Self-Supervised Representation Learning

Nikita Konodyuk

Higher School of Economics

Self-supervised learning

- Supervised learning is great

Self-supervised learning

- Supervised learning is great if you have enough labels

Self-supervised learning

- Supervised learning is great if you have enough labels
- Amount of unlabeled data is much greater than of labeled

Self-supervised learning

- Supervised learning is great if you have enough labels
- Amount of unlabeled data is much greater than of labeled
- What if we somehow generate labels from unlabeled data?

Self-supervised learning

- Supervised learning is great if you have enough labels
- Amount of unlabeled data is much greater than of labeled
- What if we somehow generate labels from unlabeled data?
- Exactly what BERT does

Self-supervised representation learning

- Learning informative image representations is crucial for transfer learning

Self-supervised representation learning

- Learning informative image representations is crucial for transfer learning
- Can obtain them only by training a supervised classifier

Self-supervised representation learning

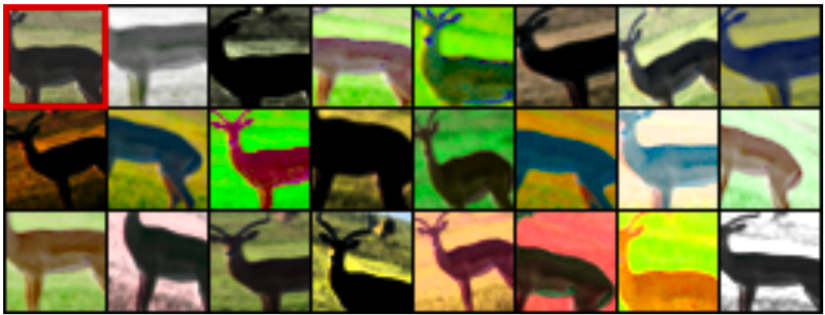
- Learning informative image representations is crucial for transfer learning
- Can obtain them only by training a supervised classifier
- Let’s just convert an arbitrary set of images into a classification dataset

Self-supervised tasks: Surrogate classes[1]

- Sample N patches of size 32×32 from images
- Augment each patch multiple times to get a set of images originating from the same patch and call this set a surrogate class
- Train a neural net to solve N-class classification task

[1] Dosovitskiy et al. (2015)

Self-supervised tasks: Surrogate classes

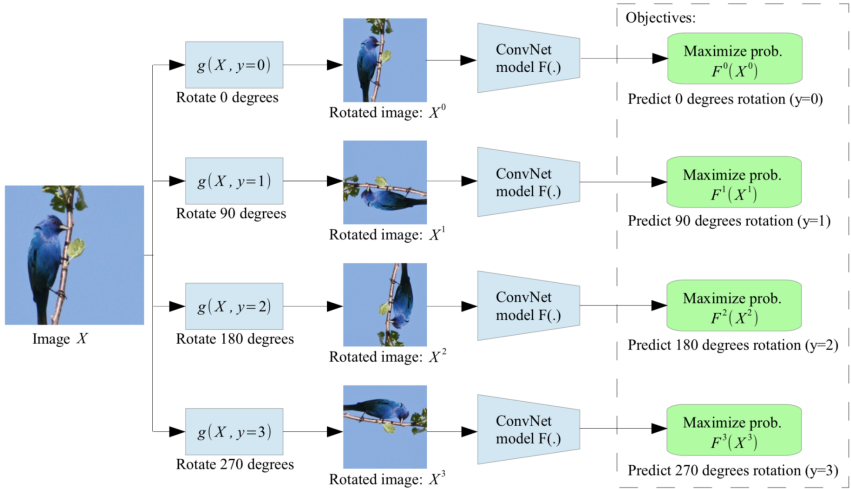


Self-supervised tasks: Rotation prediction[2]

- Randomly rotate an image by a multiple of 90°
- Train a neural net to predict the rotation angle

[2]Gidaris et al. (2018)

Self-supervised tasks: Rotation prediction

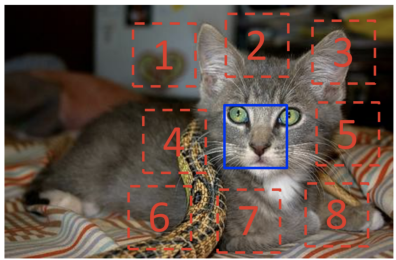


Self-supervised tasks: Relative position prediction[3]

- Sample a random patch from an image
- Sample the second patch from a 3×3 grid around the first patch
- Train a network to predict the relative position of the second patch, i.e. solve 8-class classification task

[3] Doersch et al. (2015)

Self-supervised tasks: Relative position prediction



X = (,); Y = 3

Example:

Question 1:

?

Question 2:

?

SimCLR: A simple framework for contrastive learning of visual representations[4]

- Sample a batch of images

[4] Chen et al. (2020)

SimCLR: A simple framework for contrastive learning of visual representations[4]

- Sample a batch of images
- Augment each image twice (augmentations are sampled at random)

[4] Chen et al. (2020)

SimCLR: A simple framework for contrastive learning of visual representations[4]

- Sample a batch of images
- Augment each image twice (augmentations are sampled at random)
- Get representations of each augmented image

[4] Chen et al. (2020)

SimCLR: A simple framework for contrastive learning of visual representations[4]

- Sample a batch of images
- Augment each image twice (augmentations are sampled at random)
- Get representations of each augmented image
- Get projections from representations (one hidden non-linear layer)

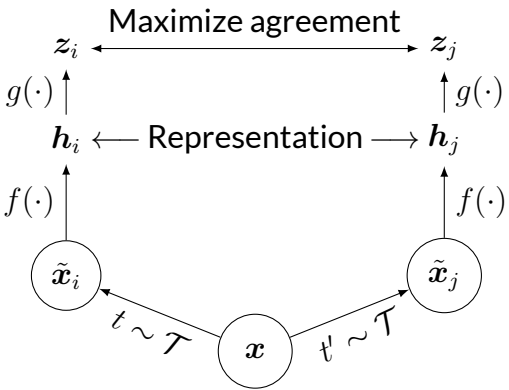
[4] Chen et al. (2020)

SimCLR: A simple framework for contrastive learning of visual representations[4]

- Sample a batch of images
- Augment each image twice (augmentations are sampled at random)
- Get representations of each augmented image
- Get projections from representations (one hidden non-linear layer)
- Maximize agreement between the projections of the augmented versions of the same image

[4] Chen et al. (2020)

SimCLR



- $\boldsymbol{h}_i = f(\tilde{\boldsymbol{x}}_i) = \mathrm{ResNet}(\tilde{\boldsymbol{x}}_i)$
- $\boldsymbol{z}_i = g(\boldsymbol{h}_i) = W^{(2)}\sigma(W^{(1)}\boldsymbol{h}_i)$

SimCLR: Contrastive loss

For a positive pair (i, j), the loss function is defined as

$$\ell(i,j) = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$$

where

$$\text{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / (\|\boldsymbol{u}\| \|\boldsymbol{v}\|)$$

SimCLR: Algorithm overview

for sampled minibatch $\{\boldsymbol{x}_k\}_{k=1}^{N}$ **do**
  for $k \in \{1, \ldots, N\}$ **do**
    $t, t' \leftarrow \text{sample}(\mathcal{T})$
    $\boldsymbol{z}_{2k-1} \leftarrow g(f(t(\boldsymbol{x}_k))),\ \boldsymbol{z}_{2k} \leftarrow g(f(t'(\boldsymbol{x}_k)))$
  end for
  $\ell(i, j) = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$
  $\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
  update f and g to minimize $\mathcal{L}$
end for

SimCLR: Augmentations

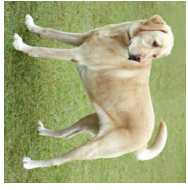
(a) Original

(b) Crop and resize

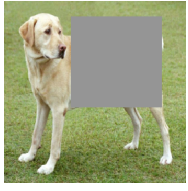
(c) Crop, resize (and flip)

(d) Color distort (drop)

(e) Color distort (jitter)



(f) Rotate {90°,180°,270°}



(g) Cutout

(h) Gaussian noise

(i) Gaussian blur

(j) Sobel filtering

Evaluation protocol

1. Train self-supervised representations on ImageNet

Evaluation protocol

1. Train self-supervised representations on ImageNet
2. Freeze the representation network

Evaluation protocol

1. Train self-supervised representations on ImageNet
2. Freeze the representation network
3. Train a linear classifier on top of the representations

Evaluation protocol

1. Train self-supervised representations on ImageNet
2. Freeze the representation network
3. Train a linear classifier on top of the representations
4. Evaluate its accuracy on a test dataset

SimCLR: Benchmarks

Method	Architecture	Param (M)	Top 1	Top 5
Methods using ResNet-50:				
Local Agg.	ResNet-50	24	60.2	-
MoCo	ResNet-50	24	60.6	-
PIRL	ResNet-50	24	63.6	-
CPC v2	ResNet-50	24	63.8	85.3
SimCLR (ours)	ResNet-50	24	**69.3**	**89.0**

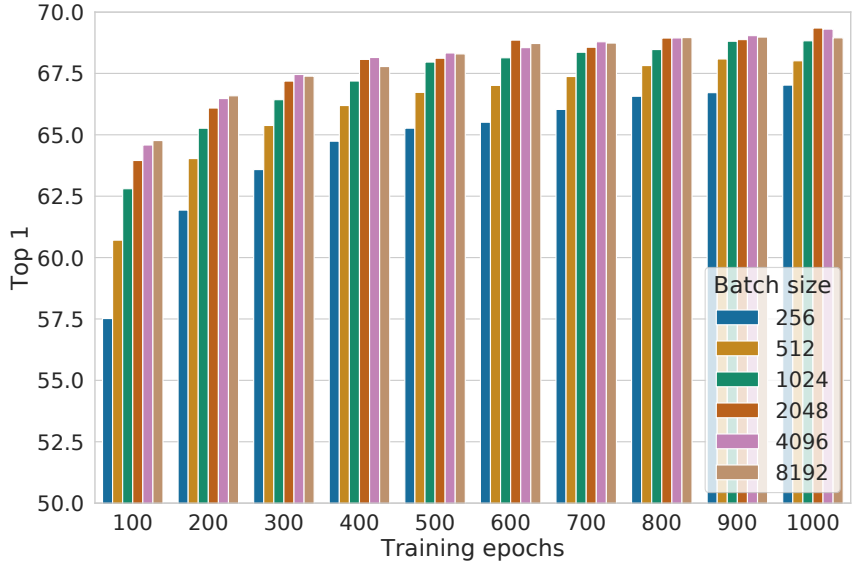
SimCLR: Benchmarks

Method	Architecture	Param (M)	Top 1	Top 5
Methods using other architectures:				
Rotation	RevNet-50 ($4\times$)	86	55.4	-
BigBiGAN	RevNet-50 ($4\times$)	86	61.3	81.9
AMDIM	Custom-ResNet	626	68.1	-
CMC	ResNet-50 ($2\times$)	188	68.4	88.2
MoCo	ResNet-50 ($4\times$)	375	68.6	-
CPC v2	ResNet-161 ($*$)	305	71.5	90.1
SimCLR (ours)	ResNet-50 ($2\times$)	94	74.2	92.0
SimCLR (ours)	ResNet-50 ($4\times$)	375	**76.5**	**93.2**

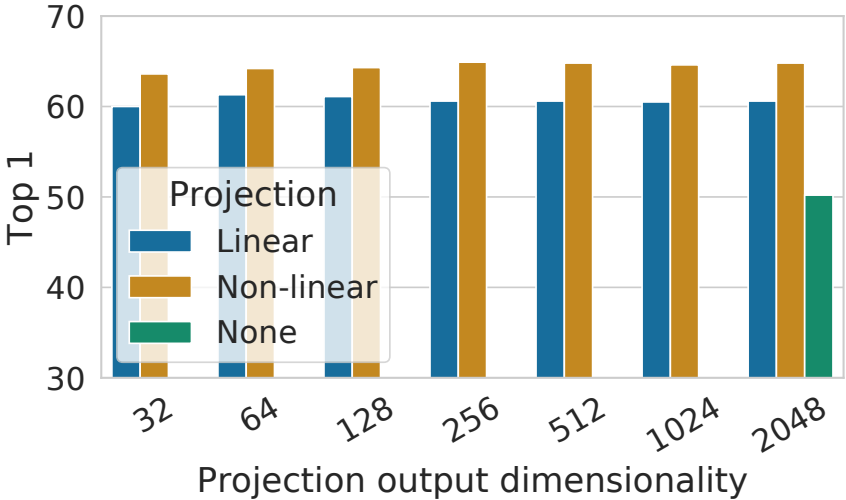
SimCLR: Benchmarks

Method	Architecture	Label fraction (Top 5) 1%	10%
Supervised baseline	ResNet-50	48.4	80.4
Methods using representation learning only:			
InstDisc	ResNet-50	39.2	77.4
BigBiGAN	RevNet-50 ($4\times$)	55.2	78.8
PIRL	ResNet-50	57.2	83.8
CPC v2	ResNet-161($*$)	77.9	91.2
SimCLR (ours)	ResNet-50	75.5	87.8
SimCLR (ours)	ResNet-50 ($2\times$)	83.0	91.2
SimCLR (ours)	ResNet-50 ($4\times$)	**85.8**	**92.6**

SimCLR: Batch size



SimCLR: Projection dimensionality



SimCLRv2[5]: Current SOTA

- Wider and deeper network (+29% in top-1 accuracy): ResNet-50 $\rightarrow$ ResNet-152 (3$\times$+SK)

[5] Chen et al. (2020)

SimCLRv2[5]: Current SOTA

- Wider and deeper network (+29% in top-1 accuracy): ResNet-50 $\rightarrow$ ResNet-152 (3$\times$+SK)
- Deeper projection head (+14% in top-1 accuracy): 2-layer $\rightarrow$ 3-layer, fine-tuning from the 1st layer

[5] Chen et al. (2020)

SimCLRv2[5]: Current SOTA

- Wider and deeper network (+29% in top-1 accuracy): ResNet-50 $\rightarrow$ ResNet-152 (3$\times$+SK)
- Deeper projection head (+14% in top-1 accuracy): 2-layer $\rightarrow$ 3-layer, fine-tuning from the 1st layer
- Memory bank from MoCo (+$\approx$1%)

[5] Chen et al. (2020)

SimCLRv2: Benchmarks

Method	Architecture	Top-1 Label fraction		Top-5 Label fraction	
		1%	10%	1%	10%
Supervised baseline	ResNet-50	25.4	56.4	48.4	80.4
Methods using unlabeled data in a task-agnostic way:					
SimCLR	ResNet-50	48.3	65.6	75.5	87.8
SimCLR	ResNet-50 ($2\times$)	58.5	71.7	83.0	91.2
SimCLR	ResNet-50 ($4\times$)	63.0	74.4	85.8	92.6
BYOL (concurrent work)	ResNet-50	53.2	68.8	78.4	89.0
BYOL (concurrent work)	ResNet-200 ($2\times$)	71.2	77.7	89.5	93.7
Methods using unlabeled data in both ways:					
SimCLRv2 distilled (ours)	ResNet-50	73.9	77.5	91.5	93.4
SimCLRv2 distilled (ours)	ResNet-50 ($2\times$+SK)	75.9	80.2	93.0	95.0
SimCLRv2 self-distilled (ours)	ResNet-152 ($3\times$+SK)	**76.6**	**80.9**	**93.4**	**95.5**

Further reading

- Lilian Weng's blospost on self-supervised representation learning
- SimCLR paper
- SimCLRv2 paper