CompReSS: Self-Supervised Learning by Compressing Representations

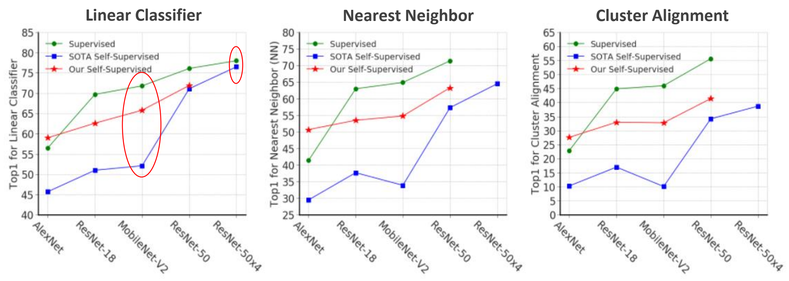
Soroush Abbasi Koohpayegani* Ajinkya Tejankar* Hamed Pirsiavash
University of Maryland Baltimore County
* Equal contribution

PyTorch Code

NEURAL INFORMATION PROCESSING SYSTEMS

Motivation

- Learning rich representations from unlabeled images
- Reducing the gap between supervised and self-supervised learning (SSL) for smaller models (e.g., MobileNet)
- Compressing a rich SSL model to a smaller one to enable learning at the edge for preserving privacy



Key Ideas

- Train a high-capacity model using an off-the-shelf self-supervised method and compress it to a small model
- Each data point should have the same relationship with its neighbors in both teacher and student embeddings.

Method

- Capture the similarity of each data point (query) to the other training data (anchors) in the teacher's embedding space f :

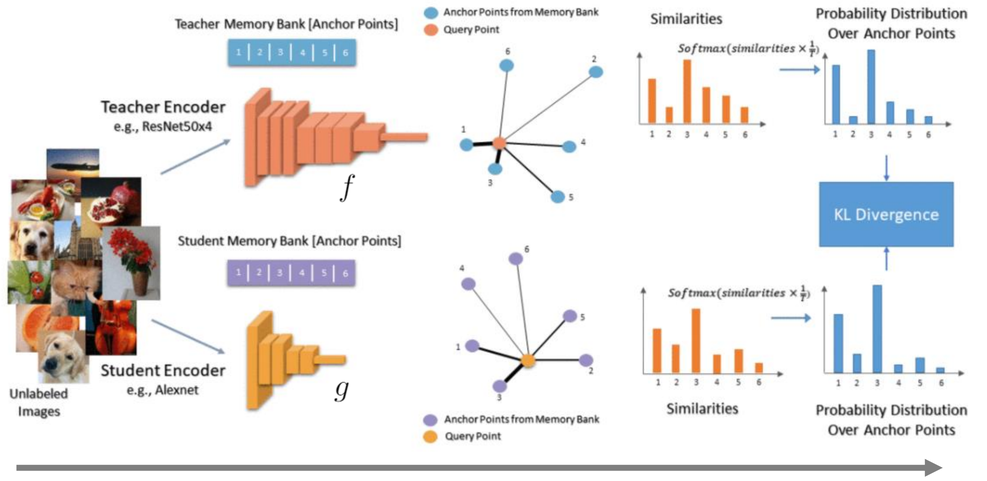
Query point, Anchor point

$$p_i(q,a,f) = \frac{exp(f(q)^\intercal f(a_i)/\tau)}{\sum_j exp(f(q)^\intercal f(a_j)/\tau)}$$

- Do the same on the student's embedding space g .
- Define the loss for a particular query point as the KL divergence between the probabilities over all anchor points under the teacher and student models:

$$L = KL\big(p(q,a,f) \;||\; p(q,a,g)\big)$$

- We optimize the student by minimizing the summation of L over all images



Compare each query image with a random set of anchor points.

Convert the similarities to a probability distribution over anchor points.

Get the same distribution for the student as well.

Train the student to minimize the KL divergence between the two distributions.

- **Ours-2q** : Use a separate memory bank for teacher and student to decouple the embeddings.
- **Ours-1q** : Use the teacher's anchor points in calculating the similarity for the student model.

Results

- Our method is better than other compression methods by a large margin across 3 different evaluation benchmarks and 2 different teacher SSL methods.
- For the first time, a self-supervised AlexNet outperforms supervised one on ImageNet classification
- We reduce the gap between supervised and SSL in smaller models
- A linear classifier on our embeddings outperforms finetuning SSL methods on small ImageNet

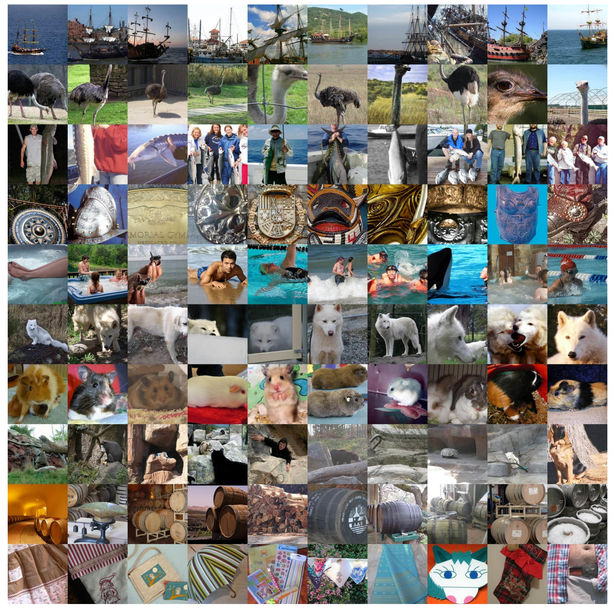
AlexNet

Method	ImageNet top-1	Places top-1	PASCAL VOC Cls. (mAP)	PASCAL VOC Det. (mAP)
Sup. on ImageNet	56.5 (f7)	39.4 (c4)	79.9 (all)	59.1 (all)
Context	31.7 (c4)	32.7 (c4)	65.3 (all)	51.1 (all)
Jigsaw	34.0 (c3)	35.0 (c3)	67.6 (all)	53.2 (all)
Counting	34.3 (c3)	36.3 (c3)	67.7 (all)	51.4 (all)
RotNet	38.7 (c3)	35.1 (c3)	73.0 (all)	54.4 (all)
DeepCluster	39.8 (c4)	37.5 (c4)	73.7 (all)	55.4 (all)
RFDecouple	44.3 (c5)	38.6 (c5)	74.7 (all)	58.0 (all)
SeLa	44.7 (c5)	37.9 (c4)	77.2 (all)	59.2 (all)
MoCo	45.7 (f7)	36.6 (c4)	71.3 (f8)	55.8 (all)
Ours-2q	57.6 (f7)	**40.4** (c5)	**79.7** (f8)	58.1 (all)
Ours-1q	**59.0** (f7)	40.3 (c5)	76.2 (f8)	**59.3** (all)

ResNet-50

Method	ImageNet top-1
Sup. on ImageNet	76.2 (L5)
SimCLR	69.3 (L5)
MoCo	71.1 (L5)
InfoMin	73.0 (L5)
BYOL	74.3 (L5)
SwAV	**75.3** (L5)
Ours-1q	71.9 (L5)

Compressed from SimCLR ResNet-50 (x4)



- K-means on our AlexNet embeddings (k = 1000)
- Each row is a cluster
- No cherry-picking: random images from random clusters

Comparison with other compression methods

Teacher / Student	MoCo ResNet-50 / ResNet-18 linear	NN	CA	SwAV ResNet-50 / ResNet-18 linear	NN	CA
Teacher	70.8	57.3	34.2	75.6	60.7	27.6
Supervised	69.8	63.0	44.9	69.8	63.0	44.9
CC	61.1	51.1	25.2	60.8	51.0	22.8
CRD	58.4	43.7	17.4	58.2	44.7	16.9
Reg-BN	58.2	47.3	27.2	60.6	47.6	20.8
Ours-1q	**62.6**	**53.5**	**33.0**	**65.6**	**56.0**	**26.3**

Comparison with other SSL methods on small labelled ImageNet for ResNet-50

Method	Top-1 1%	Top-1 10%	Top-5 1%	Top-5 10%
Supervised	25.4	56.4	48.4	80.4
All layers are fine-tuned.				
InstDisc	-	-	39.2	77.4
PIRL	-	-	57.2	83.8
SimCLR	48.3	65.6	75.5	87.8
BYOL	53.2	68.8	78.4	89.0
SwAV	53.9	**70.2**	78.5	**89.9**
Only the linear layer is trained.				
Ours-1q	**59.7**	67.0	**82.3**	87.5