DMQA Open Seminar

OOD Detection for Image Classification Part3 : Outlier Exposure Approach

2025. 05. 02

고려대학교 산업경영공학과

Data Mining & Quality Analytics Lab.

임 새 린

발표자 소개

❖ 임새린 (Saerin Lim)

- 고려대학교 산업경영공학과 Data Mining & Quality Analytics Lab.
- Ph.D. Student (2021.03 ~ Present)
- 지도 교수 : 김성범 교수님

❖ Research Interest

- Self-supervised learning & Semi-supervised learning

❖ Contact

- E-mail : momo_om@korea.ac.kr

목차
Outline

1. Overview
2. Methods
3. Conclusions

Overview

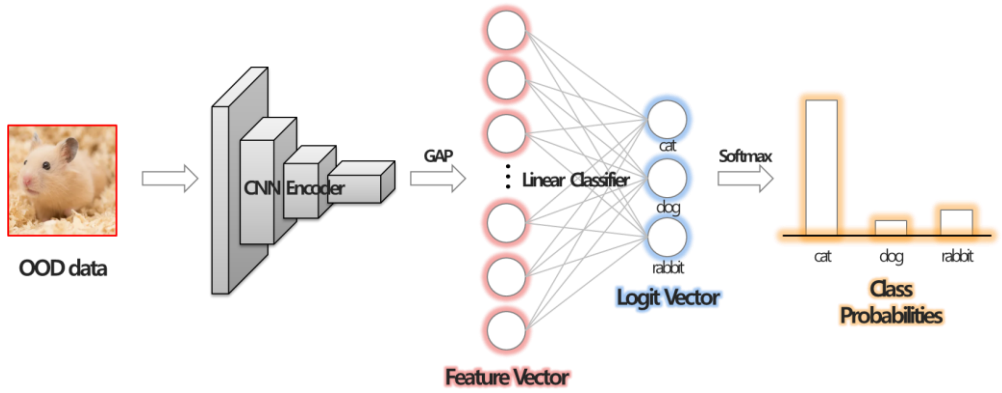
OOD Detection Approach

❖ OOD detection을 해결하기 위한 접근법은 크게 post-hoc 방식과 outlier exposure 방식이 있음

OOD detection

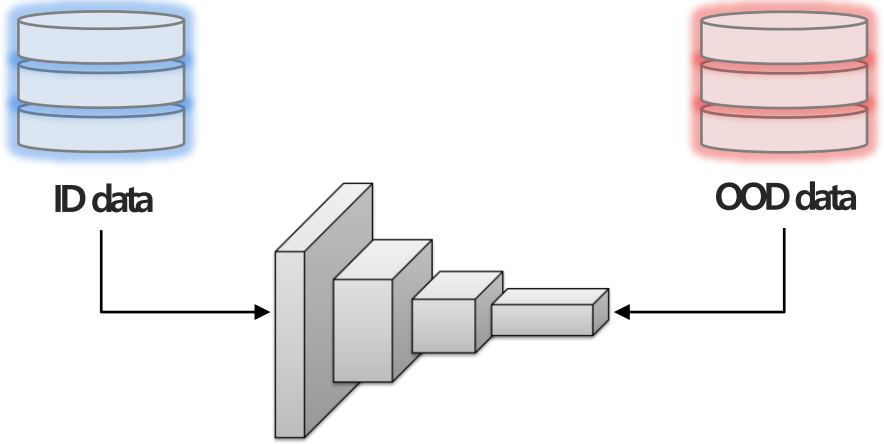
Post-hoc Approach

- 추가적인 학습 없이 모델에서 나오는 정보 (feature vector, logit vector 등)를 활용해서 ID와 OOD를 구분할 수 있는 OOD score function을 개발



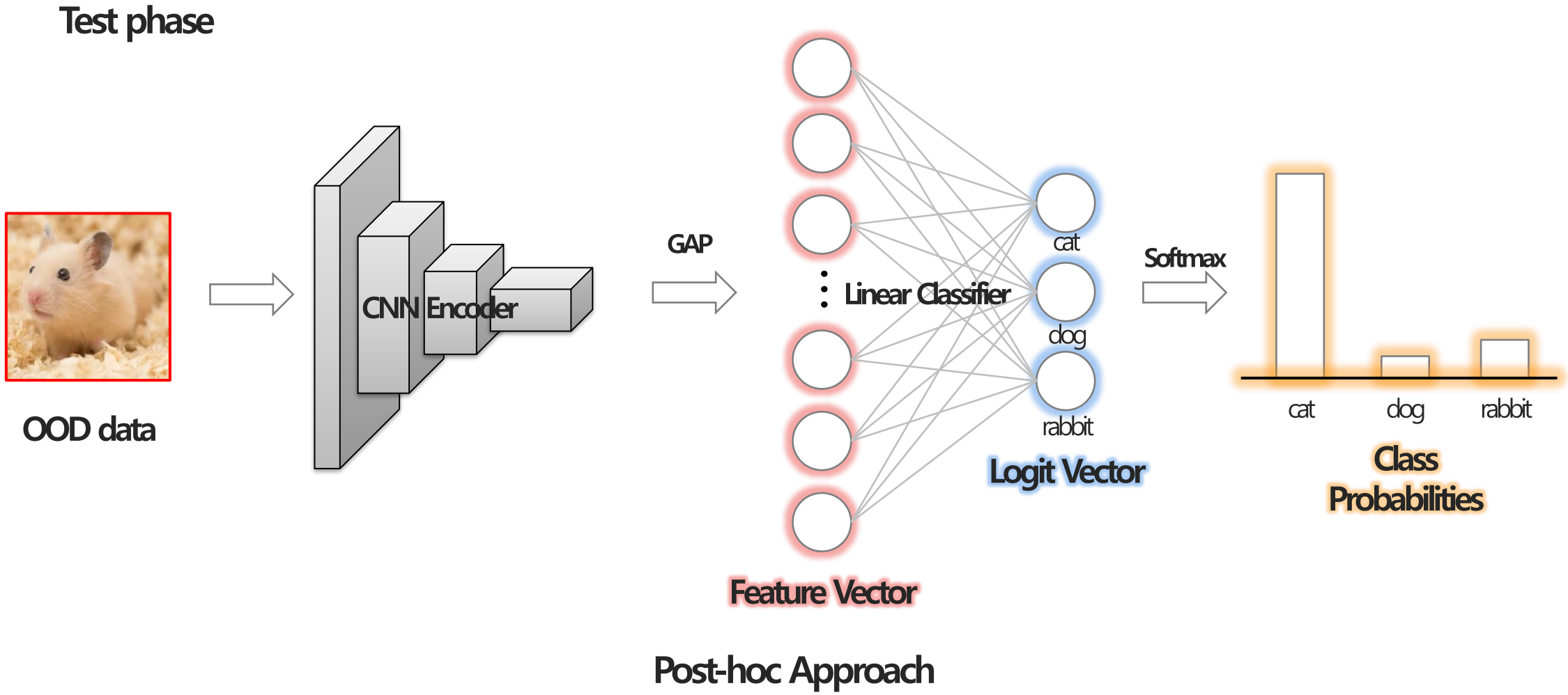
Outlier Exposure Approach

- 실제 혹은 가상 OOD 데이터를 활용하여 모델이 직접적으로 ID와 OOD를 구분할 수 있는 loss function을 개발



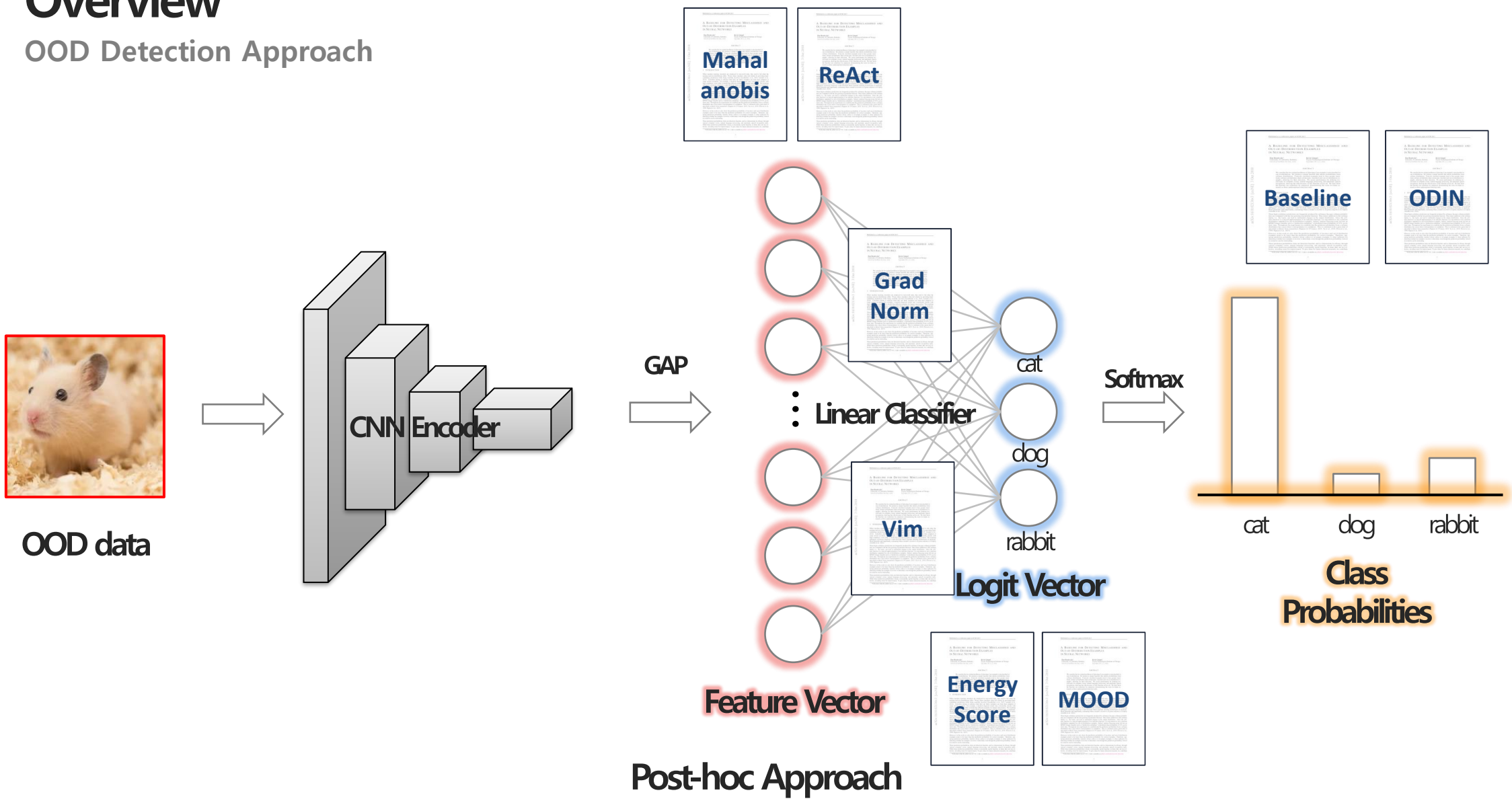
Overview

OOD Detection Approach



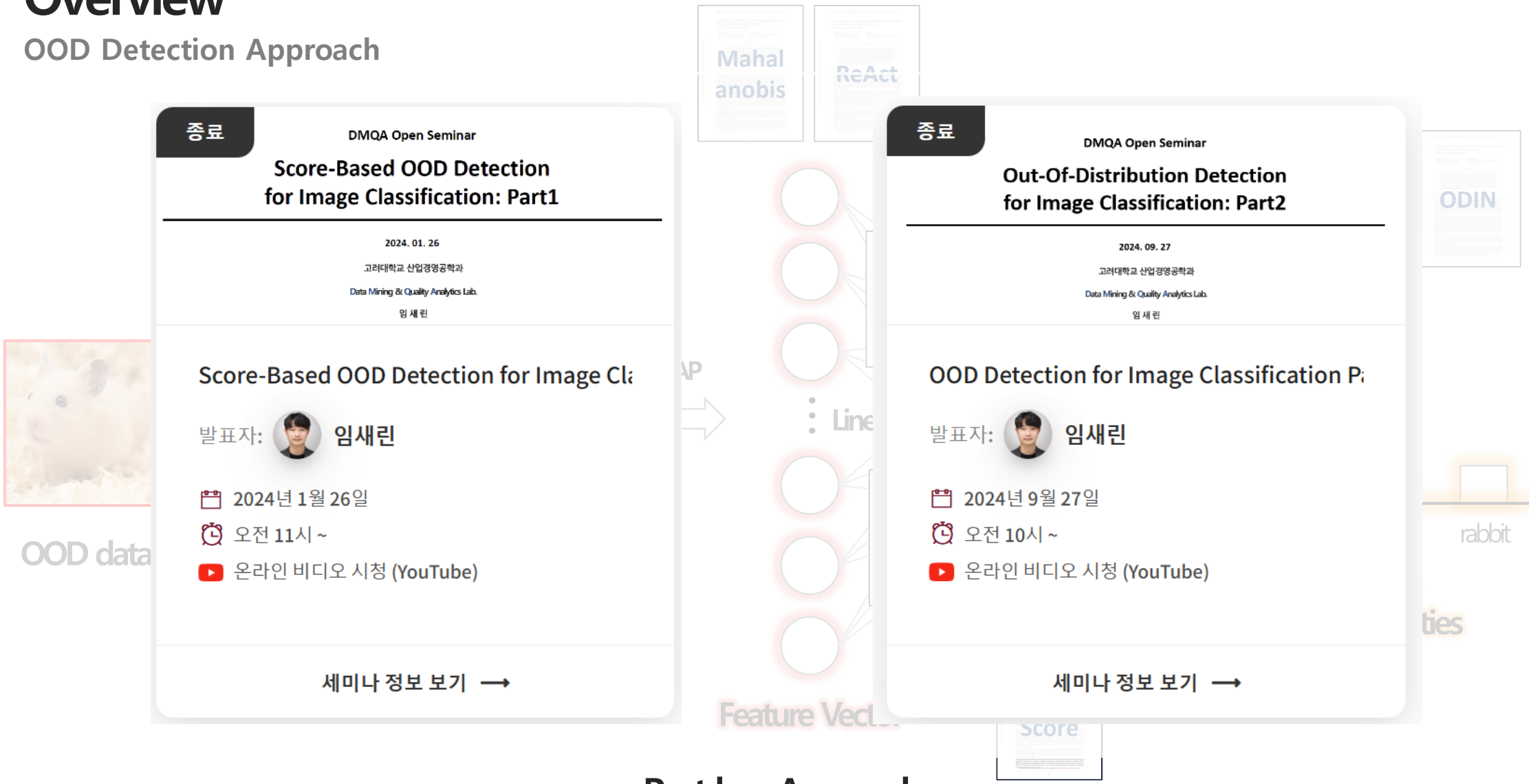
Overview

OOD Detection Approach



Overview

OOD Detection Approach



Post-hoc Approach

Overview

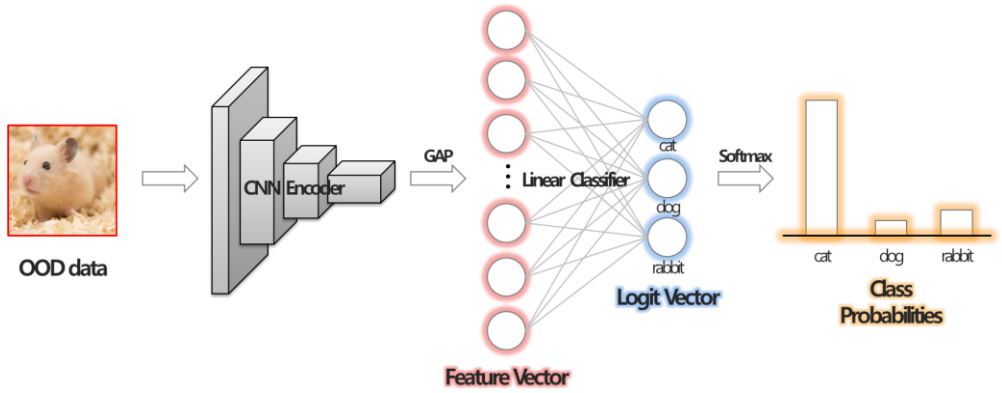
OOD Detection Approach

❖ OOD detection을 해결하기 위한 접근법은 크게 post-hoc 방식과 outlier exposure 방식이 있음

OOD detection

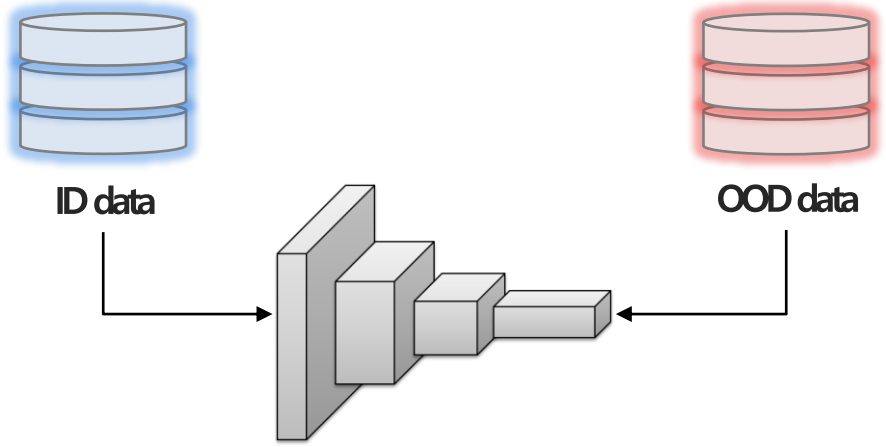
Post-hoc Approach

- 추가적인 학습 없이 모델에서 나오는 정보 (feature vector, logit vector 등)를 활용해서 ID와 OOD를 구분할 수 있는 OOD score function을 개발



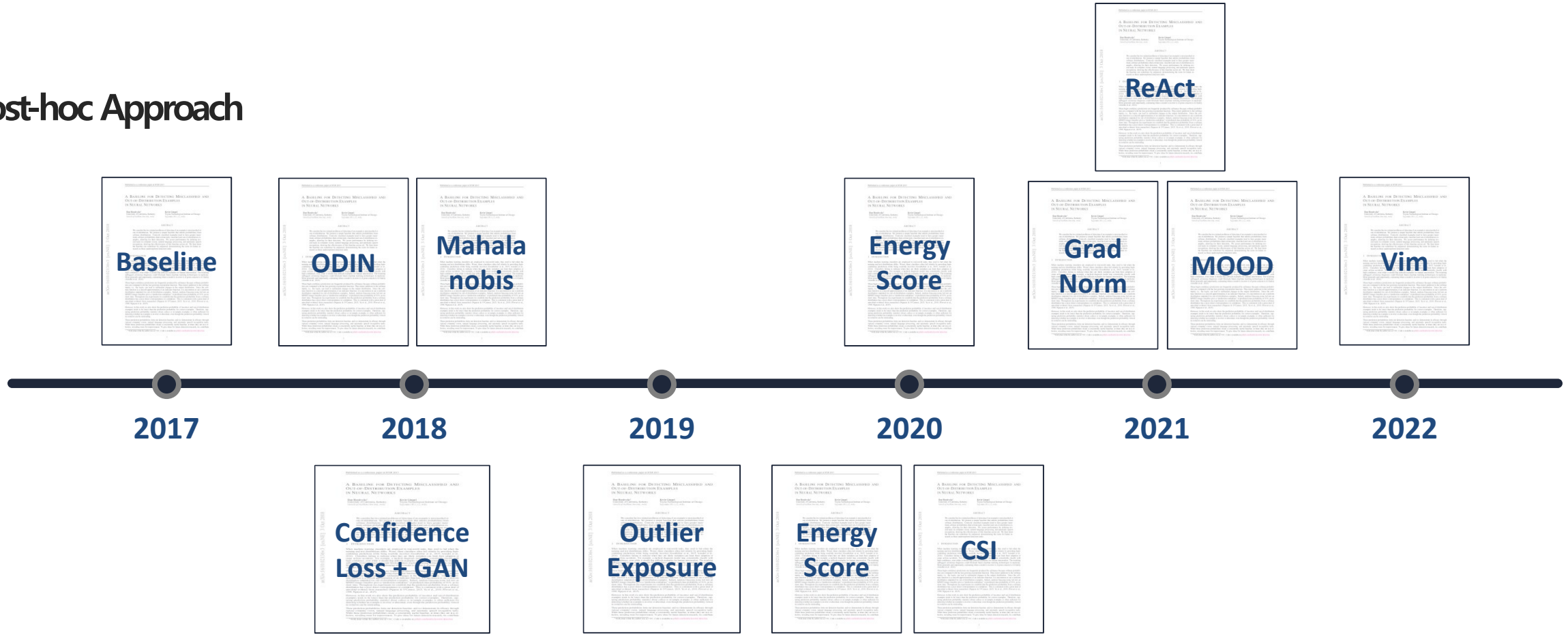
Outlier Exposure Approach

- 실제 혹은 가상 OOD 데이터를 활용하여 모델이 직접적으로 ID와 OOD를 구분할 수 있는 loss function을 개발



Overview

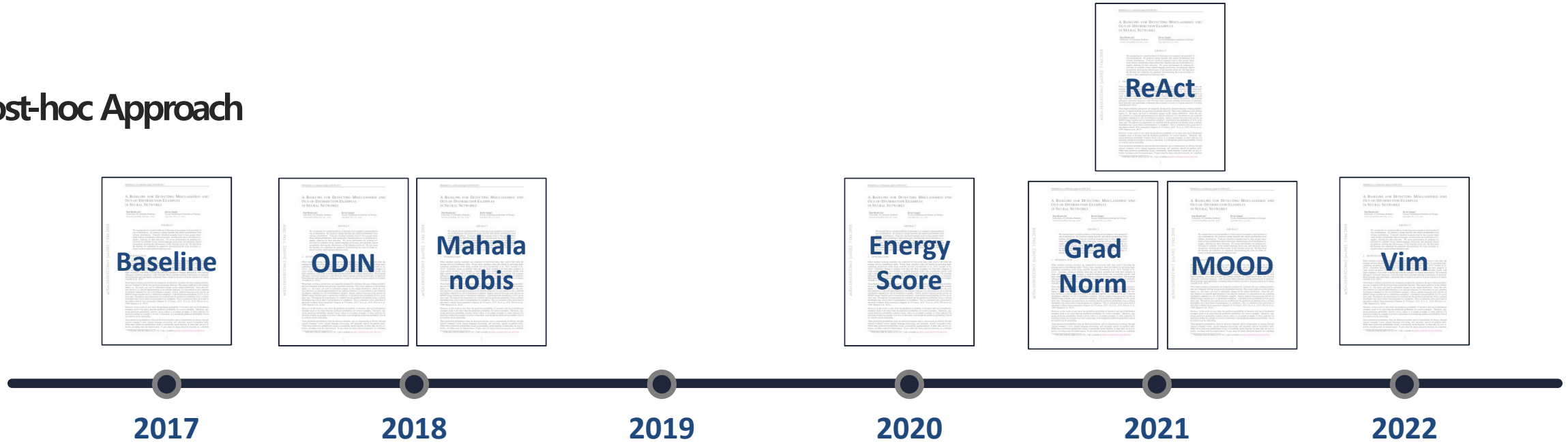
Post-hoc Approach



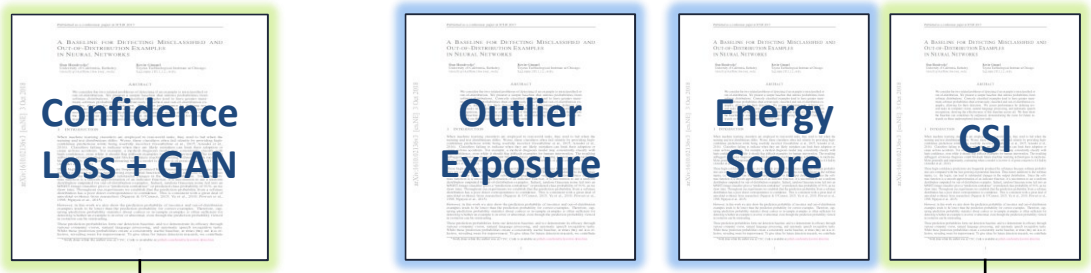
Outlier Exposure Approach

Overview

Post-hoc Approach



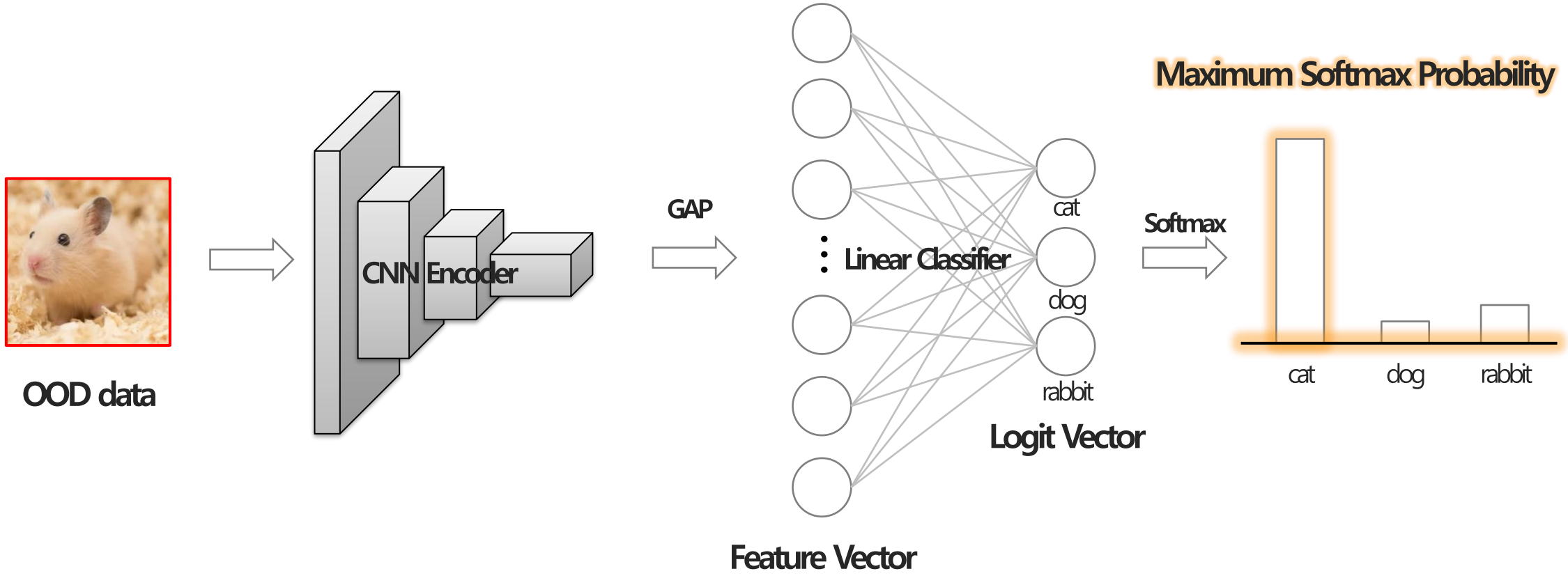
Outlier Exposure Approach



Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

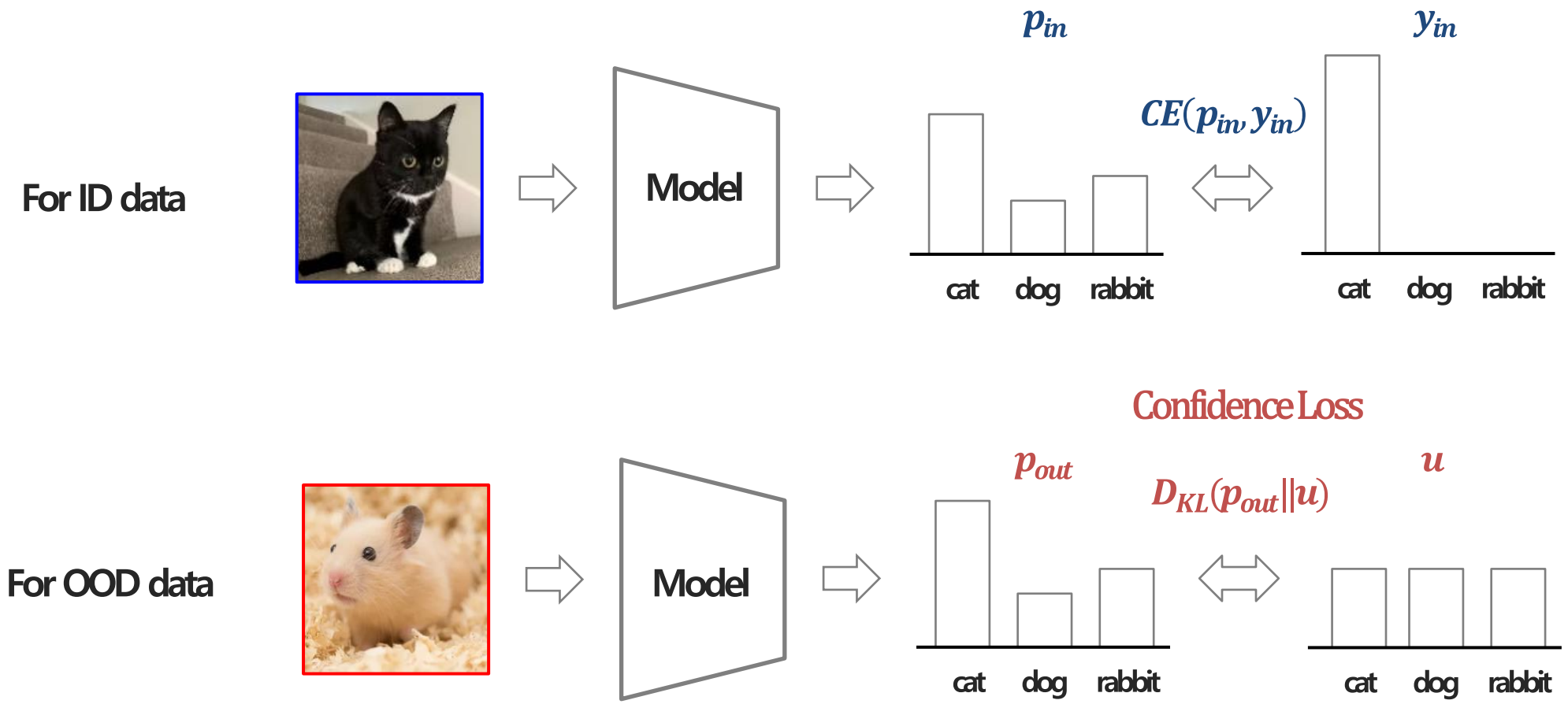
Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)



Method 1

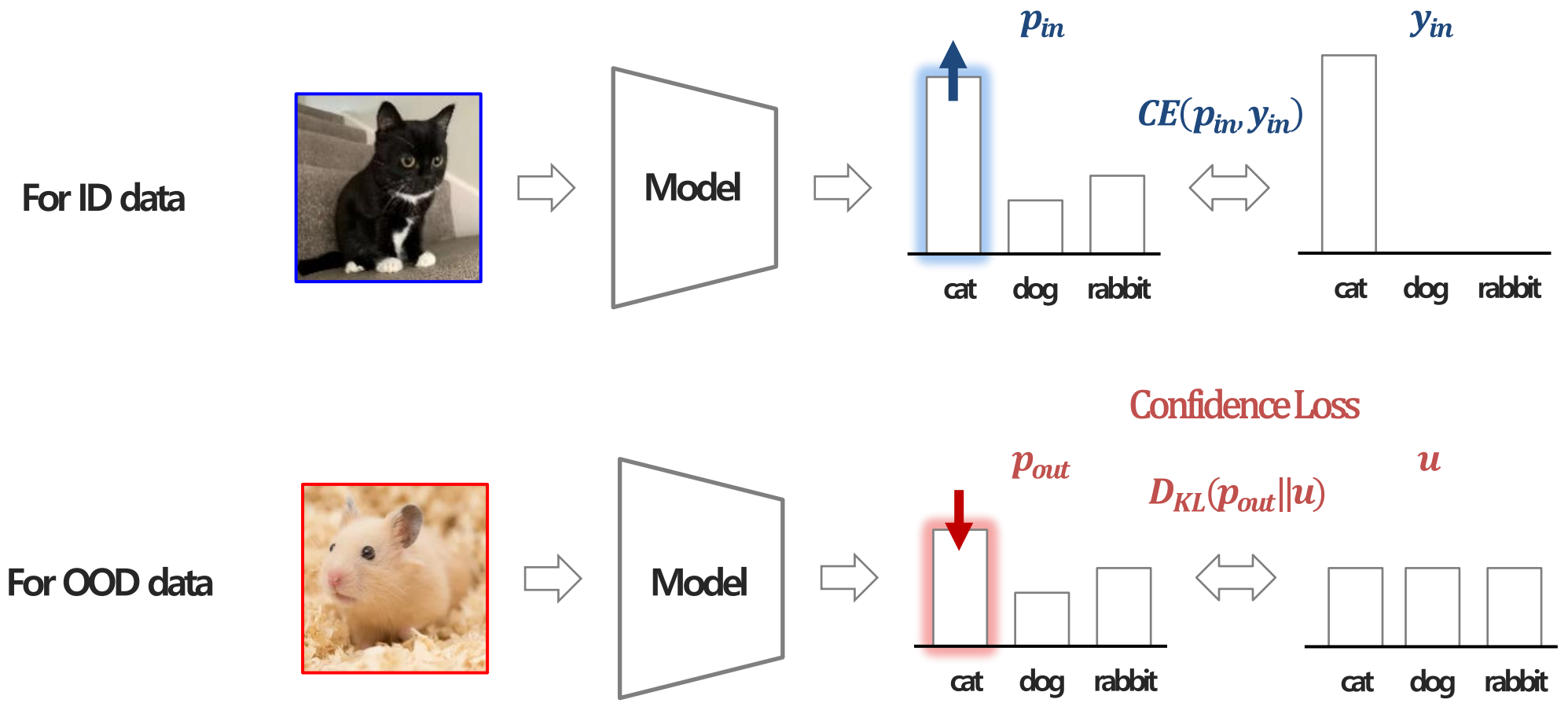
Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)



Training Confidence-Calibrated Classifiers

Method 1

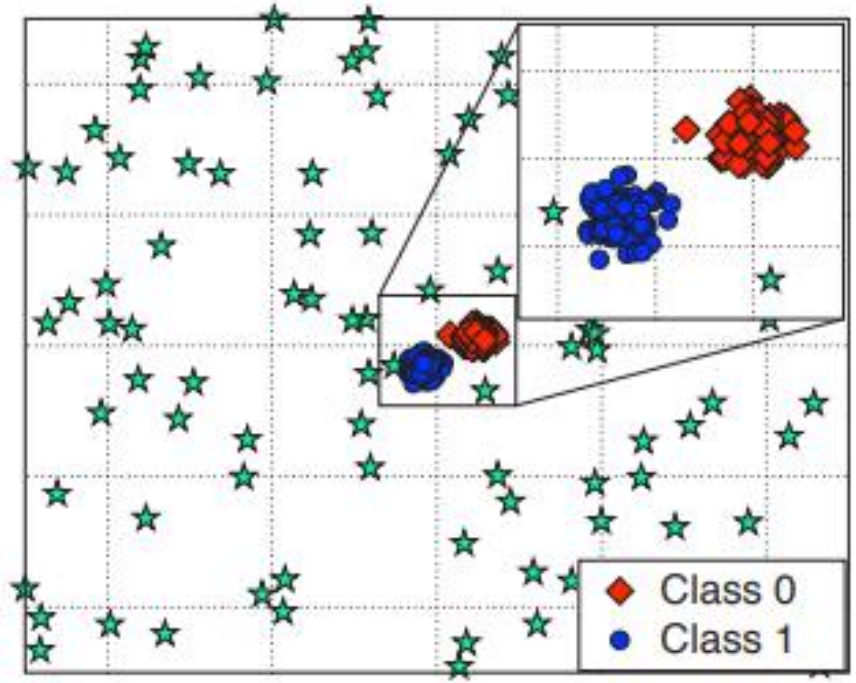
Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

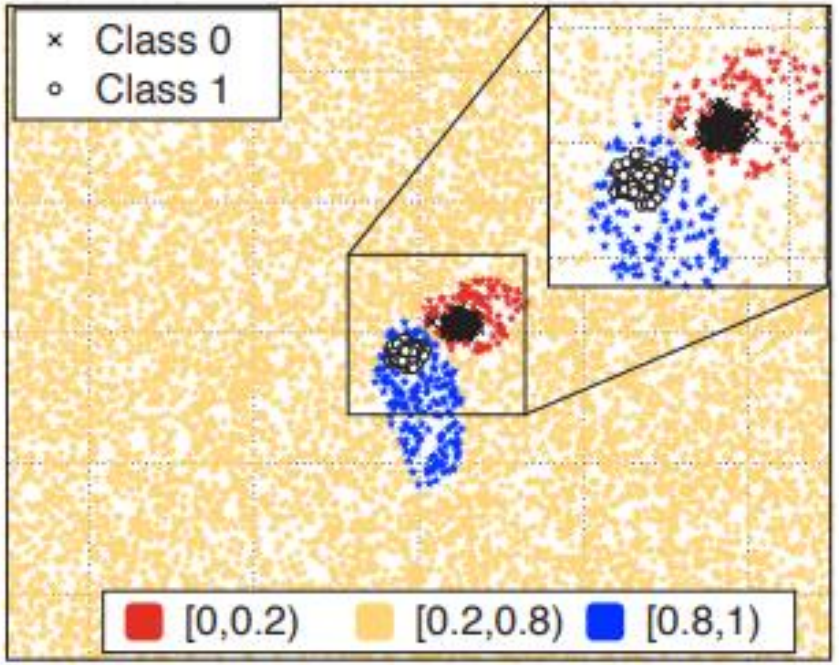


Training Confidence-Calibrated Classifiers

Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)





● ◆ : Training ID sample

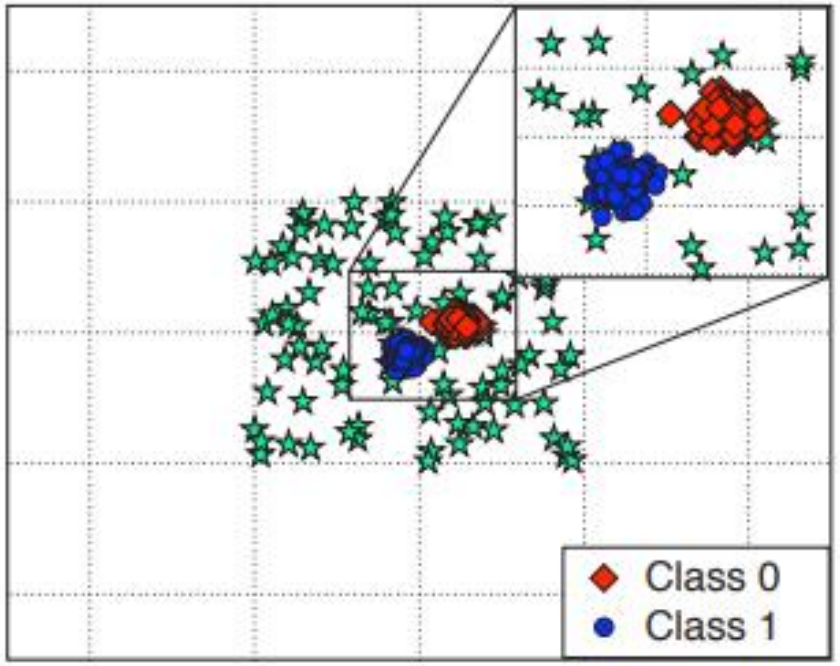
★ : Training OOD sample

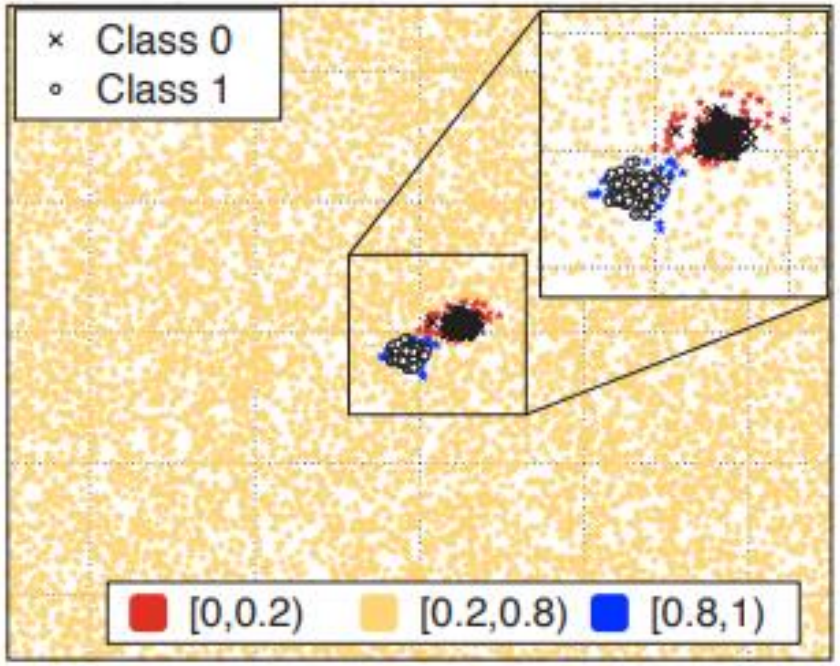
★ ★ : High confidence area

☆ : Low confidence area

Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)





● ◆ : Training ID sample

★ : Training OOD sample

★ ★ : High confidence area

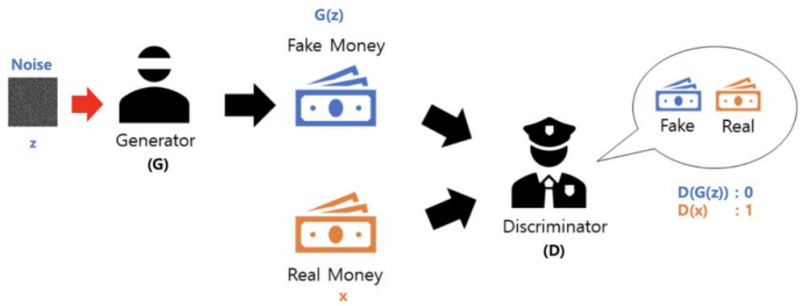
☆ : Low confidence area

Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

Adversarial Generative Network (GAN)

$$\min_{G}\max_{D}\mathbb{E}_{p_{in}(x)}[logD(x)]+\mathbb{E}_{p_{pri}(z)}[log\,1-D(G(z))]$$

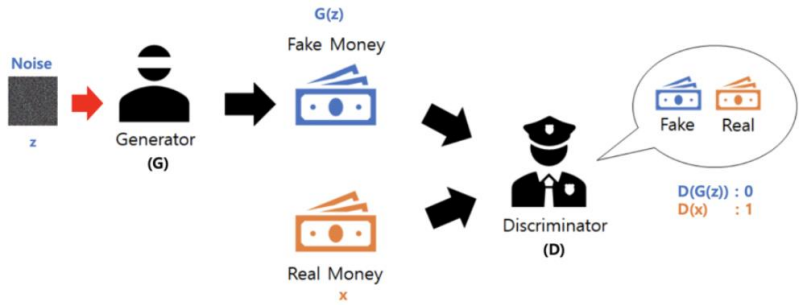


Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

Adversarial Generative Network (GAN)

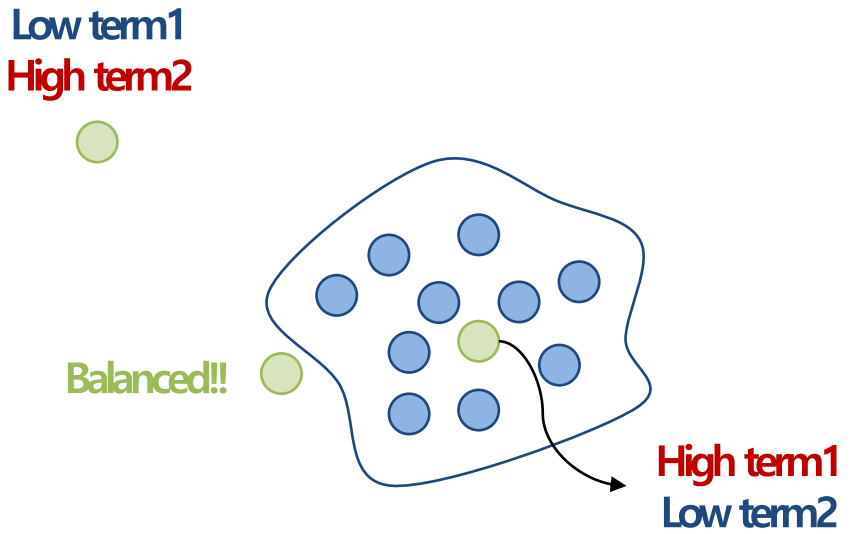
$$\min_G \max_D \mathbb{E}_{p_{in}(x)}[logD(x)] + \mathbb{E}_{p_{pri}(z)}[log\,1 - D(G(z))]$$



Adversarial Generative Network for OOD sampling

$$\min_G \max_D \mathbb{E}_{p_{in}(x)}[logD(x)] + \mathbb{E}_{p_{pri}(z)}[log\,1 - D(G(z))]$$

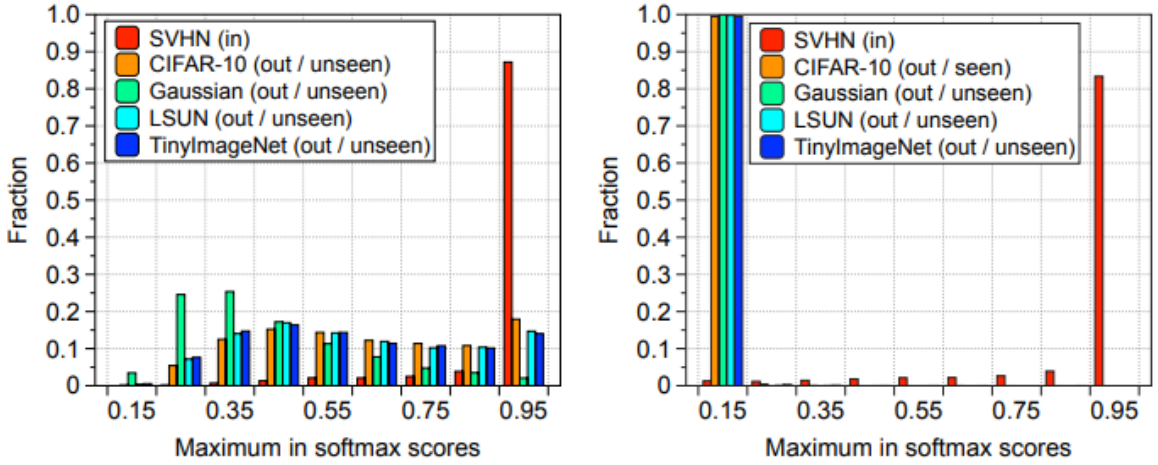
$$+\beta\,\mathbb{E}_{p_{pri}(z)}[KL(u(y)||P_\theta(y|G(z)))]$$



Method 1

Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

In-dist	Out-of-dist	Classification accuracy	TNR at TPR 95%	AUROC	Detection accuracy	AUPR in	AUPR out
			Cross entropy loss / Confidence loss				
SVHN	CIFAR-10 (seen)	93.82 / **94.23**	47.4 / **99.9**	62.6 / **99.9**	78.6 / **99.9**	71.6 / **99.9**	91.2 / **99.4**
	TinyImageNet (unseen)		49.0 / **100.0**	64.6 / **100.0**	79.6 / **100.0**	72.7 / **100.0**	91.6 / **99.4**
	LSUN (unseen)		46.3 / **100.0**	61.8 / **100.0**	78.2 / **100.0**	71.1 / **100.0**	90.8 / **99.4**
	Gaussian (unseen)		56.1 / **100.0**	72.0 / **100.0**	83.4 / **100.0**	77.2 / **100.0**	92.8 / **99.4**
CIFAR-10	SVHN (seen)	80.14 / **80.56**	13.7 / **99.8**	46.6 / **99.9**	66.6 / **99.8**	61.4 / **99.9**	73.5 / **99.8**
	TinyImageNet (unseen)		**13.6** / 9.9	**39.6** / 31.8	**62.6** / 58.6	**58.3** / 55.3	**71.0** / 66.1
	LSUN (unseen)		**14.0** / 10.5	**40.7** / 34.8	**63.2** / 60.2	**58.7** / 56.4	**71.5** / 68.0
	Gaussian (unseen)		2.8 / **3.3**	10.2 / **14.1**	50.0 / 50.0	48.1 / **49.4**	39.9 / **47.0**



(a) Cross entropy loss (b) Confidence loss in (1)

Method 1

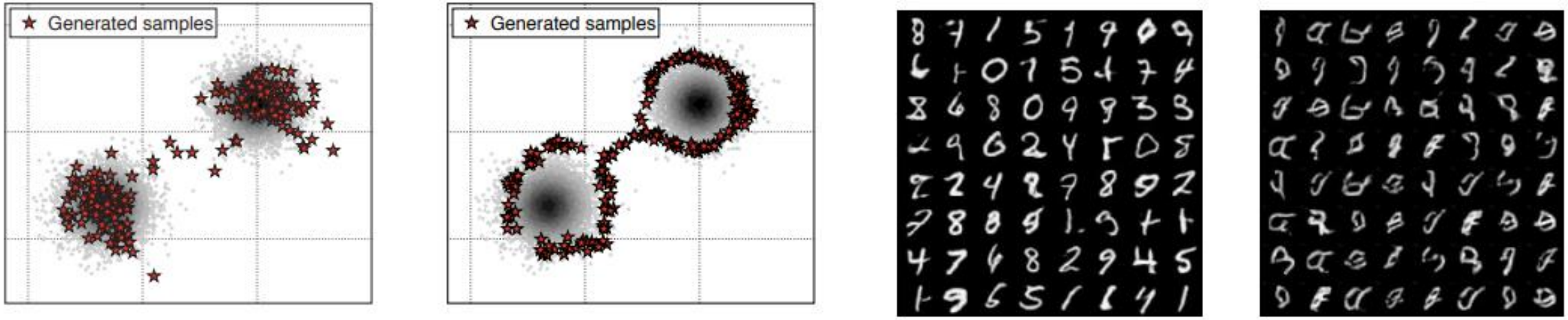
Training Confidence-Calibrated Classifiers for Detecting Out-of-Distribution Samples (2018, ICLR)

기존 GAN

제안 GAN

기존 GAN

제안 GAN



(a) (b) (c) (d)

Figure 3: The generated samples from original GAN (a)/(c) and proposed GAN (b)/(d). In (a)/(b), the grey area is the 2D histogram of training in-distribution samples drawn from a mixture of two Gaussian distributions and red points indicate generated samples by GANs.

Deep Anomaly Detection with Outlier Exposure

(2019, ICLR)

Method 2

Deep Anomaly Detection with Outlier Exposure (2019, ICLR)

3 OUTLIER EXPOSURE

We consider the task of deciding whether or not a sample is from a learned distribution called $\mathcal{D}_{\text{in}}$. Samples from $\mathcal{D}_{\text{in}}$ are called "in-distribution," and otherwise are said to be "out-of-distribution" (OOD) or samples from $\mathcal{D}_{\text{out}}$. In real applications, it may be difficult to know the distribution of outliers one will encounter in advance. Thus, we consider the realistic setting where $\mathcal{D}_{\text{out}}$ is unknown. Given a parametrized OOD detector and an Outlier Exposure (OE) dataset $\mathcal{D}_{\text{out}}^{\text{OE}}$, disjoint from $\mathcal{D}_{\text{out}}^{\text{test}}$, we train the model to discover signals and learn heuristics to detect whether a query is sampled from $\mathcal{D}_{\text{in}}$ or $\mathcal{D}_{\text{out}}^{\text{OE}}$. We find that these heuristics generalize to unseen distributions $\mathcal{D}_{\text{out}}$.

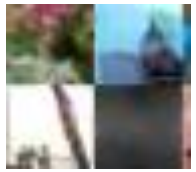
Deep parametrized anomaly detectors typically leverage learned representations from an auxiliary task, such as classification or density estimation. Given a model f and the original learning objective $\mathcal{L}$, we can thus formalize Outlier Exposure as minimizing the objective

$$\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{in}}}[\mathcal{L}(f(x),y)+\lambda\mathbb{E}_{x'\sim\mathcal{D}_{\text{out}}^{\text{OE}}}[\mathcal{L}_{\text{OE}}(f(x'),f(x),y)]]$$

over the parameters of f. In cases where labeled data is not available, then y can be ignored. Outlier Exposure can be applied with many types of data and original tasks. Hence, the specific formulation of $\mathcal{L}_{\text{OE}}$ is a design choice, and depends on the task at hand and the OOD detector used. For example, when using the maximum softmax probability baseline detector (Hendrycks & Gimpel, 2017), we set $\mathcal{L}_{\text{OE}}$ to the cross-entropy from $f(x')$ to the uniform distribution (Lee et al., 2018). When the original objective $\mathcal{L}$ is density estimation and labels are not available, we set $\mathcal{L}_{\text{OE}}$ to a margin ranking loss on the log probabilities $f(x')$ and $f(x)$.

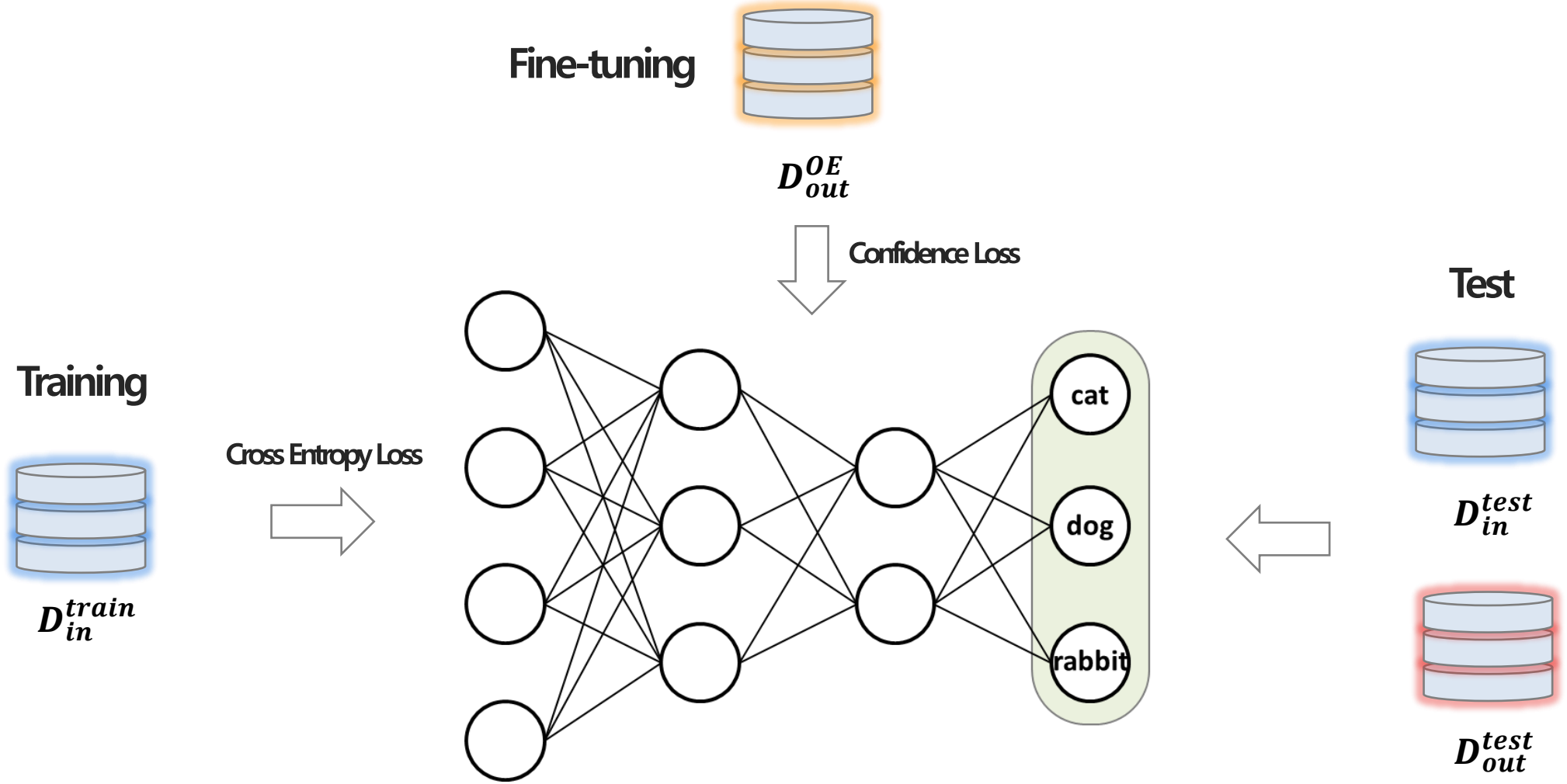
Method 2

Deep Anomaly Detection with Outlier Exposure (2019, ICLR)

		저해상도 이미지셋		고해상도 이미지셋
	SVHN	CIFAR10&100	TinyImageNet	Places365
$D_{in}^{train}, D_{in}^{test}$				
D_{out}^{OE}		80 Million Tiny Images		ImageNet-22K
D_{out}^{test}	Gaussian, Bernoulli, Blobs, Icons-50, Textures, Places365, LSUN, CIFAR10, Chars74K	Gaussian, Rademacher, Blobs, Textures, SVHN, Places365, LSUN, CIFAR10, CIFAR100	Gaussian, Rademacher, Blobs, Textures, SVHN, Places365, LSUN, ImageNet	Gaussian, Rademacher, Blobs, Icons-50, Textures, SVHN, Places69, ImageNet

Method 2

Deep Anomaly Detection with Outlier Exposure (2019, ICLR)



Method 2

Deep Anomaly Detection with Outlier Exposure (2019, ICLR)

	FPR95 ↓		AUROC ↑		AUPR ↑	
$\mathcal{D}_{\text{in}}$	MSP	+OE	MSP	+OE	MSP	+OE
SVHN	6.3	0.1	98.0	100.0	91.1	99.9
CIFAR-10	34.9	9.5	89.3	97.8	59.2	90.5
CIFAR-100	62.7	38.5	73.1	87.9	30.1	58.2
Tiny ImageNet	66.3	14.0	64.9	92.2	27.2	79.3
Places365	63.5	28.2	66.5	90.6	33.1	71.0

Table 1: Out-of-distribution image detection for the maximum softmax probability (MSP) baseline detector and the MSP detector after fine-tuning with Outlier Exposure (OE). Results are percentages and also an average of 10 runs. Expanded results are in Appendix A.

Method 2

Deep Anomaly Detection with Outlier Exposure (2019, ICLR)

	FPR95 ↓			AUROC ↑			AUPR ↑		
$\mathcal{D}_{\text{in}}$	MSP	+GAN	+OE	MSP	+GAN	+OE	MSP	+GAN	+OE
CIFAR-10	32.3	37.3	11.8	88.1	89.6	97.2	51.1	59.0	88.5
CIFAR-100	66.6	66.2	49.0	67.2	69.3	77.9	27.4	33.0	44.7

Table 4: Comparison among the maximum softmax probability (MSP), MSP + GAN, and MSP + GAN + OE OOD detectors. The same network architecture is used for all three detectors. All results are percentages and averaged across all $\mathcal{D}_{\text{out}}^{\text{test}}$ datasets.

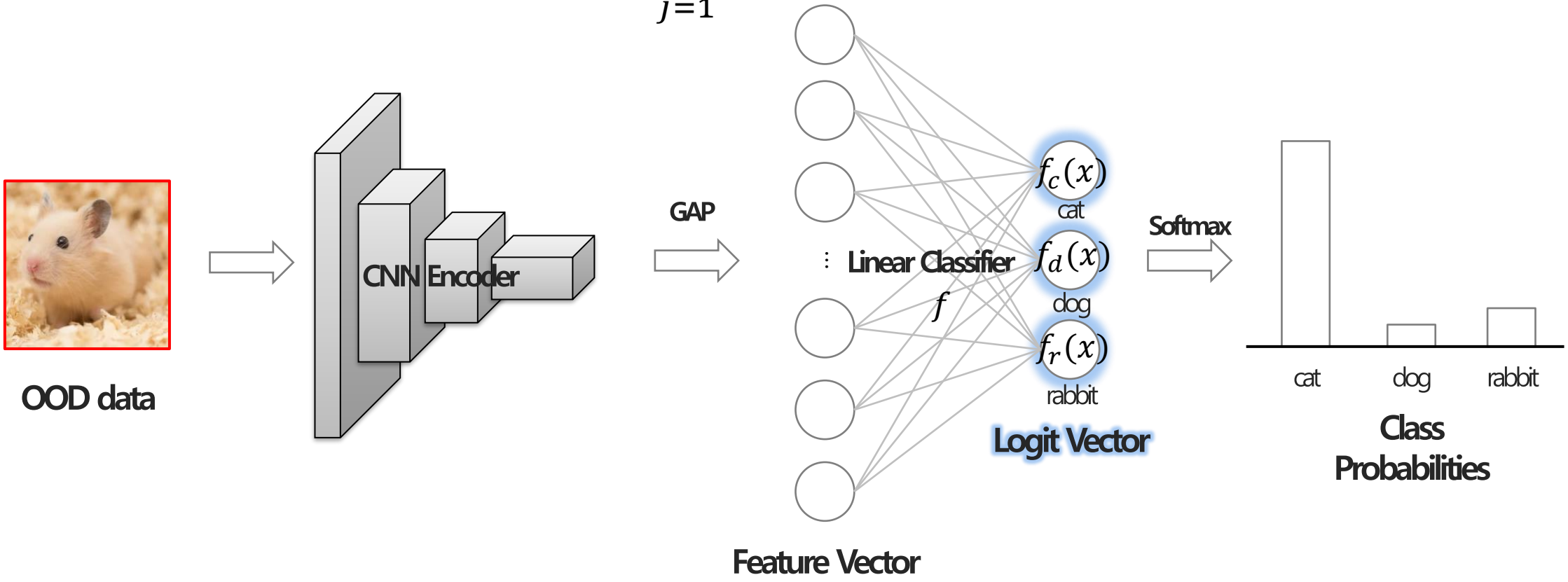
Energy-based Out-of-Distribution Detection

(2020, NeurIPS)

Method 3

Energy-based Out-of-Distribution Detection (2010, NeurIPS)

$$E(x,y) = -f_y(x) \qquad E(x) = -T * \log \sum_{j=1}^{K} e^{f_j(x)/T} \qquad g(x;\tau,f) = \begin{cases} OOD & if \ -E(x;f) \leq \tau, \\ ID & if \ -E(x;f) > \tau \end{cases}$$



Method 3

Energy-based Out-of-Distribution Detection (2010, NeurIPS)

❖ Softmax score를 통해서 negative log likelihood를 최소화하는 과정은 energy surface를 학습하는 과정과 동일

$$\because E(x,y) = -f_y(x)$$

$$L_{nll} = -\log p(y|x) = \mathbb{E}_{(x,y)\sim P_{in}}\left[-\log \frac{e^{f_y(x)/T}}{\sum_{j=1}^{K} e^{f_j(x)/T}}\right]$$

다른 페어는 에너지를 높게

$$= \mathbb{E}_{(x,y)\sim P_{in}}\left[\frac{1}{T}E(x,y) + \log \sum_{j=1}^{K} e^{-E(x,j)/T}\right]$$

정답 페어는 에너지를 낮게

Classification model

||

Energy-based model

Method 3

Energy-based Out-of-Distribution Detection (2010, NeurIPS)

❖ 사용 가능한 OOD 데이터셋을 통해서 OOD 샘플에 대한 에너지를 명시적으로 낮추는 손실 함수 제안

$$\min_{\theta} Loss = L_{nll} + \lambda * L_{energy}$$

ID는 에너지를 m_{in} 보다 낮도록 학습

$$where, L_{energy} = \mathbb{E}_{(x_{in},y)\sim D_{in}^{train}}[\max(0, E(x_{in}) - m_{in})]^2$$

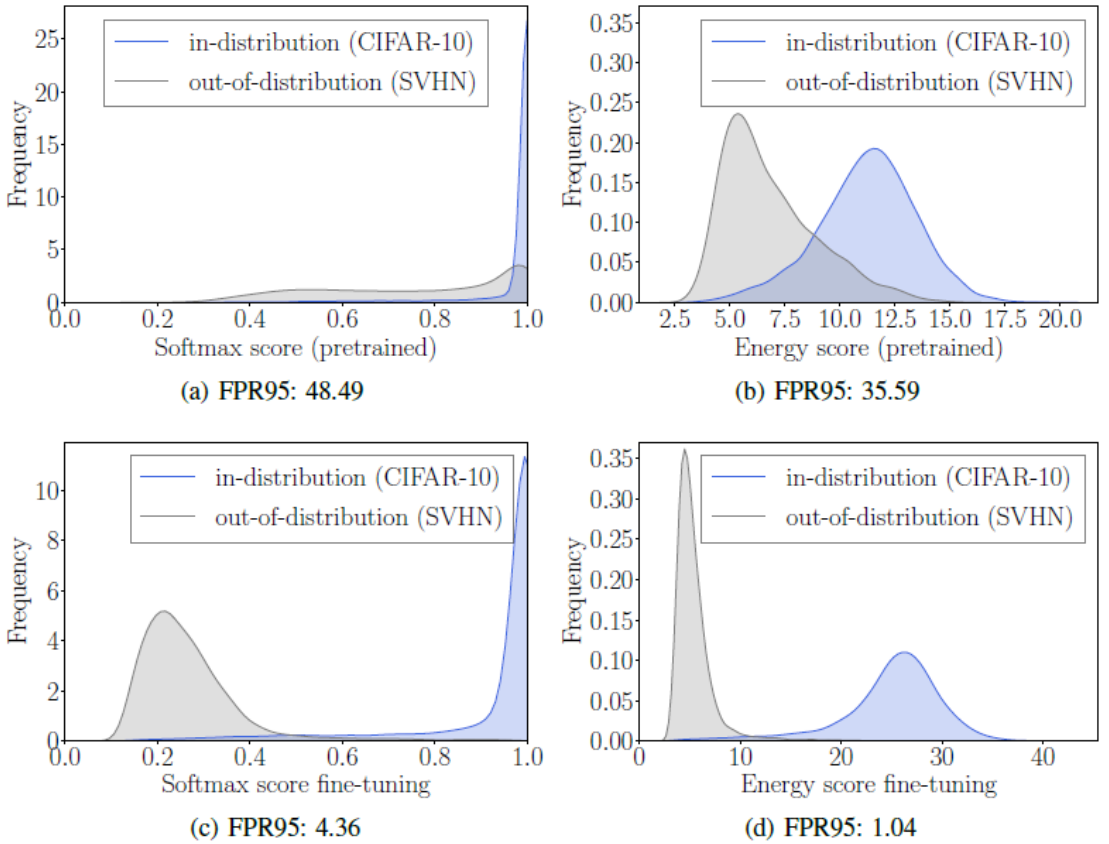
$$+\mathbb{E}_{(x_{out})\sim D_{out}^{train}}[\max(0, m_{out} - E(x_{out}))]^2$$

OOD는 에너지를 m_{out} 보다 높도록 학습

Method 3

Energy-based Out-of-Distribution Detection (2010, NeurIPS)

$\mathcal{D}_{\text{in}}^{\text{test}}$	Method	FPR95 ↓	AUROC ↑	AUPR ↑	In-dist Test Error ↓
CIFAR-10 (WideResNet)	Softmax score [13]	51.04	90.90	97.92	5.16
	Energy score (ours)	33.01	91.88	97.83	5.16
	ODIN [24]	35.71	91.09	97.62	5.16
	Mahalanobis [23]	37.08	93.27	98.49	5.16
	OE [14]	8.53	98.30	99.63	5.32
	Energy fine-tuning (ours)	**3.32**	**98.92**	**99.75**	**4.87**
CIFAR-100 (WideResNet)	Softmax score [13]	80.41	75.53	93.93	**24.04**
	Energy score (ours)	73.60	79.56	94.87	24.04
	ODIN [24]	74.64	77.43	94.23	24.04
	Mahalanobis [23]	54.04	84.12	95.88	24.04
	OE [14]	58.10	85.19	96.40	24.30
	Energy fine-tuning (ours)	**47.55**	**88.46**	**97.10**	24.58

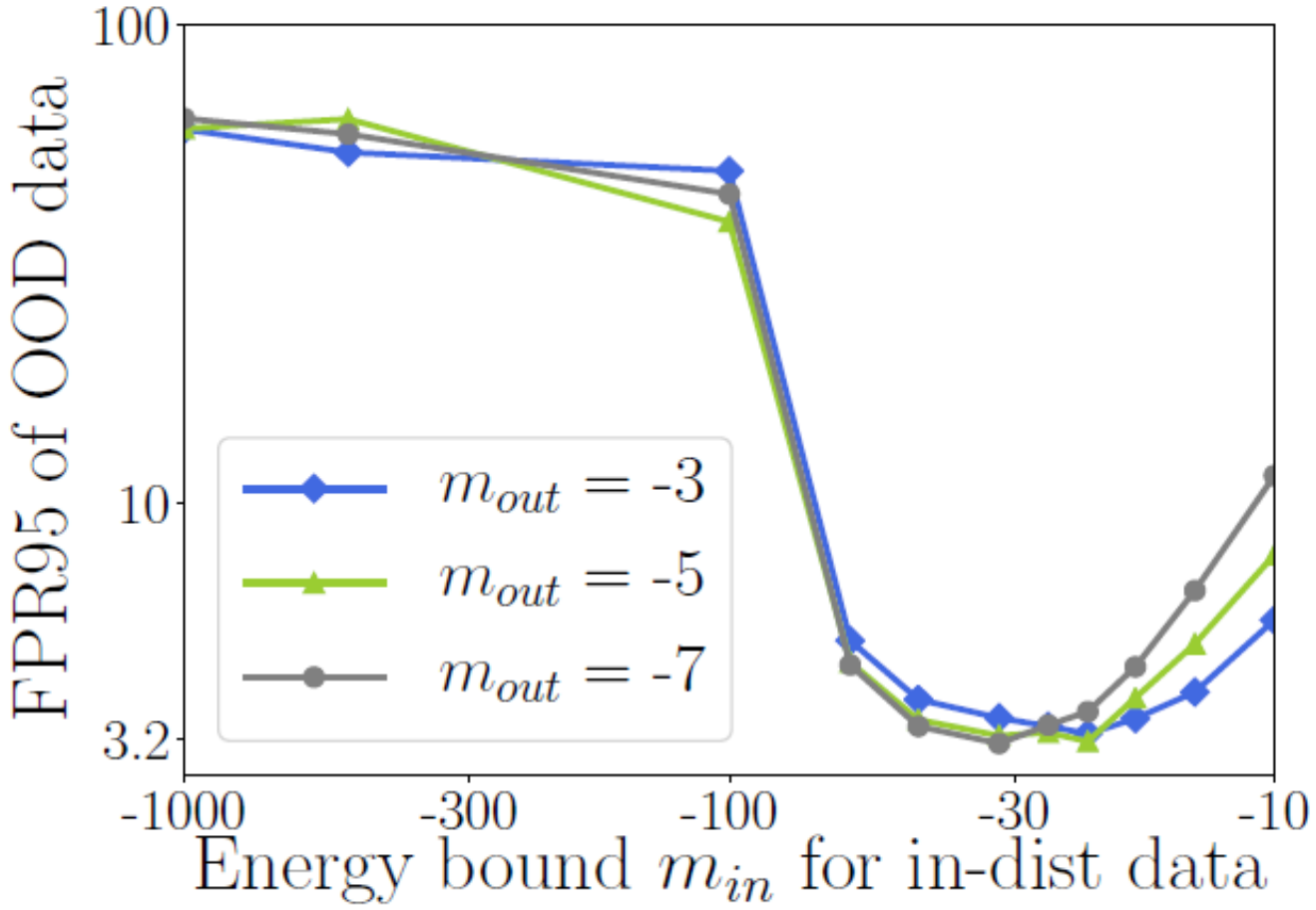


(a) FPR95: 48.49 (b) FPR95: 35.59

(c) FPR95: 4.36 (d) FPR95: 1.04

Method 3

Energy-based Out-of-Distribution Detection (2010, NeurIPS)

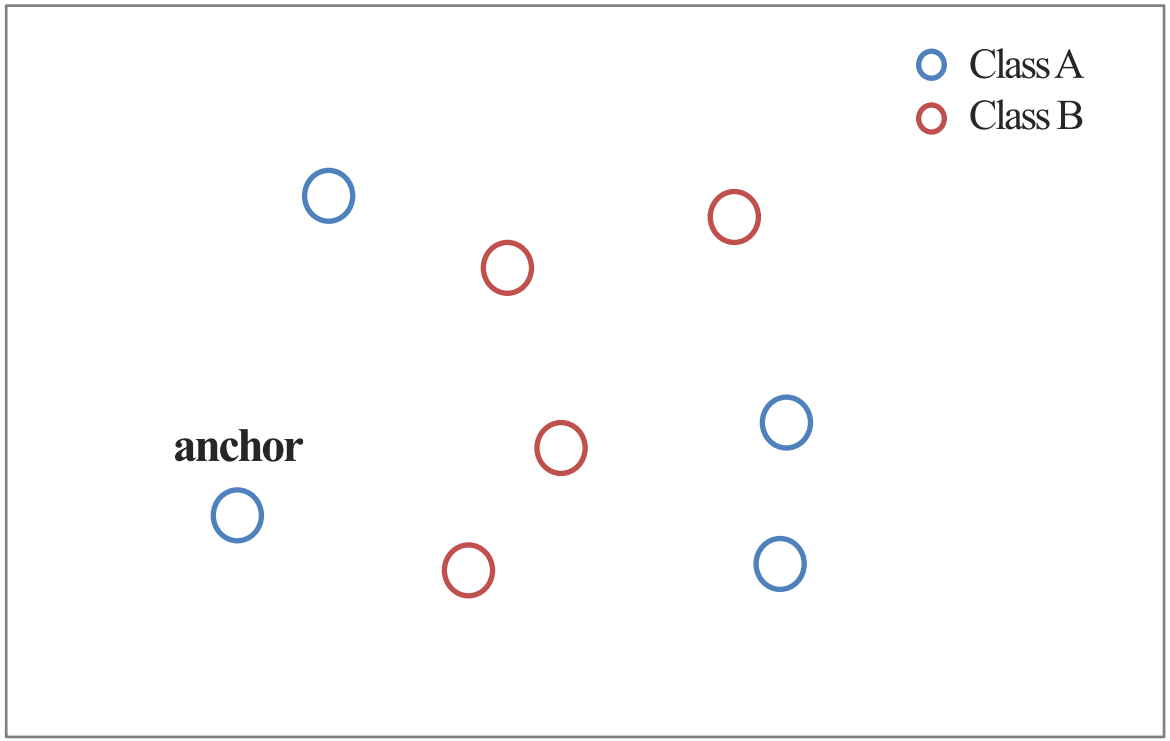


CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

- Contrastive Learning은 Metric Learning 방법론 중 하나로 데이터 간 유사도 정보를 통해서 데이터들을 구분하기 쉽게 해주는 거리 함수를 학습하는 것
- **Anchor**를 기준으로 **Positive samples**는 가깝도록, **Negative samples**는 멀도록 학습



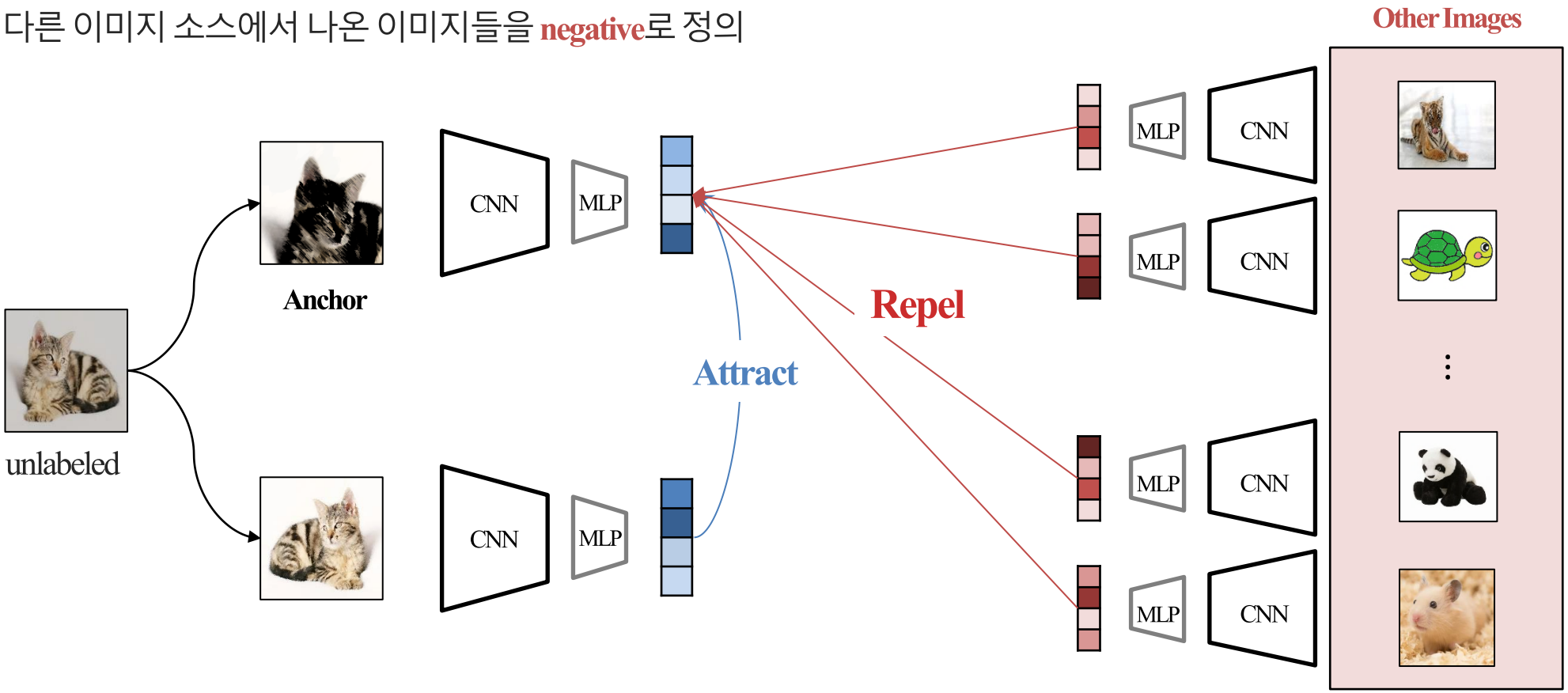
Feature Space

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Self-supervised Contrastive Learning

- 이미지 증강 기법을 통해서 같은 소스에서 나온 이미지들을 **positive**로 정의
- 다른 이미지 소스에서 나온 이미지들을 **negative**로 정의

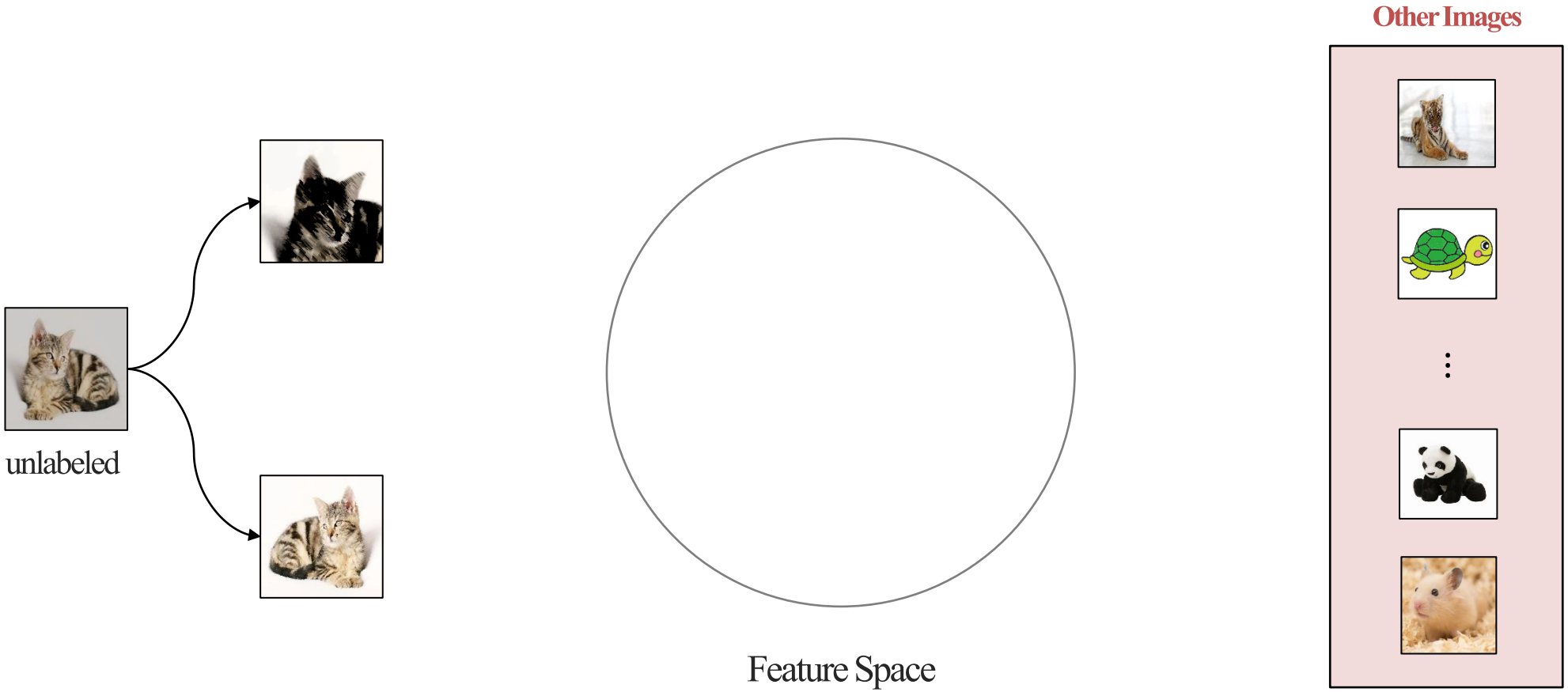


Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Self-supervised Contrastive Learning

- Self-supervised contrastive learning을 통해 학습하면 discriminative feature vector를 학습할 수 있음

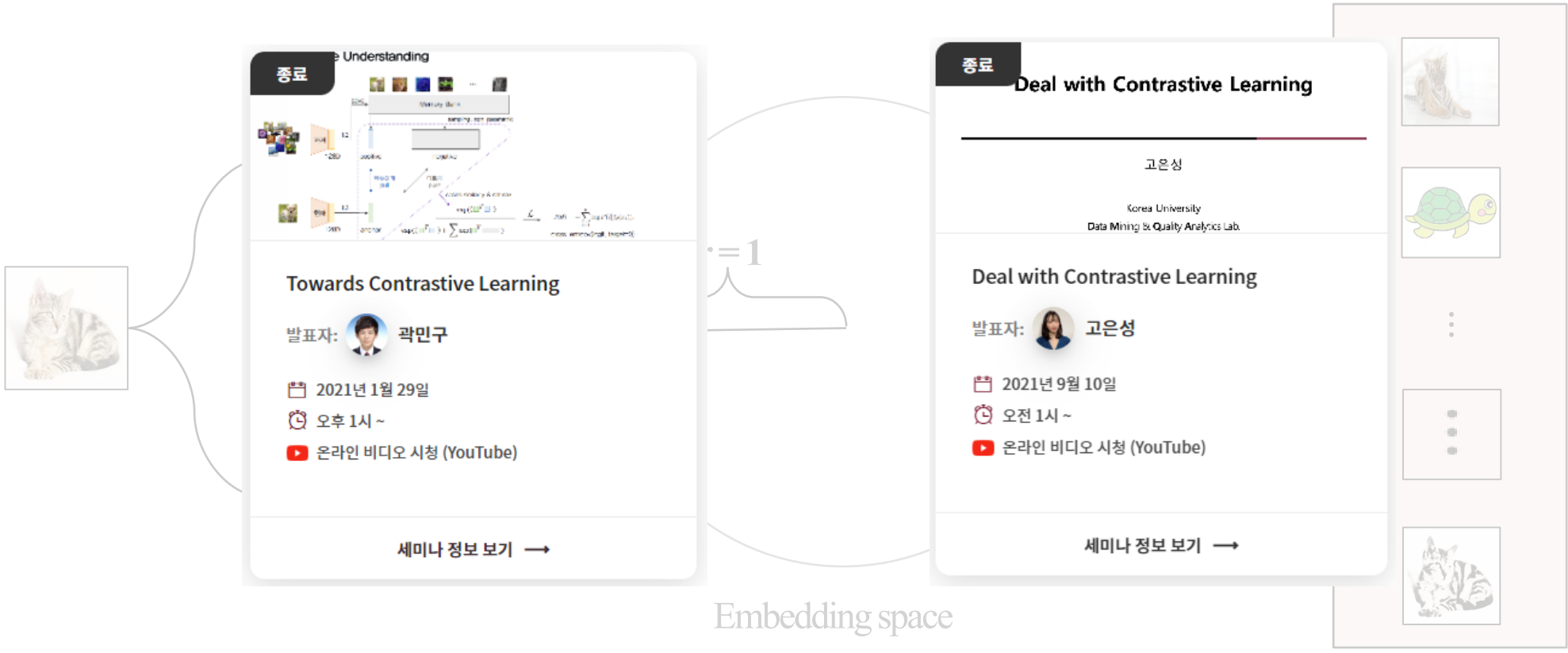


Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Self-supervised Contrastive Learning

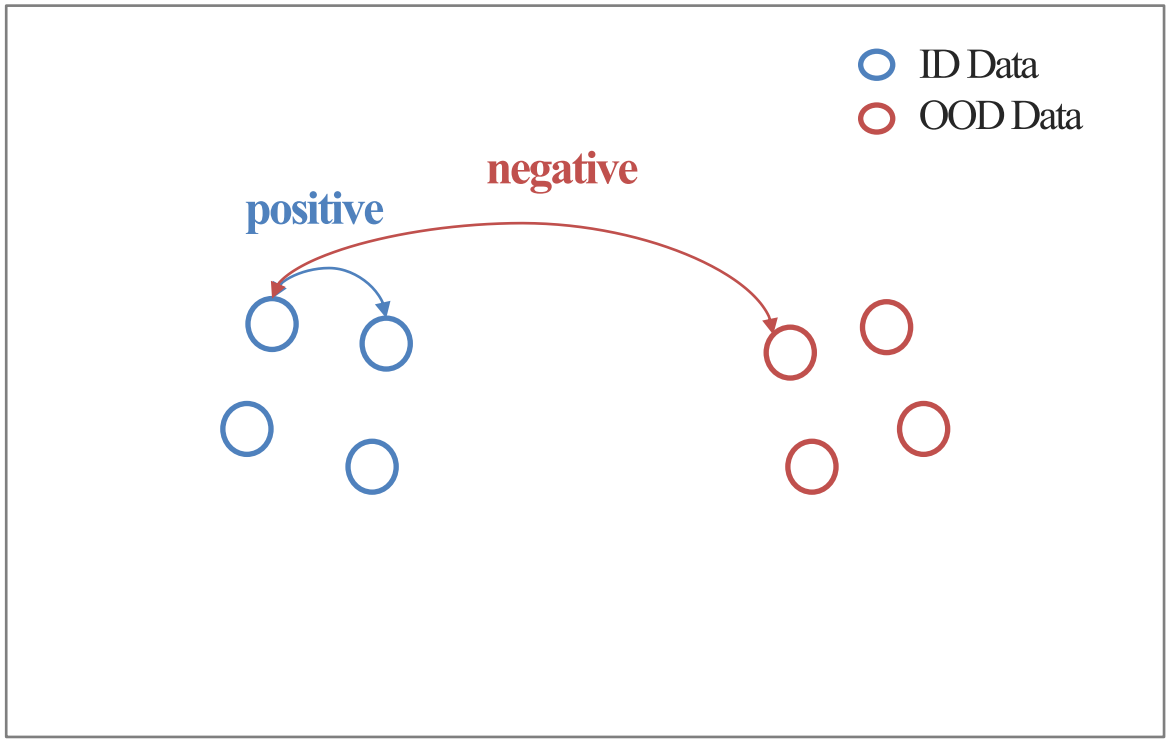
- 이미지 증강 기법을 통해서 같은 소스에서 나온 이미지들을 positive로 정의
- 다른 이미지 소스에서 나온 이미지들을 negative로 정의하며 false negative 영향을 줄이기 위해 개수를 늘림



Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ ID 데이터와 OOD 데이터를 negative로 설정하여 contrastive learning을 진행하면 ID와 OOD가 분리된 feature space 학습



Method 4

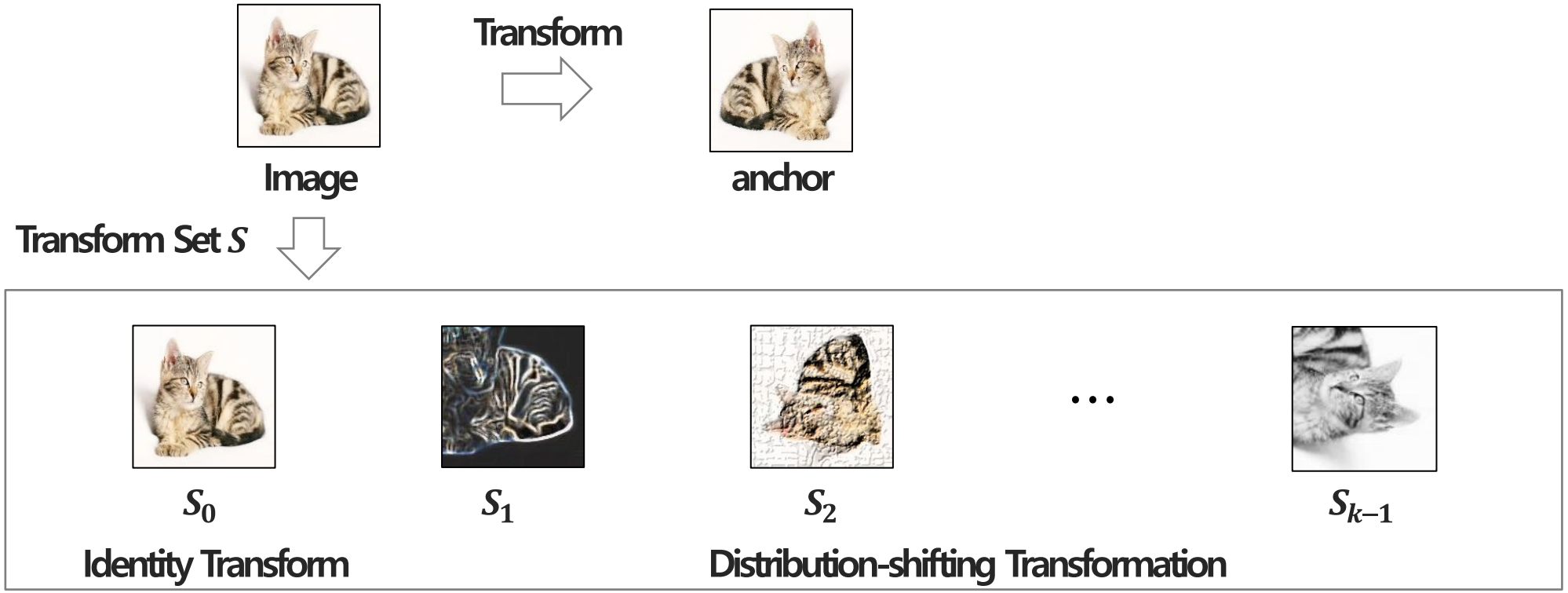
CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ 매우 강한 이미지 변형 기법을 적용하여 ID 데이터와 매우 유사한 OOD 데이터 생성

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

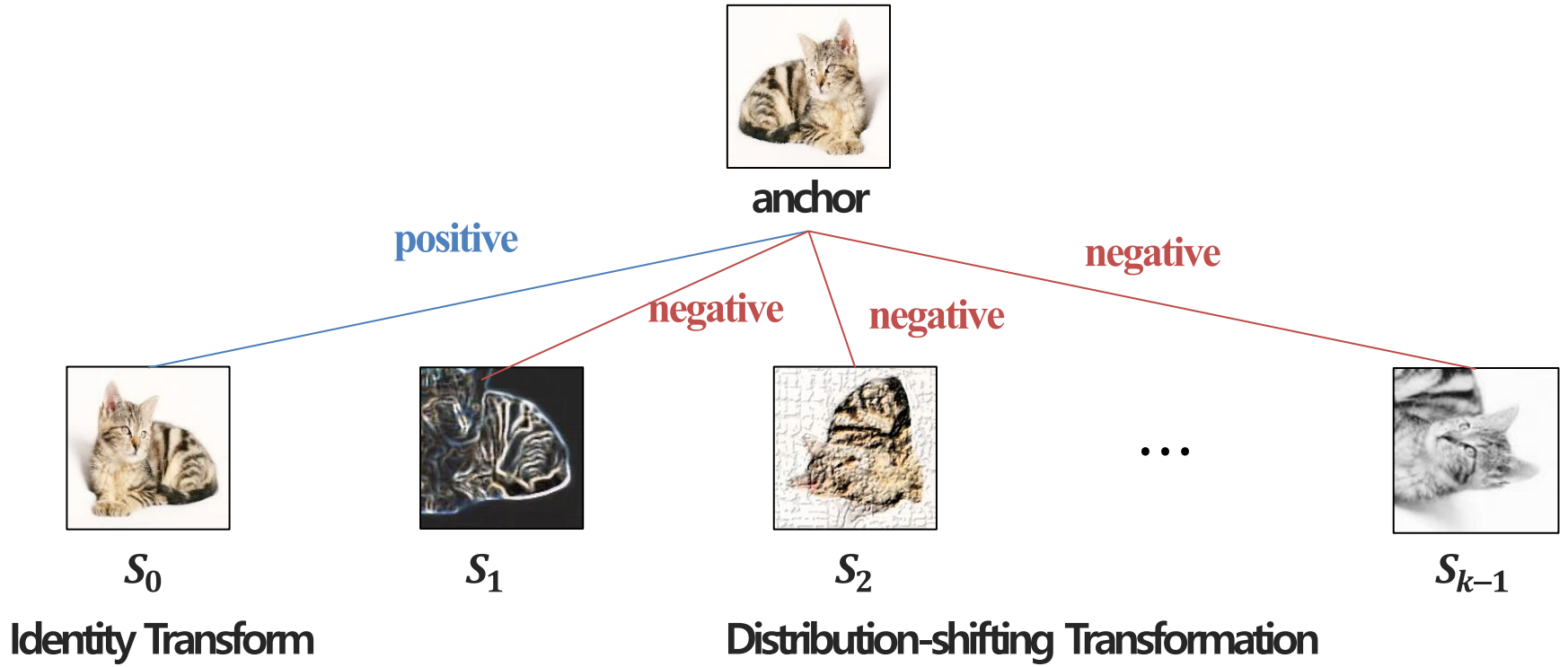
- ❖ 같은 이미지 소스에서 나온 샘플들을 positive로 정의하는 SimCLR과 다르게 negative로 정의
- ❖ Identity transform에서 나온 이미지를 positive pair로 설정
- ❖ K-1개의 Distribution-shifting Transformation에서 나온 K-1개의 OOD-like data를 negative samples로 설정



Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

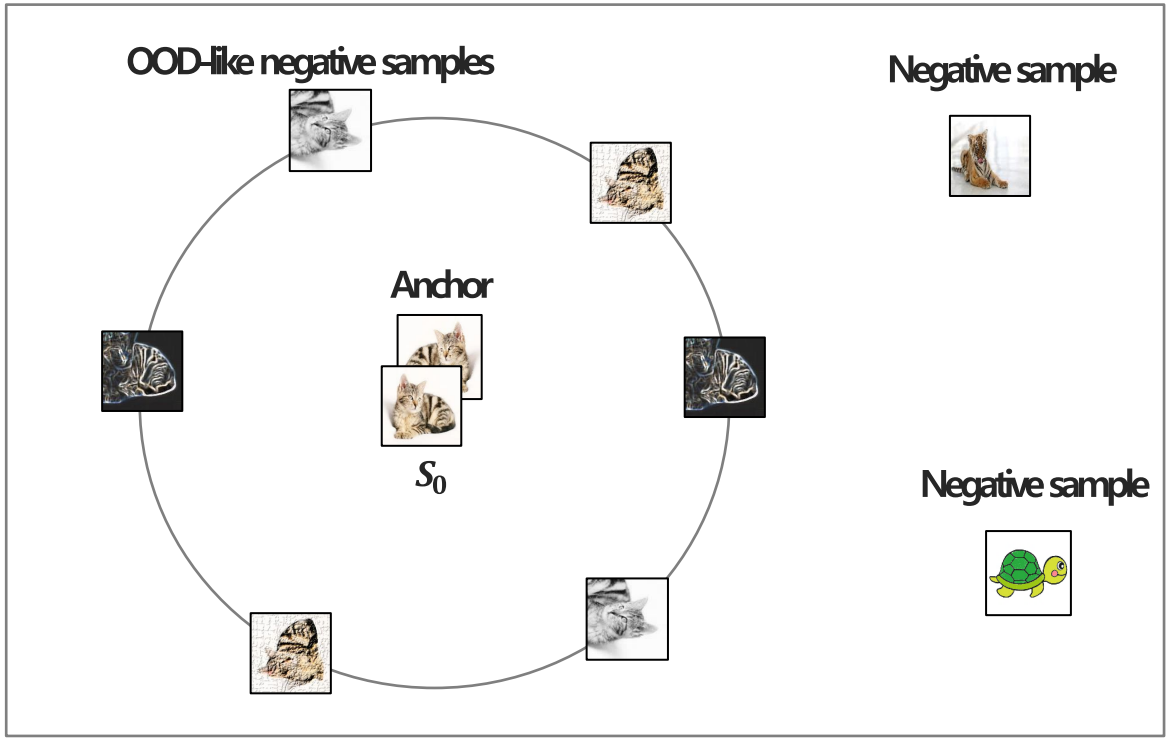
- 같은 이미지 소스에서 나온 샘플들을 positive로 정의하는 SimCLR과 다르게 negative로 정의
- Identity transform에서 나온 이미지를 positive pair로 설정
- K-1개의 Distribution-shifting Transformation에서 나온 K-1개의 OOD-like data를 negative samples로 설정



Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

- 같은 이미지 소스에서 나온 샘플들을 positive로 정의하는 SimCLR과 다르게 negative로 정의
- Identity transform에서 나온 이미지를 positive pair로 설정
- K-1개의 Distribution-shifting Transformation에서 나온 K-1개의 OOD-like data를 negative samples로 설정

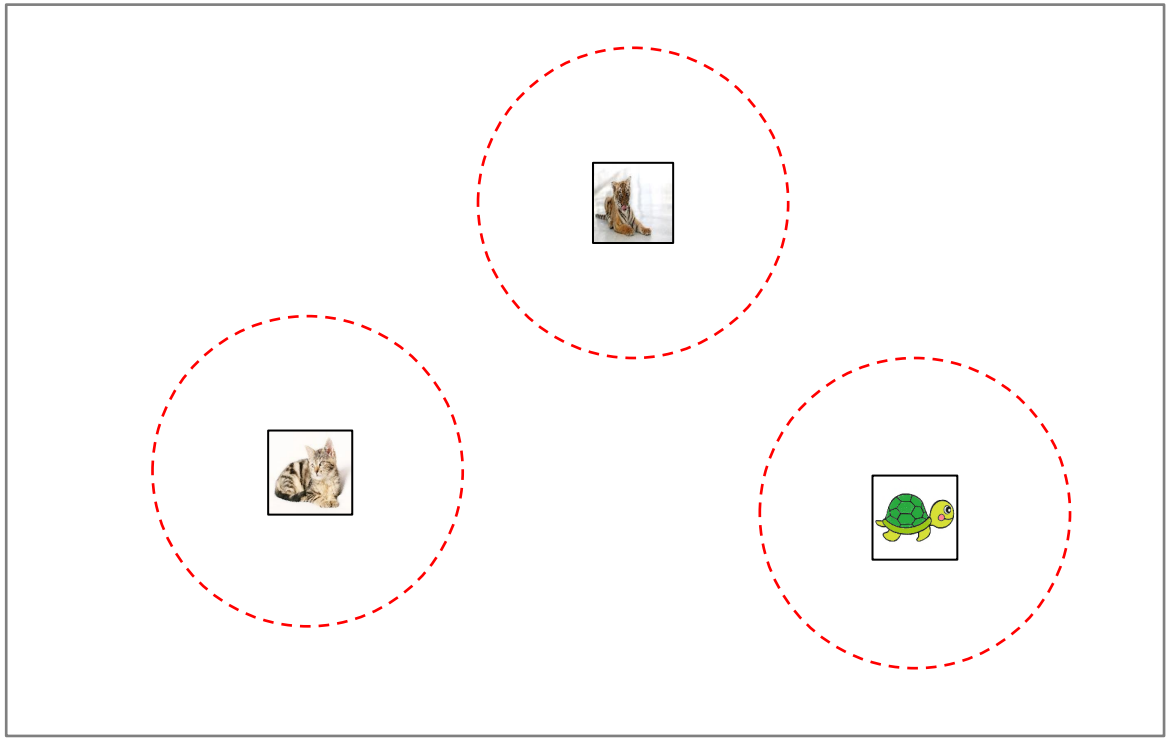


Feature Space

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

- 같은 이미지 소스에서 나온 샘플들을 positive로 정의하는 SimCLR과 다르게 negative로 정의
- Identity transform에서 나온 이미지를 positive pair로 설정
- K-1개의 Distribution-shifting Transformation에서 나온 K-1개의 OOD-like data를 negative samples로 설정

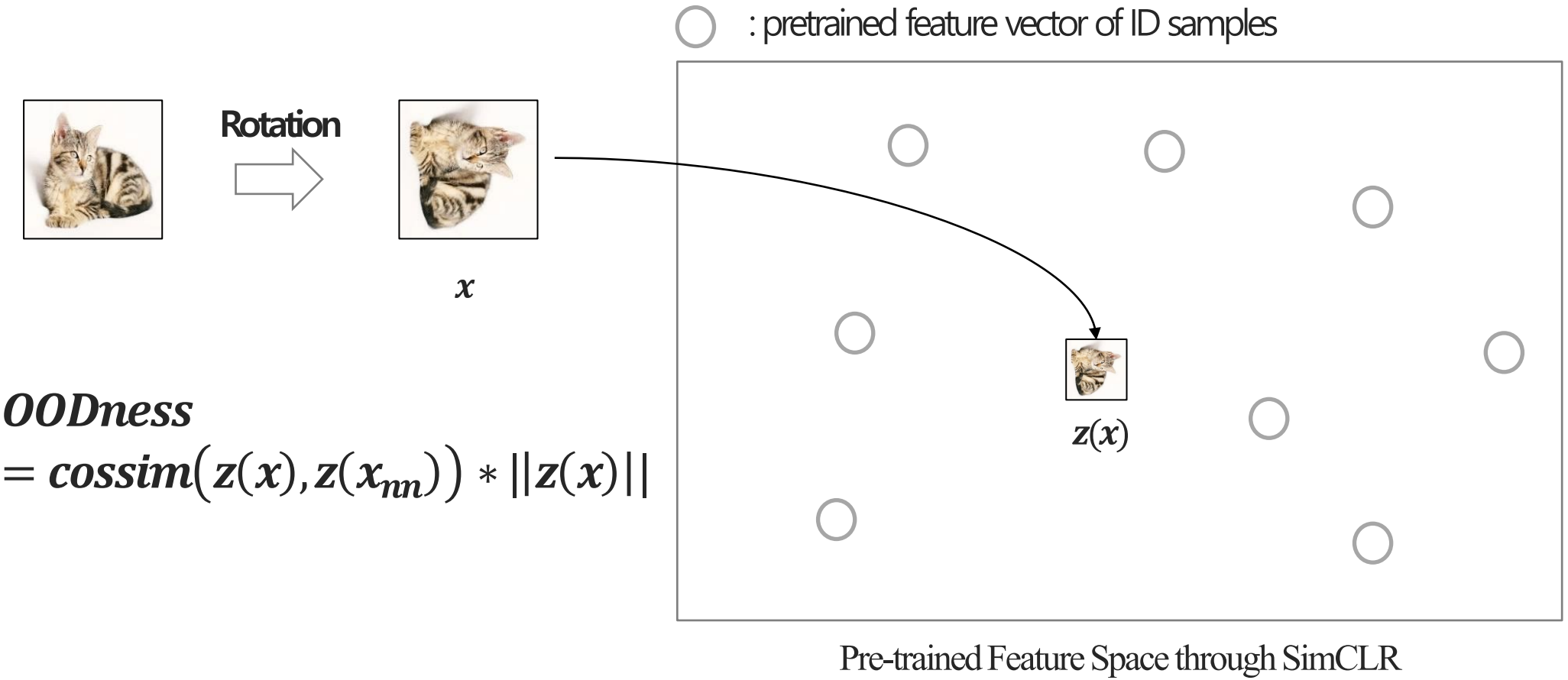


Feature Space

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Distribution shifting transformation을 찾기 위해 변형된 이미지와 가장 가까운 ID 샘플에 대한 OODness를 계산

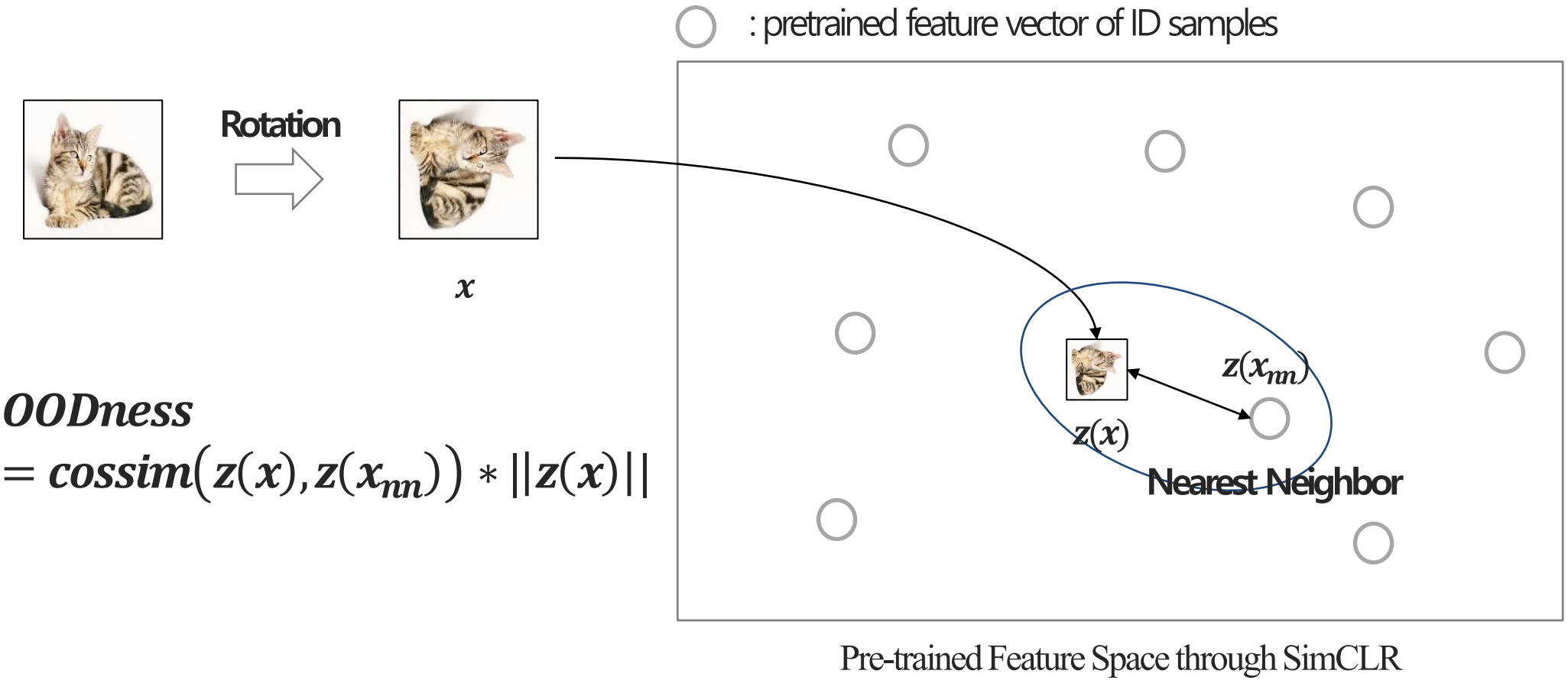


$$OODness = cossim\big(z(x), z(x_{nn})\big) * ||z(x)||$$

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Distribution shifting transformation을 찾기 위해 변형된 이미지와 가장 가까운 ID 샘플에 대한 OODness를 계산



$$OODness = cossim\big(z(x), z(x_{nn})\big) * ||z(x)||$$

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ Distribution shifting transformation을 찾기 위해 변형된 이미지와 가장 가까운 ID 샘플에 대한 OODness를 계산

(a) Original (b) Cutout (c) Sobel (d) Noise (e) Blur (f) Perm (g) Rotate

Figure 1: Visualization of the original image and the considered shifting transformations.

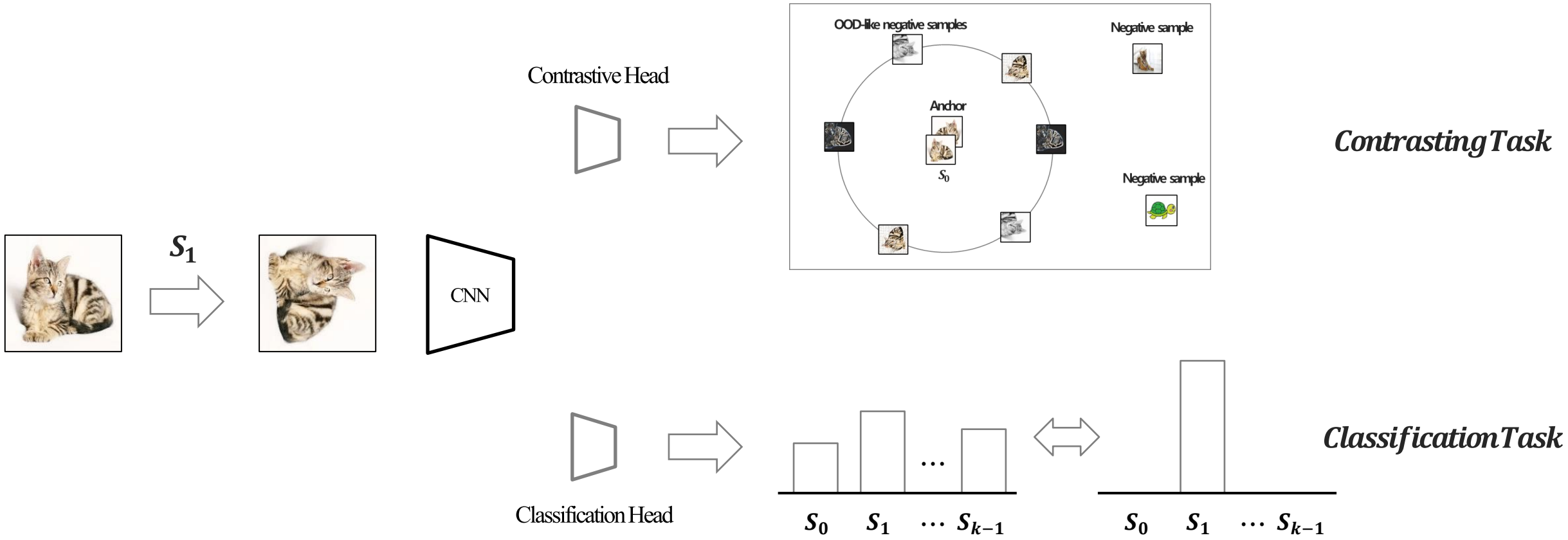
Table 4: OOD-ness (%), *i.e.*, the AUROC between in-distribution vs. transformed samples under the vanilla SimCLR (see Section 2.2), of various transformations. The vanilla SimCLR is trained on one-class CIFAR-10 under ResNet-18. Each column denotes the applied transformation.

	Cutout	Sobel	Noise	Blur	Perm	Rotate
OOD-ness	79.5	69.2	74.4	76.0	83.8	85.2

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

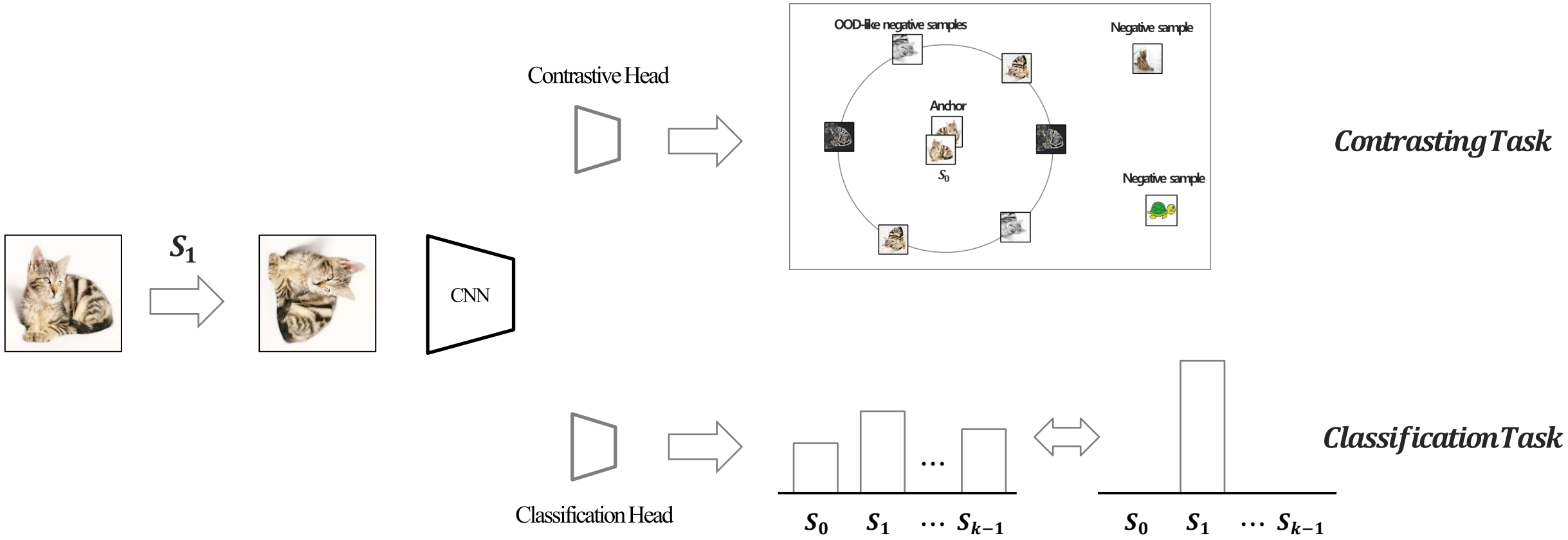
❖ 또한, 저자들은 성능을 더 향상시키기 위해서 transformation classificastion task를 추가



Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ 또한, 저자들은 성능을 더 향상시키기 위해서 transformation classificastion task를 추가



Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

❖ 각 태스크에 맞는 OOD score를 제안하고 이를 합하여 최종적인 OOD score로 활용

$$\boldsymbol{OODness} = \boldsymbol{s_{con}}(\boldsymbol{x}, \boldsymbol{x_{nn}}) = \boldsymbol{cossim}\big(\boldsymbol{z}(\boldsymbol{x}), \boldsymbol{z}(\boldsymbol{x_{nn}})\big) * ||\boldsymbol{z}(\boldsymbol{x})||$$

$$\boldsymbol{s_{con-SI}}(\boldsymbol{x}, \boldsymbol{x_{nn}}) = \sum_{T \in \boldsymbol{S}} \boldsymbol{\lambda_T^{con}} * \boldsymbol{s_{con}}\big(T(\boldsymbol{x}), T(\boldsymbol{x_{nn}})\big)$$

$$\boldsymbol{s_{cls-SI}}(\boldsymbol{x}) = \sum_{T \in \boldsymbol{S}} \boldsymbol{\lambda_T^{cls}} * \boldsymbol{W}_T \boldsymbol{z}_{T(\boldsymbol{x})}$$

$$\boldsymbol{s_{CSI}}(\boldsymbol{x}, \boldsymbol{x_{nn}}) = \boldsymbol{s_{con-SI}}(\boldsymbol{x}, \boldsymbol{x_{nn}}) + \boldsymbol{s_{cls-SI}}(\boldsymbol{x})$$

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

Table 3: Test accuracy (%), ECE (%), and AUROC (%) of confidence-calibrated classifiers trained on labeled (a) CIFAR-10 and (b) ImageNet-30. The reported results are averaged over five trials for CIFAR-10 and one trial for ImageNet-30. Subscripts denote standard deviation, and bold denote the best results. CSI-ens denotes the ensembled prediction, *i.e.*, 4 times slower (as we use rotation).

(a) Labeled CIFAR-10

			CIFAR10 →						
Train method	Test acc.	ECE	SVHN	LSUN	ImageNet	LSUN (FIX)	ImageNet (FIX)	CIFAR100	Interp.
Cross Entropy	93.0±0.2	6.44±0.2	88.6±0.9	90.7±0.5	88.3±0.6	87.5±0.3	87.4±0.3	85.8±0.3	75.4±0.7
SupCLR [30]	93.8±0.1	5.56±0.1	97.3±0.1	92.8±0.5	91.4±1.2	91.6±1.5	90.5±0.5	88.6±0.2	75.7±0.1
CSI (ours)	94.8±0.1	4.40±0.1	96.5±0.2	96.3±0.5	96.2±0.4	92.1±0.5	92.4±0.0	90.5±0.1	78.5±0.2
CSI-ens (ours)	**96.1**±0.1	**3.50**±0.1	**97.9**±0.1	**97.7**±0.4	**97.6**±0.3	**93.5**±0.4	**94.0**±0.1	**92.2**±0.1	**80.1**±0.3

(b) Labeled ImageNet-30

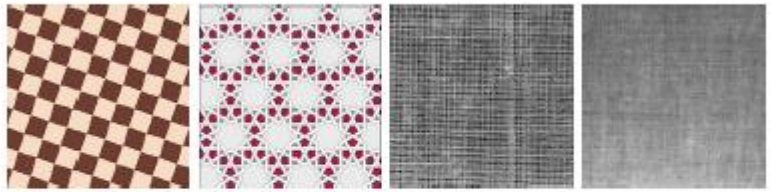
			ImageNet-30 →							
Train method	Test acc.	ECE	CUB-200	Dogs	Pets	Flowers	Food-101	Places-365	Caltech-256	DTD
Cross Entropy	94.3	5.08	88.0	96.7	95.0	89.7	79.8	90.5	90.6	90.1
SupCLR [30]	96.9	3.12	86.3	95.6	94.2	92.2	81.2	89.7	90.2	92.1
CSI (ours)	97.0	2.61	93.4	97.7	96.9	96.0	87.0	92.5	91.9	93.7
CSI-ens (ours)	**97.8**	**2.19**	**94.6**	**98.3**	**97.4**	**96.2**	**88.9**	**94.0**	**93.2**	**97.4**

Method 4

CSI: Novelty Detection via Contrastive Learning on Distributionally Shifted Instances (2020, NeurIPS)

(a) Add transformations								(b) Remove transformations			
Base		Cutout	Sobel	Noise	Blur	Perm	Rotate		Crop	Jitter	Gray
87.9	+Align	84.3	85.0	85.5	88.0	73.1	76.5	-Align	55.7	78.8	78.4
	+Shift	88.5	88.3	89.3	89.2	90.7	94.3	+Shift	-	-	88.3

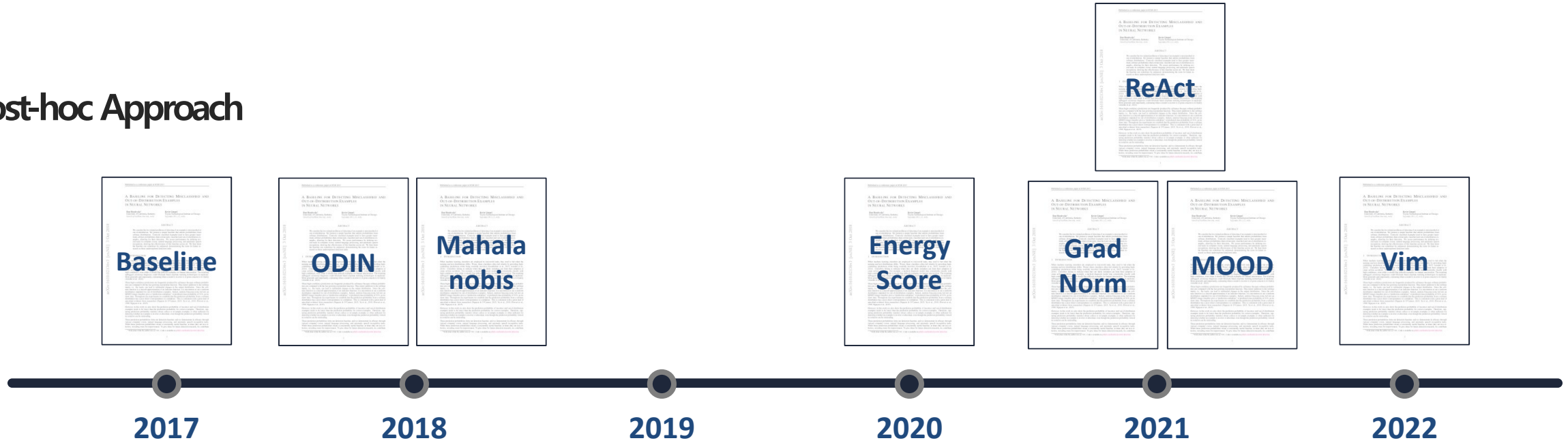
Table 6: OOD-ness (%) and AUROC (%) on DTD, where Textile is used for OOD.



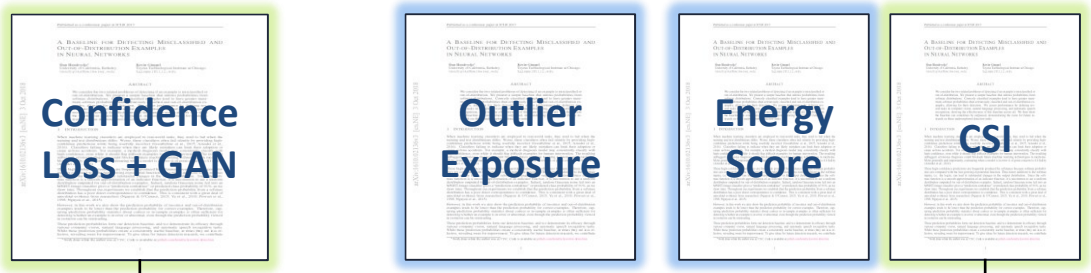
(a) OOD-ness		(b) AUROC		
Rot.	Noise	Base	CSI(R)	CSI(N)
50.6	**75.7**	70.3	65.9	**80.1**

Conclusions

Post-hoc Approach



Outlier Exposure Approach



감사합니다