Enhancing Multiple Reliability Measures via Nuisance-extended Information Bottleneck

Jongheon Jeong[1] Sihyun Yu[1] Hankook Lee[2] Jinwoo Shin[1]

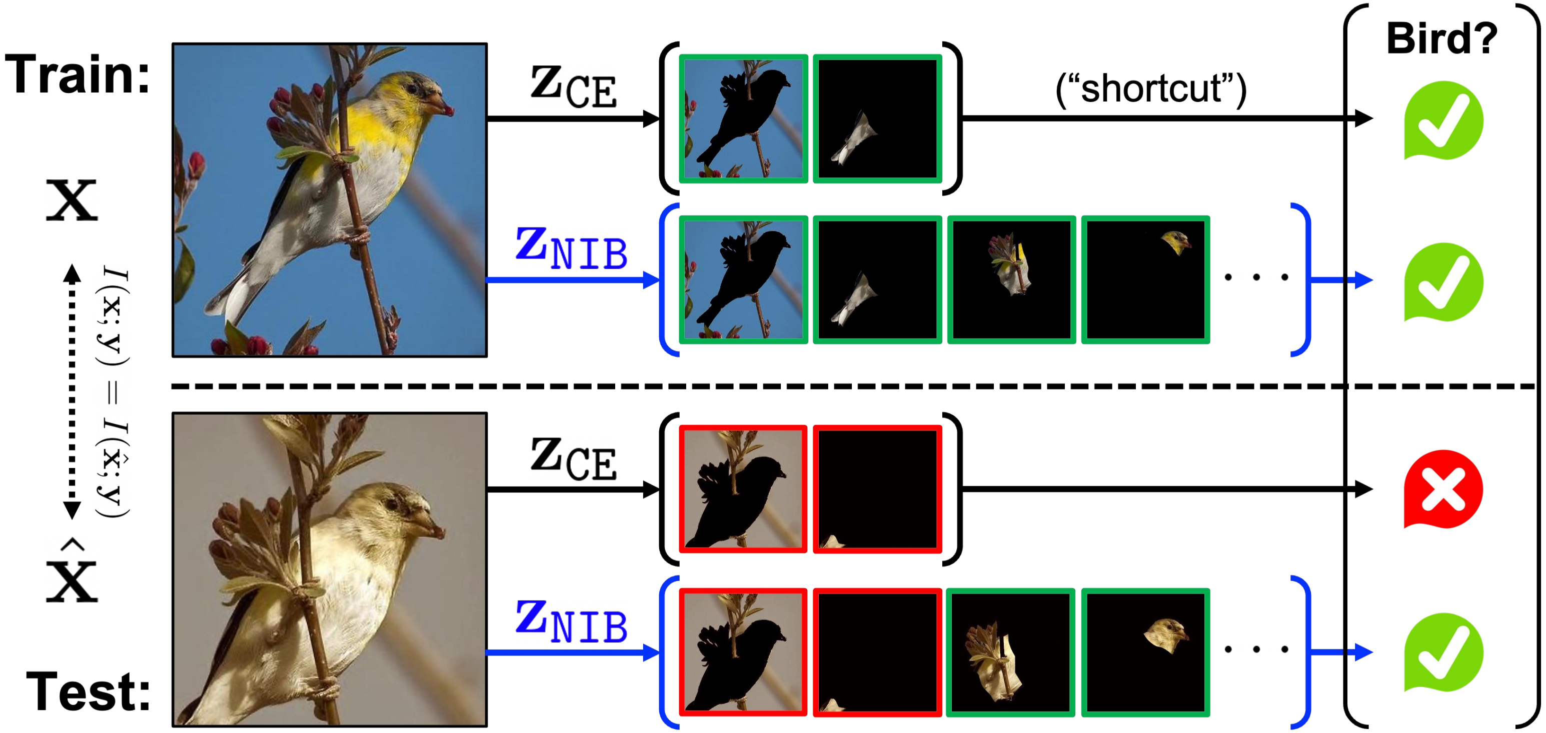
[1] KAIST [2] LG AI Research

CVPR 2023
WED-PM-367

Nuisance-extended Information Bottleneck (NIB)

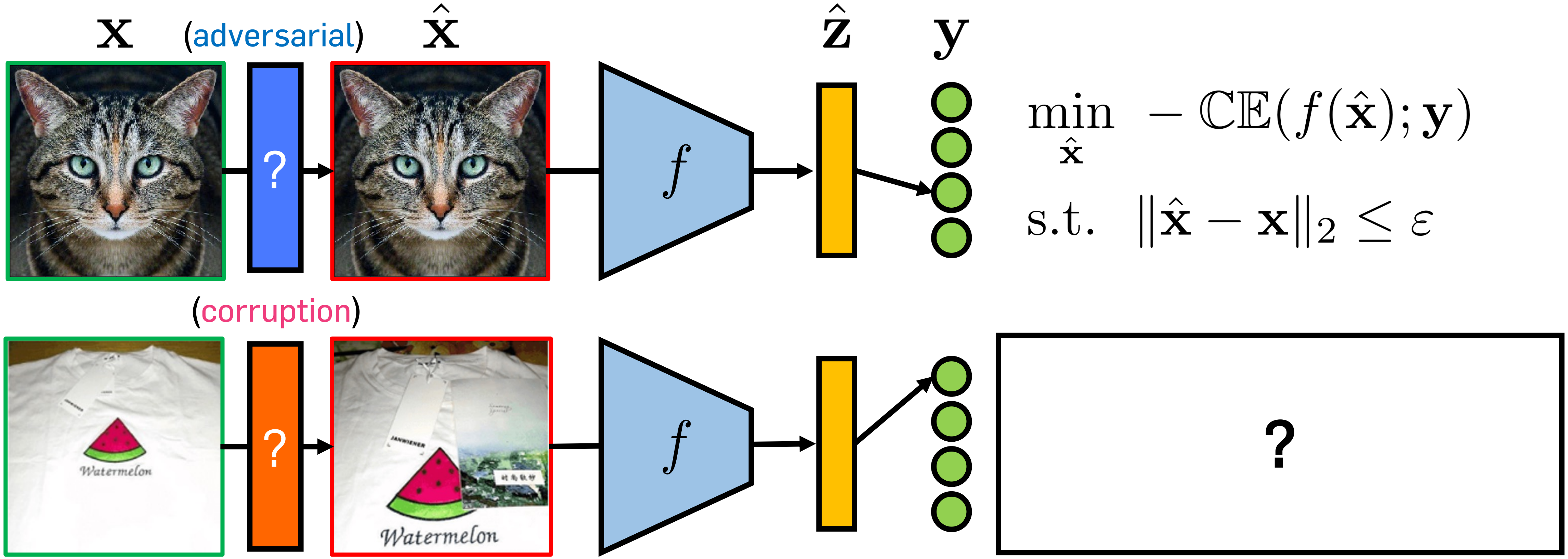
Robustness-aware training without relying on specific priors, e.g., augmentation

- Cross-entropy (CE) is prone to extract only a few "shortcuts"
- NIB instead aims to extract every $\mathbf{y}$-signal in $\mathbf{x}$ → can be more reliable under distribution shifts



Motivation: From adversarial to natural corruption

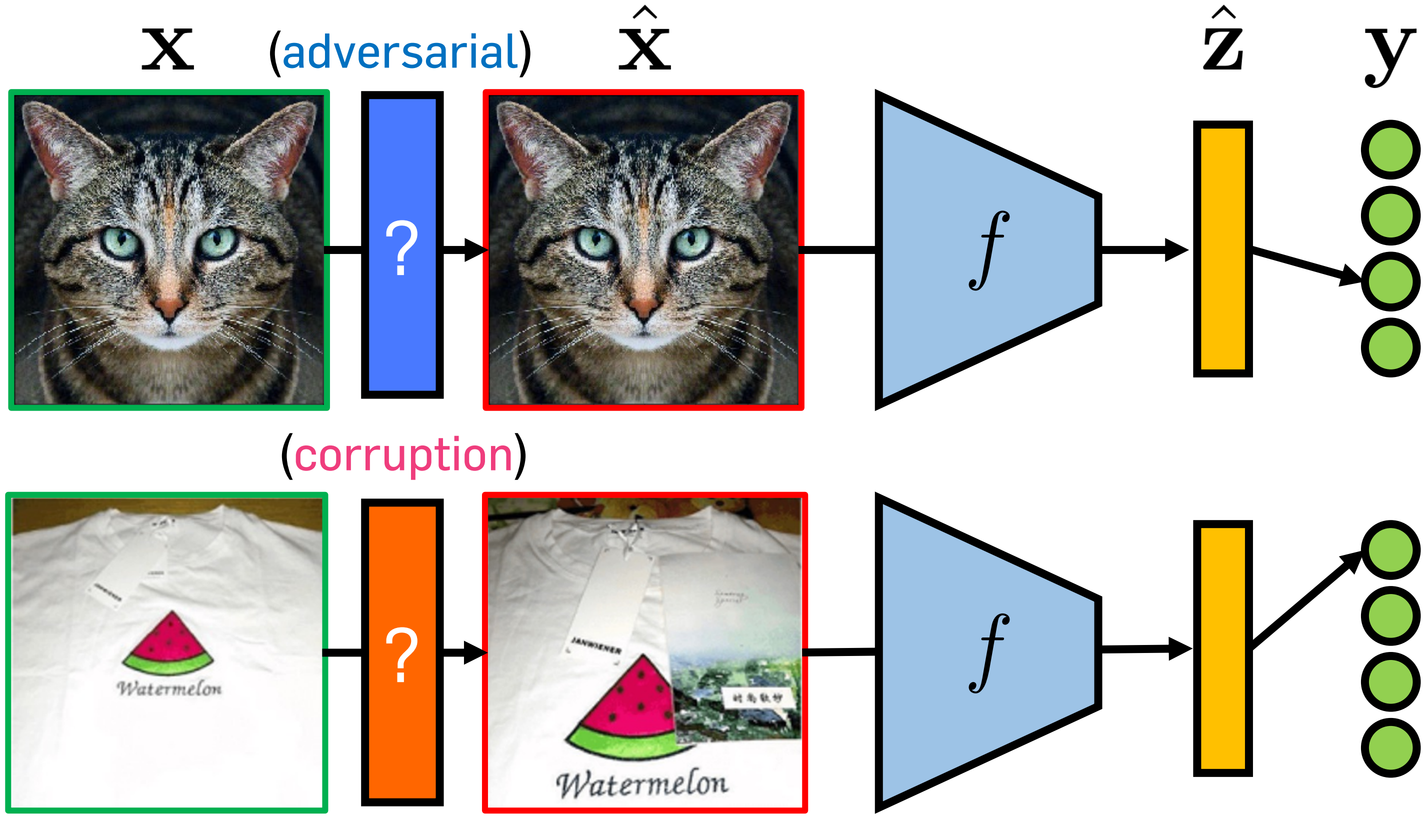
🤔 "How to better abstract the threat model of natural corruptions?"



Motivation: From adversarial to natural corruption

“How to better abstract the threat model of natural corruptions?”

Approach: Apply the “adversarial” framework, but in terms of mutual information



$$\min_{\hat{\mathbf{x}}} \; -\mathbb{CE}(f(\hat{\mathbf{x}}); \mathbf{y}) \quad \text{s.t.} \; \|\hat{\mathbf{x}} - \mathbf{x}\|_2 \leq \varepsilon$$

$$\boxed{\min_{\hat{\mathbf{x}}} \; I(f(\hat{\mathbf{x}}); \mathbf{y}) \quad \text{s.t.} \; I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y})}$$

Nuisance-extended Information Bottleneck (NIB)

🤔 **"How to better abstract the threat model of natural corruptions?"**

💡 **Approach:** Apply the "adversarial" framework, but in terms of mutual information

$$\min_{\hat{\mathbf{x}}} \ I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \ \text{ subject to } \ I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y}) \qquad (*)$$

Goal: Optimizing f via adversarial training with respect to $(*)$?

$$\max_f R_{\texttt{AT}}(f) := \max_f \left(\min_{\hat{\mathbf{x}}} \ I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \right),$$
$$\text{subject to } \ I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y})$$

🤔 **"Is it possible to solve $(*)$ in practice? If not, how to approximate it?"**

💡 **No, it is hard:** we instead introduce a nuisance representation $\mathbf{z}_n$

Nuisance-extended Information Bottleneck (NIB)

🤔 "How to better abstract the threat model of natural corruptions?"

💡 **Approach:** Apply the "adversarial" framework, but in terms of mutual information

$$\min_{\hat{\mathbf{x}}} \ I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \ \text{ subject to } \ I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y}) \qquad (*)$$

💡 We instead introduce a nuisance representation $\mathbf{z}_n$ to model the "remainder"

$$\max_f R_{\texttt{NIB}}(f) := I(\mathbf{z}; \mathbf{y}) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n) - \beta I(\mathbf{x}; \mathbf{z})$$

$$= R^{\beta}_{\texttt{IB}}(f) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n)$$

Nuisance-extended Information Bottleneck (NIB)

🤔 "How to better abstract the threat model of natural corruptions?"

💡 **Approach:** Apply the "adversarial" framework, but in terms of mutual information

$$\min_{\hat{\mathbf{x}}} \; I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \quad \text{subject to} \quad I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y}) \qquad (*)$$

💡 We instead introduce a nuisance representation $\mathbf{z}_n$ to model the "remainder"

$$\max_f R_{\mathtt{NIB}}(f) := I(\mathbf{z}; \mathbf{y}) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n) - \beta I(\mathbf{x}; \mathbf{z})$$

$$= R^{\beta}_{\mathtt{IB}}(f) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n)$$

1. Make sure that $(\mathbf{z}, \mathbf{z}_n)$ can recover the full information of $\mathbf{x}$

Nuisance-extended Information Bottleneck (NIB)

🤔 "How to better abstract the threat model of natural corruptions?"

💡 **Approach:** Apply the "adversarial" framework, but in terms of mutual information

$$\min_{\hat{\mathbf{x}}} \; I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \quad \text{subject to} \quad I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y}) \qquad (*)$$

💡 We instead introduce a nuisance representation $\mathbf{z}_n$ to model the "remainder"

$$\max_f R_{\mathtt{NIB}}(f) := I(\mathbf{z}; \mathbf{y}) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n) - \beta I(\mathbf{x}; \mathbf{z})$$

$$= R^{\beta}_{\mathtt{IB}}(f) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n)$$

1. Make sure that $(\mathbf{z}, \mathbf{z}_n)$ can recover the full information of $\mathbf{x}$
2. Force $\mathbf{z}_n$ not to encode any information related to $\mathbf{y}$ → Let $\mathbf{z}$ do it instead

Nuisance-extended Information Bottleneck (NIB)

🤔 "How to better abstract the threat model of natural corruptions?"

💡 **Approach:** Apply the "adversarial" framework, but in terms of mutual information

$$\min_{\hat{\mathbf{x}}} \; I(\hat{\mathbf{z}} := f(\hat{\mathbf{x}}); \mathbf{y}) \;\; \text{subject to} \;\; I(\mathbf{x}; \mathbf{y}) = I(\hat{\mathbf{x}}; \mathbf{y}) \qquad (*)$$

💡 We instead introduce a nuisance representation $\mathbf{z}_n$ to model the "remainder"

$$\max_f R_{\mathtt{NIB}}(f) := I(\mathbf{z}; \mathbf{y}) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n) - \beta I(\mathbf{x}; \mathbf{z})$$

Why (N-)IB?
: The objective extends the
Information Bottleneck (IB)

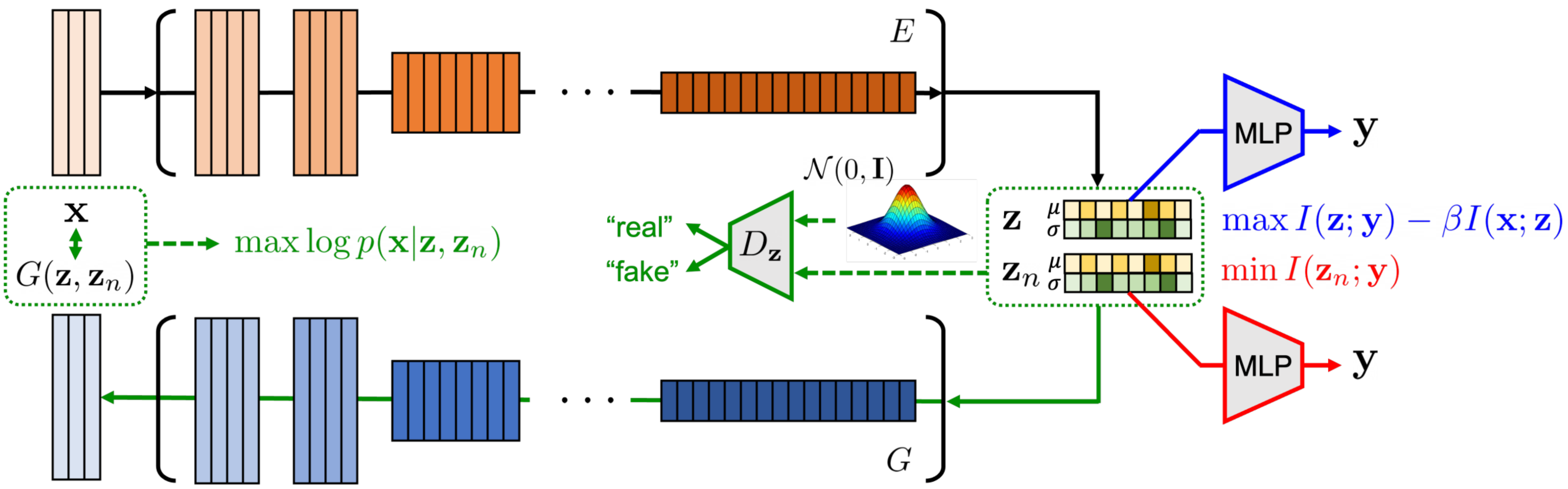
$$= R^{\beta}_{\mathtt{IB}}(f) - I(\mathbf{z}_n; \mathbf{y}) + \alpha I(\mathbf{x}; \mathbf{z}, \mathbf{z}_n)$$

1. Make sure that $(\mathbf{z}, \mathbf{z}_n)$ can recover the full information of $\mathbf{x}$
2. Force $\mathbf{z}_n$ not to encode any information related to $\mathbf{y}$ → Let $\mathbf{z}$ do it instead
3. Still, we do not want $\mathbf{z}_n$ to be trivial → Avoid it by rather compressing out $\mathbf{z}$

AE-NIB: A Practical Autoencoder-based Design

We implement NIB based on an "encoder + decoder" architecture:

$$\max_{f} R_{\mathtt{NIB}}(f) \;=\; \underbrace{R^{\beta}_{\mathtt{IB}}(f)}_{\text{Variational IB (VIB)}} - \underbrace{I(\mathbf{z}_n;\mathbf{y})}_{\text{Nuisance loss}} + \underbrace{\alpha I(\mathbf{x};\mathbf{z},\mathbf{z}_n)}_{\text{Reconstruction loss}}$$



Experiments: Summary

AE-NIB improves security metrics with no additional priors (e.g., augmentation):

1. Natural robustness
    - **Corruption robustness**: CIFAR-10/100-C, and ImageNet-C
    - **OOD Generalization**: CIFAR-10.1/10.2, CINIC-10, ImageNet-R, and ImageNet-Sketch
    - **Background bias**: Backgrounds Challenge [Xiao et al., 2020]
2. Novelty detection
    - Standard / Full-spectrum OOD [Yang et al., 2022] benchmarks
3. Certified adversarial robustness [Cohen et al., 2019]
    - Certified test accuracy @ radius $\boldsymbol{r}$

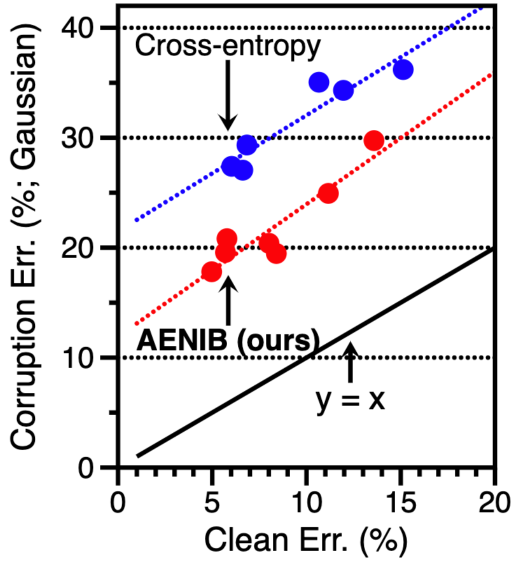
… these and more results can be found in the paper!

[Xiao et al., 2020] Noise or Signal: The Role of Image Backgrounds in Object Recognition, ICLR 2020.
[Hendrycks and Gimpel, 2017] A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. ICLR 2017.
[Yang et al., 2022] Full-Spectrum Out-of-Distribution Detection. 2022.
[Cohen et al., 2019] Certified adversarial robustness via randomized smoothing. ICML 2019.

Experiments: Natural robustness

AE-NIB improves diverse types of robustness without stronger augmentation

- It solely outperforms AugMix and PixMix: Even they use more data or augmentation
- The effectiveness of the training could generalize to ImageNet-scale
- It exhibits better trend in clean- vs. corruption accuracy compared to Cross-entropy



Method	C10	C10-C	C10.1	C10.2	CINIC
Cross-entropy	6.08	16.0	13.4	18.3	23.7
VIB [1]	5.98	15.2	13.6	16.8	23.6
NLIB [65]	6.81	17.0	14.6	17.5	24.3
sq-NLIB [106]	6.02	15.5	13.0	17.1	23.7
DisenIB [92]	5.76	15.2	13.2	17.2	23.7
AugMix [39]	6.52	15.1	14.2	17.2	24.2
PixMix [40]	5.43	10.3	13.1	16.6	23.2
AENIB (ours)	4.97	12.3	11.6	15.5	22.2
+ AugMix [39]	5.35	12.0	12.5	15.8	22.6
+ PixMix [40]	**4.67**	**8.08**	**10.4**	**14.8**	**22.1**

- (Top) Results on CIFAR-10/-C and CIFAR-variants
- (Upper right) Results on ImageNet and -C/R/Sketch
- (Lower right) Results on Backgrounds Challenge

	ViT-S/16		ViT-B/16	
Dataset	Baseline	**AENIB (ours)**	Baseline	**AENIB (ours)**
IN-1K	**25.1**	**25.1**	21.8	**21.9**
IN-C (mCE)	65.9	**65.2** (−0.7)	58.6	**57.5** (−1.1)
IN-R	70.3	**67.1** (−3.2)	66.3	**64.4** (−1.9)
IN-Sketch	80.3	**77.7** (−2.6)	76.5	**74.4** (−2.1)

BG-Challenge	ViT-S/16		ViT-B/16	
Dataset	Baseline	**AENIB (ours)**	Baseline	**AENIB (ours)**
Original (IN-9; ↑)	95.3	**95.5**	96.0	**96.1**
Only-BG-T (↓)	20.3	**17.8** (−2.5)	24.2	**21.1** (−3.1)
Mixed-Same (↑)	86.3	**88.3** (+2.0)	87.4	**88.9** (+1.5)
Mixed-Rand (↑)	77.8	**80.5** (+2.7)	80.1	**81.8** (+0.7)
BG-gap (↓)	8.5	**7.8** (−0.7)	7.3	**7.1** (−0.2)

[Hendrycks et al., 2020] AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. ICLR 2020.
[Hendrycks et al., 2022] PixMix: Dream-like pictures comprehensively improve safety measures, CVPR 2022.

Experiments: Novelty detection

AE-NIB defines a strong OOD score based on the likelihood of $(\mathbf{z}, \mathbf{z}_n)$:

$$\text{score}(\mathbf{z}, \mathbf{z}_n) \coloneqq \log \mathrm{Dir}_\alpha(\mathbf{y}) + \log \mathcal{N}(\mathbf{z}_n) \qquad \left(\begin{array}{l} \log \mathcal{N}(\mathbf{z}_n; 0, I) = -\frac{1}{2}\|\mathbf{z}_n\|^2 \\ \log \mathrm{Dir}_\alpha(\mathbf{y}) = (\alpha - 1)\sum_i \log y_i \end{array} \right)$$

Results on OBJECTS benchmark [Yang et al., 2022]

- OBJECTS = CIFAR-10 (train) + {CIFAR-10-C, ImageNet-10}

FS-OOD: OBJECTS		AUROC (%; ↑) / AUPR (%; ↑) / FPR@TPR95 (%; ↓)			
Method	Score	MNIST	FashionMNIST	Texture	CIFAR-100-C
Cross-entropy	$\max_y p(y\|x)$ [36]	66.98 / 52.66 / 93.54	73.78 / 90.15 / 88.08	74.18 / 93.34 / 85.64	74.12 / 89.74 / 87.26
	ODIN [73]	70.31 / 49.58 / 82.04	80.98 / 91.53 / **68.73**	70.14 / 89.97 / 72.91	67.51 / 83.97 / 84.26
	Energy-based [75]	54.55 / 34.14 / 92.23	76.50 / 89.80 / 72.40	68.63 / 89.51 / 75.57	68.37 / 85.54 / 83.64
	Mahalanobis [71]	77.04 / 65.31 / 84.59	80.33 / 92.28 / 77.17	72.02 / 88.46 / 72.98	68.13 / 82.97 / 85.53
	SEM [120]	75.69 / 76.61 / 99.70	79.40 / 93.14 / 93.72	79.69 / 95.48 / 82.15	78.89 / 92.07 / 83.92
	log Dir$_{0.05}(y)$	76.75 / 66.26 / 83.51	82.88 / 93.97 / 77.19	70.69 / 92.68 / 91.35	78.80 / 92.21 / 82.50
VIB [1]	$\max_y p(y\|x)$ [36]	80.23 / 73.50 / 80.69	76.35 / 91.22 / 84.75	74.67 / 94.09 / 87.22	76.12 / 91.03 / 84.99
	log Dir$_{0.05}(y)$	86.13 / 79.45 / 64.92	81.11 / 93.12 / 77.82	73.84 / 93.50 / 88.00	78.54 / 91.85 / 81.47
AENIB (ours)	$\max_y p(y\|x)$ [36]	79.67 / 71.50 / 80.22	77.33 / 91.63 / 84.31	74.95 / 93.97 / 86.01	74.31 / 89.89 / 86.26
	log Dir$_{0.05}(y)$	90.53 / 85.68 / 52.08	84.56 / 94.61 / 74.24	75.04 / 93.83 / 86.01	79.39 / 92.33 / 81.51
	+ log $\mathcal{N}(z_n; 0, I)$	**92.43 / 89.38 / 48.10**	**84.85 / 94.84** / 74.67	**88.91 / 97.49 / 48.44**	**82.66 / 93.62 / 74.14**

Summary

TL;DR: Nuisance modeling can be a tangible approach for threat-free robust training

Robustness-aware training without relying on domain prior, e.g., data augmentation

- NIB aims to extract every $\mathbf{y}$-signal in $\mathbf{x}$ → can be more reliable under distribution shifts
- AE-NIB implement NIB with an autoencoder + variational IB architecture

More details can be found:

- Paper: https://arxiv.org/abs/2303.14096
- Code: https://github.com/jh-jeong/nuisance_ib

Please drop by our poster session for more information!

- WED-PM-367 / West Building Exhibit Halls ABC