# Learning Loss for Active Learning

Donggeun Yoo[1,2] and In So Kweon[2]

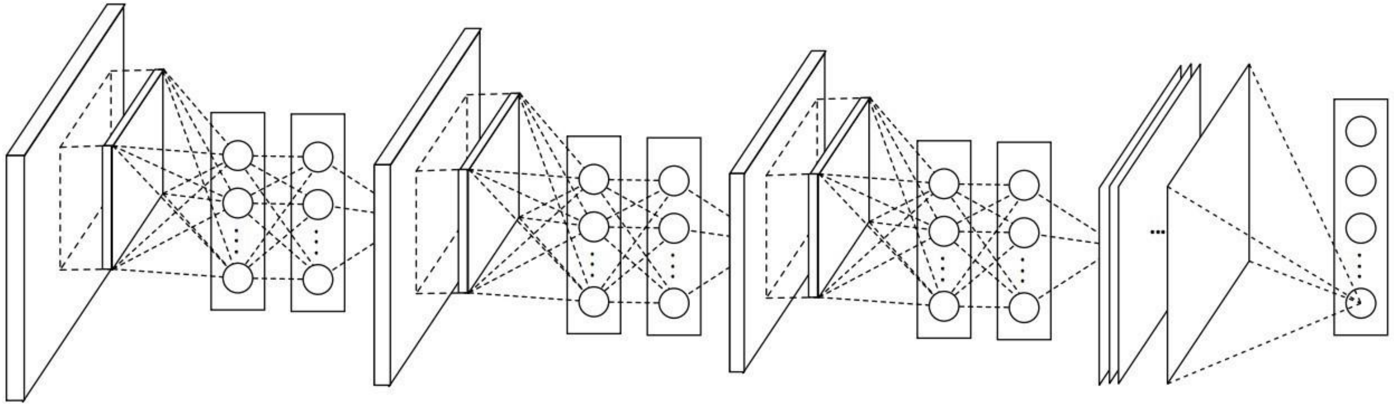
[1]Lunit Inc., Seoul, South Korea.
[2]KAIST, Daejeon, South Korea.

dgyoo@lunit.io iskweon77@kaist.ac.kr

CVPR 2019

발표자 : 김용규

# Active Learning




- model

- data selection

## Why? & When?

- Supervised Learning Setting > Unsupervised Learning, Semi-Supervised Learning

[42] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems*, pages 3546–3554, 2015.

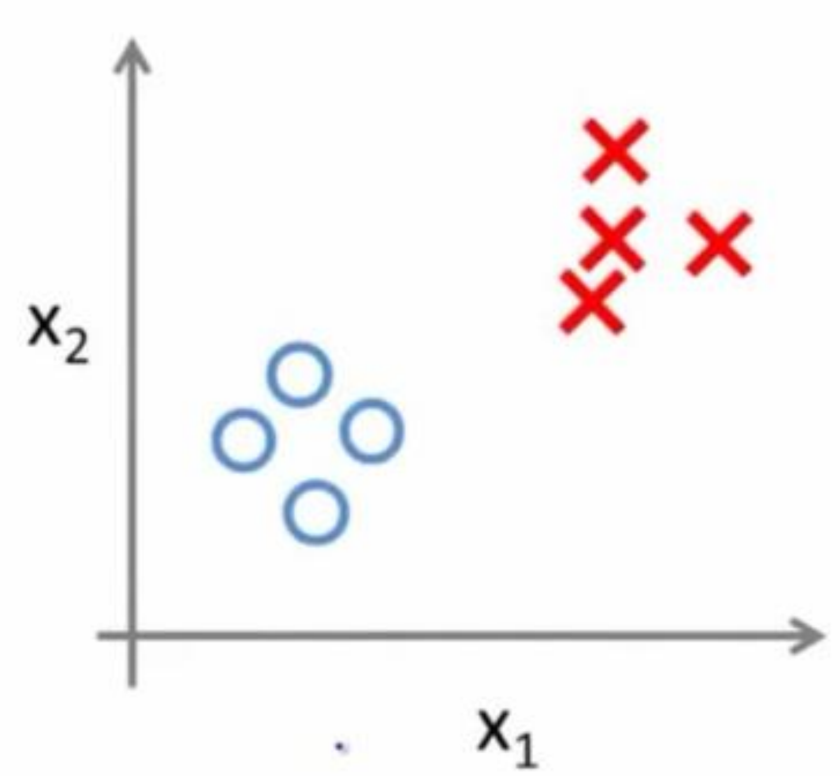
[45] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

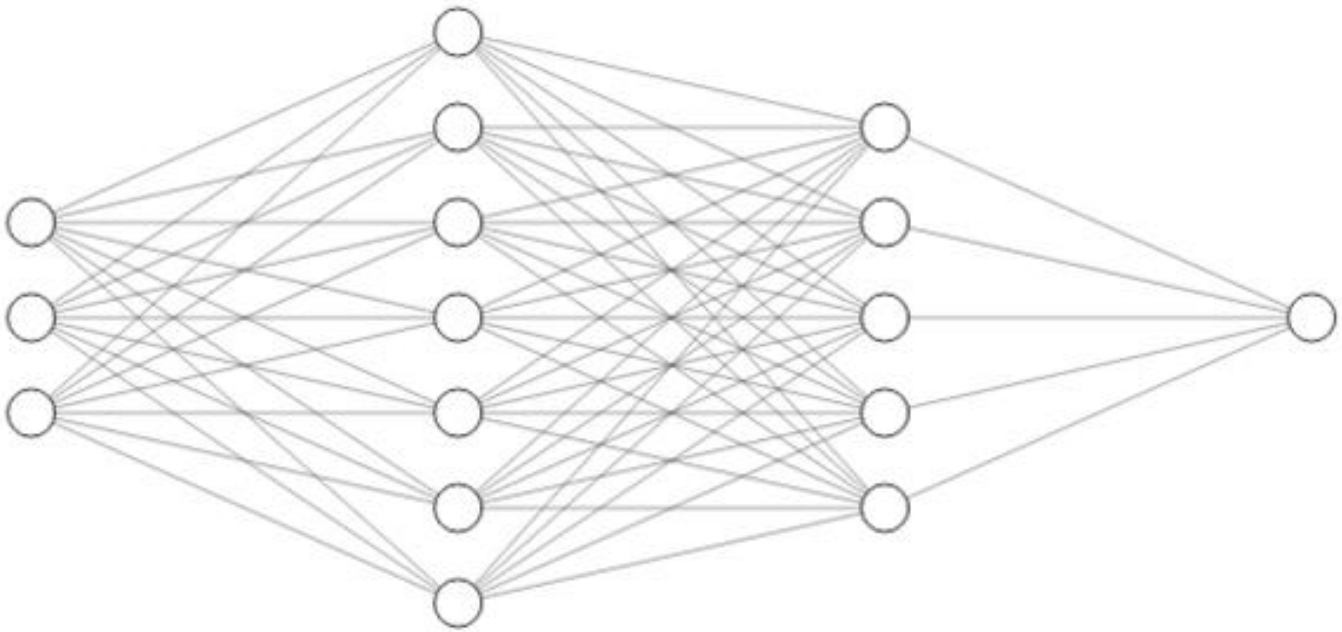
- Annotation time(burden) shortage (e.g. Medical image segmentation data)

# Active Learning

## Main Idea

Random sampling data 보다 Model이 선택한 유의미한 data로부터의 학습이 더 효율적이다.



# Existing Active Learning

1. The uncertainty approach
 - Previous slide

2. The diversity approach
 - Selects diverse data points that represent the whole distribution of the unlabeled data.

3. Expected model change
 - Selects data points that would cause the greatest change to the current model parameters or outputs if we knew their labels.

# Introduction

- Existing active learning is Specific for their target task.
  For more complex recognition tasks, it is required to re-define task-specific uncertainty such as object detection, semantic segmentation, and human pose estimation.

- Existing active learning is computationally indfficient for large networks.
  As a task-agnostic uncertainty approach, [49, 4] train multiple models to construct a committee, and measure the consensus between the multiple predictions from the committee.

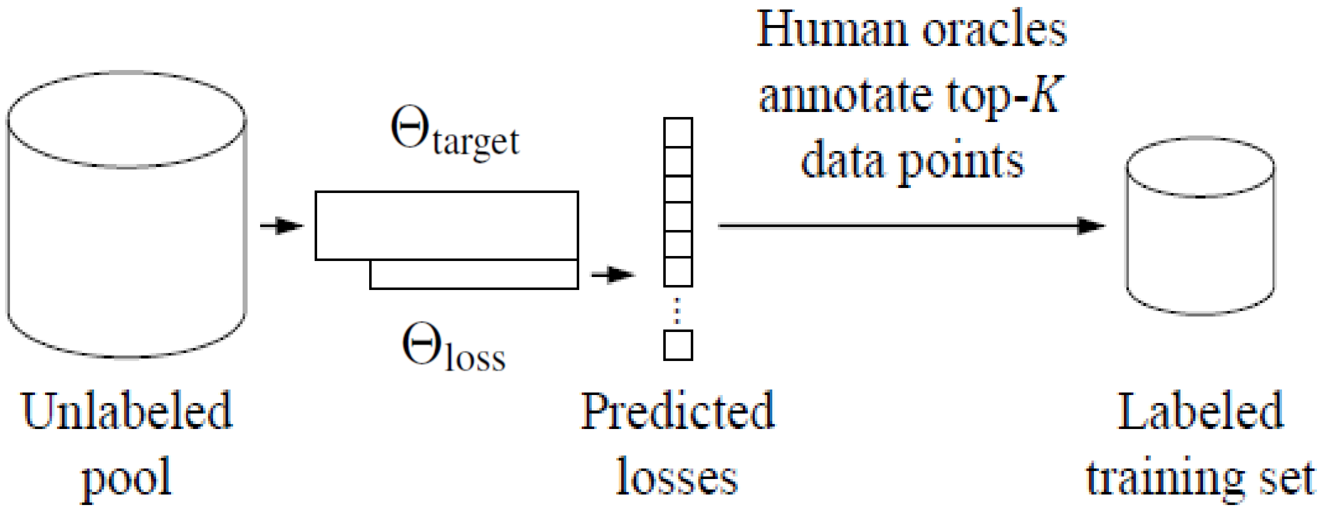
# Contribution

- The method is task-agnostic as networks are learned from a single loss regardless of target tasks.
  Loss prediction module

- The method works efficiently with the deep networks.

- The method works well in Classification, Regression, Hybrid model

# Method



(b) Active learning with a loss prediction module

- Unlabeled data : $\mathcal{U}_N$
- Initial labeled data : $\mathcal{L}_K^0$

- Initial target model : $\Theta^0_{\text{target}}$
- Initial loss prediction module : $\Theta^0_{\text{loss}}$

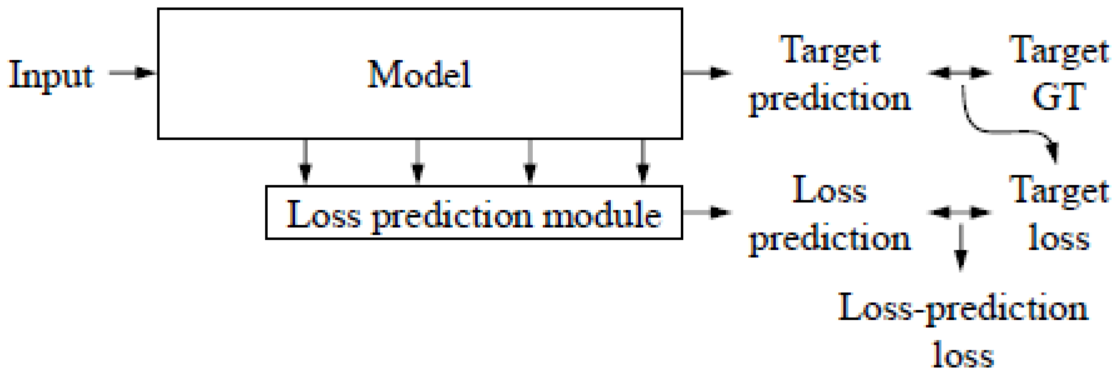
- Update labled data : $\mathcal{L}_{2K}^1$
- model : $\{\Theta^1_{\text{target}}, \Theta^1_{\text{loss}}\}$

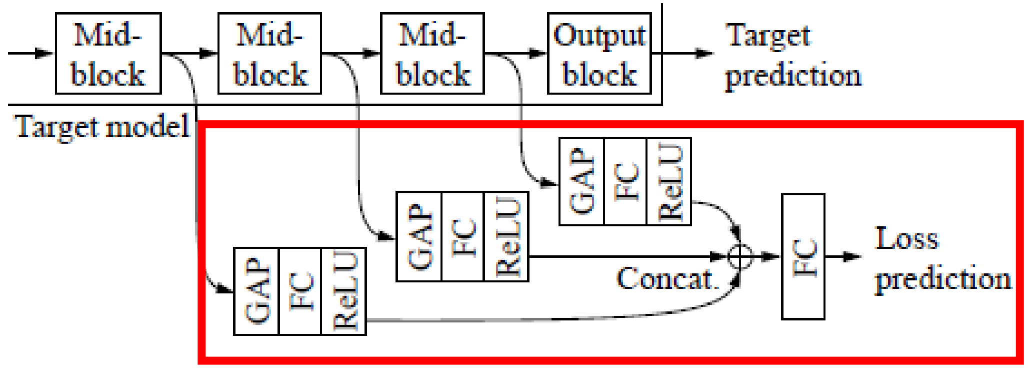
•
•
•

- The model reaches the desired performance

- End of data annotation

# Method




- Model : $\hat{y} = \Theta_{\text{target}}(x)$
- Loss prediction module : $\hat{l} = \Theta_{\text{loss}}(h)$

The s-th active learning stage

Label dataset : $\mathcal{L}^{s}_{K\cdot(s+1)}$

Model : $\{\Theta^{s}_{\text{target}}, \Theta^{s}_{\text{loss}}\}$

1. Loss prediction, Target prediction
2. Target GT selection
3. Target loss : $l = L_{\text{target}}(\hat{y}, y)$
4. Loss-prediction loss : $L_{\text{loss}}(\hat{l}, l)$
5. Total loss : $L_{\text{target}}(\hat{y}, y) + \lambda \cdot L_{\text{loss}}(\hat{l}, l)$

# Method

Loss-prediction loss function

MSE : $L_{\text{loss}}(\hat{l}, l) = (\hat{l} - l)^2$

However, MSE is not a suitable choice for this problem since the scale of the real loss $l$ changes (decreases in overall) as learning of the target model progresses. Minimizing MSE would let the loss prediction module adapt roughly to the scale changes of the loss $l$, rather than fitting to the exact value.

- 모델이 학습됨에 따라서 loss의 scale이 변화

- 모델이 정확한 loss 값에 fitting되는 게 아니라 loss의 scale 변화에 적응 (경향성에 의존)

Difference between a pair of loss predictions

mini-batch : $\mathcal{B}^s \subset \mathcal{L}^s_{K\cdot(s+1)}$

Data pair : $\{x^p = (x_i, x_j)\}$

$$L_{\text{loss}}(\hat{l^p}, l^p) = \max\left(0, -\mathbb{1}(l_i, l_j) \cdot (\hat{l}_i - \hat{l}_j) + \xi\right)$$
$$\text{s.t.} \quad \mathbb{1}(l_i, l_j) = \begin{cases} +1, & \text{if} \quad l_i > l_j \\ -1, & \text{otherwise} \end{cases} \tag{2}$$

where $\xi$ is a pre-defined positive margin

when $l_i > l_j$

If $\hat{l}_i > \hat{l}_j + \xi$, then loss = 0

# Method

$$\frac{1}{B}\sum_{(x,y)\in\mathcal{B}^s} L_{\text{target}}(\hat{y}, y) + \lambda\frac{2}{B}\cdot\sum_{(x^p,y^p)\in\mathcal{B}^s} L_{\text{loss}}(\hat{l^p}, l^p)$$

$$\text{s.t.}\quad \begin{aligned} \hat{y} &= \Theta_{\text{target}}(x) \\ \hat{l^p} &= \Theta_{\text{loss}}(h^p) \\ l^p &= L_{\text{target}}(\hat{y^p}, y^p). \end{aligned} \qquad (3)$$

# Experiment

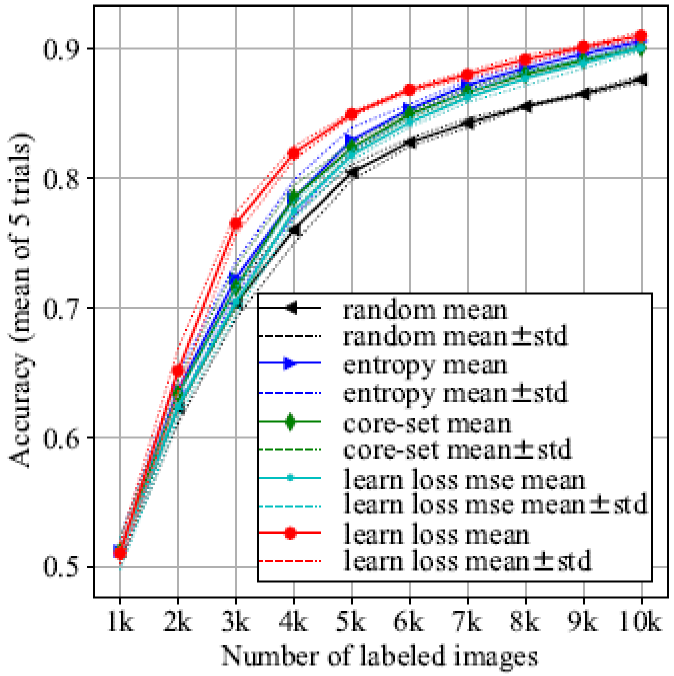



Figure 4. Active learning results of image classification over CIFAR-10.

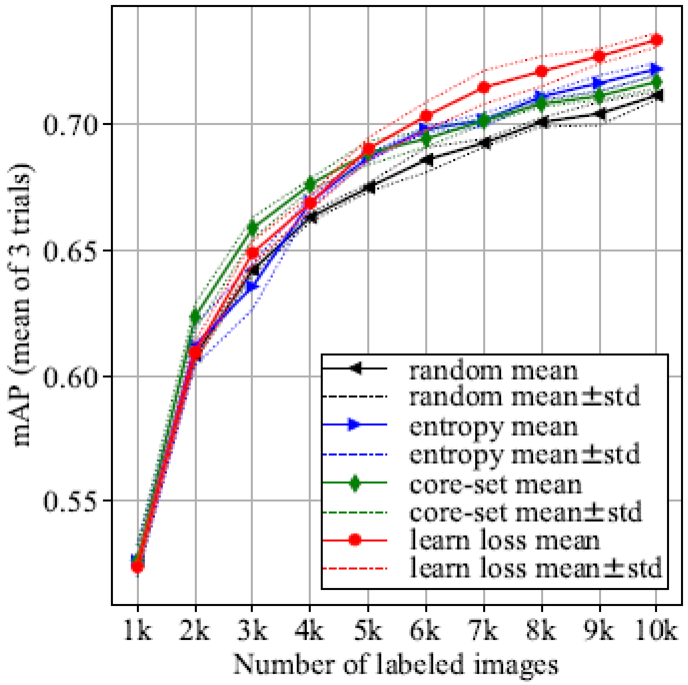



Figure 6. Active learning results of object detection over PASCAL VOC 2007+2012.

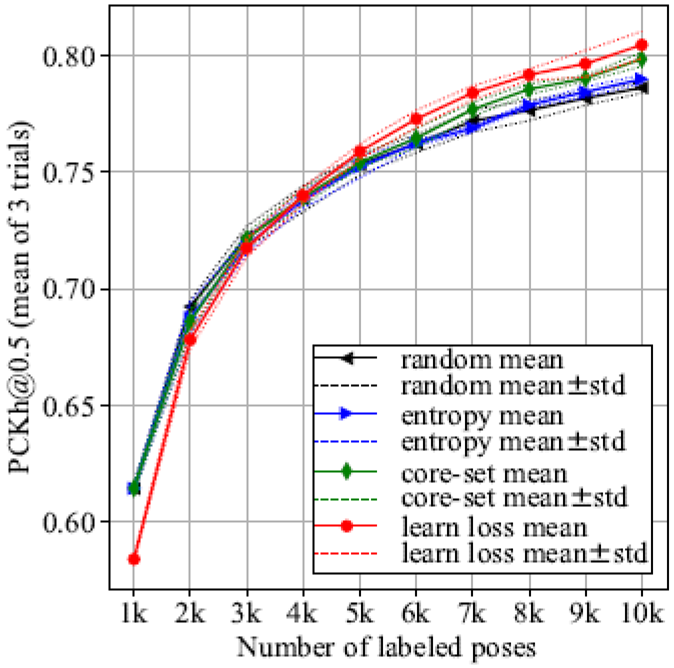



Figure 7. Active learning results of human pose estimation over MPII.

<classification>

<object detection>

<Human Pose Estimation>