Are Neuromorphic Architectures Inherently Privacy-preserving? An Exploratory Study

Ayana Moshruba
George Mason University
amoshrub@gmu.edu

Ihsen Alouani
Centre for Secure Information Technologies (CSIT)
Queen's University Belfast
i.alouani@qub.ac.uk

Maryam Parsa
George Mason University
mparsa@gmu.edu

Abstract

While machine learning (ML) models are becoming mainstream, including in critical application domains, concerns have been raised about the increasing risk of sensitive data leakage. Various privacy attacks, such as membership inference attacks (MIAs), have been developed to extract data from trained ML models, posing significant risks to data confidentiality. While the predominant work in the ML community considers traditional Artificial Neural Networks (ANNs) as the default neural model, neuromorphic architectures, such as Spiking Neural Networks (SNNs) have recently emerged as an attractive alternative mainly due to their significantly low power consumption. These architectures process information through discrete events, i.e., spikes, to mimic the functioning of biological neurons in the brain. While the privacy issues have been extensively investigated in the context of traditional ANNs, they remain largely unexplored in neuromorphic architectures, and little work has been dedicated to investigate their privacy preserving properties. In this paper, we investigate the question whether SNNs have inherent privacy preserving advantage. Specifically, we investigate SNNs' privacy properties through the lens of MIAs across diverse datasets, in comparison with ANNs. We explore the impact of different learning algorithms (surrogate gradient and evolutionary learning), programming frameworks (snnTorch, TENNLab, and LAVA), and various parameters on the resilience of SNNs against MIA. Our experiments reveal that SNNs demonstrate consistently superior privacy preservation compared to ANNs, with evolutionary algorithms further enhancing their resilience. For example, on the CIFAR-10 dataset, SNNs achieve an AUC as low as 0.59 compared to 0.82 for ANNs, and on CIFAR-100, SNNs maintain a low AUC of 0.58, whereas ANNs reach 0.88. Furthermore, we investigate the privacy-utility trade off through Differentially Private Stochastic Gradient Descent (DPSGD) observing that SNNs incur a notably lower accuracy drop than ANNs under equivalent privacy constraints.

Keywords

Membership Inference Attack, Spiking Neural Network, Differentially Private Stochastic Gradient Descent, Neuromorphic Computing

1 Introduction

As ML systems become more sophisticated and widespread, individuals are increasingly relying on these systems, entrusting them with personal and professional data. Consequently, the risk of sensitive information exposure [65] is growing significantly in multiple sectors [6] including healthcare [1], finance [79], national security [78], education [15] and consumer services [39]. It is particularly alarming in fields such as healthcare, where the confidentiality of patient data is extremely sensitive, as a breach could result in severe personal and financial implications, which can affect patient care and institutional credibility [30, 44, 80]. In finance, the integrity of financial transactions and records is fundamental for maintaining market stability and preventing fraud [88], while in national security, safeguarding classified information is essential to protect national interests and prevent threats to public safety [34].

This has led to the development of various privacy attacks targeting ML models to extract sensitive information, including Model Inversion Attacks [17], Attribute Inference Attacks [20], Model Stealing Attacks [31], and Membership Inference Attacks (MIAs) [72]. In MIAs, an adversary seeks to ascertain if a specific data point was part of the dataset used to train the model. This intrusion risks exposing classified information about individuals in the training dataset, potentially compromising personal data confidentiality [74].

Designed to replicate the dynamic behavior of biological neurons [18], SNNs process information through discrete, temporally encoded spikes [63], enabling them to handle time-sensitive data efficiently [26]. Their suitability for edge computing [73] and resource-constrained environments further enhances their value, as SNNs effectively process real-world spatiotemporal patterns [82]. This capability positions SNNs as a promising alternative to traditional neural networks for applications requiring dynamic, real-time data processing. While the security of SNNs has been investigated in the literature [12, 81], relatively little attention has been given to their privacy-preserving capabilities.

This work addresses the privacy concerns associated with SNNs through a structured investigation of three core areas: (i) the resilience of ANN and SNN models to Membership Inference Attacks (MIAs), (ii) the factors influencing the privacy-preserving properties of SNNs, and (iii) the privacy-utility trade-off in ANN and SNN models using the DPSGD algorithm.

We consider that the potential resilience of SNNs against MIAs is based on two key aspects. Firstly, the non-differentiable and discontinuous nature of SNNs may weaken the correlation between the model and individual data points, making it more challenging for an attacker to identify the membership of a particular data point in the

Proceedings on Privacy Enhancing Technologies 2025(2), 243–257

https://doi.org/10.56553/popets-2025-0060

training set [47]. Secondly, the unique encoding mechanisms employed by SNNs introduce an additional layer of stochasticity [53] and variability to the data representation. This added complexity can make it more difficult for an attacker to deduce the unique characteristics of individual data points, thereby making them more indistinguishable.

Investigating the resilience of SNNs against MIAs, our experimental results consistently demonstrate that SNNs exhibit higher resilience to MIAs across the datasets including MNIST, F-MNIST, Iris, Breast Cancer, CIFAR-10, and CIFAR-100. This is evidenced by the lower Area Under the Curve(AUC) values for the Receiver Operating Characteristic(ROC) curves in SNNs compared to their ANN counterparts. Furthermore, our exploration domain encompasses various learning algorithms (surrogate gradient-based and evolutionary learning), programming frameworks (snnTorch, TENNLab, and LAVA), and a wide range of parameters within them, providing a comprehensive analysis of the factors influencing the inherent privacy preserving properties of SNNs. This in-depth exploration indicated that evolutionary learning algorithms shown to boost this resilience more effectively compared to the gradient based methods.

In order to enhance data privacy and explore the compromises between privacy and utility, we study the implementation of the DPSGD algorithm as a privacy defense mechanism [85]. This introduces controlled noise into the training process, making it harder for attackers to infer the presence of specific data points. However, improved privacy often comes at the cost of reduced model performance, known as the privacy-utility trade-off [77]. Through the experiments, we observe that SNNs exhibit a notably lower performance drop compared to ANNs for the same level of privacy guarantee. This finding further reinforces our hypothesis regarding the inherent privacy-preserving properties of SNNs.

This paper offers the following notable contributions in the field of data privacy, particularly in the context of SNNs:

- SNNs exhibit higher resilience against MIAs compared to ANNs, with lower AUC scores on CIFAR-10 (SNN: 0.59 vs. ANN: 0.82) and CIFAR-100 (SNN: 0.58 vs. ANN: 0.88), highlighting their potential as a more secure alternative in privacy sensitive applications.
- Evolutionary learning algorithms outperform gradient based methods in MIA resilience, maintaining a consistent AUC of 0.50 across all parameters for Iris and Breast Cancer datasets, compared to 0.57 and 0.55 AUC scores for gradient-based algorithms, respectively.
- Privacy-utility trade off analysis revealing that SNNs incur a lower accuracy drop compared to ANNs when applying DPSGD: For F-MNIST, with privacy guarantees ranging from 0.22 to 2.00, the average accuracy drop is 12.87% for SNNs comparatively lower than the 19.55% drop observed in ANNs.

We emphasize that while this investigation highlights SNNs' enhanced privacy characteristics, our findings specifically address privacy preservation applications. The architectural properties of SNNs that enable efficient hardware implementation and reduce computational overhead make them particularly appealing for resource constrained environments. However, these findings do not suggest an overall superiority of SNNs over ANNs across different application domains. Rather, we base this work on the intuition that SNNs' unique information processing mechanisms may offer specific advantages in privacy preservation, which warrants systematic investigation in this particular domain.

2 Background

2.1 Neuromorphic Architecture

Neuromorphic architectures [45] are designed to mimic the neural structures and functionalities of the biological brain, offering an alternative to the traditional von Neumann computing systems [4]. These architectures integrate memory and processing units, enabling massive parallel processing [49] and event driven computation [28]. By modeling neurons and synapses that communicate via discrete spikes or events [43], neuromorphic systems can process data asynchronously, reducing power consumption and latency. Hardware implementations like IBM's TrueNorth [2] and Intel's Loihi [9] demonstrate the potential for scalable, energy efficient architectures capable of performing complex tasks with a structure resembling biological neural networks.

SNNs constitute the foundational architecture of neuromorphic systems, operating through a mechanism where neurons accumulate membrane potential over time and generate discrete spike events upon reaching a threshold potential. This temporal encoding paradigm fundamentally differentiates SNNs from traditional neural networks, as information is encoded in both spike timing and frequency. The computational models of SNNs span from elementary integrate-and-fire neurons [5] to sophisticated biologically inspired implementations such as the Hodgkin Huxley model, which incorporates detailed ionic conductance dynamics. The event driven nature of spike-based computation intrinsically aligned with neuromorphic hardware architectures, enabling efficient asynchronous processing. This integration of temporal dynamics and synaptic plasticity in SNNs facilitates adaptive learning mechanisms particularly suited for applications demanding real time processing and minimal latency [67].

The architectural distinctiveness of SNNs, characterized by spike based information encoding and asynchronous processing, introduces computational complexities that potentially influence their susceptibility to privacy attacks. While traditional ANNs demonstrate vulnerability to Membership Inference Attacks through their deterministic output patterns and continuous gradients, SNNs operate via discrete, non-differentiable spike events [90]. The temporal dynamics of spike generation and the stochastic nature of neuronal activation contribute to output variability. The membrane potential accumulation in SNN neurons, coupled with probabilistic firing thresholds, generates diverse spike patterns for comparable inputs. This intrinsic variability and the discontinuous activation characteristics inherent to SNNs potentially obscure input-output relationships, thereby complicating the pattern recognition essential for successful MIAs. Although these properties, particularly when combined with differential privacy techniques like DPSGD, suggest enhanced privacy preservation capabilities, such assumptions require rigorous validation. This investigation examines whether the spike based computational paradigm of SNNs exhibits superior resilience to performance degradation under privacy constraints compared to traditional ANNs.

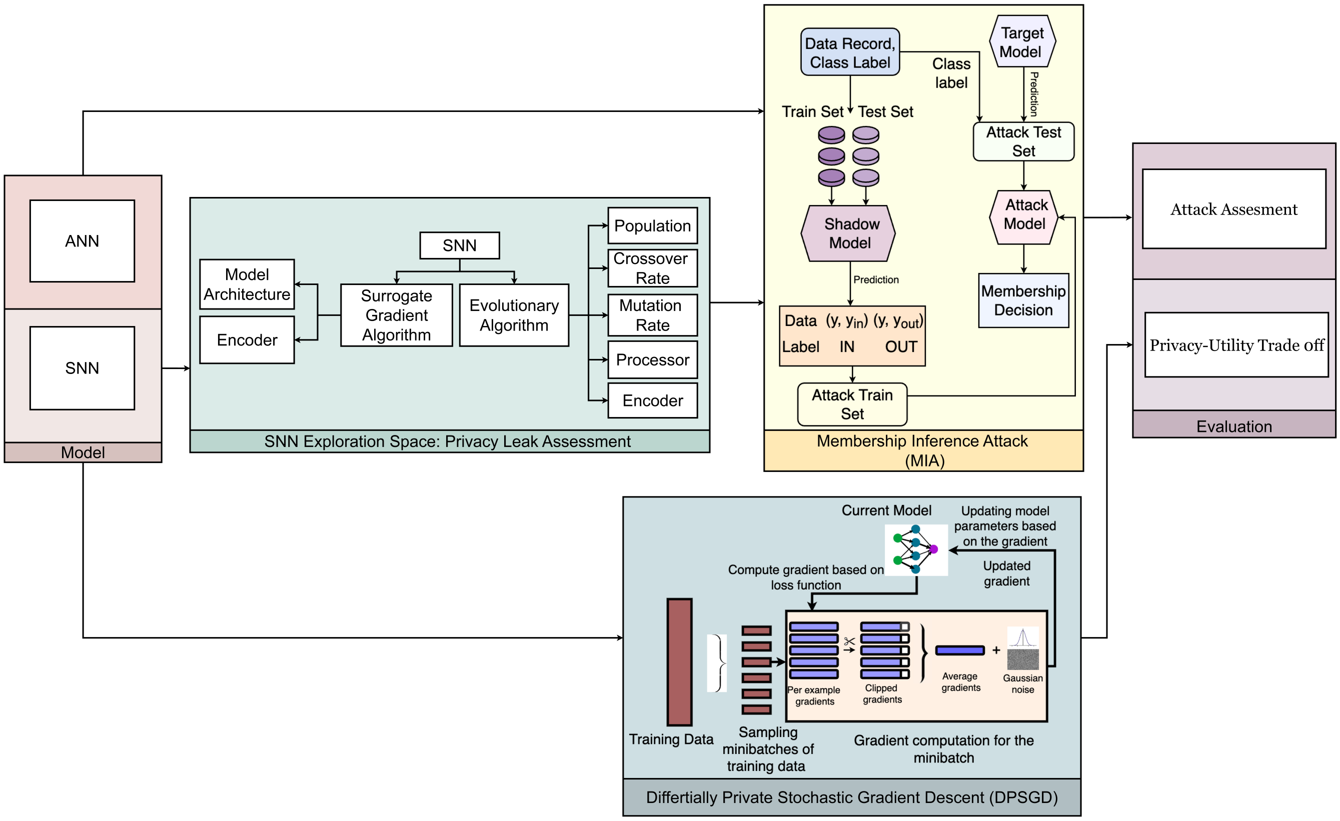


Figure 1: Architectural overview of the experimental framework. The methodology has 3 components: (i) comparative privacy assessment between ANNs and SNNs through MIA (yellow), (ii) exploration of SNN specific privacy characteristics considering surrogate gradient and evolutionary algorithms (green), and (iii) Comparative evaluation of privacy-utility trade offs through DPSGD (blue)

2.2 Membership Inference Attack (MIA)

constitute a class of privacy vulnerabilities that enable adversaries to determine whether specific data points were used in a model's training dataset. These attacks exploit differential behavioral patterns in model responses between training and non-training samples [60]. Neural networks typically demonstrate heightened prediction confidence and distinctive error distributions for previously encountered training samples compared to unseen data [51]. Through systematic analysis of these response characteristics, adversaries can extract sensitive training set information, potentially compromising data privacy [10] in domains handling confidential data. The implementation of MIAs encompasses both the development of specialized attack models and the application of statistical inference methods to differentiate between model responses to training and non-training samples. The efficacy of these attacks correlates strongly with the degree of model overfitting and the distinctiveness of individual sample responses. Beyond immediate privacy implications, MIAs serve as indicators of model generalization deficiencies [19], highlighting vulnerabilities in the neural network architecture.

2.3 Differentially Private Stochastic Gradient Descent (DPSGD)

Differential Privacy (DP) [11] establishes a mathematical framework that quantifies and bounds the privacy risk when operating on sensitive data. The framework ensures that statistical queries on a dataset remain nearly unchanged regardless of the inclusion or exclusion of any individual record, providing formal privacy guarantees. This privacy guarantee is parameterized by ϵ (privacy budget) and δ (failure probability), formalized through the following inequality:

$$P(M(D) \in S) \leq e^{\epsilon} P(M(D') \in S) + \delta$$

where D and D' are datasets differing by one element, M is a randomized algorithm, and S is a subset of possible outputs. This inequality limits the probability of M producing an output within S when applied to D is limited by the exponential of ϵ. This probability is then multiplied by the probability of M producing the same output from D'. Additionally, it is adjusted by a small term δ. In this context, ϵ controls the sensitivity of the output to variations in the input, where smaller values of ϵ indicate enhanced privacy by limiting the permissible changes in output probabilities. δ, ideally a small value near zero, accounts for the rare cases where this privacy guarantee might fail, thus ensuring that changes to any single data point in D minimally affect the output, reinforcing data privacy across the dataset.

DPSGD implements these theoretical guarantees in the context of neural network training by incorporating calibrated noise into the gradient computation process. In DPSGD, the key step involves perturbing the gradients computed during each training iteration with noise that is calibrated to the sensitivity of the function being optimized. This sensitivity measures how much the output of a function can change in response to changes in its input, which in the context of machine learning, translates to how much a single training example can influence the overall model parameters. The noise added is typically drawn from a Gaussian distribution [32], scaled according to ϵ and the desired level of privacy guarantee, δ. The function of DPSGD can be expressed mathematically as:

$$\theta_{t+1} = \theta_t - \eta \left(g_t + \mathcal{N}(0, \sigma^2 \mathbf{I})\right)$$

where θ_t represents the model parameters at iteration t, η is the learning rate, g_t is the gradient of the loss function with respect to θ_t, clipped to a norm bound C, and $\mathcal{N}(0, \sigma^2 \mathbf{I})$ denotes the Gaussian noise added to the gradient, with σ being determined by C, ϵ, and δ.

3 Methodology

This investigation examines the comparative privacy resilience of SNNs and ANNs through a systematic experimental framework comprising three distinct phases: (1) assessment of privacy vulnerabilities through MIA in both architectures, (2) analysis of SNN specific privacy characteristics across diverse algorithmic implementations, and (3) evaluation of privacy-utility trade offs through the implementation of DPSGD. The experimental methodology and interrelationships between architectural components are illustrated in Figure 1.

3.1 Comparison of Privacy Vulnerability between ANNs and SNNs

The comparative privacy risk assessment utilizes MIAs across equivalent ANN and SNN architectures, implementing convolutional baseline models, ResNet18, and VGG16 configurations. The experimental framework employs shadow models trained on labeled datasets (Figure 1, top-right) to emulate the target model characteristics, while attack models are developed to ascertain training set membership of individual data points. Privacy vulnerability is quantified through AUC metrics for both architectures, enabling systematic comparison of their susceptibility to MIAs (Figure 1, yellow block).

3.2 Algorithmic Exploration within the SNN Architecture

The second phase examines SNN privacy resilience across diverse algorithmic implementations through three distinct frameworks: Surrogate Gradient Algorithm, Evolutionary Algorithm, and Intel's LAVA framework. The Surrogate Gradient implementation evaluates the baseline SNN model using three distinct encoding mechanisms from the snnTorch [13] library: Delta, Latency, and Delta Modulation (Figure 1, left section, "SNN Exploration Space"). This analysis examines how encoding methods affect privacy vulnerability by correlating spike generation mechanisms with MIA efficacy.

Table 1: Model Architectures and Configurations

Network	Variant	Structure	Parameters*
Baseline (ConvNet)	ANN	• 2 Conv (32,64 filters) • 2 MaxPool • 2 FC (1000, num_classes) • ReLU	~2.3M
	SNN	• 2 Conv (32,64 filters) • 2 MaxPool • LIF neurons • Temporal processing	~2.3M
ResNet18	ANN	• 4 BasicBlocks (64→512) • GroupNorm • Skip connections • Adaptive pool	~11.7M
	SNN	• 4 BasicBlocks (64→512) • GroupNorm+BNTT • Spike residuals • Membrane threshold	~11.7M
VGG16	ANN	• 13 Conv (64→512) • 5 MaxPool • 3 FC (4096, classes) • GroupNorm, ReLU	~138M
	SNN	• 13 Conv (64→512) • 5 AvgPool • Binary spikes • Membrane reset	~138M

*Params vary with input channels (1/3) and classes (10/100)

The Evolutionary Algorithm implementation, utilizing the TennLab framework, modulates architectural parameters including population size, crossover rate, mutation rate, processor configurations, and encoder settings. This parameter space exploration evaluates how architectural variations influence SNN susceptibility to MIAs (Figure 1, green block).

3.3 Privacy-Utility Trade-off Analysis

The final experimental phase implements DPSGD across both ANN and SNN architectures ((Figure 1, bottom) to quantify the privacy-utility trade-offs. The DPSGD implementation incorporates Gaussian noise during gradient computation, establishing differential privacy guarantees while measuring the corresponding impact on model performance. The analysis spans convolutional, ResNet18, and VGG16 architectures, evaluating both model accuracy and resilience through attack AUC metrics under equivalent privacy constraints (Figure 1, blue block).

4 Experimental Framework and Setup

4.1 Dataset and Model Architecture

The proposed method is evaluated on both image and tabular datasets. For image classification tasks, MNIST [38] and Fashion-MNIST [84], comprising 28×28 grayscale images across 10 classes, are utilized. CIFAR-10 [62] and CIFAR-100 [71], containing 32×32 RGB images with 10 and 100 classes, respectively, are also included.

Two tabular datasets, Iris [54], consisting of 4 features and 3 classes, and Breast Cancer [92], comprising 30 features and 2 classes, are used.

The experimental evaluation encompasses both image and tabular datasets. The image classification tasks utilize MNIST [38] and Fashion-MNIST [84], comprising 28×28 grayscale images across 10 classes, and CIFAR-10 [62] and CIFAR-100 [71], containing 32×32 RGB images with 10 and 100 classes, respectively. The tabular datasets include Iris [54] (4 features, 3 classes) and Breast Cancer [92] (30 features, 2 classes).

The architectural implementations, as detailed in Table 1, are composed of three model configurations that are adapted for both ANN and SNN frameworks. The baseline architecture is implemented with dual convolutional layers for image processing and fully connected layers for tabular data analysis. ResNet18 is constructed with four basic blocks that incorporate group normalization and residual connections, while VGG16 is designed with 13 convolutional layers, progressively expanding from 64 to 512 channels, and culminating in three fully connected layers.

In the SNN variants, ReLU activations are replaced with leaky integrate-and-fire (LIF) neurons, and temporal processing mechanisms are incorporated. The implementation framework employs PyTorch [33] for ANN architectures, while snnTorch is used for the baseline model and SpikingJelly [14] is utilized for ResNet and VGG16 configurations.

4.2 MIA

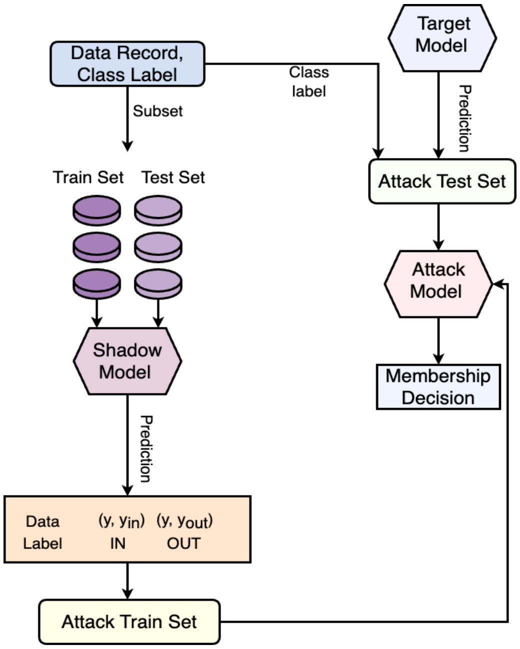


Figure 2: Membership Inference Attack(MIA) Framework

The MIA framework employs a dual model architecture as illustrated in Figure 2. The experimental setup includes a 'target' model and a 'shadow' model, where we evaluate three different architectures: baseline model, ResNet18, and VGG16. The shadow model simulates the target model's functionality by training on 80% of the corresponding dataset while maintaining architectural parity. For the attack phase, we employ a Support Vector Machine (SVM) as our attack model, chosen for its effectiveness as a binary classifier to determine whether a data point is a member or non-member of the training set. We evaluate the models by querying with their respective training and test sets, labeling the responses as 'IN' if the data was in the model's training set and 'OUT' otherwise. To reduce classifier bias towards more frequent classes, we implement undersampling of the predominant class. The SVM attack model, trained on these labeled predictions from the shadow model, provides a quantified measure of the model's susceptibility to MIA through its ability to accurately classify membership status.

4.3 DPSGD

The DPSGD implementation utilizes the Opacus Privacy Engine [87], as illustrated in Figure 3. The DPSGD algorithm 1 begins by selecting a minibatch L_t uniformly sampled from the set of indices $\{1, \ldots, N\}$ with size L. For each data point x_i in the minibatch, the gradient of the loss function $\mathcal{L}(\theta_t, x_i)$ with respect to the parameters θ_t is computed as $g_t(x_i)$. These gradients are then clipped using an $L2$ norm operation, $\hat{g}_t(x_i) = g_t(x_i)/\max(1, \|g_t(x_i)\|_2/C)$. Gaussian noise $\mathcal{N}(0, \sigma^2 C^2 I)$ is added to the average of the clipped gradients, with σ being the noise scale, to ensure differential privacy. The noisy gradient $\tilde{g}_t$ is used to update the model parameters: $\theta_{t+1} = \theta_t - \eta_t \tilde{g}_t$.

To investigate the privacy-utility tradeoff, we define the same target privacy budget (ε) of range 0.1-2.00, while fixing the privacy compromise $\delta = 1e - 5$ and evaluate the utility of the SNN architectures comparatively with their ANN counterparts.

Algorithm 1 DPSGD Algorithm

```
Input: Dataset $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(\theta, x_i)$
Parameters: Learning rate $\eta$, noise scale $\sigma$, batch size $L$, gradient norm bound $C$
Initialize $\theta_0$ randomly
for $t = 1$ to $T$ do
    Take a random sample $L_t$ with sampling probability $\frac{L}{N}$
    for each $i \in L_t$ do
        Compute gradient $g_t(x_i) \leftarrow \nabla_\theta \mathcal{L}(\theta_t, x_i)$
        Clip gradient $g_t(x_i) \leftarrow \frac{g_t(x_i)}{\max(1, \frac{\|g_t(x_i)\|_2}{C})}$
    end for
    Add noise $\tilde{g}_t \leftarrow \frac{1}{L}\left(\sum_{i \in L_t} g_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 I)\right)$
    Update parameters $\theta_{t+1} \leftarrow \theta_t - \eta \tilde{g}_t$
end for
Output: Return $\theta_T$ and compute privacy cost $(\epsilon, \delta)$
```

4.4 SNN Exploration Space

In the SNN exploration space as shown in Figure 1, we investigate two major learning algorithms: surrogate gradient based learning [50] and evolutionary learning [69].

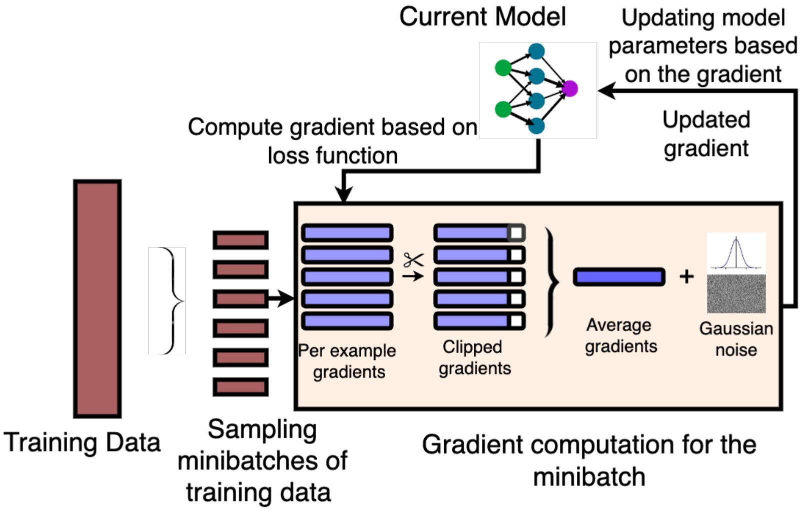

Current Model
Updating model parameters based on the gradient
Updated gradient
Compute gradient based on loss function
Per example gradients
Clipped gradients
Average gradients
+
Gaussian noise
Training Data
Sampling minibatches of training data
Gradient computation for the minibatch

Figure 3: Differential Privacy via Stochastic Gradient Descent (DPSGD) Algorithm

4.4.1 *Surrogate Gradient based Learning:*

Surrogate gradient methodologies facilitate SNN training through differentiable approximations of the non differentiable spiking activation function, enabling gradient based optimization techniques. These algorithms substitute the discrete spike function gradient with continuous, differentiable surrogates, permitting backpropagation based training while preserving the essential characteristics of spike-based computation. The experimental implementation employs two distinct approaches: the Arc tangent surrogate gradient algorithm [61] implemented via snnTorch [75] framework, and the biologically-inspired Spike Timing Dependent Plasticity (STDP) [7] learning mechanism through Intel's LAVA framework [29].

SnnTorch Framework:

The snnTorch framework, built upon PyTorch, provides specialized implementations for Spiking Neural Networks. Our experimental framework implements the arc tangent learning algorithm, which provides differentiable approximations of the spiking activation function. This implementation is evaluated across MNIST, F-MNIST, Iris, and Breast Cancer datasets to assess architectural resilience against membership inference attacks. In SNN architectures, the encoding mechanism determines the transformation of input signals into temporal spike patterns, fundamentally influencing the network's computational characteristics and performance metrics. Our investigation examines the relationship between encoding methodologies and privacy preservation characteristics across three distinct encoding schemes [13]:

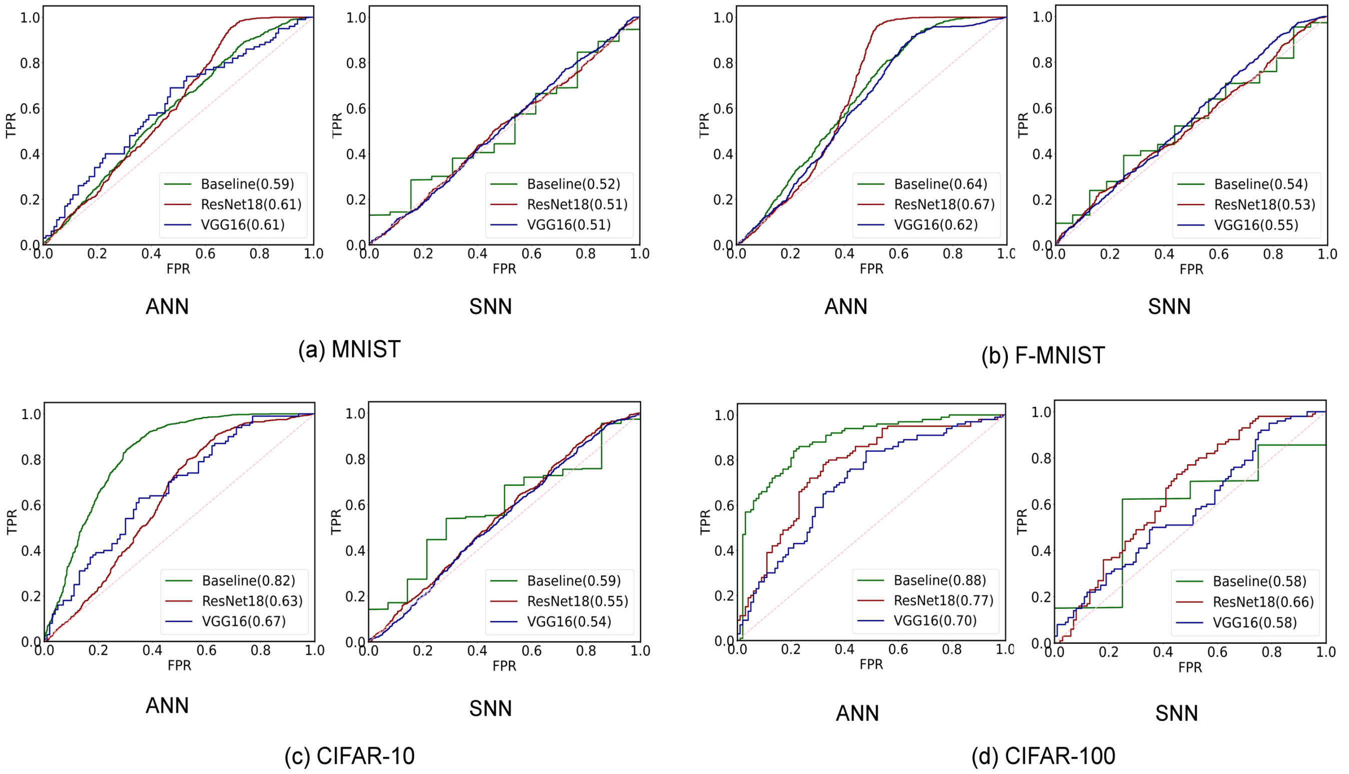
- *Rate Encoding:* Transforms input features into spikes by representing each feature as a probability of spike occurrence at each time step, with the neuron's firing rate directly tied to the intensity of the input signal. For our experiments, we set number of steps to 10.
- *Latency Encoding:* Encodes information based on the timing of spikes, where inputs with higher values result in earlier spikes, effectively using the temporal dimension to convey input magnitude. We used time step of 10 and set the RC constant, *Tau* to 5, and Threshold to 0.1 for our experiments.
- *Delta Encoding:* Event-driven and produces spikes in response to changes in input features over time, making it adept at capturing dynamic variations in data. In our experiments, we set the threshold to 0.1.

LAVA Framework:

LAVA is an innovative open source software framework designed for neuromorphic computing [63]. It provides a versatile and flexible environment for developing, simulating, and deploying neuromorphic applications. By abstracting hardware details through a process-based model, LAVA enables the creation of scalable and modular systems that can operate asynchronously across various neuromorphic and conventional hardware platforms. The framework supports rapid prototyping and detailed optimization, making it accessible to both researchers and developers interested in leveraging the unique capabilities of neuromorphic technologies. LAVA aligns particularly well with Intel's Loihi chip [9], a specialized neuromorphic processor. LAVA provides a seamless interface for developing applications that can efficiently run on Loihi. The LAVA framework also supports Spike-Timing-Dependent Plasticity (STDP), a biological learning rule that adjusts synaptic strengths based on the precise timing of spikes. This mechanism allows SNNs to learn temporal patterns and adapt to dynamic inputs. In our experiment, we used framework to evaluate the model's resilience against privacy attack (MIA).

4.4.2 *Evolutionary Learning Algorithm:*

Evolutionary algorithms, such as the Evolutionary Optimization for Neuromorphic Systems (EONS) [69], are capable of enabling the rapid prototyping of SNN applications. These applications can be adapted to hardware constraints and various learning scenarios, including classification [67, 68] and control tasks [56, 66]. EONS, implemented within the TENNLab framework, facilitates the co-design process between neuromorphic hardware and software, helping developers



(a) MNIST

(b) F-MNIST

(c) CIFAR-10

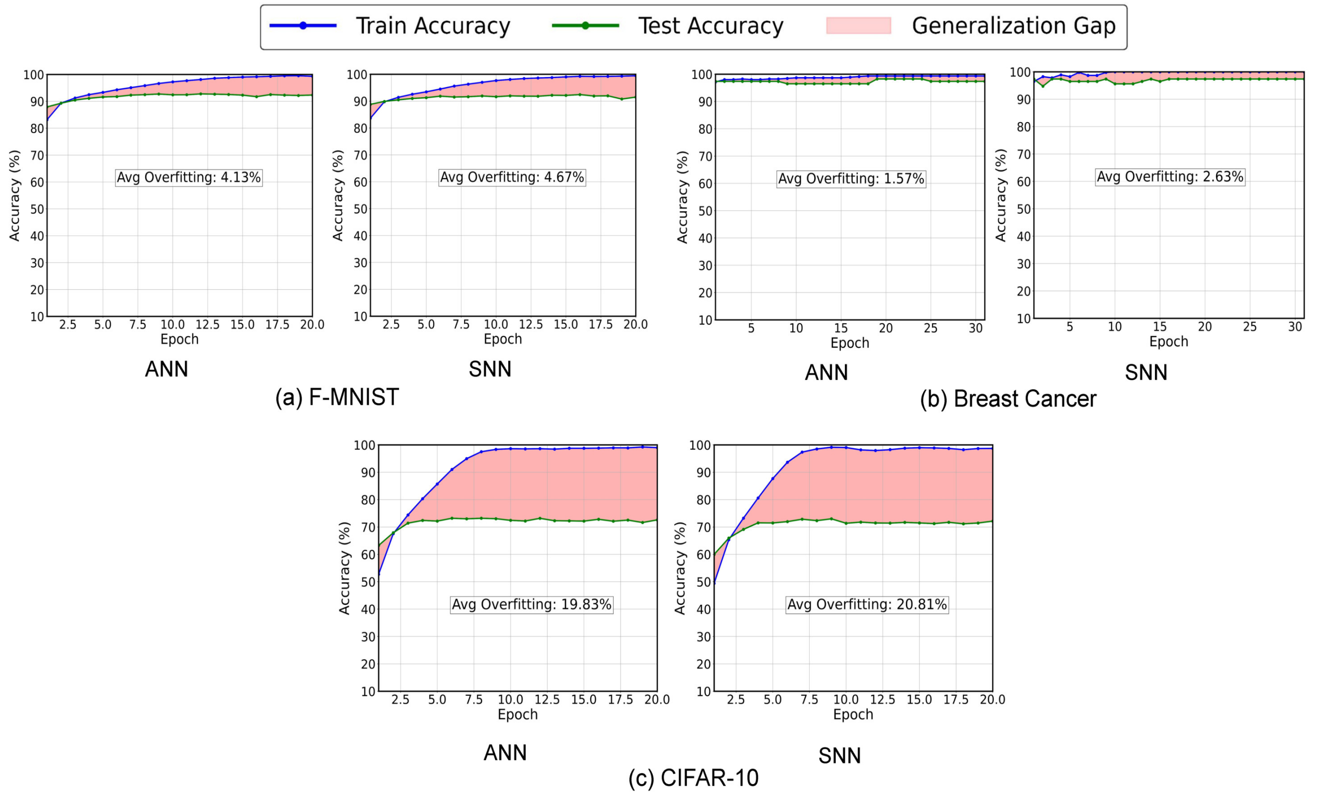
(d) CIFAR-100

Figure 4: ROC curves comparing ANN and SNN models across different datasets: (a) MNIST, (b) F-MNIST, (c) CIFAR-10, and (d) CIFAR-100.

Table 2: Comparison of ANN and SNN Resilience to MIA Across Diverse Datasets

Dataset	Architecture		ANN			SNN	
		Train Acc	Test Acc	Attack AUC	Train Acc	Test Acc	Attack AUC
MNIST	Baseline	99.96(±0.01)%	99.21(±0.03)%	**0.59(±0.008)**	99.95(±0.05)%	99.22(±0.02)%	**0.52(±0.002)**
	ResNet18	99.98(±0.10)%	99.57(±0.04)%	**0.61(±0.005)**	99.96(±0.009)%	99.51(±0.06)%	**0.51(±0.006)**
	VGG16	99.67(±0.05)%	99.50(±0.02)%	**0.61(±0.007)**	99.58(±0.12)%	99.37(±0.14)%	**0.51(±0.004)**
F-MNIST	Baseline	99.52(±0.13)%	92.77(±0.20)%	**0.64(±0.011**	99.42(±0.32)%	92.44(±0.19)%	**0.54(±0.008)**
	ResNet18	98.72(±0.21)%	93.66(±0.15)%	**0.67(±0.004)**	99.12(±0.09)%	91.79(±0.03)%	**0.53(±0.003)**
	VGG16	97.14(±0.17)%	93.06(±0.20)%	**0.62(±0.005)**	96.53(±0.29)%	90.34(±0.19)%	**0.55(±0.011)**
CIFAR-10	Baseline	99.24(±0.08)%	73.20(±0.43)%	**0.82(±0.095)**	99.13(±0.45)%	72.99(±0.33)%	**0.59(±0.005)**
	ResNet18	97.65(±0.51)%	90.81(±0.24)%	**0.63(±0.021)**	96.74(±0.44)%	86.85(±0.13)%	**0.55(±0.011)**
	VGG16	97.45(±0.41)%	88.74(±0.29)%	**0.67(±0.013)**	79.77(±0.31)%	74.92(±0.20)%	**0.53(±0.003)**
CIFAR-100	Baseline	98.31(±0.19)%	42.46(±0.35)%	**0.88(±0.016)**	99.42(±0.59)%	39.92(±0.67)%	**0.58(±0.019)**
	ResNet18	98.66(±0.23)%	70.92(±0.48)%	**0.77(±0.009)**	88.43(±0.27)%	60.65(±0.74)%	**0.66(±0.002)**
	VGG16	81.61(±0.17)%	58.32(±0.52)%	**0.70(±0.019)**	60.53(±0.22)%	52.32(±1.36)%	**0.58(±0.004)**
Iris	Baseline	100(±0.0)%	96.67(±2.34)%	**0.77(±0.13)**	97.04(±0.34)%	96.67(±0.0)%	**0.57(±0.020)**
Breast Cancer	Baseline	99.54(±0.04)%	97.37(±0.24)%	**0.65(±0.008)**	99.62(±0.0)%	96.98(±0.17)%	**0.55(±0.017)**

optimize neural network structures within the constraints of neuromorphic hardware. EONS interacts seamlessly with a wide variety of devices [69], architectures [70], and application [58] without necessitating any modifications to its underlying algorithm. This approach, implemented within the TENNLab framework, utilizes an

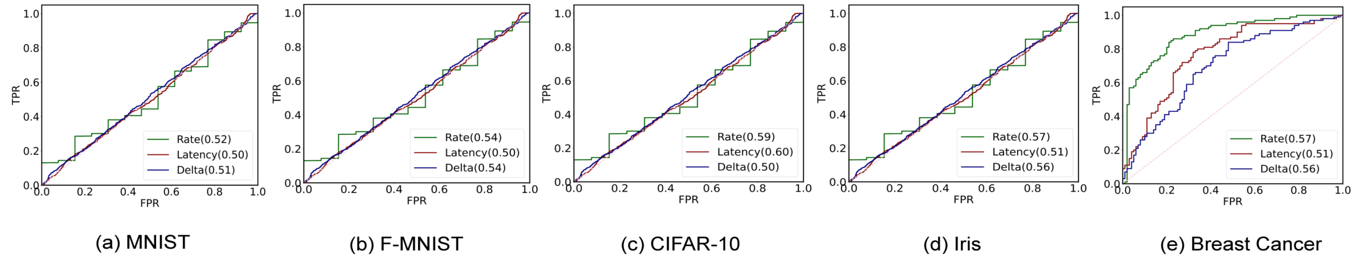


(a) F-MNIST

(b) Breast Cancer

(c) CIFAR-10

Figure 5: Overfitting analysis in ANN and SNN models across datasets: (a) F-MNIST, (b) Breast Cancer, and (c) CIFAR-10



(a) MNIST

(b) F-MNIST

(c) CIFAR-10

(d) Iris

(e) Breast Cancer

Figure 6: ROC curves showing the impact of Rate, Latency, and Delta modulation encoding under MIA for (a) MNIST, (b) F-MNIST, (c) CIFAR-10, (d) Iris, and (e) Breast Cancer datasets.

evolutionary algorithm to optimize neuromorphic network structure and weights. This process begins with generating a population of potential network solutions, which can be either randomly created or seeded with pre-existing networks [69]. The EONS process then evaluates these networks for fitness based on user-defined criteria and uses common genetic algorithm selection techniques such as tournament or fitness score to choose the most promising networks for reproduction. During reproduction, selected parent networks undergo crossover to swap segments of their structure and merging, which combines their entire structures into a single offspring. Mutation operations introduce random modifications to a network's nodes and edges, enhancing diversity and adaptation in the population. These operations are performed on a generalized network representation consisting of nodes and edges, allowing flexibility and adaptability across different hardware implementations. The entire EONS cycle, designed to optimize network parameters and structure efficiently, repeats until achieving desired performance metrics.

In our experiments we depict the impact of different EONs parameters, encoders and neuro processors inside the framework on the vulnerability of the SNN model against MIA on Iris and Breast Cancer dataset.

Population Size: In the EONS approach, population consists of potential solutions that form the genetic pool for evolution. This initial population can be randomly generated or seeded with pre-existing solutions. Each network is evaluated, and the best-performing networks are selected for reproduction. In our experiments, we vary the population size over 50, 100, 200 and 400 to understand its impact on resilience against MIAs.

Mutation Rate: The mutation rate in the EONS approach specifies the frequency at which random changes are introduced to the solutions. These mutations are essential for exploring new solution spaces and preventing the population from stagnating in local optima. Mutations can be structural, such as adding or deleting nodes and edges, or they can involve changes to parameters like thresholds or weights. In our experiments, we vary the mutation rate over 0.1, 0.5 and 0.9 to study its effects on the inherent privacy preservation in SNNs.

Crossover Rate: The crossover rate determines how often parts of two solutions are recombined to create new solutions, enhancing genetic diversity. In EONS, the algorithm uses a node-edge recombination method, where it mixes components from parent networks to form child networks. By varying the crossover rate, the algorithm maintains a diverse genetic pool which is essential for exploring the solution space and identifying high-performing network configurations. By varying the crossover rate over 0.1, 0.5 and 0.9, we aim to explore its impact on the resilience of SNNs against MIAs.

Neuroprocessors: We explore 3 neuroprocessors available in the TENNLab framework:

- *RISP* [59]: A lightweight neuro processor employing an integrate and fire model with discrete time steps for accumulating and evaluating action potentials, suitable for networks with integer synaptic delays.
- *Caspian* [36]: Provides a high-level API and a fast spiking simulator integrated with FPGA [37] architecture, enhancing development and deployment of neuromorphic solutions in size, weight, and power-constrained environments.
- *RAVENS* [16]: A versatile neuroprocessor from TENNLab with multiple implementations including software simulation, microcontroller, FPGA, ASIC [21], and Memristive ASIC [83](mRAVENS), catering to a wide range of computational needs in neuromorphic systems.

Encoding Techniques:

- *Flip Flop Technique*: It assigns inverted percentage values in even-numbered bins, ensuring smoother transitions and preserving information about minimum values. It is effective in applications like proximity sensing in LIDAR systems [57].
- *Triangle Technique*: It smooths input space by overlapping bins where values rise to 100% at bin boundaries and then fall, facilitating a more gradual representation of input data. This is useful in refined control applications.

5 Results

5.1 MIA Assessment

This section presents a comprehensive evaluation of ANN and SNN architectural resilience against MIAs across baseline, ResNet18, and VGG16 configurations. The privacy vulnerability is meausured using ROC-AUC metrics, where lower values indicate enhanced privacy preservation. Each experiment was repeated three times, with Table 2 reporting the mean and standard deviation. The low standard deviations across all metrics indicate the statistical stability of the results.

The experimental results, illustrated in Figure 4 and Table 2, demonstrate consistent privacy advantages of SNNs while maintaining competitive accuracy. For MNIST, while both architectures achieve comparable test accuracy (≃ 99%), SNNs exhibit lower AUC values (0.51-0.52) compared to ANNs (0.59-0.61). This privacy advantage becomes more pronounced in F-MNIST, where SNNs maintain AUC values between 0.53-0.55 while achieving 90-92% test accuracy, compared to ANNs' higher vulnerability (AUC 0.62-0.67) at similar accuracy levels.

The privacy advantage becomes more pronounced in complex datasets. For CIFAR-10, the baseline SNN maintains an AUC of 0.59 compared to ANN's 0.82, while ResNet18 and VGG16 SNN variants achieve AUCs of 0.55 and 0.53, respectively, substantially lower than their ANN counterparts (0.63 and 0.67). This trend persists in CIFAR-100, where the baseline SNN demonstrates an AUC of 0.58 versus ANN's 0.88, with ResNet18 and VGG16 SNN implementations maintaining AUCs of 0.66 and 0.58 compared to ANNs' 0.77 and 0.70. The ROC curves in Figure 4 visually demonstrate this enhanced privacy preservation across all architectural configurations. Notably, this privacy advantage does not significantly compromise model utility, as evidenced by the competitive accuracy metrics maintained across implementations.

Key Finding 1:

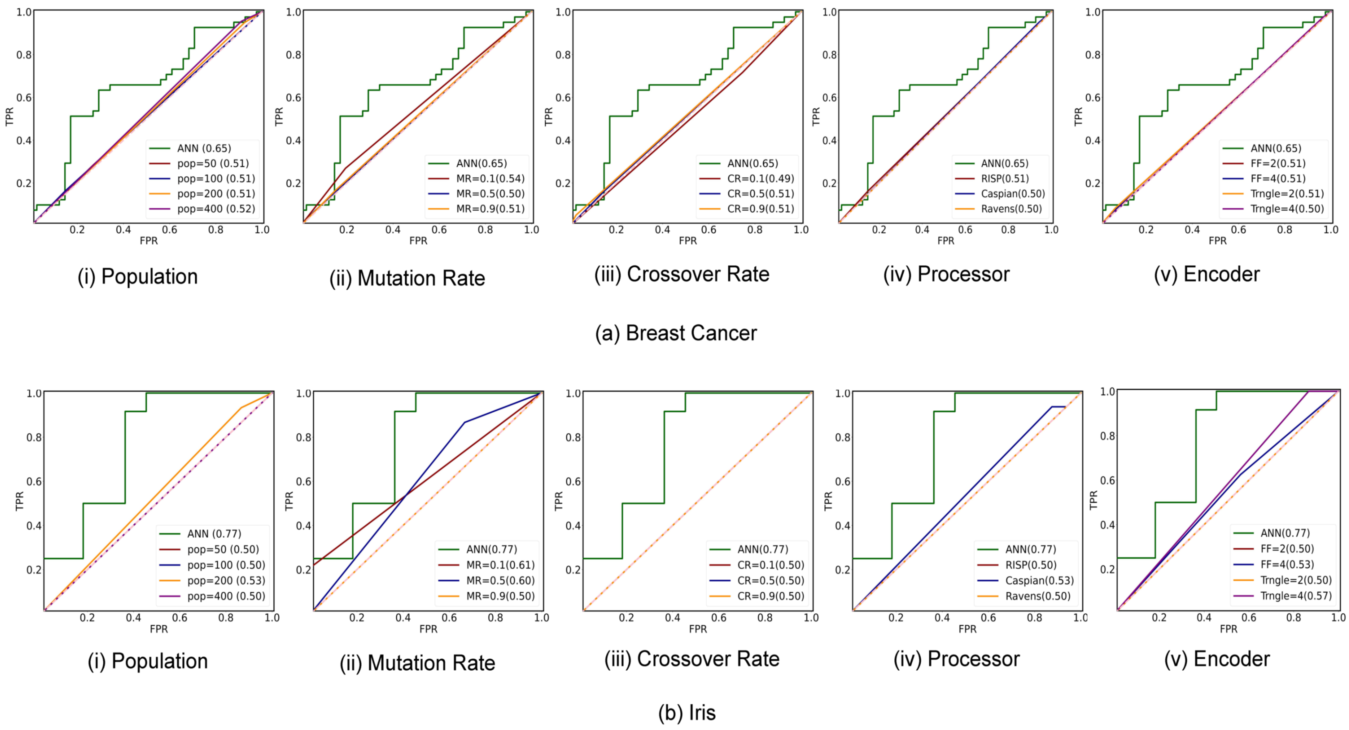
SNNs demonstrate enhanced privacy preservation across all architectural configurations compared to ANNs, while maintaining competitive accuracy.

5.2 Is SNN resilience driven by ANN overfitting?

To assess whether the improved resilience of SNNs is simply a result of ANN overfitting, the relationship between architectural privacy advantages and model generalization is examined across multiple datasets, including F-MNIST, Breast Cancer, and CIFAR-10. Overfitting is known to increase a model's vulnerability to MIA by reducing its generalizability. Overfitted models tend to memorize training data, which increases the likelihood that an adversary can successfully infer whether a particular data point was part of the training set, thus leading to a higher MIA AUC.

As demonstrated in Figure 5, both ANN and SNN models exhibit comparable levels of overfitting, as reflected by the similar gaps between their training and testing accuracies. For example, in F-MNIST(Figure 5a), the average overfitting for ANN is 4.13%, while for SNN, it is 4.67%. In the Breast Cancer dataset(Figure 5b), the average overfitting values are much lower, at 1.57% for ANN and 2.63% for SNN. These results indicate that overfitting occurs similarly in both ANNs and SNNs, ruling out overfitting in ANNs as the sole explanation for the enhanced resilience of SNNs against MIAs.

The consistency in overfitting patterns across both architectures demonstrated through comparable generalization gaps indicates that ANN's increased vulnerability to MIAs cannot be attributed



(i) Population (ii) Mutation Rate (iii) Crossover Rate (iv) Processor (v) Encoder

(a) Breast Cancer

(i) Population (ii) Mutation Rate (iii) Crossover Rate (iv) Processor (v) Encoder

(b) Iris

Figure 7: ROC curves comparing the impact of EONs Parameters in TennLab Framework under MIA on (a)Breast Cancer and (b)Iris Dataset

to differential overfitting behavior. Despite similar levels of overfitting, SNNs consistently exhibit lower MIA AUC values compared to ANNs. This finding establishes that the enhanced privacy preservation in SNNs originates from inherent architectural characteristics rather than advantages in generalization capability, as both architectures demonstrate comparable overfitting tendencies while exhibiting distinctly different privacy vulnerabilities.

5.3 SNN Exploration Space Assessment:

This section examines SNN privacy characteristics across different learning algorithms, frameworks, and their associated parameters.

5.3.1 ***Surrogate Gradient Algorithm*** In this section we analyze the effects of different encoding schemes within the snnTorch framework and the LAVA framework.

Impact of Encoding Schemes in snnTorch Framework: Figure 6 illustrates the differential impact of snnTorch's three encoding schemes on SNN vulnerability to membership inference attacks. For lower complexity datasets such as MNIST (Figure 6a) and F-MNIST (Figure 6b), Rate and Delta encoding mechanisms demonstrate comparable privacy preservation characteristics, with AUC values converging around 0.51 and 0.53 respectively. The CIFAR-10 dataset (Figure 6c) exhibits a broader vulnerability profile across all encoding implementations, with AUC values extending to 0.58, indicating slightly elevated susceptibility compared to simpler datasets. In the Iris dataset (Figure 6d), the impact of encoding choice becomes more pronounced, with Delta encoding demonstrating marginally higher vulnerability (AUC = 0.56), suggesting increased encoding sensitivity in lower-dimensional datasets. Despite these variations across datasets and encoding methods, the consistently lower range of AUC values underscores the inherent privacy preserving characteristics of SNN architectures.

Impact of LAVA Framework:

The LAVA framework implementation employs a sophisticated three-layer feed-forward neural network architecture incorporating Leaky Integrate-and-Fire (LIF) neurons with pre-trained synaptic weights. The experimental results, illustrated in Figure 8, show substantial privacy advantages in the LAVA-based SNN architecture, achieving an AUC of 0.52. This represents a significant 14.75% reduction in vulnerability compared to the conventional PyTorch-based ANN implementation, highlighting the framework's efficacy in preserving privacy while maintaining neuromorphic computing capabilities.

5.3.2 ***Evolutionary Algorithm:*** Figures 7a and 7b present comprehensive evaluations of Evolutionary Optimization parameters within the TennLab Framework across the Breast Cancer and Iris datasets respectively. The Iris dataset demonstrates robust privacy preservation across multiple parametric variations: population size modifications (Figure 7b(i)) yield AUCs ranging from 0.50 to 0.53,

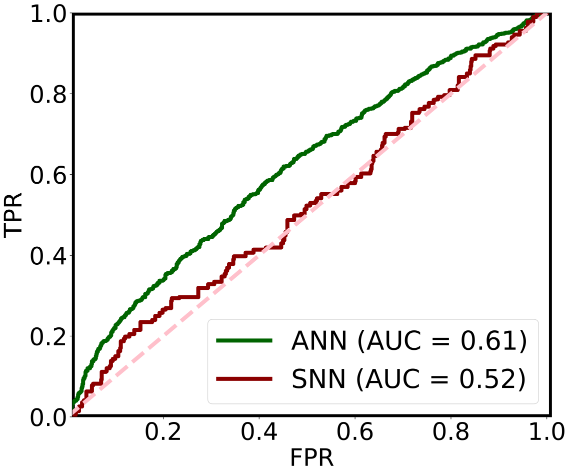


Figure 8: Impact of MIA on MNIST Dataset in LAVA Framework

while variations in crossover rates (Figure 7b(iii)) and mutation rates (Figure 7b(ii)) consistently maintain AUCs approaching 0.50, indicating stable privacy protection. The Breast Cancer dataset exhibits similarly robust characteristics, maintaining AUC values near 0.50 across diverse parameters including population size variations (Figure 7a(i)), crossover rate adjustments (Figure 7a(iii)), and different processor implementations (Figure 7a(iv)) such as RISP, Caspian, and Ravens. When compared to surrogate gradient learning implementations (Table 2), where Iris and Breast Cancer datasets demonstrated AUCs of 0.57 and 0.55 respectively, evolutionary algorithms consistently achieve superior privacy preservation with significantly lower AUC values. This enhanced resilience to membership inference attacks is particularly noteworthy given the practical hardware implementation capabilities of evolutionary algorithms within the TennLab framework. The combination of robust privacy preservation and hardware practicality suggests that evolutionary optimization approaches may offer compelling advantages for secure neuromorphic computing applications.

Key Finding 2:

Different SNN algorithms (surrogate gradient methods, STDP, and evolutionary optimization) demonstrate consistent privacy preservation across architectures and parameters, with evolutionary algorithms showing particular promise in combining privacy resilience with hardware practicality.

5.4 Privacy-Utility Trade off Assessment

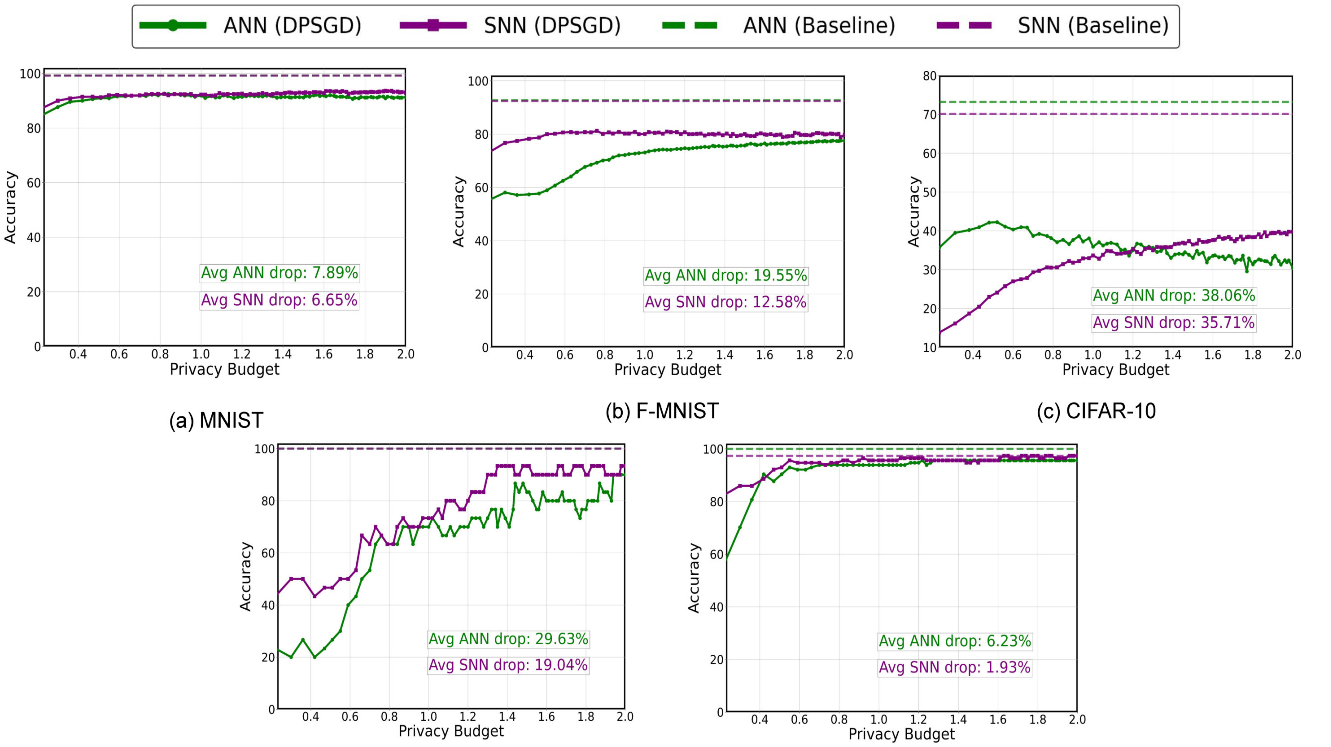
The privacy-utility evaluation implements DPSGD across both architectures with a standardized privacy budget (ϵ) range of 0.22-2. The comparative analysis, presented in Figure 9 and Table 3, quantifies accuracy degradation as the differential between baseline accuracy (pre-DPSGD) and DPSGD implementation accuracy throughout training epochs. The results demonstrate enhanced utility preservation in SNN implementations across all datasets under equivalent privacy constraints. On the MNIST dataset (Figure 9a), SNNs show an average accuracy reduction of 6.65% compared to ANNs' 7.89%. This pattern extends to F-MNIST (Figure 9b), where SNNs demonstrate a 12.58% accuracy decrease versus ANNs' 19.55%. CIFAR-10 (Figure 9c) maintains this trend with SNNs showing 27.87% reduction compared to ANNs' 34.43%, indicating sustained utility preservation even with increased data complexity. The enhanced utility preservation extends to tabular datasets, with Breast Cancer (Figure 9e) showing minimal SNN accuracy degradation of 1.93% compared to ANNs' 6.23%. Similarly, the Iris dataset (Figure 9d) demonstrates SNN accuracy reduction of 19.04% versus ANNs' 29.63%. These results highlight SNNs' consistent ability to maintain utility under privacy constraints across diverse data modalities.

Key Finding 3:

Under equivalent differential privacy budget, SNNs consistently demonstrate **lower accuracy degradation** than ANNs across all datasets, indicating better utility preservation while maintaining privacy guarantees.

6 Related Work

As machine learning systems handle increasingly sensitive data, the potential for privacy violations becomes increasingly significant. Li et al. [42] categorize these privacy challenges into two primary areas: privacy attacks and privacy preserving techniques. Privacy attacks have emerged as a significant concern in ML due to the growing realization that models can inadvertently leak sensitive data. These attacks can broadly be classified into different types, such as model inversion attacks, model extraction attacks, MIA. Model inversion attacks [17] reconstruct input data from outputs, while extraction attacks [31] replicate model's functionality without direct access to its architecture or parameters. Among these, MIAs are notable for inferring whether a specific data point was used in training. According to the survey conducted by Hu et al. [27], MIAs were first proposed in the context of genomic data by Homer et al. [25] where an attacker could identify an individual's genome in a dataset based on summary statistics. Later, Shokri et al.[74] introduced the first systematic MIA framework, showing how adversaries could use shadow models to infer training data membership. Salem et al. [64] reduced the complexity by demonstrating that a single shadow model can perform well compared to using multiple models, and they introduced metric based attacks that rely on confidence scores and entropy without the need for identical data distribution between shadow and target models. Nasr et al. [48] further expanded MIA into white box settings, demonstrating that attackers with access to internal model parameters can perform even more effective MIAs. Melis et al. [46] extended MIA to federated learning, highlighting vulnerabilities in distributed learning settings, where multiple parties collaboratively train a model. Song and Mittal [76] highlighted the increased privacy risks in generative models such as GANs, where membership inference attacks could be carried out on synthetic data generators. Recent



(a) MNIST

(b) F-MNIST

(c) CIFAR-10

(d) Iris

(e) Breast Cancer

Figure 9: Model Performance over increasing Privacy Budget (ε) with DPSGD across (a) MNIST, (b) F-MNIST, (c) CIFAR-10, (d) Iris, and (e) Breast Cancer datasets.

Table 3: Privacy-Utility Trade off Comparison between ANN and SNN

Dataset	Privacy Budget Range	ANN Avg Accuracy Drop	SNN Avg Accuracy Drop
MNIST	0.22 - 2.00	7.89(±0.14)%	6.65(±0.08)%
Fashion-MNIST	0.22 - 2.00	19.55(±0.84)%	12.58(±0.78)%
CIFAR-10	0.22 - 2.00	34.43(±0.20)%	27.87(±0.11)%
Breast Cancer	0.22 - 2.00	6.23(±0.33)%	1.93(±0.47)%
Iris	0.22 - 2.00	29.63(±0.81)%	19.04(±0.72)%

work by Ilyas et al. introduced LiRA (Likelihood Ratio Attack) [8], a method that further improves the accuracy of MIAs by leveraging confidence scores more effectively to distinguish training from non-training data points. In 2024, Zarifzadeh et al. [89] introduced RMIA, a high-power membership inference attack that outperforms prior methods like LiRA and Attack-R, demonstrating superior robustness, particularly at low false positive rates (FPRs), using likelihood ratio tests.

To counteract these privacy attacks, several privacy preserving techniques have been developed.These techniques range from cryptographic approaches like homomorphic encryption(HE) [86] and secure multi-party computation(SMPC) [91] to learning based defenses such as model obfuscation [23] and knowledge distillation [22]. However, these methods often suffer from computational inefficiencies, particularly in large scale systems. Another approach is Federated Learning(FL) [40], which enables collaborative model training without sharing raw data, but remains vulnerable to attacks like MIA. Among these, Differential Privacy(DP) [11] has gained prominence due to its strong theoretical guarantees and practical applicability in machine learning settings. It provides a systematic framework for protecting individual data points by introducing noise during computations. In ML, this concept has been adapted through various algorithms, with DPSG) [77] being the most prominent which applies DP principles by adding noise to the gradient updates during training, offering a practical way to maintain privacy while training large models without significantly compromising accuracy.

While much of the data privacy research has centered around ANNs, expanding these investigations to SNNs is necessary. SNNs not only offer performance levels comparable to ANNs but also exhibit superior energy efficiency and hardware integration capabilities, positioning them as promising candidates for exploring inherent privacy features. Although privacy attacks on neuromorphic architectures remain underexplored, existing studies have yet to confirm SNNs' potential resistance to such threats. However, significant strides have been made in privacy preserving techniques within the neuromorphic domain. For instance, recent efforts by Han et al. [24] focus on developing privacy preserving methods for SNNs, particularly utilizing FL and DP to address both computational efficiency and privacy challenges. Li et al.[41] introduced a framework that combines Fully Homomorphic Encryption (FHE) with SNNs, enabling encrypted inference while preserving SNNs' energy efficiency and computational advantages. Similarly, Nikfam et al.[52] developed an HE framework tailored for SNNs, offering enhanced accuracy over Deep Neural Networks(DNNs) under encryption, while balancing computational efficiency. Additionally, Safronov et al.[35] proposed PrivateSNN, a privacy preserving framework for SNNs that employs differential privacy to mitigate membership inference attacks, maintaining the energy efficient nature of SNNs.

7 Conclusion

The increasing deployment of machine learning systems in privacy sensitive domains has heightened the need for architectures that inherently protect data privacy while maintaining computational efficiency. This investigation addresses these requirements through a thorough examination of privacy characteristics in SNNs, evaluating their resilience against privacy attacks compared with traditional neural architectures. The discrete, event driven nature of spike based processing and temporal dynamics in SNNs may inherently limit information leakage compared to the continuous activations in ANNs, providing natural defense mechanisms against privacy attacks.

The experimental analysis establishes enhanced privacy preservation in SNN architectures, with attack AUC values significantly lower than traditional ANNs across all evaluated datasets (CIFAR-10: 0.59 vs 0.82; CIFAR-100: 0.58 vs 0.88). The privacy gain is particularly pronounced when employing evolutionary learning algorithms, which demonstrate superior resilience compared to gradient-based methods. Additionally, SNNs exhibit improved utility preservation under differential privacy constraints, maintaining higher accuracy levels compared to ANNs when implementing DPSGD across diverse datasets.

While these findings highlight SNNs' potential for privacy sensitive applications, particularly in resource constrained environments, they are focused on privacy preservation applications. Despite their privacy advantages, SNNs face challenges including complex training processes, potential scalability limitations, and and reliance on specialized hardware, which is necessary for optimal performance. However, within the scope of privacy preservation, their unique computational characteristics offer promising directions for secure neural architectures. Future research directions include hardware implementation analysis through Intel's Loihi neuromorphic processor, expanding privacy threat models, and integrating differential privacy mechanisms with evolutionary optimization in the TennLab framework. These investigations aim to further understand and enhance privacy preservation capabilities in neuromorphic architectures while maintaining their computational advantages.

Acknowledgments

This work was funded in part by National Science Foundation through award CCF2319619 and in part by the CHIST-ERA grant TruBrain, by the UK's Engineering and Physical Sciences Research Council (EPSRC) EP/Y03631X/1. The authors used AI-based tools, ChatGPT [55] and Claude [3] to revise the text in all Sections to correct any typos or grammatical errors.

References

[1] Karim Abouelmehdi, Abderrahim Beni-Hssane, Hayat Khaloufi, and Mostafa Saadi. 2017. Big data security and privacy in healthcare: A Review. *Procedia Computer Science* 113 (2017), 73–80.
[2] Filipp Akopyan, Jun Sawada, Andrew Cassidy, Rodrigo Alvarez-Icaza, John Arthur, Paul Merolla, Nabil Imam, Yutaka Nakamura, Pallab Datta, Gi-Joon Nam, et al. 2015. Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip. *IEEE transactions on computer-aided design of integrated circuits and systems* 34, 10 (2015), 1537–1557.
[3] Anthropic. 2024. Claude: Anthropic Language Model. https://www.anthropic.com/claude Accessed: [insert date].
[4] William Aspray. 1990. *John von Neumann and the origins of modern computing.* Mit Press, x.
[5] Daniel Auge, Julian Hille, Etienne Mueller, and Alois Knoll. 2021. A survey of encoding techniques for signal processing in spiking neural networks. *Neural Processing Letters* 53, 6 (2021), 4693–4710.
[6] Elisa Bertino. 2016. Data security and privacy: Concepts, approaches, and research directions. In *2016 IEEE 40th Annual computer Software and Applications conference (cOMPSAc)*, Vol. 1. IEEE, x, x, 400–407.
[7] Natalia Caporale and Yang Dan. 2008. Spike timing–dependent plasticity: a Hebbian learning rule. *Annu. Rev. Neurosci.* 31 (2008), 25–46.
[8] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP).* IEEE, x, x, 1897–1914.
[9] Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. 2018. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro* 38, 1 (2018), 82–99.
[10] Emiliano De Cristofaro. 2020. An overview of privacy in machine learning. *arXiv preprint arXiv:2005.08679* x, x (2020), x.
[11] Cynthia Dwork. 2006. Differential privacy. In *International colloquium on automata, languages, and programming.* Springer, x, x, 1–12.
[12] Rida El-Allami, Alberto Marchisio, Muhammad Shafique, and Ihsen Alouani. 2021. Securing Deep Spiking Neural Networks against Adversarial Attacks through Inherent Structural Parameters. In *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE).* x, x, 774–779. https://doi.org/10.23919/DATE51398.2021.9473981
[13] Jason K. Eshraghian. 2023. SNNTorch: Tutorial 1. https://snntorch.readthedocs.io/en/latest/tutorials/tutorial_1.html Accessed: 2024-05-28.
[14] Wei Fang, Yanqi Chen, Jianhao Ding, Zhaofei Yu, Timothée Masquelier, Ding Chen, Liwei Huang, Huihui Zhou, Guoqi Li, and Yonghong Tian. 2023. SpikingJelly: An open-source machine learning infrastructure platform for spike-based intelligence. *Science Advances* 9, 40 (2023), eadi1480. https://doi.org/10.1126/sciadv.adi1480 arXiv:https://www.science.org/doi/pdf/10.1126/sciadv.adi1480
[15] Diana Florea and Silvia Florea. 2020. Big Data and the ethical implications of data privacy in higher education research. *Sustainability* 12, 20 (2020), 8744.
[16] Adam Z Foshie, James S Plank, Garrett S Rose, and Catherine D Schuman. 2023. Functional specification of the ravens neuroprocessor. *arXiv preprint arXiv:2307.15232* x, x (2023), x.
[17] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security.* x, x, 1322–1333.
[18] Samanwoy Ghosh-Dastidar and Hojjat Adeli. 2009. Spiking neural networks. *International journal of neural systems* 19, 04 (2009), 295–308.

[19] Roger Gomm, Martyn Hammersley, and Peter Foster. 2000. Case study and generalization. *Case study method* x, x (2000), 98–115.

[20] Neil Zhenqiang Gong and Bin Liu. 2018. Attribute inference attacks in online social networks. *ACM Transactions on Privacy and Security (TOPS)* 21, 1 (2018), 1–30.

[21] Tim Good and Mohammed Benaissa. 2008. ASIC hardware performance. In *New Stream Cipher Designs: The eSTREAM Finalists*. Springer, x, 267–293.

[22] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision* 129, 6 (2021), 1789–1819.

[23] Yunqi Guo, Zhaowei Tan, Kaiyuan Chen, Songwu Lu, and Ying Nian Wu. 2021. A Model Obfuscation Approach to IoT Security. In *2021 IEEE Conference on Communications and Network Security (CNS)*. x, x, 1–9. https://doi.org/10.1109/CNS53000.2021.9705028

[24] Bing Han, Qiang Fu, and Xinliang Zhang. 2023. Towards Privacy-Preserving Federated Neuromorphic Learning via Spiking Neuron Models. *Electronics* 12, 18 (2023), 3984.

[25] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. 2008. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS genetics* 4, 8 (2008), e1000167.

[26] Jianan Hong, Kaiping Xue, Yingjie Xue, Weikeng Chen, David SL Wei, Nenghai Yu, and Peilin Hong. 2017. TAFC: Time and attribute factors combined access control for time-sensitive data in public cloud. *IEEE Transactions on Services Computing* 13, 1 (2017), 158–171.

[27] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. 2022. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)* 54, 11s (2022), 1–37.

[28] Chi-Hong Hwang and Allen C-H Wu. 2000. A predictive system shutdown method for energy saving of event-driven computation. *ACM Transactions on Design Automation of Electronic Systems (TODAES)* 5, 2 (2000), 226–241.

[29] Intel Labs. 2023. *LAVA – An Open Source Framework for Neuromorphic Computing*. Intel. https://lava-nc.org/

[30] Balajee JM et al. 2018. Data wrangling and data leakage in machine learning for healthcare. *x* x, x (2018), x.

[31] Mika Juuti, Sebastian Szyller, Samuel Marchal, and N Asokan. 2019. PRADA: protecting against DNN model stealing attacks. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, x, x, 512–527.

[32] Saleem A Kassam. 2012. *Signal detection in non-Gaussian noise*. Springer Science & Business Media, x.

[33] Nikhil Ketkar, Jojo Moolayil, Nikhil Ketkar, and Jojo Moolayil. 2021. Introduction to pytorch. *Deep learning with python: learn best practices of deep learning models with PyTorch* x, x (2021), 27–91.

[34] Anya Kim, John McDermott, and Myong Kang. 2010. Security and architectural issues for national security cloud computing. In *2010 IEEE 30th International Conference on Distributed Computing Systems Workshops*. IEEE, x, x, 21–25.

[35] Youngeun Kim, Yeshwanth Venkatesha, and Priyadarshini Panda. 2022. Privatesnn: privacy-preserving spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. x, x, 1192–1200.

[36] Shruti Kulkarni, Maryam Parsa, J Parker Mitchell, and Catherine Schuman. 2021. Training spiking neural networks with synaptic plasticity under integer representation. In *International Conference on Neuromorphic Systems 2021*. x, x, 1–7.

[37] Ian Kuon, Russell Tessier, Jonathan Rose, et al. 2008. FPGA architecture: Survey and challenges. *Foundations and Trends® in Electronic Design Automation* 2, 2 (2008), 135–253.

[38] Yann LeCun, Corinna Cortes, and CJ Burges. 2010. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010), x.

[39] Seungsin Lee, Younghee Lee, Joing-In Lee, and Jungkun Park. 2015. Personalized e-services: consumer privacy concern and information sharing. *Social Behavior and Personality: an international journal* 43, 5 (2015), 729–740.

[40] Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. 2020. A review of applications in federated learning. *Computers & Industrial Engineering* 149 (2020), 106854.

[41] Pengbo Li, Huifang Huang, Ting Gao, Jin Guo, and Jinqiao Duan. 2023. Efficient Privacy-Preserving Convolutional Spiking Neural Networks with FHE. *arXiv preprint arXiv:2309.09025* x, x (2023), x.

[42] Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. 2021. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–36.

[43] Dingbang Liu, Hao Yu, and Yang Chai. 2021. Low-power computing with neuromorphic engineering. *Advanced Intelligent Systems* 3, 2 (2021), 2000150.

[44] Entao Luo, Md Zakirul Alam Bhuiyan, Guojun Wang, Md Arafatur Rahman, Jie Wu, and Mohammed Atiquzzaman. 2018. Privacyprotector: Privacy-protected patient data collection in IoT-based healthcare systems. *IEEE Communications Magazine* 56, 2 (2018), 163–168.

[45] Danijela Marković, Alice Mizrahi, Damien Querlioz, and Julie Grollier. 2020. Physics for neuromorphic computing. *Nature Reviews Physics* 2, 9 (2020), 499–510.

[46] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, x, x, 691–706.

[47] Qingyan Meng, Mingqing Xiao, Shen Yan, Yisen Wang, Zhouchen Lin, and Zhi-Quan Luo. 2022. Training high-performance low-latency spiking neural networks by differentiation on spike representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. x, x, 12444–12453.

[48] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, x, x, 739–753.

[49] Jonathan J Nassi and Edward M Callaway. 2009. Parallel processing strategies of the primate visual system. *Nature reviews neuroscience* 10, 5 (2009), 360–372.

[50] Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine* 36, 6 (2019), 51–63.

[51] Anh Nguyen, Jason Yosinski, and Jeff Clune. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Vol. x. x, x, 427–436.

[52] Farzad Nikfam, Raffaele Casaburi, Alberto Marchisio, Maurizio Martina, and Muhammad Shafique. 2023. A Homomorphic Encryption Framework for Privacy-Preserving Spiking Neural Networks. *Information* 14, 10 (2023), 537.

[53] Wilkie Olin-Ammentorp, Karsten Beckmann, Catherine D Schuman, James S Plank, and Nathaniel C Cady. 2021. Stochasticity and robustness in spiking neural networks. *Neurocomputing* 419 (2021), 23–36.

[54] Lubos Omelina, Jozef Goga, Jarmila Pavlovicova, Milos Oravec, and Bart Jansen. 2021. A survey of iris datasets. *Image and Vision Computing* 108 (2021), 104109.

[55] OpenAI. 2024. ChatGPT: OpenAI Language Model. https://www.openai.com/chatgpt Accessed: [insert date].

[56] Robert Patton, Catherine Schuman, Shruti Kulkarni, Maryam Parsa, J Parker Mitchell, N Quentin Haas, Christopher Stahl, Spencer Paulissen, Prasanna Date, Thomas Potok, et al. 2021. Neuromorphic computing for autonomous racing. In *International conference on neuromorphic systems 2021*. x, x, 1–5.

[57] James Plank, Charles Rizzo, Kirolos Shahat, Grant Bruer, Trevor Dixon, Michael Goin, Grace Zhao, Jeremy Anantharaj, Catherine Schuman, Mark Dean, et al. 2019. *The TENNLab suite of LIDAR-based control applications for recurrent, spiking, neuromorphic systems*. Technical Report. Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States).

[58] James S. Plank, Catherine D. Schuman, Grant Bruer, Mark E. Dean, and Garrett S. Rose. 2018. The TENNLab Exploratory Neuromorphic Computing Framework. *IEEE Letters of the Computer Society* 1, 2 (2018), 17–20. https://doi.org/10.1109/LOCS.2018.2885976

[59] James S Plank, ChaoHui Zheng, Bryson Gullett, Nicholas Skuda, Charles Rizzo, Catherine D Schuman, and Garrett S Rose. 2022. The case for risp: A reduced instruction spiking processor. *arXiv preprint arXiv:2206.14016* x, x (2022), x.

[60] Md Atiqur Rahman, Tanzila Rahman, Robert Laganière, Noman Mohammed, and Yang Wang. 2018. Membership Inference Attack against Differentially Private Deep Learning Model. *Trans. Data Priv.* 11, 1 (2018), 61–79.

[61] S. Rajan, Sichun Wang, R. Inkol, and A. Joyal. 2006. Efficient approximations for the arctangent function. *IEEE Signal Processing Magazine* 23, 3 (2006), 108–111. https://doi.org/10.1109/MSP.2006.1628884

[62] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2018. Do cifar-10 classifiers generalize to cifar-10? *arXiv preprint arXiv:1806.00451* x, x (2018), x.

[63] Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. 2019. Towards spike-based machine intelligence with neuromorphic computing. *Nature* 575, 7784 (2019), 607–617.

[64] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. 2018. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246* x, x (2018), x.

[65] David Salomon. 2012. *Data privacy and security*. Springer Science & Business Media, x.

[66] Catherine Schuman, Robert Patton, Shruti Kulkarni, Maryam Parsa, Christopher Stahl, N Quentin Haas, J Parker Mitchell, Shay Snyder, Amelie Nagle, Alexandra Shanafield, et al. 2022. Evolutionary vs imitation learning for neuromorphic control at the edge. *Neuromorphic Computing and Engineering* 2, 1 (2022), 014002.

[67] Catherine D Schuman, Shruti R Kulkarni, Maryam Parsa, J Parker Mitchell, Bill Kay, et al. 2022. Opportunities for neuromorphic computing algorithms and applications. *Nature Computational Science* 2, 1 (2022), 10–19.

[68] Catherine D Schuman, J Parker Mitchell, Maryam Parsa, James S Plank, Samuel D Brown, Garrett S Rose, Robert M Patton, and Thomas E Potok. 2020. Automated Design of Neuromorphic Networks for Scientific Applications at the Edge. In

2020 International Joint Conference on Neural Networks (IJCNN). IEEE, x, x, 1–7.

[69] Catherine D Schuman, J Parker Mitchell, Robert M Patton, Thomas E Potok, and James S Plank. 2020. Evolutionary optimization for neuromorphic systems. In *Proceedings of the 2020 Annual Neuro-Inspired Computational Elements Workshop*. x, x, 1–9.

[70] Catherine D. Schuman, J. Parker Mitchell, Robert M. Patton, Thomas E. Potok, and James S. Plank. 2020. Evolutionary Optimization for Neuromorphic Systems. In *x*. Association for Computing Machinery, New York, NY, USA, x. https://doi.org/10.1145/3381755.3381758

[71] Neha Sharma, Vibhor Jain, and Anju Mishra. 2018. An analysis of convolutional neural networks for image classification. *Procedia computer science* 132 (2018), 377–384.

[72] Virat Shejwalkar, Huseyin A Inan, Amir Houmansadr, and Robert Sim. 2021. Membership inference attacks against nlp classification models. In *NeurIPS 2021 Workshop Privacy in Machine Learning*. x, x, x.

[73] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. 2016. Edge computing: Vision and challenges. *IEEE internet of things journal* 3, 5 (2016), 637–646.

[74] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*. IEEE, x, x, 3–18.

[75] SNN-Torch Development . 2021. SNN-Torch Tutorial 1: Introduction to Spiking Neural Networks with snnTorch. https://snntorch.readthedocs.io/en/latest/tutorials/tutorial_1.html

[76] Liwei Song and Prateek Mittal. 2021. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security 21)*. x, x, 2615–2632.

[77] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. 2013. Stochastic gradient descent with differentially private updates. In *2013 IEEE global conference on signal and information processing*. IEEE, x, x, 245–248.

[78] Bhavani Thuraisingham. 2002. Data mining, national security, privacy and civil liberties. *ACM SIGKDD Explorations Newsletter* 4, 2 (2002), 1–5.

[79] Manas Tripathi and Arunabha Mukhopadhyay. 2020. Financial loss due to a data privacy breach: An empirical analysis. *Journal of Organizational Computing and Electronic Commerce* 30, 4 (2020), 381–400.

[80] U.S. Department of Health & Human Services. 2024. Health Information Privacy. https://www.hhs.gov/hipaa/index.html Accessed: 2024-06-03.

[81] Valerio Venceslai, Alberto Marchisio, Ihsen Alouani, Maurizio Martina, and Muhammad Shafique. 2020. NeuroAttack: Undermining Spiking Neural Networks Security through Externally Triggered Bit-Flips. In *2020 International Joint Conference on Neural Networks (IJCNN)*. x, x, 1–8. https://doi.org/10.1109/IJCNN48605.2020.9207351

[82] Wei Wang, Giacomo Pedretti, Valerio Milo, Roberto Carboni, Alessandro Calderoni, Nirmal Ramaswamy, Alessandro S Spinelli, and Daniele Ielmini. 2018. Learning of spatiotemporal patterns in a spiking neural network with resistive switching synapses. *Science advances* 4, 9 (2018), eaat4752.

[83] Ryan Weiss. 2022. Hardware for memristive neuromorphic systems with reliable programming and online learning. *Hardware Programming* x, x (2022), x.

[84] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* x, x (2017), x.

[85] Jie Xu, Wei Zhang, and Fei Wang. 2021. A (DP)2 SGD: Asynchronous Decentralized Parallel Stochastic Gradient Descent With Differential Privacy. *IEEE transactions on pattern analysis and machine intelligence* 44, 11 (2021), 8036–8047.

[86] Xun Yi, Russell Paulet, Elisa Bertino, Xun Yi, Russell Paulet, and Elisa Bertino. 2014. *Homomorphic encryption*. Springer, x.

[87] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, et al. 2021. Opacus: User-friendly differential privacy library in PyTorch. *arXiv preprint arXiv:2109.12298* x, x (2021), x.

[88] Haifei Yu and Xinyu He. 2021. Corporate Data Sharing, Leakage, and Supervision Mechanism Research. *Sustainability* 13, 2 (2021), 931.

[89] Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. 2024. Low-Cost High-Power Membership Inference Attacks. In *Forty-first International Conference on Machine Learning*. x, x, x.

[90] Tielin Zhang, Yi Zeng, Dongcheng Zhao, and Mengting Shi. 2018. A plasticity-centric approach to train the non-differential spiking neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32. x, x, x.

[91] Chuan Zhao, Shengnan Zhao, Minghao Zhao, Zhenxiang Chen, Chong-Zhi Gao, Hongwei Li, and Yu-an Tan. 2019. Secure multi-party computation: theory, practice and applications. *Information Sciences* 476 (2019), 357–372.

[92] Matjaz Zwitter and Milan Soklic. 1988. Breast Cancer. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C51P4M.