Adversarial robustness of STDP-trained spiking neural networks

서울시립대학교
UNIVERSITY OF SEOUL

CIDA Lab.

Presentation: Baekryun Seong

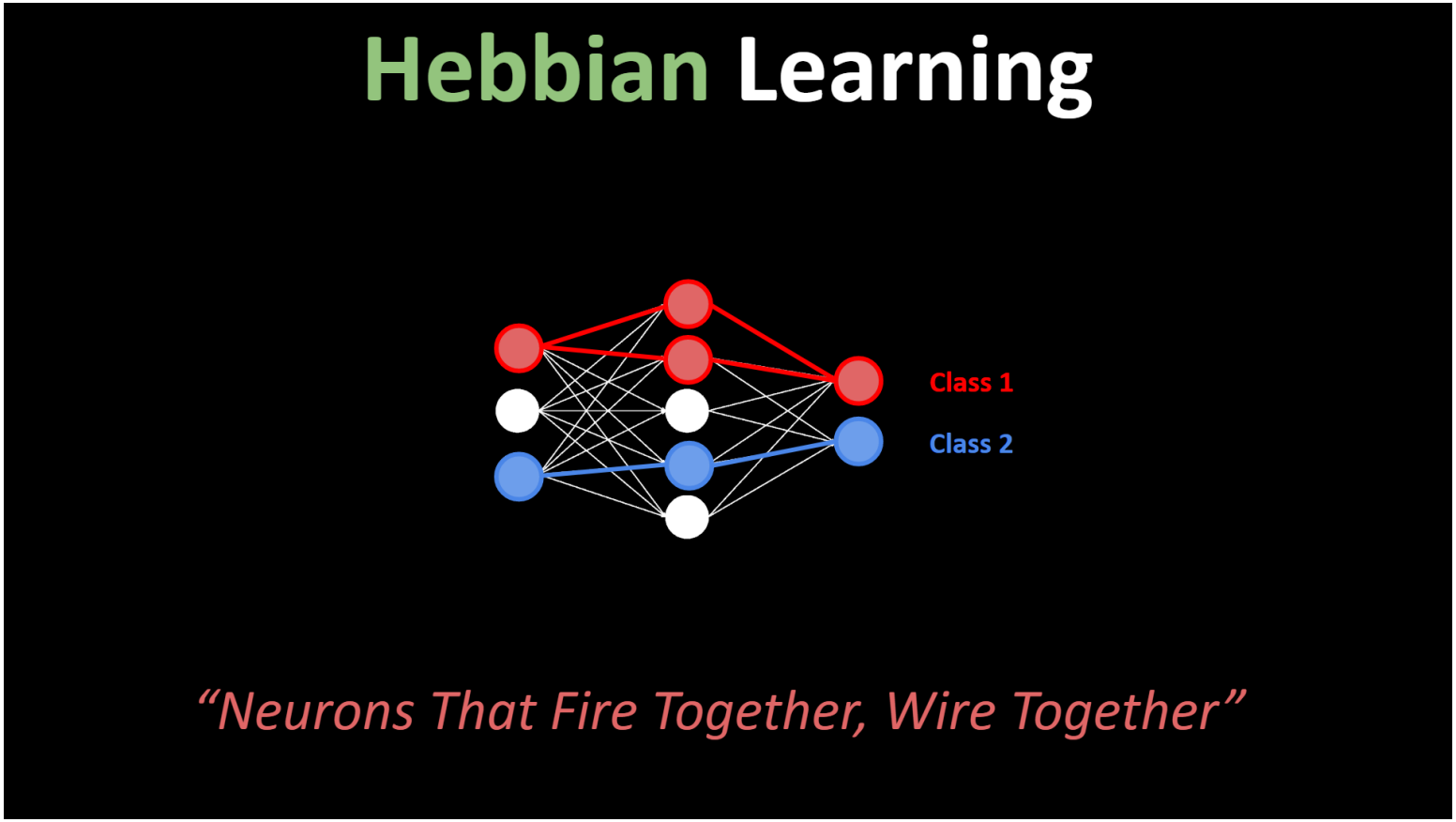
2024.08.01

Table of Contents

"... reinforcement learning has also interacted strongly with psychology and neuroscience, with substantial benefits going both ways. Of all the forms of machine learning, reinforcement learning is the closest to the kind of learning that humans and other animals do, and many of the core algorithms of reinforcement learning were originally inspired by biological learning systems. Reinforcement learning has also given back, both through a psychological model of animal learning that better matches some of the empirical data, and through an influential model of parts of the brain's reward system."

Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

Hebbian Learning



Hebbian Learning Over Reinforcement Learning For Game Intelligence? | by Chintan Trivedi | deepgamingai | Medium

Hebbian Learning

"Let us assume that the persistence or repetition of a reverberatory activity (or "trace") tends to induce lasting cellular changes that add to its stability. ... When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased."

This statement is called Hebb's Theory, or Hebbian learning rule. It is often summarized as

"Neurons that fire together, wire together."

In the early days of artificial intelligence, it was also used for perceptrons, and now it is mainly used for biologically plausible neural networks.

There are non-Hebbian learning dynamics in real neurons, but we will not discuss about it.

Hebb, D. O. (1949). Organization of behavior. new york: Wiley. J. Clin. Psychol, 6(3), 335-307.
Hebbian theory - Wikipedia

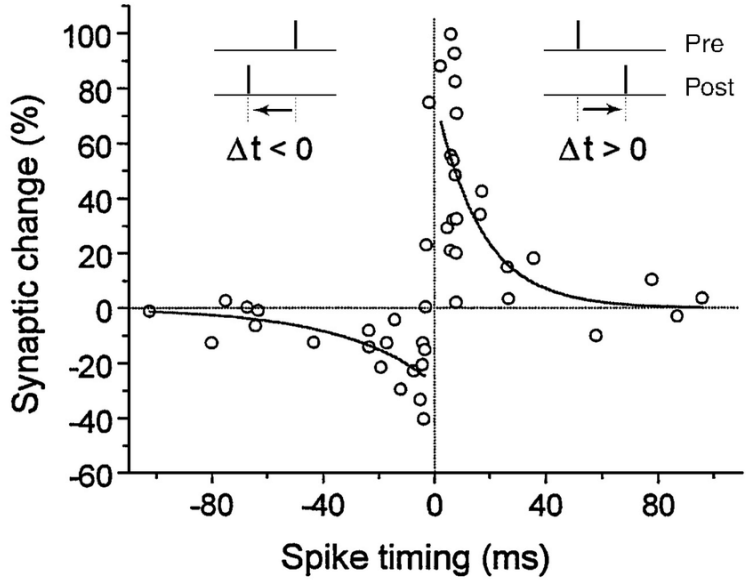
Hebbian Learning: Spike-Timing-Dependent Plasticity

"Spike-timing-dependent plasticity (STDP) is a biological process that adjusts the strength of connections between neurons in the brain. The process adjusts the connection strengths based on the relative timing of a particular neuron's output and input action potentials (or spikes). The STDP process partially explains the activity-dependent development of nervous systems, especially with regard to long-term potentiation and long-term depression."

Spike-timing-dependent plasticity - Wikipedia

In general, the weights of synapses are updated along to expression below:

$$\Delta w = \begin{cases} -A^{-}\exp\left(\frac{\Delta t}{\tau^{-}}\right) & \text{where } \Delta t < 0 \\ A^{+}\exp\left(-\frac{\Delta t}{\tau^{+}}\right) & \text{where } \Delta t \geq 0. \end{cases}$$



In limited range form for biological plausibility:

$$\Delta w = \begin{cases} -A^{-}\exp\left(\frac{\Delta t}{\tau^{-}}\right) & \text{where } \Delta t < 0 \\ A^{+}(w_{max} - w)\exp\left(-\frac{\Delta t}{\tau^{+}}\right) & \text{where } \Delta t \geq 0. \end{cases}$$

In the figure, $A^{+} = 0.86, A^{-} = 0.25,\ \tau^{+} = 19\ ms, \tau^{-} = 34\ ms$.

Spike-timing-dependent plasticity – Wikipedia

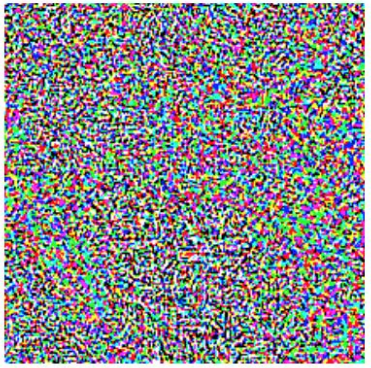
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Adversarial Attacks

$\boldsymbol{x}$

"panda"
57.7% confidence

$+ .007 \times$



$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"nematode"
8.2% confidence

$=$

$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$

"gibbon"
99.3 % confidence

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.

Adversarial Attacks: Categories

White-box attacks: The attackers are provided the target model information.

Black-box attacks: The attackers do not have the information.

Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Fast Gradient Sign Method (FGSM)

$$x^{adv} = x + \alpha \cdot \text{sign}\left(\nabla_x \mathcal{L}(x, y_{true})\right)$$

It can be modified to iterative form for more performance.

$$x_n^{adv} = \begin{cases} x & \text{for } n = 0 \\ \text{Clip}_{x,\epsilon}\left\{x_{n-1}^{adv} + \alpha \cdot sign\left(\nabla_x \mathcal{L}\left(x_{n-1}^{adv}, y_{true}\right)\right)\right\} & \text{for } n > 0 \end{cases}$$

There is more modified version with attack target class.

$$x_n^{adv} = \begin{cases} x & \text{for } n = 0 \\ \text{Clip}_{x,\epsilon}\left\{x_{n-1}^{adv} - \alpha \cdot sign\left(\nabla_x \mathcal{L}\left(x_{n-1}^{adv}, y_{target}\right)\right)\right\} & \text{for } n > 0 \end{cases}$$

Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Methodology

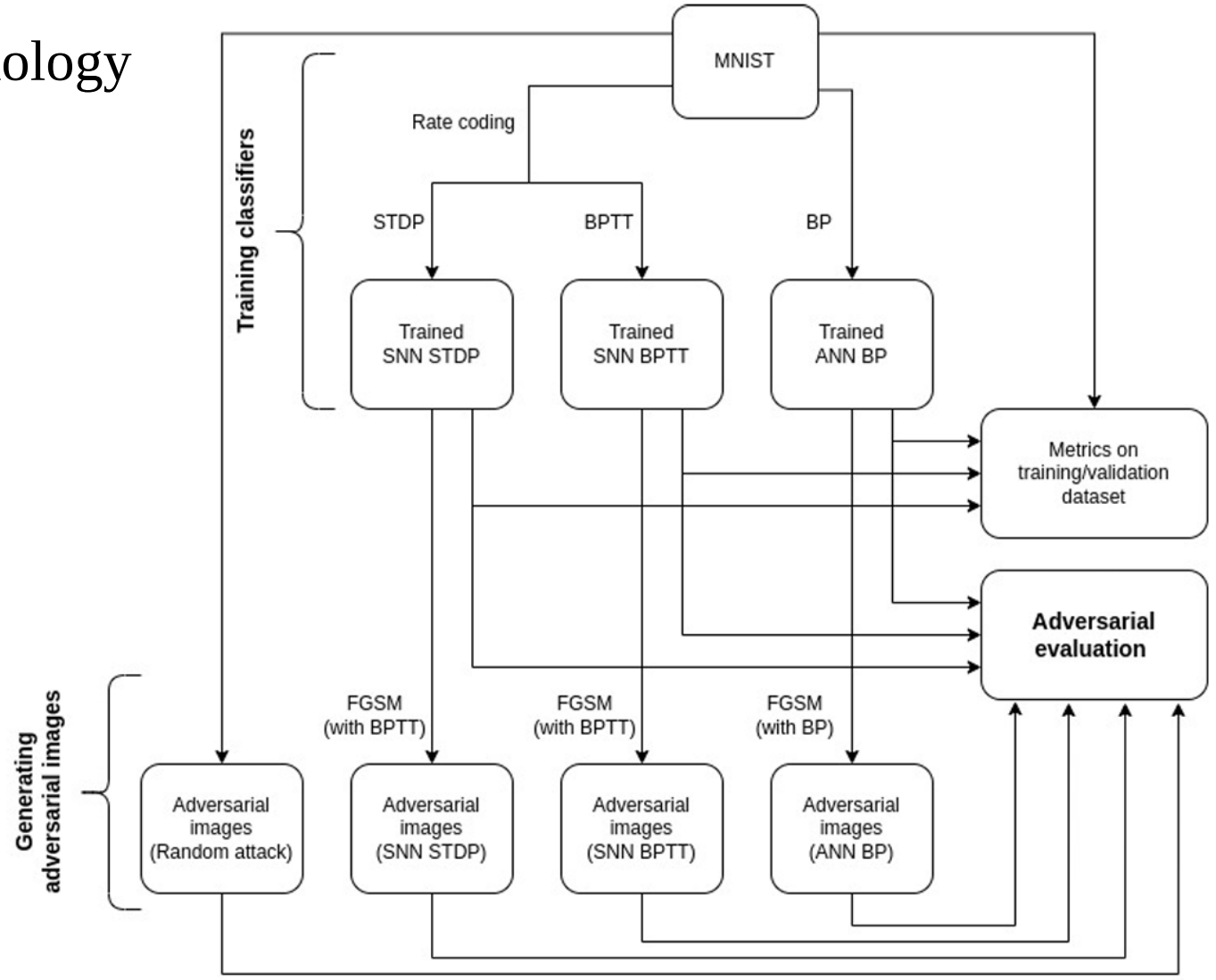


Figure 7: Flowchart of attack- and evaluation process

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572.

Methodology: Data

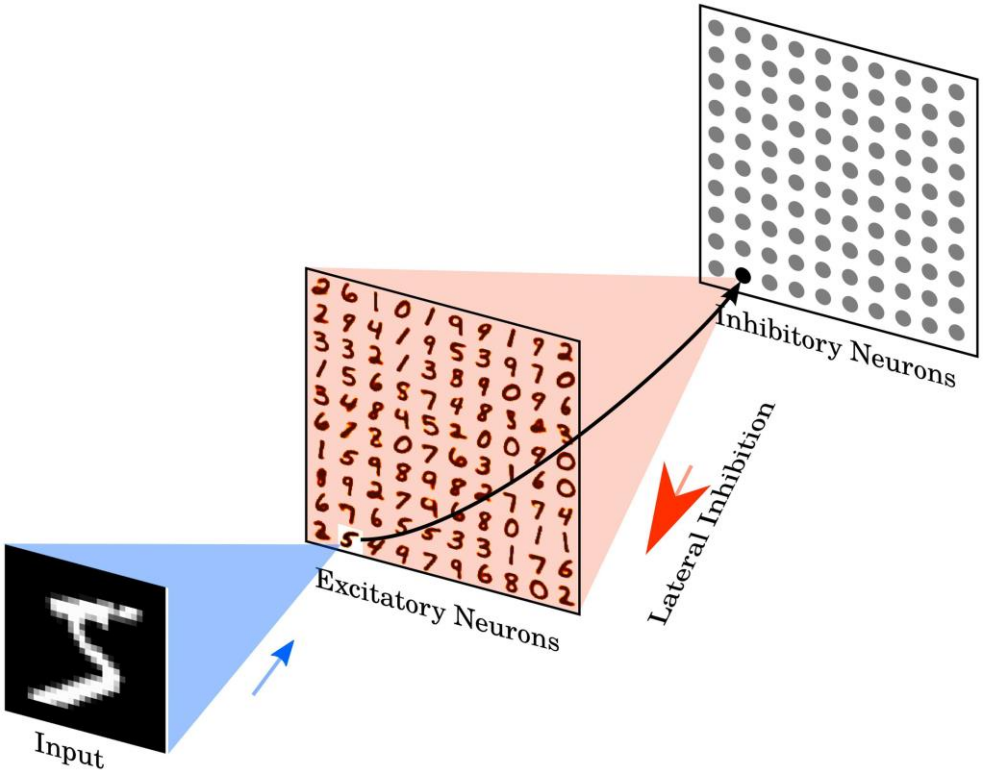
The writers chose MNIST because of two main reasons:

1. STDP method performs poorly on complex image classification problem.
2. Easy to analyze generated adversarial attacks.

Data are preprocessed by resized and gray scaled, and finally normalized.

Methodology: Networks

The authors utilize the model of two layer: Input and processing layer.



Each excitatory neurons are assigned a class, which is the class of the images of digits that has caused most spikes in the neuron.

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Methodology: Networks

2.5. Input Encoding

The input to the network is based on the MNIST dataset which contains 60,000 training examples and 10,000 test examples of 28 × 28 pixel images of the digits 0–9 (LeCun et al., 1998). The input is presented to the network for 350 ms in the form of Poisson-distributed spike trains, with firing rates proportional to the intensity of the pixels of the MNIST images. Specifically, the maximum pixel intensity of 255 is divided by 4, resulting in input firing rates between 0 and 63.75 Hz. Additionally, if the excitatory neurons in the second layer fire less than five spikes within 350 ms, the maximum input firing rate is increased by 32 Hz and the example is presented again for 350 ms. This process is repeated until at least five spikes have been fired during the entire time the particular example was presented.

2.6. Training and Classification

To train the network, we present digits from the MNIST training set (60,000 examples) to the network. Before presenting a new image, there is a 150 ms phase without any input to allow all variables of all neurons decay to their resting values (except for the adaptive threshold). After training is done, we set the learning rate to zero, fix each neuron's spiking threshold, and assign a class to each neuron, based on its highest response to the ten classes of digits over one presentation of the training set. This is the only step where labels are used, i.e., for the training of the synaptic weights we do not use labels.

The response of the class-assigned neurons is then used to measure the classification accuracy of the network on the MNIST test set (10,000 examples). The predicted digit is determined by averaging the responses of each neuron per class and then choosing the class with the highest average firing rate.

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Methodology: Performance of STDP SNN

TABLE 1 | Classification accuracy of spiking neural networks on MNIST test set.

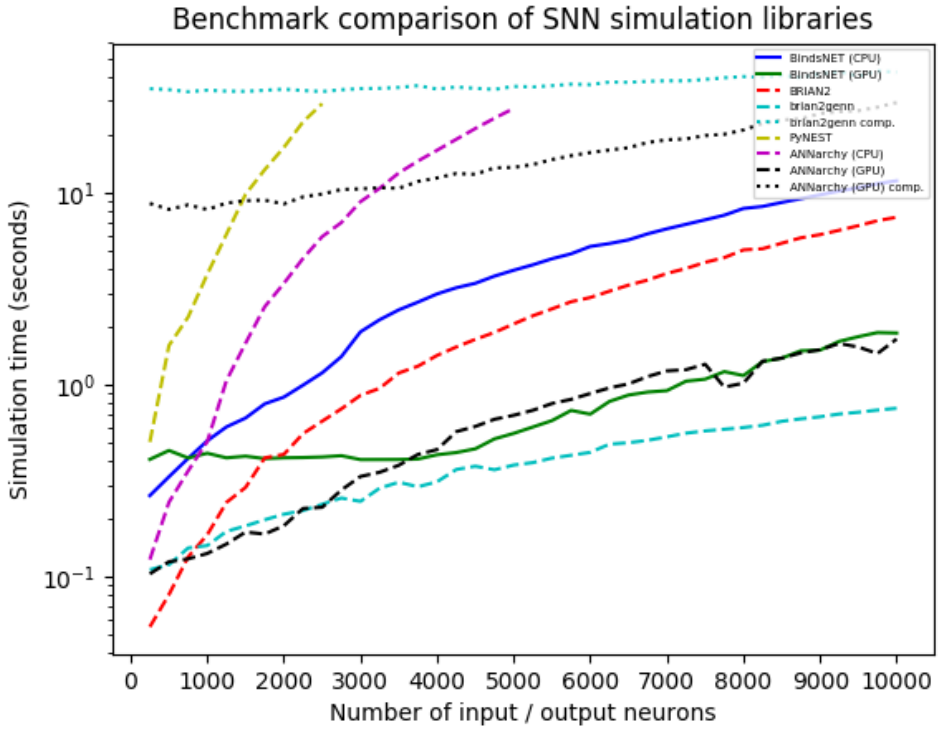
Architecture	Preprocessing	Training-type	(Un-)supervised	Learning-rule	Performance
Dendritic neurons (Hussain et al., 2014)	Thresholding	Rate-based	Supervised	Morphology learning	90.3%
Spiking RBM (Merolla et al., 2011)	None	Rate-based	Supervised	Contrastive divergence, linear classifier	89.0%
Spiking RBM (O'Connor et al., 2013)	Enhanced training set to 120,000 examples	Rate-based	Supervised	Contrastive divergence	94.1%
Spiking convolutional neural network (Diehl et al., 2015)	None	Rate-based	Supervised	Backpropagation	99.1%
Spiking RBM (Neftci et al., 2013)	Thresholding	Rate-based	Supervised	Contrastive divergence	92.6%
Spiking RBM (Neftci et al., 2013)	Thresholding	Spike-based	Supervised	Contrastive divergence	91.9%
Spiking convolutional neural network (Zhao et al., 2014)	Scaling, orientation detection, thresholding	Spike-based	Supervised	Tempotron rule	91.3%
Two layer network (Brader et al., 2007)	Edge-detection	Spike-based	Supervised	STDP with calcium variable	96.5%
Multi-layer hierarchical network (Beyeler et al., 2013)	Orientation-detection	Spike-based	Supervised	STDP with calcium variable	91.6%
Two layer network (Querlioz et al., 2013)	None	Spike-based	Unsupervised	Rectangular STDP	93.5%
Two layer network (this paper)	None	Spike-based	Unsupervised	Exponential STDP	95.0%

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Methodology: Implementation

The python package BindsNET was used for STDP implementation.

BindsNET is implementation of SNN based on pytorch and provides framework to solve ODE.



Hazan, H., Saunders, D. J., Khan, H., Patel, D., Sanghavi, D. T., Siegelmann, H. T., & Kozma, R. (2018). Bindsnet: A machine learning-oriented spiking neural networks library in python. Frontiers in neuroinformatics, 12, 89.

	Numerical value
Number of input neurons	784
Number of hidden neurons	100
Strength of excitatory weights	22.5
Strength of inhibitory weights	120
Increment of membrane potential upon spike	0.05
Length of Poisson spike train per input variable	250
Intensity multiplier for input	128
Excitatory layer weights normalization	78.4

Table 2: Hyperparameters for STDP-network on MNIST

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Methodology: Limitations

- Only MNIST dataset is tested.
- There are various STDP rules, but only one rule is utilized.
- Only FGSM attacks are evaluated.

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Results

	Accuracy (%)
ANN	87
SNN STDP	86
SNN BPTT	94

Table 3: Accuracy of the models on the MNIST test dataset

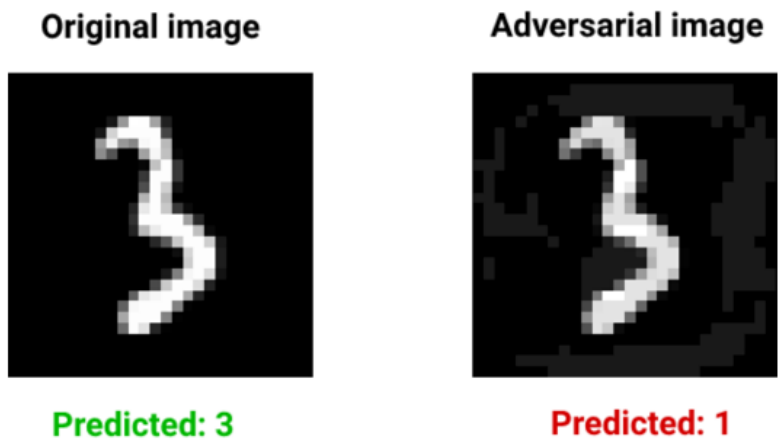


Figure 13: Example of adversarial image generated with FGSM from the STDP-trained network with ϵ set to 0.1

Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Results

	Accuracy (%)
ANN	85.6
SNN STDP	81
SNN BPTT	94.6

Table 4: Accuracy of the models for the subset of 500 images from the MNIST test dataset

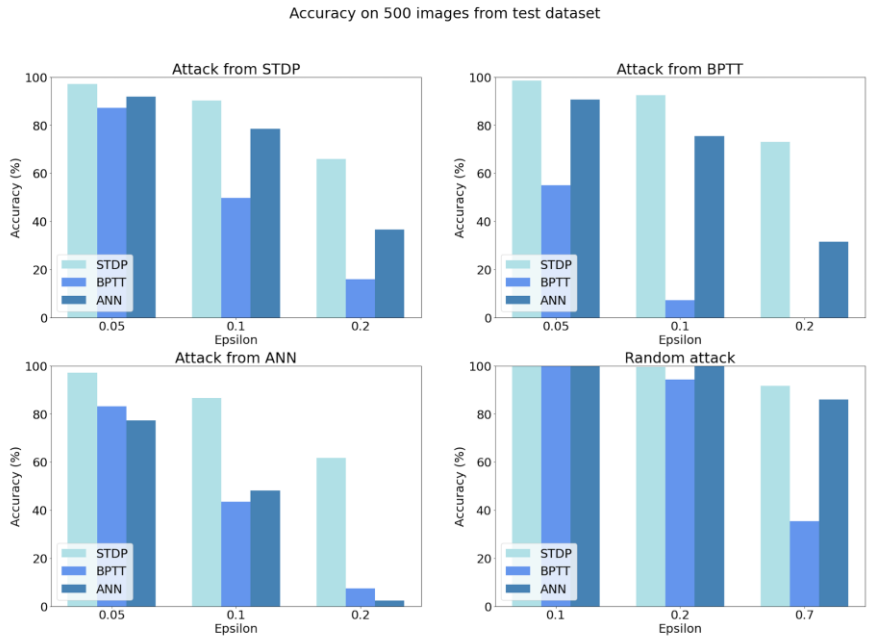


Figure 14: Proportional accuracies after attacks from the STDP-trained model, the BPTT-trained model, the ANN and the random attacker

Results

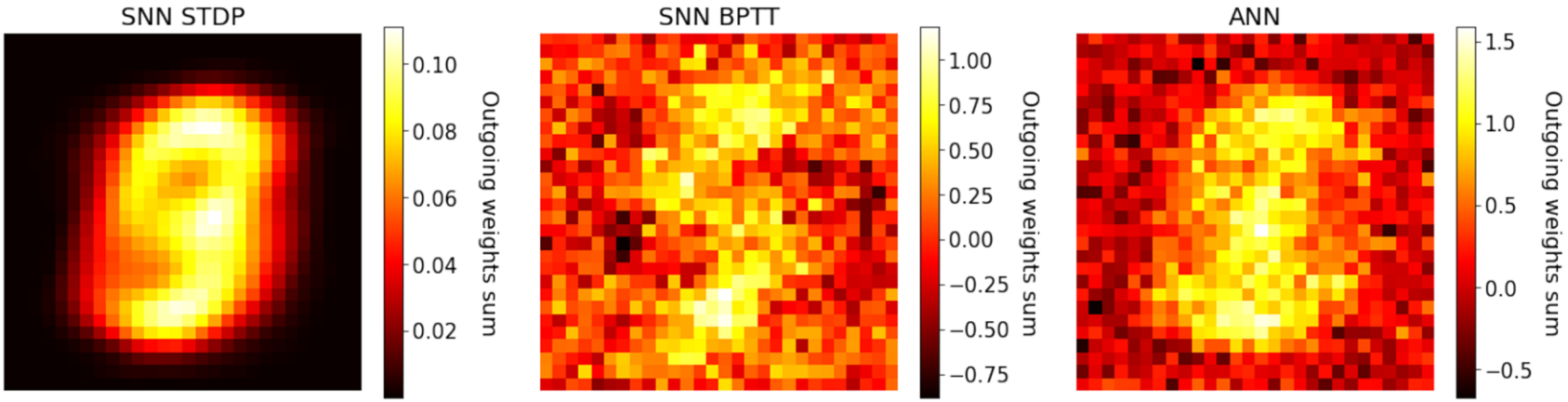


Figure 15: Heatmaps of sum of outgoing weights for each input neuron for the STDP-trained network, BPTT-trained network and ANN

Discussion

Discussion

- FSGM is effective on STDP network. It drops accuracy.
- In fact, it is not sufficient evidence to tell that FSGM is effective. So, authors conducted random attack, and networks defended random attack well.
- ANN is more robust than BPTT SNN. It is different result from previous research.
- Maybe STDP trained network shows high bias, low variance.
- But regularized ANN shows bad performance, despite of similar heatmap.

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.

Discussion: Future work

- Why does STDP-SNN have adversarial robustness?
  - Analyze STDP-SNN with better accuracy.
    - More datasets.
- Analyze the effect of hyperparameters and architectures to robustness
  - More categories of attacks.

Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in computational neuroscience, 9, 99.
Lindblad, K., & Nilsson, A. (2023). Adversarial robustness of STDP-trained spiking neural networks.