Efficient Malware Classification with Spiking Neural Networks: A Case Study on N-BaIoT Dataset

Muhammad Umair
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
1221400084@student.mmu.edu.my

Wooi-Haw Tan
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
twhaw@mmu.edu.my

Yee-Loo Foo*
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
ylfoo@mmu.edu.my

***Abstract*—In recent years, there has been growing interest in the application of spiking neural networks (SNNs) for classification tasks. Compared to traditional neural networks, Spiking Neural Networks (SNNs) are a class of neural networks that model the dynamics of biological neurons, where information is represented in the form of spikes or action potential. This study explores the effectiveness of spiking neural networks (SNNs) in classifying the N-BaIoT dataset. SNNs model the dynamics of biological neurons and represent information through spikes or action potentials, enabling them to process temporal information and exhibit event-driven behaviour. This makes them a promising alternative for resource-constrained environments and low-power neuromorphic hardware implementations. The dataset was balanced and split into training and testing sets, and a two-layered SNN model with two LIF neurons was developed. The model achieved 71% accuracy on the test dataset, highlighting the importance of pre-processing steps for reliable results. While the accuracy may not be as high as some deep neural network models, the SNN's event-triggered feature and sparse representation of information through spike patterns make it a promising alternative for classification tasks.**

Keywords—Spiking neural network, N-BaIoT, SNN, IoT attacks

I. INTRODUCTION

The Internet of Things (IoT) has seen a tremendous growth in recent years, with billions of connected devices and sensors spread across the globe [1, 2]. While this interconnected network of devices offers numerous benefits, such as improved convenience and automation, it also creates a vast attack surface for cybercriminals [3-5]. IoT devices often lack adequate security measures, making them vulnerable to malware attacks that can compromise sensitive data or even cause physical harm [5]. Malicious software (malware) is among the frequently encountered forms of cyber threats that cause significant damage to individuals and organizations [4]. With the increasing number of internet-connected devices and the Internet of Things (IoT) devices, the threat of malware attacks has become even more pronounced [6]. Malware attacks can infect devices and systems through various means, such as phishing, social engineering, and software vulnerabilities [7, 8].

To counter the threat of malware attacks, various machine learning-based methods have been proposed for malware classification and detection [9-12]. Among these, spiking neural networks (SNNs) have gained considerable attention due to their ability to capture the temporal dynamics of data and their energy efficiency, making them a promising solution for IoT and edge computing applications [13]. SNNs are a type of artificial neural network that employs spikes, or time-based events, to represent and process information, mimicking the way neurons communicate in biological systems [14].

The classification of malware through SNNs involves training the network on a dataset of malware samples and using the trained network to classify new, unseen malware samples. The process involves extracting features from the malware samples, such as opcode sequences or API calls, and using these features to create spike trains that can be inputted into the SNN. The SNN then processes these spike trains and produces an output that indicates the type of malware present in the sample [15-17].

Additionally, SNNs have the potential to operate with low power consumption, making them suitable for resource-constrained IoT devices [16]. However, challenges such as the need for large datasets, the difficulty of feature extraction, and the complexity of SNN training remain to be addressed.

In this paper, we investigate the landscape of IoT malware attacks. We provide an overview for the classification of the most common types of IoT malware. For this purpose, we utilized N-BaIoT dataset, This dataset was introduced in a study presented by Y. Median, M. Bohadana and et al. [18]. The dataset contains both benign and malicious traffic, making it well-suited for the development of a classification model to detect IoT malware attacks.

To tackle this classification problem, spiking neural networks (SNNs) are used as they have been shown to be effective in solving complex pattern recognition tasks [19-21]. SNNs are a type of artificial neural network that simulates the behaviour of biological neurons by using spikes, or brief changes in voltage, to communicate information [16, 22]. These networks are particularly useful for processing temporal information, which is a key feature of network traffic.

The use of SNNs in malware detection is still a relatively new field, One advantage of using SNNs is their ability to operate in an event-driven manner, meaning they only require computation when a new spike is received [23]. This allows them to process large amounts of data efficiently, making them well-suited for real-time classification tasks. Additionally, SNNs can be trained using unsupervised learning methods, allowing them to learn patterns in the data without the need for labelled training data, which can be difficult to obtain in the case of IoT malware attacks.

Overall, the use of the N-BaIoT dataset and SNNs offers a novel approach to detecting IoT malware attacks. By leveraging the temporal nature of network traffic and the efficient computation of SNNs, this approach shows promise in enhancing the efficiency of IoT malware detection, ultimately contributing to the security and reliability of IoT systems.

In this paper, we have trained a spiking neural network by using the snnTorch framework [24], for the classification of 11 different IoT attacks. The proposed algorithm has been trained on jypter notebook on 11th generation i7 laptop with NVIDIA GeForce RTX 3060 6GB GPU. The main contribution of this study are as:

- Proposed a spiking neural network algorithm for classifying IoT malware attacks. This is an important contribution to the field of cybersecurity, as it shows that SNNs can be a powerful tool for detecting and identifying malicious activity in the rapidly expanding world of IoT devices.
- Analysis of the N-BaIoT dataset, which is a valuable resource for researchers and practitioners in the field of cybersecurity. By training and evaluating SNN model on this dataset, we contribute to our understanding of the characteristics of IoT malware attacks and the types of data that are most informative for classifying them.

The rest of the paper is organized as follows: Section II presents a basic background and literature review for this study, Section III contains the information related to the dataset and data pre-processing, Section IV discussed about the algorithm and the results obtained and Section V is related to the discussion and conclusion.

II. LITERATURE REVIEW

Deep learning techniques have been widely used in classification tasks due to their ability to automatically learn features from large amounts of data [25]. Convolutional Neural Networks (CNNs) and Artificial Neural Networks (ANNs) are examples of deep learning techniques that have been successful in various domains, including image and speech recognition [26-28].

In recent years, there has been growing interest in the application of spiking neural networks (SNNs) for classification tasks. Compared to traditional neural networks, Spiking Neural Networks (SNNs) are a class of neural networks that model the dynamics of biological neurons, where information is represented in the form of spikes or action potentials [14]. Compared to traditional Artificial Neural Networks (ANNs) [29] and Convolutional Neural Networks (CNNs) [30], SNNs have the ability to process temporal information and exhibit event-driven behaviour. SNNs have been gaining attention in recent years due to their potential for efficient and low-power neuromorphic hardware implementations [31].

In the context of deep learning, SNNs are a relatively new area of research, and there have been significant advances in the development of training algorithms and architectures for SNNs. One of the key challenges in training SNNs is the non-differentiable nature of the spike function, which is used to model the firing behaviour of biological neurons [32]. To address this, several techniques have been proposed, including surrogate gradient methods and backpropagation through time [33].

The authors of this study [34], presented an iterative SNN model and a training algorithm specifically designed for spatial temporal spike pattern classification. To enable this, a coding method is proposed that converts continuous time series to discrete spikes, which allows for the representation of information using sparse spike patterns and significantly reduces computation overhead. The effectiveness of the algorithm and coding method were evaluated on multiple multivariate time series datasets, and the results showed superior performance compared to the standard 1-Nearest Neighbour classifiers. Additionally, the performance was also competitive with deep neural network-based approaches.

Similarly, the authors of [35], proposed a framework on Spiking One-Class Anomaly Detection Framework (SOCCADF) that is based on evolving Spiking Neural Network (eSNN) model for one class anomaly detection for cyber security. Their results shows that they achieved a total classification accuracy an average classification accuracy of 98% for three different datasets while using eSNN model.

The authors of [36], presented a classification problem and utilized a SNN model along with a CNN model for the ECG signals. The results shows that even CNN model achieved a higher accuracy, but the power consumption of CNN model was 0.67 Watt compared with their proposed SNN model which consumes 0.007 Watt, thus they concluded that SNN model is suitable for wearable and low powered devices.

Moreover, In [37], the authors proposed a technique for the intrusion detection from IoT networks, the named this technique as IDS-SNNDT which refers to intrusion detection system based on SNN and decision tree. Their study shows, they achieved a higher detection accuracy for the cyberattacks on their proposed IDS-SNNDT based on three different parameters i.e., accuracy, latency and energy usage. In another study i.e., [38], proposed an unsupervised method for anomaly detection in IoT data streams using the OeSNN algorithm. This approach uses a Gaussian receptive field (GRF) input encoding layer to simplify the SNN training process and identify abnormal modifications in the data stream. The main focus is on single time series analysis, allowing for more efficient and effective detection of anomalies.

Moreover, the authors of [39] proposed a technique for anomaly detection using the spiking neural networks (SNN). The method is comprised of three phases, the first of which initializes the weight values using the Rectified Orthogonal Projection Estimation (ROPE) method. In the second phase, the real input data is converted into spike values using the Gaussian Receptive Field (GRF) approach. Finally, the anomaly detection process occurs only when the neuron in SNN is spiked.

In this paper, we investigate the landscape of IoT malware attacks, analysing the latest threats, trends, and techniques used by attackers to exploit IoT vulnerabilities. We provide an overview of the most common types of IoT malware. For this purpose we utilized N-BaIoT dataset [18] that contains 11 different recorded IoT attacks data. Moreover, we developed a lightweight Spiking neural network with very less memory size with the aim of classifying the attacks into their respective class.

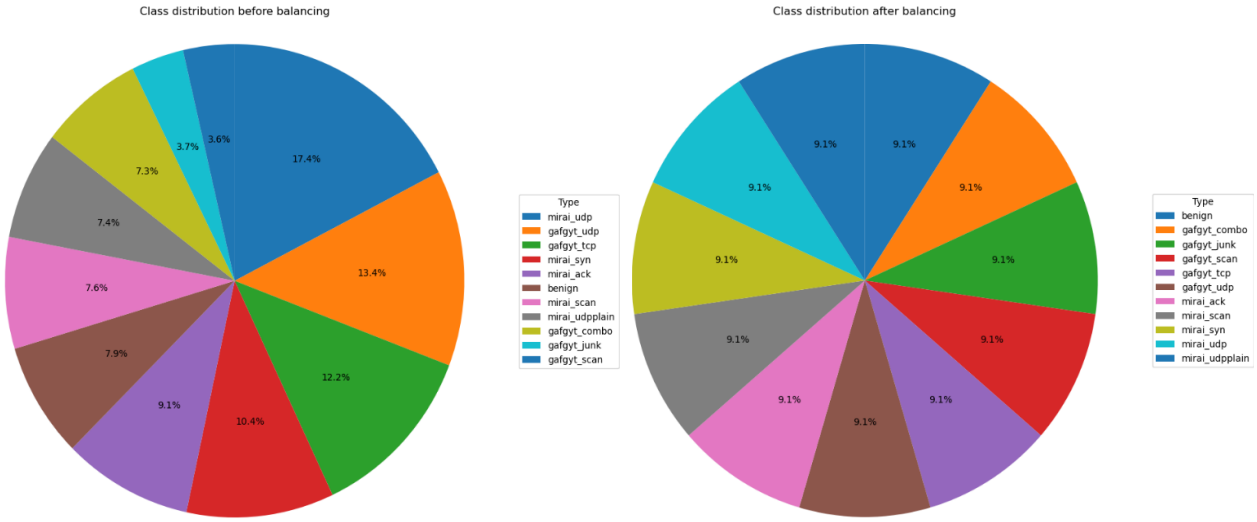

Fig. 1. Pie chart of the data distribution based on output labels.

III. Dataset

The N-BaIoT dataset, available on Kaggle [40] as well as in UCI archive [41] was firstly introduced by the authors of [18]. This dataset consists of network traffic traces for Internet of Things (IoT) devices. The traces were collected from a testbed of heterogeneous IoT devices, including cameras, routers, smart thermostats, and smart TVs. The dataset comprises approximately 7 million network flows, with 11 different classes of IoT attacks. The malicious flows correspond to several common types of attacks on IoT devices, including reconnaissance, denial of service, botnet, and injection attacks.

A. Data Preprocessing

We have accessed this dataset from Kaggle i.e., a publicly available website provides the publicly available dataset, the dataset is accessed via using the reference [40]. Dataset contains a total size of 8.14 gigabyte (GB) in csv files. It contains 93 different csv files, the files names were with their respective attack. A total of 11 different attacks categories with 115 input columns is being recorded in the csv files. The categories has been given in the dataset are as: mirai_udp, gafgyt_udp, gafgyt_tcp, mirai_syn, mirai_ack, benign, mirai_scan, mirai_udpplain, gafgyt_scan, , gafgyt_combo and gafgyt_junk

Furthermore, we merged available 93 csv files into 1 csv file, the dataset was quite unbalanced and very large for us to process, thus to use our memory efficiently and to avoid the unbalancing problem, we reduced the number of samples, so that we could utilized our RAM for the training of the model easily, the total number of samples with their respective class before and after has been given in Table I. And to show the total number of percentages of each class before and after balancing has also been given in a pie chart Fig 1. The balanced dataset was then normalized to the range of 0 to 1.

For normalization we have divided each column values by the maximum value presented in their respective column. Furthermore, we used Label encoder library to convert our string type output labels into numerical form. After then, we converted all our dataset into torch tensors values, so that it could be processed easily with the integration of snnTorch framework [24].

TABLE I. Dataset details with their respective class

Classes	Class Samples	
	Before Balancing	***After Balancing***
mirai_udp	1229999	5000
gafgyt_udp	946366	5000
gafgyt_tcp	859850	5000
mirai_syn	733299	5000
mirai_ack	643821	5000
benign	555932	5000
mirai_scan	537979	5000
mirai_udpplain	523304	5000
gafgyt_combo	515156	5000
gafgyt_junk	261789	5000
gafgyt_scan	255111	5000

IV. Experiment and Results

A. Data Splitting

After data pre-processing stage, we split the dataset into two branches i.e., train and test with a ratio of 90:10. Thus, the training dataset has been used for the training of our proposed model and the evaluation of model has been done on test dataset after the training of our model. For training, each class contains 4500 samples however, 500 samples from each class has been used in the test dataset.

B. Spiking neural network model

In order to build the spiking neural network model, we used snnTorch library [24]. The utilized dataset contains total 116 columns, i.e., 115 input columns and 1 output columns, this we also defined the input spikes to be equal as our inputs i.e., 115 and output spikes as our output labels i.e., 1. Furthermore, the threshold value for the spikes has been set to 0.5, so that whenever a spike with value greater than 0.5 has been countered it will generate and output spike, the beta coefficient value has been set to 0.6 so that

with the batch size of 32, furthermore we categorical cross entropy because of categorical dataset. And Adam optimizer has been used with the learning rate value of 0.001. The summary of the model is given in Table II.

Similar to the ANNs and CNNs where the model layer also contains activation functions [42], here in SNNs we have used leaky integrate-and-fire (LIF) neuron. Leaky integrate-and-fire (LIF) neuron model is commonly used in spiking neural network (SNN) models because it strikes a balance between biological plausibility and computational practicality. LIF neurons take the sum of weighted inputs and integrate them over time with a leakage, similar to an RC circuit. If the integrated value exceeds a threshold, the neuron emits a voltage spike. The simplicity of the LIF neuron allows for efficient computational implementation while still capturing the essence of neural dynamics. Additionally, the discrete event representation of the LIF neuron (based on timing or frequency of spikes) is well-suited for modeling neural communication in biological systems [43].

TABLE II. DATASET DETAILS WITH THEIR RESPECTIVE CLASS

Layers	*Input shape*	*Output shape*	*Parameters*
Linear-1	None, 115	-1,128	14,848
Leaky-2	-1,128	-1,11	0
Linear-3	-1,11	-1,11	1,419
Leaky-4	-1,11	-1,11	0
Total parameters			16,267
Trainable parameters			16,267

Model summary that has given above in Table 2 shows that a total of 2 layered SNN model has proposed first layer is Linear-1 and second layer is Linear-3, each layer has their corresponding LIF neurons i.e., Leaky-2 and Leaky-4. Proposed model contains total of 6,43,821 trainable parameters these are spikes which are trained on the input dataset. Moreover, our model contains a total size in MB is 0.19 with forward and backward pass size of 0.12 MB, which is very less and suitable for low memory size devices. The model has been trained over 150 epochs with 20-time steps. The visualization of the spikes with input spikes and the hidden layer i.e., the Leaky-2 and Leaky-4 their neurons itself act as a small model layer, has been shown in Fig 2.

C. Output spikes over time steps

Output spikes in SNN models is associated with specific time steps, reflecting the temporal nature of the model. This can be particularly useful for tasks where the timing of the output spikes is important, such as in event-based processing. Time stamps can also be used to align the output of the SNN model with the input data, facilitating analysis and interpretation of the results. Fig 3. Represents the output spikes after the training of the model over time steps with their corresponding labels.

D. Model evulation on the testing dataset

After the training of the spikes, we utilized the test dataset for the evaluation of the model, as this is a classification problem, so we utilized a classification report to perform the evaluation on the test dataset. The classification report with each class is given in Table III. The results shows that the model achieved an accuracy of 71.94%, although the accuracy is considered to be less but considering the fact that we have utilized a spiking neural network, one of the cons of the SNN is that it is difficult to achieve a higher accuracy because of its biological behaviour, although an important fact to be noted is that this model is very lightweight and does not possess very high power and memory size, which makes this model to be deployable for wearable devices and those devices which contains very less memory size.

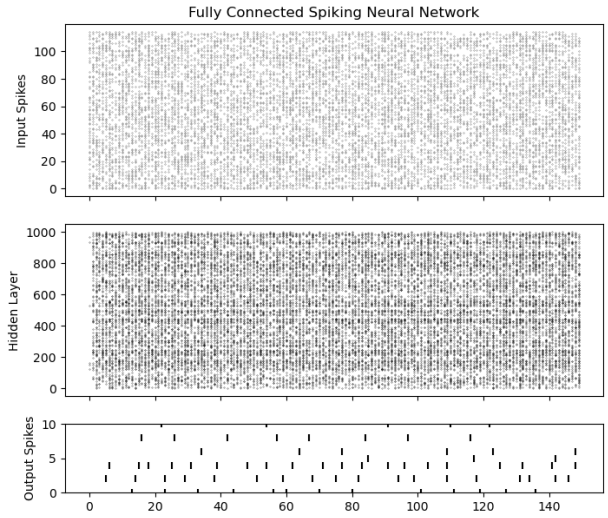

Fig. 2. Spiking neural networks input spikes over epochs.

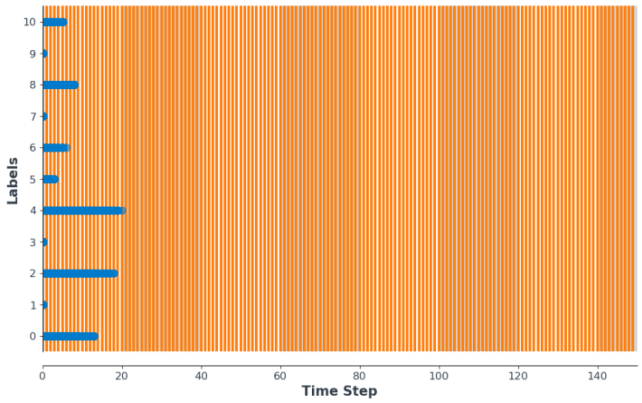

Fig. 3. Output spikes with the labels over time step.

TABLE III. CLASSIFICATION REPORT ON TEST DATASET

Classes	Precision %	Recall %	F1- score	Accuracy
mirai_udp	69.6	70.1	0.69	
gafgyt_udp	71.5	73.2	0.73	
gafgyt_tcp	78.3	54.2	0.64	
mirai_syn	43.5	48.3	0.45	
mirai_ack	65.1	55.3	0.59	
benign	74.1	78.9	0.76	71.94%
mirai_scan	56.2	60.3	0.58	
mirai_udpplain	77.3	45.3	0.57	
gafgyt_combo	78.9	66.2	0.71	
gafgyt_junk	75.2	70.5	0.72	
gafgyt_scan	71.2	74.9	0.73	

V. Discussion

The presented work aims to investigate the applicability of spiking neural networks (SNNs) for the task of IoT traffic classification. To achieve this goal, we used the publicly available N-BaIoT dataset, which contains a large number of traffic samples captured from a smart home IoT network. Before training the SNN model, we performed some preprocessing steps to balance the dataset by ensuring that each output label has an equal number of samples. Then, dataset is split into training and testing sets. We built a two-layered SNN model with two leaky integrate-and-fire (LIF) neurons in each layer. The model was trained and its performance was evaluated on the test dataset. The results showed that the proposed SNN model achieved an accuracy of 71% on the test dataset, demonstrating the potential of SNNs for IoT traffic classification. However, it is worth noting that the accuracy of the SNN model is lower than some state-of-the-art non-spiking deep learning models, which suggests that further research is needed to improve the performance of SNNs for this task. Nonetheless, this work provides a foundation for future research in applying SNNs to IoT traffic classification and encourages the exploration of more complex SNN architectures and training algorithms for this purpose. Future research in this area may involve the development of more efficient feature extraction methods, the exploration of novel SNN architectures, and the investigation of techniques for improving the interpretability of SNNs in the context of malware classification.

VI. Conclusion

In conclusion, this study demonstrated the effectiveness of spiking neural networks (SNNs) for the classification of the N-BaIoT dataset. The pre-processing steps of balancing the dataset and splitting it into training and testing sets were critical for obtaining reliable results. The SNN model, consisting of two layers and two LIF neurons, achieved an accuracy of 71% on the test dataset. While this accuracy may not be as high as that achieved by some deep neural network (DNN) models, the SNN's event-triggered feature and the sparse representation of information through spike patterns make it a promising alternative for resource-constrained environments. Future work could explore the use of more complex SNN architectures and explore the potential of combining SNNs with other machine learning techniques to further improve classification performance.

Acknowledgment

The authors would like to acknowledge the support of the Fundamental Research Grant Scheme (FRGS) FRGS/1/2021/ICT08/MMU/03/1, Ministry of Higher Education, Malaysia.

References

[1] E. F. Orumwense and K. Abo-Al-Ez, "Internet of Things for smart energy systems: A review on its applications, challenges and future trends," *AIMS Electronics Electrical Engineering,* vol. 7, no. 1, pp. 50-74, 2023.

[2] G. Akandere, E. Khajeh, and T. Paksoy, "The Internet of Things and Cyber-Physical Systems," in *Smart and Sustainable Operations and Supply Chain Management in Industry 4.0*: CRC Press, 2023, pp. 277-305.

[3] V. Demertzi, S. Demertzis, and K. Demertzis, "An Overview of Cyber Threats, Attacks and Countermeasures on the Primary Domains of Smart Cities," *Applied Sciences,* vol. 13, no. 2, p. 790, 2023.

[4] R. Montasari, "Internet of Things and Artificial Intelligence in National Security: Applications and Issues," in *Countering Cyberterrorism: The Confluence of Artificial Intelligence, Cyber Forensics and Digital Policing in US and UK National Cybersecurity*: Springer, 2023, pp. 27-56.

[5] Z. Muhammad, Z. Anwar, B. Saleem, and J. Shahid, "Emerging Cybersecurity and Privacy Threats to Electric Vehicles and Their Impact on Human and Environmental Sustainability," *Energies,* vol. 16, no. 3, p. 1113, 2023.

[6] S. Jimo, T. Abdullah, and A. Jamal, "IoE Security Risk Analysis in a Modern Hospital Ecosystem," in *Cybersecurity in the Age of Smart Societies: Proceedings of the 14th International Conference on Global Security, Safety and Sustainability, London, September 2022*, 2023, pp. 451-467: Springer.

[7] V. Vatsyayan, A. Chakraborty, G. Rajarajan, and A. L. Fernandez, "A Detailed Investigation of Popular Attacks on Cyber Physical Systems," in *Cyber Security Applications for Industry 4.0*: Chapman and Hall/CRC, 2023, pp. 1-42.

[8] T. N. Nguyen, "A review of cyber crime," *Journal of Social Review Development,* vol. 2, no. 1, pp. 01-03, 2023.

[9] R. Chaganti, V. Ravi, and T. D. Pham, "A multi-view feature fusion approach for effective malware classification using Deep Learning," *Journal of Information Security Applications,* vol. 72, p. 103402, 2023.

[10] O. Habibi, M. Chemmakha, and M. Lazaar, "Performance Evaluation of CNN and Pre-trained Models for Malware Classification," *Arabian Journal for Science Engineering,* pp. 1-15, 2023.

[11] A. R. Zarzoor, N. A. S. Al-Jamali, and D. A. A. Qader, "Intrusion detection method for internet of things based on the spiking neural network and decision tree method," *International Journal of Electrical Computer Engineering,* vol. 13, no. 2, p. 2278, 2023.

[12] X. Lin, C. Dong, X. Liu, and D. Cheng, "Spiking Neural Networks Subject to Adversarial Attacks in Spiking Domain," in *International Conference on Machine Learning for Cyber Security*, 2023, pp. 457-471: Springer.

[13] S. Liu *et al.*, "An Area-and Energy-Efficient Spiking Neural Network with Spike-Time-Dependent Plasticity Realized with SRAM Processing-in-memory Macro and On-chip Unsupervised Learning," *IEEE Transactions on Biomedical Circuits Systems,* 2023.

[14] S. Ghosh-Dastidar and H. Adeli, "Spiking neural networks," *International journal of neural systems,* vol. 19, no. 04, pp. 295-308, 2009.

[15] M. Pagkalos, S. Chavlis, and P. Poirazi, "Introducing the Dendrify framework for incorporating dendrites to spiking neural networks," *Nature Communications,* vol. 14, no. 1, p. 131, 2023.

[16] Z. Yi, J. Lian, Q. Liu, H. Zhu, D. Liang, and J. Liu, "Learning Rules in Spiking Neural Networks: A Survey," *Neurocomputing,* 2023.

[17] Q. Zhan, G. Liu, X. Xie, M. Zhang, and G. Sun, "Bio-inspired Active Learning method in spiking neural network," *Knowledge-Based Systems,* vol. 261, p. 110193, 2023.

[18] Y. Meidan *et al.*, "N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," *IEEE Pervasive Computing,* vol. 17, no. 3, pp. 12-22, 2018.

[19] Z. Lin, D. Ma, J. Meng, and L. Chen, "Relative ordering learning in spiking neural network for pattern recognition," *Neurocomputing,* vol. 275, pp. 94-106, 2018.

[20] S. Dora, S. Sundaram, and N. Sundararajan, "A two stage learning algorithm for a growing-pruning spiking neural network for pattern classification problems," in *2015 international joint conference on neural networks (IJCNN)*, 2015, pp. 1-7: IEEE.

[21] X. Wu *et al.*, "Improving NeuCube Spiking Neural Network for EEG-based Pattern Recognition Using Transfer Learning," *Neurocomputing,* 2023.

[22] H.-M. Huang, Z. Wang, T. Wang, Y. Xiao, and X. Guo, "Artificial neural networks based on memristive devices: From device to system," *Advanced Intelligent Systems,* vol. 2, no. 12, p. 2000149, 2020.

[23] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient

event-driven networks for image classification," *Frontiers in neuroscience,* vol. 11, p. 682, 2017.
[24] *snnTorch*. Available: https://snntorch.readthedocs.io/en/latest/
[25] U. Muhammad and F. Yee-Loo, "Industrial Safety Helmet Detection Using Single Shot Detectors Models and Transfer Learning," in *Proceedings of the Multimedia University Engineering Conference (MECON 2022)*, 2022, pp. 390-400: Atlantis Press.
[26] H. Ahmed, M. Umair, A. Iftikhar, and K. Sultana, "Covid-19 Variants Detection & Classification Using Self Proposed Two stage MNN-2: Robust Comparison with Yolo V5 & Faster R-CNN," in *2022 IEEE International Conference on Blockchain, Smart Healthcare and Emerging Technologies (SmartBlock4Health)*, 2022, pp. 1-7.
[27] M. Umair *et al.*, "Detection of COVID-19 Using Transfer Learning and Grad-CAM Visualization on Indigenously Collected X-ray Dataset," vol. 21, no. 17, p. 5813, 2021.
[28] S. Minaee, A. Abdolrashidi, H. Su, M. Bennamoun, and D. Zhang, "Biometrics recognition using deep learning: a survey," *Artificial Intelligence Review,* 2023.
[29] P. Dey, "Artificial Neural Network in Pathology: Basic Principles and Applications," in *Basic and Advanced Laboratory Techniques in Histopathology and Cytology*: Springer, 2023, pp. 267-275.
[30] X. Lei, H. Pan, and X. Huang, "A dilated CNN model for image classification," *IEEE Access,* vol. 7, pp. 124087-124095, 2019.
[31] D.-A. Nguyen, X.-T. Tran, and F. Iacopi, "A review of algorithms and hardware implementations for spiking neural networks," *Journal of Low Power Electronics Applications,* vol. 11, no. 2, p. 23, 2021.
[32] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, "Deep learning in spiking neural networks," *Neural networks,* vol. 111, pp. 47-63, 2019.
[33] N. Perez-Nieves and D. Goodman, "Sparse spiking gradient descent," *Advances in Neural Information Processing Systems,* vol. 34, pp. 11795-11808, 2021.
[34] H. Fang, A. Shrestha, and Q. Qiu, "Multivariate Time Series Classification Using Spiking Neural Networks," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1-7.
[35] K. Demertzis, L. Iliadis, and S. Spartalis, "A Spiking One-Class Anomaly Detection Framework for Cyber-Security on Industrial Control Systems," in *Engineering Applications of Neural Networks*, Cham, 2017, pp. 122-134: Springer International Publishing.
[36] Z. Yan, J. Zhou, and W.-F. Wong, "Energy efficient ECG classification with spiking neural network," *Biomedical Signal Processing and Control,* vol. 63, p. 102170, 2021.
[37] Ahmed R. Zarzoor, Nadia Adnan Shiltagh Al-Jamali, and D. A. A. Qader, "Intrusion detection method for internet of things based on the spiking neural network and decision tree method," *International Journal of Electrical and Computer Engineering,* vol. 13(2), p. 2278~2288, 2023.
[38] P. S. Maciąg, M. Kryszkiewicz, R. Bembenik, J. L. Lobo, and J. Del Ser, "Unsupervised Anomaly Detection in Stream Data with Online Evolving Spiking Neural Networks," *Neural Networks,* vol. 139, pp. 118-139, 2021.
[39] B. Yusob, Z. Mustaffa, and J. Sulaiman, "Anomaly Detection in Time Series Data Using Spiking Neural Network," *Advanced Science Letters,* vol. 24, no. 10, pp. 7572-7576(5), 2018.
[40] *N-BaIoT Dataset to Detect IoT Botnet Attacks*. Available: https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset
[41] *detection_of_IoT_botnet_attacks_N_BaIoT Data Set*. Available: https://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_BaIoT
[42] R. Parhi and R. D. Nowak, "The Role of Neural Network Activation Functions," *IEEE Signal Processing Letters,* vol. 27, pp. 1779-1783, 2020.
[43] Z. Wu, H. Zhang, Y. Lin, G. Li, M. Wang, and Y. Tang, "LIAF-Net: Leaky Integrate and Analog Fire Network for Lightweight and Efficient Spatiotemporal Information Processing," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 33, no. 11, pp. 6249-6262, 2022.