PIC: Parametric Instance Classification for Unsupervised Visual Pre-training

Yue Cao*, Zhenda Xie*, Bin Liu*, Yutong Lin, Zheng Zhang, Han Hu
Microsoft Research Asia
Tsinghua University
Xi'an Jiaotong University

Pre-training

- NLP

Unsupervised Pre-training on nearly unlimited text corpus (e.g. BERT)

Fine-tuning on downstream tasks
- Text Classification
- Question Answering
- Commonsense Reasoning

- Computer Vision

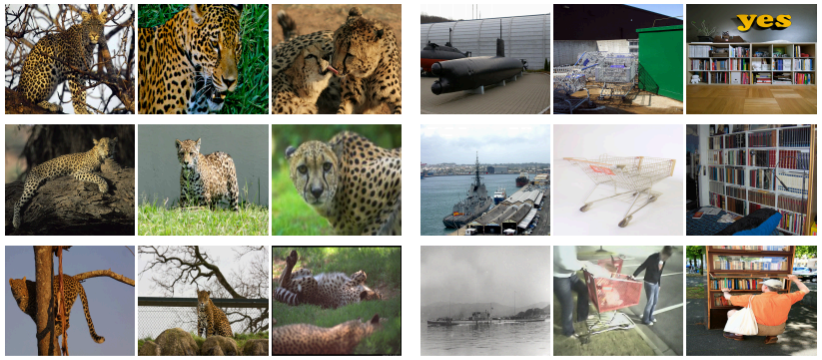
Supervised Pre-training on ImageNet

Fine-tuning on downstream tasks
- Object Detection
- Action Recognition
- Semantic Segmentation

Recent Progress in Unsupervised Visual Learning

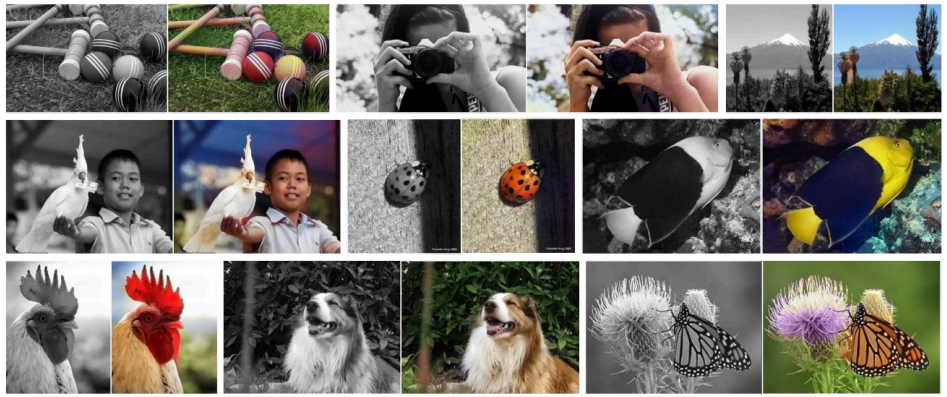
- Different ***pretext tasks*** in visual representation pretraining

Jigsaw Puzzle

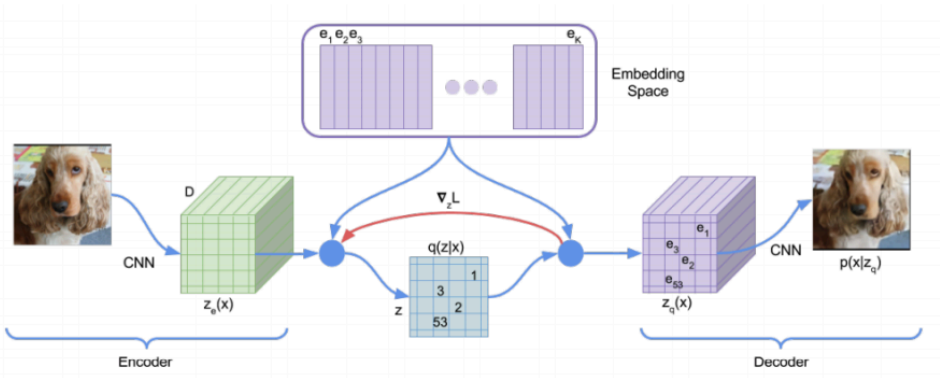


leopard jaguar cheetah lifeboat shopcart bookcase

Instance Discrimination



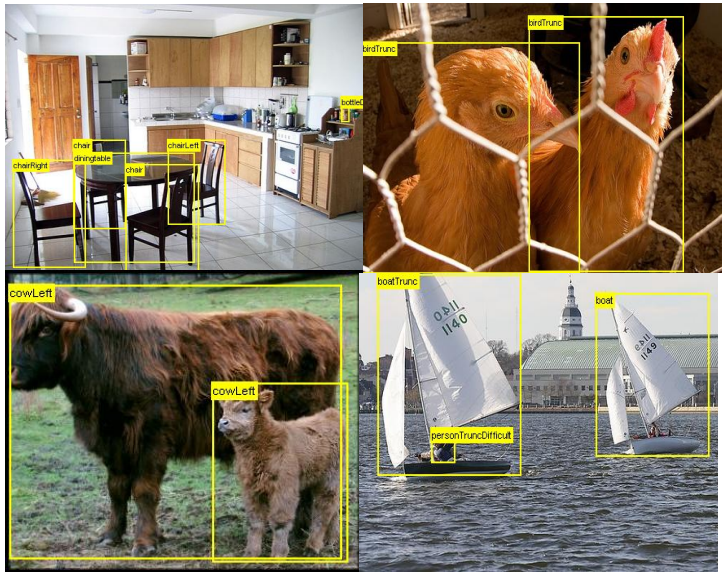
Colorization



Reconstruction

Recent Progress in Unsupervised Visual Learning

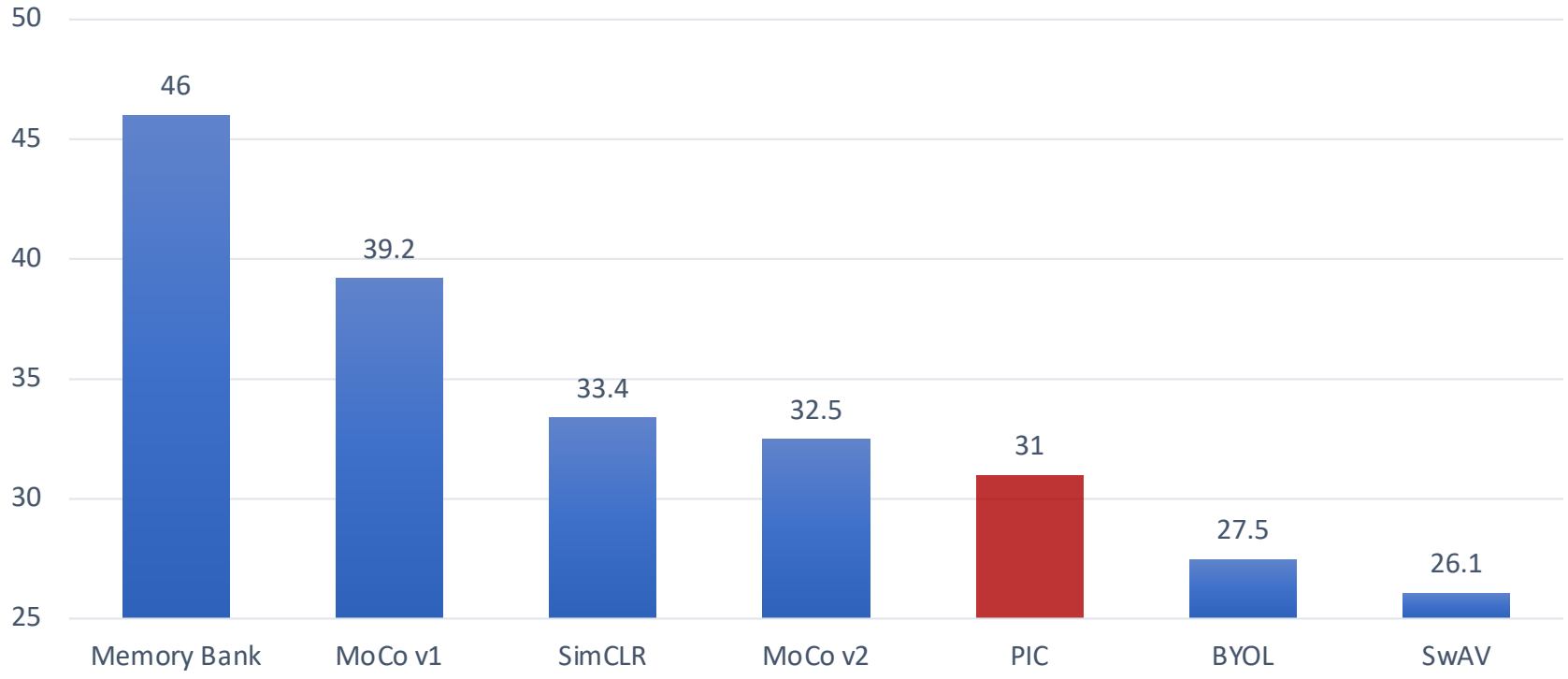
- Fine-tuning using an unsupervised pre-trained model could achieve on par or even better performance than that of supervised counterparts on downstream tasks



Object Detection on Pascal VOC

Method	AP50	AP	AP75
Supervised	81.3	53.5	58.8
MoCo v1	81.5	55.9	62.6
MoCo v2	82.4	57.0	**63.6**
PIC (ours)	**82.4**	**57.1**	**63.4**

Recent Progress in Unsupervised Visual Learning



Top-1 error using ResNet-50 with 200-epoch pre-training on ImageNet

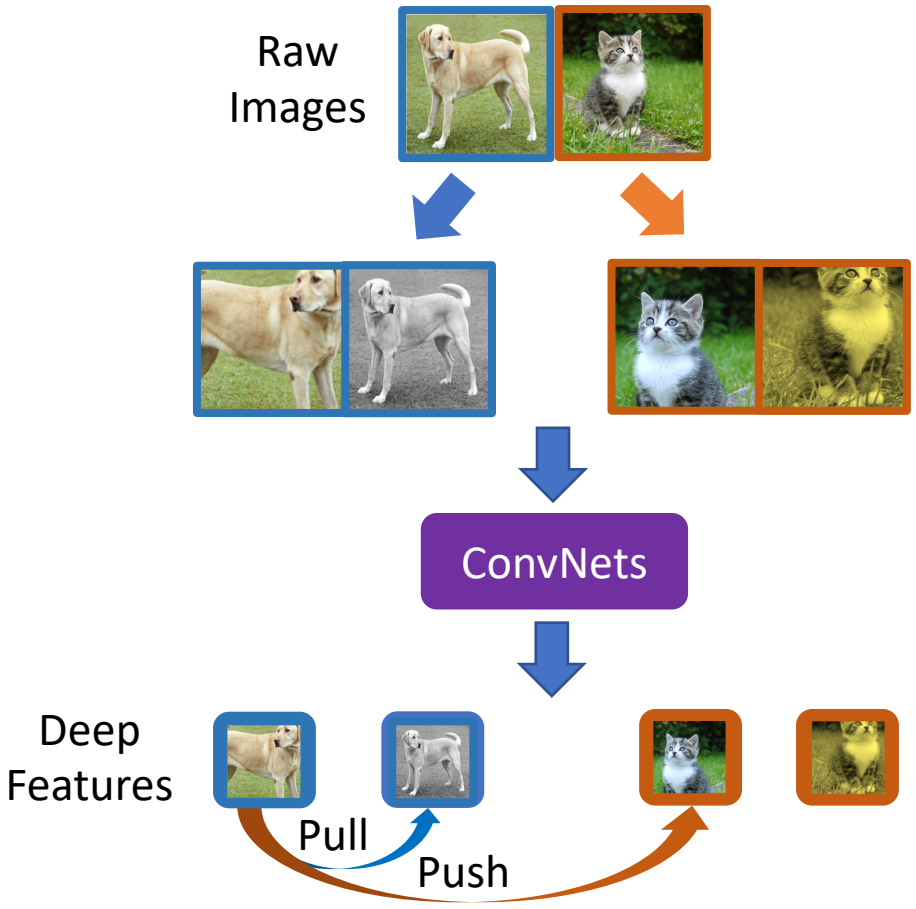
Unsupervised Learning via Instance Discrimination

Instance 1

Instance 2

Instance 3

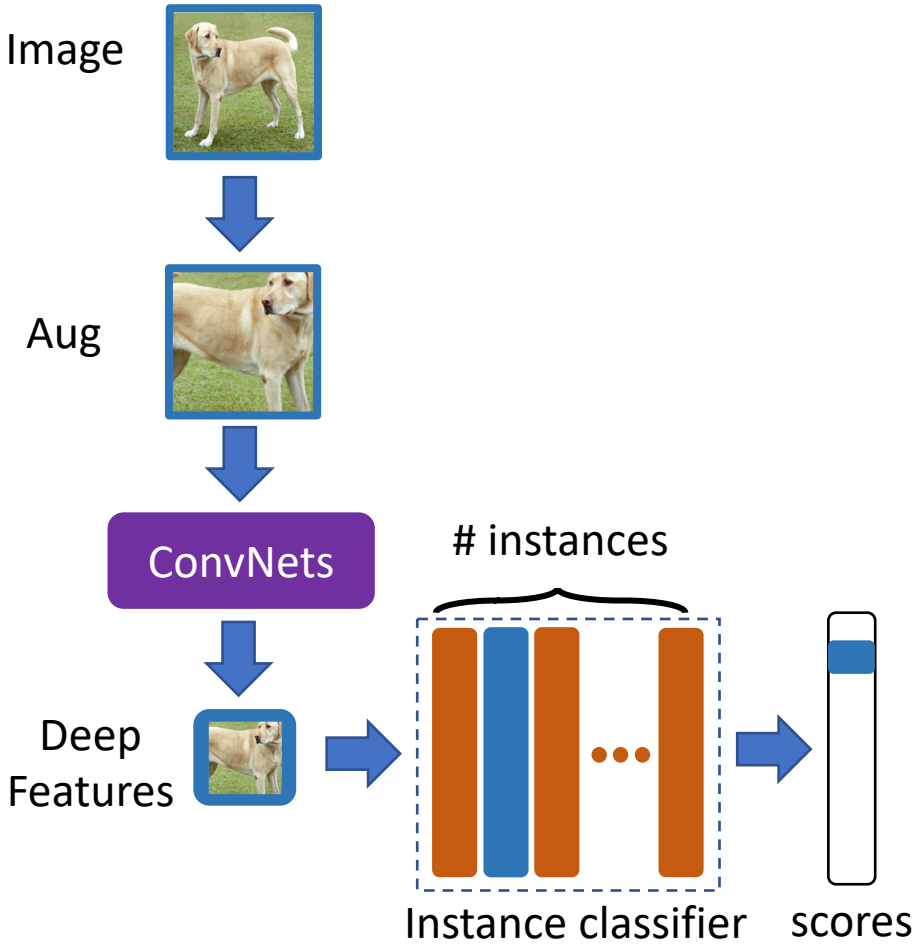
Motivation



- Non-Parametric Instance Classification
  - MoCo, He et al, CVPR20, FAIR
  - SimCLR, Hinton et al, ICML20, Google Brain
- Disadvantages
  - Complex
  - Information leakage: the network could find easy solution to distinguish positive and negative examples
    - BatchNorm would communicate between examples in the same iteration, which contains both positive and negative examples

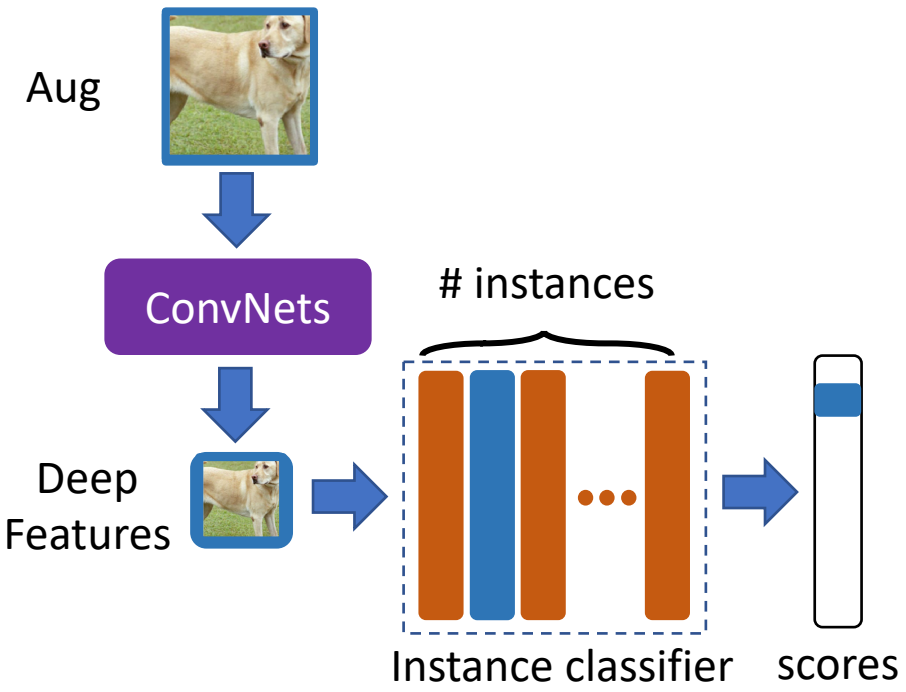
Could we make it simple but effective?

Parametric Instance Classification

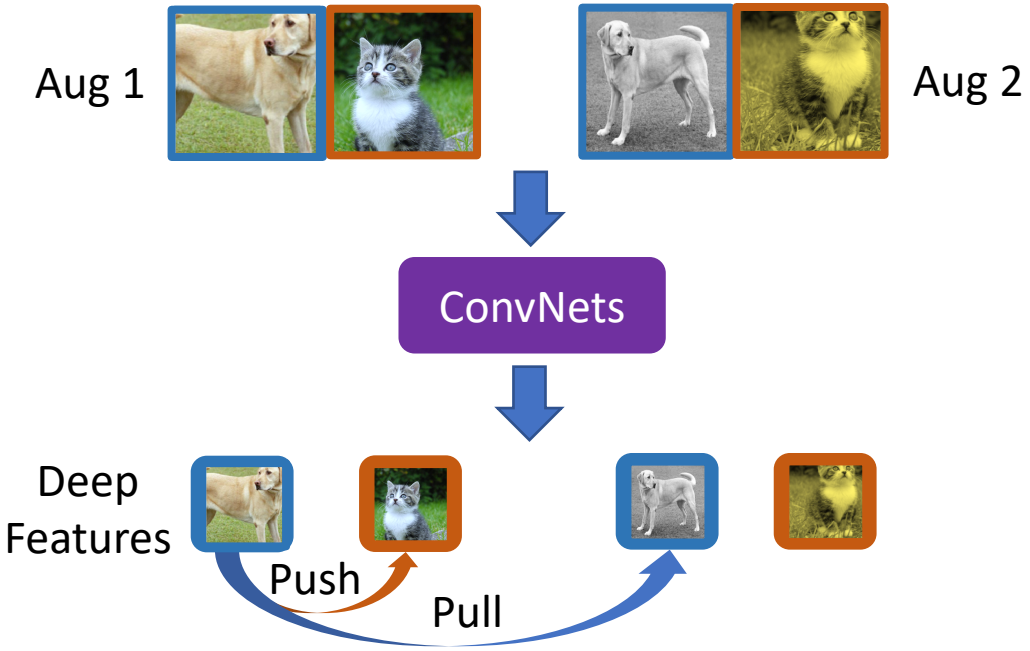


- Simple & easy to implement
- One augmentation per batch
  -> no information leakage
- Good performance?

Fair Comparison



One augmentation per iteration



Two augmentations per iteration

Fair Policy: Number of augmentations observed by the network should be the same.

Results - Linear Evaluation on ImageNet

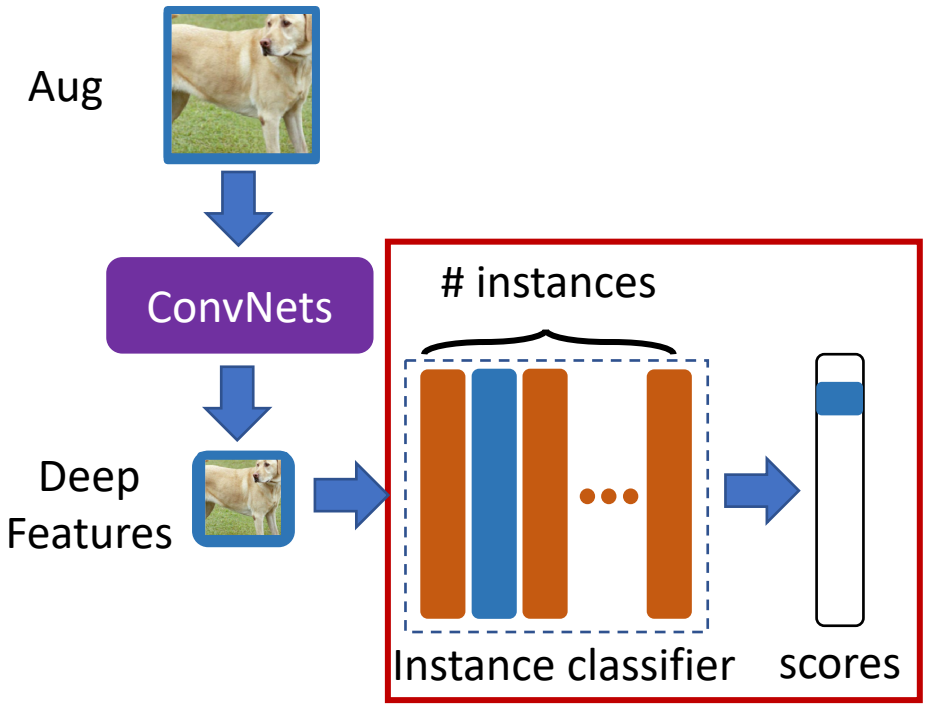
Method	#Aug/iter x Epoch	Top-1 Acc	Top-5 Acc
SimCLR	2 x 100 = 200	64.7	86.0
MoCo v2	2 x 100 = 200	64.1	85.7
PIC (ours)	1 x 200 = 200	**66.2** +1.5	**87.0**
SimCLR	2 x 200 = 400	66.6	87.3
MoCo v2	2 x 200 = 400	67.5	88.0
PIC (ours)	1 x 400 = 400	**68.5** +1.0	**88.8**

Major Concerns

- The weight matrix of classifier layer is too large
  - # instances $\times$ dimension, e.g. 1.28M $\times$ 128

- The weight matrix of classifier layer is updated too slowly
  - Each instance vector is updated as positive only once per epoch

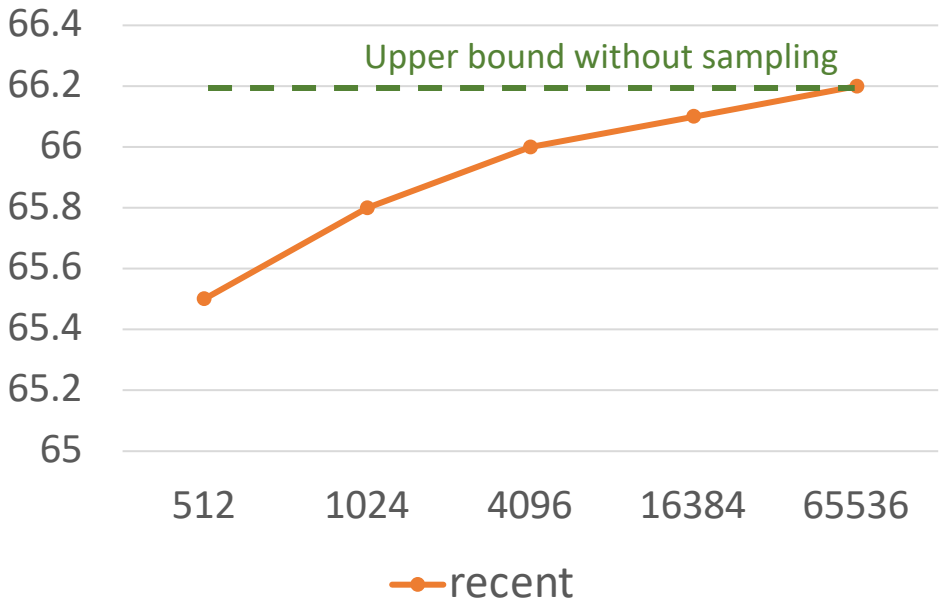
Weight Matrix of Classifier is too Large



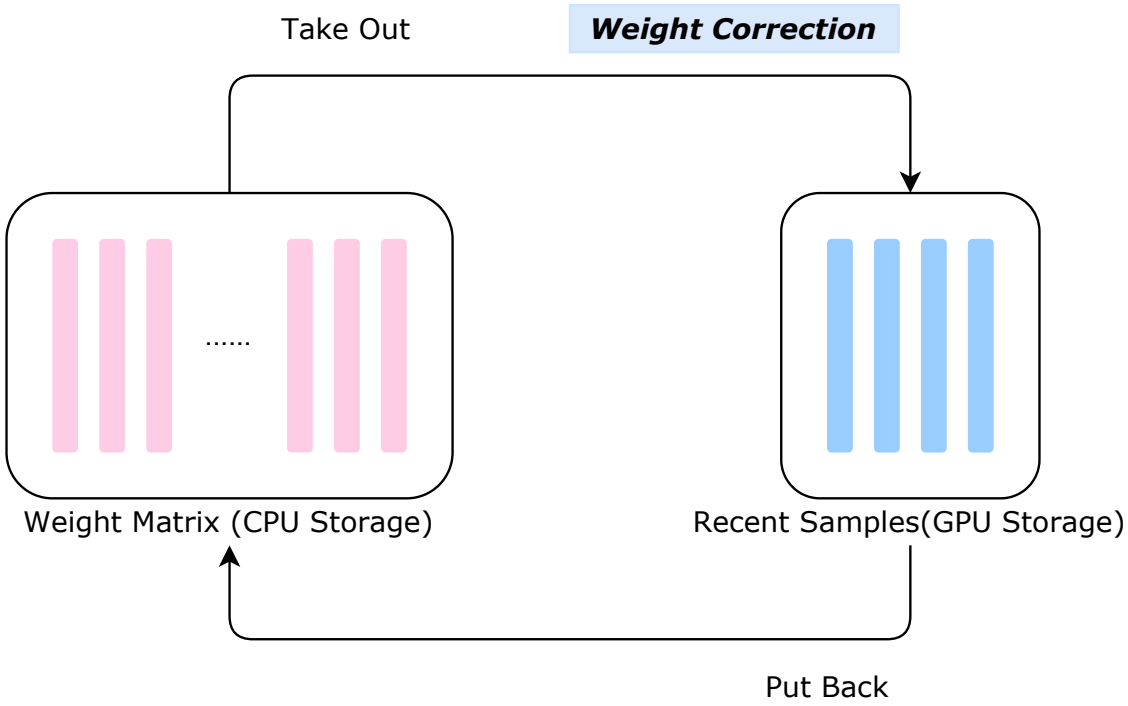
- Need to compute the similarity with the features and **all the instance vectors**
  - (batch_size × 128) × (128 × 1.28M)
- How to sample the negative instance vectors for each iteration?
  - Goal: # negative instance vectors is small & performance almost does not drop
  - e.g. (batch_size × 128) × (128 × 4096)

Results on Sampling Strategy

- Sampling Strategy
  - Sample the instance vectors in recent iterations (abbv. recent)



Weight Correction for Instance Classifier



SGD optimizer:

$$\mathbf{u}_i^{(t+1)} := m\mathbf{u}_i^{(t)} + (\mathbf{g}_i^{(t)} + \lambda \mathbf{w}_i^{(t)})$$

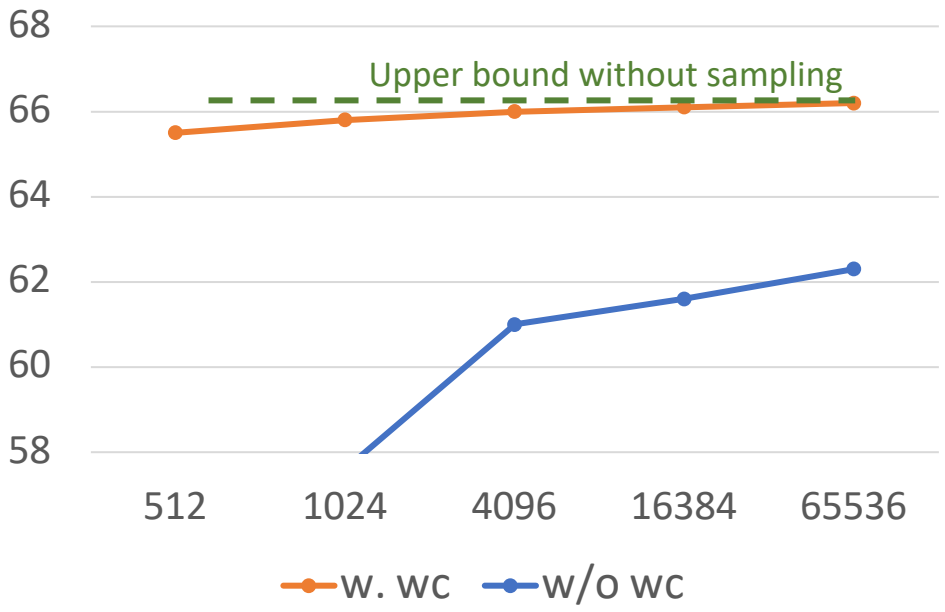
$$\mathbf{w}_i^{(t+1)} := \mathbf{w}_i^{(t)} - \eta \mathbf{u}_i^{(t+1)}$$

Weight Correction:

$$\begin{pmatrix} \mathbf{w}_i^{(t+t')} \\ \mathbf{u}_i^{(t+t')} \end{pmatrix} := \begin{pmatrix} 1-\eta\cdot\lambda & -\eta\cdot m \\ \lambda & m \end{pmatrix}^{t'} \begin{pmatrix} \mathbf{w}_i^{(t)} \\ \mathbf{u}_i^{(t)} \end{pmatrix}$$

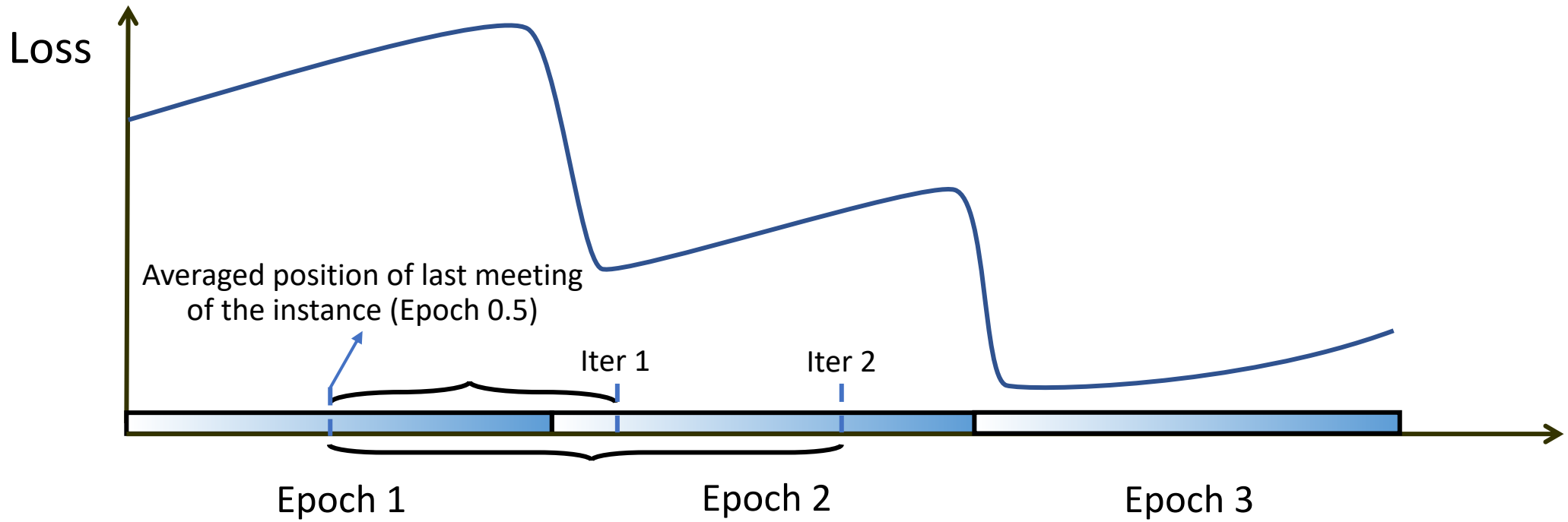
Results on Weight Correction

- Weight correction
  - With weight correction (abbv. w. wc)
  - Without weight correction (abbv. w/o wc)



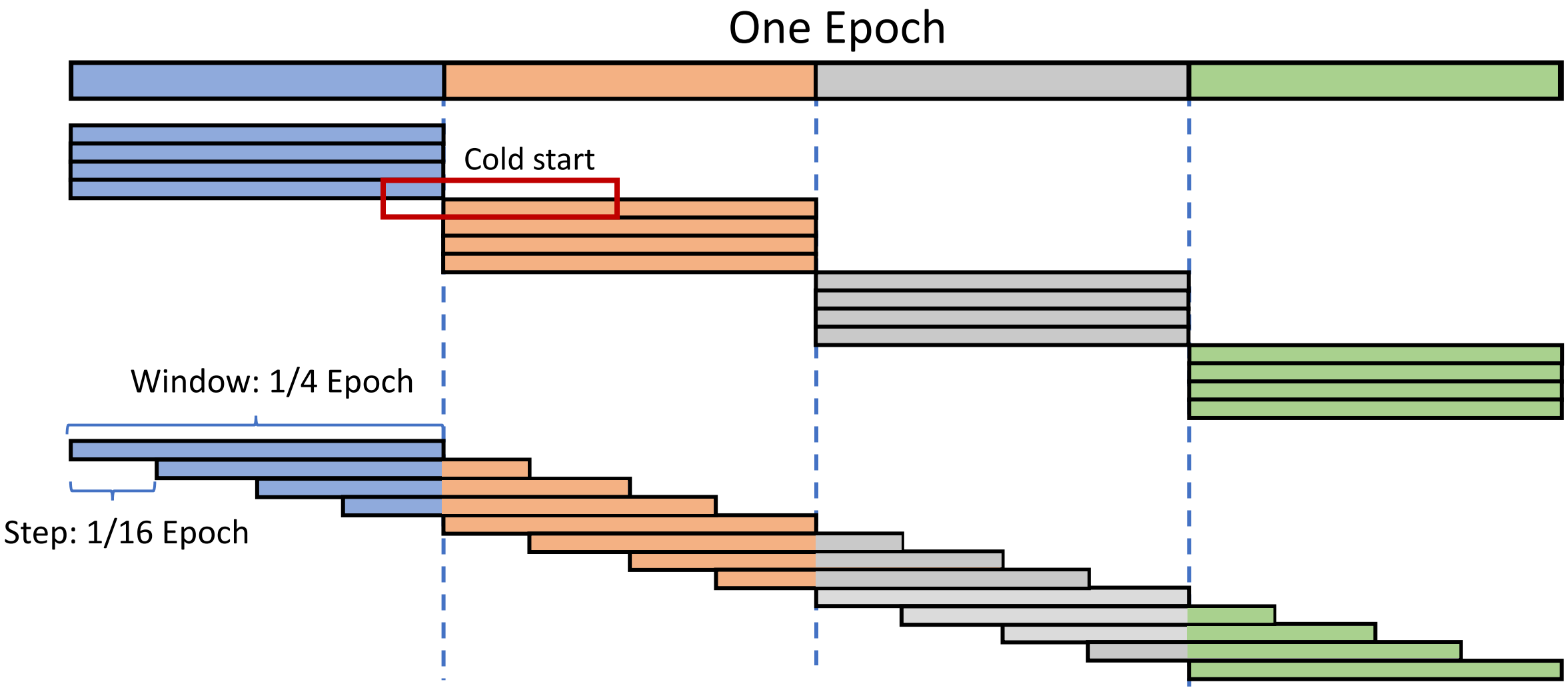
Weight Matrix of Classifier is Updated too Slowly

- Observation: loss goes up inside each epoch



- **Example forgetting**: Further away from the last meeting of this instance, easier to be forgotten, the greater the loss

Solution: Decrease # examples per epoch



Results - Linear Evaluation on ImageNet

Method	Setting	Top-1 Acc	Top-5 Acc
SimCLR		64.7	86.0
MoCo v2		64.1	85.7
PIC (ours)		66.2	87.0
PIC (ours)	+ sliding window	**67.3** +2.6	**87.6**
SimCLR		66.6	87.3
MoCo v2		67.5	88.0
PIC (ours)		68.5	88.7
PIC (ours)	+ sliding window	**69.0** +1.5	**88.8**

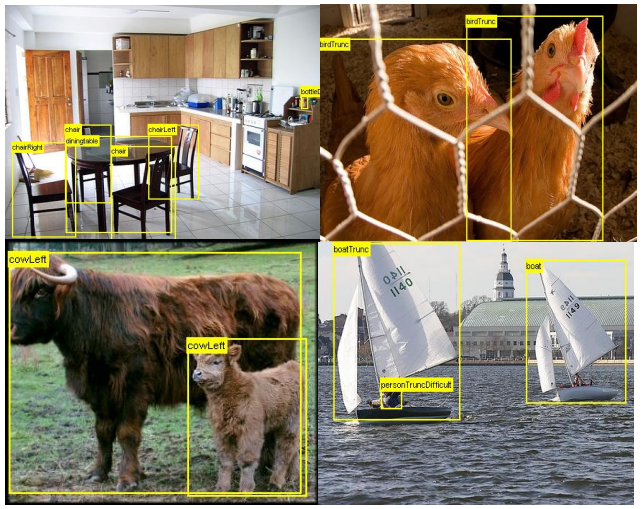
Results – System-Level Comparison on ImageNet

Method	Conference	Top-1 Acc	Top-5 Acc
PIRL	CVPR 2020	63.2	-
CPC v2	ICML 2020	63.8	85.3
CMC	ECCV 2020	64.1	-
SimCLR	ICML 2020	69.3	89.0
MoCo v2	Tech Report	**71.1**	-
PIC (ours)	NeurIPS 2020	**70.8**	**90.0**

Performance on Downstream Tasks

Fine-grained Recognition on iNaturalist

Method	Top-1 acc	Top-5 acc
Supervised	66.0	85.6
MoCo	65.7	85.7
PIC (ours)	**66.0**	**85.7**



Object Detection on Pascal VOC

Method	AP50	AP	AP75
Supervised	81.3	53.5	58.8
MoCo	81.5	55.9	62.6
PIC (ours)	**82.4**	**57.1**	**63.4**

Analysis

Relations with supervised classification

- Saliency maps generated by the supervised pre-trained model and unsupervised pre-trained model (PIC)

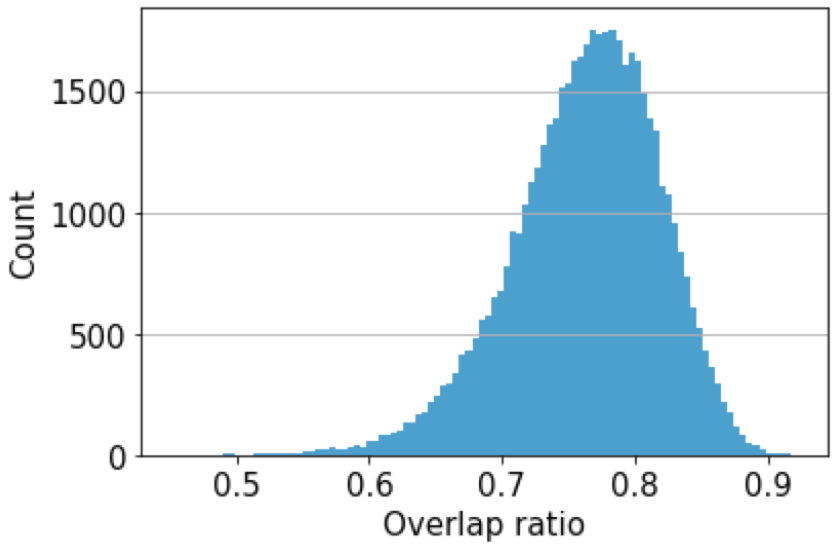
Raw Images

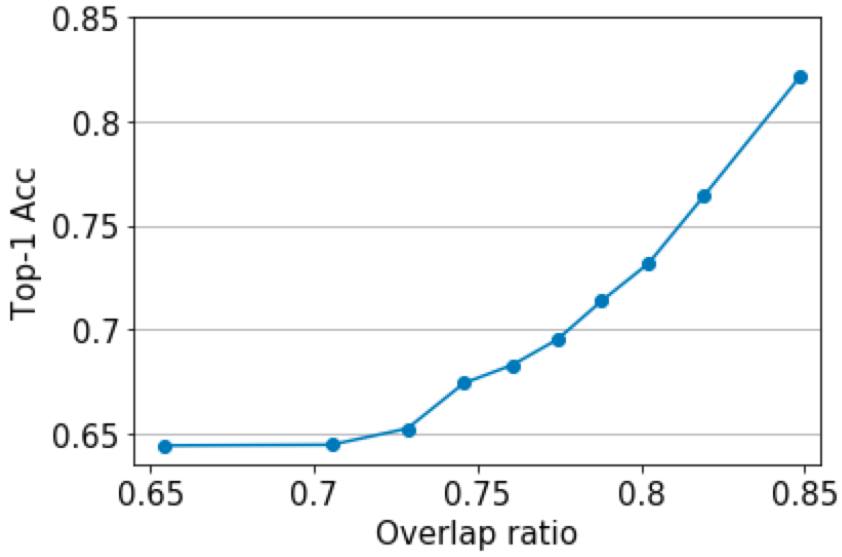
PIC

Supervised Model

Relations with supervised classification

- Statistical analysis
  - compute the **overlap over saliency maps** of supervised model and PIC
  - study the correlations between the overlap and the accuracy





Failure case

Q&A