Incremental Learning of GAN for Detecting Multiple Adversarial Attacks

Zibo Yi, Jie Yu, Shasha Li, Yusong Tan, and Qingbo Wu
College of Computer, National University of Defense Technology,
Changsha, Hunan Province, China
{yizibo14,yj,shashali,yusong.tan,qingbo.wu}@nudt.edu.cn

Abstract

Neural networks are vulnerable to adversarial attack. Carefully crafted small perturbations can cause misclassification of neural network classifiers. As adversarial attack is a serious potential problem in many neural network based applications and new attacks always come up, it's urgent to explore the detection strategies that can adapt new attacks quickly. Moreover, the detector is hard to train with limited samples. To solve these problems, we propose a GAN based incremental learning framework with Jacobian-based data augmentation to detect adversarial samples. To prove the proposed framework works on multiple adversarial attacks, we implement FGSM, LocSearchAdv, PSO-based attack on MNIST and CIFAR-10 dataset. The experiments show that our detection framework performs well on these adversarial attacks

Motivation

– New attacks always come up. A method that can recognize new kind of adversarial samples is urgently needed. We propose a GAN incremental learning framework to detect multiple adversarial samples. Through incremental learning, the framework can detect new adversarial attack patterns.

– Attackers tend to use partial adversarial samples when they conduct a new kind of attack. The detector is hard to train with limited samples. With Jacobian-based data augmentation applied, the problem of learning from limited adversarial samples are solved.

Method

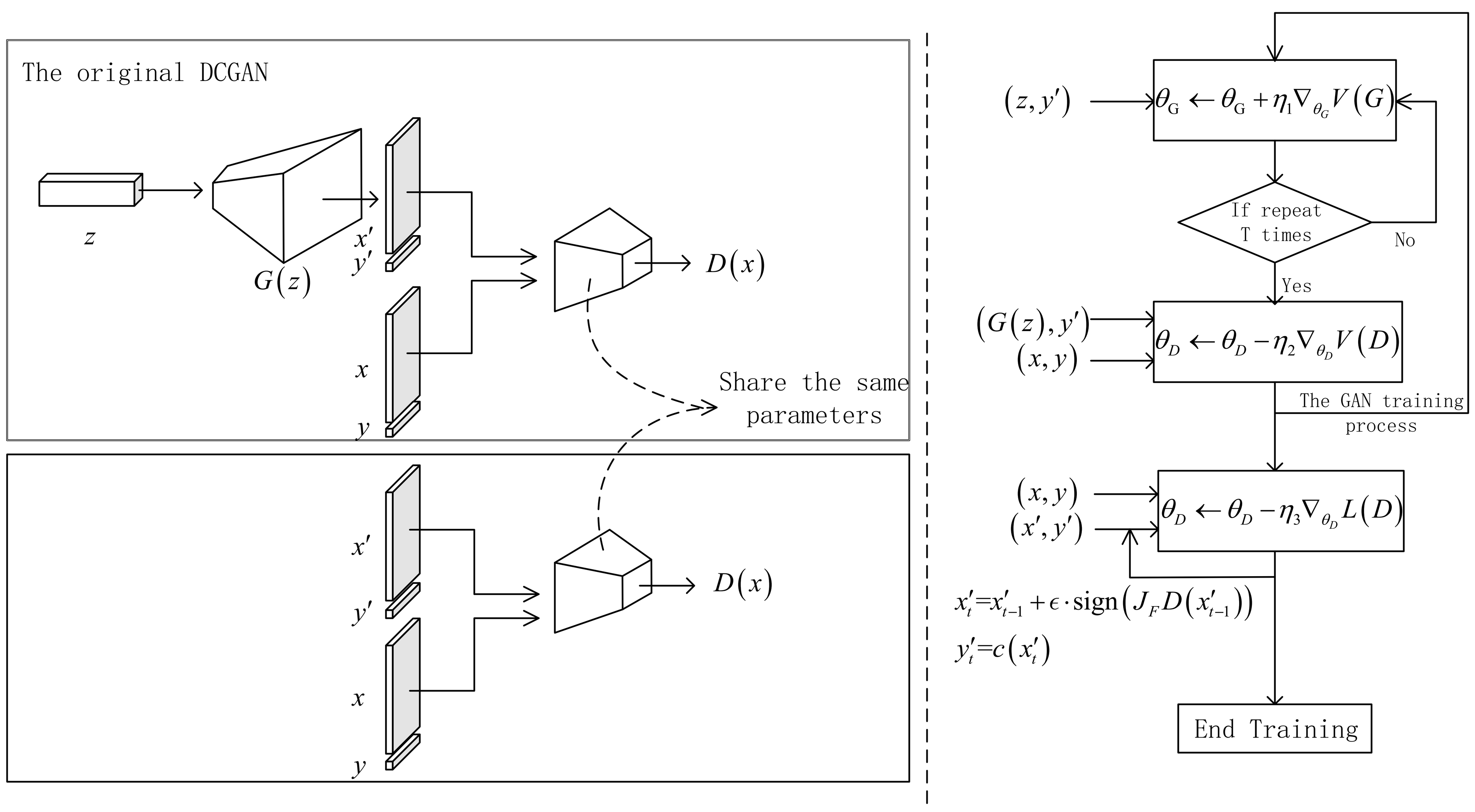

Fig. 1. The incremental GAN learning framework. The left part of the figure depicts the schematic layout of the framework and the right side shows it's flowchart. We implement incremental training of the discriminator by parameters sharing. We trained a preliminary discriminator using the original DCGAN (See the original DCGAN in the layout and the GAN training process in the flowchart). Then the discriminator are incremental trained using (x, y), (x', y'), which are generated with Jacobian-based data augmentation technique.

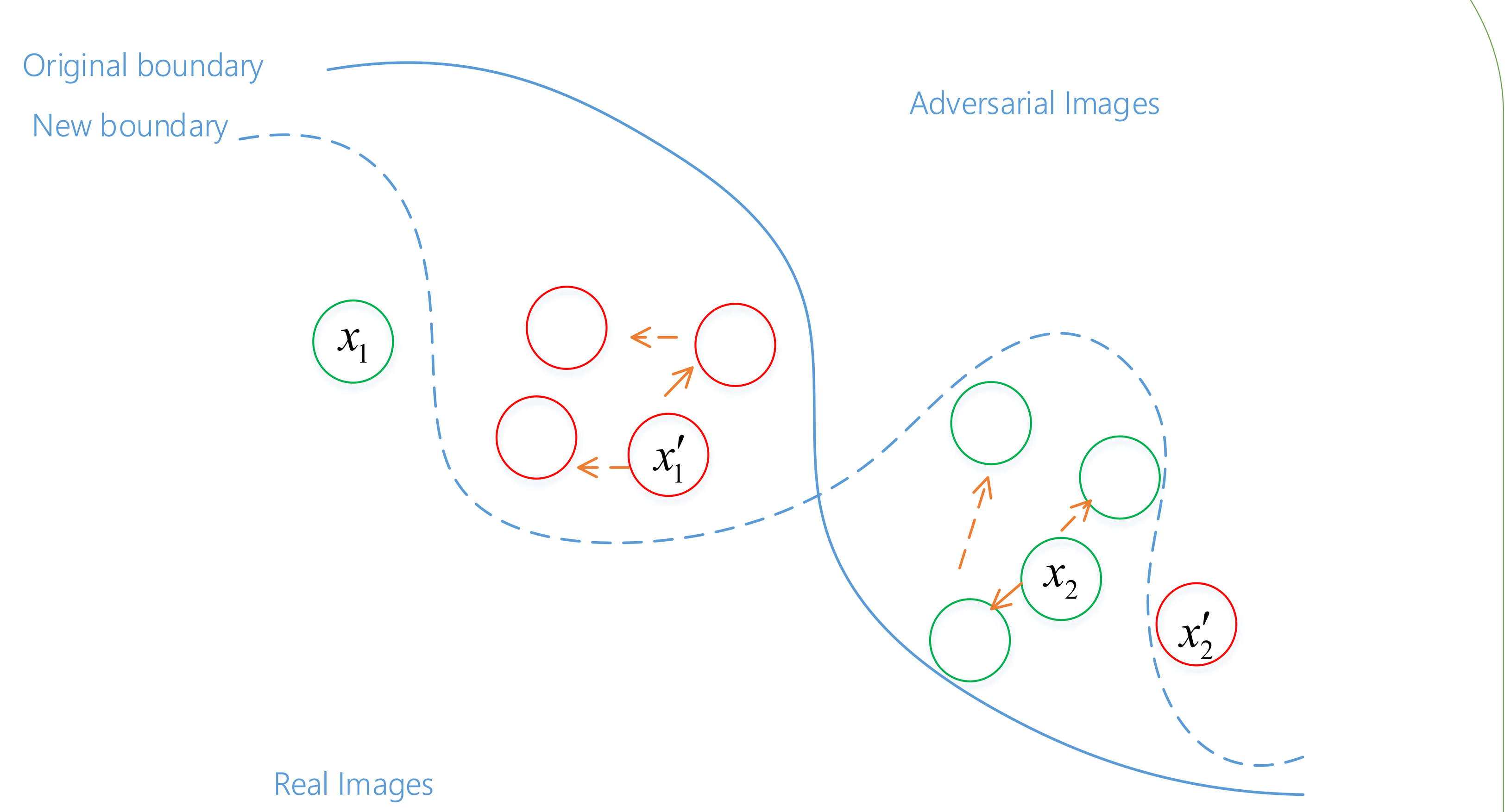

Fig. 2. The illustration of Jacobian-based incremental learning. The discriminator separates the adversarial images and the real images with the classification boundary. x_1, x_2 are adversarial images while x'_1 and x'_2 are real images. Both x'_1 and x_2 are misclassified by the original discriminator. Incremental training adjust the boundary with additional samples. The arrows in the figure represent the data generation process using Jacobian-based data augmentation. After incremental training, the boundary is adjusted to obtain the correct classification result.

Algorithm 1 Jacobian-based incremental learning of GAN

Input:

The adversarial sample dataset, D_A;
The size of initial adversarial sample dataset, $k = |D_A|$;
The dataset (only contains normal samples) of GAN, D_N;
The parameters of discriminator, θ_D;
The normal dataset larger than initial adversarial dataset, n;
The number of incremental training rounds, r;
The step size, ϵ;

Output:

The updated parameters of discriminator, θ_D;

for each t **in** $[1, 2, ..., r]$ **do**
  Random split D_N into $D_N^{(1)}$, $D_N^{(2)}$, ..., $D_N^{(n)}$
  for each j **in** $[1, 2, ..., n]$ **do**
    $\theta_D \leftarrow \theta_D - \eta_3 \nabla_{\theta_D}(-\frac{1}{k}\sum_{i=1}^{k}[\log D(\boldsymbol{x}^{(i)}, y^{(i)}) + \log(1 - D(\boldsymbol{x}'^{(i)}, y'^{(i)}))])$,
    where $(\boldsymbol{x}^{(i)}, y^{(i)}) \in D_N^{(j)}$, $(\boldsymbol{x}'^{(i)}, y'^{(i)}) \in D_A$
  end for
  Replace each $(\boldsymbol{x}'_t, y'_t) \in D_A$ with $(\boldsymbol{x}'_{t+1}, y'_{t+1})$
  where $\boldsymbol{x}'_{t+1} = \boldsymbol{x}'_t + \epsilon \cdot \mathrm{sign}(J_F D(\boldsymbol{x}'_t))$, $y'_{t+1} = c(\boldsymbol{x}'_{t+1})$
end for
return θ_D

Conclusions

In this paper, we propose a GAN based incremental learning framework to detect multiple adversarial attacks. The GAN framework is improved to let discriminator learn the modification pattern of adversarial attack. By using the Jacobianbased data augmentation technology we solve the problem of training from limited samples. The experiments show that our incremental learning approach can detect multiple adversarial samples.

Experiments

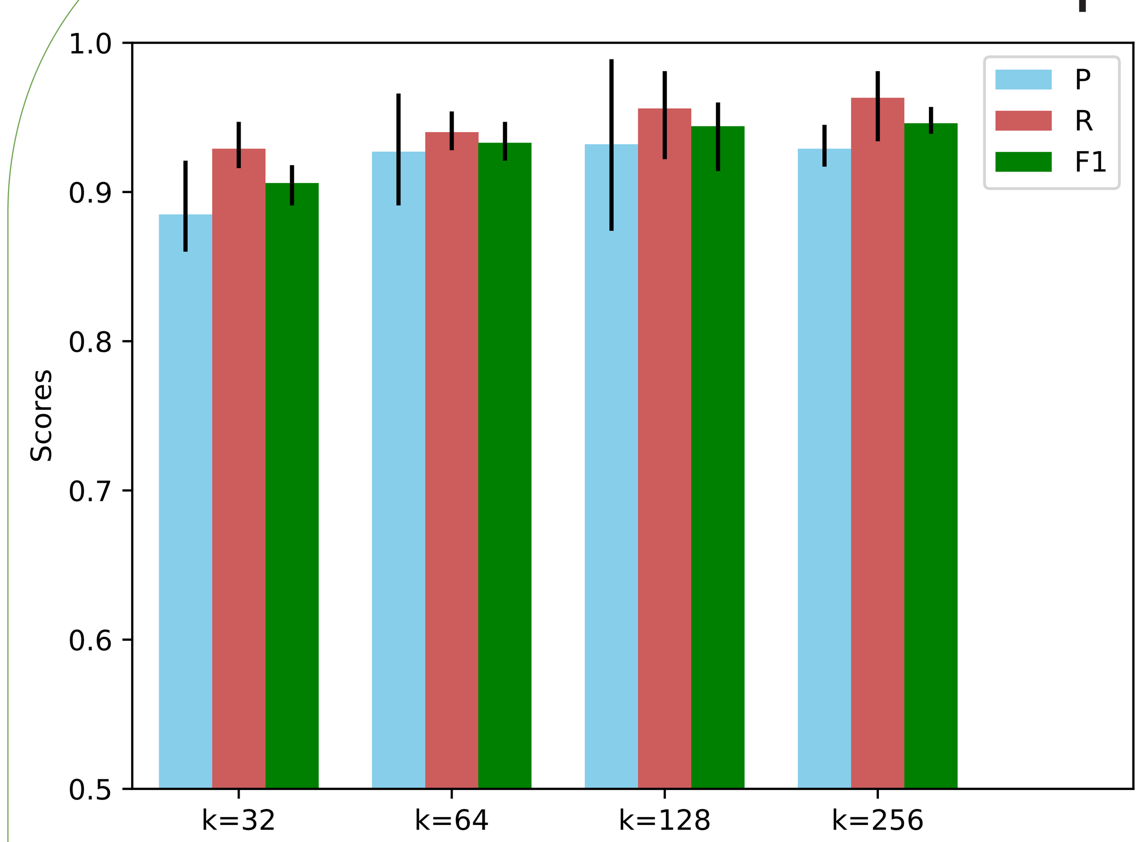

Fig. 4. Detection performance of FGSM's adversarial samples on MNIST dataset. Precision, Recall, and F1-score (P, R, F1) are calculated in 5 times repeated experiments. The macro-averaged P, R, F1 are shown in the figure. The maximum and minimum P, R, F1 in the multiple experiments were also shown. k values in this figure are the number of initial adversarial samples for incremental learning (See Algorithm 1). We can conclude that even with only a small number of adversarial samples, a fine detection performance will be obtained with Jacobian-based data augmentation.

Fig. 3. The examples of original images and the three attacks' adversarial samples. The first row shows the images from MNIST while the images in second row are from CIFAR-10. Three adversarial attacks (FGSM, LSA, PSO) are used to perturb the original images. The three attacks use different modification patterns which cause the images are classified incorrectly.

Table 1. The performance of multiple adversarial attacks detection on two datasets

Dataset	Attack	k	P	R	F1	Dataset	Attack	k	P	R	F1
MNIST	FGSM	32	0.885	0.929	0.906	CIFAR-10	FGSM	32	0.856	0.937	0.895
		64	0.927	0.94	0.933			64	0.927	0.899	0.913
		128	0.932	0.956	0.943			128	0.928	0.967	0.947
		256	0.929	0.963	0.945			256	0.917	0.968	0.942
		baseline	0.886	0.908	0.896			baseline	0.904	0.834	0.868
	LSA	32	0.865	0.769	0.814		LSA	32	0.834	0.822	0.828
		64	0.837	0.819	0.827			64	0.805	0.869	0.835
		128	0.875	0.846	0.86			128	0.864	0.878	0.871
		256	0.93	0.877	0.902			256	0.899	0.905	0.902
		baseline	0.878	0.8	0.837			baseline	0.909	0.669	0.771
	PSO	32	0.887	0.944	0.914		PSO	32	0.861	0.948	0.903
		64	0.913	0.943	0.927			64	0.867	0.967	0.914
		128	0.906	0.97	0.936			128	0.928	0.967	0.947
		256	0.964	0.982	0.972			256	0.935	0.981	0.957
		baseline	0.895	0.917	0.905			baseline	0.894	0.91	0.902