DMQA Open Seminar

Out-Of-Distribution Detection for Image Classification: Part2

2024. 09. 27

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

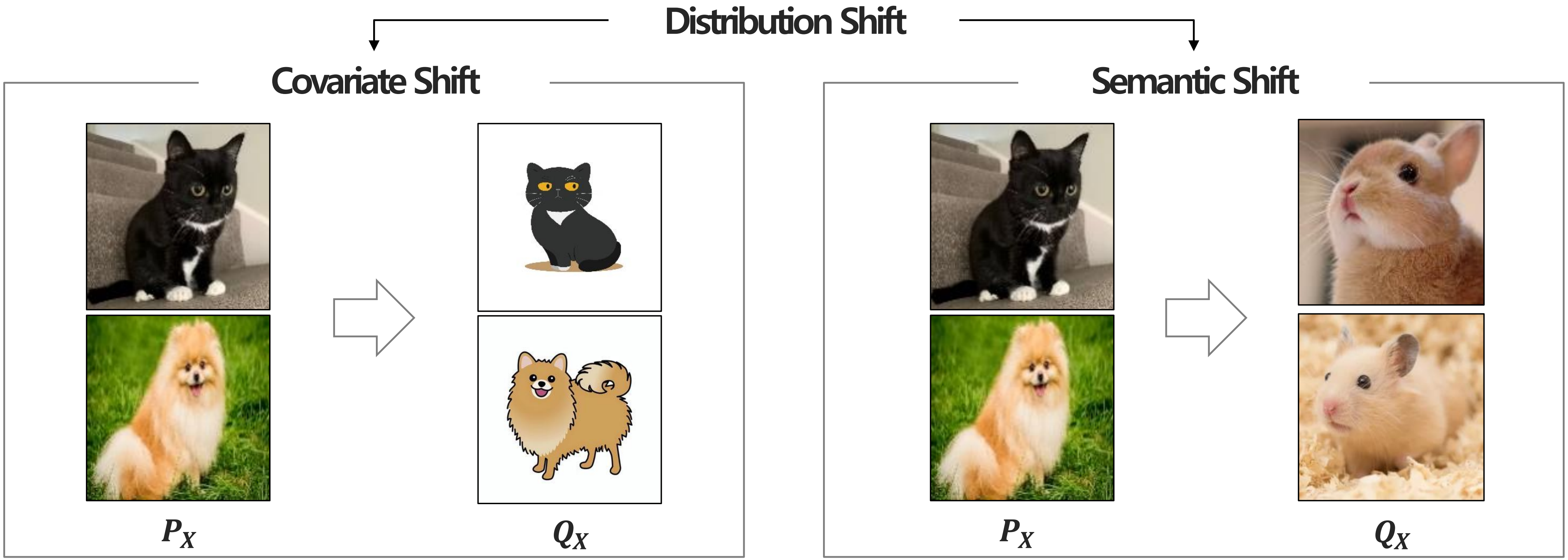
목차

Contents

Part1 Review

Out-of-Distribution: OOD

- ❖ OOD 데이터란 학습에 활용된 입력 데이터 분포 P_X와 다른 분포 Q_X에서 샘플링된 입력 데이터
- ❖ 일반적으로 이미지 분류 문제에서는 학습 데이터에 존재하지 않는 클래스를 가진 이미지를 의미

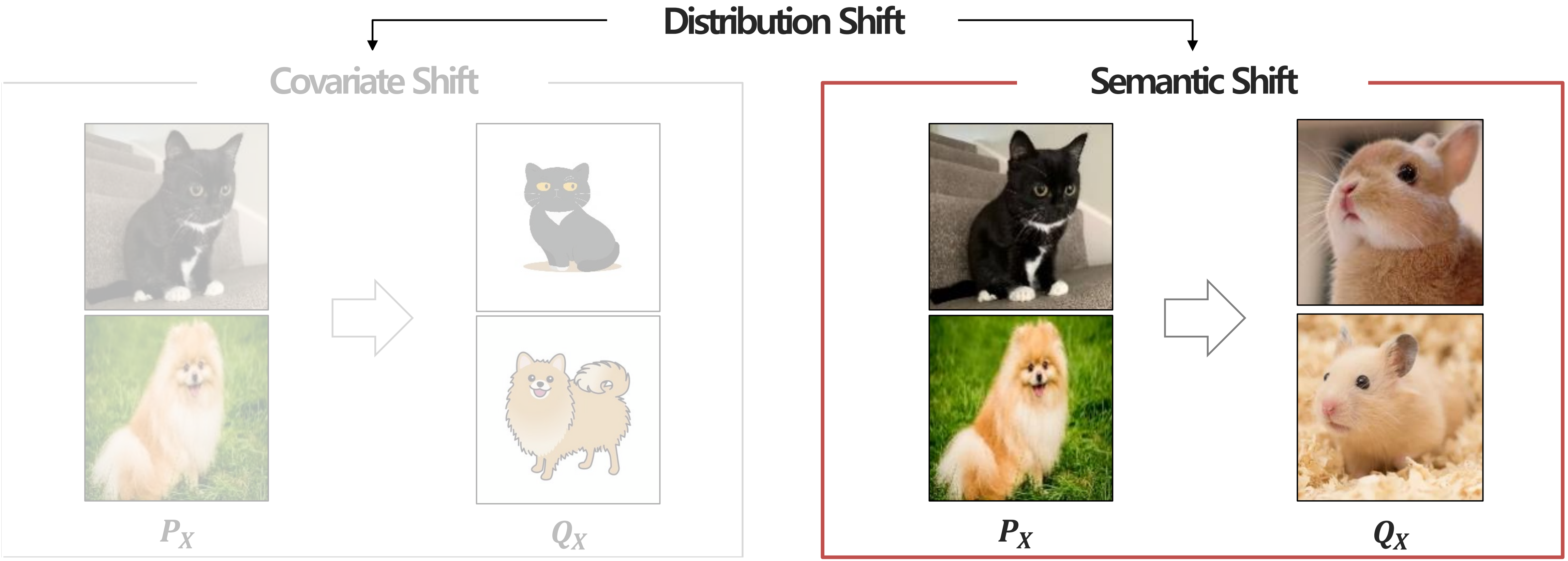


Part1 Review

Out-of-Distribution: OOD

❖ OOD 데이터란 학습에 활용된 입력 데이터 분포 P_X와 다른 분포 Q_X에서 샘플링된 입력 데이터

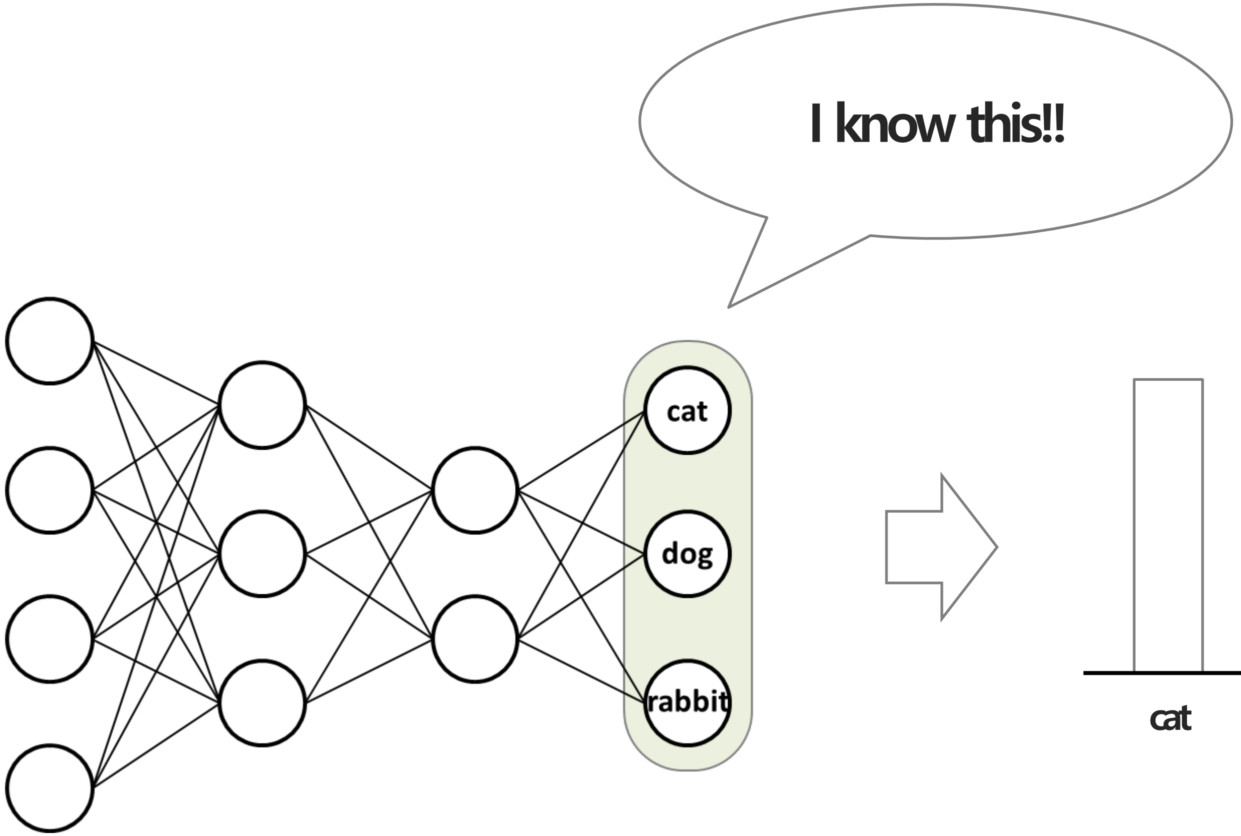
❖ 일반적으로 이미지 분류 문제에서는 학습 데이터에 존재하지 않는 클래스를 가진 이미지를 의미



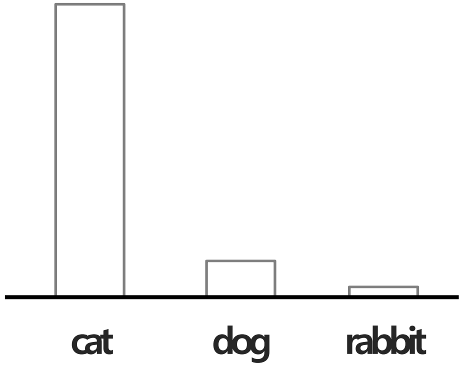
Part1 Review

Overconfidence Problem

In-distribution data



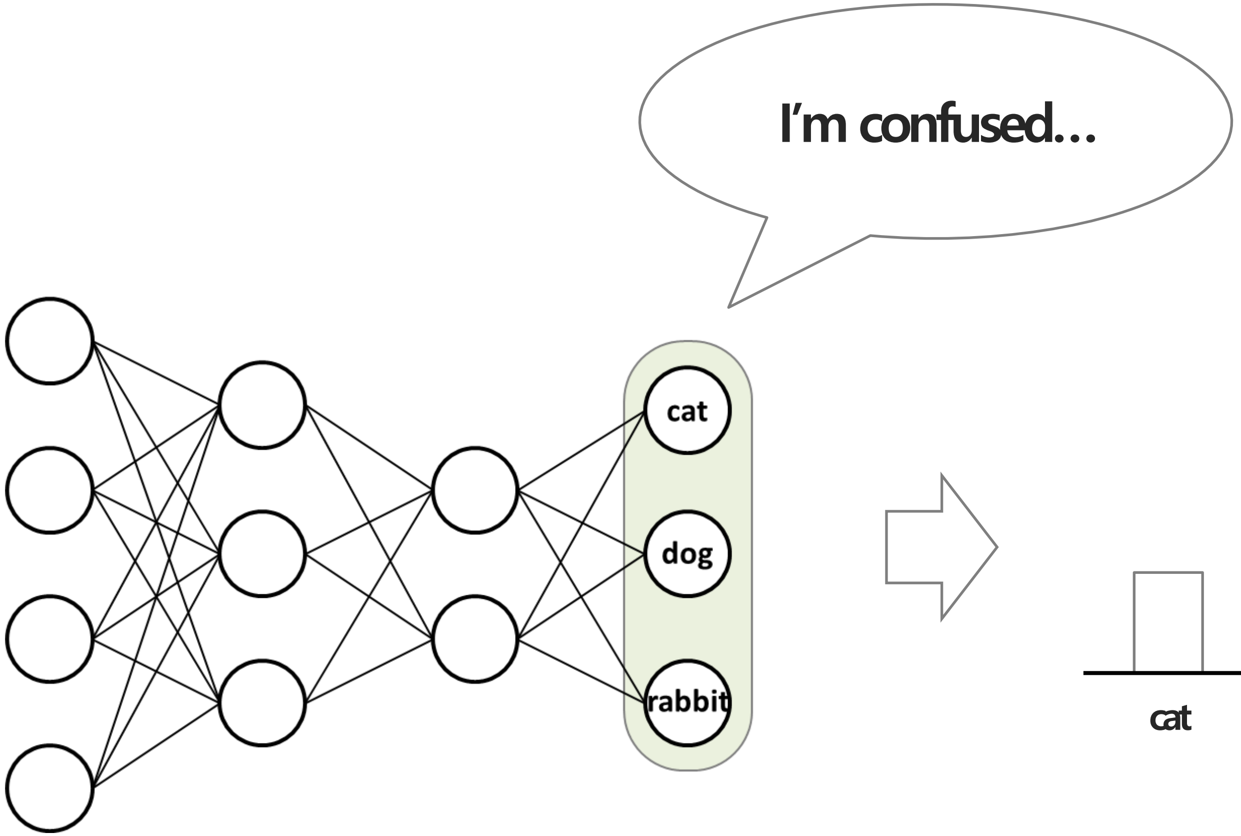
고양이, 강아지, 토끼를
분류하는 모델



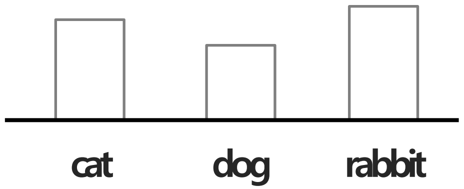
Part1 Review

Overconfidence Problem

Out-of-distribution data



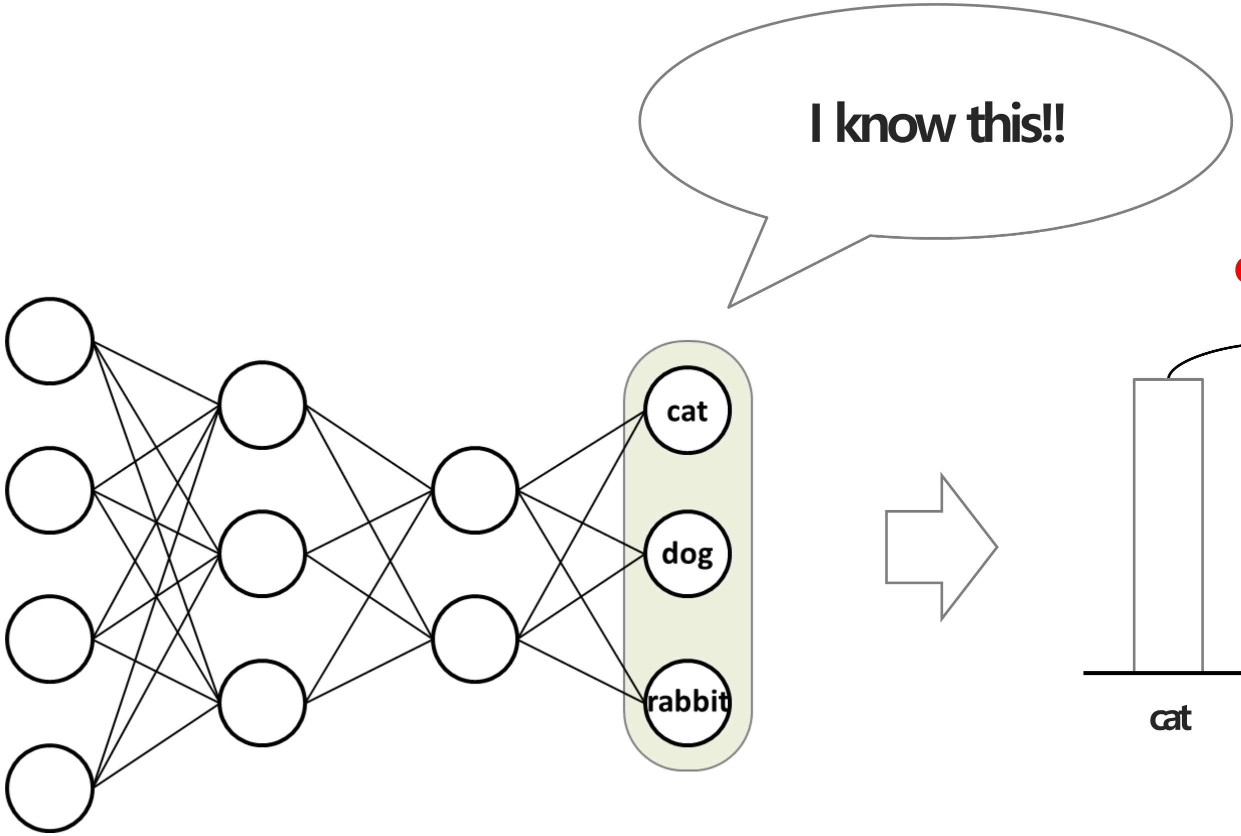
고양이, 강아지, 토끼를
분류하는 모델



Part1 Review

Overconfidence Problem

Out-of-distribution data



고양이, 강아지, 토끼를
분류하는 모델

Part1 Review

Part1

Baseline

OOD 탐지 필요성
제시

이미지 분류 모델이
에너지 모델임을 증명
하고 **에너지 스코어**를
통해 OOD탐지에 활용

Energy Score

Vim

2017

2018

2019

2020

2021

2022

Temperature scaling과
adversarial noise injection
기법을 최대확률값에
적용하여 OOD 탐지 성
능 향상

ODIN

Mahalanobis

특징 공간에서 클래스
prototype과 샘플 간
Mahalanobis distance를
계산하여 OOD 탐지에
활용

ReAct

GradNorm

MOOD

Part1 Review

Part1

Baseline

OOD 탐지 필요성 제시

이미지 분류 모델이 에너지 모델임을 증명하고 **에너지 스코어**를 통해 OOD탐지에 활용

Energy Score

2017

2018

2019

2020

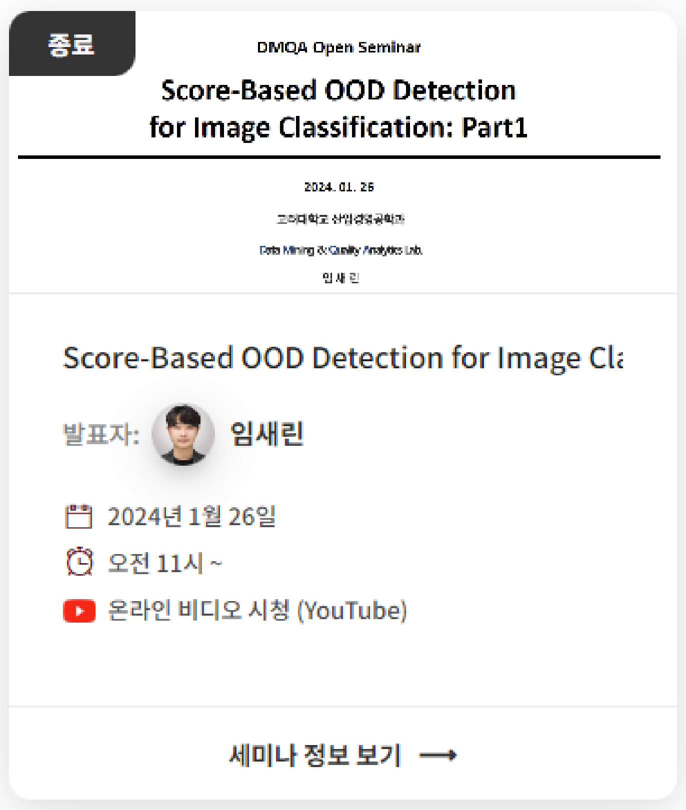
Temperature scaling과 **adversarial noise injection** 기법을 최대확률값에 적용하여 OOD 탐지 성능 향상

ODIN

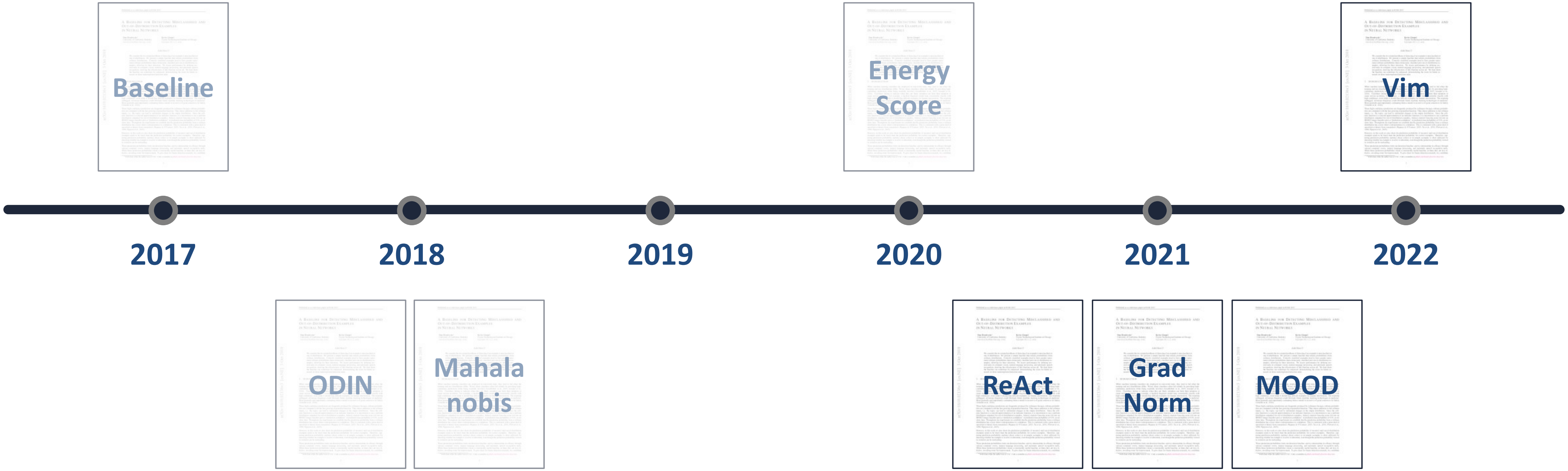
Mahalanobis

특징 공간에서 클래스 prototype과 샘플 간 **Mahalanobis distance**를 계산하여 OOD 탐지에 활용

ReAct



Timeline

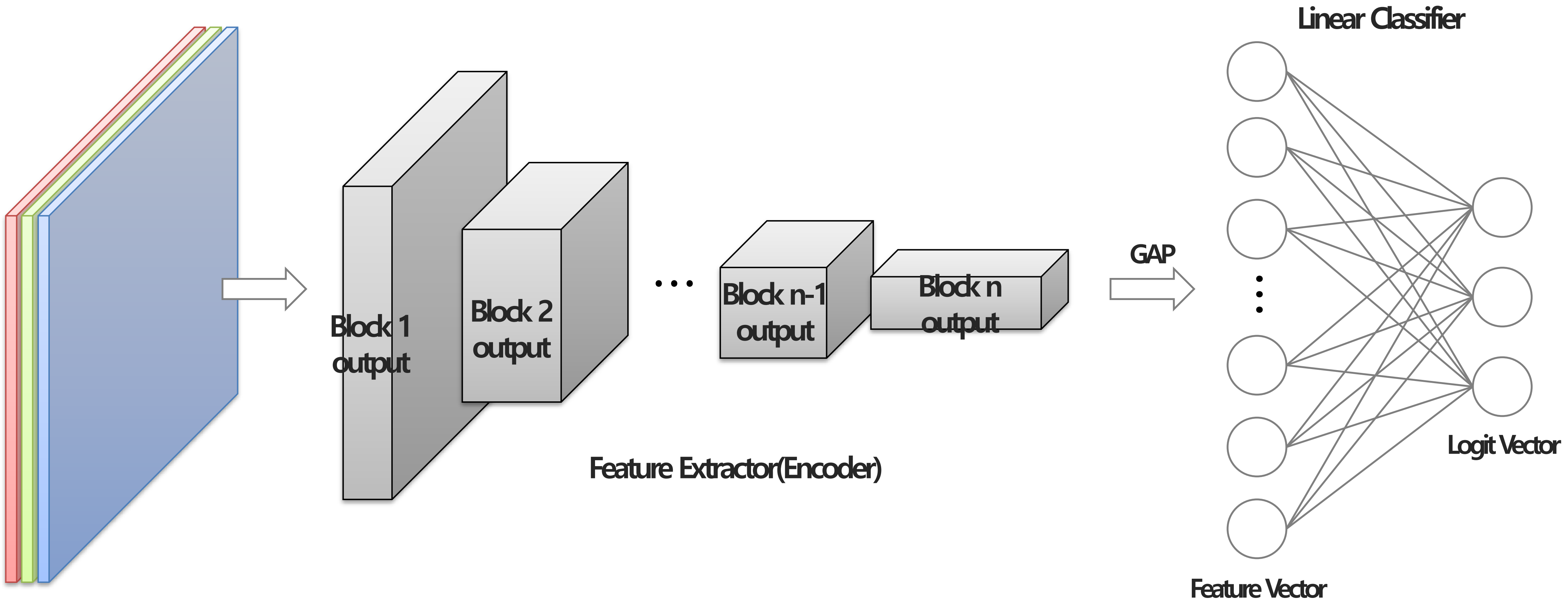

Baseline
Energy Score
Vim
2017
2018
2019
2020
2021
2022
ODIN
Mahalanobis
ReAct
GradNorm
MOOD

ReAct: Out-Of-Distribution Detection with Rectified Activations

(2021, NeurIPS)

Paper 1: ReAct

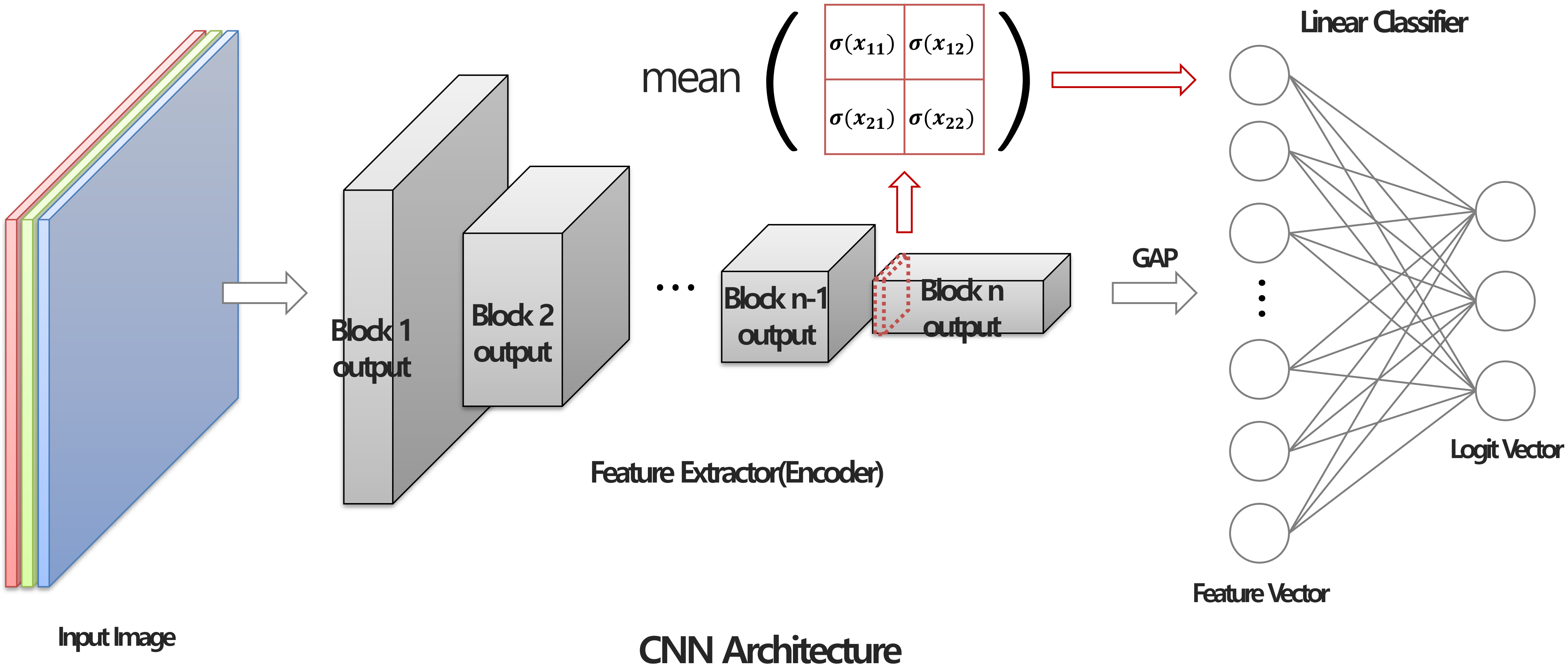
Introduction



CNN Architecture

Paper 1: ReAct

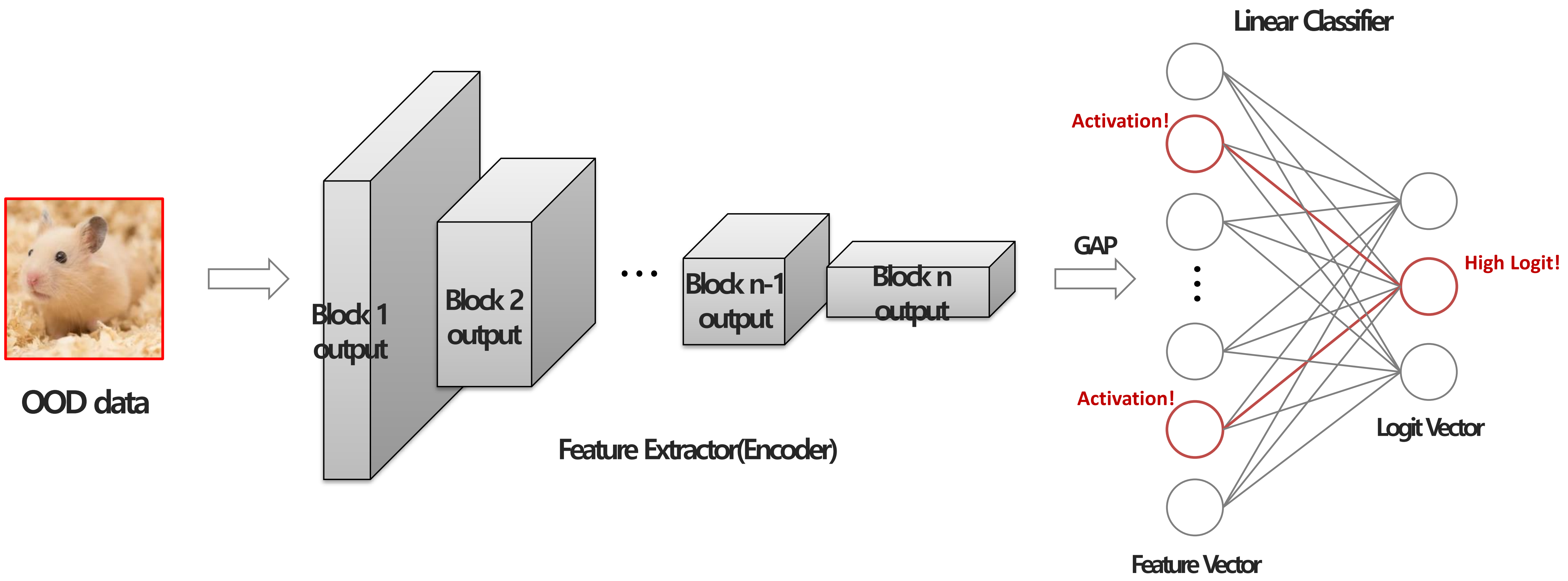
Introduction



CNN Architecture

Paper 1: ReAct

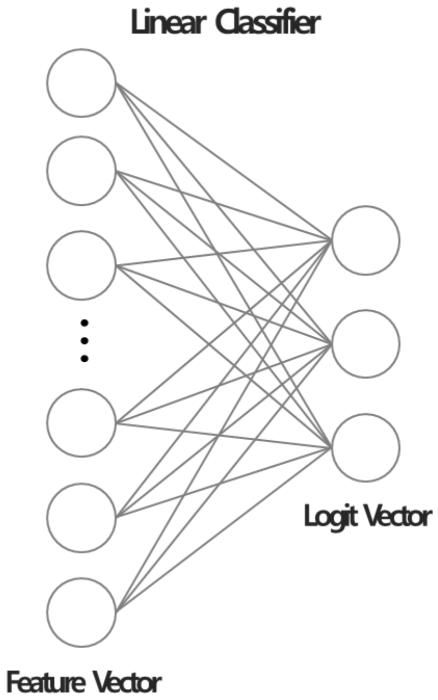
Introduction

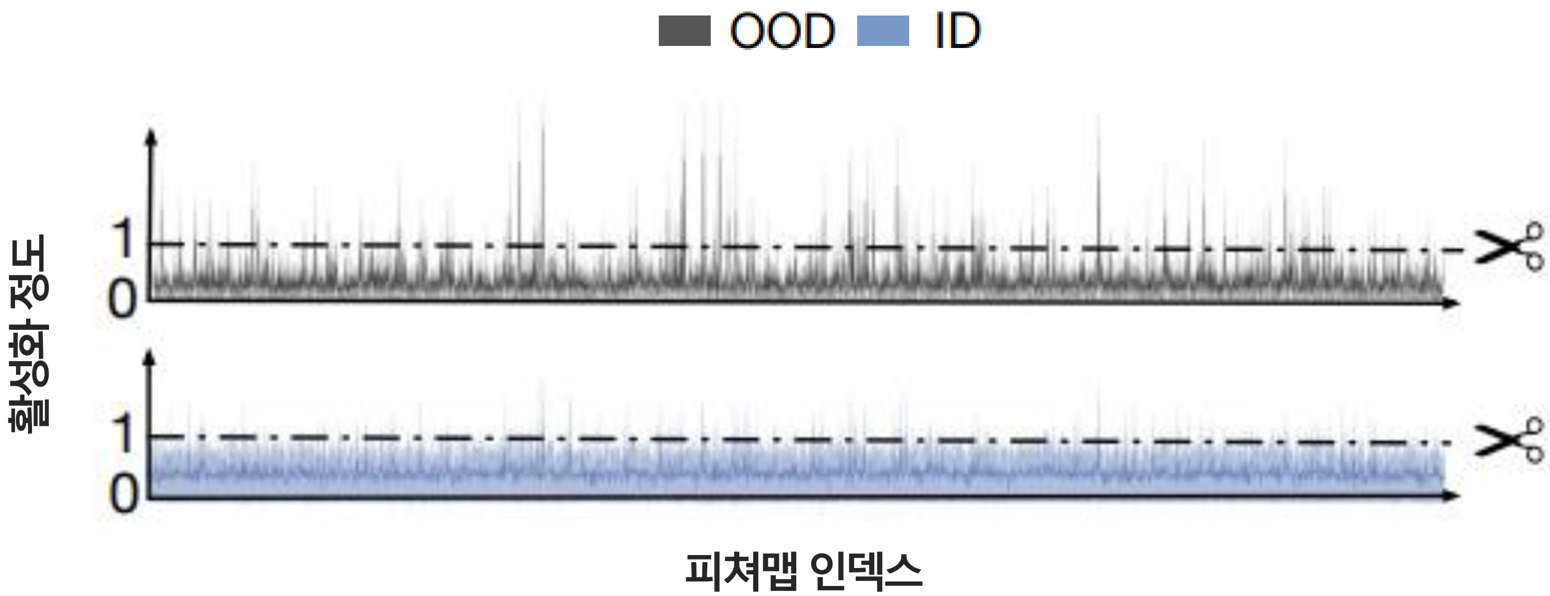


CNN Architecture

Paper 1: ReAct

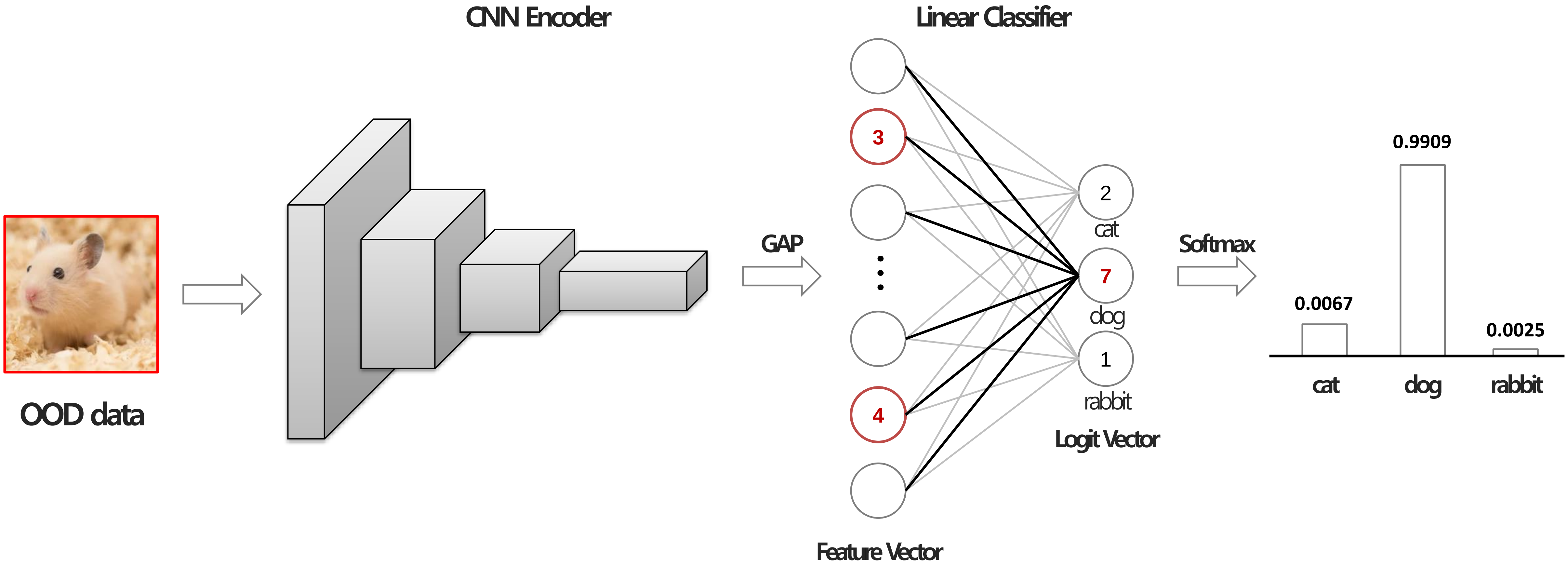
Introduction





Paper 1: ReAct

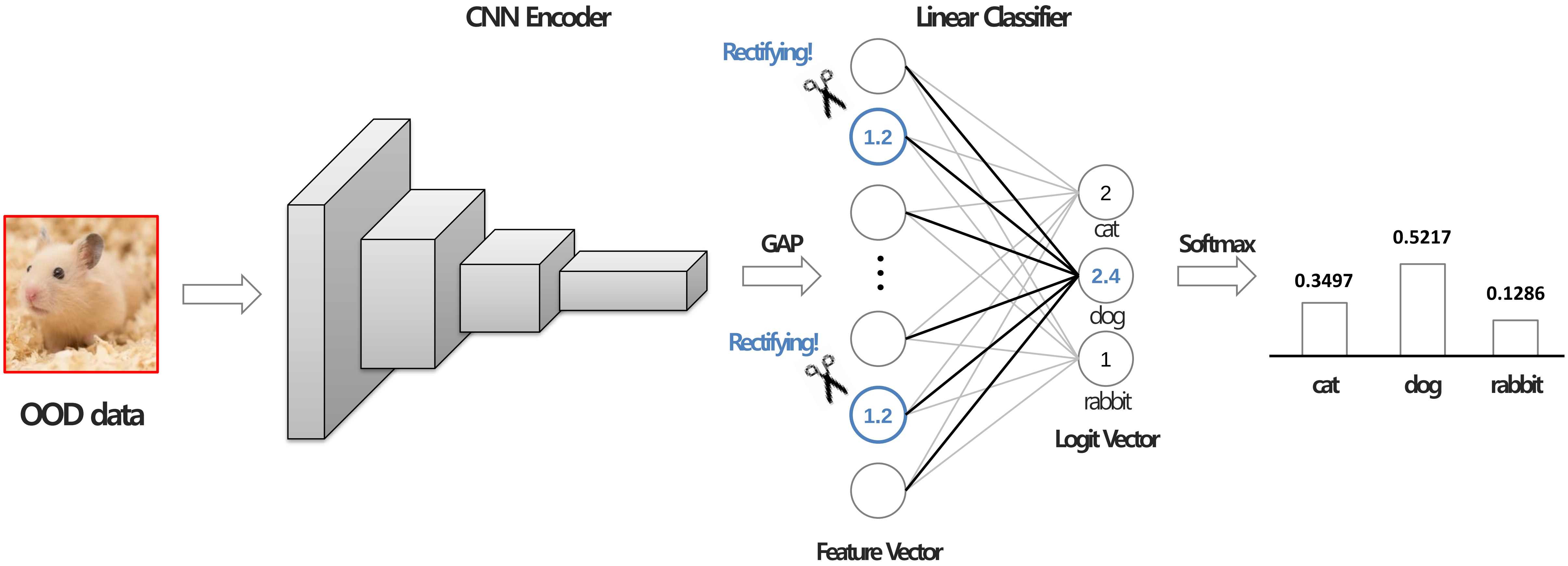
Method



Before ReAct

Paper 1: ReAct

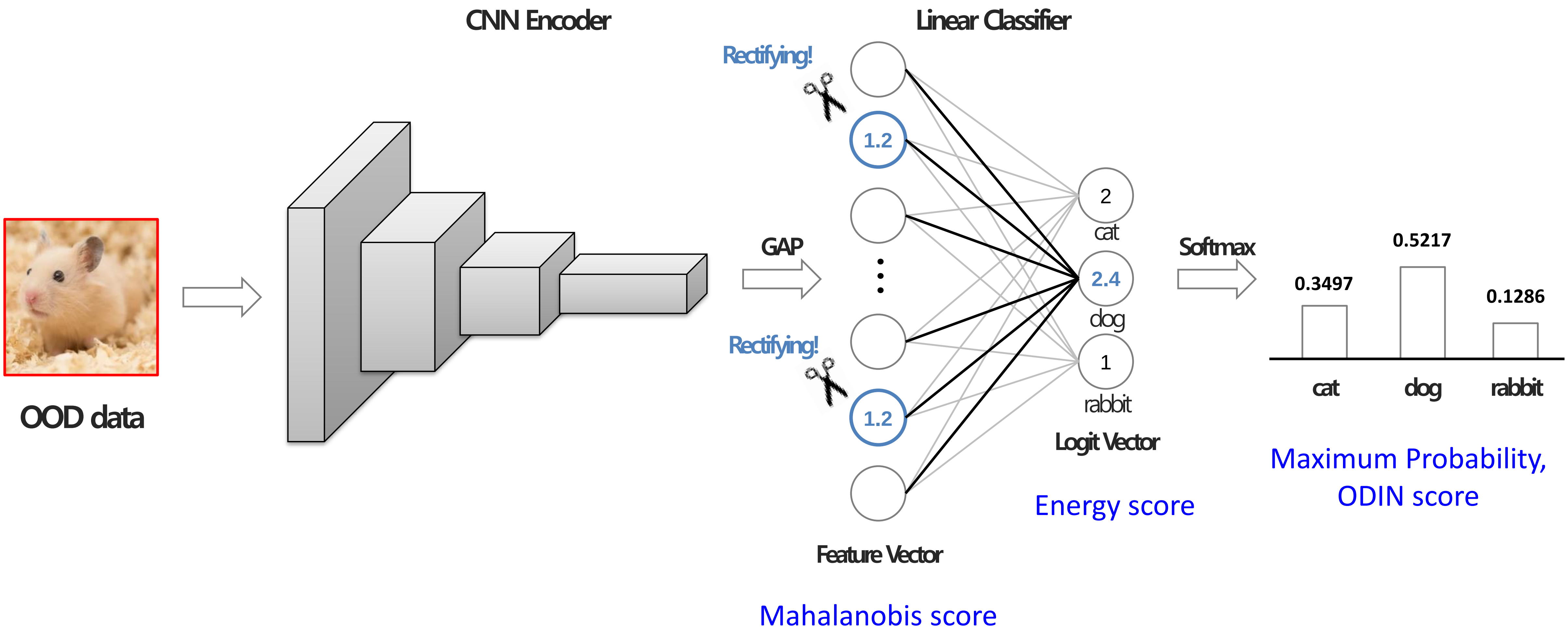
Method



After ReAct

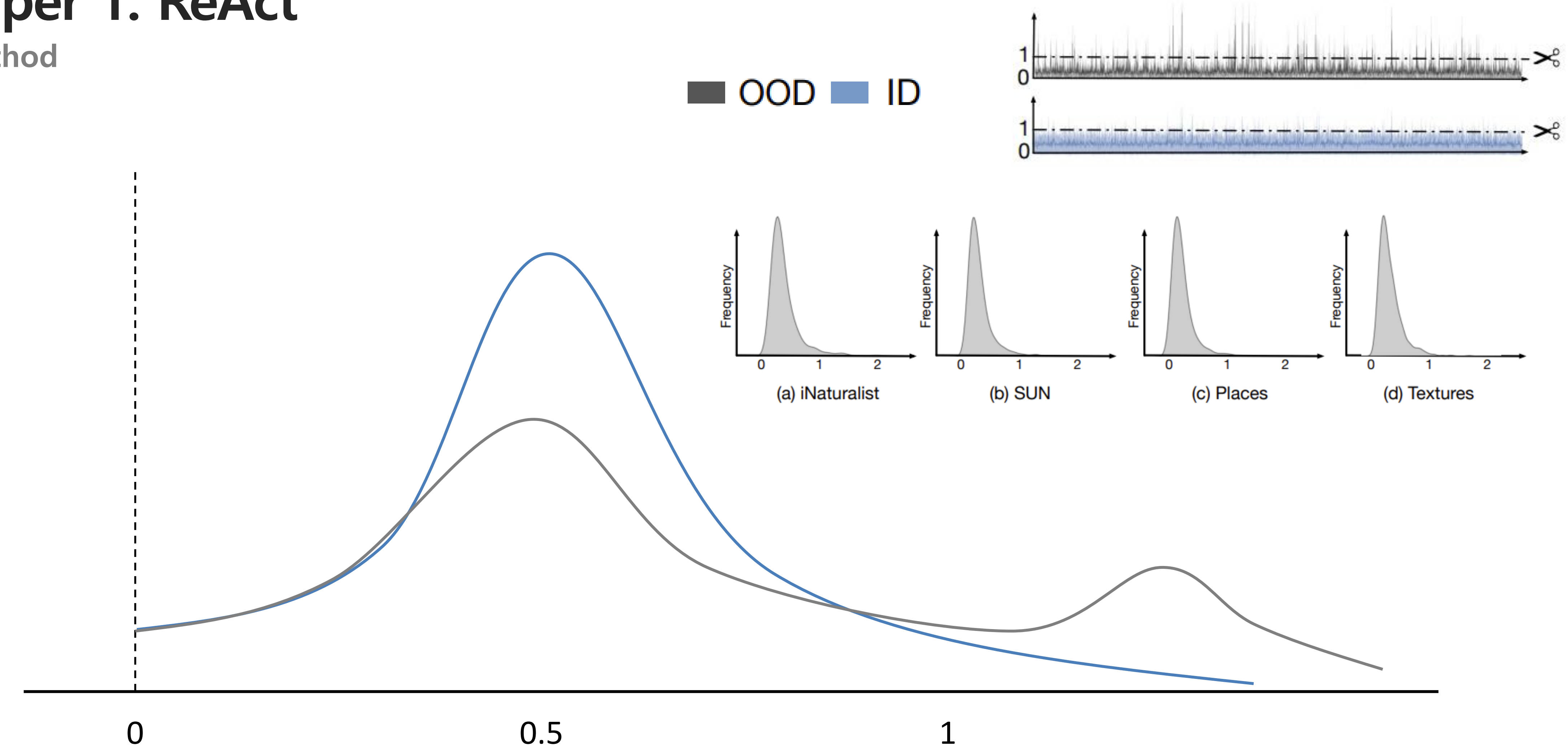
Paper 1: ReAct

Method



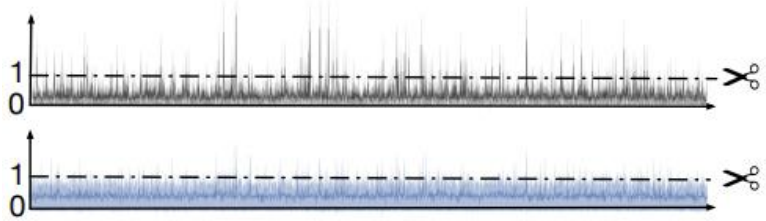
Paper 1: ReAct

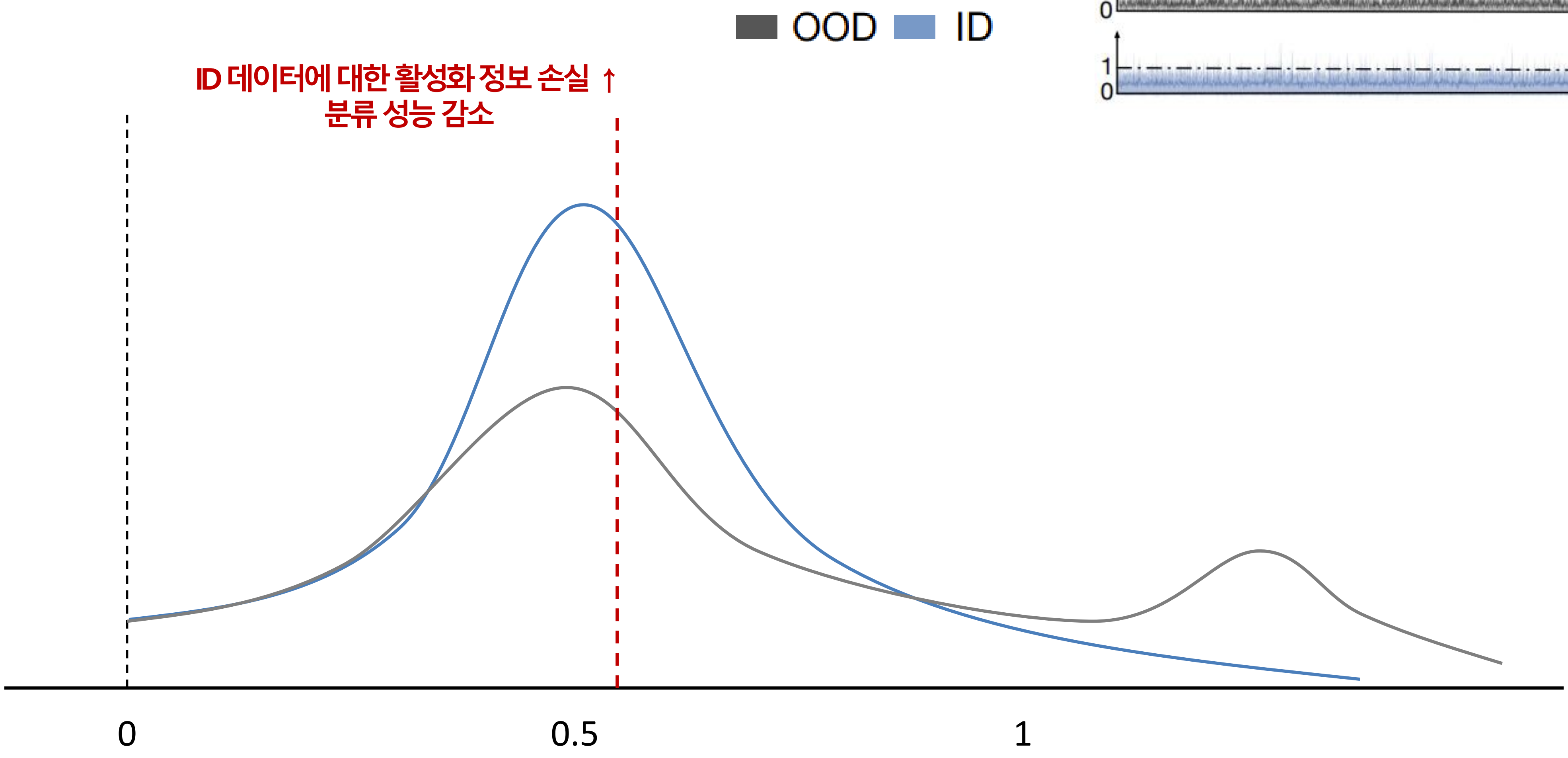
Method



Paper 1: ReAct

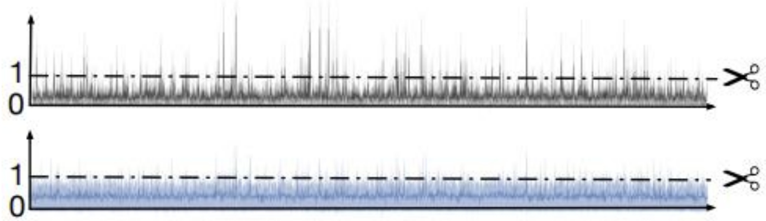
Method

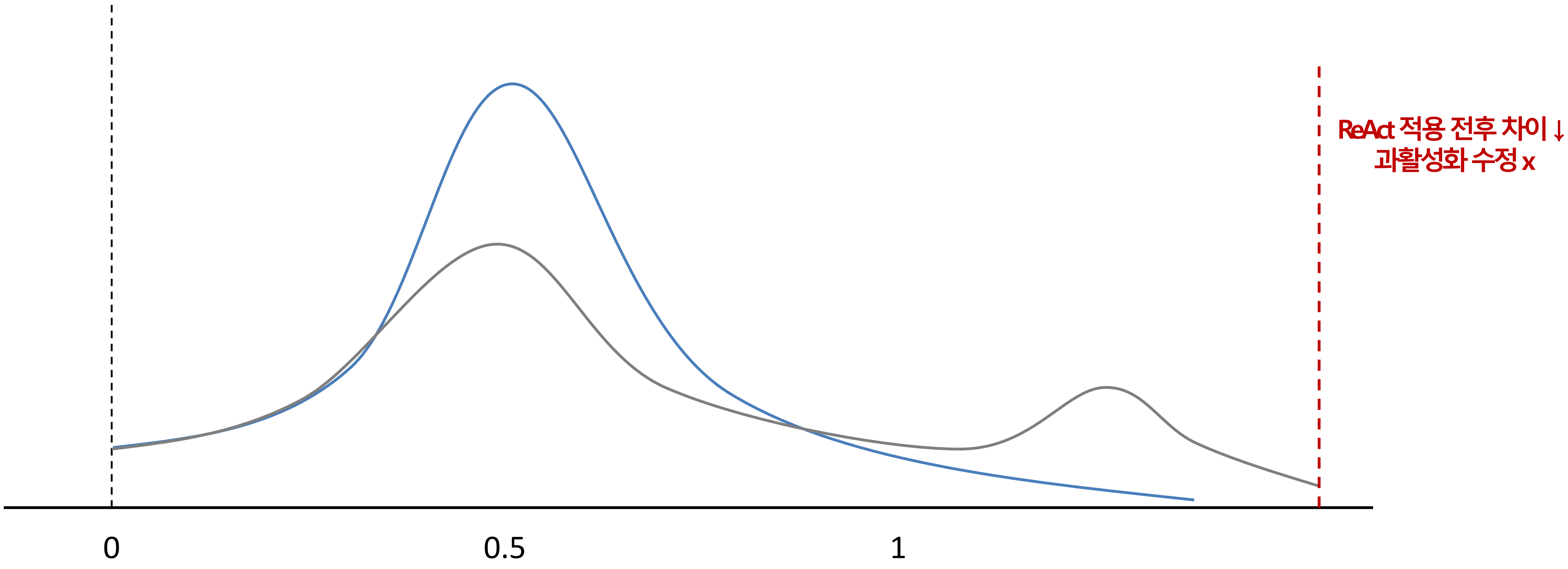




Paper 1: ReAct

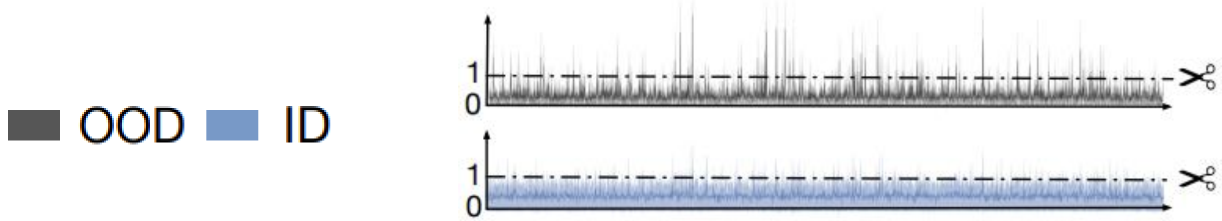
Method

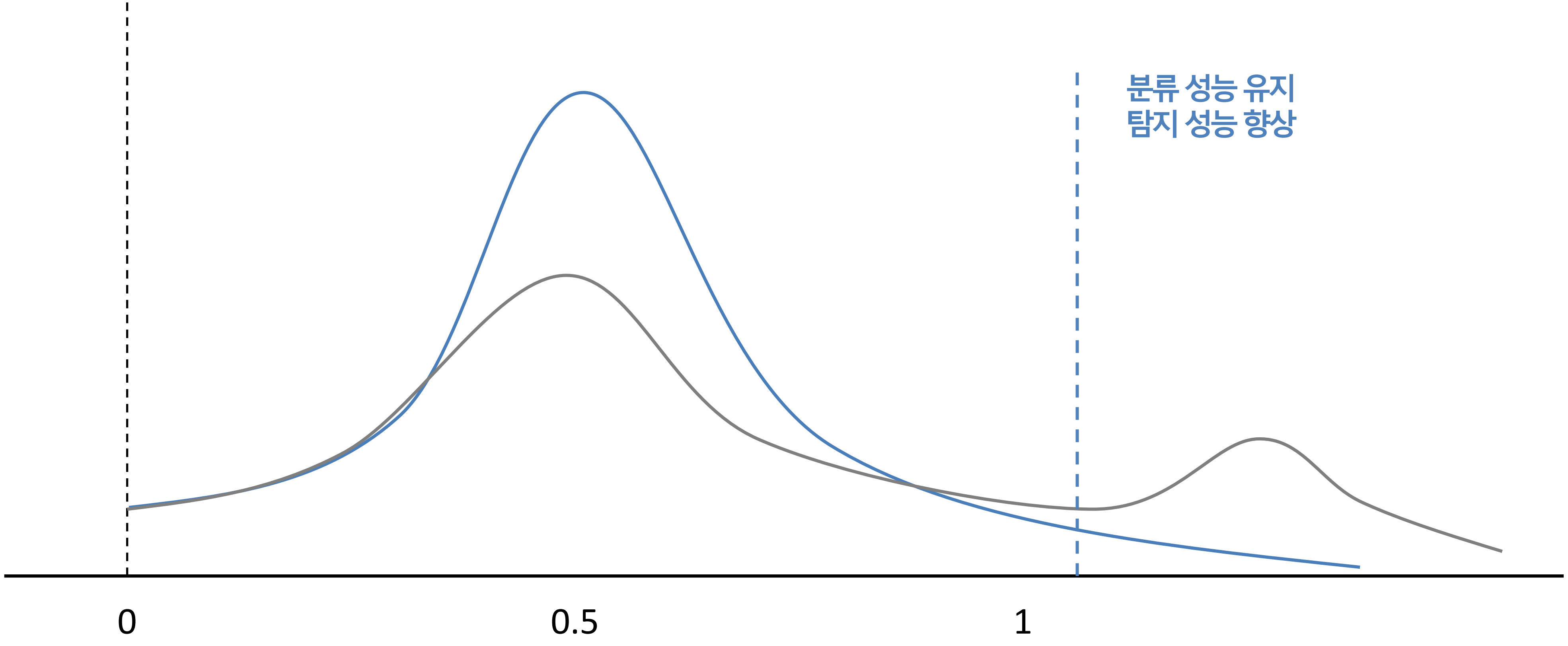




Paper 1: ReAct

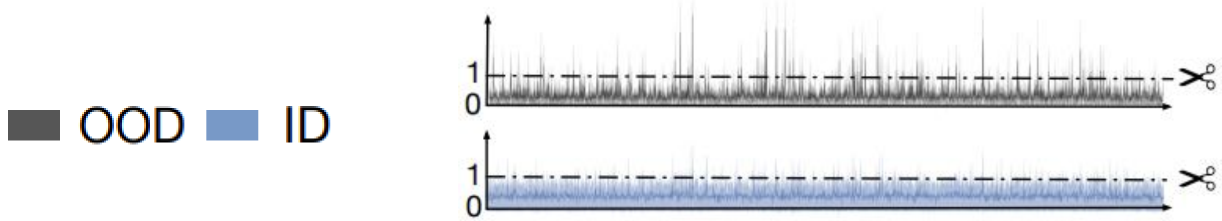
Method

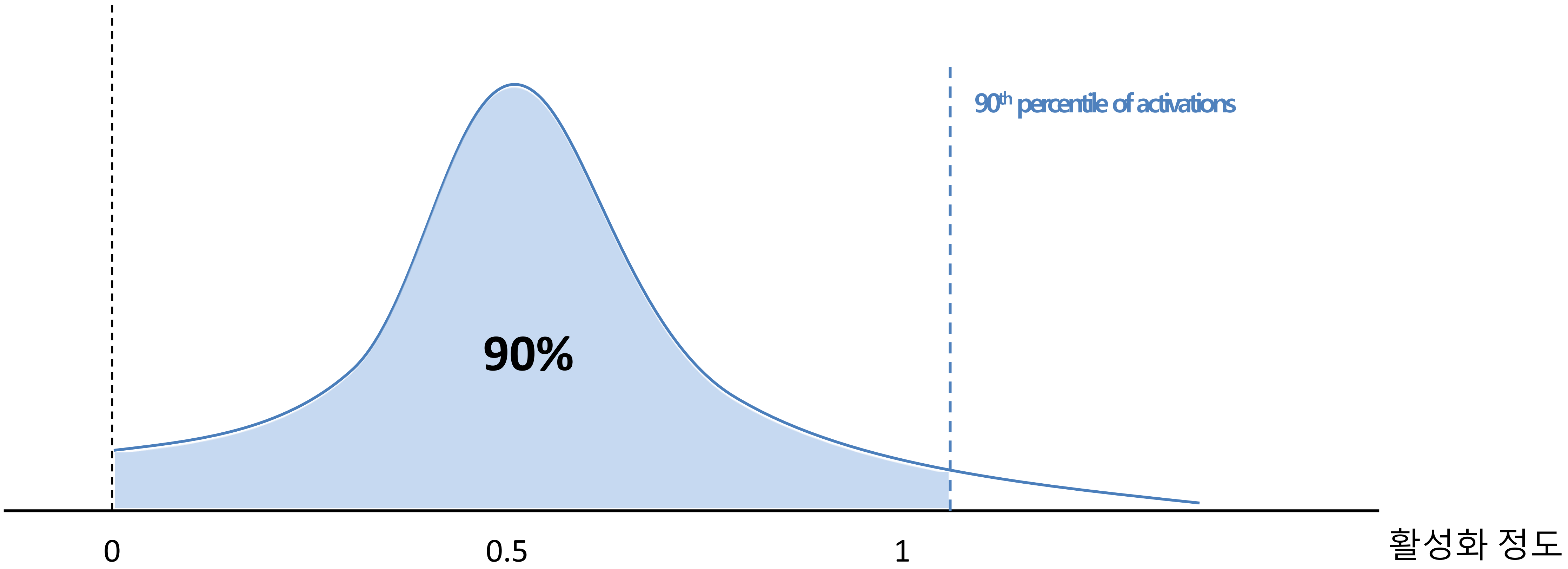




Paper 1: ReAct

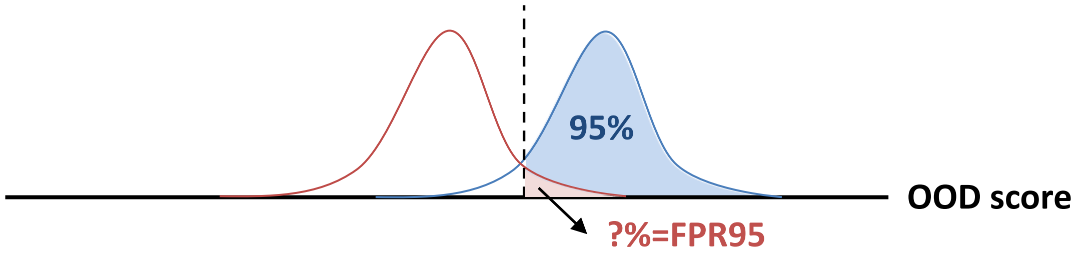
Method





Paper 1: ReAct

Experimental Results



Model	Methods	OOD Datasets								Average	
		iNaturalist		SUN		Places		Textures			
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
ResNet	MSP [15]	54.99	87.74	70.83	80.86	73.99	79.76	68.00	79.61	66.95	81.99
	ODIN [31]	47.66	89.66	60.15	84.59	67.89	81.78	50.23	85.62	56.48	85.41
	Mahalanobis [29]	97.00	52.65	98.50	42.41	98.40	41.79	55.80	85.01	87.43	55.47
	Energy [33]	55.72	89.95	59.26	85.89	64.92	82.86	53.72	85.99	58.41	86.17
	ReAct (Ours)	**20.38**	**96.22**	**24.20**	**94.20**	**33.85**	**91.58**	**47.30**	**89.80**	**31.43**	**92.95**
MobileNet	MSP [15]	64.29	85.32	77.02	77.10	79.23	76.27	73.51	77.30	73.51	79.00
	ODIN [31]	55.39	87.62	54.07	85.88	57.36	84.71	49.96	85.03	54.20	85.81
	Mahalanobis [29]	62.11	81.00	47.82	86.33	52.09	83.63	92.38	33.06	63.60	71.01
	Energy [33]	59.50	88.91	62.65	84.50	69.37	81.19	58.05	85.03	62.39	84.91
	ReAct (Ours)	**42.40**	**91.53**	**47.69**	**88.16**	**51.56**	**86.64**	**38.42**	**91.53**	**45.02**	**89.47**

Table 1: **Main results.** Comparison with competitive *post hoc* out-of-distribution detection methods. All methods are based on a model trained on **ID data only** (ImageNet-1k), without using any auxiliary outlier data. ↑ indicates larger values are better and ↓ indicates smaller values are better. All values are percentages.

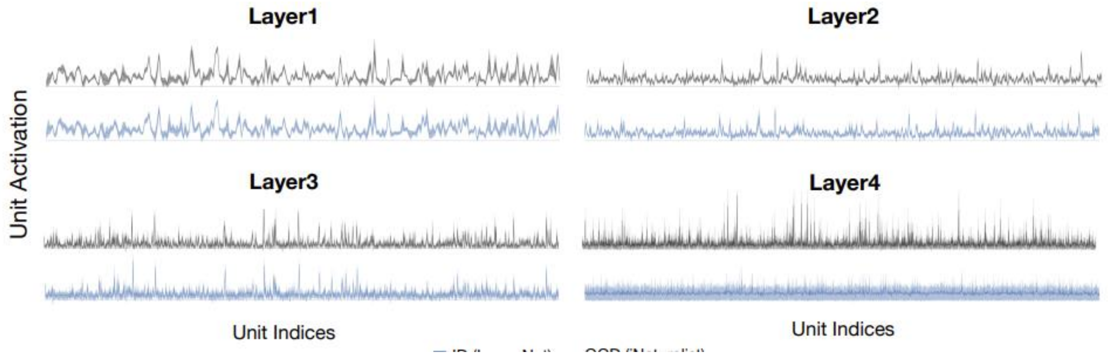
Paper 1: ReAct

Experimental Results

Rectification percentile	FPR95 ↓	AUROC ↑	AUPR ↑	ID ACC. ↑	Rectification threshold c
No ReAct [33]	58.41	86.17	96.88	75.08	∞
$p = 99$	44.57	90.45	97.96	75.12	2.25
$p = 95$	35.39	92.39	98.37	74.76	1.50
$p = 90$	31.43	92.95	98.50	73.75	1.00
$p = 85$	34.08	92.05	98.35	72.91	0.84
$p = 80$	41.51	89.54	97.91	71.93	0.72
$p = 65$	74.62	74.14	94.39	67.14	0.50
$p = 10$	74.70	57.55	86.06	1.22	0.06

Table 2: Effect of rectification threshold for inference. Model is trained on ImageNet using ResNet-50 [11]. All numbers are percentages and are averaged over 4 OOD test datasets.

Layers of applying ReAct	ID: CIFAR-100		ID: ImageNet	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
Layer1	90.86	68.17	84.83	74.88
Layer2	84.12	75.32	76.25	79.37
Layer3	73.4	80.91	63.87	86.46
Layer4 (ReAct)	**59.61**	**87.48**	**31.43**	**92.95**
No ReAct [33]	71.93	82.82	58.41	86.17



Paper 1: ReAct

Further Analysis

$$\mathrm{BatchNorm}(z;\gamma,\beta,\epsilon)=\frac{z-\mathbb{E}_{\mathrm{in}}[z]}{\sqrt{\mathrm{Var}_{\mathrm{in}}[z]+\epsilon}}\cdot\gamma+\beta$$

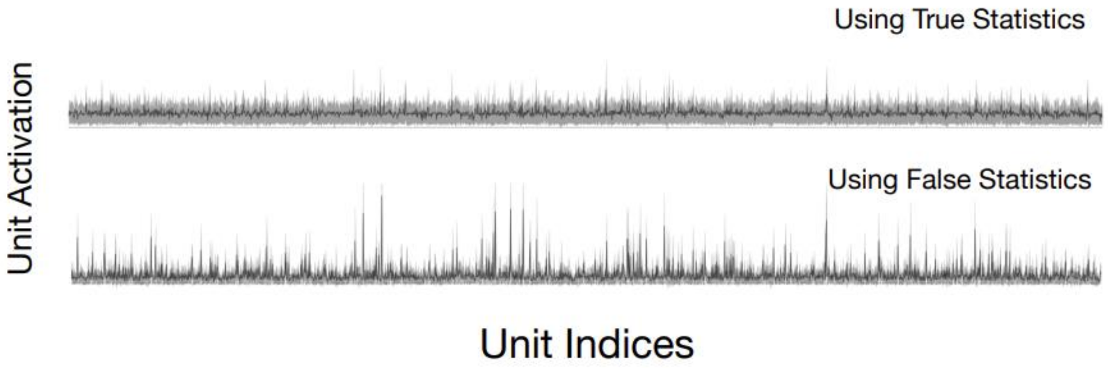


Figure 4: The distribution of per-unit activations in the penultimate layer for OOD data (iNaturalist) by using *true* (top) vs. *mismatched* (bottom) BatchNorm statistics for OOD data.

Method	iNaturalist	Places	SUN	Textures
Oracle (batch OOD for estimating BN statistics)	99.59	99.09	98.32	91.43
ReAct (single OOD)	96.22	94.20	91.58	89.80
No ReAct [33]	89.95	85.89	82.86	85.99

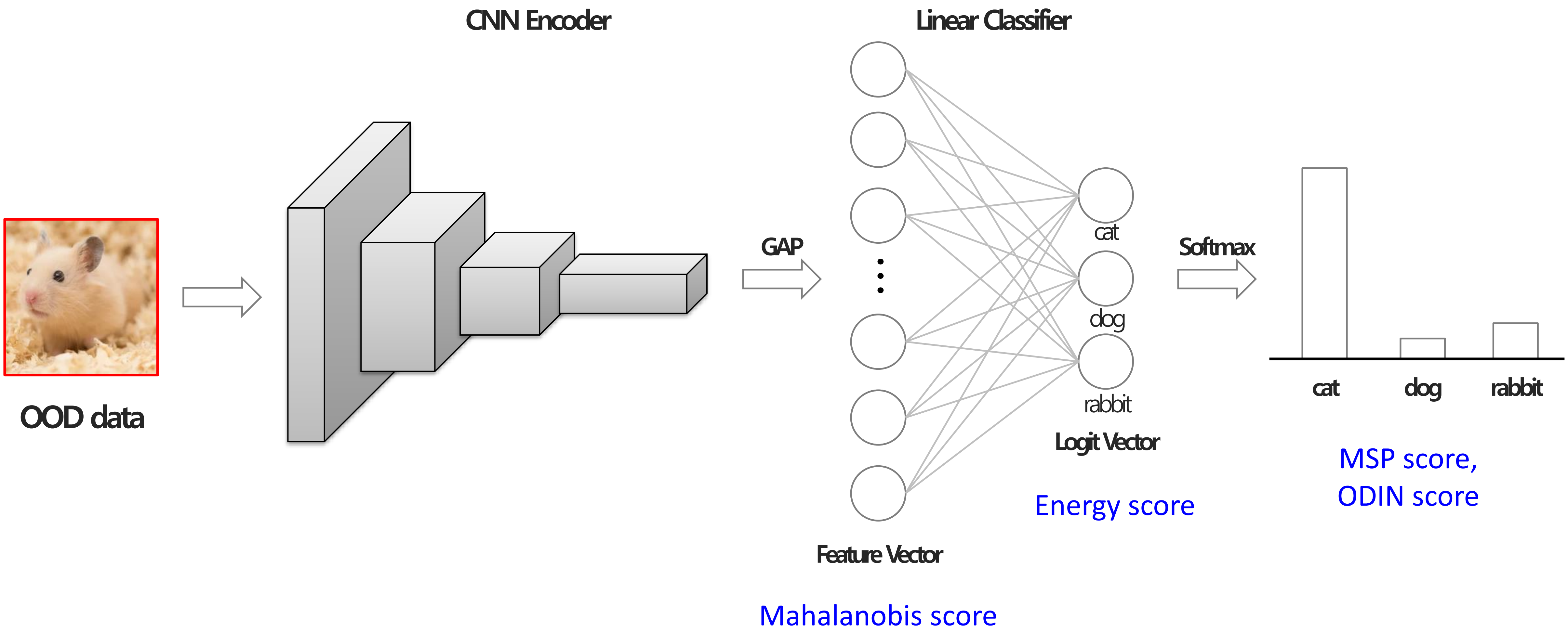
Table 4: Comparison with oracle using OOD BN statistics. Model is trained on ImageNet (see Section 4.1). Values are AUROC.

On the Importance of Gradients for Detecting Distributional Shifts in the Wild (GradNorm)

(2021, NeurIPS)

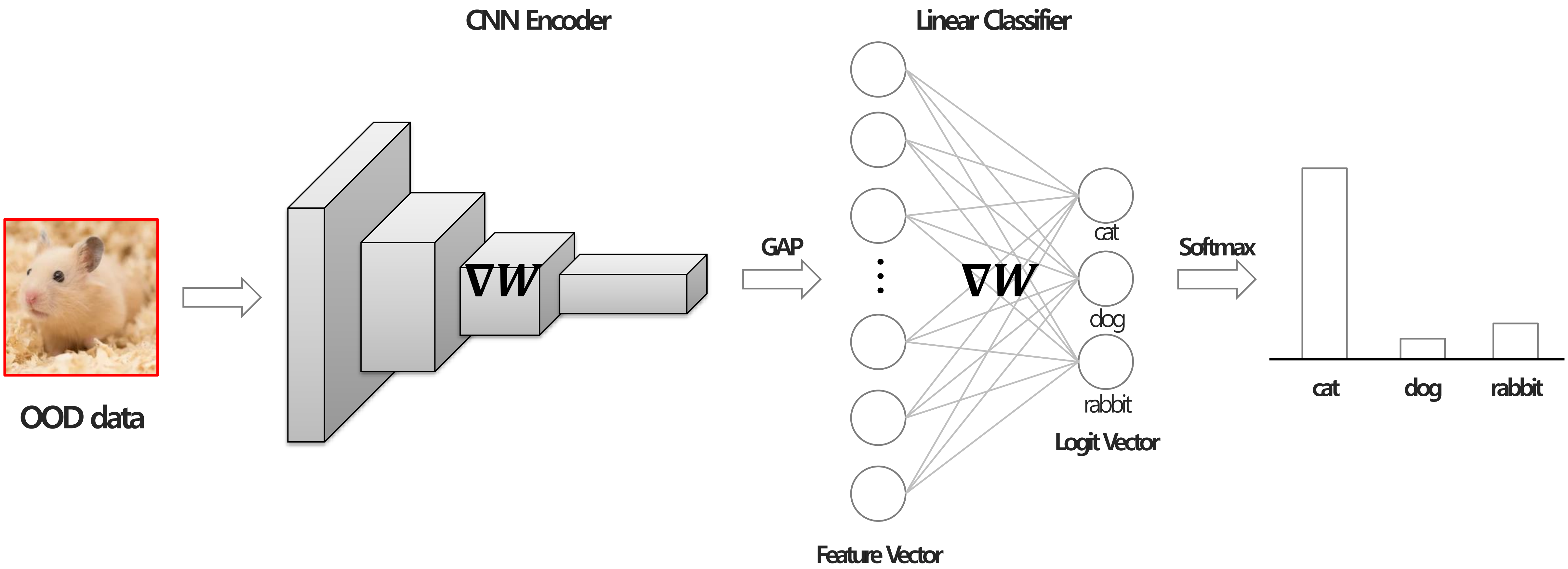
Paper 2: GradNorm

Introduction



Paper 2: GradNorm

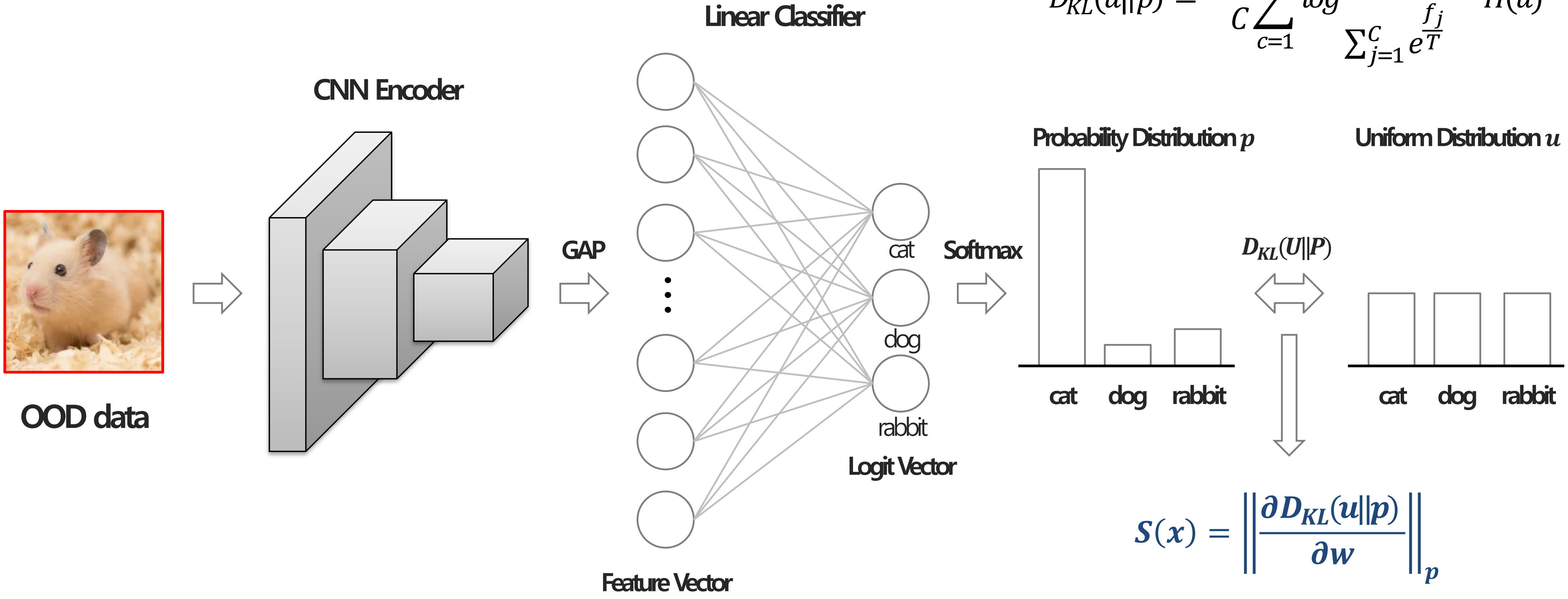
Introduction



Paper 2: GradNorm

Method

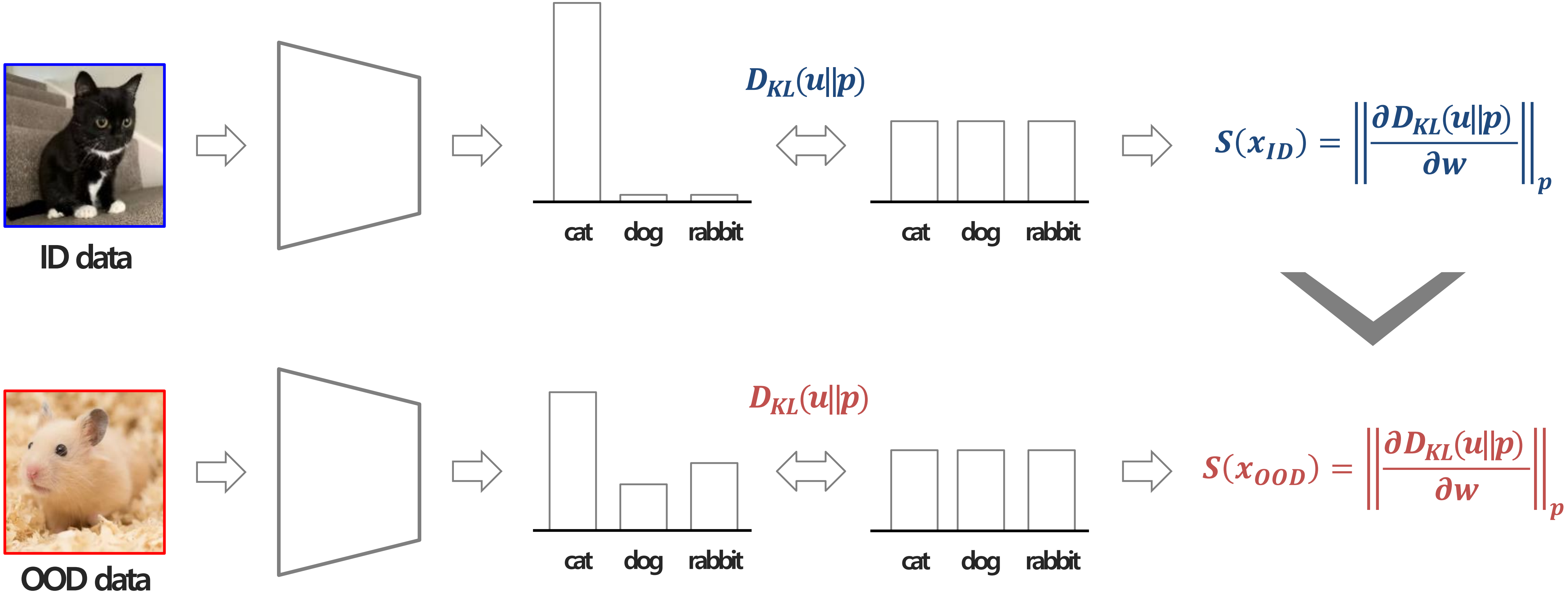
$$D_{KL}(u\|p) = -\frac{1}{C}\sum_{c=1}^{C} log \frac{e^{\frac{f_c}{T}}}{\sum_{j=1}^{C} e^{\frac{f_j}{T}}} - H(u)$$



$$S(x) = \left\| \frac{\partial D_{KL}(u\|p)}{\partial w} \right\|_p$$

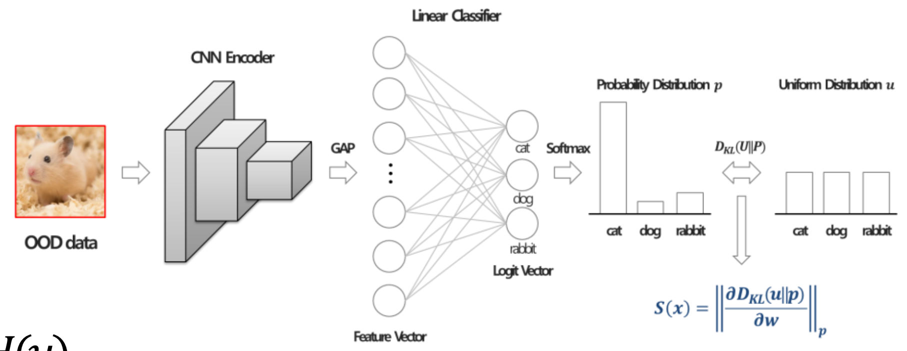
Paper 2: GradNorm

Method



Paper 2: GradNorm

Method



$$D_{KL}(u\|p) = -\frac{1}{C}\sum_{c=1}^{C} \log \frac{e^{\frac{f_c}{T}}}{\sum_{j=1}^{C} e^{\frac{f_j}{T}}} - H(u) = -\frac{1}{C}\left(\frac{1}{T}\sum_{c=1}^{C} f_c - C\cdot\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}\right) - H(u)$$

$$\frac{\partial D_{KL}}{\partial f_c} = -\frac{1}{CT}\left(1 - CT\cdot\frac{\partial\left(\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}\right)}{\partial f_c}\right) = -\frac{1}{CT}\left(1 - CT\cdot\frac{e^{\frac{f_c}{T}}}{\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}}\right) \quad \because \frac{\partial \ln f(x)}{\partial x} = \frac{f'(x)}{f(x)}$$

$$f = Wx + b$$

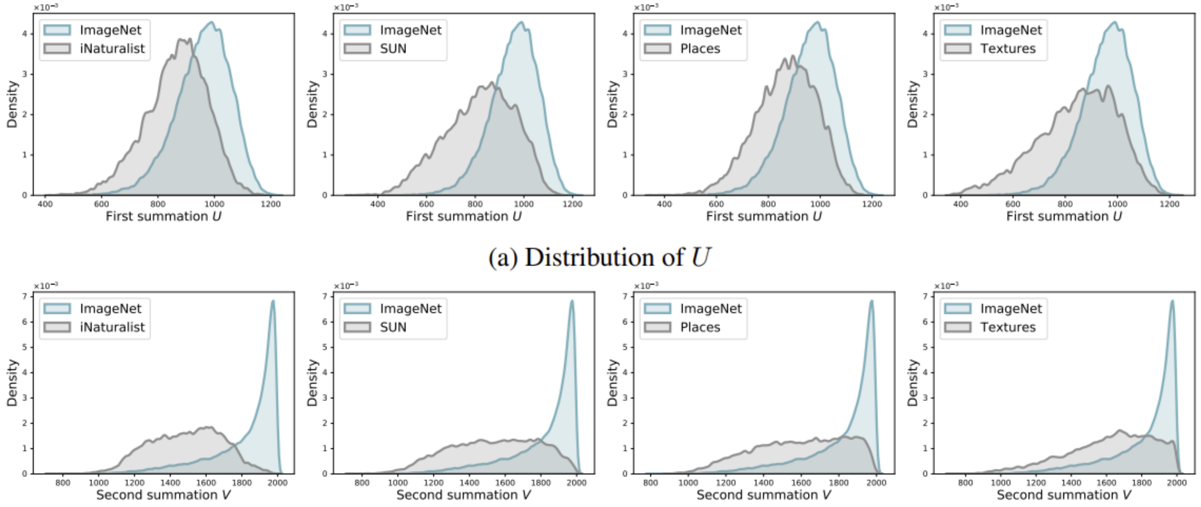
$$\frac{\partial D_{KL}}{\partial W} = \frac{\partial D_{KL}}{\partial f}\cdot\frac{\partial f}{\partial W} = x\cdot\frac{\partial D_{KL}}{\partial f} = -\frac{1}{CT}\cdot[x_1, x_2, \ldots, x_m]^T\left[1 - CT\cdot\frac{e^{\frac{f_1}{T}}}{\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}}, 1 - CT\cdot\frac{e^{\frac{f_2}{T}}}{\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}}, \ldots, 1 - CT\cdot\frac{e^{\frac{f_C}{T}}}{\log\sum_{c=1}^{C} e^{\frac{f_j}{T}}}\right]$$

Feature space 정보 Output space 정보

$$S(x) = \sum_{i=1}^{m}\sum_{j=1}^{C}\left|\left(\frac{\partial D_{KL}}{\partial W}\right)_{ij}\right| = \frac{1}{CT}\left(\sum_{i=1}^{m}|x_i|\right)\left(\sum_{j=1}^{C}\left|1 - C\cdot\frac{e^{\frac{f_j}{T}}}{\log\sum_{c=1}^{C} e^{\frac{f_c}{T}}}\right|\right) = \frac{1}{CT}U\cdot V$$

Paper 2: GradNorm

Method



(a) Distribution of U

(b) Distribution of V

Figure 7: We show the distributions of the two summations decomposed from the L_1-norm of the last layer gradient, for both in-distribution data (blue) and out-of-distribution data (gray).

Method	iNaturalist		SUN		Places		Textures		Average	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
U (feature space)	77.84	74.33	61.90	78.74	76.42	72.75	67.84	72.77	71.00	74.65
V (output space)	66.14	88.45	69.49	83.13	75.95	78.98	81.13	76.06	73.18	81.66
U · V (Joint space)	**50.05**	**90.33**	**46.48**	**89.03**	**60.86**	**84.82**	**61.42**	**81.07**	**54.70**	**86.31**

Table 5: OOD detection performance using the decomposed U (feature space) and V (output space) as scoring functions. Model is ResNetv2-101 trained on ImageNet-1k [6].

Paper 2: GradNorm

Experimental Results

Method Space	Method	iNaturalist		SUN		Places		Textures		Average	
		FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
Output	MSP [13]	63.69	87.59	79.98	78.34	81.44	76.76	82.73	74.45	76.96	79.29
	ODIN [26]	62.69	89.36	71.67	83.92	76.27	80.67	81.31	76.30	72.99	82.56
	Energy [28]	64.91	88.48	65.33	85.32	73.02	81.37	80.87	75.79	71.03	82.74
Feature	Mahalanobis [25]	96.34	46.33	88.43	65.20	89.75	64.46	**52.23**	72.10	81.69	62.02
Gradient	**GradNorm (ours)**	**50.03**	**90.33**	**46.48**	**89.03**	**60.86**	**84.82**	61.42	**81.07**	**54.70**	**86.31**

Table 1: **Main Results.** OOD detection performance comparison between `GradNorm` and baselines. All methods utilize the standard ResNetv2-101 model trained on ImageNet [6]. The classification model is trained on ID data only. ↑ indicates larger values are better, while ↓ indicates smaller values are better. All values are percentages. All methods are post hoc and can be directly used for pre-trained models.

Paper 2: GradNorm

Experimental Results

Gradient Space	FPR95 ↓	AUROC ↑
Block 1	73.52	76.41
Block 2	74.34	76.63
Block 3	71.73	78.11
Block 4	65.07	85.11
All params	69.35	81.14
Last layer params	**54.70**	**86.31**

Table 2: Effect of `GradNorm` using different subset of gradients. Gradient norm derived from deeper layers yield better OOD detection performance.

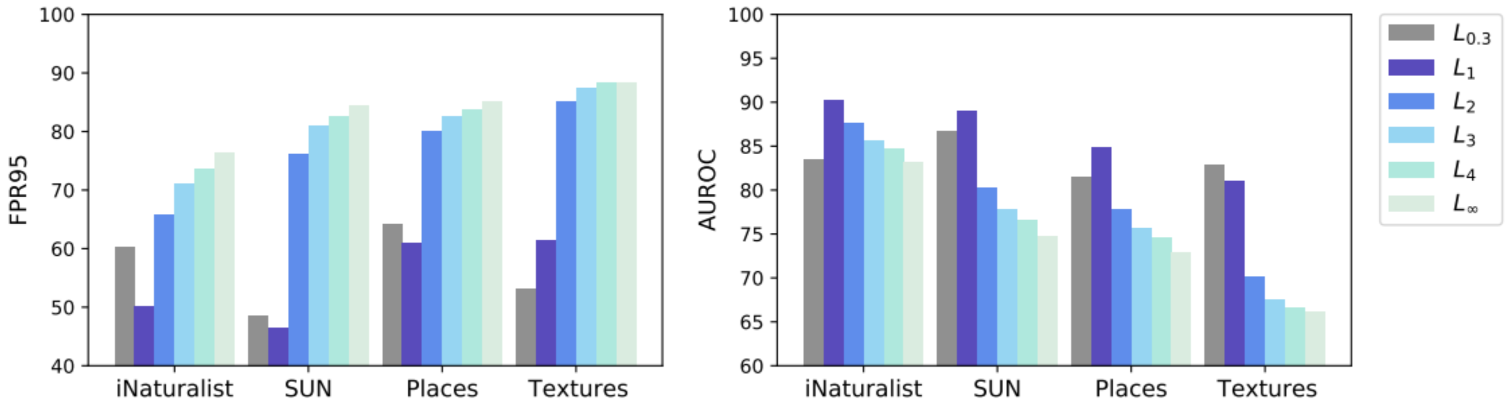


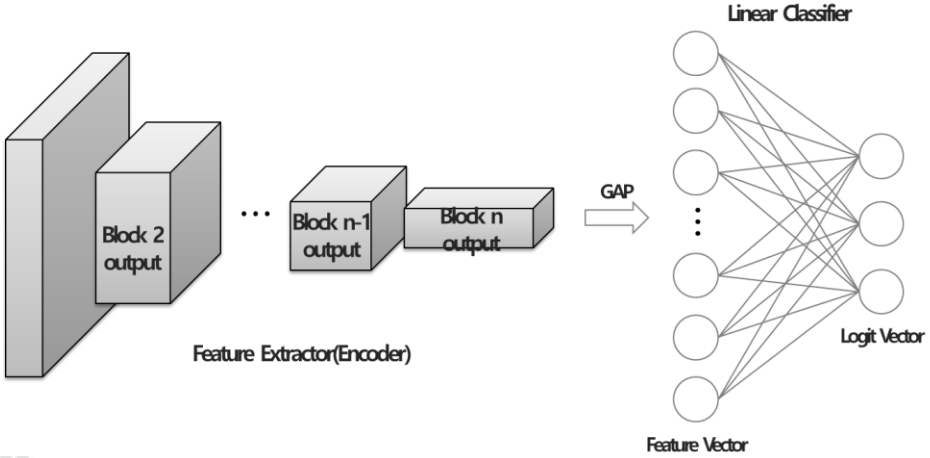
Figure 4: OOD detection performance comparison under different L_p-norms. We show FPR95 (*left*) and AUROC (*right*).

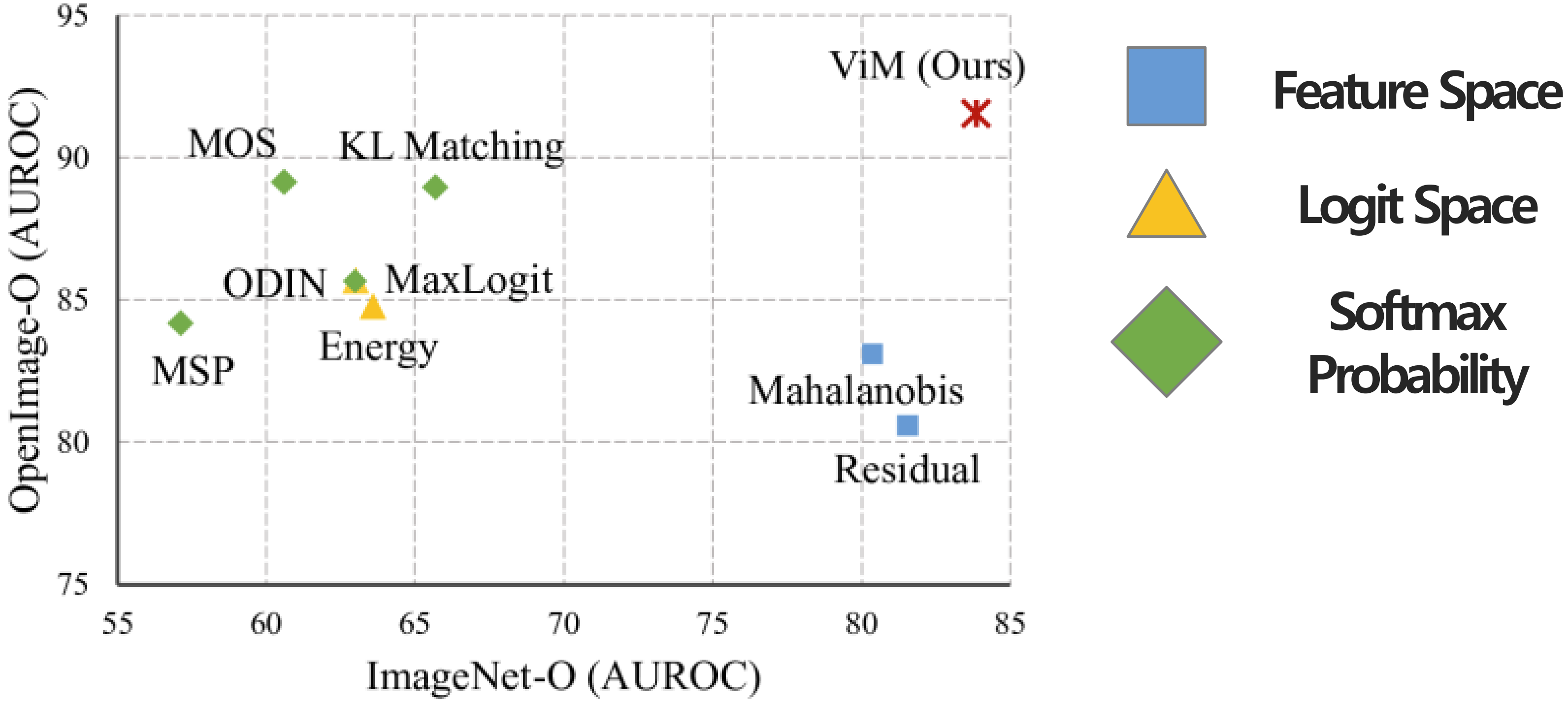
ViM: Out-Of-Distribution with Virtual-Logit Matching

(2022, CVPR)

Paper 3: Vim

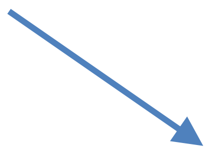
Introduction





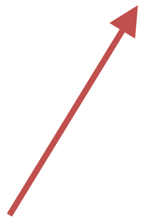
Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

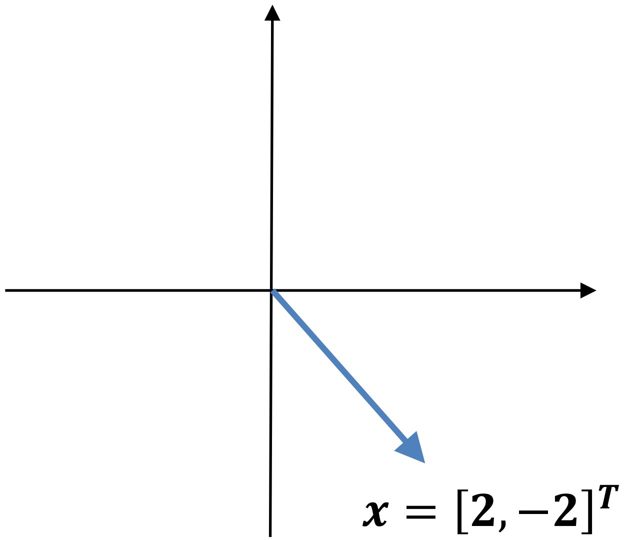


$x \in R^n$

$A \in R^{m \times n}$



$y = Ax \in R^m$

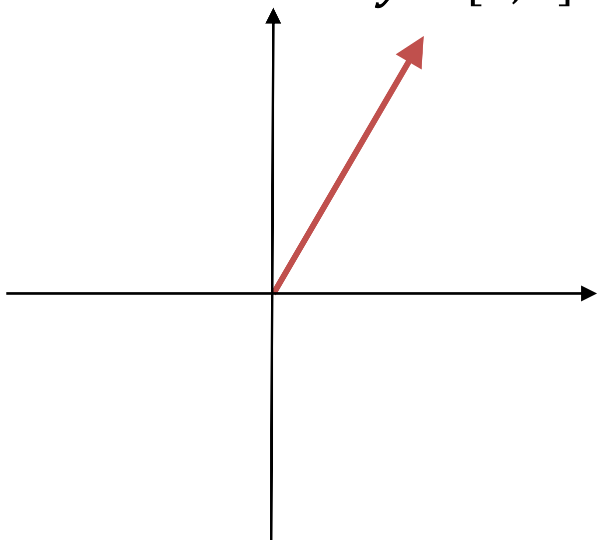


$x = [2, -2]^T$

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix}$$

$$y = Ax = \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ -2 \end{bmatrix} = [2, 4]^T$$

$y = [2, 4]^T$

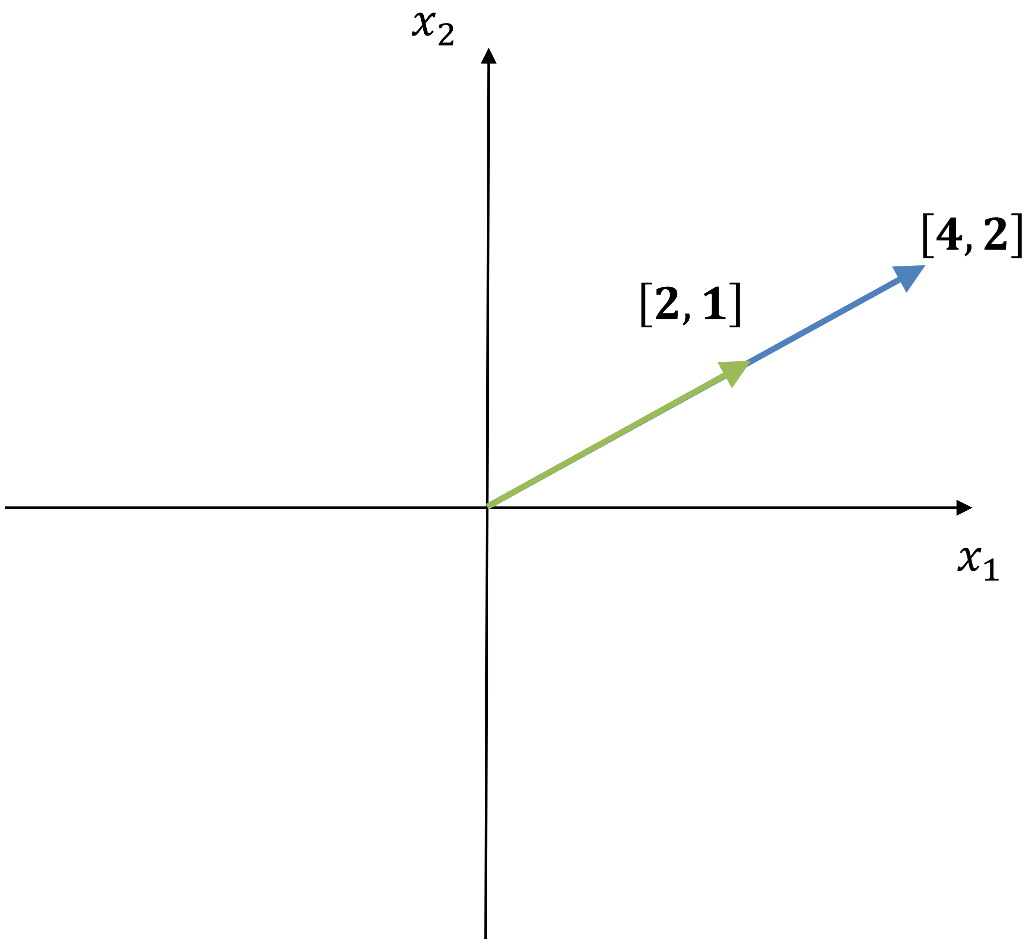


Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{2} & \mathbf{1} \\ \mathbf{4} & \mathbf{2} \end{bmatrix}$$

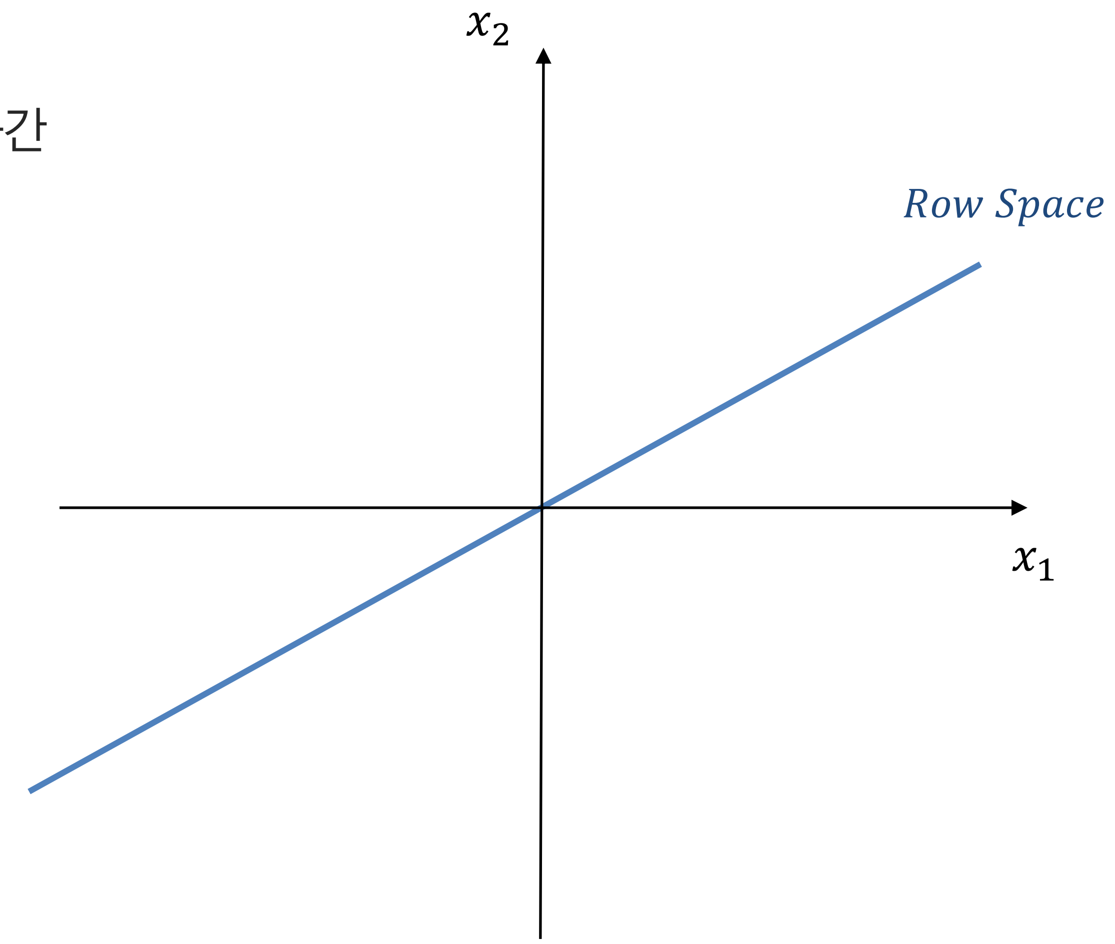


Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix}$$

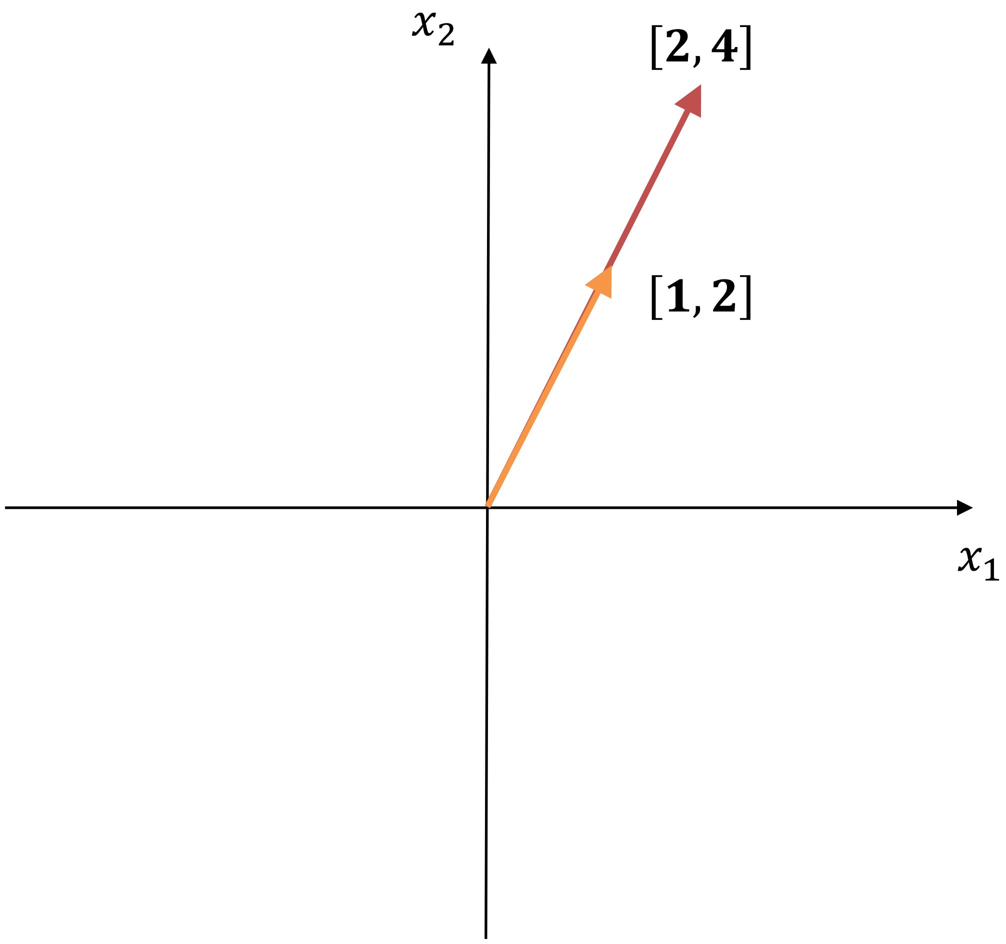


Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix}$$

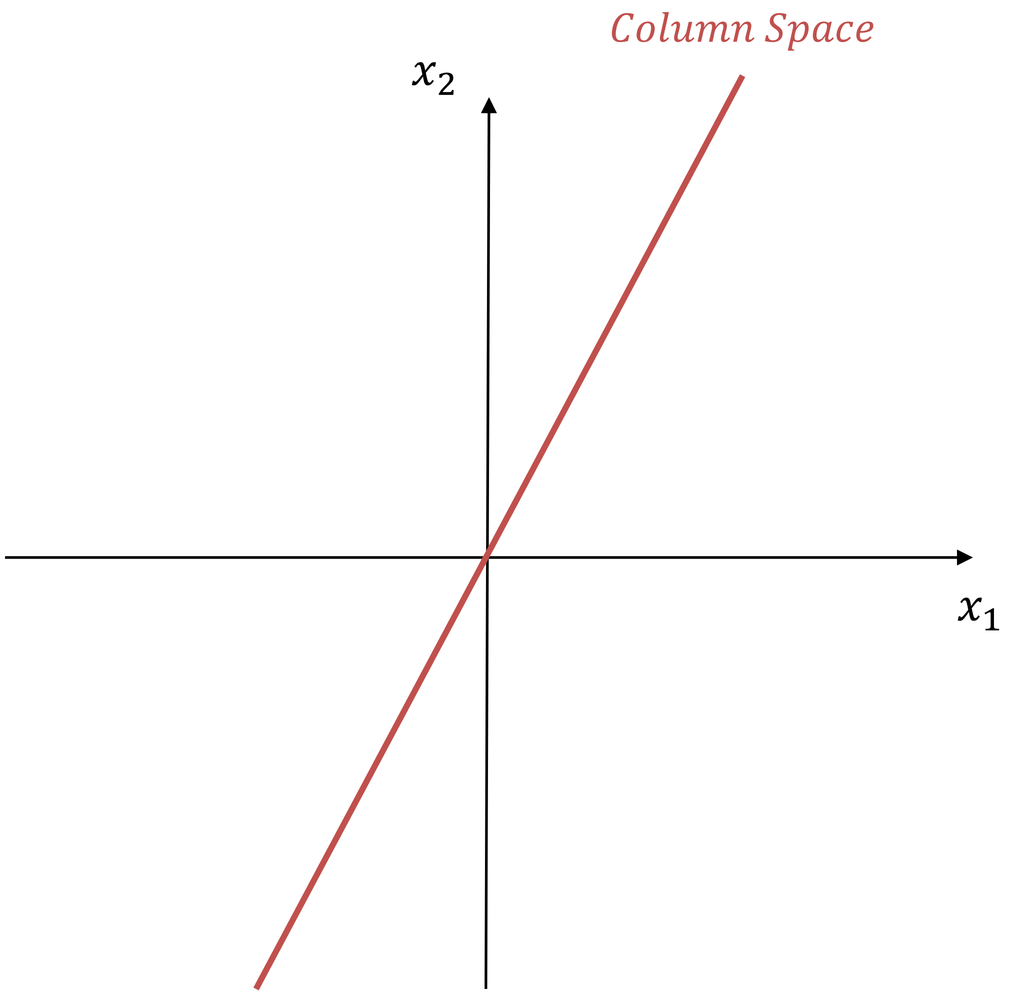


Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$A = \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix}$$



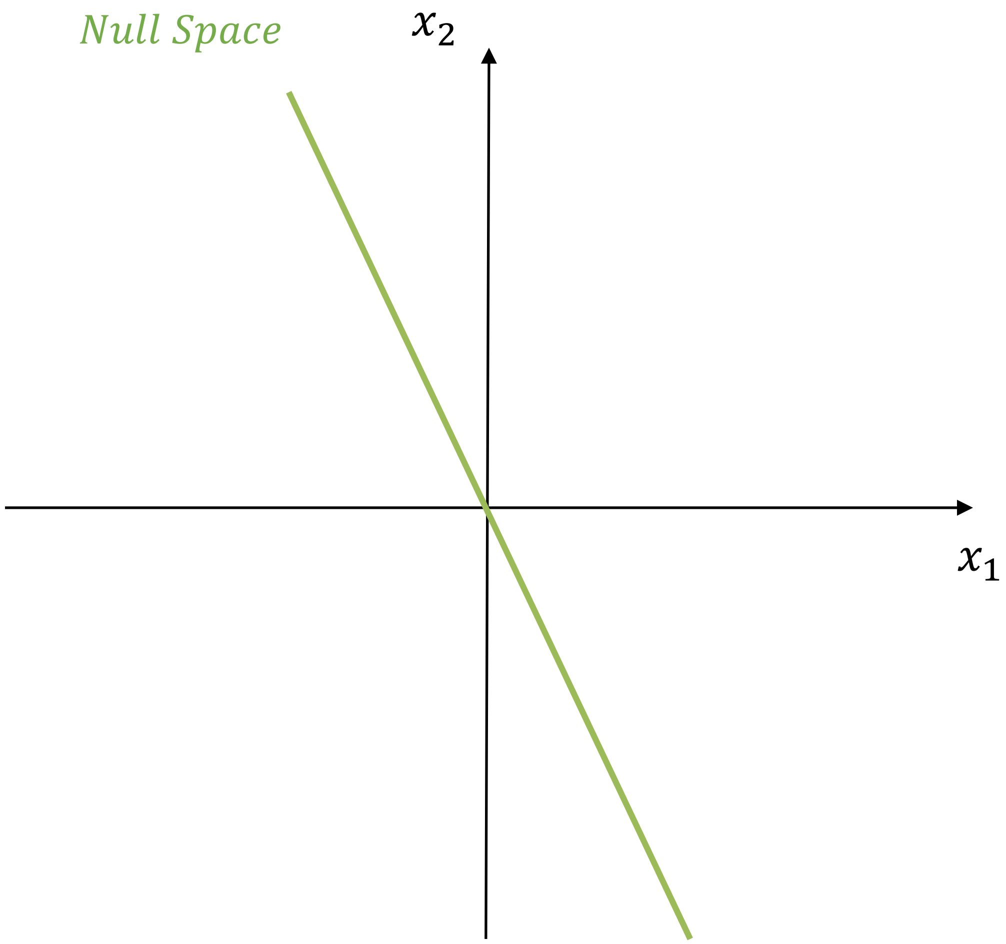
Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$Ax = 0 \rightarrow \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$2x_1 + x_2 = 0 \rightarrow x_2 = -2x_1$$



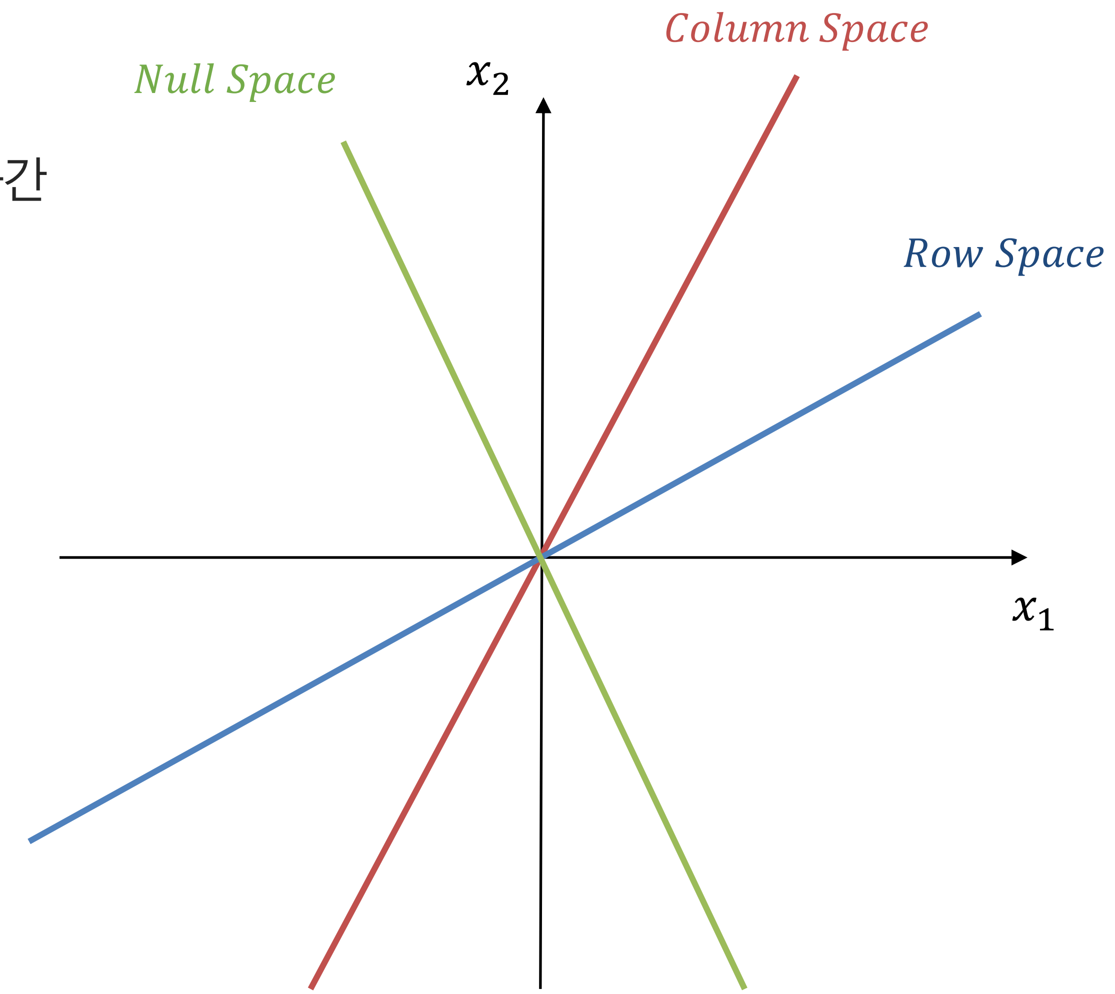
Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

$$Ax = y \rightarrow \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2x_1 + x_2 \\ 4x_1 + 2x_1 \end{bmatrix}$$

$$Let, 2x_1 + x_2 = x', then\ Ax = \begin{bmatrix} x' \\ 2x' \end{bmatrix}$$



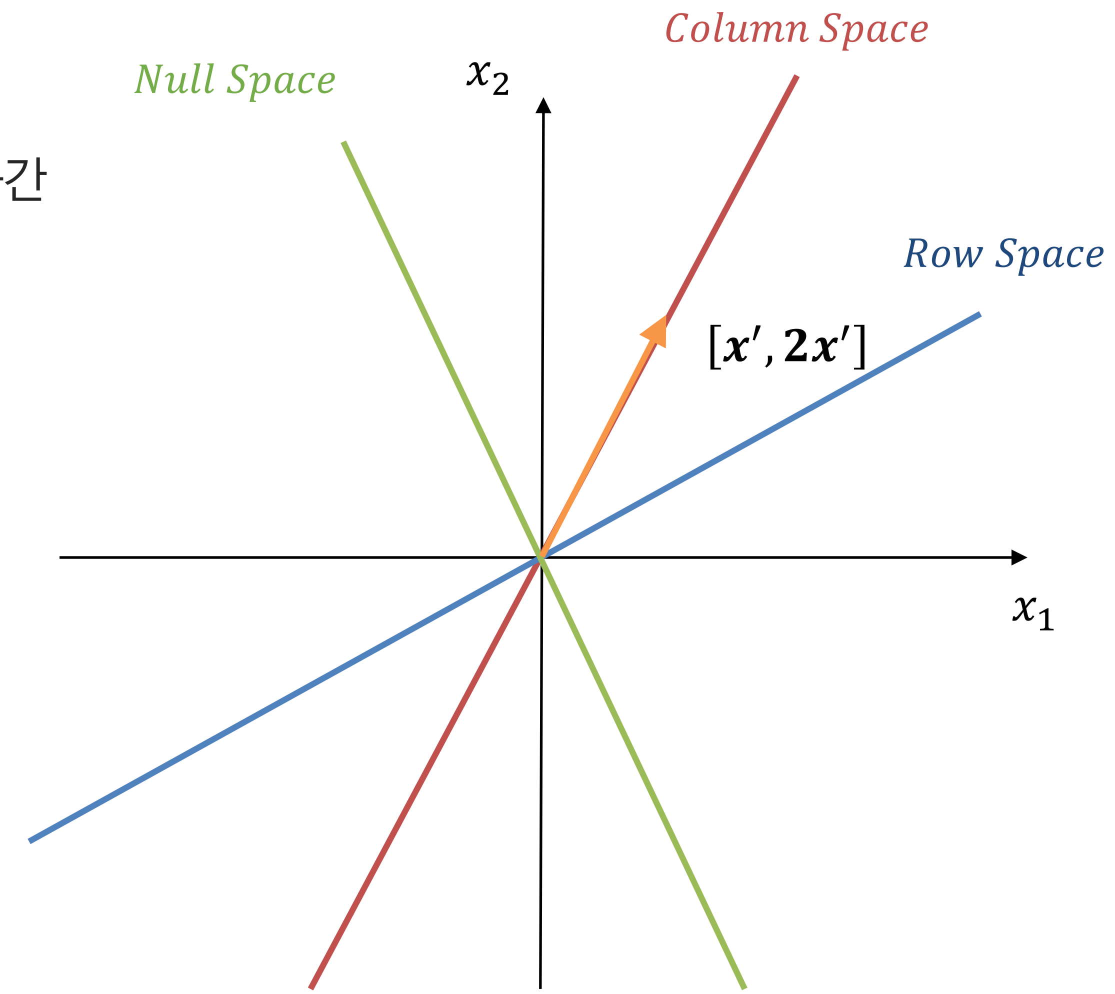
Paper 3: Vim

Background: 행렬을 통한 선형 변환과 함수

- Row space : row vector로 표현할 수 있는 벡터 공간
- Column space : column vector로 표현할 수 있는 벡터 공간
- Null space : $Ax = 0$을 만족하는 모든 벡터 x들의 집합

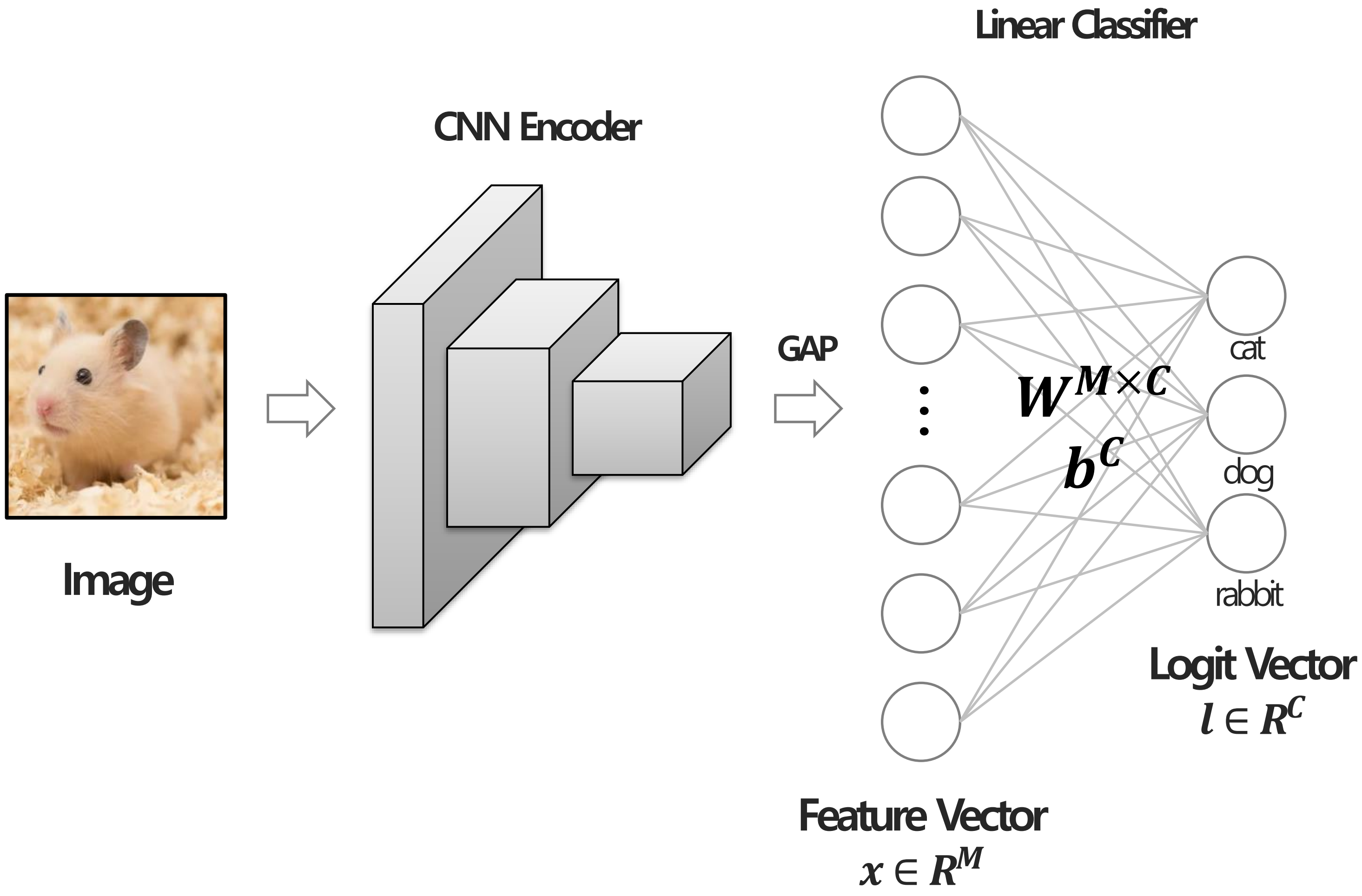
$$\boldsymbol{Ax = y} \rightarrow \begin{bmatrix} 2 & 1 \\ 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2x_1 + x_2 \\ 4x_1 + 2x_2 \end{bmatrix}$$

$$\boldsymbol{Let, 2x_1 + x_2 = x', then\ Ax = \begin{bmatrix} x' \\ 2x' \end{bmatrix}}$$



Paper 3: Vim

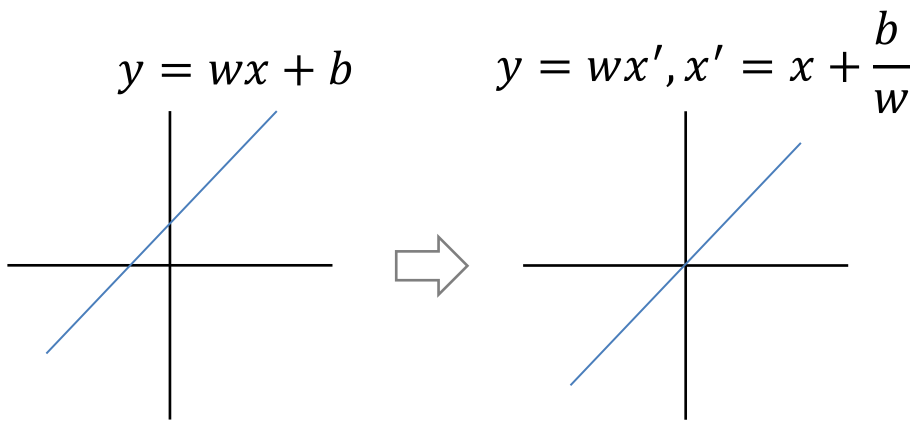
Background: Linear Classifier와 벡터 공간



$$l = W^T x + b$$

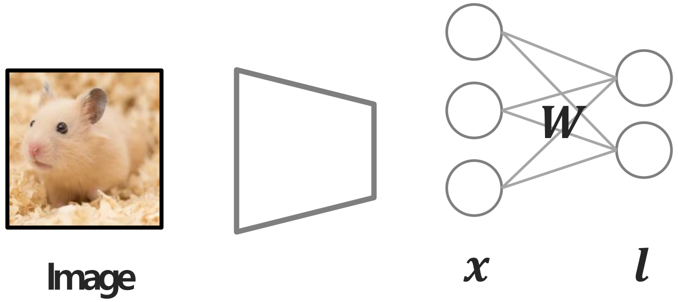
$$= W^T(x - o) = W^T x'$$

$$where\ o = -\left(W^T\right)^+ b$$



Paper 3: Vim

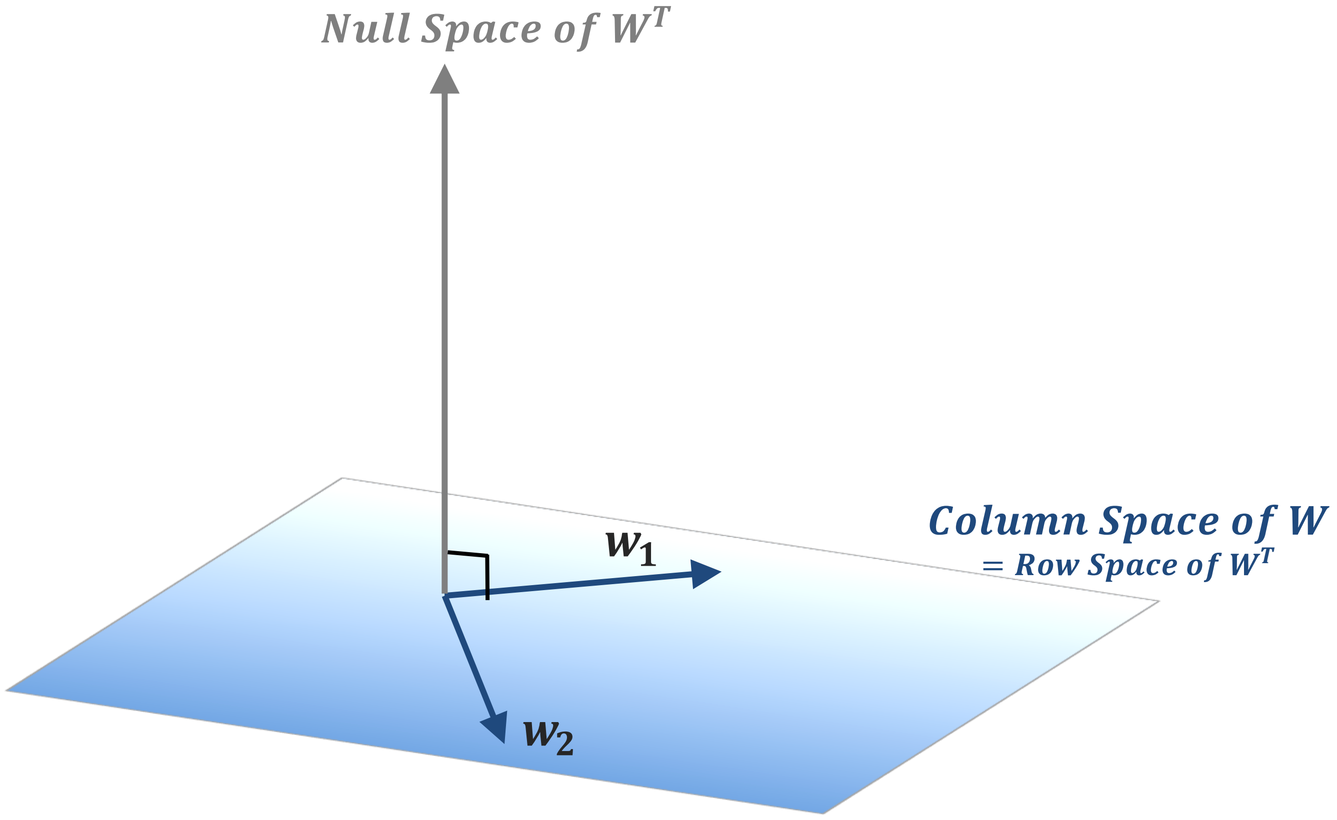
Background: Linear Classifier와 벡터 공간



Feature Vector $x = [x_1\ x_2\ x_3]^T \in R^3$

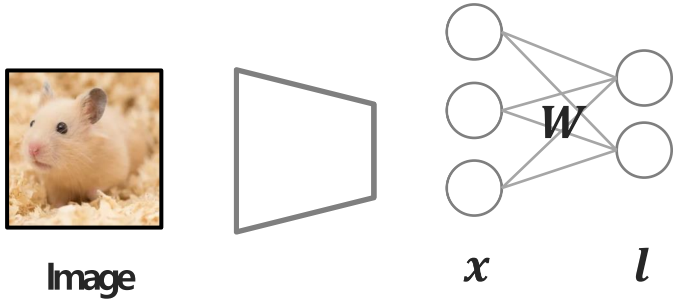
Logit Vector $l = [l_1\ l_2]^T \in R^2$

$$W^{3\times 2} = \begin{bmatrix} | & | \\ w_1 & w_2 \\ | & | \end{bmatrix}$$



Paper 3: Vim

Background: NuSA(Outlier Detection through Null Space Analysis of Neural Networks)

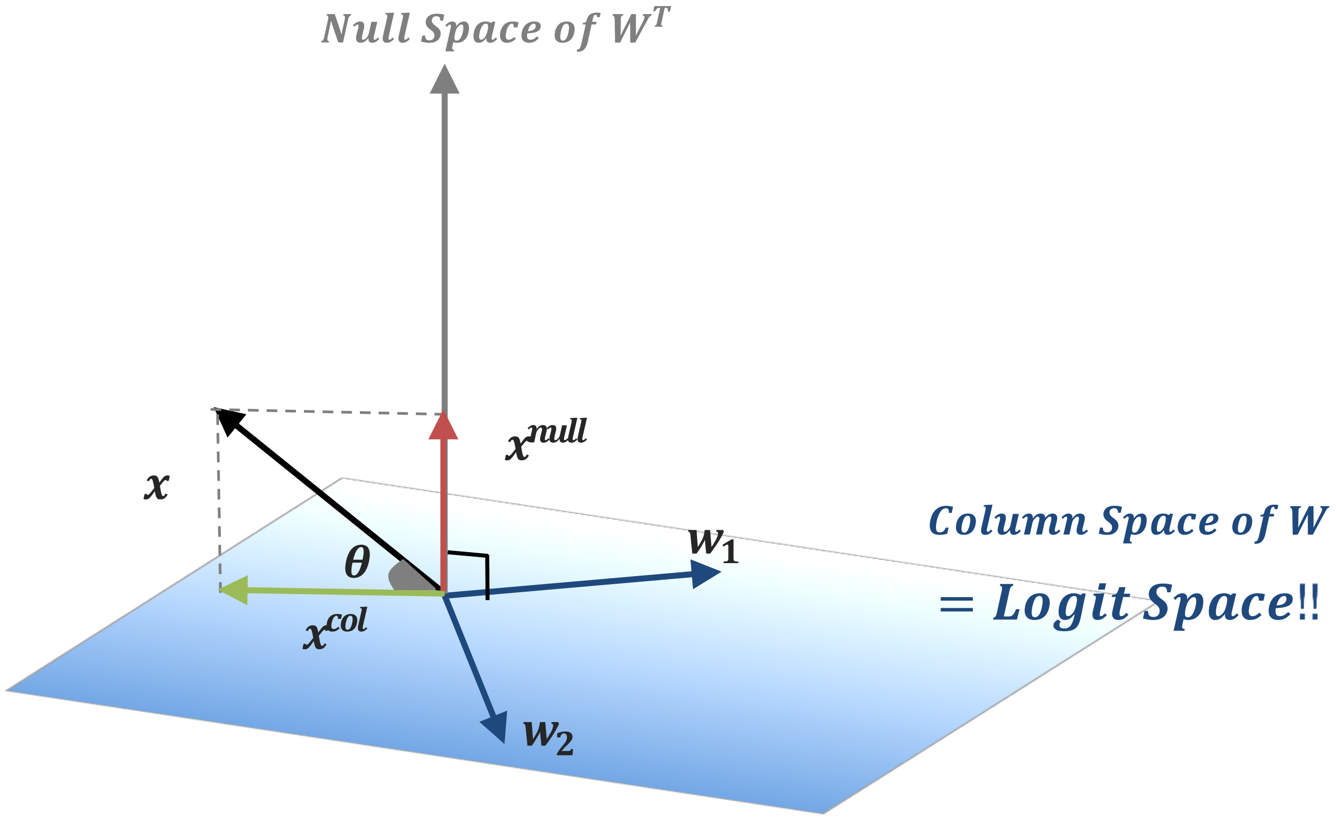


Feature Vector $x = x^{col} + x^{null}$

Logit Vector $l = [l_1\ l_2]^T \in R^2$

$$W^{3\times2} = \begin{bmatrix} | & | \\ w_1 & w_2 \\ | & | \end{bmatrix}$$

$$NuSA(x) = \frac{\|x^{col}\|}{\|x\|}$$



Paper 3: Vim

Method

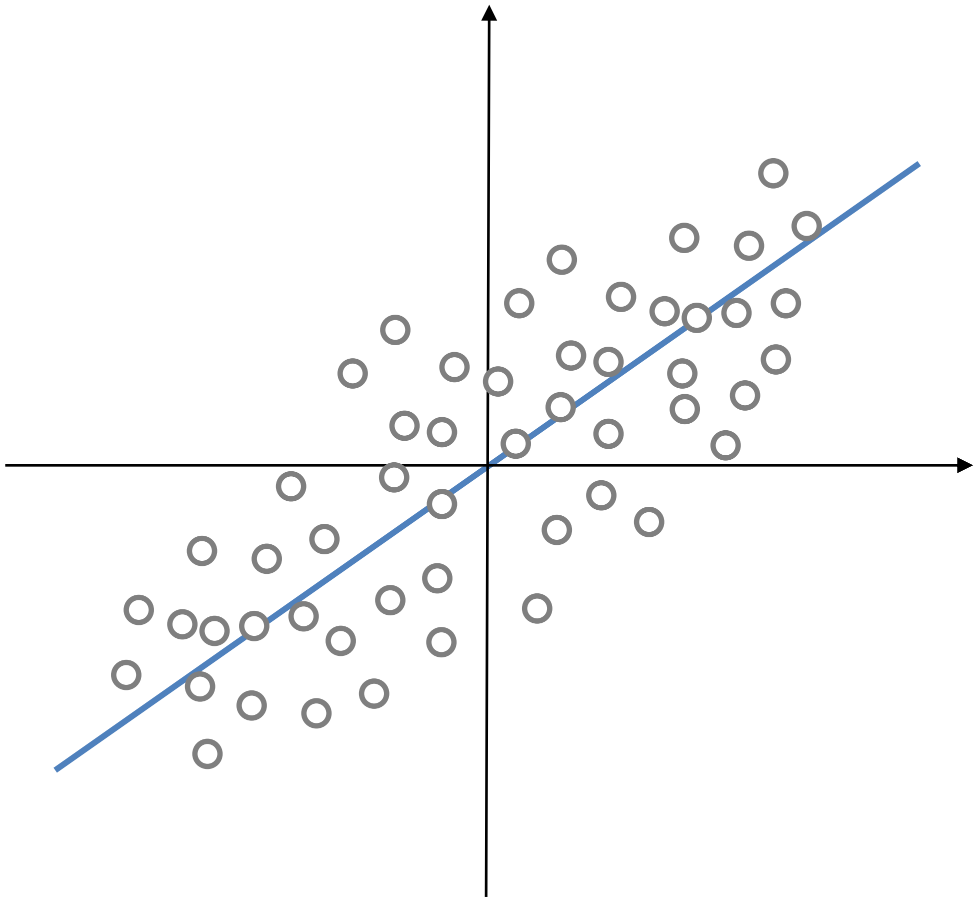
$$NuSA: \ Column\ Space \ \rightarrow ViM: \ Principal\ Space$$

Paper 3: Vim

Method

$$NuSA : \ Column\ Space \ \rightarrow ViM : \ Principal\ Space$$

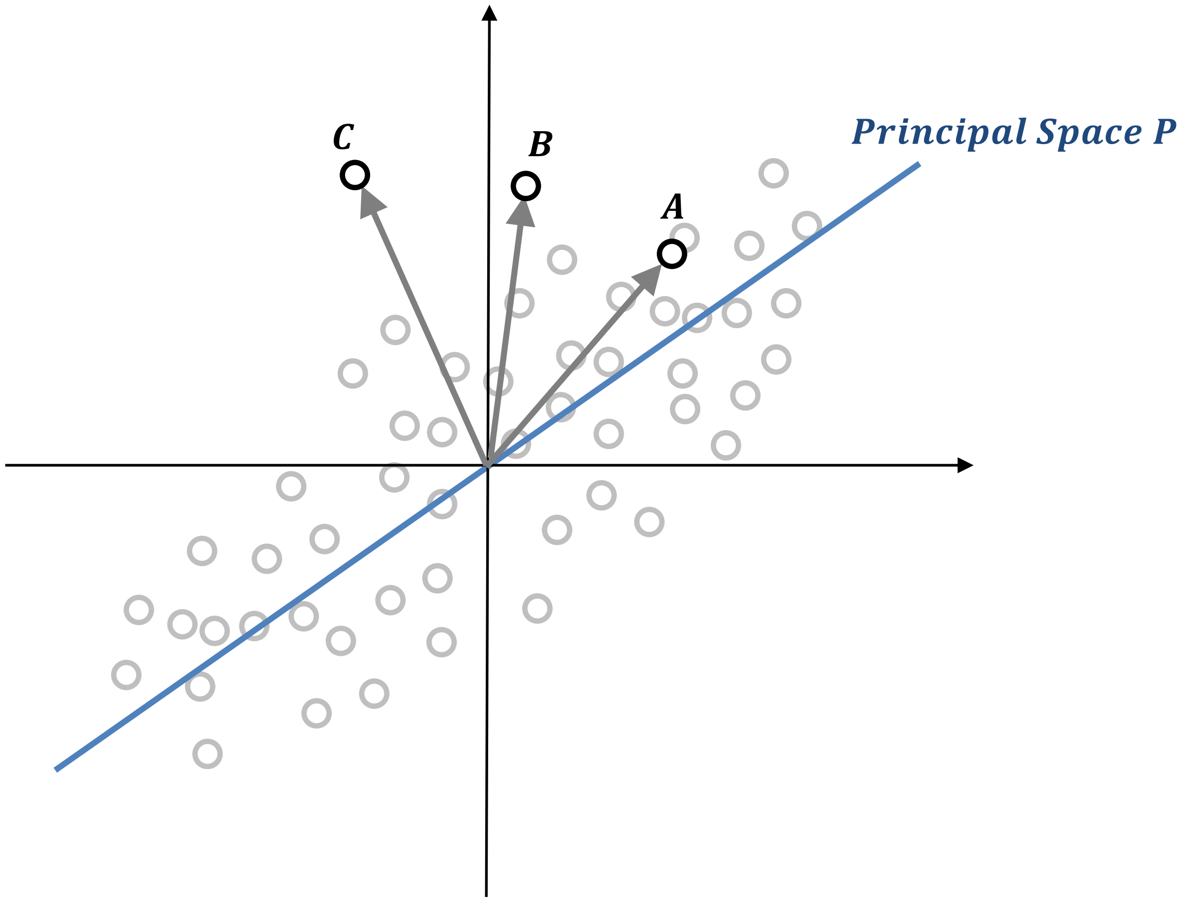
$$X^T X = QDQ^{-1}$$



Paper 3: Vim

Method

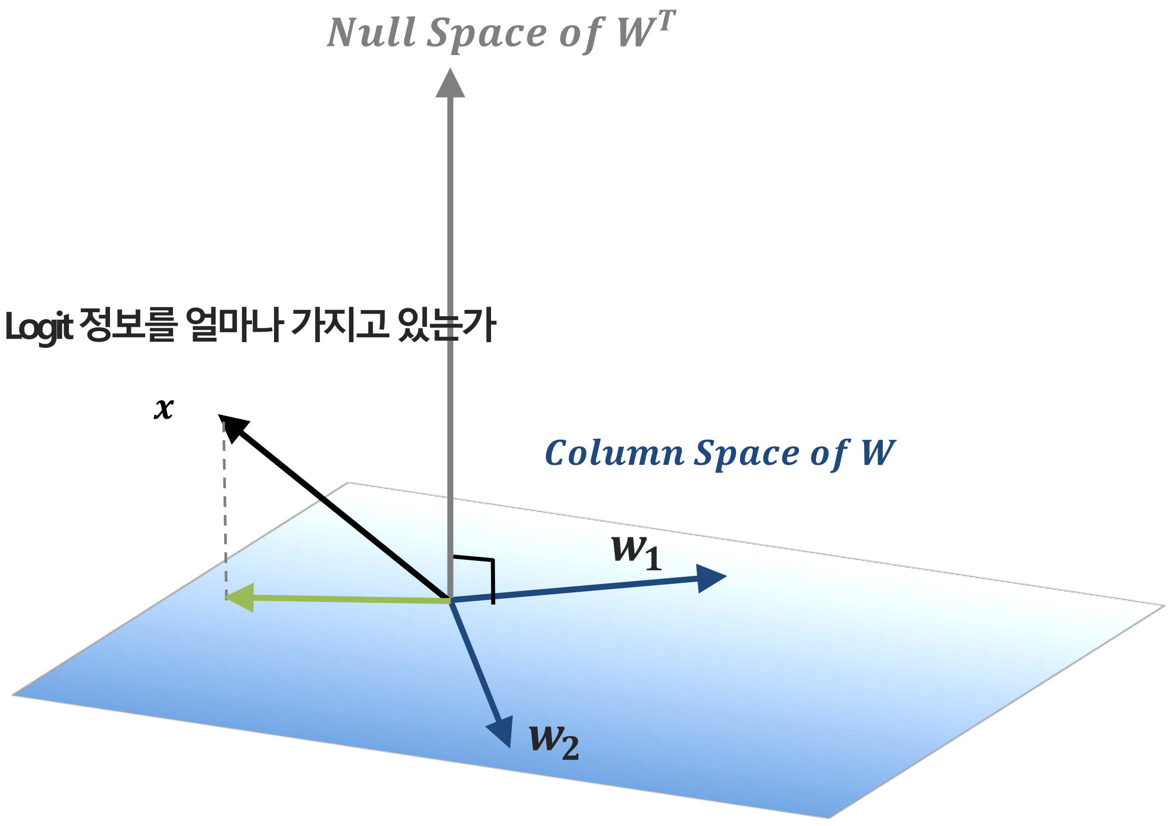
$$NuSA: \ Column\ Space \ \rightarrow ViM: \ Principal\ Space$$

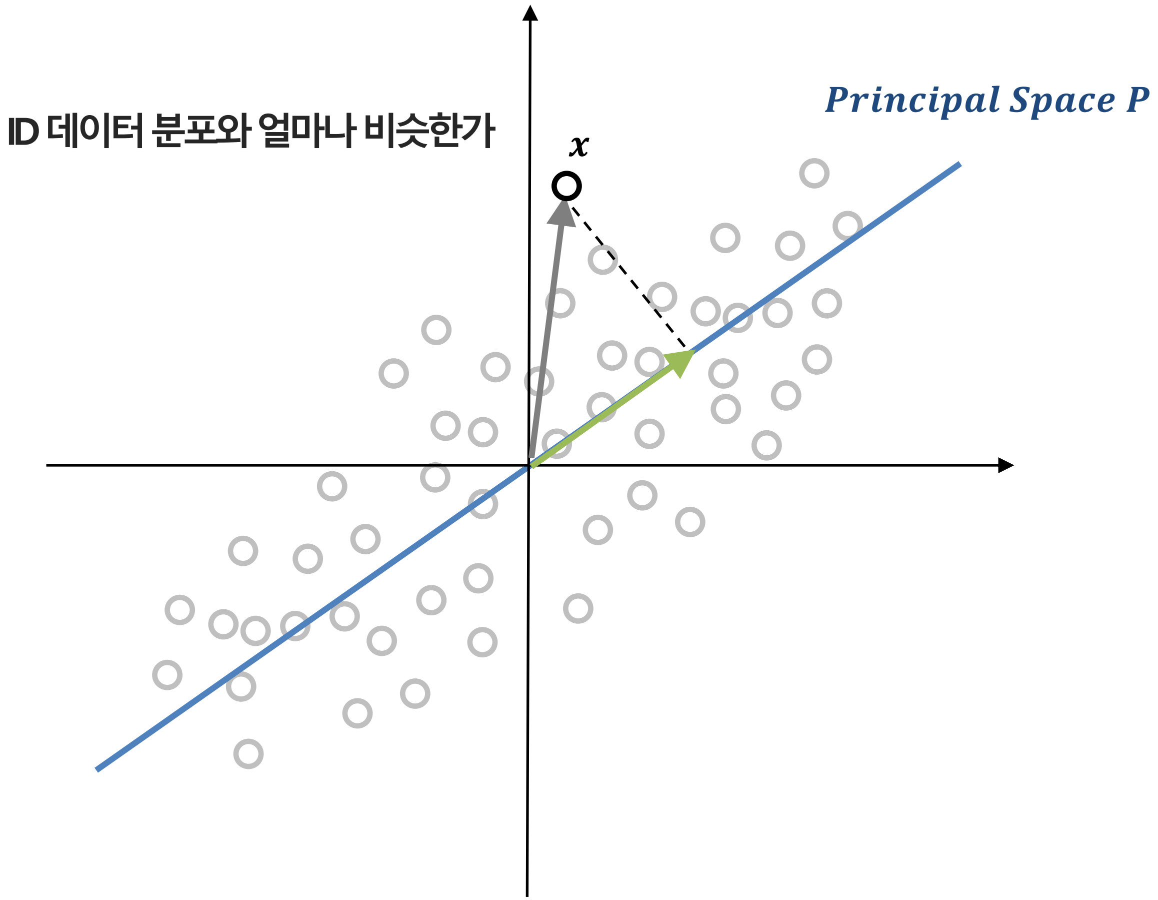


Paper 3: Vim

Method

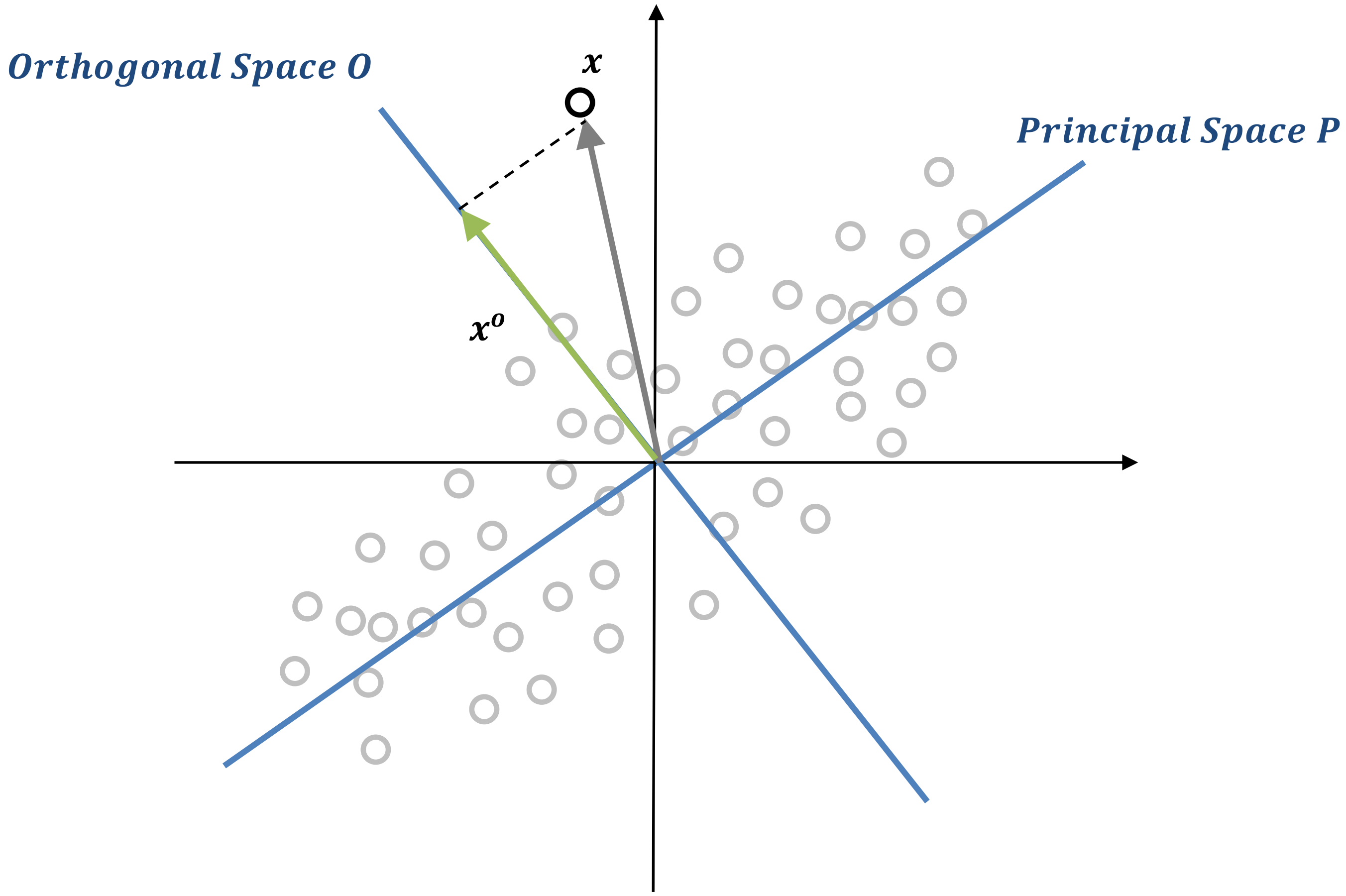
$$NuSA: \ Column\ Space \ \rightarrow ViM: \ Principal\ Space$$





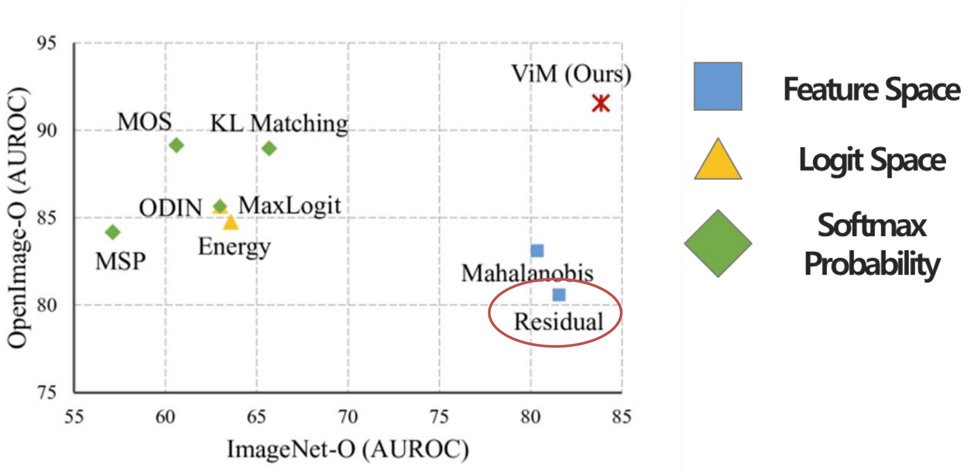
Paper 3: Vim

Method



ID 데이터 분포와 얼마나 떨어져 있는가

$$Residual(x) = \|x^o\|$$



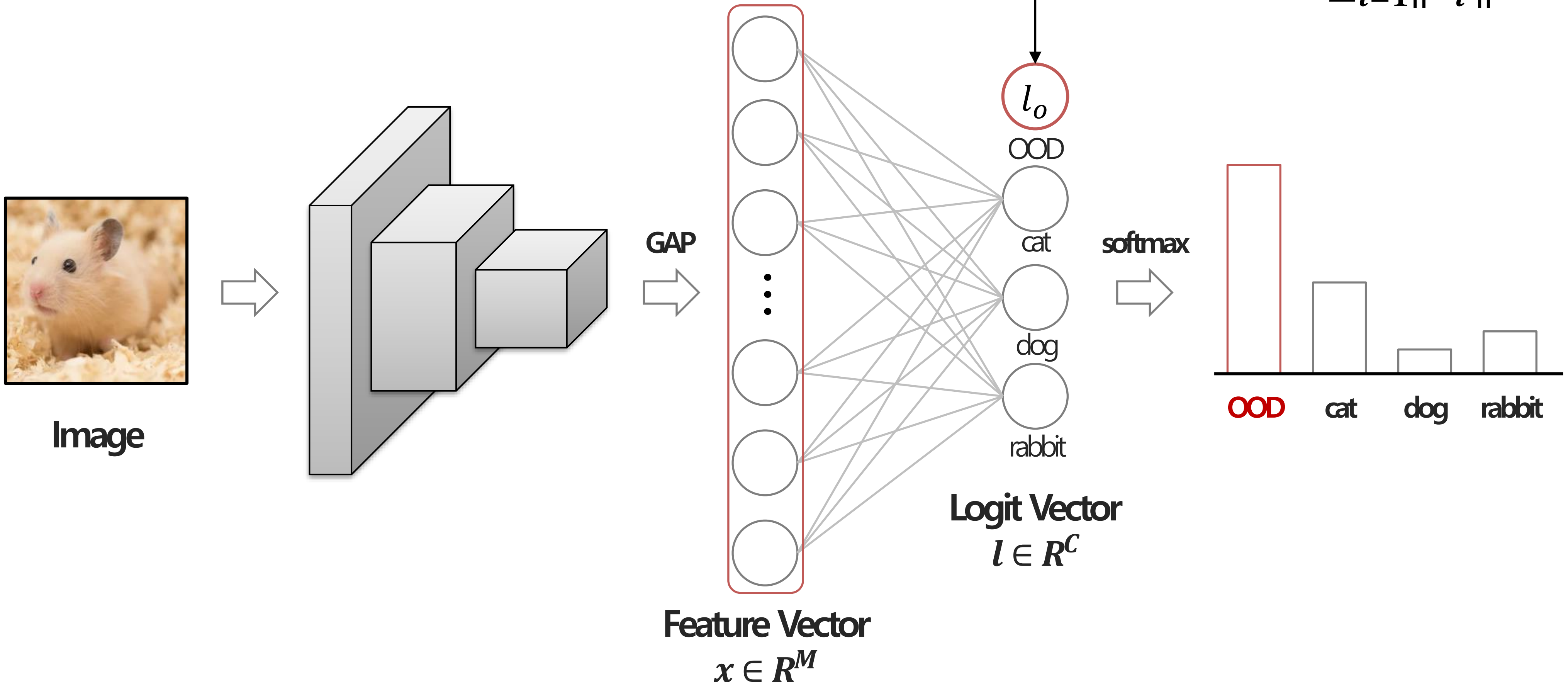
Paper 3: Vim

Method

ID 데이터 분포와 얼마나 떨어져 있는가

$$X^T X = QDQ^{-1} \longrightarrow Space\ P\&O \longrightarrow Residual(x) = \|x^o\|$$

$$\times\ \alpha \qquad \alpha = \frac{\sum_{i=1}^{K} max_{j=1,\dots,C}\{l_j^i\}}{\sum_{i=1}^{K}\|x_i^o\|}$$



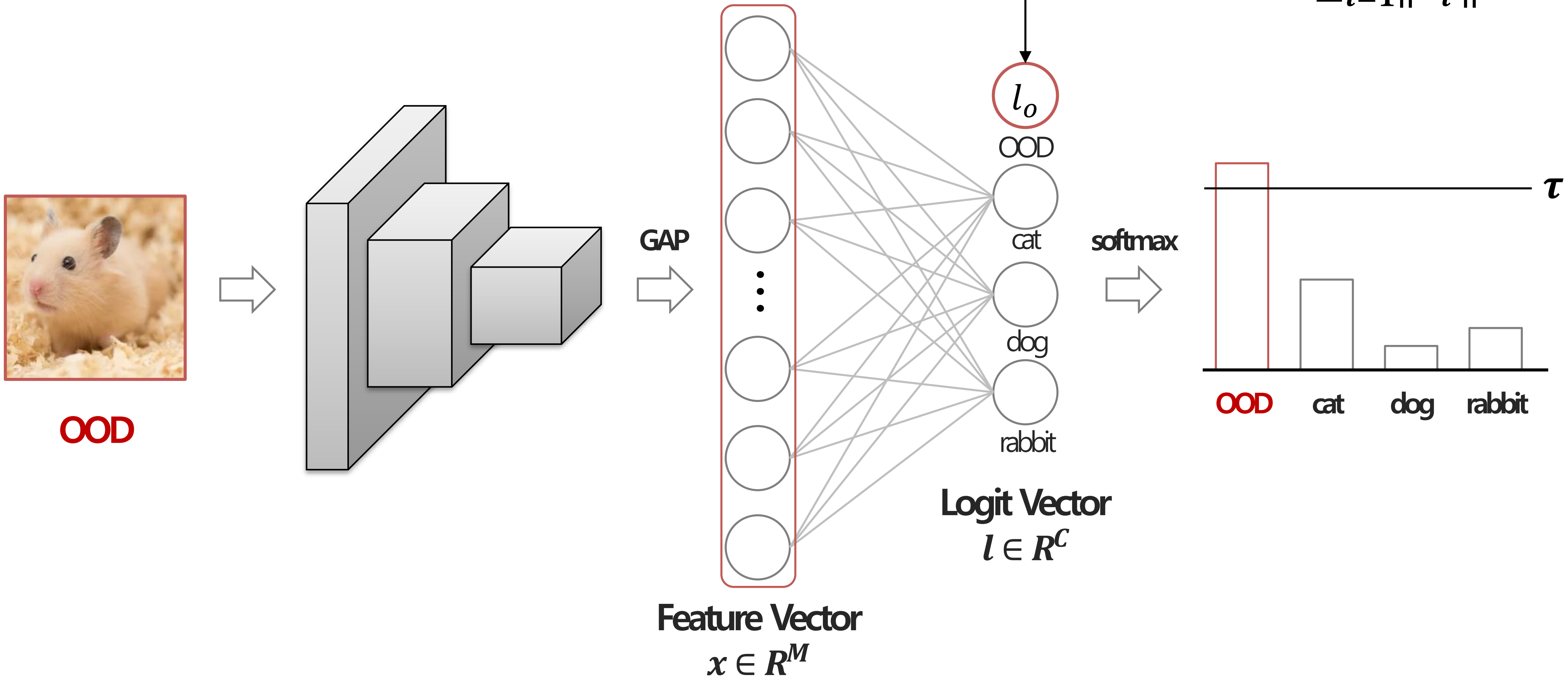
Paper 3: Vim

Method

ID 데이터 분포와 얼마나 떨어져 있는가

$$X^T X = QDQ^{-1} \longrightarrow Space\ P\&O \longrightarrow Residual(x) = \|x^o\| \quad \times \alpha$$

$$\alpha = \frac{\sum_{i=1}^{K} max_{j=1,\dots,C}\{l_j^i\}}{\sum_{i=1}^{K} \|x_i^o\|}$$



Paper 3: Vim

Experimental Results

Model	Method	Source	OpenImage-O AUROC↑	OpenImage-O FPR95↓	Texture AUROC↑	Texture FPR95↓	iNaturalist AUROC↑	iNaturalist FPR95↓	ImageNet-O AUROC↑	ImageNet-O FPR95↓	Average AUROC↑	Average FPR95↓
BiT	MSP [13]	prob	84.16	73.72	79.80	76.65	87.92	64.09	57.12	96.85	77.25	77.83
	Energy [25]	logit	84.77	73.42	81.09	73.91	84.47	74.98	63.59	96.40	78.48	79.68
	ODIN [24]	prob+grad	85.64	72.83	81.60	74.07	86.73	70.75	63.00	96.85	79.24	78.63
	MaxLogit [12]	logit	85.67	72.68	81.66	73.72	86.76	70.59	63.01	96.85	79.27	78.46
	KL Matching [12]	prob	88.96	51.51	86.92	51.05	**92.95**	**33.28**	65.68	86.65	83.63	55.62
	Residual†	feat	80.58	67.85	97.66	11.16	76.76	80.41	81.57	65.50	84.14	56.23
	ReAct [32]	feat+logit	88.94	54.97	90.64	50.25	91.45	48.60	67.07	91.70	84.53	61.38
	Mahalanobis [23]	feat+label	83.10	64.32	97.33	14.05	85.70	64.95	80.37	70.05	86.62	53.34
	ViM (Ours)	feat+logit	**91.54**	**43.96**	**98.92**	**4.69**	89.30	55.71	**83.87**	**61.50**	**90.91**	**41.46**
ViT	MSP [13]	prob	92.53	34.18	87.10	48.55	96.11	19.04	81.86	64.85	89.40	41.65
	Energy [25]	logit	97.11	14.04	93.39	28.22	98.66	6.16	90.46	41.30	94.90	22.43
	ODIN [24]	prob+grad	96.86	15.68	93.01	30.60	98.57	6.58	89.85	44.15	94.57	24.25
	MaxLogit [12]	logit	96.87	15.68	93.01	30.60	98.57	6.58	89.85	44.15	94.57	24.25
	KL Matching [12]	prob	93.80	28.49	88.76	44.09	96.88	14.79	84.12	55.70	90.89	35.77
	Residual†	feat	92.72	32.63	92.21	33.80	98.57	6.63	88.23	47.85	92.93	30.23
	ReAct [32]	feat+logit	97.38	13.50	93.34	28.49	99.00	4.31	90.71	42.60	95.11	22.22
	Mahalanobis [23]	feat+label	97.48	13.54	94.24	25.17	**99.54**	**2.12**	**92.81**	36.95	96.02	19.45
	ViM (Ours)	feat+logit	**97.61**	**12.61**	**95.34**	**20.31**	99.41	2.60	92.55	**36.75**	**96.23**	**18.07**

Paper 3: Vim

Experimental Results

$$\alpha = \frac{\sum_{i=1}^{K} max_{j=1,\dots,C}\{l_j^i\}}{\sum_{i=1}^{K}\|x_i^o\|}$$

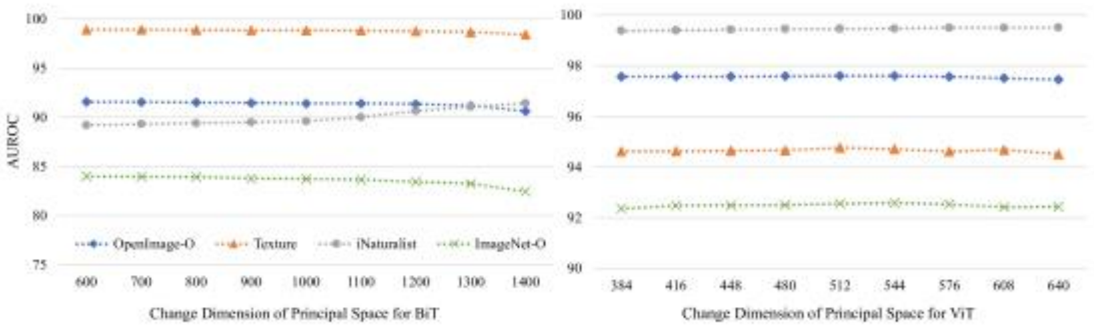


Figure 4. Robustness against principal space dimension. Left is BiT and right is ViT. The performance changes are small when D varies in a wide range of values.

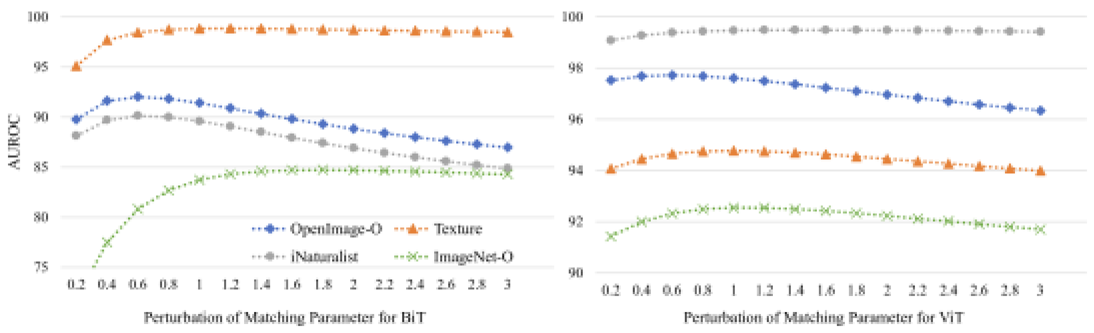


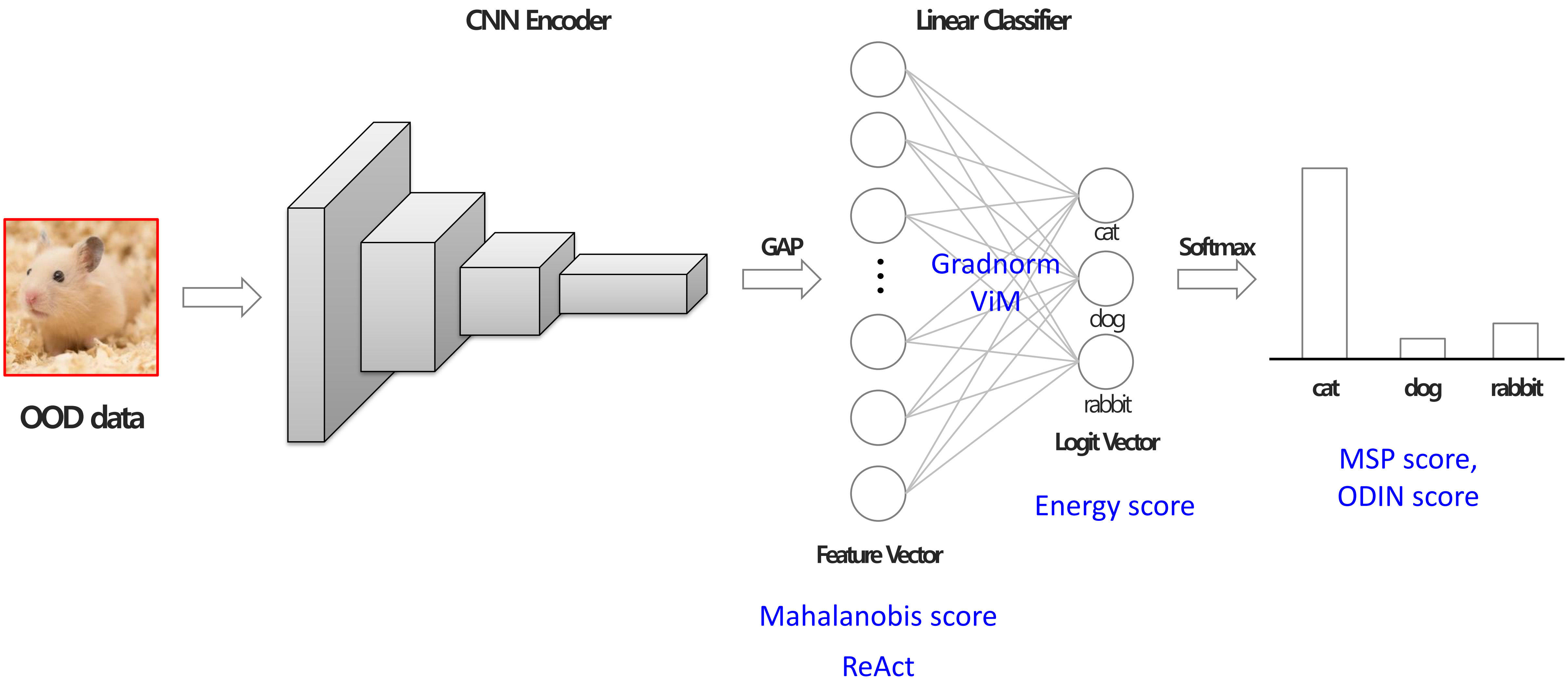
Figure 5. Perturbation of α by multiplying a factor. Left is BiT, and right is ViT. For both models, the proposed matching parameter fits well for the trends.

MOOD: Multi-Level Out-Of-Distribution Detection

(2021, NeurIPS)

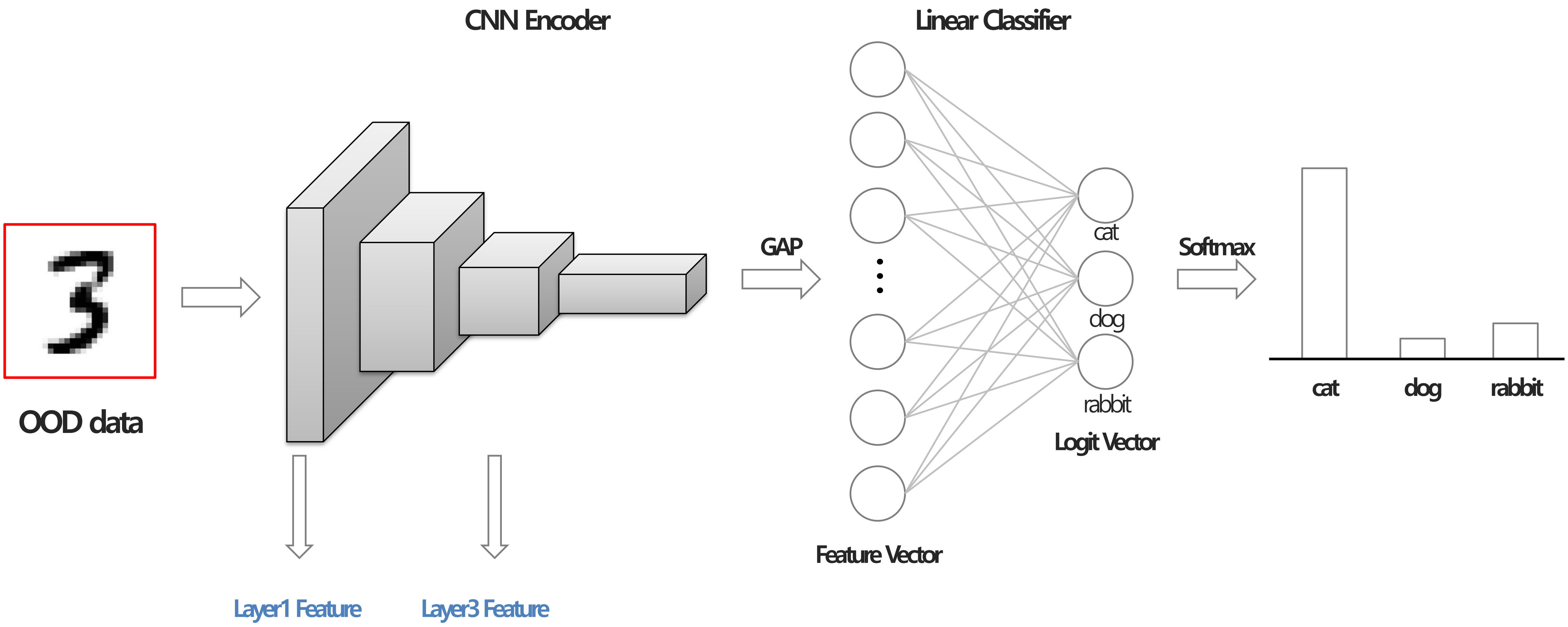
Paper 4: MOOD

Introduction



Paper 4: MOOD

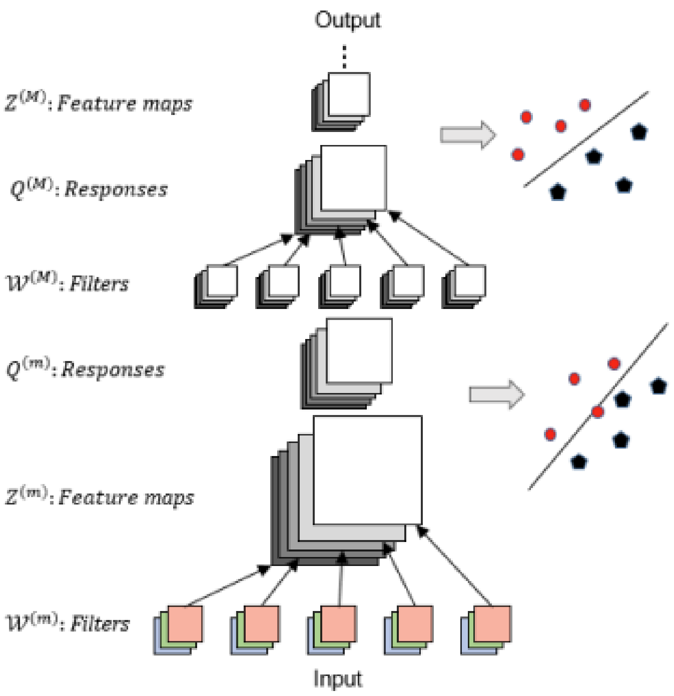
Introduction



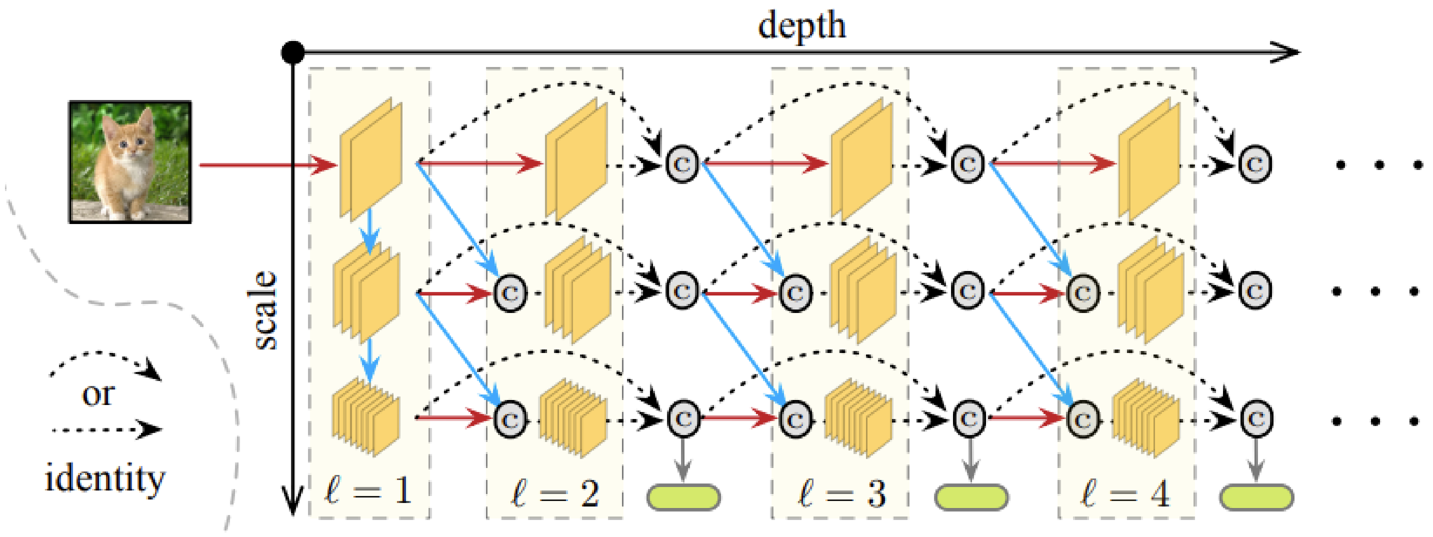
Paper 4: MOOD

Background

Adaptive Inference Networks



Deeply-Supervised Nets

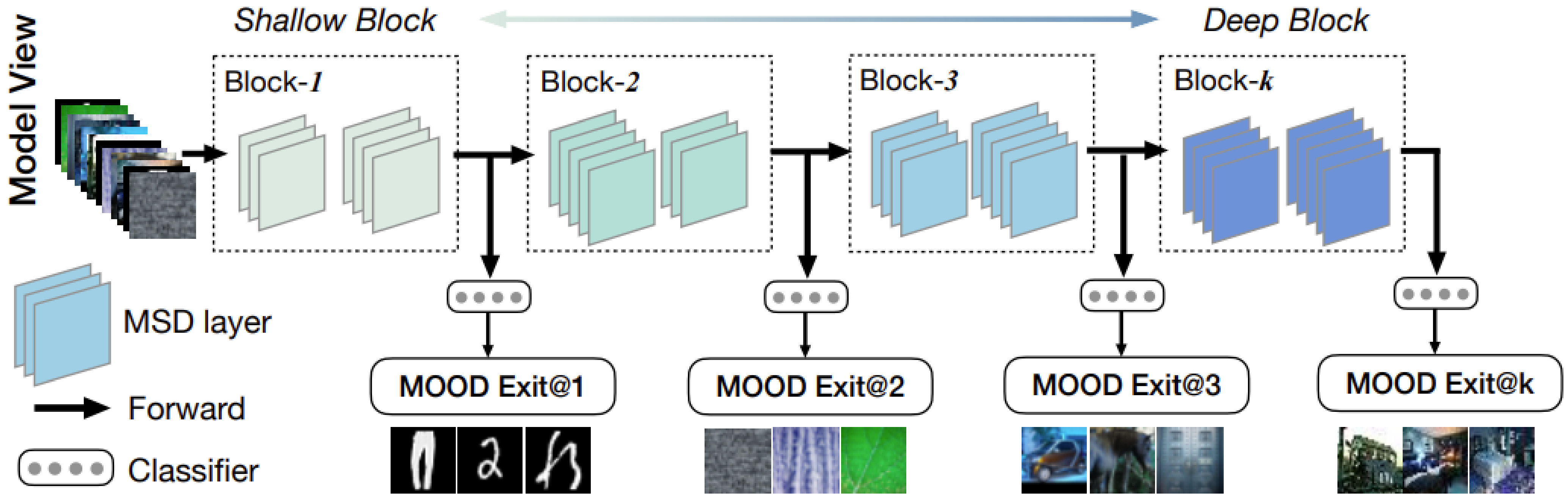


Multi-Scale Dense Networks

+ OOD Detection = MOOD

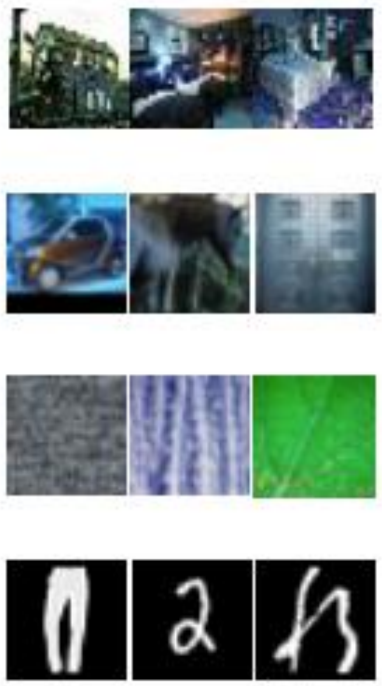
Paper 4: MOOD

Method

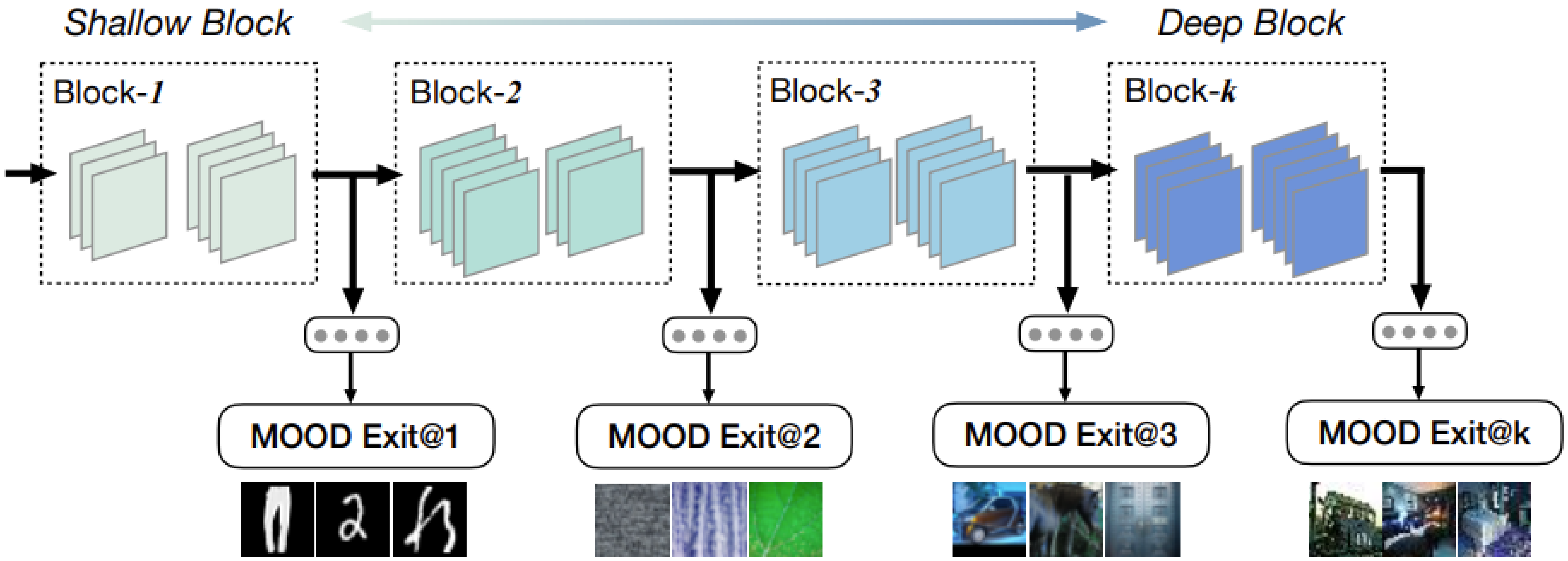


Paper 4: MOOD

Method

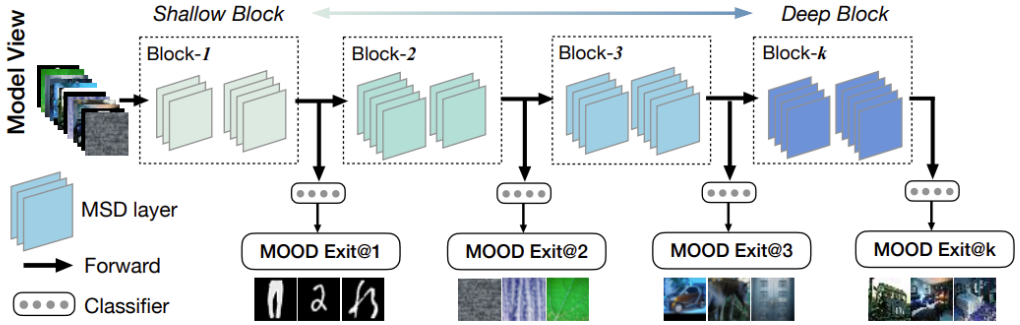


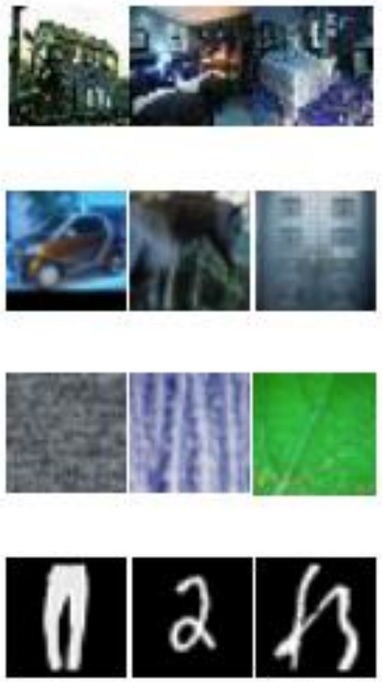
1. 어떻게 최적의 Exit를 할당할 것인가? = 어떻게 데이터의 복잡도를 정량화 할 것인가?



Paper 4: MOOD

Method





Lossless Compression Algorithm c

PNG

JPEG

복잡도 $L(x)$ = 이미지 용량 $Bit\,lenth(c(x))$

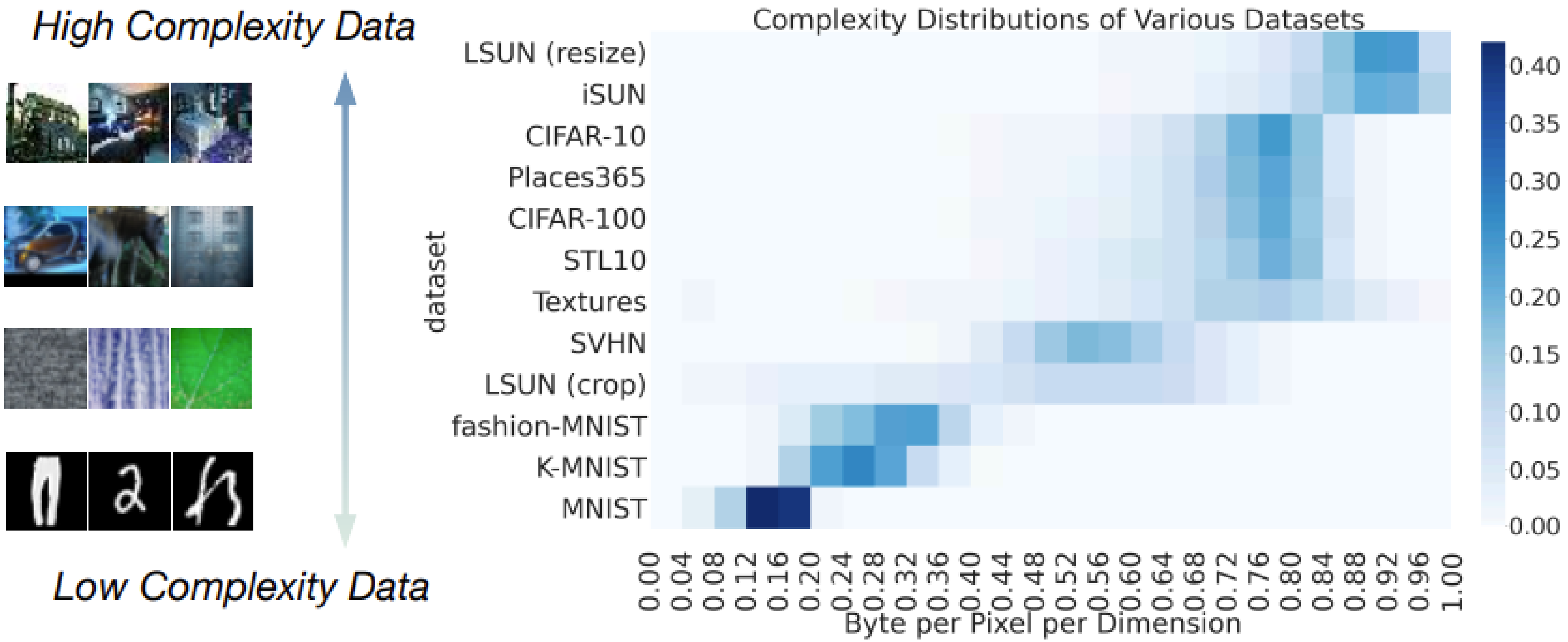
복잡도 $L_{normalized} = L(x)/L_{max}$

Exit 1	Exit 2	Exit 3	…	Exit k

0 1

Paper 4: MOOD

Method



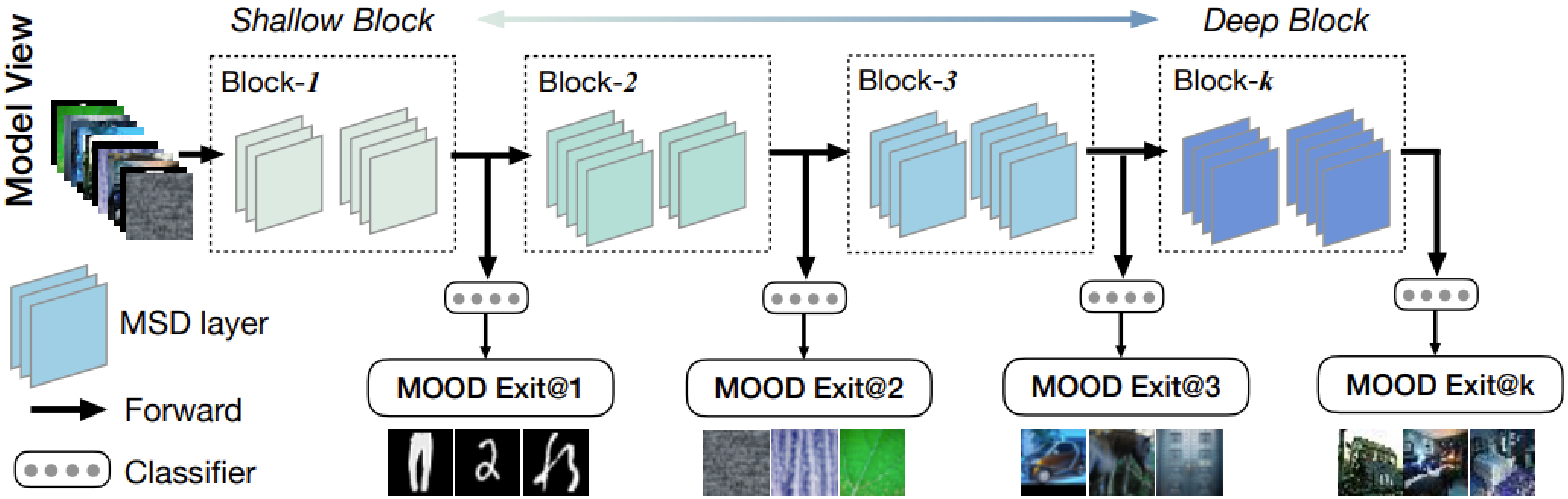
복잡도 $L_{normalized}$

Paper 4: MOOD

Method

1. 어떻게 최적의 Exit를 할당할 것인가?

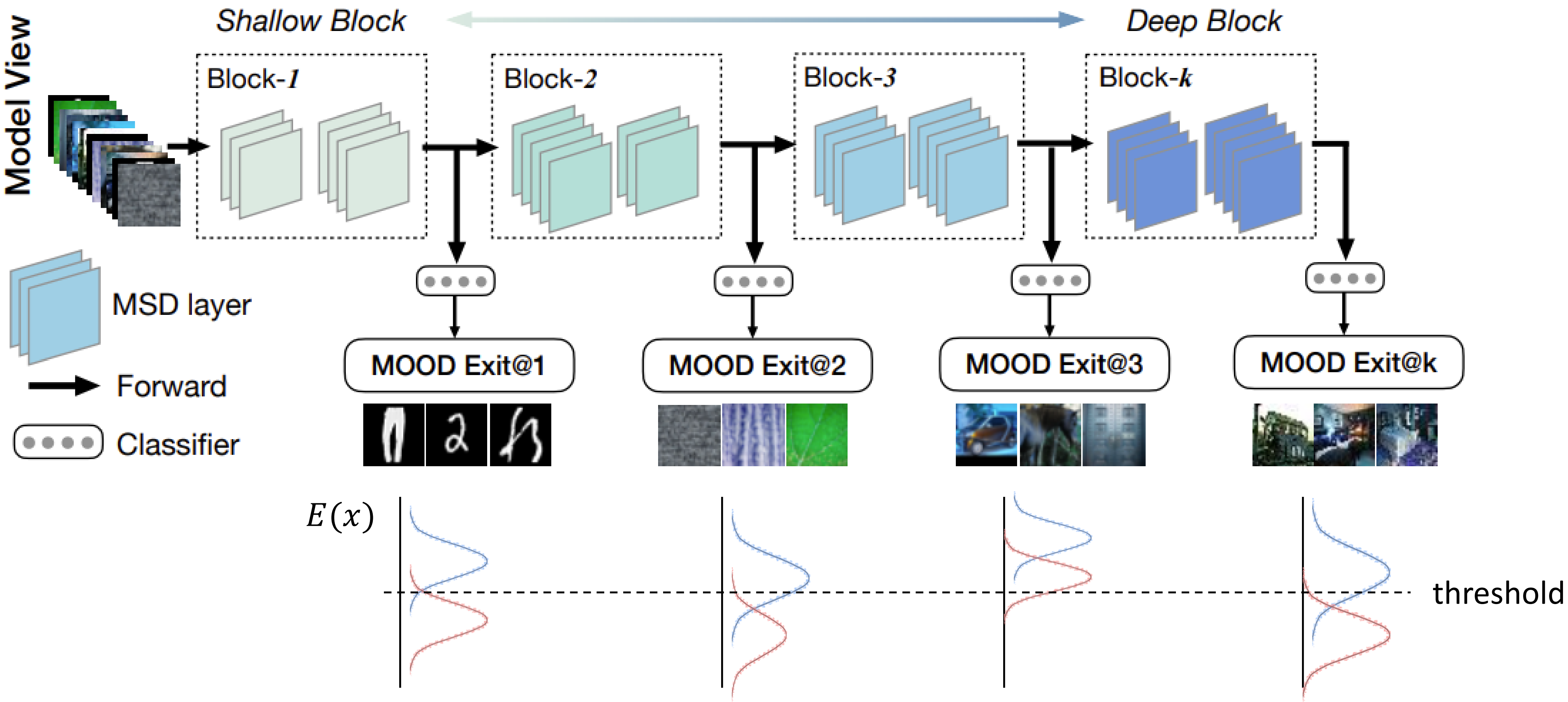
2. 어떻게 OOD를 탐지할 것인가?



Paper 4: MOOD

Method

$$energy\ score\ E(x) = -T * \log \sum^{K} e^{f_j(x)/T}$$



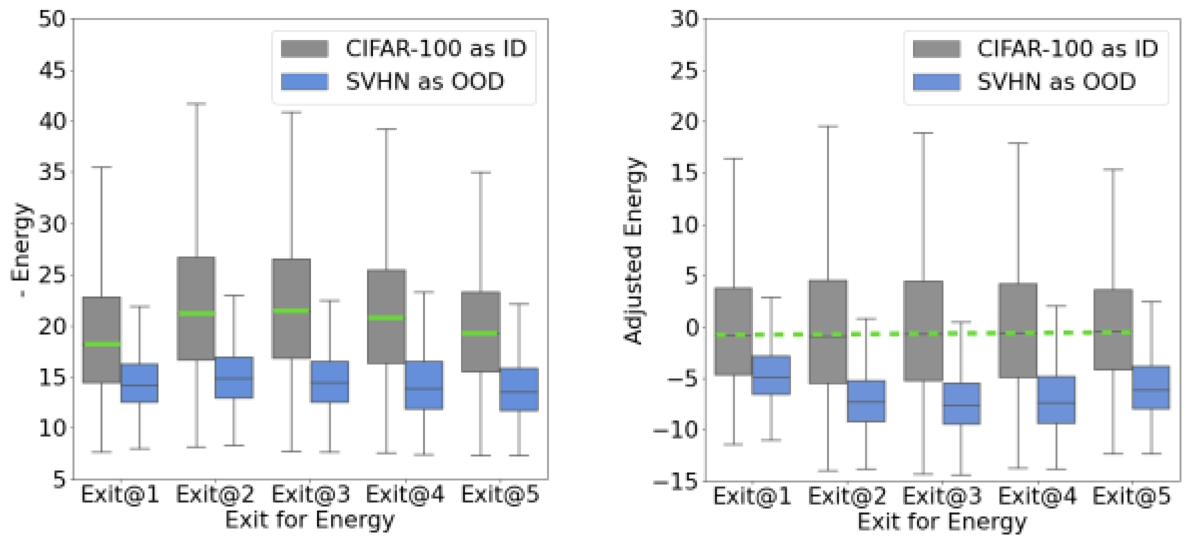
Paper 4: MOOD

Method

$$energy\ score\ E(x) = -T * \log \sum_{j=1}^{K} e^{f_j(x)/T}$$

$$adjusted\ energy\ score$$

$$E_{adjusted}(x;\theta_i) = E(x;\theta_i) - \mathbb{E}_{x \in D_{in}}[-E(x;\theta_i)]$$



(a) Energy Score [32] (b) Adjusted Energy Score (ours)

Figure 2. *Left*: Energy scores are not comparable across exits. *Right*: Adjusted energy scores are more comparable across exits (shown in dashed green line).

Paper 4: MOOD

Experimental Results

In-distribution (ID)	Architecture	Method	FLOPs $\downarrow (\times 10^8)$	AUROC $\uparrow$	FPR95 $\downarrow$	ID Acc $\uparrow$ (%)
CIFAR-10	WideResNet-40-4	MSP [13]	13.00	0.8898	0.5681	94.93
		ODIN [31]	13.00	0.9011	0.3531	94.93
		Mahalanobis [28]	13.00	0.8933	0.3548	94.93
		Energy [32]	13.00	0.9004	0.3526	94.93
	MSDNet Exit@last	MSP [13]	1.05	0.8972	0.4987	94.09
		ODIN [31]	1.05	0.9033	0.2930	94.09
		Mahalanobis [28]	1.05	0.8284	0.7519	94.09
		Energy [32]	1.05	0.9048	0.3362	94.09
	MSDNet (dynamic exit)	MOOD (*ours*)	**0.79**	**0.9126** ($\pm$0.0016)	**0.2805** ($\pm$0.0051)	94.13
CIFAR-100	WideResNet-40-4	MSP [13]	13.00	0.7710	0.7751	76.90
		ODIN [31]	13.00	0.8466	0.5722	76.90
		Mahalanobis [28]	13.00	0.8319	0.5352	76.90
		Energy [32]	13.00	0.8369	0.6271	76.90
	MSDNet Exit@last	MSP [13]	1.05	0.7833	0.7671	75.43
		ODIN [31]	1.05	0.8489	0.5745	75.43
		Mahalanobis [27]	1.05	0.7380	0.7806	75.43
		Energy [32]	1.05	0.8451	0.5915	75.43
	MSDNet (dynamic exit)	MOOD (*ours*)	**0.79**	**0.8497** ($\pm$0.0026)	**0.5722** ($\pm$0.0068)	75.26

Paper 4: MOOD

Experimental Results

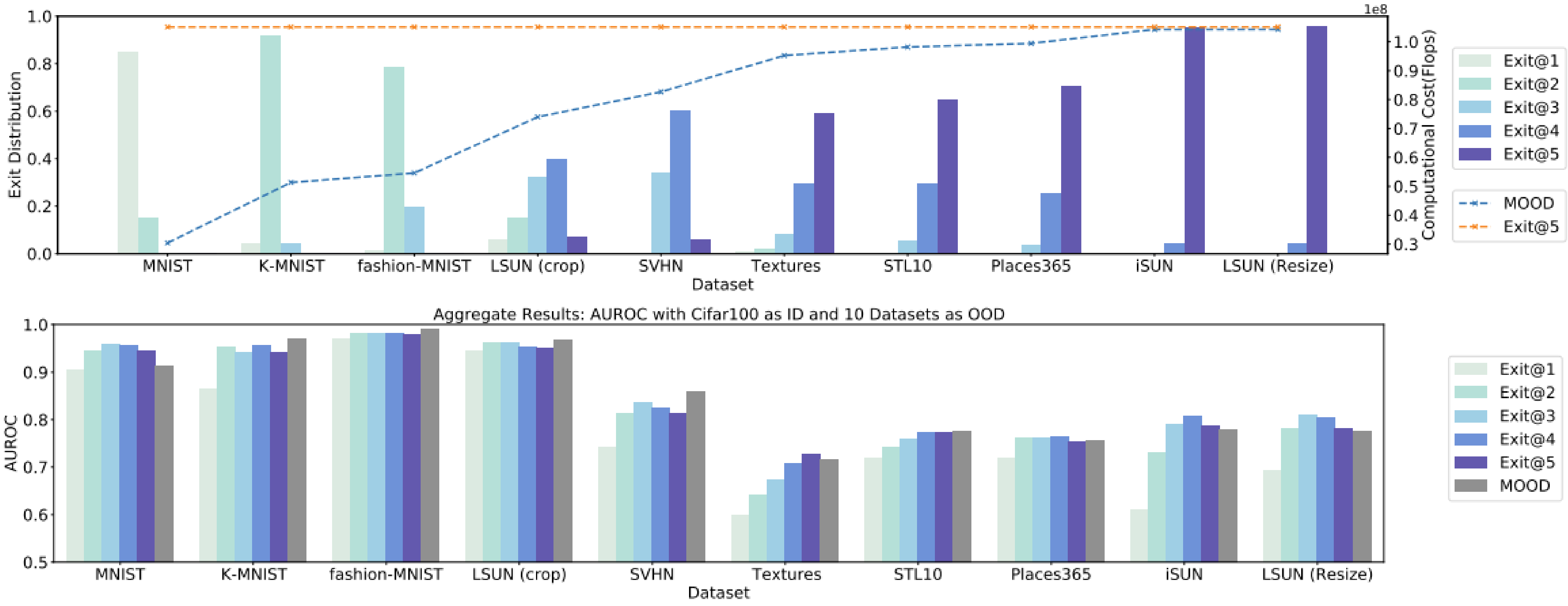
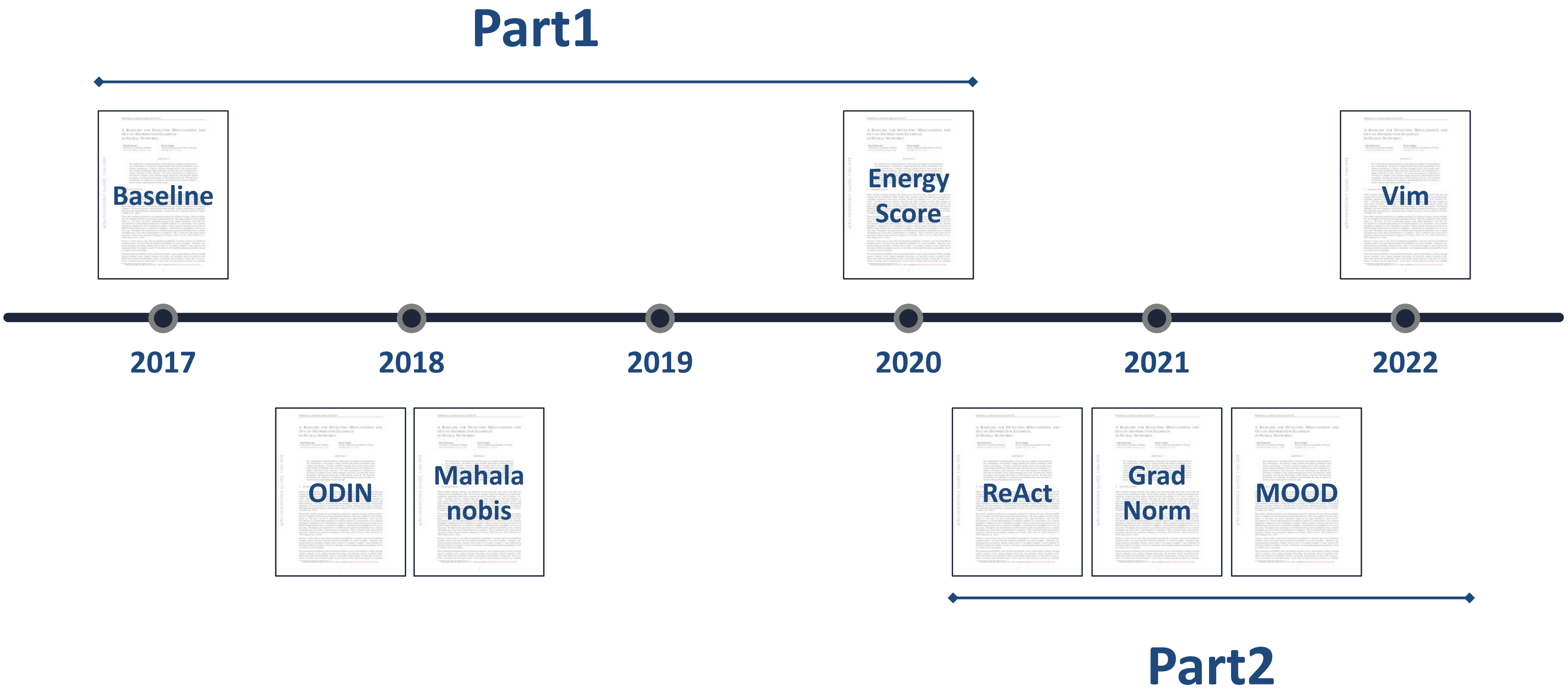


Figure 3. *Top*: Average computational cost (FLOPs) with MOOD, and the normalized frequency distribution of exits chosen by MOOD. Gap in between the orange and blue lines indicates the computational savings. *Bottom*: Average AUROC by taking constant exits at different levels. Model is trained on CIFAR-100 as in-distribution, and evaluated on 10 OOD test datasets described in Section 3.1.

Timeline


Part1
Baseline
Energy Score
Vim
2017
2018
2019
2020
2021
2022
ODIN
Mahalanobis
ReAct
GradNorm
MOOD
Part2

Reference

Review Papers

1. Sun, Y., Guo, C., & Li, Y. (2021). React: Out-of-distribution detection with rectified activations. Advances in Neural Information Processing Systems, 34, 144-157.
2. Huang, R., Geng, A., & Li, Y. (2021). On the importance of gradients for detecting distributional shifts in the wild. Advances in Neural Information Processing Systems, 34, 677-689.
3. Lin, Z., Roy, S. D., & Li, Y. (2021). Mood: Multi-level out-of-distribution detection. In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition (pp. 15313-15323).
4. Wang, H., Li, Z., Feng, L., & Zhang, W. (2022). Vim: Out-of-distribution with virtual-logit matching. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4921-4930).

감사합니다