DeepSleep: A Biologically inspired SNN-based Adversarial Defense Method

Zhengyue Zhao

SKL of Computer Architecture, Institute of Computing Technology, CAS

Department of Computer Science and Technology, UCAS

Abstract

CNNs are wildly used and efficient image processing models, the application of CNN includes face recognition, auto pilot and so on. However, recent researches have shown that CNNs can be very unstable for some elaborately designed adversarial noise, which can mislead the CNN model with high rate of success. This is probably a fatal threat to current applications. Compared with CNNs, biological neural networks can obtain higher robustness considered that the perturbation adversarial noise can hardly fool human's eyes. In this paper, we advance a biologically inspired SNN-based adversarial defense method enlightened by slow and fast-wave sleep mechanism of human brains. We evaluate our method on MNIST dataset and compared it with other defense method against multiple adversarial attacks. The result shows that our approach achieves the state of the art compared with other biologically inspired methods. The full code of our approach *DeepSleep* can be downloaded from https://github.com/Zz-0-0-zZ/DeepSleep.

Keywords: Adversarial Defense; Spiking Neural Network; Convolutional Neural Network

1 Introduction

Convolutional neural network (CNN) is one of the representative algorithms of deep learning which has revolutionized the field of computer vision and is also increasingly being used in other fields (Krizhevsky et al. 2012). CNN models perform well in extracting features of the input image, but their lack of generalization presents a weakness and for which adversarial attacks have been proposed. Adversarial attacks generally find the locations the model thinks important, then try to change the pixel value of the location to make the DNN make a misjudgment of the input (Szegedy et al. 2013). It can be applied in many fields such as convex optimization, image classification, and face recognition (Zantedeschi et al. 2017) which would cause great harm.

Adversarial attacks mentioned above have presented a potential security threat to widely deployed artificial neural networks. As a result, it is urgent to propose effective defense methods against these attacks. Uesato et al. (2018) have shown that existing defensive approaches, including defensive distillation (Papernot N. et al. 2016) and adversarial retraining (Madry A. et al. 2017), perform ineffectively against modified versions of attacks. However, these perturbations are imperceptible to human eyes (Szegedy C. et al. 2013), which implies that the biological mechanisms of the human brain can resist such disturbance. Convolutional Neural Networks utilize floating-point numbers and Back Propagation Algorithm (LeCun Y. et al. 1988) to construct an optimizer while Spiking Neural Networks (SNN) (Chosh-Dastidar S. et al. 2009) process information through temporal spikes, consistent with action potentials in biological neural networks. Considering complexity and realizability, the Leaky Integrate-and-Fire (LIF) Neuron Model (Tal D. et al. 1997) has been applied as the basic unit of SNN. Moreover, according to the principles of spike-timing-dependent plasticity (STDP) (Song S. et al. 2000), the sleep algorithm has been proposed to optimize the network by updating weights depending on the relative timing of pre and postsynaptic spikes.

It is speculated that in the mammalian brain, sleep helps to establish knowledge for learning in the awake state (Stickgold et al. 2013; Lewis et al. 2011). Sleep is considered to be the key to memory

consolidation - a process of transforming short-term memory into long-term memory (Rasch et al. 2013). By activating and replaying local synaptic plasticity, such as peak time-dependent plasticity (STDP), synapses involved in learning tasks can be strengthened. Changes in plasticity can improve the ability of subjects to form connections between memories, and summarize knowledge learned in the awake state (Payne et al. 2009). It is assumed that the sleep phase helps to reduce the sensitivity of neural networks to adversarial attacks, and improve generalization performance by reducing the impact of imperceptible input changes on task output. Therefore, we can use the concept of sleep in biology and apply the offline unsupervised "sleep" phase to modify the parameters of a fully connected ANN.

Inspired by the sleep mechanism of the human brain, we propose a biologically inspired SNN-based adversarial defense method *DeepSleep* against adversarial attacks on artificial neural networks. The framework of *DeepSleep* includes: (1) Convert a trained CNN model to an SNN model by mapping the weights from CNN with ReLU units to a network of leaky integrate-fire units and converting input train data to Poisson-distributed spiking activity. (2) Alternate the unsupervised STDP algorithm (used to simulate slow-wave sleep in the brain) and supervised BPTT (Back-propagation Through Time) (Wu Y. et al. 2018) algorithm with adversarial training (used to simulate fast-wave sleep in the brain) to train the SNN model with discrete-distributed neuron pulse signals. (3) Convert the trained SNN model to a robust CNN model by directly mapping the weights from the leaky integrate-fire network to a ReLU network.

Our contributions are summarized below:

1. We propose an effective biologically inspired SNN-based adversarial defense method *DeepSleep* against adversarial attacks on artificial neural networks, which can increase ANN robustness to noise and adversarial attacks.

2. We use a more accurate neuron model to build SNN and propose a more robust SNN training framework. We show that the proposed framework achieves better performance when it's attacked by adversarial attack methods compared with the traditional unsupervised learning method.

3. We evaluate our defense method on a wildly used image datasets MNIST against multiple adversarial methods. The results show that we achieve the SOTA (i.e., the best adversarial accuracy) in adversarial defense methods compared with other biologically inspired methods.

2 Preliminary

In this section, we introduce some necessary background and some related works as well. First, we provide the preliminary knowledge of adversarial attack and adversarial defense methods respectively. Then we introduce the basic knowledge of spiking neural networks. Finally, we describe the biological inspiration of our approach, including slow-wave sleep and fast-wave sleep.

2.1 Adversarial Attack

Adversarial attacks try to change test image by adding perturbation noises to decrease the performance of an artificial neuron networks. Most of related adversarial attack methods attack CNN models, which are wildly used in multiple image tasks such as image classification and image segmentation. Adversarial attacks can be divided into two categories, white-box attacks and black-box attacks. White-box attacks assume that attackers completely know the structure of the the target model, such as network weights and the loss function. Black-box attacks suppose that attackers know nothing about the target model but can get input-output samples of the model. Here we gave a briefly introduction to some well-used adversarial attack methods.

Fast Gradient Sign Method (FGSM) FGSM (Goodfellow I. J. et al. 2014) is a white-box attack method which get perturbation by calculate the CNN model's gradient to input images. The adversarial noise is calculated by applying a symbolic function with perturbation coefficient ϵ on the gradient, and the noise is added to a clear image to get a adversarial sample:

$$x' = x + \epsilon \cdot (\nabla_x J(x, y)) \tag{1}$$

The principle of FGSM attack is that the adversarial perturbation noise increase the loss of neural networks. By limiting the scale of ϵ, the perturbation can be small enough so that it's difficult for human eyes to recognize the adversarial noise.

Project Gradient Descent (PGD) PGD (Madry A. et al. 2017) attack is an iterative gradient-based adversarial attack method. For a linear model, multiple iteration of FGSM hardly increase the attack effect. But for non-linear models, iteration of small scale perturbation can effectively adjust the gradient of loss, which can achieve better attack performance compared with one step with large perturbation scale. The noise-add process of PGD attack is described below:

$$x_{t+1} = \prod_{x+S} \left(x_t + \epsilon \cdot \left(\nabla_x J\left(x_t, y\right)\right)\right) \tag{2}$$

C&W Attack (CW) CW (Carlini N. et al. 2017) is an optimization-based attack method. Adversarial attack problems can be regarded as a optimization problem:

$$\begin{aligned} &minimization \quad \mathcal{D}\left(x, x+\delta\right) \\ &such \quad that \quad \mathcal{C}\left(x+\delta\right) = t, \quad x+\delta \in [0,1]^n \end{aligned} \tag{3}$$

Where $\mathcal{D}$ is the distance between a clear sample and a perturbed sample, $\mathcal{C}$ is a classification and t is the target adversarial label. According to the non-linear constraints of $\mathcal{C}$, it's difficult to solve the optimization problem directly. CW attack applies an objective function f with a scale c to replace the original optimization conditions:

$$\begin{aligned} &minimization \quad \mathcal{D}\left(x, x+\delta\right) + c \cdot f(x+\delta) \\ &where \quad x+\delta \in [0,1]^n \end{aligned} \tag{4}$$

AutoAttack AutoAttack AutoAttack (Croce F. et al. 2020) is an ensemble adversarial attack method including Auto-PGD, FAB and SquareAttack. Auto-PGD can adaptively choose the perturbation scale and the number of iterations of PGD attack. The loss function of Auto-PGD can be cross entropy loss ($APGD_{CE}$) or logits ratio loss ($APGD_{DLR}$).

$$AutoAttack = APGD_{CE} + APGD_{DLR} + FAB + SquareAttack \tag{5}$$

2.2 Adversarial Defense

Similar to adversarial attack methods, defense methods also includes white-box defense and black-box defense against white and black-box attacks respectively. White-box defense is more difficult compared with the black one because a large amount of model information is mastered by attackers. Here we introduce two wildly used ANN-based defense method and a biologically-inspired defense method.

Adversarial Train Adversarial train (Nayebi A. et al. 2017) is a min-max optimization method which adds noise to data during model training to increase the robustness and generalization ability of an ANN model:

$$\min_{\theta} \mathbf{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\Delta x \in \Omega} L(x+\Delta x, y; \theta)\right] \tag{6}$$

Where θ is the parameter of the model, Δx is the perturbation and Ω is the perturbation space. The inner max is to maximize the loss function to mislead the classification model as far as possible, and the outer min is to find the robustest parameter θ that conform to dataset distribution.

Defense Distillation Defense distillation Distillation (Papemot N. et al. 2016) method trains two neural networks with a training dataset. The softmax function of both two networks is alternated by a softmax with distillation temperature T:

$$softmax_T(x) = \frac{e^{\frac{z_i(x)}{T}}}{\sum_{l=0}^{N-1} e^{\frac{z_l(x)}{T}}} \tag{7}$$

The first network is trained with training data X, training label Y and distillation temperature T, and then obtains the soft label $F(X)$ (probability distribution) of each X. The second network is then trained with data X and soft labels $F(X)$, which has the same structure and distillation temperature

with the first network. Trough distillation, the Jacobian amplitude of the network is reduced so that the gradient of the model decreases and the model becomes more smooth, which means that the model is less sensitive to perturbation.

Sleep Defense Sleep Defense (Tadros T. et al. 2019) is a biologically-inspired defense method, which tries to use the unsupervised algorithm STDP to increase the adversarial robustness of ANN. In Sleep algorithm, an ANN is trained firstly and then the structure is converted into a integrate-and-fire spiking neural network. After the SNN is built, a simplified STDP method is applied to modify the network connectivity. Finally, the STDP-trained SNN is transferred into ANN. The STDP process smooth parameters of the neural network and hide the gradient because of the pulse-based training of SNN.

However, above two CNN-base defense method can only defense some weak attacks such as FGSM. But for improved attack method like AutoAttack, both of two method are almost ineffective. Besides, these methods have no biological inspiration. The biologically-inspired Sleep method simply utilize STDP to improve the model robustness, but the accuracy and generalization ability seriously decrease after Sleep process. We consider that the neuron model IF-node used in Sleep Defense is too simple and the unsupervised STDP process reduces features that are learned during CNN training, and most importantly, the defense performance of Sleep method is not good enough facing multiple attacks. Motivated by these facts, we propose *DeepSleep*, a more robust biologically-inspired SNN-based defense method.

2.3 Spiking Neural Network

SNN is designed to simulate the biological neuron in human brains. Different from floating-point learning in artificial neural networks, spiking neural networks are trained with binary signals which imitate neuron pulses in biological neuron networks. There are many kinds of neuron models to simulate biological neurons. One of the simplest models is the Integrate-and-Fire model (IF), and the Leaky Integrate-and-Fire model (LIF) is a model closer to the real neuron (Tal D. et al. 1997). The LIF model can be described as:

$$\tau_m \frac{du(t)}{d(t)} = -[u(t) - u_{rest}] + RI(t) \tag{8}$$

Spiking neuron networks can be trained both by unsupervised learning and supervised learning method. Spike-Timing Dependent Plasticity (STDP) algorithm is a frequently-used unsupervised learning method for SNN training (Song S. et al. 2000). STDP can be simply described as follows: the weight between neurons increases when a pre-synaptic spike induces a post-synaptic spike, and decreases when the post-synaptic dose not spike. The mathematical description of STDP is the weights update formula:

$$\frac{dw_j}{dt} = A_+(w_j)x(t)\sum_n \delta(t - t^n) - A_-(w_j)y(t)\sum_f \delta(t - t^f) \tag{9}$$

The supervised method to train SNN is back-propagation through time (BPTT), which is adapted from the spatial temporal information propagation process in recurrent neural networks (Wu Y. et al. 2018). Compared with STDP, BPTT is more friendly for SNN training and can achieve higher accuracy.

2.4 Slow-wave and Fast-wave Sleep

The sleep duration of humans is about eight hours, which contains 4 or 5 sleep cycle. The sleep cycle of humans can be divided into to types: slow-wave sleep and fast-wave sleep. The duration of slow-wave sleep contains four stages. Stage 1 (falling sleep) and stage 2 (light sleep) are the light sleep phase, and stage 3 (moderate sleep) and stage 4 (deep sleep) are the deep sleep phase. Researches have shown that during slow-wave sleep, synaptic strengths between neurons in biological networks are modified, which helps brain eliminate fatigue e (Stickgold et al. 2013; Lewis et al. 2011).Fast-wave sleep, which is also called rapid eye movement sleep (R.E.M), is a much more active sleep duration compared with the slow-wave sleep. During fast-wave sleep, the activity of neurons in most areas of the brain increases, which helps to form new neural connections and improve the effect of learning and memory because of the reproduction of our memory information (Payne et al. 2009; Rasch et al. 2013). In general, the sleep mechanism plays a vital role in improving the memory and learning ability of the human brain.

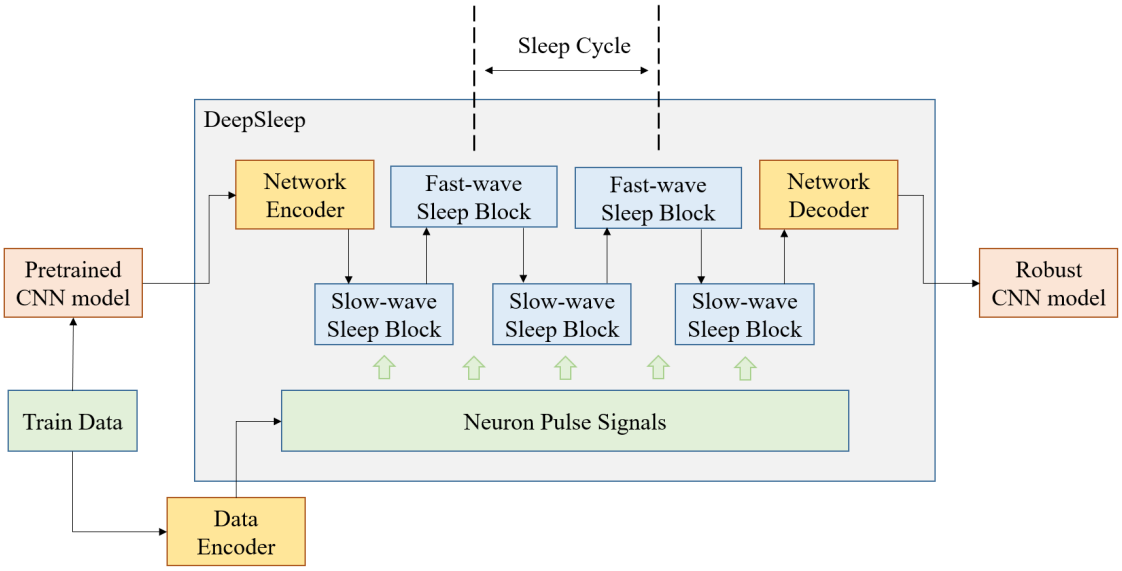

Figure 1: The framework of DeepSleep

3 DeepSleep

Motivated by the sleep mechanism of human's brains and the sleep cycle during sleeping, we propose *DeepSleep*, a biologically-inspired SNN-based adversarial defense method. Overall, we imitate the sleep cycle of slow-wave sleep and fast-wave sleep by SNN-training. We adapt a U-type STDP process to simulate the slow-wave sleep which contains multiple stages, and utilize the BPTT training method with randomly-added adversarial noise to simulate the fast-wave sleep. In this section, we introduce our approch in details.

3.1 Framework

The framework of *DeepSleep* is shown in Figure 1. Inputs of *DeepSleep* includes a CNN model and a binary-sequence dataset. The CNN model is trained on a training dataset and the dataset is transferred into a time spiking sequence dataset trough a Data Encoder. The structure of *DeepSleep* mainly contains four kinds of sub-modules: two converters, Network Encoder and Network Decoder, and two sleep-based algorithm blocks, Fast-wave Sleep block and Slow-wave Sleep block. The aim of network converter is to transfer a CNN model to an SNN model (or transfer an SNN model to a CNN model), and two sleep-based blocks are to imitate slow and fast-wave sleep respectively, a collection of cascaded slow-wave sleep block and Fast-wave sleep block constitutes a Sleep Cycle. Firstly, we convert the pretrained CNN model to an SNN model trough the Network Encoder. Then er alternate the Slow-wave Sleep and Fast-wave Sleep process to simulate the sleep mechanism of the human brain to train the SNN model with neuron pules signals. Finally, the trained SNN is converted to a robust CNN.

3.2 Network Converter

As CNN models are wildly applied and the deployment on GPU is easier, we need converting the model between CNN and SNN before and after *DeepSleep* process. We convert the pretrained CNN model to an SNN model by mapping the weights from CNN with ReLU units to network of LIF-node units and er convert the trained SNN model to a robust CNN model by directly mapping the weight from LIF-node network to ReLU network. In detials, we built an uninitialized SNN model which has the same network structure with the CNN, but replace all the CNN layers with SNN layers. For instance, all ReLU units are alternated by LIF-nodes. Then we extract parameters of the CNN model (e.g. weights of each layer) and utilize these parameters to initialize the SNN model. After processes of sleep blocks, we obtain a trained SNN. Similar to the Network Encoder, we extract parameters of

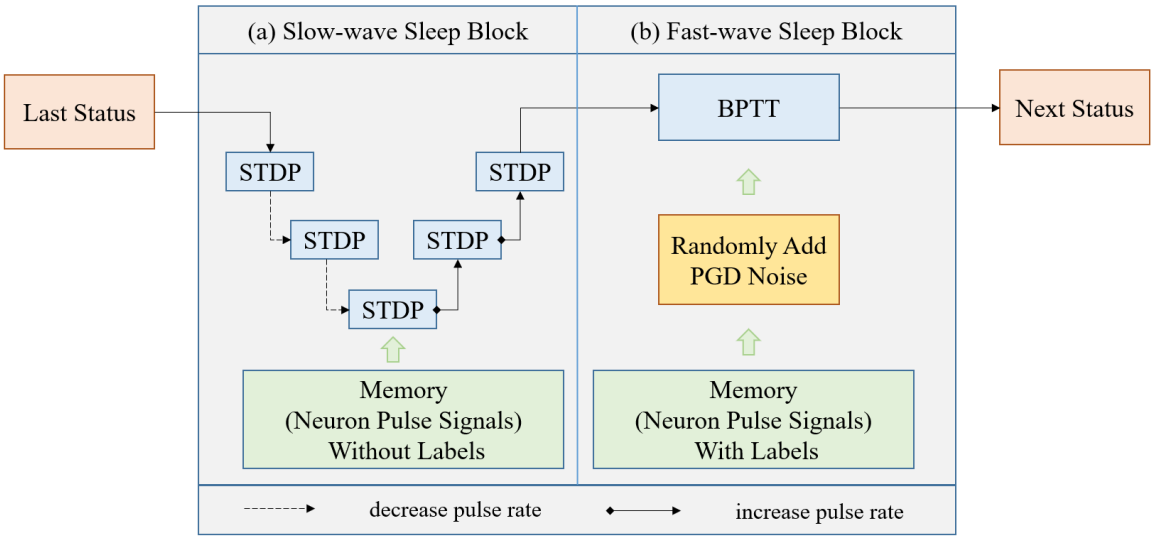

Figure 2: The Sleep Cycle of sleep processes. (a) is the slow-wave sleep block and (b) is the fast-wave sleep block, these two blocks cascaded to construct a sleep cycle block.

the SNN and use these to modify the CNN connections. The static dataset for CNN training is also converted into time spiking binary sequence which represents neuron pulse signals at the same time when the CNN is converted to an SNN.

3.3 Sleep Cycle Process

Same as sleep cycle of brain's sleep mechanism, we divided the proposed sleep process several sleep cycle processes (Figure 1). The detailed structure of a sleep cycle process is shown in Figure 2. Every sleep cycle process is a cascaded construction of a slow-wave sleep block and a fast-wave sleep block.

Slow-wave Sleep Block The slow-wave sleep in the sleep cycle is an unsupervised SNN training process with neuron pulse signals without labels. As shown in Figure 2 (a), we propose an U-type STDP learning structure to imitate the multiple-stage slow-wave sleep in the human's sleep cycle. In the descending part of the U-shaped structure, the rate of neuron pules gradually decrease, which represents that with the deepening of sleep, the activity frequency of neurons in the neural network decrease. And in the rising part the rate gradually increase, which shows that the network progressively enters the shallow sleep state. During the U-type STDP process, connections between neurons in the neural network are modified into a more balance form. i.e., By the propagation of inter-synaptic pulse signal, local features which are learned in awake state (CNN training process) are strengthening, and this helps the model to defense the locally-added-noise attack. Besides, the discrete learning process achieves an effect similar to gradient obfuscation, which means that it's much more difficult for gradient-base attack methods to calculate the gradient of the modified network. The aim of this slow-wave sleep process is to smooth model parameters and reinforce the locally learned features which can increase the robustness of the model.

Fast-wave Sleep Block The fast-wave sleep in the sleep cycle is a supervised SNN training process with randomly adversarial noise-added neuron pulse signal with labels. As shown in Figure 2 (b), after the slow-wave sleep process, the SNN model is trained by back propagation algorithm, and some adversarial noised samples are randomly added to the training spiking dataset during BPTT process. The motivation of this block is the fast-wave sleep or R.E.M sleep, and dreams occur in the human brain during this period. Take into consideration that dreams are usually strongly related to what humans' thinking during awake state and activity of neurons reaches the maximum, we apply the supervised BPTT algorithm in this process. The noised dataset with labels is just like dreams during fast-wave sleep, considering that there is usually a gap between dream and real world. Compared with STDP process, BPTT can learning features with supervising labels to achieve higher accuracy while the back propagation learning makes the model more vulnerable. For this condition, the randomly added adversarial noise can improve the adversarial robustness during the BPTT process.

Noise Scale	*Defense Method*	Clear Sample	FGSM	PGD	BIM	CW	AutoAttack
64/255	Control	0.9917	0.3428	0.0789	0.7415	0.5989	0.0
	Adversarial Train	**0.9925**	**0.9537**	0.6161	**0.9697**	**0.9887**	0.3261
	Defense Distilaltion	0.9916	0.9428	0.5381	0.9637	0.9828	0.2034
	Sleep Defense	0.3025	0.2710	0.2334	0.2971	0.2845	0.2200
	DeepSleep	0.9310	0.9019	**0.7703**	0.9028	0.9081	**0.7531**
128/255	Control	0.9917	0.0478	0.0543	0.7415	0.5989	0.0
	Adversarial Train	**0.9934**	0.8349	0.0699	**0.9713**	**0.9877**	0.0033
	Defense Distilaltion	0.9923	0.6851	0.0618	0.9663	0.9827	0.0029
	Sleep Defense	0.2013	0.1837	0.1523	0.1992	0.2021	**0.1235**
	DeepSleep	0.9255	**0.8648**	**0.2615**	0.8934	0.8988	0.0592

Table 1: Test accuracy of the LeNet classifier with or without defense against multiple attacks.

Generally, the slow-wave process improve the model's adversarial robustness but decrease the generalization ability because of the unsupervised learning method, while the fast-wave process improves model's performance and tries maintaining the robustness as much as possible by adversarial training. In this case, by alternate slow-wave sleep process and fast-wave sleep process, we can get a trade-off between model's robustness and performance, which means that both of these two evaluating indicator are boosted.

4 Experiments

In this section, we introduce our experiment settings and results of the experiment. We test our method by image classification task.

4.1 Experiment Setup

To effectively evaluate our method, we design the experiment on a public image dataset and compare our approach with other three different kinds of defense methods, and the attack models of the experiment contain five efficient attacks.

Dataset We evaluate our method on MNIST, a lightweight wildly used image dataset. MNIST dataset includes gray handwritten digital images from number 0 to number 9.

Attacks Attack methods include FGSM, PGD, BIM (Kurakin A. et al. 2018), CW and AutoAttack. FGSM, PGD and BIM attacks are gradient-based methods, while PGD and BIM attacks are improved iterative version of FGSM. CW attack is a optimization-based method, which regarded attacks as a bounded convex optimization problem. AutoAttack is an ensemble method of multiple different attacks and is one of the strongest attack methods. Some details of these attacks are shown in Part 2.1.

Baseline We compared our method with three defense methods, including two wildly accepted defense model, Adversarial Train and Defense Distillation, and a biologically-inspired defense method Sleep Defense. Details of these defense method are shown in Part 2.2.

Model Settings We apply a simple but efficient CNN-based classifier LeNet-5 to complete the image classification task and super parameters of the classification are chosen by grad search. We implement *DeepSleep* method with one sleep cycle blocks, and the deepth of the U-type slow-wave sleep is 2, the length of the each STDP in U-type process is 1 epoch, adversarial noises in fast-wave sleep are generated by FGSM. These settings built the simplest *DeepSleep* for evaluation in general experiments. Further discussion of *DeepSleep* is given in abolution experiments.

4.2 Results and Analysis

General Experiment We compared *DeepFool* with multiple defense methods against multiple attacks on MNIST dataset, and we use test accuracy to estimate the defense performance. Perturba-

Noise Scale	*Defense Method*	Clear Sample	FGSM	PGD	BIM	CW	AutoAttack
64/255	Control	0.9917	0.3428	0.0789	0.7415	0.5989	0.0
	Slow-wave only	0.5034	0.4768	0.3898	0.4838	0.4960	0.4040
	Fast-wave only	0.9179	0.7430	0.3817	0.8165	0.8561	0.3833
	DeepSleep	0.9310	0.9019	0.7703	0.9028	0.9081	0.7531
	DeepSleep (without noise)	0.9666	0.9118	0.6795	0.9303	0.9385	0.6567
	DeepSleep (depth=3)	0.9352	0.9019	0.7841	0.9104	0.9139	0.7763
	DeepSleep (length=2)	0.9426	0.9137	0.7875	0.9180	0.9202	0.7873
	DeepSleep (cycles=2)	0.8654	0.8370	0.7084	0.8490	0.8521	0.7203
128/255	Control	0.9917	0.0478	0.0543	0.7415	0.5989	0.0
	Slow-wave only	0.4382	0.4044	0.1709	0.4224	0.4316	0.1918
	Fast-wave only	0.9407	0.5275	0.1182	0.8431	0.8892	0.0098
	DeepSleep	0.9255	0.8648	0.2615	0.8934	0.8988	0.0592
	DeepSleep (without noise)	0.9664	0.8375	0.1433	0.9325	0.9433	0.0070
	DeepSleep (depth=3)	0.9356	0.8708	0.2765	0.9132	0.9156	0.1263
	DeepSleep (length=2)	0.9465	0.8749	0.2545	0.9222	0.9254	0.1292
	DeepSleep (cycles=2)	0.8769	0.8300	0.3253	0.8660	0.8607	0.1690

Table 2: Test accuracy in Ablation Experiments

tion scales of adversarial noise contains 64/256 and 128/256, considering that images in MNIST are grayscale images. Results are shown in Table 1. The first line in the Table is test accuracy of control group, which seriously decrease when the classifier is attacked, and the accucray can even decrease to 0 when it's attacked by a strong attack method such as AutoAttack.

Adversarial Train reaches the highest test accuracy against most of attacks when the perturbation is small (64/255). However, Adversarial Train dose not performance well against AutoAttack and PGD. *DeepSleep* achieve the best accuracy against PGD and AutoAttack, which indicates that our approach can handle adversarial attacks which are difficult for CNN-base method to defense. After the increasing of perturbation scales, performance of almost all defense methods descent. Adversarial Train method can still defense BIM and CW attacks with the highest accuracy, but it's performances against PGD and AutoAttack are nearly 0, which means that Adversarial Train and Defense Distillation are ineffective for PGD and AutoAttack with high perturbation noise scale. Sleep Defense achieves the best accuracy against AutoAttack but its performance on other attacks is too bad. Compared with these defense method, performance of *DeepSleep* exceeds Adversarial Train against FGSM attack and maintains its advantage against PGD. Generally, *DeepSleep* performs well against all these attacks and exceed another biologically-inspired defense by great advantage, which indicates that *DeepSleep* is an effective and general defense method and achieves SOTA in biologically-inspired defense.

Ablation Experiment In order to verified and evaluate the effect of each process in *DeepSleep*, we design an ablation experiment in this part. We evaluate the test accuracy with multiple attacks of these forms of *DeepSleep*: (1)Slow-wave sleep only; (2)Fast-wave sleep only; (3)The full model of the simplest *DeepSleep* which is also shown in the General Experiment part. Results are shown in Table 2. We can indicate from Table 2 that slow-wave-only method actually effective for all attacks method and reaches the highest accuracy against AutoAttack with a large perturbation scales (128/255). However, because of the unsupervised process of slow-wave sleep, slow-wave-only method underperforms on clear samples and some weak attacks such as FGSM. Compared with the slow-wave-only method, the fast-wave-only method reaches excellent accuracy on clear samples. By alternating the slow-wave sleep and fast-wave sleep, *DeepSleep* can have a better performance. It's also shown that *DeepSleep* performs better than both fast-wave-only and slow-wave-only method, this is because the cascade-connected structure of these two blocks aggregating the features they extracted separately and the number of training epochs are evidently more than that of each of them.

Table 2 also shows performances of *DeepSleep* with different structures and parameters. As we have mentioned above, the simplest *DeepSleep* contains only one sleep cycle and the depth of slow-wave sleep is 2 while each stage involves only one epoch. We modified the simplest *DeepSleep* by: (1)Remove the random adversarial noised added during fast-wave sleep; (2)Change the depth of slow-wave sleep

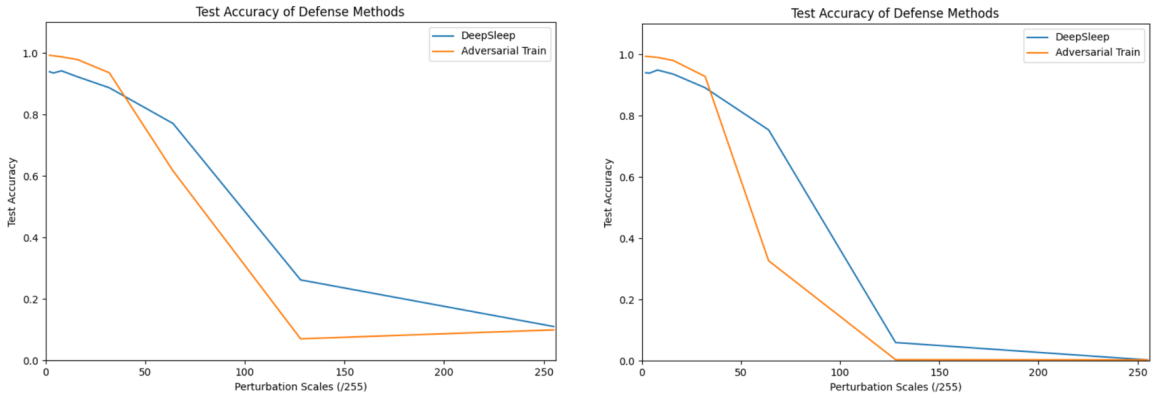

Figure 3: Robustness of *DeepSleep* and Adversarial Train. (a) is the test accuracy against PGD attack and (b) is the test accuracy against AutoAttack.

Noise Scale	Clear Sample	FGSM	PGD	BIM	CW	AutoAttack
2/255	0.9397	0.9386	0.9386	0.9386	0.9206	0.9397
4/255	0.9389	0.9348	0.9346	0.9345	0.9148	0.9386
8/255	0.9501	0.9414	0.9417	0.9417	0.9292	0.9485
16/255	0.9438	0.9253	0.9224	0.9224	0.9228	0.9351
32/255	0.9428	0.9145	0.8866	0.9145	0.9185	0.8907
64/255	0.9310	0.9019	0.7703	0.9028	0.9081	0.7531
128/255	0.9255	0.8648	0.2615	0.8934	0.8988	0.0592
255/255	0.9409	0.3705	0.1099	0.9124	0.9183	0.0025

Table 3: Test accuracy of the simplest *Deepsleep* with multiple scales of noise.

from 2 to 3; (3) Change the length of each stage during slow-wave sleep from 1 epoch to 2 epochs; (4) Change the number of sleep cycles from 1 to 2. Results show that after remove the adversarial train during BPTT of fast-wave sleep, performances of *DeepFool* on PGD and AutoAttack decline sharply, and the descent of performance on clear samples after adding adversarial noised is acceptable (less than 5%). After modified the structure of *DeepSleep* in the way of (2) (3) (4), performances of *DeepSleep* get obvious improvement especially on AutoAttack with large perturbation scales. This fast indicates that our approach of U-type STDP structure in slow-wave sleep process and the design of sleep cycle are effective for attack defense. It's also suggest that the performance of *DeepSleep* can have further improvement by adjusting the structure and parameters.

Adversarial Robustness To evaluate the adversarial robustness of *DeepSleep*, we carry out experiments of *DeepSleep* with different perturbation scales from 0 to 1 (255/255). Results are shown in Table 3. We statics the accuracy attenuation of Adversarial Traing meanwhile and results are shown in Figure 3. It's shown that test accuracy of *DeepSleep* (blue line) decreases later than Adversarial Train (oriange line) when they are attacked by both the PGD and AutoAttack, which shows that *DeepSleep* method can tolerate larger adversarial perturbation scales than Adversarial Train. This fact indicates that *DeepSleep* have great adversarial robustness.

5 Conclusion

The experiment show that we implement an effective biologically-inspired adversarial defense method based on SNN. Results indicate that our approach can successfully defense AutoAttack, one of the strongest attack method for which CNN-based defense methods are almost ineffective. This shows that biologically-inspired methods and SNN training methods indeed have the robustness which CNN-base method dose not have. Compared with another method, our approach performs well both in adversarial robustness (strong attacks) and generalization ability (clear samples and weak attacks), which indicates that *DeepSleep* achieves the SOTA in biologically-inspired defense.

Limitation There are some problems of *DeepSleep* to be solved. Firstly, due to the alternative process in sleep cycle and the high time complexity of STDP, it takes a long time to generate a more robust CNN model. Secondly, although there is no well designed attack against SNN, it should be noted that our defense method and other biologically-inspired methods may be pwned if someone meticulously designs a SNN-based attack method. Besides, because of the limited time, we do not evaluate our method on a huger dataset such as CIFAR and with longer sleep process or deeper slow-wave sleep.

Future Works In the case of limitations mentioned above, we propose some interesting works that may be researched in the future. First, the essence of robustness of SNN compared woth CNN is still an open problem. By extracting the inherent mechanism of robust SNNs or real biological networks, both the robustness and time complexity of SNN and CNN may be improved. Second, there are few works towards adversarial attacks against SNN, but the robustness of SNN should also be concerned considering the increasing application of SNNs.

6 References

- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp. 1097–1105, 2012.
- Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2013.
- Zantedeschi, V., Nicolae, M. I., Rawat, A. (2017, November). Efficient defenses against adversarial attacks. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (pp. 39-49).
- Lu, J., Issaranon, T., Forsyth, D. (2017). Safetynet: Detecting and rejecting adversarial examples robustly. In Proceedings of the IEEE international conference on computer vision (pp. 446-454).
- Chen, P. Y., Zhang, H., Sharma, Y., Yi, J., Hsieh, C. J. (2017, November). Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In Proceedings of the 10th ACM workshop on artificial intelligence and security (pp. 15-26).
- Yao, Z., Gholami, A., Xu, P., Keutzer, K., Mahoney, M. W. (2019). Trust region based adversarial attack on neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 11350-11359). Chen, S., He, Z., Sun, C., Yang, J., Huang, X. (2020). Universal adversarial attack on attention and the resulting dataset damagenet. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Uesato, J., O'donoghue, B., Kohli, P., Oord, A. (2018, July). Adversarial risk and the dangers of evaluating against weak attacks. In International Conference on Machine Learning (pp. 5025-5034). PMLR.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.
- LeCun, Y., Touresky, D., Hinton, G., Sejnowski, T. (1988, June). A theoretical framework for back-propagation. In Proceedings of the 1988 connectionist models summer school (Vol. 1, pp. 21-28).
- Tal, D., Schwartz, E. L. (1997). Computing with the leaky integrate-and-fire neuron: logarithmic computation and multiplication. Neural computation, 9(2), 305-318. Ghosh-Dastidar, S., Adeli, H. (2009). Spiking neural networks. International journal of neural systems, 19(04), 295-308.

- Song, S., Miller, K. D., Abbott, L. F. (2000). Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. Nature neuroscience, 3(9), 919-926.
- Papernot, N., McDaniel, P., Wu, X., Jha, S., Swami, A. (2016, May). Distillation as a defense to adversarial perturbations against deep neural networks. In 2016 IEEE symposium on security and privacy (SP) (pp. 582-597). IEEE.
- Robert Stickgold and Matthew P Walker. (2013). Sleep-dependent memory triage: evolving generalization through selective processing. Nature neuroscience. 16(2): 139.
- Penelope A Lewis and Simon J Durrant. (2011). Overlapping memory replay during sleep builds cognitive schemata. Trends in cognitive sciences, 15(8): 343-351.
- Bjorn Rasch and Jan Born. (2013). About sleep's role in memory. Physiological reviews, 93(2): 681-766.
- Jessica D Payne, Daniel L Schacter, Ruth E Propper, Li-Wen Huang, Erin J Wamsley, Matthew A Tucker, Matthew P Walker, and Robert Stickgold. (2009). The role of sleep in false memory formation. Neurobiology of learning and memory, 92(3):327–334.
- Wu, Y., Deng, L., Li, G., Zhu, J., Shi, L. (2018). Spatio-temporal backpropagation for training high-performance spiking neural networks. Frontiers in neuroscience, 12, 331.
- Nayebi, A., Ganguli, S. (2017). Biologically inspired protection of deep networks from adversarial attacks. arXiv preprint arXiv:1703.09202.
- Tadros, T., Krishnan, G., Ramyaa, R., Bazhenov, M. (2019, September). Biologically inspired sleep algorithm for increased generalization and adversarial robustness in deep neural networks. In International Conference on Learning Representations.
- Goodfellow, I. J., Shlens, J., Szegedy, C. et al. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A. et al. Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.
- Kurakin, A., Goodfellow, I. J., Bengio, S. et al. Adversarial examples in the physical world. In Artificial intelligence safety and security (pp. 99-112). Chapman and Hall/CRC.
- Carlini, N., Wagner, D. et al. Towards evaluating the robustness of neural networks. In 2017 IEEE symposium on security and privacy (sp) (pp. 39-57).