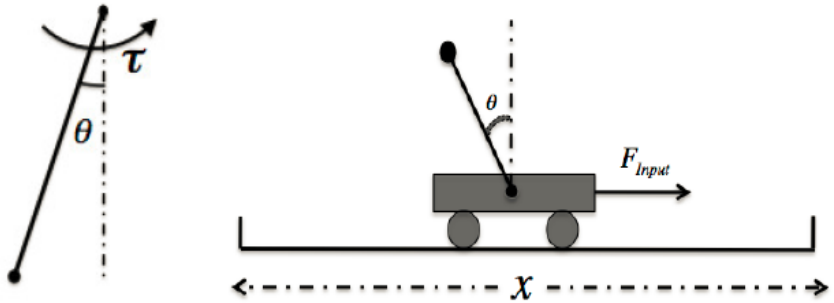
$$\min_{\{\mathbf{z}_j\},\{c_j^{(i)}\}} \sum_{i=1}^{m} \frac{1}{2\sigma^2} ||\langle \mathbf{a}_{1:d_1}\rangle^{(i)} - \sum_{j=1}^{d_n} \mathbf{z}_j c_j^{(i)}||_2^2 + \beta \sum_{i=1}^{m}\sum_{j=1}^{d_n} ||c_j^{(i)}||_1 \;\; s.t. \;\; ||\mathbf{z}_j||_2^2 \leq o, \forall j = \{1, 2, \ldots, d_n\}$$

- Then for each triplet in the target task, - sparse projection is used to find its coefficients

$$\hat{\phi}^{(i)}(\langle s_t, a_t, s_t^{'}\rangle) \;\; = \;\; \arg\min_{\phi^{(i)}} ||\langle s_t, a_t, s_t^{'}\rangle^{(i)} - \sum_{j=1}^{d_n} \phi_j^{(i)} \mathbf{z}_j||_2^2 \; + \beta ||\phi^{(i)}||_1$$

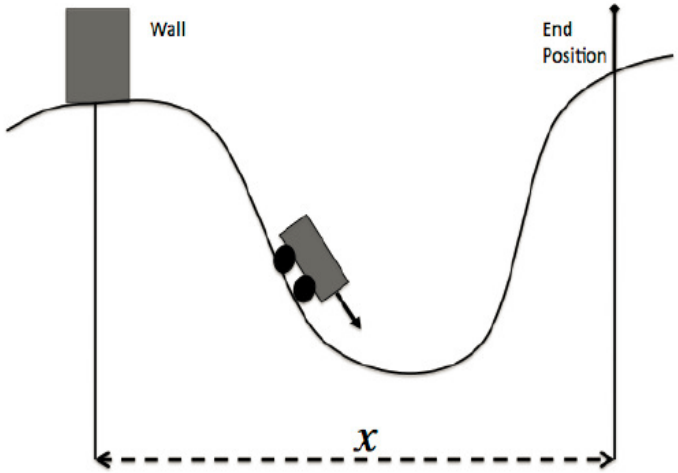
- As a result, the inter-task mapping can be learned!

Reinforcement Learning Transfer via Sparse Coding



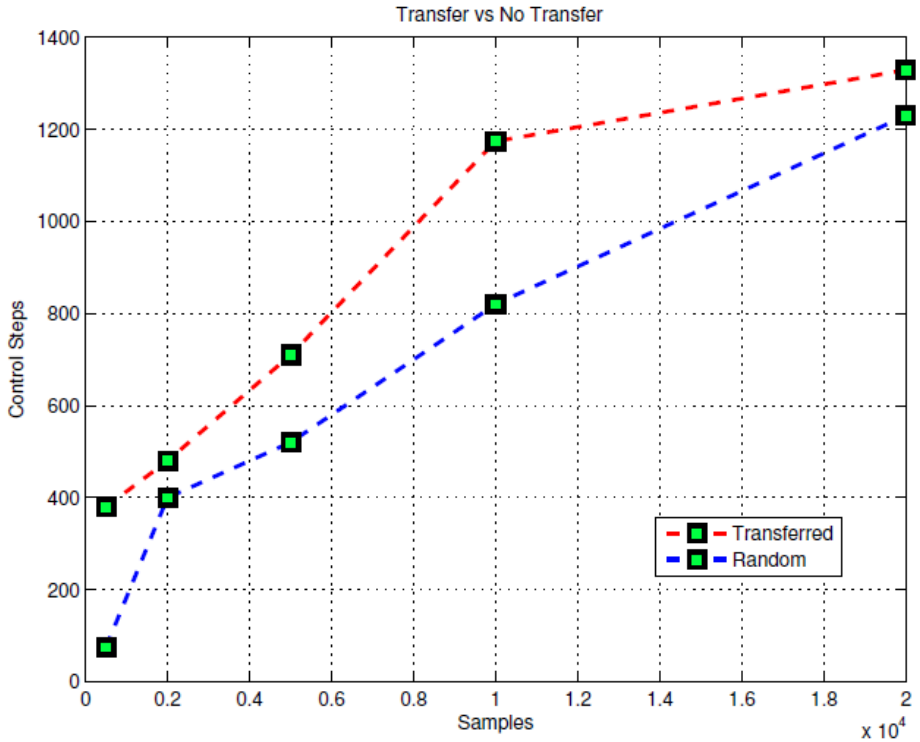
(a) Inverted Pendulum

(b) Cart Pole



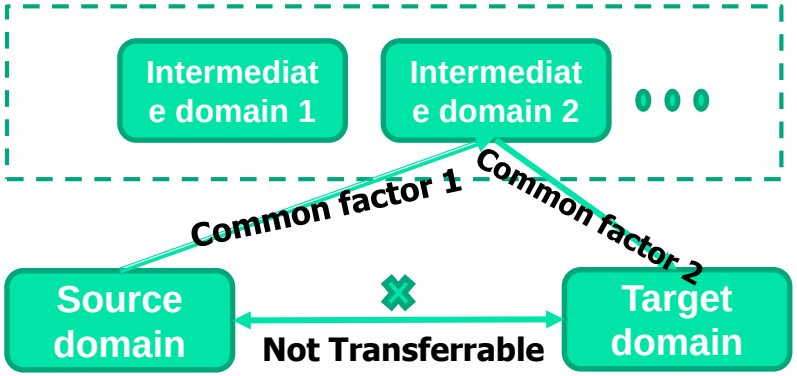
(c) Mountain Car

Authors measured the performance as the number of steps during an episode to control the pole in an upright position on a given fixed amount of samples.



Transitive Transfer Learning

- Source and target domains have no overlap
  - May we use intermediate domains as bridge?
- Can we build a path of transfer learning?



Text-to-image Classification with co-occurrence data as intermediate domain

Transitive Transfer Learning

- Intermediate domain selection, then propagate knowledge
  - Use domain distance, such as A-distance, to identify domains
  - Transitive trnasfer through shared hidden factors in a row

Coupled the two knowledge transfer processes

$$\min_{U,S,V\geq 0} \left\| X_s - \begin{bmatrix} U_{si} \\ U_{ss} \end{bmatrix} \begin{bmatrix} S_{si} \\ S_{ss} \end{bmatrix} V_s^T \right\|_F^2 + \left\| X_i - \begin{bmatrix} U_{si} \\ U_{ii} \end{bmatrix} \begin{bmatrix} S_{si} \\ S_{ii} \end{bmatrix} V_i^T \right\|_F^2 + \left\| X_i - \begin{bmatrix} U_{it} \\ U'_{ii} \end{bmatrix} \begin{bmatrix} S_{it} \\ S'_{ii} \end{bmatrix} V_i^T \right\|_F^2 + \left\| X_t - \begin{bmatrix} U_{it} \\ U'_{tt} \end{bmatrix} \begin{bmatrix} S_{it} \\ S'_{tt} \end{bmatrix} V_t^T \right\|_F^2$$

Transfer Knowledge between the source and intermediate domains

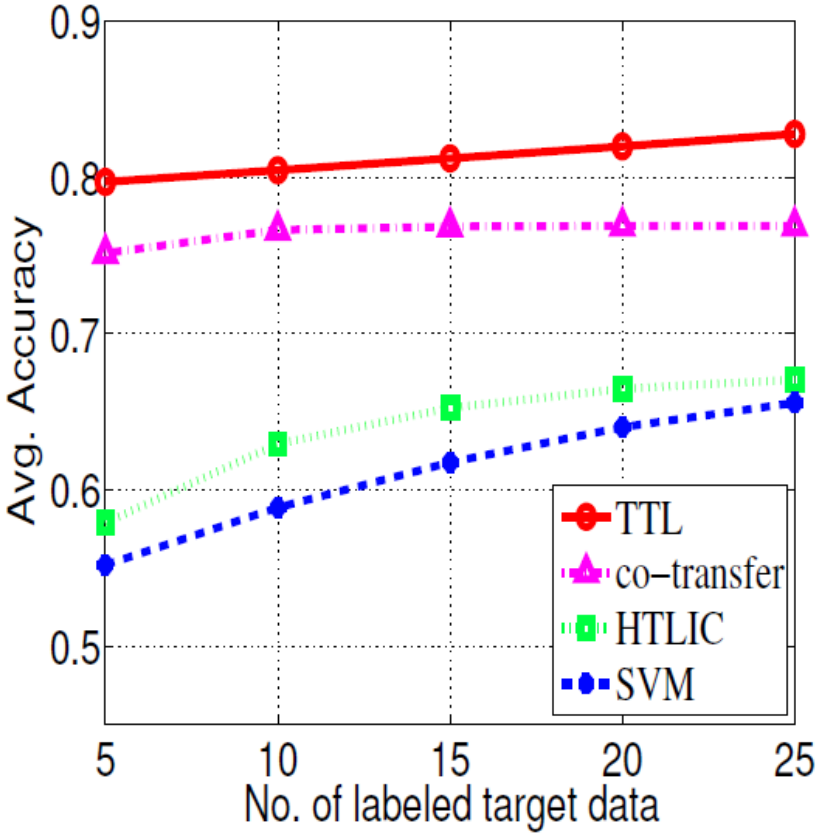
Transfer Knowledge between the intermediate and target domains

$$s.t. U_i^T \mathbf{1} = \mathbf{1}, V_i^T \mathbf{1} = \mathbf{1}, i \in \{s, i, t\}$$

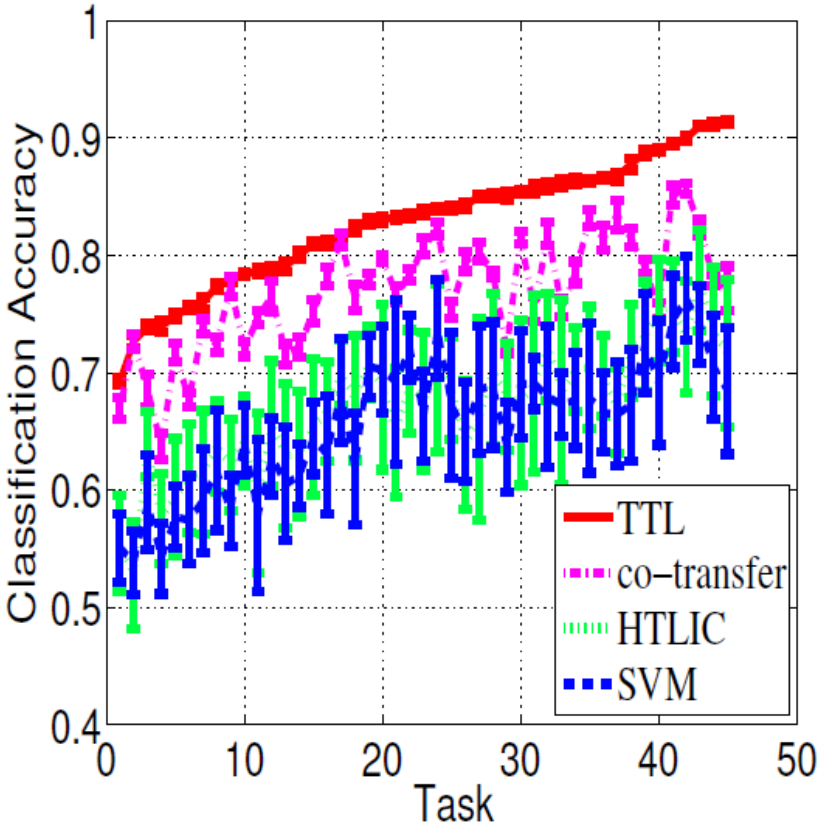
B. Tan, YQ Song, E. Zhong, Q. Yang: Transitive Transfer Learning. ACM KDD 2015.

Transitive Transfer Learning

- The NUS-WISE data set are used
  - 45 text-to-image tasks.
  - Each task is composed of 1200 text documents, 600 images, and 1600 co-occurred text-image pairs.



(a) Average performance



(b) Detailed performance

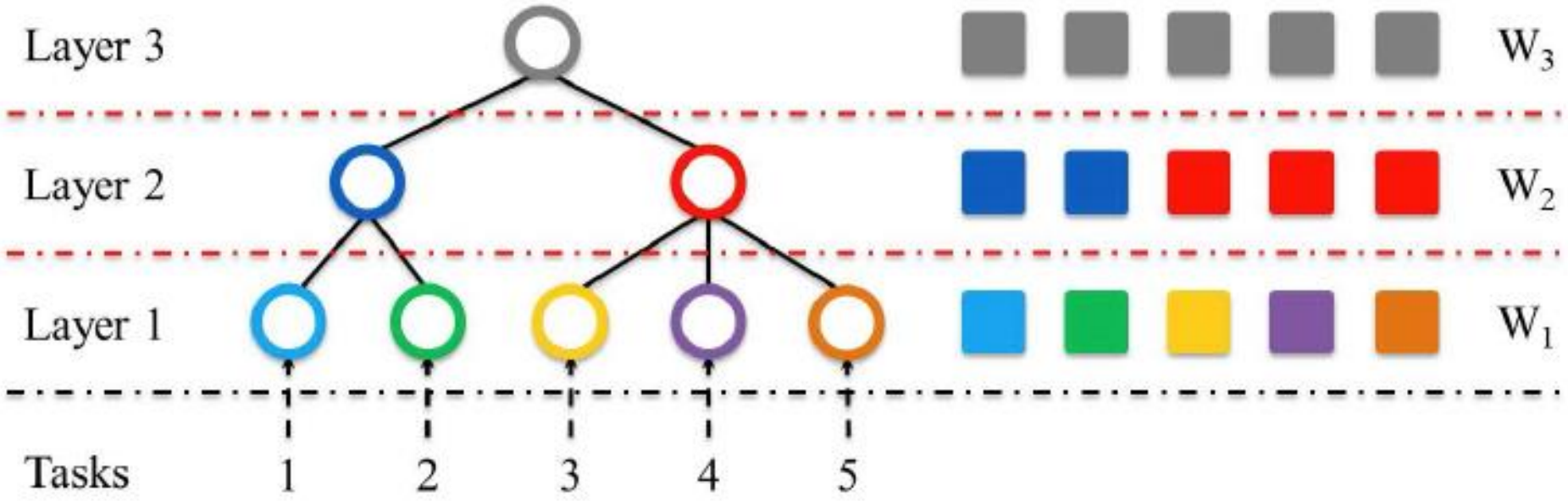
Learning Task Trees

Han and Zhang:
Learning Tree Structure in Multi-Task Learning.
ACM KDD 2015.

- Learning task relations in transfer learning:
  - m tasks, decompose **W** into *H* components

$$\mu_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x}, \qquad \mathbf{W} = [\mathbf{w}_1, \cdots, \mathbf{w}_m] \in \mathbb{R}^{d \times m}$$

$$\mathbf{W} = \sum_{h=1}^{H} \mathbf{W}_h. \qquad \mathbf{W}_h = [\mathbf{w}_{h,1}, \cdots, \mathbf{w}_{h,m}] \in \mathbb{R}^{d \times m}$$



Learning Tree Structure among Tasks

- The objective function is formulated as

$$\min_{\mathbf{W}} \mathcal{L}(\mathbf{W}) + \sum_{h=1}^{H} \lambda_h \sum_{i<j}^{d} \|\mathbf{w}_{h,i} - \mathbf{w}_{h,j}\|_2$$

$$\text{s.t.} \; |\mathbf{w}_{h-1,i} - \mathbf{w}_{h-1,j}| \succeq |\mathbf{w}_{h,i} - \mathbf{w}_{h,j}| \; \forall h \geq 2, \forall i < j,$$

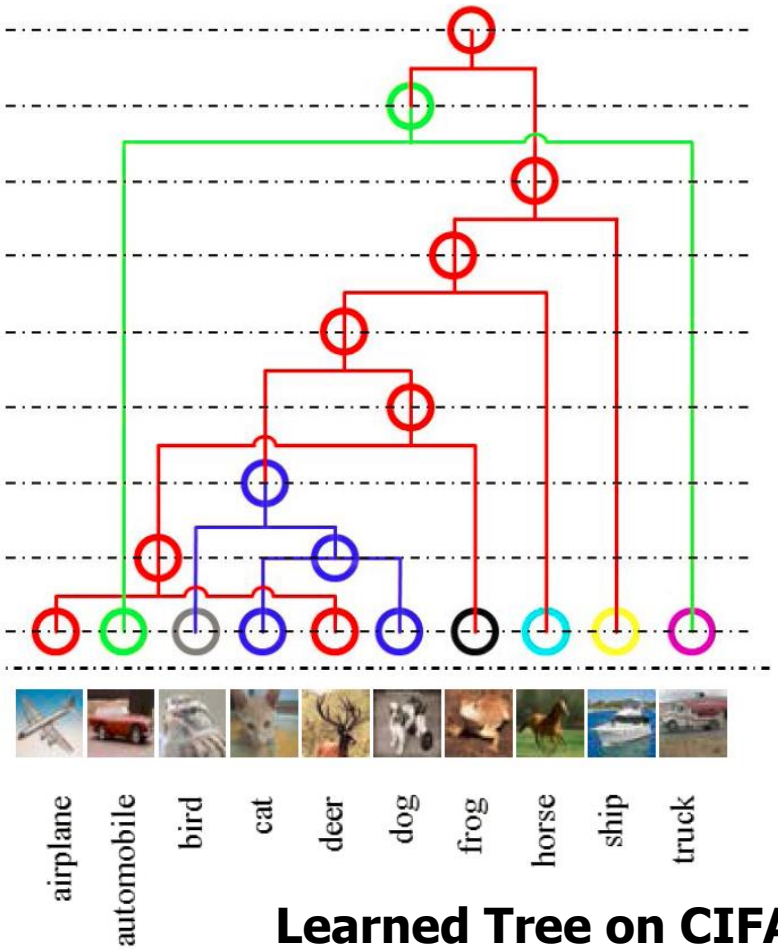
To make the model form a task tree

- λ_h controls the strength of the task similarity at the *h*-th layer.
  - A proximal method is used to solve this problem.

Learning Tree Structure among Tasks

- Two object recognition databases, the CIFAR-10 and CIFAR-100 datasets, are used
  - Each dataset consists of 50,000 color images for training and 10,000 images for testing.
  - CIFAR-10: 10 classes; CIFAR-100: 100 classes
- Performance measure: Accuracy

Data	MTFL	Dirty	Cascade	CMTL	MeTaG	TAT
CIFAR-10	71.15	72.56	74.76	74.49	75.99	**76.49**
CIFAR-100	39.61	42.45	46.12	46.70	48.62	**49.63**



Learned Tree on CIFAR-10

Some findings:

(1) Tasks 'cat' and 'dog' always belong to the same group in the task tree;

(2) All tasks related to animals (i.e., bird, cat, deer, dog, frog, and horse) are discovered to belong to a group at the 5th layer and above

Conclusions

- Transfer Learning
  - When training and application domains differ
  - Transfer instances, features, topic models, dictionary, hidden layers, concept trees
  - Lifelong Machine Learning
- Future
  - The Case-based Reasoning Challenge: Reduce the number of source domain examples to few, or even one?