Poster: Backdoor Attacks on Spiking NNs and Neuromorphic Datasets

Gorka Abad
Radboud University
Nijmegen, The Netherlands
Ikerlan Technology Research Centre
Arrasate - Mondragón, Spain
abad.gorka@ru.nl

Oğuzhan Ersoy
Radboud University
Nijmegen, The Netherlands
oguzhan.ersoy@ru.nl

Stjepan Picek
Radboud University
Nijmegen, The Netherlands
stjepan.picek@ru.nl

Víctor Julio Ramírez-Durán
Ikerlan Technology Research Centre
Arrasate - Mondragón, Spain
vjramirez@ikerlan.es

Aitor Urbieta
Ikerlan Technology Research Centre
Arrasate - Mondragón, Spain
aurbieta@ikerlan.es

ABSTRACT

Neural networks provide state-of-the-art results in many domains. Yet, they often require high energy and time-consuming training processes. Therefore, the research community is exploring alternative, energy-efficient approaches like *spiking neural networks* (SNNs). SNNs mimic brain neurons by encoding data into sparse spikes, resulting in energy-efficient computing. To exploit the properties of the SNNs, they can be trained with neuromorphic datasets that capture the differences in motion. SNNs, just like any neural network model, can be susceptible to security threats that make the model perform anomalously. One of the most crucial threats is the backdoor attacks that modify the training set to inject a trigger in some samples. After training, the neural network will perform correctly on the main task. However, under the presence of the trigger (backdoor) on an input sample, the attacker can control its behavior. The existing works on backdoor attacks consider standard datasets and not neuromorphic ones.

In this paper, to the best of our knowledge, we present the first backdoor attacks on neuromorphic datasets. Due to the structure of neuromorphic datasets, we utilize two different triggers, i.e., static and *moving* triggers. We then evaluate the performance of our backdoor using spiking neural networks, achieving top accuracy on both main and backdoor tasks, up to 99%.

CCS CONCEPTS

• **Security and privacy**; • **Computing methodologies** → **Computer vision**; **Machine learning**;

KEYWORDS

Backdoor attacks, Spiking neural networks, Neuromorphic datasets

CCS '22, November 7–11, 2022, Los Angeles, CA, USA

ACM ISBN 978-1-4503-9450-5/22/11.
https://doi.org/10.1145/3548606.3563532

ACM Reference Format:
Gorka Abad, Oğuzhan Ersoy, Stjepan Picek, Víctor Julio Ramírez-Durán, and Aitor Urbieta. 2022. Poster: Backdoor Attacks on Spiking NNs and Neuromorphic Datasets. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7–11, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3548606.3563532

1 INTRODUCTION

Deep neural networks (DNNs) have top performance in many challenging machine learning tasks, like image recognition [8] and speech recognition [5]. However, DNNs require numerous layers, neurons, and parameters to achieve such outstanding performances, resulting in high energy consumption. For example, training one of the state-of-the-art models, GPT-3, took weeks with powerful machines and consumed an estimated 190,000 kWh of electricity [2].

As an alternative, energy-efficient neural networks are currently being researched [3, 7]. A new generation of brain-inspired neural networks, *spiking neural networks* (SNNs) [3, 4, 10], is being developed to perform deep learning tasks while consuming low energy. SNNs resemble brain neurons by encoding data into spikes, electrical impulses caused by neuron activation. Instead of binary encoding through pulses, timing is another dimension used to carry more data from a single spike. SNNs encode this information in a sparse format, which is an efficient way of storing the data since, most of the time, neurons are at rest and do not fire a spike. Deep SNNs can have up to 12.2× better energy efficiency than DNNs with similar parameters [9]. Even though SNNs can handle *regular* data, they can also handle neuromorphic data, which is captured by a *Dynamic Vision Sensor* (DVS) camera, creating event-driven datasets, see Figure 1.

SNNs, and in general, DNNs, are vulnerable to privacy and security attacks, e.g., adversarial examples [13] and backdoors [6]. Backdoors are a particular case of data poisoning attacks, where the attacker poisons some (small) part of the training set with a trigger—resulting in the input misclassification only under the presence of the trigger. While training, the model recognizes both the unaltered samples and those with the trigger. Then, at inference, the model works as expected except under the presence of a trigger in a patch input. To the best of our knowledge, there are no works considering the security of SNNs. To address it, we create a set of

attacks to construct a lower bound of backdoor success guarantees. We evaluate the backdoor success using state-of-the-art models and neuromorphic datasets as a case study. The algorithms for generating neuromorphic backdoors are available online[1] for evaluation and reproducibility of our work. Our main contributions are:

- We introduce the first backdoor attacks on neuromorphic datasets and SNNs.
- We validate the efficiency of this attack under state-of-the-art SNNs.
- We additionally develop a moving backdoor, which is stealthier, yet maintains excellent performance.

2 BACKGROUND

Spiking neural networks. An SNN is a type of DNN that utilizes *spiking neurons*. Similar to regular neurons, spiking neurons work on a weighted sum of inputs. Instead of using nonlinear activation functions, e.g., ReLu, spiking neurons are cumulatively *excited* until reaching a threshold θ. Once reached, the spiking neurons *fire* and the weights are propagated to the next layer; then, the spiking neurons get reset [3]. The spiking ability enables SNNs to spike coding for efficient computing.

Neuromorphic data is commonly used in spiking neural networks, copying the behavior of the brain. The sensory system is more responsive to changes in the input space rather than no changes or static data [3]. This behavior can be created in the real world using DVS cameras, which respond to brightness changes, i.e., polarity. Different trigger polarities create different trigger colors, as seen in Figure 1. An incoming change is represented by a green pixel (ON polarity). An outgoing change is represented by a dark blue pixel (OFF polarity). In contrast, combining both polarities results in a light blue pixel (mixed polarity).

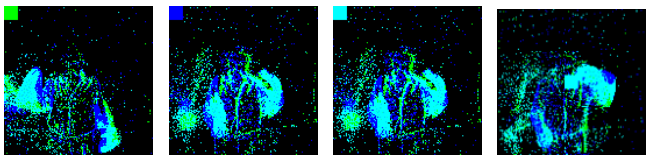
NN Training. A parameterized function $\mathcal{F}_\theta : \mathbb{R}^N \rightarrow \mathbb{R}^M$ maps an input $\mathbf{x} \in \mathbb{R}^M$ to an output $y \in \mathbb{R}^M$. The training process aims to find the parameters minimizing the distance between the network's prediction and the ground truth label y. It is done by leveraging a loss function $\mathcal{L}$ via an iterative process following Eq. (1):

$$\theta' = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n} \mathcal{L}(\mathbb{F}_\theta(\{\mathbf{x}_i, y_i\})). \tag{1}$$

Backdoor attacks. A backdoor in a deep learning model includes a hidden behavior that is only activated under the presence of a specific trigger in the input. The hidden behavior usually misclassifies some inputs labeled by an attacker's chosen target label. A backdoor is created using triggers $\mathcal{T}$, target labels $\mathcal{Y}$, and a backdoor creation function $\mathcal{A}$. $\mathcal{A}$ is defined as $\mathcal{A}(\mathbf{x}, t_i, k, s) = \{\hat{\mathbf{x}}, \hat{y}\}$ where $\mathbf{x}$ is the input sample, $\hat{y} \in \mathcal{Y}$ is the target label, $t_i \in \mathcal{T}$ is the trigger, k is the trigger position, and s is the trigger size expressed as the percentage of the input. The amount of poisoned data is controlled by $\epsilon = \frac{m}{n+m}; m \ll n$, where n is the size of the clean set and m is the poisoned one. Thus, the training process gets modified to include the backdoor behavior:

$$\theta' = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n} \mathcal{L}(\mathbb{F}_\theta(\{\mathbf{x}_i, y_i\})) + \sum_{j=1}^{m} \mathcal{L}(\mathbb{F}_\theta(\{\hat{\mathbf{x}}_j, \hat{y}_j\})). \tag{2}$$

[1] https://github.com/GorkaAbad/NeuromorphicBackdoors



(a) ON polarity **(b) OFF polarity** **(c) Mixed polarity** **(d) Moving trigger**

Figure 1: Static and moving backdoors with different trigger polarities.

3 OUR ATTACK

In our backdoor attack, we utilize the generic parameters mentioned in the previous section (ϵ, s, and k) and parameters specific to neuromorphic datasets such as the polarity. In image recognition, static backdoor attacks create the trigger in one fixed position for all samples [6]. On the other hand, dynamic backdoors can vary their position between different samples [11]. Since neuromorphic datasets are motion captures, triggers can vary their position on the same input. Thus, we can intuitively leverage *moving* backdoors that change in polarity and location, even in the same input sample. We additionally leverage other backdoor parameters to be used in the neuromorphic backdoor attack, i.e., k, s, ϵ, p (the polarity of the trigger), and trigger type: either static or *moving*.

3.1 Static and Moving Triggers

Static triggers have shown good backdoor performance and stealthiness over the years [6]. In the neuromorphic domain, placing a static square over time with a given polarity succeeds in tricking the model into focusing on the trigger, as we show in our experimental results. Since the trigger has a given polarity, the model is tricked into focusing on it, pretending it is a change in the illuminations (movement) and, therefore, relevant. However, under human inspection, a static trigger of a specific polarity that is not moving is easily noticeable, see Figures 1a, 1b, and 1c. Following that intuition, we decided to investigate the viability of stealthier triggers. Our moving triggers change the position horizontally over time, which makes it congruent with the polarity, thus making it more complicated for the human eye to detect. When placed near the action of the image, the trigger merges with the action, making the procedure more natural and thus stealthier, see Figure 1d.

3.2 Experimental Results

We train two spiking neural networks using the DVS Gesture [1] and N-MNIST [12] datasets. For our experiments, we attack state-of-the-art networks proposed by Fang et al. [4]. For the DVS Gesture dataset, the network is composed of five convolutional and one fully connected layer. In the N-MNIST case, it is composed of two convolutional and a single fully connected layer. Regarding the datasets, we used neuromorphic datasets captured by a DVS camera. The DVS Gesture dataset is composed of 128×128 data samples, containing 11 hand gestures from 29 subjects and three different illumination conditions. The N-MNIST is created by converting the 10,000 samples of the static MNIST into events streams with a

DVS camera. Although the size of the samples is increased to 34×34 from 28×28, it still maintains ten classes.

We define the amount of poisoned data in training set with ϵ 0.001, 0.01, and 0.1. We also use three trigger positions, $k \in \{top\text{-}left, middle, bottom\text{-}right\}$. The trigger size, s, is set to 0.1 of the input size. Lastly, we try different trigger polarities p. Our backdoor takes $\mathbf{x}$ as input and constructs a poisoned sample $\hat{\mathbf{x}}$, as seen in Figure 1. The model with the DVS Gesture dataset is trained for 65 epochs, achieving an accuracy of 76% and N-MNIST for 20, achieving an accuracy of 99%, as in [4]. Then, we evaluate two metrics: main task accuracy and backdoor accuracy.

Table 1: Result of our attack under different settings.

Dataset	ϵ	k	s	p	Static/Moving	Epochs	Main task accuracy	Backdoor accuracy
DVS Gesture	-	-	-	-	-	65	76%	-
DVS Gesture	0.01	top-left	0.1	2	Static	65	76%	1%
DVS Gesture	0.1	middle	0.1	0	Static	65	74%	99%
DVS Gesture	**0.1**	**bottom-right**	**0.1**	**1**	**Static**	**65**	**76%**	**100%**
DVS Gesture	**0.01**	**top-left**	**0.3**	**0**	**Static**	**65**	**76%**	**99%**
DVS Gesture	**0.1**	**bottom-right**	**0.1**	**1**	**Moving**	**65**	**76%**	**99%**
N-MNIST	-	-	-	-	-	20	99%	-
N-MNIST	**0.001**	**bottom-right**	**0.1**	**0**	**Static**	**20**	**99%**	**98%**
N-MNIST	0.001	middle	0.1	1	Static	20	97%	1%
N-MNIST	0.01	top-left	0.1	0	Static	20	98%	100%
N-MNIST	**0.001**	**middle**	**0.1**	**0**	**Moving**	**20**	**98%**	**93%**

Results on the N-MNIST dataset (Table 1) show that static and moving backdoor performance is outstanding when the trigger is in the corners and regardless of the polarity. Experiments show that with $\epsilon = 0.01 \wedge s = 0.1$ (roughly 50 poisoned images), the backdoor achieves 99% accuracy on the backdoor task with less than 1% degradation on the main task. However, when the trigger is centered, the backdoor accuracy drops to 1%, and the main task accuracy lowers by 3%. Since the samples are centered, the model cannot distinguish between the trigger and the clean image when placing the trigger over the main action, i.e., $k = middle$. However, this effect is not visible in the DVS Gesture dataset since the action is not happening in the middle. Surprisingly, with a moving trigger placed in the middle, see Figure 1d, the model can recognize it, achieving outstanding results, even when the trigger is hardly noticeable to a human.

Results on the DVS Gesture dataset reveal that static and moving backdoor performance is up to 100% backdoor accuracy. However, some parameters have to be carefully chosen. Since the DVS Gesture dataset is small (1,342 samples), the ϵ selection influences the backdoor performance. Our experiments reveal that with $\epsilon = 0.001 \wedge 0.01$ roughly (1 or 10 poisoned samples) is not enough to inject the backdoor, achieving at best 20% backdoor accuracy with $s = 0.1$. A larger trigger size $s = 0.3$ provides better results without reducing the model performance on the main task. Experiments with $\epsilon = 0.1, s = 0.1$ show up to 100% backdoor accuracy when placing the trigger in the corners and up to 99% when placed in the middle. In both scenarios, the model is not degraded on the main task.

Overall, the spiking neural network focuses on polarity changes. When placing the trigger (static or moving), the model must pay attention to it. We observed that the attack achieves high accuracy if the static trigger does not overlap the clean image's main action and, at the same time, maintains high accuracy on the main task. We also noticed that overlapping the main action is not problematic and is stealthier when the trigger is moving.

4 CONCLUSIONS AND FUTURE WORK

In this work, we have identified and evaluated new security concerns regarding neuromorphic datasets and spiking neural networks, motivated by the increasing usage tendency of low energy-consuming neural networks. We showed that neuromorphic datasets could be easily attacked using existing backdoor attacks. However, *classic* backdoor methods are not stealthy; a human eye can easily detect them. Thus, we leverage the nature of neuromorphic datasets by injecting *moving* backdoors, which are stealthier while being powerful. We have implemented our attack in the image recognition domain on two different architectures and datasets. We evaluated the performance of our attack under different settings with different parameters achieving high accuracy on both main and backdoor tasks. A greater focus on the usage of *moving* triggers could produce interesting findings that account more for stealthier triggers that could potentially evade state-of-the-art defenses. Although this research established the viability of our attack under two datasets, further experiments using a broader set of datasets and spiking neural networks would shed more light on the security of this type of neural network.

ACKNOWLEDGMENTS

This research is funded by the Horizon Europe, Spanish CDTI, and ELKARTEK programs. Grant agreements 101021911 (IDUNN), CER-20191012 (EGIDA), and KK-2021/00091 (REMEDY), respectively.

REFERENCES

[1] Arnon Amir, Brian Taba, David Berg, Timothy Melano, Jeffrey McKinstry, Carmelo Di Nolfo, Tapan Nayak, Alexander Andreopoulos, Guillaume Garreau, Marcela Mendoza, et al. 2017. A low power, fully event-based gesture recognition system. In Proceedings of the IEEE conference on CVPR. 7243–7252.

[2] Payal Dhar. 2020. The carbon impact of artificial intelligence. Nat. Mach. Intell. 2, 8 (2020), 423–425.

[3] Jason K Eshraghian, Max Ward, Emre Neftci, Xinxin Wang, Gregor Lenz, Girish Dwivedi, Mohammed Bennamoun, Doo Seok Jeong, and Wei D Lu. 2021. Training spiking neural networks using lessons from deep learning. arXiv preprint arXiv:2109.12894 (2021).

[4] Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. 2021. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 2661–2671.

[5] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In 2013 IEEE international conference on acoustics, speech and signal processing. Ieee, 6645–6649.

[6] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. IEEE Access 7 (2019), 47230–47244.

[7] Tara Hamilton. 2021. The best of both worlds. Nature Machine Intelligence 3, 3 (2021), 194–195.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems 25 (2012).

[9] Souvik Kundu, Gourav Datta, Massoud Pedram, and Peter A Beerel. 2021. Spike-thrift: Towards energy-efficient deep spiking neural networks by limiting spiking activity via attention-guided compression. In Proceedings of the IEEE/CVF WACV. 3953–3962.

[10] Jun Haeng Lee, Tobi Delbruck, and Michael Pfeiffer. 2016. Training deep spiking neural networks using backpropagation. Frontiers in neuroscience 10 (2016), 508.

[11] Tuan Anh Nguyen and Anh Tran. 2020. Input-aware dynamic backdoor attack. Advances in Neural Information Processing Systems 33 (2020), 3454–3464.

[12] Garrick Orchard, Ajinkya Jayawant, Gregory K Cohen, and Nitish Thakor. 2015. Converting static image datasets to spiking neuromorphic datasets using saccades. Frontiers in neuroscience 9 (2015), 437.

[13] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In ICLR. http://arxiv.org/abs/1312.6199