DMQA Seminar 20200219

Dive into BYOL

Bootstrap Your Own Latent

일반대학원 산업경영공학과
김재훈

Introduction

발표자 소개

- 이름: 김재훈
- 학력
  - ✓ 2013.03 – 2019.02 | 학사 | 동국대학교 경영학과
  - ✓ 2020.03 – 현재 | 석박사통합과정 | 고려대학교 산업경영공학과 (지도교수: 김성범)
- 연구분야
  - ✓ Self-supervised learning
  - ✓ Reinforcement learning
- e-mail : jhoon0418@korea.ac.kr

CONTENTs

- Representation learning
- Self-supervised learning
- Bootstrap Your Own Latent
- Understanding self-supervised and contrastive learning with “Bootstrap Your Own Latent”
- BYOL doesn’t even need batch statistics

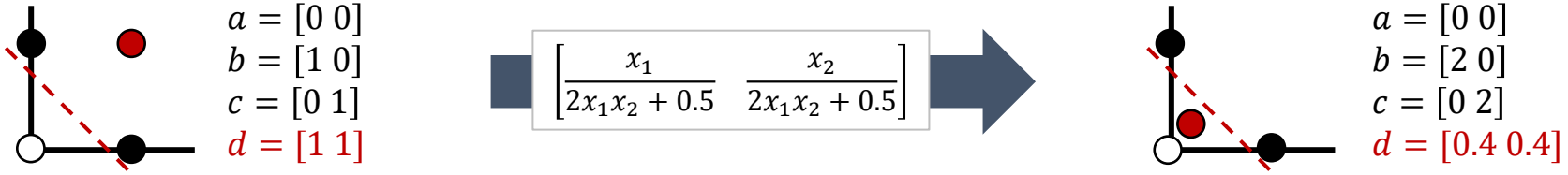
Representation Learning

Dive into BYOL

❖ Representation Learning이란?

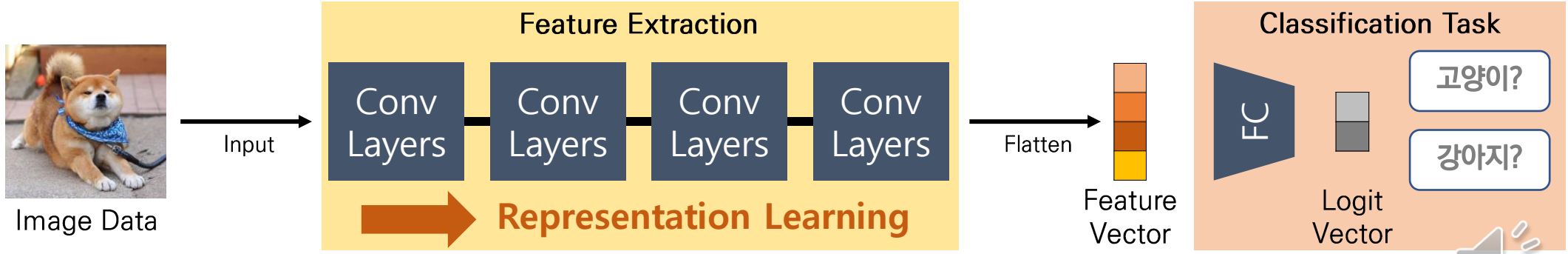
- 최종적으로 수행하려는 작업을 잘 하기 위해서 모델이 입력 데이터의 특징을 잘 표현할 수 있도록 훈련하는 것

✓ Feature space의 변화
: 최종적으로 수행하려는 작업을 잘 하기 위해서 입력 데이터의 특징을 변환함



→ 고차원의 데이터 분포에서는 보다 깊은 신경망으로 구성된 Representation Learning 알고리즘을 사용함

✓ Representation Learning 예시



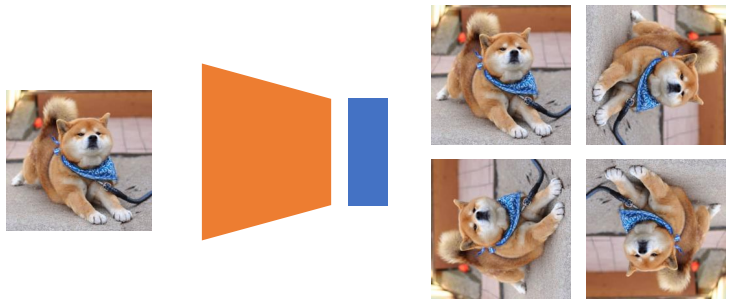
Self-supervised Learning

Dive into BYOL

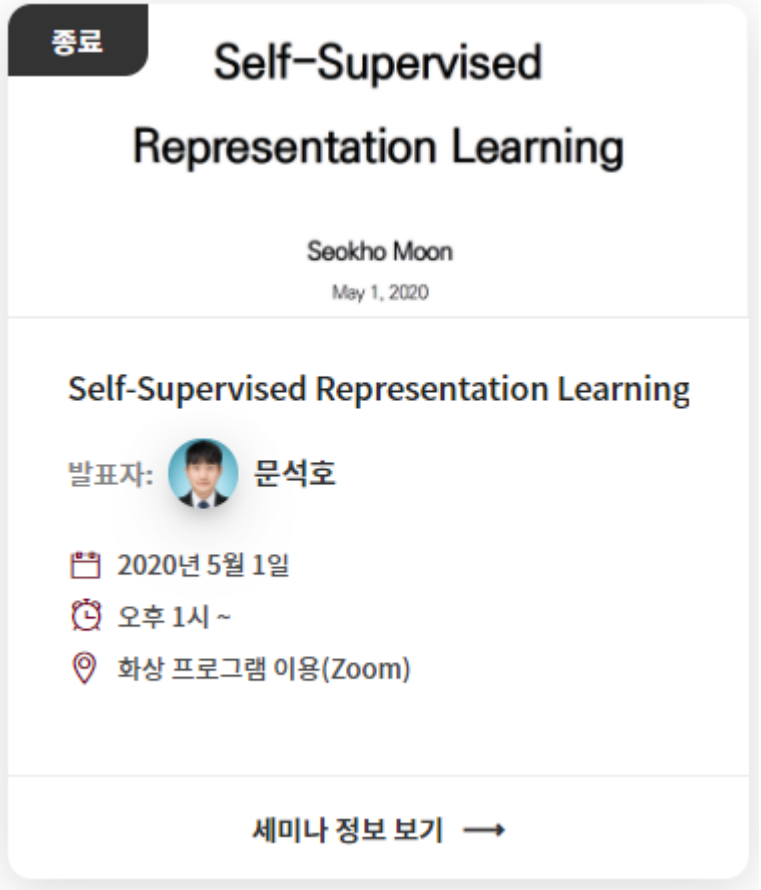
❖ Self-supervised Learning이란?

- 레이블이 없는 데이터를 활용하여 모델이 좋은 특징을 추출할 수 있도록 학습하는 방법론

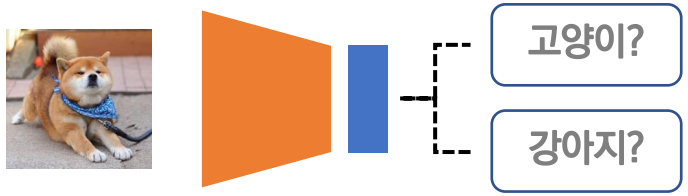
✓ Step 1) Self-supervised 방식으로 모델을 사전 학습함
: Self-supervised learning은 크게 pretext task와 contrastive learning 방식으로 나뉠 수 있음



Pretext Task
(ex. Rotation)



✓ Step 2) 충분히 사전 학습한 모델에 대하여 downstream task로 전이학습을 진행

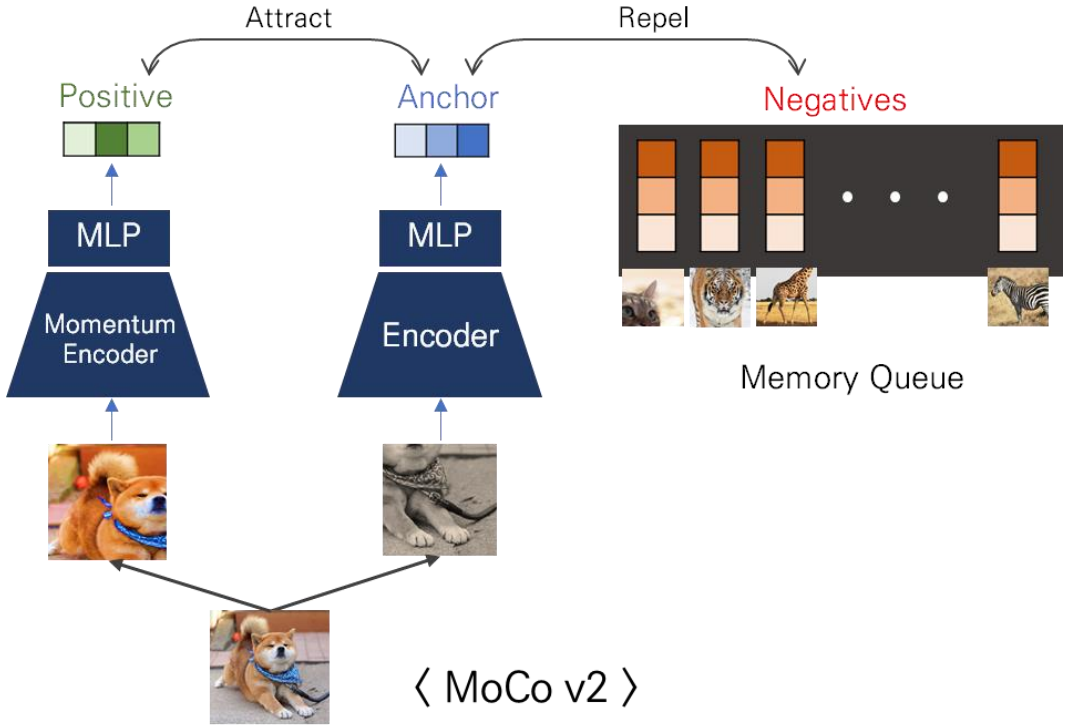


Self-supervised Learning

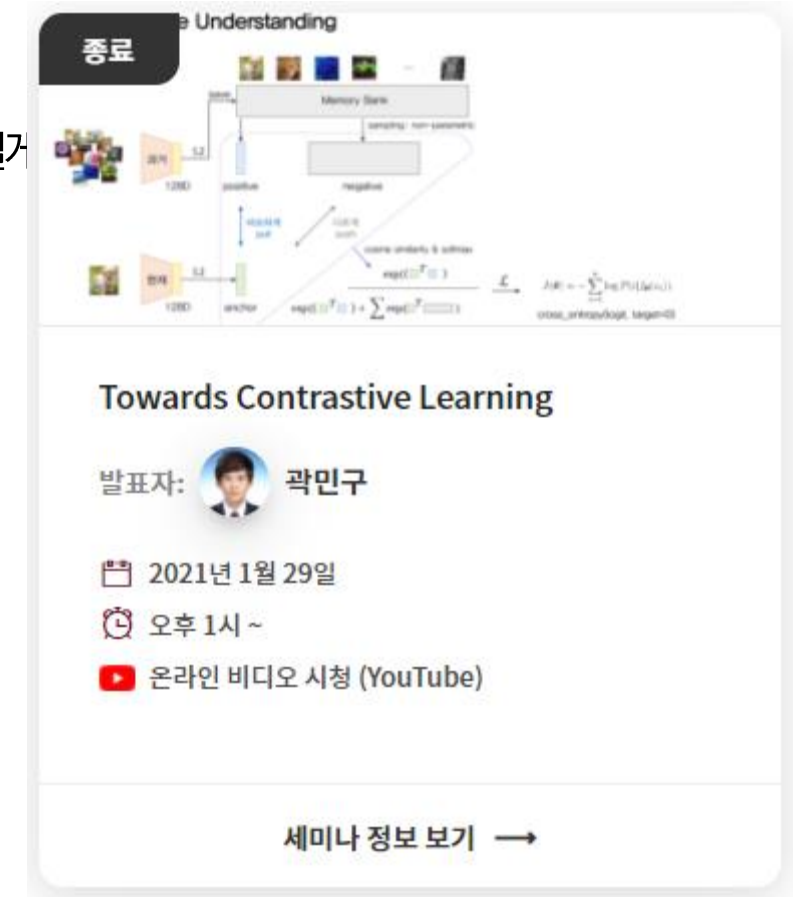
Dive into BYOL

❖ Contrastive Learning 이란?

- Representation space 상에서 자신과 유사한 이미지는 가깝게 다른 이미지와는 멀게
- Anchor, positive, negative에 해당하는 representation vector를 생성하여 학습



〈 MoCo v2 〉



He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9729-9738).

Bootstrap Your Own Latent

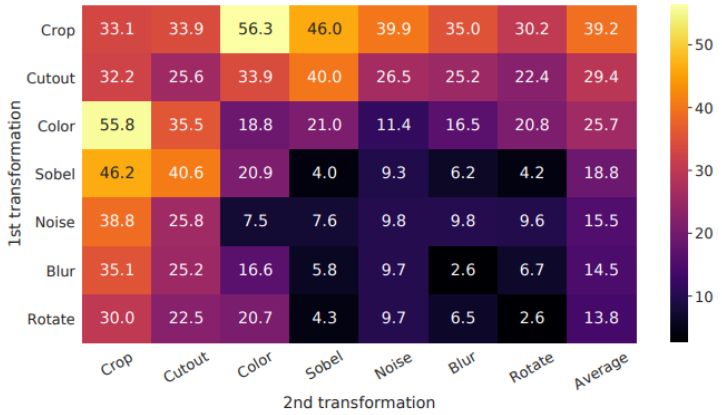
Dive into BYOL

❖ Contrastive learning의 단점

- Require careful treatment of negative pairs
  - ✓ Negative pair를 제공하는 전략을 고려할 필요가 있음

    ex) MoCo → Memory Queue
    SimCLR → Batch size

- Choice of Image Augmentation
  - ✓ Augmentation 조합에 따라서 모델의 성능이 크게 좌우되는 것을 확인할 수 있음



- Requires comparing each representation of an augmented view with many negative examples

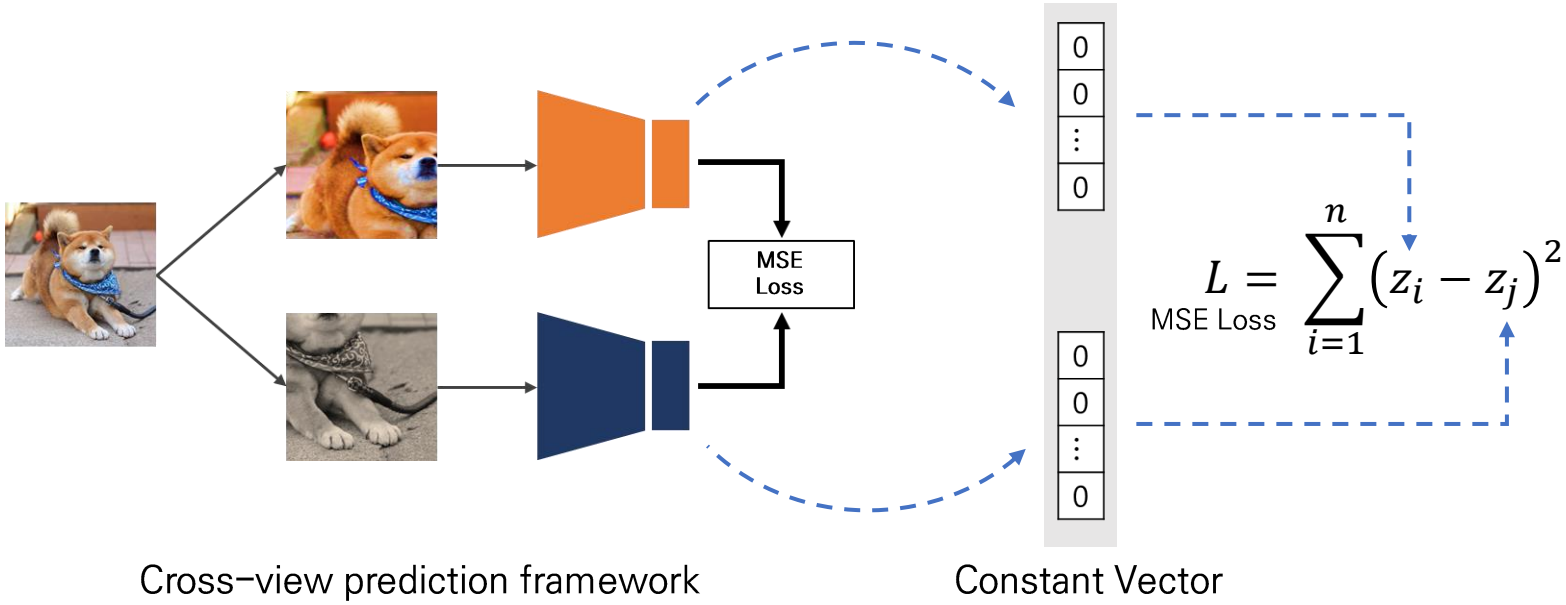
Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597-1607). PMLR.

Bootstrap Your Own Latent

Dive into BYOL

❖ Collapsed representation

- Positive pairs로만 학습을 하는 경우 모델이 constant vector를 출력하는 문제
  - ✓ Contrastive Loss가 나온 배경이자 Negative pairs를 학습에 더하는 것이 필요한 이유
  - ✓ Train loss는 작아지지만 학습은 전혀 안 되는 문제 발생



Becker, S., & Hinton, G. E. (1992). Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, *355*(6356), 161-163.

Bootstrap Your Own Latent

Dive into BYOL

❖ Collapsed representation

- Contrastive loss는 positive와 negative sample을 모두 사용하여 collapse를 방지함
  - ✓ Positive pair 간의 유사도가 크고 negative pair 간의 유사도가 작을 수록 loss 값이 낮아짐

$$L_{i,j} = -\log \frac{\exp\left(\frac{sim(z_i, z_j)}{\tau}\right)}{\sum_{k=1\ [k\neq i]}^{N} \exp\left(\frac{sim(z_i, z_k)}{\tau}\right)}$$

Contrastive Loss

Cosine similarity (Positive pair)

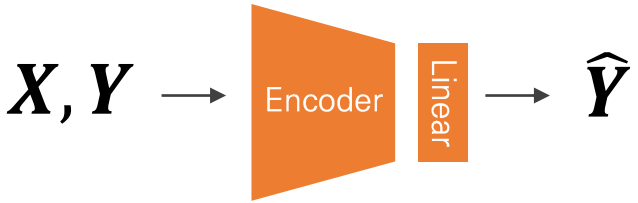
Cosine similarity (Negative pair)

Bootstrap Your Own Latent

Dive into BYOL

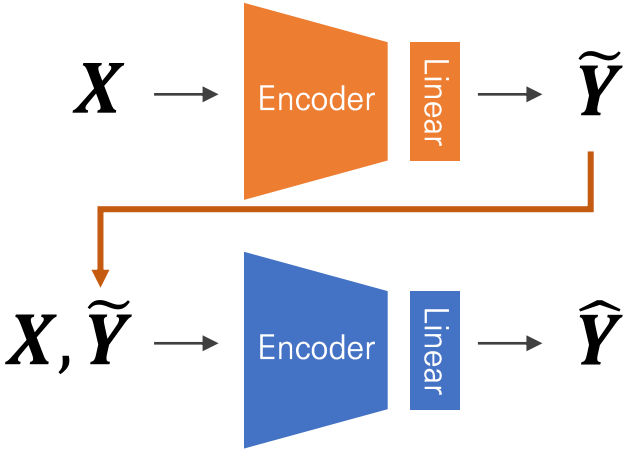
❖ Core motivation of BYOL

- Negative sample 없이도 representation collapse가 나오지 않는 연구 진행
  - ✓ Case 1) Encoder (random parameter / freeze) + Linear Layer



➡ Top 1 Acc. 1.4%

  - ✓ Case 2) Case 1과 동일한 구조의 네트워크를 생성하여 Case 1의 네트워크가 출력한 값을 예측하도록 학습



Encoder (random parameter / freeze) + Linear Layer (freeze)

➡ Top 1 Acc. 18.8%

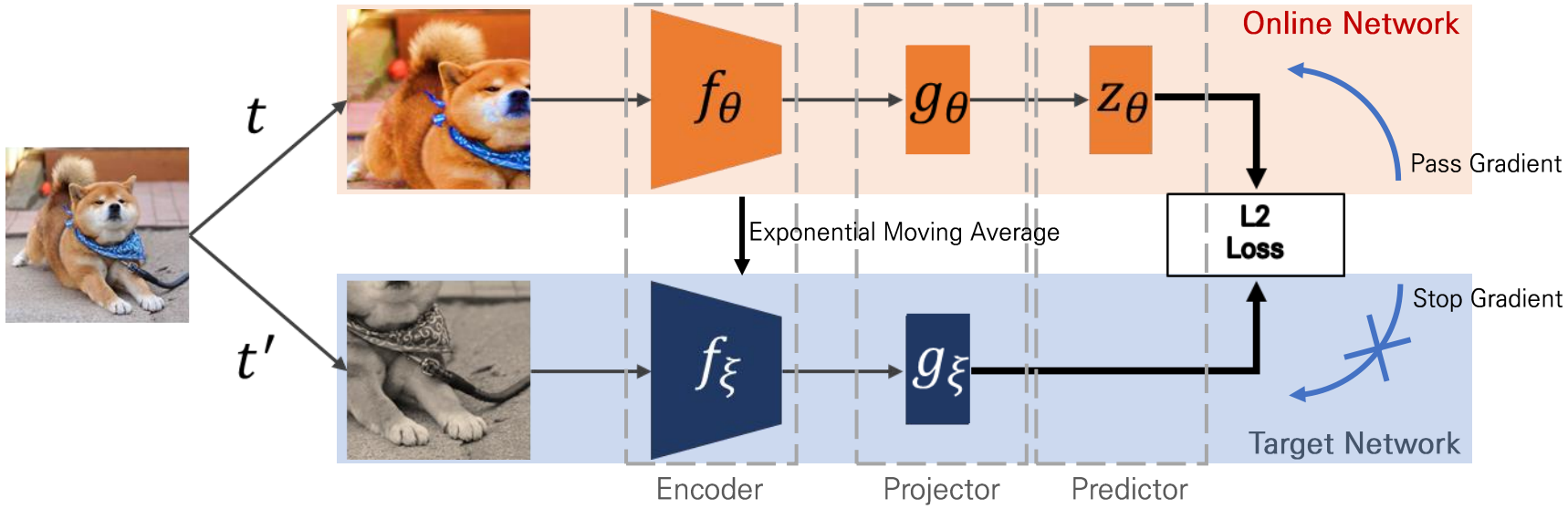
Encoder (random parameter / freeze) + Linear Layer

Bootstrap Your Own Latent

Dive into BYOL

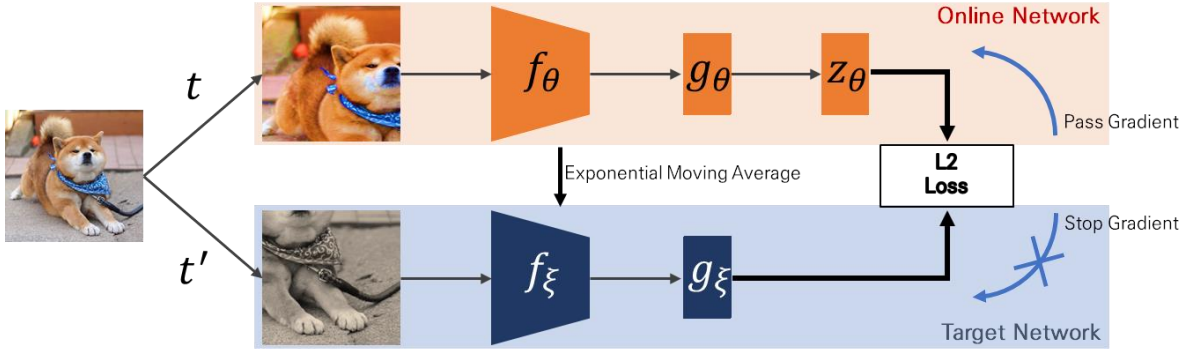
❖ Architecture of BYOL

- BYOL 구성요소
  - ✓ 파라미터를 업데이트하는 방식이 서로 다른 동일한 구조의 두 네트워크로 구성됨 (Online network / Target network)
  - ✓ 두 네트워크 모두 Encoder와 Projector를 보유하나 Predictor는 Online network만 보유함
  - ✓ Target network에서 출력한 representation vector를 Online network에서 예측하는 훈련을 진행



Bootstrap Your Own Latent

Dive into BYOL



❖ Target network update in BYOL

- Exponential moving average
  - ✓ MoCo v2에서 사용하는 momentum update 유사한 방식
  - ✓ Cosine annealing을 사용하여 학습이 진행될 수록 τ를 점점 1에 가까운 값으로 키움

Target Network (Old) Online Network

$$\xi \leftarrow \tau\xi + (1-\tau)\theta$$

Target Network (New)

$$\tau_{base} = 0.996$$

$$\tau \triangleq 1 - (1-\tau_{base}) \cdot \frac{\left(\cos\left(\frac{\pi k}{K}\right)+1\right)}{2}$$

Bootstrap Your Own Latent

Dive into BYOL

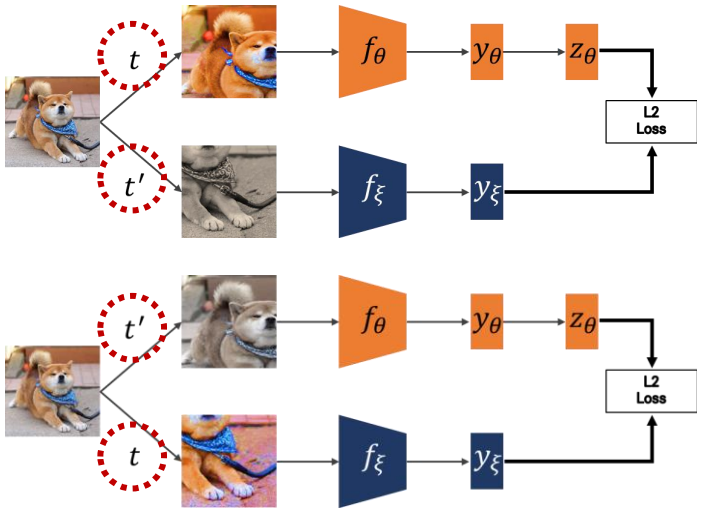
❖ Loss function in BYOL

- L2 loss
  - ✓ 각 네트워크의 prediction과 projection에 L2 정규화를 취한 뒤 loss를 계산

$$\mathcal{L}_\theta^{BYOL} \triangleq ||\overline{q_\theta}(z_\theta) - \overline{z_\xi}'||_2^2 \quad \leftarrow \quad \overline{q_\theta}(z_\theta) \triangleq \frac{q_\theta(z_\theta)}{||q_\theta(z_\theta)||_2}, \quad \overline{z_\xi}' \triangleq \frac{z_\xi'}{||z_\xi'||_2}$$

- Loss function symmetrization
  - ✓ Augmentation 조합을 교환하여 loss를 한 번 더 계산

$$Total\ Loss = \mathcal{L}_\theta^{BYOL} + \tilde{\mathcal{L}}_\theta^{BYOL}$$



$\mathcal{L}_\theta^{BYOL}$

$\tilde{\mathcal{L}}_\theta^{BYOL}$

Bootstrap Your Own Latent

Dive into BYOL

❖ Experiment Results

- Linear evaluation on ImageNet

Method	Top-1	Top-5
Local Agg.	60.2	-
PIRL [35]	63.6	-
CPC v2 [32]	63.8	85.3
CMC [11]	66.2	87.0
SimCLR [8]	69.3	89.0
MoCo v2 [37]	71.1	-
InfoMin Aug. [12]	73.0	91.1
BYOL (ours)	**74.3**	**91.6**

(a) ResNet-50 encoder.

Method	Architecture	Param.	Top-1	Top-5
SimCLR [8]	ResNet-50 (2×)	94M	74.2	92.0
CMC [11]	ResNet-50 (2×)	94M	70.6	89.7
BYOL (ours)	ResNet-50 (2×)	94M	77.4	93.6
CPC v2 [32]	ResNet-161	305M	71.5	90.1
MoCo [9]	ResNet-50 (4×)	375M	68.6	-
SimCLR [8]	ResNet-50 (4×)	375M	76.5	93.2
BYOL (ours)	ResNet-50 (4×)	375M	78.6	94.2
BYOL (ours)	ResNet-200 (2×)	250M	**79.6**	**94.8**

(b) Other ResNet encoder architectures.

Table 1: Top-1 and top-5 accuracies (in %) under linear evaluation on ImageNet.

- Semi-supervised training on ImageNet

Method	Top-1		Top-5	
	1%	10%	1%	10%
Supervised [77]	25.4	56.4	48.4	80.4
InstDisc	-	-	39.2	77.4
PIRL [35]	-	-	57.2	83.8
SimCLR [8]	48.3	65.6	75.5	87.8
BYOL (ours)	**53.2**	**68.8**	**78.4**	**89.0**

(a) ResNet-50 encoder.

Method	Architecture	Param.	Top-1		Top-5	
			1%	10%	1%	10%
CPC v2 [32]	ResNet-161	305M	-	-	77.9	91.2
SimCLR [8]	ResNet-50 (2×)	94M	58.5	71.7	83.0	91.2
BYOL (ours)	ResNet-50 (2×)	94M	62.2	73.5	84.1	91.7
SimCLR [8]	ResNet-50 (4×)	375M	63.0	74.4	85.8	92.6
BYOL (ours)	ResNet-50 (4×)	375M	69.1	75.7	87.9	92.5
BYOL (ours)	ResNet-200 (2×)	250M	**71.2**	**77.7**	**89.5**	**93.7**

(b) Other ResNet encoder architectures.

Table 2: Semi-supervised training with a fraction of ImageNet labels.

Bootstrap Your Own Latent

Dive into BYOL

❖ Experiment Results

- Transfer to other classification tasks

Method	Food101	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear evaluation:												
BYOL (ours)	**75.3**	91.3	**78.4**	**57.2**	**62.2**	**67.8**	60.6	82.5	75.5	90.4	94.2	**96.1**
SimCLR (repro)	72.8	90.5	74.4	42.4	60.6	49.3	49.8	81.4	**75.7**	84.6	89.3	92.6
SimCLR [8]	68.4	90.6	71.6	37.4	58.8	50.3	50.3	80.5	74.5	83.6	90.3	91.2
Supervised-IN [8]	72.3	**93.6**	78.3	53.7	61.9	66.7	**61.0**	**82.8**	74.9	**91.5**	**94.5**	94.7
Fine-tuned:												
BYOL (ours)	**88.5**	**97.8**	86.1	**76.3**	63.7	91.6	**88.1**	**85.4**	**76.2**	91.7	**93.8**	97.0
SimCLR (repro)	87.5	97.4	85.3	75.0	63.9	91.4	87.6	84.5	75.4	89.4	91.7	96.6
SimCLR [8]	88.2	97.7	85.9	75.9	63.5	91.3	88.1	84.1	73.2	89.2	92.1	97.0
Supervised-IN [8]	88.3	97.5	**86.4**	75.8	**64.3**	**92.1**	86.0	85.0	74.6	**92.1**	93.3	**97.6**
Random init [8]	86.9	95.9	80.2	76.1	53.6	91.4	85.9	67.3	64.8	81.5	72.6	92.0

Table 3: Transfer learning results from ImageNet (IN) with the standard ResNet-50 architecture.

- Transfer to other vision tasks

Method	AP_{50}	mIoU
Supervised-IN [9]	74.4	74.4
MoCo [9]	74.9	72.5
SimCLR (repro)	75.2	75.2
BYOL (ours)	**77.5**	**76.3**

(a) Transfer results in semantic segmentation and object detection.

Method	Higher better			Lower better	
	pct.< 1.25	pct.$< 1.25^2$	pct.$< 1.25^3$	rms	rel
Supervised-IN [83]	81.1	95.3	98.8	0.573	**0.127**
SimCLR (repro)	83.3	96.5	99.1	0.557	0.134
BYOL (ours)	**84.6**	**96.7**	**99.1**	**0.541**	0.129

(b) Transfer results on NYU v2 depth estimation.

Table 4: Results on transferring BYOL's representation to other vision tasks.

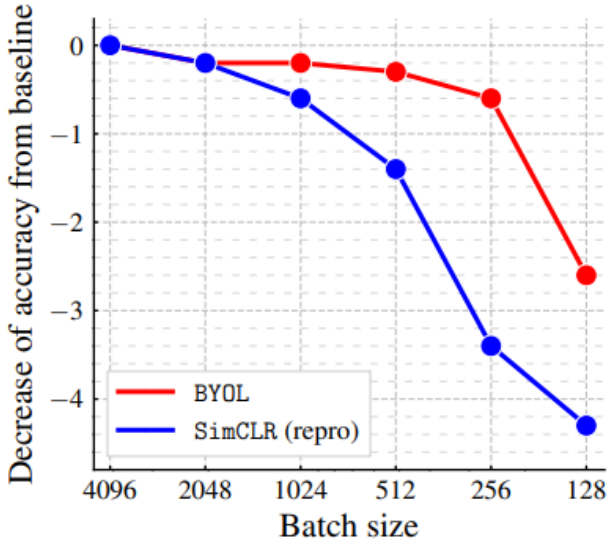
Bootstrap Your Own Latent

Dive into BYOL

❖ Ablation study

- Batch size
  - ✓ 배치 사이즈가 모델의 성능에 미치는 영향을 파악하기 위한 실험
  - ✓ SimCLR은 배치 사이즈가 작아짐에 따라서 성능저하가 BYOL보다 가파른 특징을 보임
  - ✓ BYOL은 negative sample을 쓰지 않기 때문에 배치 사이즈에 강건한 특징을 보임
  - ✓ 배치 사이즈 크기가 64일 때 보이는 급격한 성능 저하는 batch normalization의 특성에서 기인한 것으로 판단 됨

Batch size	Top-1 BYOL (ours)	Top-1 SimCLR (repro)	Top-5 BYOL (ours)	Top-5 SimCLR (repro)
4096	**72.5**	67.9	**90.8**	88.5
2048	72.4	67.8	90.7	88.5
1024	72.2	67.4	90.7	88.1
512	72.2	66.5	90.8	87.6
256	71.8	$64.3_{\pm 2.1}$	90.7	$86.3_{\pm 1.0}$
128	$69.6_{\pm 0.5}$	63.6	89.6	85.9
64	$59.7_{\pm 1.5}$	$59.2_{\pm 2.9}$	$83.2_{\pm 1.2}$	$83.0_{\pm 1.9}$



(a) Impact of batch size

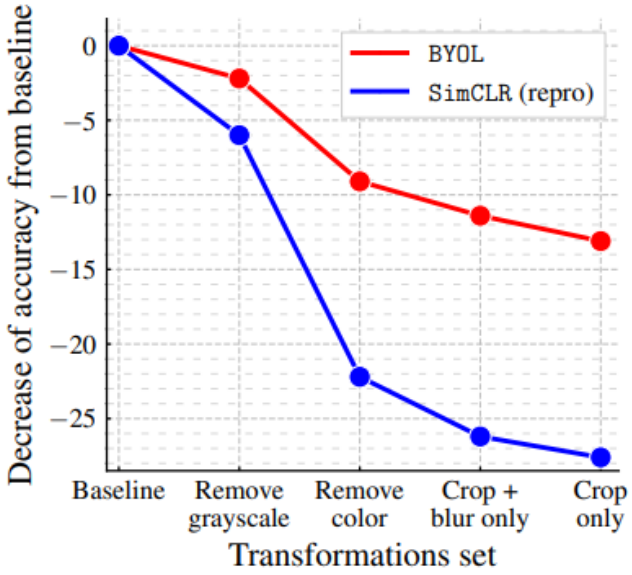
Bootstrap Your Own Latent

Dive into BYOL

❖ Ablation study

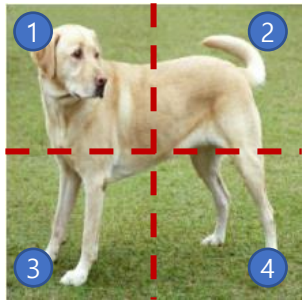
- Image Augmentation
  - ✓ Ablation study에서 BYOL과 SimCLR 모두 color distortion을 data augmentation에서 제외했을 때 성능 하락이 크게 나타남
  - ✓ SimCLR의 경우 color distortion의 유무에 따른 성능차가 확연함
  - ✓ BYOL의 경우 color distortion에 대해 비교적 강건한 성능을 보여줌
  - ✓ Target representation 정보를 online network에 저장하여 color histogram 외 추가적인 정보를 유지할 수 있음 → 어느 정도 좋은 학습을 보장

Image augmentation	Top-1 BYOL (ours)	Top-1 SimCLR (repro)	Top-5 BYOL (ours)	Top-5 SimCLR (repro)
Baseline	**72.5**	67.9	**90.8**	88.5
Remove flip	71.9	67.3	90.6	88.2
Remove blur	71.2	65.2	90.3	86.6
Remove color (jittering and grayscale)	63.4±0.7	45.7	85.3±0.5	70.6
Remove color jittering	71.8	63.7	90.7	85.9
Remove grayscale	70.3	61.9	89.8	84.1
Remove blur in $\mathcal{T}'$	72.4	67.5	90.8	88.4
Remove solarize in $\mathcal{T}'$	72.3	67.7	90.8	88.2
Remove blur and solarize in $\mathcal{T}'$	72.2	67.4	90.8	88.1
Symmetric blurring/solarization	72.5	68.1	90.8	88.4
Crop only	59.4±0.3	40.3±0.3	82.4	64.8±0.4
Crop and flip only	60.1±0.3	40.2	83.0±0.3	64.8
Crop and color only	70.7	64.2	90.0	86.2
Crop and blur only	61.1±0.3	41.7	83.9	66.4

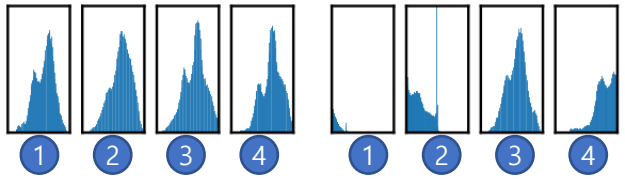


Bootstrap Your Own Latent

Dive into BYOL



〈 Color Histogram 〉

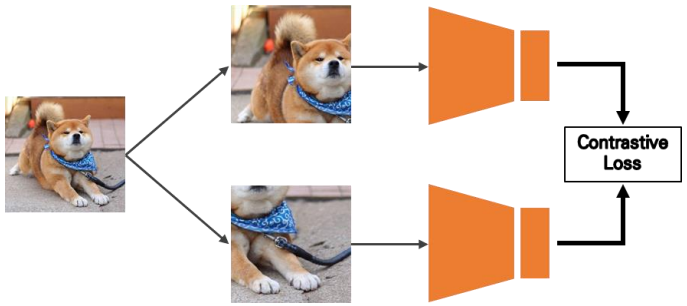


Color distortion 없음 Color distortion 있음

❖ Ablation study

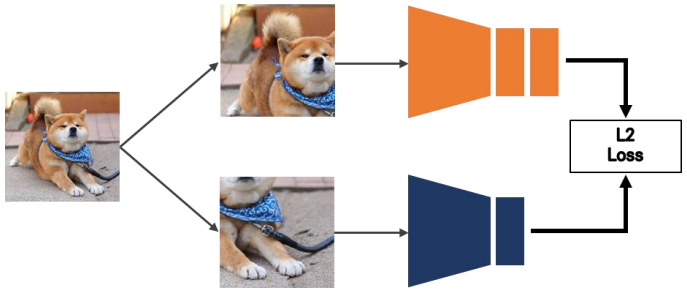
- Image Augmentation
  - ✓ 같은 이미지에서 crop된 일부 이미지라 하더라도 비슷한 color histogram 양상을 보임

SimCLR



- ✓ Color distortion을 적용하지 않을 경우 color histogram 위주로 분류 작업 진행
- ✓ Trivial solution을 학습하였기 때문에 좋은 representation을 얻을 수 없음

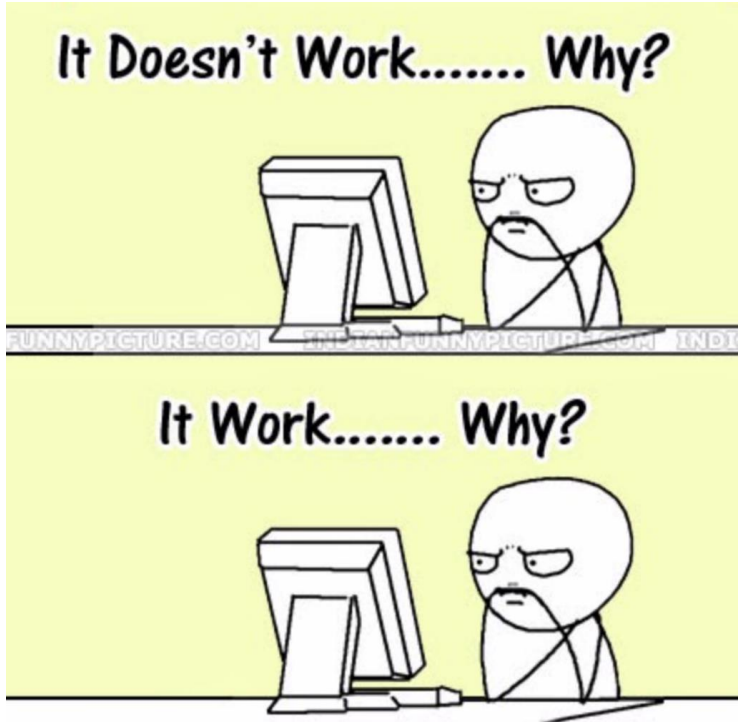
BYOL



- ✓ BYOL은 online representation으로 target representation을 예측하는 훈련을 진행함
- ✓ BYOL은 예측 성능을 높이기 위해서 color histogram 이외의 특징도 추출하려고 함
- ✓ 찾아낸 모든 정보는 loss를 통해서 online network를 업데이터 함으로써 보관됨
- ✓ 따라서 contrastive 방법론보다 augmentation에 강건한 성능을 보이게 됨

Bootstrap Your Own Latent

Dive into BYOL



어떤 요소가 BYOL에서 representation collapse를 방지해주는지에 대한 정설은 아직 없다

Bootstrap Your Own Latent

Dive into BYOL

❖ How BYOL prevents representation collapse?

- Bootstrap Your Own Latent : A New Approach to Self-Supervised Learning
  - ✓ Addition of a predictor to the online network
  - ✓ Use of a moving average of online parameters

새로운 가설 제시

- Understanding self-supervised and contrastive learning with "Bootstrap Your Own Latent"
  - ✓ Batch Normalization 때문

반박

- BYOL works even without batch statistics
  - ✓ Batch Normalization 때문 아님

Understanding self-supervised and contrastive learning with “Bootstrap Your Own Latent”

Dive into BYOL

GENERALLY INTELLIGENT

Posts

Podcast

Jobs

About us

Github

self_supervised: a Pytorch-Lightning implementation of self-supervised algorithms

jupyter_ascending: real-time Pycharm + Jupyter

Understanding self-supervised and contrastive learning with "Bootstrap Your Own Latent" (BYOL)

Aug 24, 2020 • Abe Fetterman (email), Josh Albrecht (email)

Summary

Unlike prior work like SimCLR and MoCo, the recent paper Bootstrap Your Own Latent (BYOL) from DeepMind demonstrates a state of the art method for self-supervised learning of image representations without an explicitly contrastive loss function. This simplifies training by removing the need for negative examples in the loss function. We highlight two surprising findings from our work on reproducing BYOL:

(1) BYOL often performs no better than random when batch normalization is removed, and

(2) the presence of batch normalization implicitly causes a form of contrastive learning.

Understanding self-supervised and contrastive learning with "Bootstrap Your Own Latent"

Dive into BYOL

❖ Batch normalization does prevent collapsing

- Batch Normalization을 제거할 경우 random weight인 모델의 성능과 비슷함

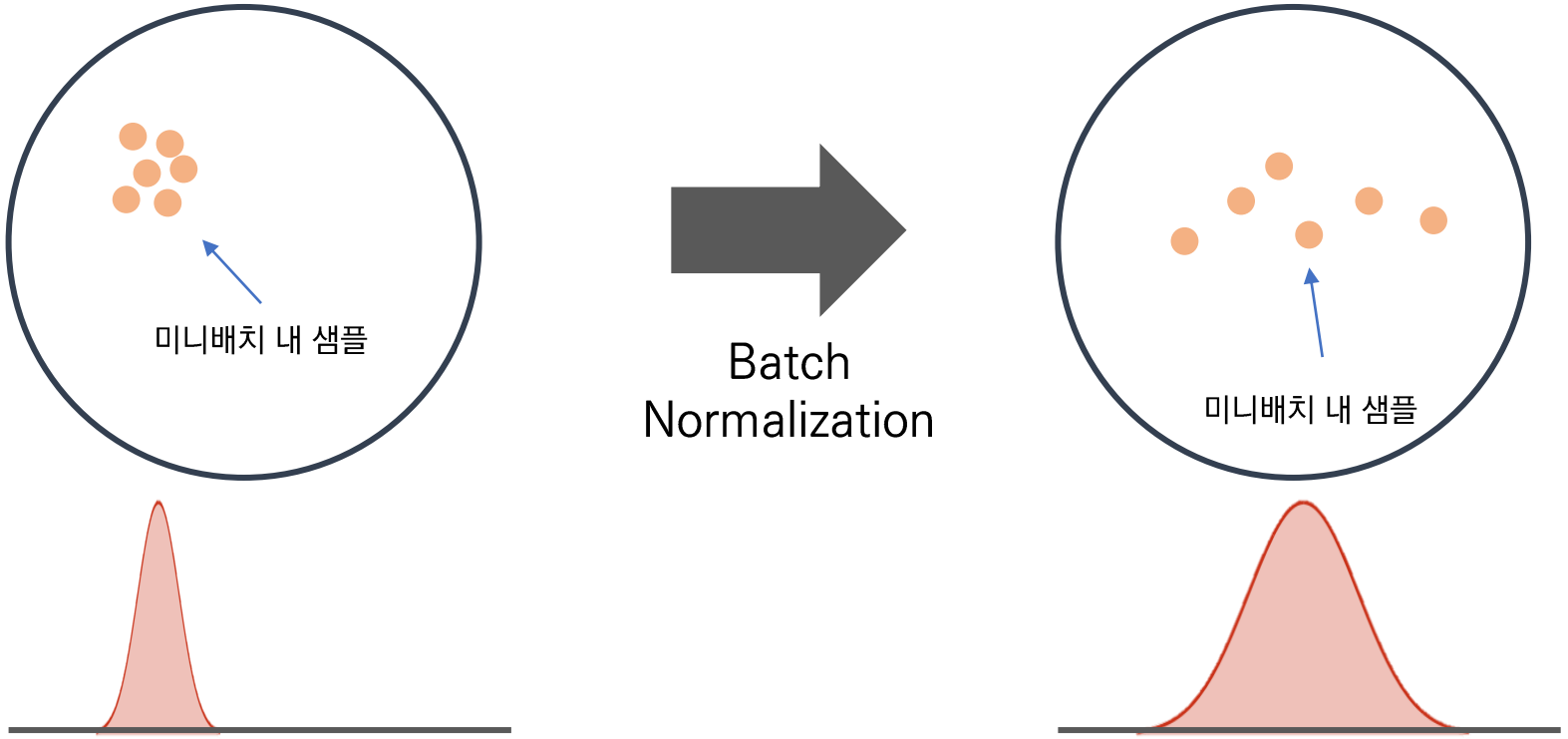
Name	Projection	Prediction	Loss Function	Contrastive	Performance
Contrastive Loss	-	-	Cross Entropy	Explicit	44.1
BYOL	Batch Norm	Batch Norm	L2	Implicit	57.7
Projection BN Only	Batch Norm	-	L2	Implicit	55.3
Prediction BN Only	-	Batch Norm	L2	Implicit	48
No Normalization	-	-	L2	-	28.3
Layer Norm	Layer Norm	Layer Norm	L2	-	29.4
Random	-	-	L2	-	28.8

Understanding self-supervised and contrastive learning with "Bootstrap Your Own Latent"

Dive into BYOL

❖ Batch normalization does prevent collapsing

- Batch Normalization의 분포 정규화 기능이 collapse를 방지함

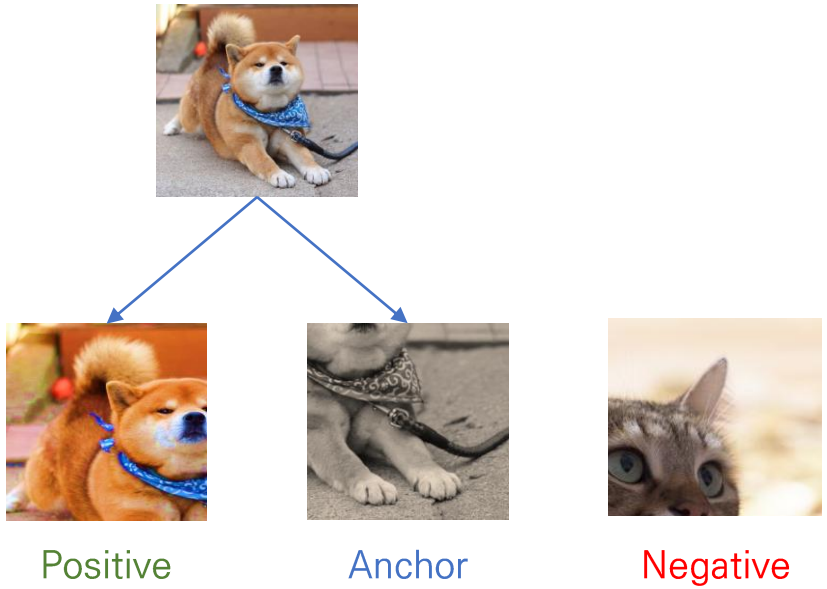


Understanding self-supervised and contrastive learning with "Bootstrap Your Own Latent"

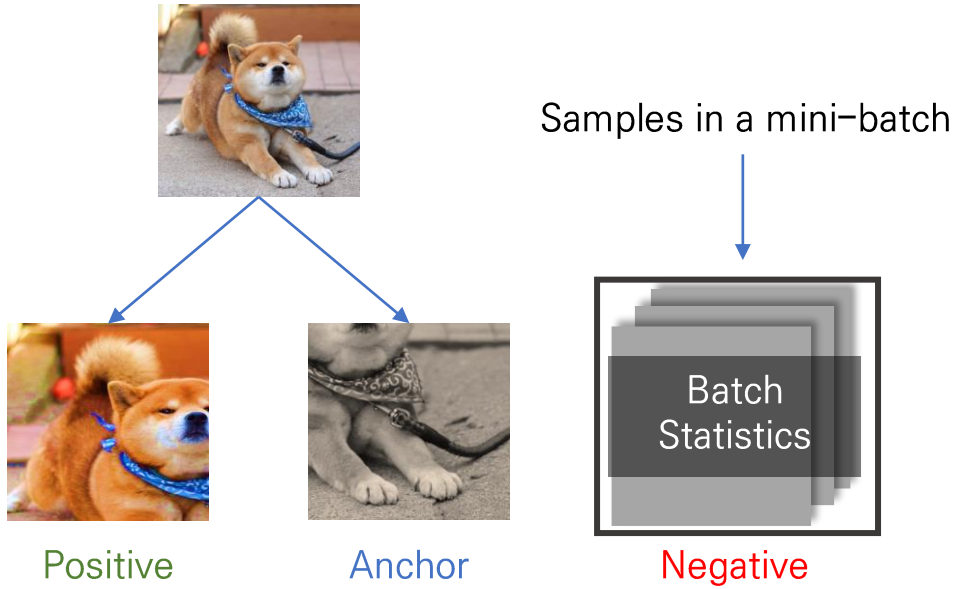
Dive into BYOL

❖ Batch normalization does prevent collapsing

- Batch Normalization이 implicit contrastive learning 역할을 수행함



Explicit Contrastive Learning



Implicit Contrastive Learning

BYOL doesn't even need batch statistics

Dive into BYOL

BYOL works *even* without batch statistics

Pierre H. Richemond[*,1,2] Jean-Bastien Grill[*,1] Florent Altché[*,1] Corentin Tallec[*,1] Florian Strub[*,1]

Andrew Brock[1] Samuel Smith[1] Soham De[1] Razvan Pascanu[1]

Bilal Piot[1] Michal Valko[1]

[1]DeepMind [2]Imperial College

phr17@ic.ac.uk [jbgrill,fstrub,altche,corentint]@google.com

➡ Batch statistics가 BYOL이 좋은 representation을 배우는데 중요한 역할을 할 것이라는 주장을 반박함

BYOL doesn't even need batch statistics

Dive into BYOL

❖ Batch normalization does not prevent collapsing

- 가설 1) BYOL은 batch normalization이 collapse를 피하는데 필요한 implicit negative term을 주기 때문에 batch normalization이 필요함

→ 모든 batch normalization을 제거할 경우 representation collapse가 잘 발생하는 것은 맞음
→ Batch normalization이 implicit negative contrastive term으로서 활용될 수 있다는 것도 맞음

→ 하지만 BYOL에서 batch normalization의 주요 기여점은 불안정한 파라미터 초기값을 잘 잡아주는 것
→ 실제로 실험 결과 batch normalization을 쓰지 않더라도 초기값을 잘 잡아주었더니 collapse가 발생하지 않았음

Table 1: *Ablation results on normalization, per network component:* The numbers correspond to top-1 linear accuracy (%), 300 epochs on ImageNet, averaged over 3 seeds.

Encoder	BN				LN				-				-		
Projector	BN		-		LN		-		BN		LN		-		
Predictor	BN	-	BN	-	LN	-	LN	-	BN	-	LN	-	BN	LN	-
BYOL	73.2	73.2	72.0	72.1	0.1	5.4	0.1	0.1	62.6	0.1	0.1	0.1	61.1	0.1	0.1
SimCLR	69.3		68.5		68.0		67.8		53.8		56.7		0.1		

Table 2: *Summary of our results*: top-1 accuracy with linear evaluation on ImageNet, at 1000 epochs

BYOL variant	Vanilla BN	No BN	Modified init.	GN + WS
Uses batch statistics	Yes	No	No	No
Top-1 accuracy (%)	74.3	0.1	65.7	73.9

BYOL doesn't even need batch statistics

Dive into BYOL

❖ Batch normalization does not prevent collapsing

- 가설 2) BYOL은 batch statistics를 통한 implicit contrastive 효과를 받지 않는 이상 높은 성능을 발휘 할 수 없음

→ Group normalization과 weight standardization을 조합하여 대체하였더니 batch normalization을 쓸 때와 비슷한 성능이 나옴

Table 2: *Summary of our results*: top-1 accuracy with linear evaluation on ImageNet, at 1000 epochs

BYOL variant	Vanilla BN	No BN	Modified init.	GN + WS
Uses batch statistics	Yes	No	No	No
Top-1 accuracy (%)	74.3	0.1	65.7	73.9

Other papers to read

Dive into BYOL

Exploring Simple Siamese Representation Learning

Xinlei Chen Kaiming He
Facebook AI Research (FAIR)

2020.11 arXiv

Understanding self-supervised Learning Dynamics without Contrastive Pairs

---

Yuandong Tian [1] Xinlei Chen [1] Surya Ganguli [1 2]

Facebook AI Research

Abstract

Contrastive approaches to self-supervised learning (SSL) learn representations by minimizing

man et al., 2019) whereby the hidden representations of two augmented views of the same object (positive pairs) are brought closer together, while those of different ob-

2021.02 arXiv