JUNE 18-22, 2023
CVPR VANCOUVER, CANADA

Backdoor Defense via Deconfounded Representation Learning

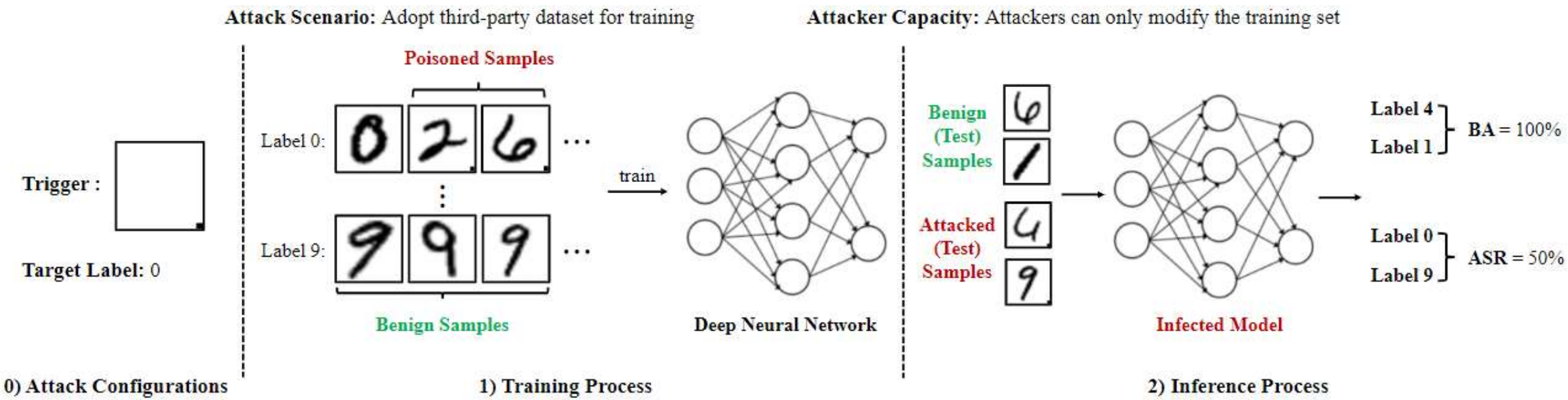
Zaixi Zhang[1], Qi Liu[1*], Zhicai Wang[1], Zepu Lu[1], Qingyong Hu[2]

1: University of Science and Technology of China

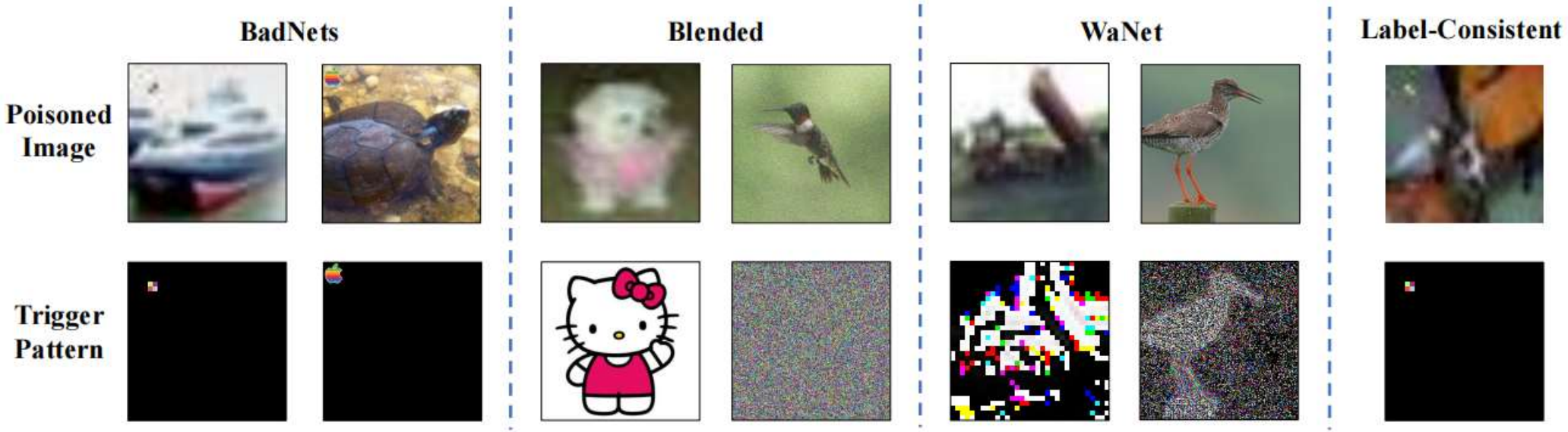
2: Hong Kong University of Science and Technology

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

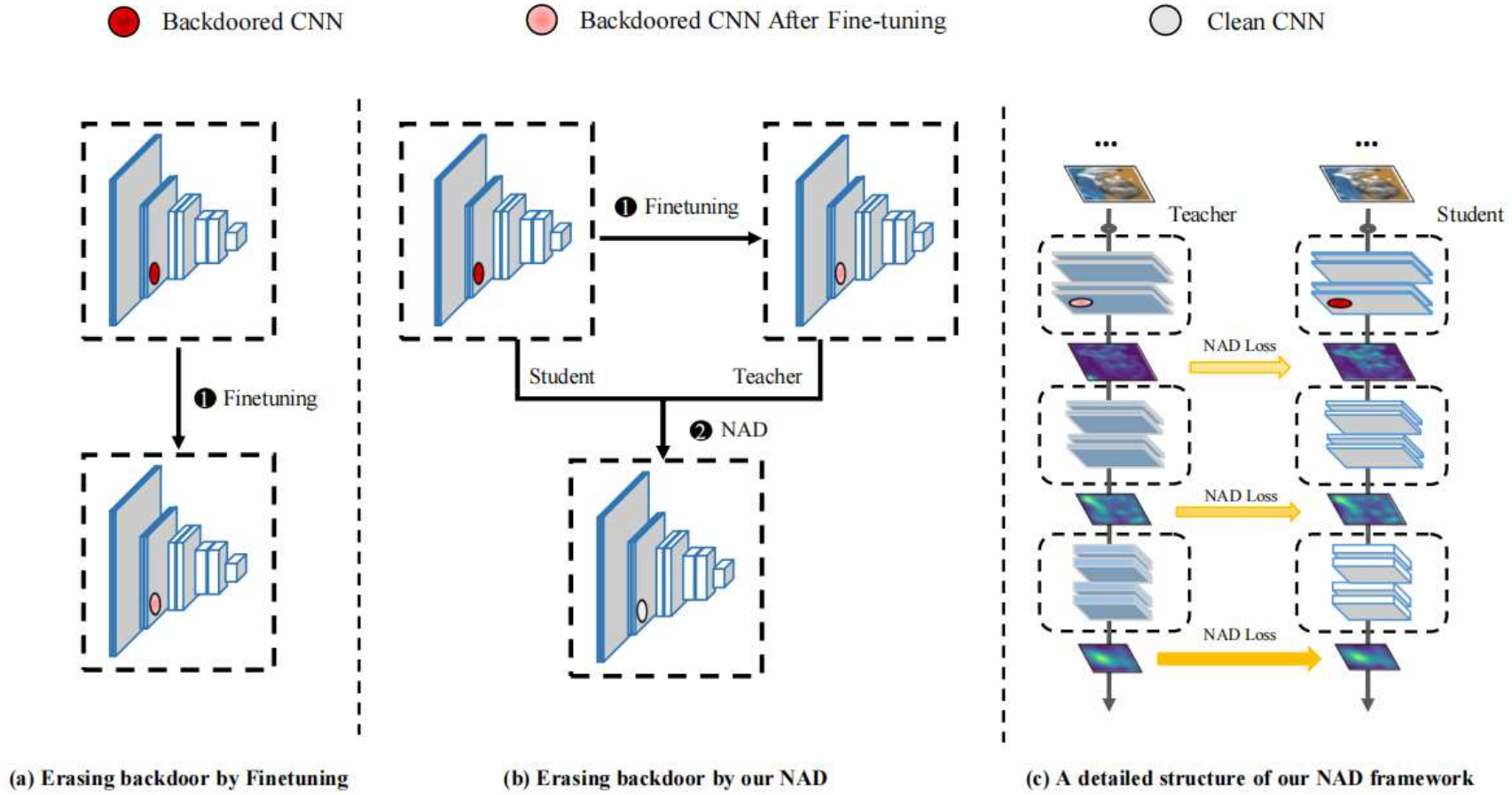
Background: Backdoor Attacks against DNNs



Background: Backdoor Attacks against DNNs



Background: Defense Methods



Yige Li, et al. Neural attention distillation: Erasing backdoor triggers from deep neural networks. ICLR 2021

Background: Defense Methods

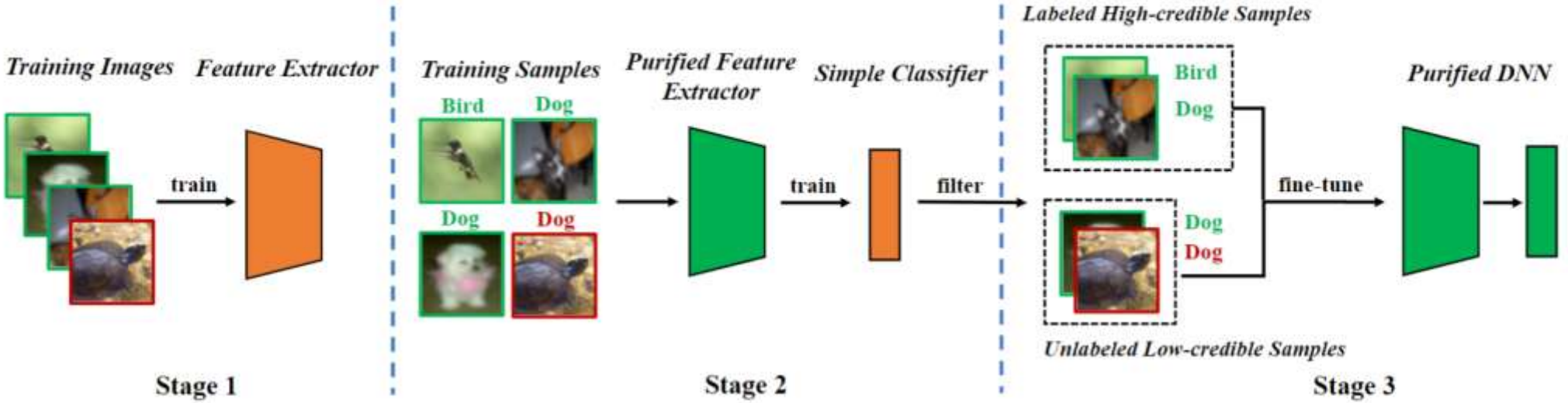


Figure 2: The main pipeline of our defense. In the first stage, we train the whole DNN model via self-supervised learning based on label-removed training samples. In the second stage, we freeze the learned feature extractor and adopt all training samples to train the remaining fully connected layers via supervised learning. After that, we filter high-credible samples based on the training loss. In the third stage, we adopt high-credible samples as labeled samples and remove the labels of all low-credible samples to fine-tune the whole model via semi-supervised learning.

Kunzhe Huang, et al. Backdoor defense via decoupling the training process. ICLR 2022

Limitations of Previous works & Motivations

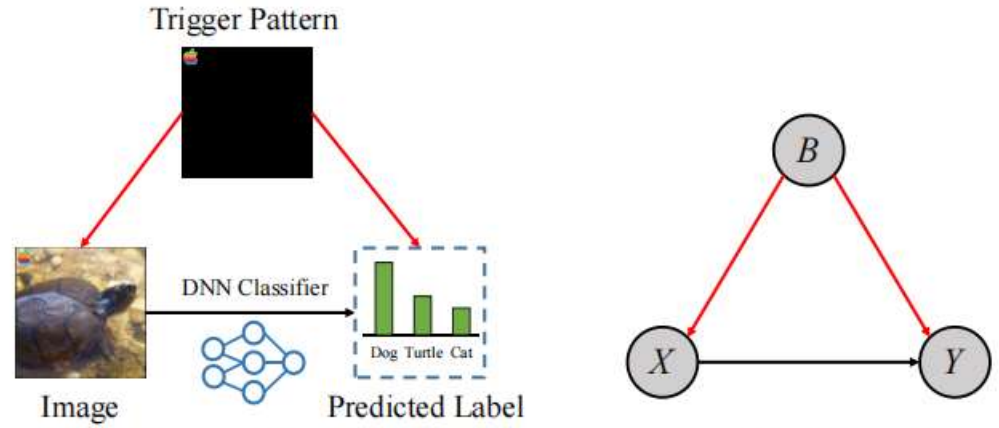
- However, most existing defense methods require clean data are inefficient. It is still unknown whether a **backdoor-free clean model** can be directly obtained from **poisoned** datasets.

- In contrast to DNNs, **human cognitive systems** are known to be immune to input perturbations such as **stealthy trigger patterns** induced by backdoor attacks. This is because humans are more sensitive to causal relations than the statistical associations of nuisance factors

- From the **causal perspective**, backdoor attack acts as the **confounder**, which brings spurious associations between the input images and target labels, making the model predictions less reliable.

CBD

Causality-inspired **B**ackdoor **D**efense

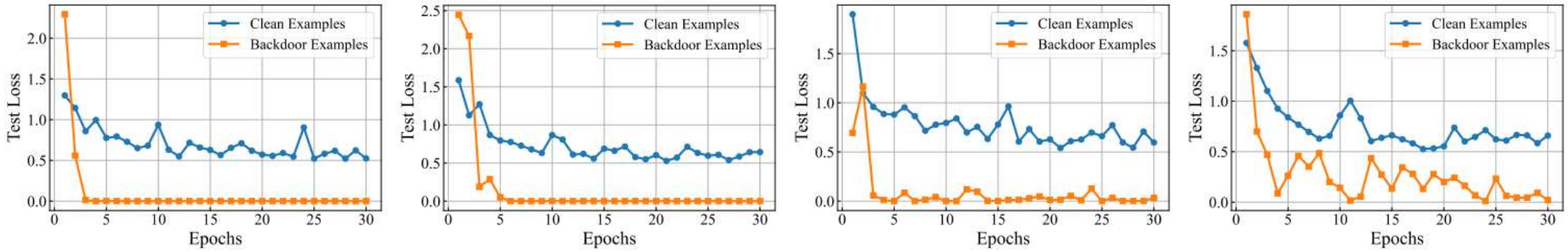


(a) Illustration of Backdoor Attack (b) Causal Graph

Figure 1. (a) A real example of the backdoor attack. The backdoored DNN classifies the "turtle" image with a trigger pattern as the target label "dog". (b) The causal graph represents the causalities among variables: X as the input image, Y as the label, and B as the backdoor attack. Besides the causal effect of X on Y ($X \rightarrow Y$), the backdoor attack can attach trigger patterns to images ($B \rightarrow X$), and change the labels to the targeted label ($B \rightarrow Y$). Therefore, as a *confounder*, the backdoor attack B opens a spurious path between X and Y ($X \leftarrow B \rightarrow Y$).

CBD

Causality-inspired **B**ackdoor **D**efense



(a) BadNets (b) Trojan (c) Blend (d) WaNet

Backdoors are easier to learn.

CBD

Causality-inspired **B**ackdoor **D**efense

Two Neural networks are trained, and only f_C is used for the prediction of the test set.

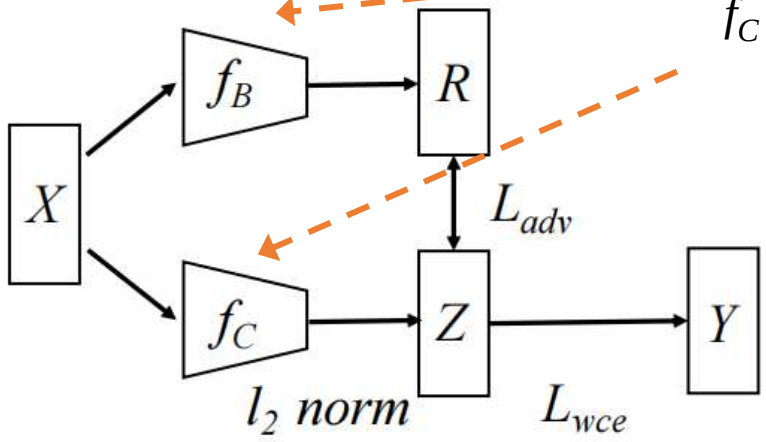


Figure 2. The model framework of CBD that includes an adversaral loss $\mathcal{L}_{adv}$ for mutual information minimization, a l_2-norm regularization on z, and a weighted cross entropy loss $\mathcal{L}_{wce}$ to augment causal effects.

$$\mathcal{L}_C = \min \underbrace{\beta I(Z;X)}_{①} - \underbrace{I(Z;Y)}_{②} + \underbrace{I(Z;R)}_{③},$$

Information bottleneck

De-confounder penalty term

CBD

$$\mathcal{L}_C = \min \underbrace{\beta I(Z;X)}_{①} - \underbrace{I(Z;Y)}_{②} + \underbrace{I(Z;R)}_{③},$$

- Term ①, L2 regularization
- Term ②, weighted cross-entropy loss:

$$w(x) = \frac{CE(f_B(x), y)}{CE(f_B(x), y) + CE(f_C(x), y)}.$$

- Term ③, adversarial process:

$$\mathcal{L}_{adv} = \min_{\theta_C} \max_{\phi} \mathbb{E}_{p(z,r)}[D_\phi(z,r)] - \mathbb{E}_{p(z)p(r)}[D_\phi(z,r)],$$

CBD

Algorithm 1 Causality-inspried Backdoor Defense (CBD)

Input: β, number of training iterations T_1, T_2
Output: Clean model f_C;

Initialize f_C, f_B, and D_ϕ
for $t = 1, \cdots, T_1$ **do**
    Train f_B on the poisoned dataset with SGD
end for
for $t = 1, \cdots, T_2$ **do**
    Train discriminator D_ϕ
    Calculate sample weight $w(x)$
    Train f_C with loss function in Equation 8
end for

Experiments

Table 1. The attack success rate (ASR %) and the clean accuracy (CA %) of 5 backdoor defense methods against 6 representative backdoor attacks. *None* means the training data is completely clean. The best results are bolded.

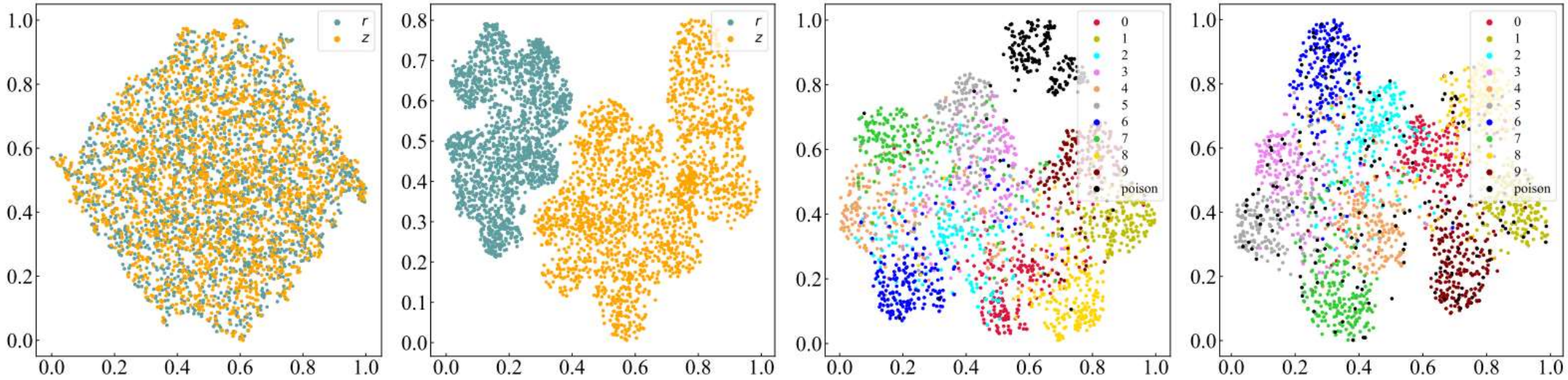
Dataset	Types	No Defense		FP		MCR		NAD		ABL		DBD		**CBD (Ours)**	
		ASR	CA	ASR	CA	ASR	CA	ASR	CA	ASR	CA	ASR	CA	ASR	CA
CIFAR-10	*None*	0	89.14	0	85.17	0	87.55	0	88.21	0	88.49	0	88.63	0	**88.95**
	BadNets	100	85.37	99.96	82.41	4.52	79.66	3.07	82.25	3.13	86.30	1.76	86.94	**1.06**	**87.46**
	Trojan	100	84.54	68.95	81.03	19.47	77.12	19.96	80.05	3.88	87.29	3.79	87.29	**1.24**	**87.52**
	Blend	100	84.56	87.14	81.57	36.15	78.24	10.65	83.71	14.60	85.02	5.12	86.83	**1.96**	**87.48**
	SIG	99.32	84.14	73.87	81.04	2.34	77.93	1.79	83.54	0.36	**88.10**	0.44	87.52	**0.25**	87.29
	Dynamic	100	85.48	89.22	80.63	25.26	75.03	25.60	74.94	17.26	85.29	10.21	85.42	**0.86**	**85.67**
	WaNet	98.55	86.77	73.12	81.58	28.59	77.12	24.15	79.50	22.24	75.74	5.86	84.60	**4.24**	**86.55**
	Average	99.65	85.14	82.03	81.38	19.39	77.52	14.20	80.67	10.25	84.62	4.53	86.43	**1.60**	**87.00**
GTSRB	*None*	0	97.74	0	90.18	0	95.27	0	95.29	0	96.47	0	96.45	0	**96.54**
	BadNets	100	96.58	99.48	88.57	1.27	93.30	0.31	89.90	**0.05**	96.01	0.24	96.05	0.16	**96.21**
	Trojan	99.95	96.49	97.40	88.51	4.62	92.99	0.56	90.32	0.47	94.91	0.56	94.69	**0.12**	**95.29**
	Blend	100	95.57	98.78	87.50	6.85	93.11	13.06	89.20	22.97	93.25	6.36	93.72	**0.90**	**94.16**
	SIG	98.24	96.55	85.04	89.97	26.80	91.14	5.35	89.27	5.09	**96.28**	**4.70**	94.58	5.41	94.37
	Dynamic	100	96.87	98.33	88.09	59.54	90.51	62.35	84.30	6.32	95.76	5.16	95.86	**0.96**	**96.02**
	WaNet	99.92	95.94	97.93	90.13	55.25	91.24	34.16	83.09	5.56	93.83	3.47	94.71	**3.13**	**95.64**
	Average	99.69	96.33	96.16	88.80	25.72	92.05	19.30	87.68	7.96	95.01	3.42	94.94	**1.82**	**95.17**
ImageNet Subset	*None*	0	88.95	0	83.05	0	85.61	0	87.34	0	88.12	0	88.30	0	**88.57**
	BadNets	100	85.24	98.03	82.76	25.14	77.90	7.38	82.11	1.02	87.47	1.27	87.61	**0.66**	**88.12**
	Trojan	100	85.65	97.29	81.46	6.65	77.06	13.80	81.49	1.68	88.21	1.48	88.20	**0.72**	**88.24**
	Blend	99.89	86.10	99.10	81.37	18.37	76.21	25.05	82.54	20.80	85.23	4.73	86.25	**1.82**	**87.95**
	SIG	98.53	86.06	77.39	82.55	24.62	78.97	5.30	83.24	**0.22**	86.65	1.95	87.09	0.45	**87.27**
	Average	99.61	85.74	92.95	82.04	18.70	77.54	12.88	82.35	5.93	86.89	2.36	87.29	**0.91**	**87.90**

Experiments

Table 2. Robustness test with the poisoning rate from 1% to 50% for 4 attacks including BadNets, Trojan, Blend, and WaNet on CIFAR10 dataset. We show ASR (%) and CA (%).

Poisoning Rate	Defense	BadNets		Trojan		Blend		WaNet	
		ASR	CA	ASR	CA	ASR	CA	ASR	CA
1%	*None*	100	85.67	100	85.15	100	85.22	97.56	86.55
	CBD	0.62	88.83	1.13	88.56	0.67	87.52	1.06	86.59
5%	*None*	100	84.68	100	84.82	100	85.06	99.83	86.27
	CBD	0.93	87.50	1.10	88.45	0.73	87.47	1.07	86.56
20%	*None*	100	83.42	100	79.32	100	82.08	100	74.41
	CBD	1.16	84.35	1.57	81.71	5.17	86.53	5.72	74.25
50%	*None*	100	79.45	100	72.83	100	69.67	100	67.25
	CBD	1.47	78.88	2.31	75.34	8.14	85.56	8.75	70.43

Experiments



(a) Initialization (b) Training Completed (c) r (d) z

Figure 3. Visualization of the hidden space with t-SNE

Experiments

Table 3. The average training time (seconds) on CIFAR10 and the ImageNet subset with no defense and CBD. The percentages in parentheses indicate the relative increase of training time.

Dataset	CIFAR-10		ImageNet subset	
	No Defense	CBD	No Defense	CBD
BadNets	1152	1317(14.3%)	2640	2987(13.1%)
Trojan	1204	1356(12.6%)	2621	2933(11.9%)
Blend	1159	1311(13.1%)	2623	3076(17.3%)
WaNet	1164	1293(11.1%)	2647	3074(16.1%)

Resistance to Potential Adaptive Attacks

- The intuition of our adaptive attack strategy is to **slow** the injection process of backdoor attacks (i.e., increasing the corresponding training losses) by adding **optimized noise** into the poisoned examples.

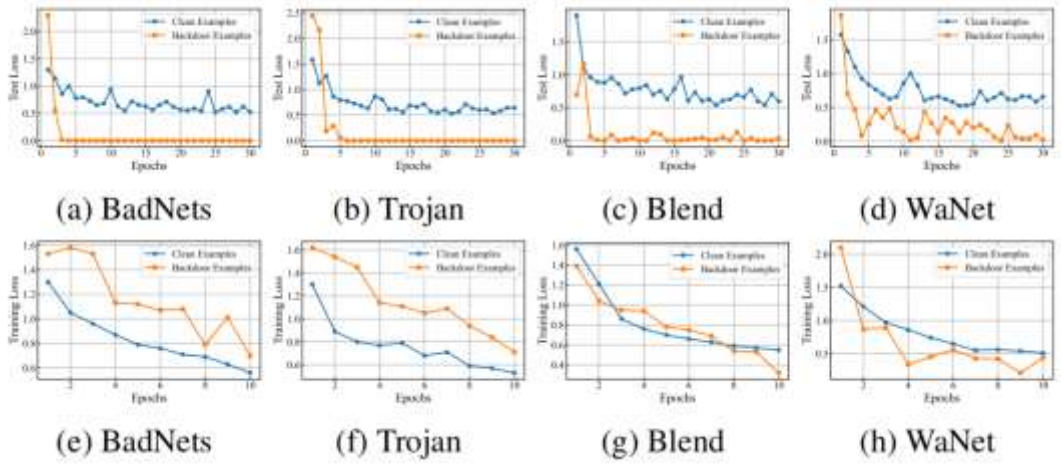
$$\min_{\theta} \left[\sum_{x \in \mathcal{D}} \mathcal{L}(f_\theta(x), y) + \sum_{x \in \mathcal{D}'} \max_{\delta_i} \mathcal{L}(f_\theta(x + \delta_i), y) \right], \qquad \|\delta_i\|_p \leq \epsilon$$

Algorithm 2 Adaptive Attack to CBD

Input: Model f_θ, poisoned dataset $\mathcal{D}'$, clean dataset $\mathcal{D}$, perturbation range ϵ, number of training iterations T, step size α, update steps M.
Output: optimized poisoned dataset $\mathcal{D}'$

Initialize f_θ
for $t = 1, \cdots, T$ **do**
  Draw a mini-batch $\mathcal{B} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$ from $\mathcal{D} \cup \mathcal{D}'$
  $\theta \leftarrow \theta - \eta \nabla_\theta \sum_{(x,y) \in \mathcal{B}} \mathcal{L}(f_\theta(x), y)$
  for (x_i, y_i) in $\mathcal{D}'$ **do**
    for $m = 1, \cdots, M$ **do**
      $x_i \leftarrow \Pi_\epsilon(x_i + \alpha \cdot \nabla_x \mathcal{L}(f_\theta(x_i), y_i))$
    end for
  end for
end for

Resistance to Potential Adaptive Attacks



(a) BadNets (b) Trojan (c) Blend (d) WaNet

(e) BadNets (f) Trojan (g) Blend (h) WaNet

Figure 4. The curve of training losses on clean/backdoor examples in the vanilla training (first line) and in the optimization of adaptive attacks (second line). This experiment is conducted with WideResNet-16-1 for CIFAR-10 under poisoning rate 10%.

Table 5. Attack success rate (ASR %) and clean accuracy (CA %) of Adaptive Attacks.

Defense	BadNets		Trojan		Blend		WaNet	
	ASR	CA	ASR	CA	ASR	CA	ASR	CA
None	99.62	84.55	99.85	84.32	97.63	84.45	97.24	85.47
CBD	4.31	84.19	3.77	84.37	2.57	84.49	5.19	85.33

Conclusion & Future Works

- Inspired by the causal perspective, we proposed **C**ausality-inspired **B**ackdoor **D**efense (CBD) to learn de-confounded representations for reliable classification.

- Extensive experiments against 6 state-of-the-art backdoor attacks show the effectiveness and robustness of CBD. Further analysis shows that CBD is robust against potential adaptive attacks.

- Our work opens up an interesting research direction to leverage causal inference to analyze and mitigate backdoor attacks in machine learning.

- Future works include extending CBD to other domains including graph learning, federated learning, and self-supervised learning.

Thank you!

- The paper: https://arxiv.org/abs/2303.06818
- The code: https://github.com/zaixizhang/CBD
- For any further questions, please email : zaixi@mail.ustc.edu.cn