

MACHINE LEARNING WITH SPIKES

Brage Wiseth
Universitiy of Oslo
2025-10-29

ABSTRACT

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliquam quaerat voluptatem. Ut enim aequa doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet, ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum defuturum, quas natura non depravata desiderat. Et quem ad me accedit, saluto: 'chaere,' inquam, 'Tite!' lictores, turma omnis chorusque: 'chaere, Tite!' hinc hostis mi Albucius, hinc inimicus. Sed iure Mucius. Ego autem mirari satis non queo unde hoc sit tam insolens domesticarum rerum fastidium. Non est omnino hic docendi locus; sed ita prorsus existimo, neque eum Torquatum, qui hoc primus cognomen invenerit, aut torquem illum hosti detraxisse, ut aliquam ex eo est consecutus? – Laudem et caritatem, quae sunt vitae sine metu degendae praesidia firmissima. – Filium morte multavit. – Si sine causa, nolle me ab eo delectari, quod ista Platonis, Aristoteli, Theophrasti orationis ornamenta neglexerit. Nam illud quidem physici, credere aliquid esse minimum, quod profecto numquam putavisset, si a Polyaeno, familiari suo, geometrica discere maluisset quam illum etiam ipsum dedocere. Sol Democrito magnus videtur, quippe homini erudito in geometriaque perfecto, huic pedalis fortasse; tantum enim esse omnino in nostris poetis aut inertissimae segnitiae est aut fastidii delicatissimi. Mihi quidem videtur, inermis ac nudus est. Tollit definitiones, nihil de dividendo ac partiendo docet, non quo ignorare vos arbitrer, sed ut.

ACKNOWLEDGEMENTS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliquam quaerat voluptatem. Ut enim aequo doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere.

CONTENTS

1· Introduction And Theory	4
1.1· Established Artificial Intelligence	5
1.1.1· Problems With The Established Methods	5
1.2· Neuroscience 101	6
1.2.1· Neuron Models	6
1.2.2· Encoding	6
1.2.3· Learning	6
1.2.4· Network	7
1.3· Neuromorphic engineering	8
2· Methodology	8
2.1· Neuron Models	10
2.2· Learning	11
2.3· Network	11
3· Results	11
4· Discussion	11
Bibliography	11

1 · INTRODUCTION AND THEORY

The concept of intelligence, how it arises and what needs to be in place for it to occur, is probably been some of the longest standing questions in human history. How and if it can be reproduced artificially is a particularly hot topic today. Getting answers to these questions will not only help us understand our own minds but also brings the promise of unlocking new technology discovering new drugs or materials, it may be the last invention humans ever need to make. In recent years we have crept ever closer to answer some of these questions. New state of the art artificial intelligence systems have achieved remarkable success like the sophisticated language capabilities of GPT models and the protein-folding predictions of AlphaFold.

Despite these triumphs, a significant gap persists between artificial systems and their biological counterparts. Evidently, these AI systems might posses superhuman capabilities in one or a few domains but none of them surpass humans in all, what we call Artificial General Intelligence (AGI). Also more relevant to this thesis is that current state-of-the-art ANNs, require vast amount of data, computatuon and energy resources. This demand stands in stark contrast to the biological brain—an extraordinarily complex and efficient organ estimated to operate on merely 20-30 Watts while also sitting comfortably in the AGI category. This profound difference in efficiency and capability suggests that contemporary ANN paradigms, might be missing or oversimplifying fundamental principles crucial for truly intelligent and scalable computation.

In this thesis we explore new approaches that first and foremost might solve the critical limitations of scalability and energy efficiency in artificial intelligence. But also hopefully lay the foundation for systems that might eventually unlock true AGI. This likely requires moving beyond current mainstream ANN architectures. We will explore the potential of incorporating more sophisticated biological principles into AI design. This involves investigating alternative computational paradigms, inspired by mechanisms such as sparse, event-driven processing observed in Spiking Neural Networks (SNNs), the role of temporal dynamics in neural coding, or the potential computational advantages of systems operating near critical states. The central challenge lies in identifying and abstracting the truly essential biological mechanisms for intelligence and efficiency, distinguishing core principles from intricate biological details that may not be necessary for artificial implementation. Concretly this thesis wants to

- Explore how information-flow based on sparse events might be implemented in a network
- Explore learning algorithms suitable for such a network

In the succeeding sections we will lay down the theoretical foundations that we base our methods on

Section 1.1 · Established Artificial Intelligence we will get familiar with the current AI methods

Section 1.2 · neuroscience 101 review neuroscience literature

neuromorphic engineering 1.3 neuromorphic engineering

method 2 bla bla bla

results 3 blabla bla bla

discussion 4 blabla bla future work bla bla

1.1 · ESTABLISHED ARTIFICIAL INTELLIGENCE

The term Artificial Intelligence forms an umbrella over many different techniques that make use of machines to do some intelligent task. The most promising way to achieve AI to day is through deep neural networks. The neural networks of today are almost exclusively based on the simple perceptron neuron model. It is a fairly old idea based on a simple model on how the brain processes information. The model of the neuron that is based on has synapses just like the biological one, the synapses functions as inputs which when firing will excite the receiving neuron more or less depending on the strength of the connection. If the receiving neuron gets excited above a threshold it will fire and pass the signal downstream to another receiving neuron. Which is conceptually similar to how real neurons operate. This simple model is called a perceptron, which introduced a learning rule for a single computational neuron capable of classifying linearly separable patterns. However, to the MLP was the understanding that stacking multiple layers of these perceptron-like units could overcome these limitations by creating more complex decision boundaries. The critical breakthrough enabling the practical use of MLPs was the independent development and subsequent popularization of the backpropagation algorithm. Backpropagation provided an efficient method to calculate the gradient of the error function with respect to the network's weights, allowing for effective training of these deeper, multi-layered architectures. This combination—multiple layers of interconnected units¹, typically using non-linear activation functions, trained via backpropagation—defines the MLP, which became a foundational architecture for neural networks and paved the way for the deep learning revolution. GPT, alphafold, etc. all use these fundamentals with different variations of architectures which boils down to how many layers how large layers how dense layers and how they should be connected (attention, RNN, CNN, resnet)

1.1.1 · PROBLEMS WITH THE ESTABLISHED METHODS

It was mentioned in the introduction that the deep learning technique is inefficient compared to the brain. The reason why is not clear, from a hardware standpoint the brain simply has better hardware much more connections per area and the computation is baked into the hardware. From an algorithmic standpoint there may also be room for improvement. In order to compute with deep learning and perceptron networks we need to compute all the entries even though they might not contribute or are zero. Take an image for example, the human visual system is really good at ignoring unimportant details and we only have a tiny area of focus. Even then we don't process much unless something interesting happens like movement. In deep learning we have to process the entire image. The status quo needs

$$y = \begin{cases} 1 & \text{if } \sum_{i=0}^n x_i w_i \geq b \\ 0 & \text{if } \sum_{i=0}^n x_i w_i < b \end{cases}$$

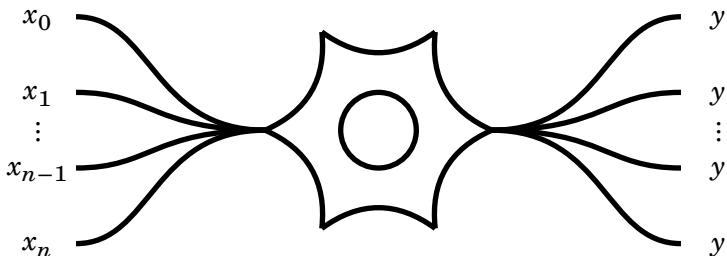


Figure 1: The perceptron—a simple model of how a neuron operates. Inputs get multiplied by weights and summed, if the sum surpasses a threshold known as the bias, the neuron fires.

Figure 1: The perceptron—a simple model of how a neuron operates. Inputs get multiplied by weights and summed, if the sum surpasses a threshold known as the bias, the neuron fires.

¹While often conceptualized in layers (e.g., layers of the neocortex), the brain's connectivity is vastly more complex than typical feedforward ANNs, featuring extensive recurrent connections, feedback loops, and long-range projections that make a simple 'unrolling' into discrete layers an oversimplification

global synchronization, every previous layer need to finish computing before the next can start, this can be hard to scale for large systems where multiple processors need to talk to each other. The same applies to backpropagation it requires freezing the entire network and separates computation and learning into two separate stages, local connections that should be independent of each other have to wait until extreme quantization models (1bit) also highlight the inefficiency

1.2 · NEUROSCIENCE 101

Although the perceptron captures common key aspects of biological neuron models A lot is left on the table. A lot of progress and new ideas has surfaced since the invention of the perceptron. The simple neuron previously thought to be simple like the perceptron model turns out to be more complex, the information encoding is also a key research topic not explored by older models. How the brain learns is also entirely different than what deep learning uses, changing the models and information encoding forces us to rethink how the learning algorithms in the brain works. Network architecture, fully asynchronous

1.2.1 · NEURON MODELS

The neuron is the fundamental building block of the brain. Comprised of an axon synapses dendrites. When presynaptic neurons fire the postsynaptic neuron increases in potential if it reaches a threshold it will itself fire. Neurons communicate with neurotransmitters such as dopamine and glutamate. There are ion channels and some calcium idk.

1.2.2 · ENCODING

It is observed that neurons fire in short bursts called spikes. Experiments show that neurons fire repeatedly. A sequence of spikes is called a spike train, and exactly how information is encoded in a spike train is a topic of hot debate in neuroscience. A popular idea is that information is encoded in the average value of spikes per time called rate encoding. Temporal encoding the brain most likely uses a combination of all. The time to first spike encoding could be understood like this it is not about the absolute timing of the neurons rather a race of which spikes come first. The first connections would excite the post-synaptic neurons first and they should inhibit the others (lateral inhibition)

1.2.3 · LEARNING

Spikes Do Not Play Nice With Gradients. While models like Spiking Neural Networks (SNNs) offer greater biological plausibility and potential advantages in processing temporal information and energy efficiency, their adoption faces significant challenges, primarily stemming from the nature of their core computational element: the discrete spike.

A cornerstone of the success of modern deep learning, particularly with Multi-Layer Perceptrons (MLPs) and related architectures, is the backpropagation algorithm. Backpropagation relies fundamentally on the network's components being differentiable; specifically, the activation functions mapping a neuron's weighted input sum to its output must have a well-defined gradient. This allows the chain rule of calculus to efficiently compute how small changes in network weights affect the final output error, enabling effective gradient-based optimization (like Stochastic Gradient Descent and its variants). These techniques have proven exceptionally powerful for training deep networks on large datasets.

However, when we transition from the continuous-valued, rate-coded signals typical of MLPs to the binary, event-based spikes used in SNNs, this differentiability is lost. The

spiking mechanism itself—where a neuron fires an all-or-none spike only when its internal state (e.g., membrane potential) crosses a threshold—is inherently discontinuous. Mathematically, this firing decision is often represented by a step function (like the Heaviside step function), whose derivative is zero almost everywhere and undefined (or infinite) at the threshold.

Consequently, standard backpropagation cannot be directly applied to SNNs. Gradients calculated using the chain rule become zero or undefined at the spiking neurons, preventing error signals from flowing backward through the network to update the weights effectively. This incompatibility represents a substantial obstacle, as it seemingly precludes the use of the highly successful and well-understood gradient-based optimization toolkit that underpins much of modern AI.

Surrogate Gradients: A popular approach involves using a “surrogate” function during the backward pass of training. While the forward pass uses the discontinuous spike generation, the backward pass replaces the step function’s derivative with a smooth, differentiable approximation (e.g., a fast sigmoid or a clipped linear function). This allows backpropagation-like algorithms (often termed “spatio-temporal backpropagation” or similar) to estimate gradients and train deep SNNs, albeit with approximations.

1.2.4 · NETWORK

However, this abstraction, while powerful, significantly simplifies the underlying neurobiology. Decades of rigorous neuroscience research reveal that brain function emerges from complex electro-chemical and molecular dynamics far richer than the simple weighted sum and static activation. While it’s crucial to discern which biological details are fundamental to computation versus those that are merely implementation specifics², moving beyond the standard MLP model is necessary to capture more sophisticated aspects of neural processing.

A primary departure lies in the nature of neural communication. Unlike the continuous-valued activations typically passed between layers in an MLP (often interpreted as representing average firing rates), biological neurons communicate primarily through discrete, stereotyped, all-or-none electrical events known as action potentials, or ‘spikes’. Information in the brain is encoded not just in the rate of these spikes (rate coding), but critically also in their precise timing, relative delays, and synchronous firing across populations (temporal coding) (Gerstner et al., 2014). For instance, the relative timing of spikes arriving at a neuron can determine its response, allowing the brain to process temporal patterns with high fidelity – a capability less naturally captured by standard MLPs. Spikes can thus be seen as event-based signals carrying rich temporal information.

Furthermore, neural systems exhibit complex dynamics beyond simple feedforward processing. Evidence suggests that cortical networks may operate near a critical state, balanced at the ‘edge of chaos,’ a regime potentially optimal for information transmission, storage capacity, and computational power. Systems like the visual cortex demonstrate this complexity, where intricate patterns of spatio-temporal spiking activity underlie feature detection, object recognition, and dynamic processing. These biologically observed principles –event-based communication, temporal coding, and complex network dynamics—motivate

²Disentangling core computational mechanisms from biological implementation details is a major ongoing challenge in neuroscience and neuromorphic engineering. Some complex molecular processes might be essential for learning or adaptation, while others might primarily serve metabolic or structural roles not directly involved in the instantaneous computation being modeled.

the exploration of Spiking Neural Networks (SNNs), which explicitly model individual spike events and their timing, offering a potentially more powerful and biologically plausible framework for computation than traditional MLPs.

1.3 · NEUROMORPHIC ENGINEERING

2 · METHODOLOGY

Say we want to detect the pattern ABC and the pattern ABD. First of all if the order does not matter set all the weights equal. If the order does matter the weights determine the order. Now if a neuron learns pattern ABC so well that it learns to fire on only AB then it can fire faster. However if a second neuron wants to learn ABD then inhibition from the AB neuron prohibits it. A solution can be that if a neuron originally learned ABC but now fires on AB but still has a strong weight on C it should remember this and if it fires on AB but then C does not arrive it should be like “oh, C did not show maybe I am wrong to fire early” eg. Decrease weights for A and B It predicts!

A second way is to have a hierarchy with bypass. So one layer detects only AB then the next layer has bypass of the first layer and the second combining AB and C or D

A second problem is how to decode order. When do we start the decreasing timer, how fast, should it be in time or in amount of spikes, what to do with phase? The phase should correct itself. The weights need to be as precise as the timing of the spikes? Or we could make the neuron sensitivity proportional to its inverse potential and add leaking

Problem of phase For rate coding phase is a non issue as we can find the instantaneous firing rate at any phase, for time to first spike encoding we need a reference signal. If the reference signal starts at time t_0 we have started the phase and if the pattern does not match up with the reference signal we could miss it. Evidence suggests that brain waves could play the role of a global reference signal. This is the fundamental trade off between the two.

A learning scheme where inputs have to occur in the same episode repeatedly two inputs that happen at the same time yields a stronger response. If the pattern is random then they would sometimes occur in the same episode and sometimes not. Two strong responses should occur in the same time frame.

If the post neuron fires then we should strengthen the weights

Footnote digression for longer patterns Longer patterns require a latched state such as neurons entering repeated firing like a state machine

An important point is to declare whether a mechanism is bio plausible. An engineer might not care whether or not a mechanism is used by the brain and is crucial to the brains function. An engineer might only care about if the mechanics works and is effective for the system that is created in the engineers vision. An engineer might just use the brain as an inspiration. Evolution although achieved remarkable feats is not guaranteed to foster up the most optimal solution, only good enough to survive to the next generation. However discussing whether or not a mechanism is bio plausible is still useful for understanding our own brain. And creating bio plausible artificial systems can contribute to more than one field.

An encoding should be fast Robust to noise Use limited resources

A neuron should accumulate change (integrate) It should fire when its threshold has been reached It should leak charge over time

Some neurons have other properties like bursting modes or continuous firing once the threshold has been reached.

Inhibition should make a neuron not fire

In a time to first spike scheme of we care about the order (the relative values since information is stored in time and order) we have to use weights and a neuron model that distinguish between inputs arriving earlier than others. I present a scheme where the first neuron that arrives starts a linear count where the slope of the counter is the weight additional inputs will increase or decrease the slope according to their weight. We can see that neurons arriving earlier will get more time to increase the counter and thus will carry a higher value. If the counter reaches a threshold the neuron will fire. The astute will notice that in this scheme the neuron will fire even for the smallest stimulus since the counter will count up a non zero value and eventually reach the threshold, to mitigate this we can simply say that if the counter is too slow the neuron will not fire we will see later that this scheme satisfies the criteria above.

The problem with this decoding is for strong stimuli we would ideally make the neuron respond immediately and fire, but it has to wait until the counter has reached the threshold to fix this we can also add the weight of the input directly to the potential while also starting a counter. Now if early strong inputs arrive they will fill up the potential and make the neuron fire almost immediately. Small inputs will take some time

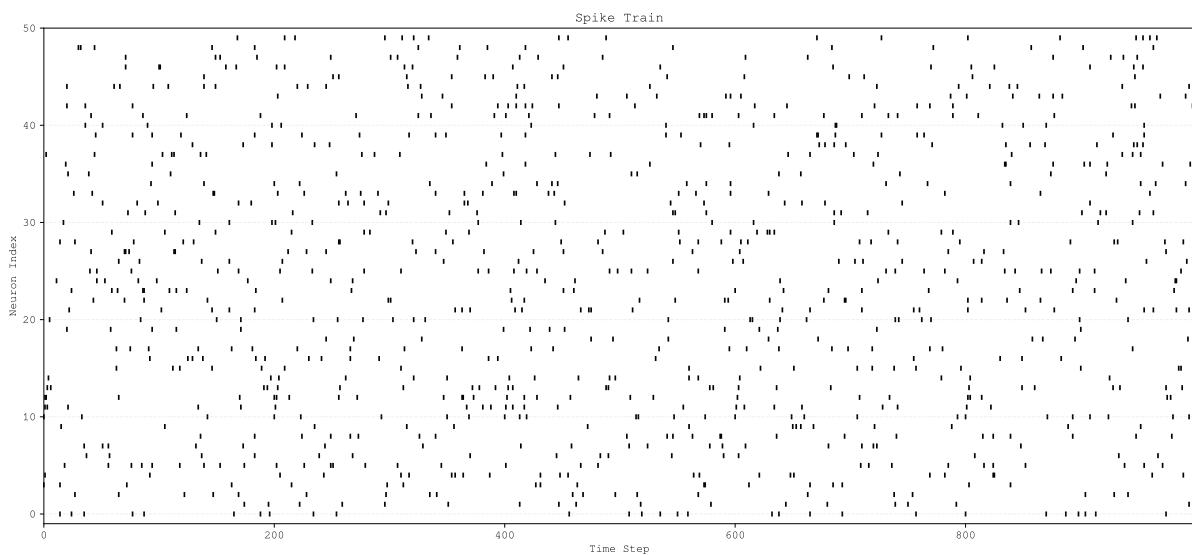


Figure 2: Spike train

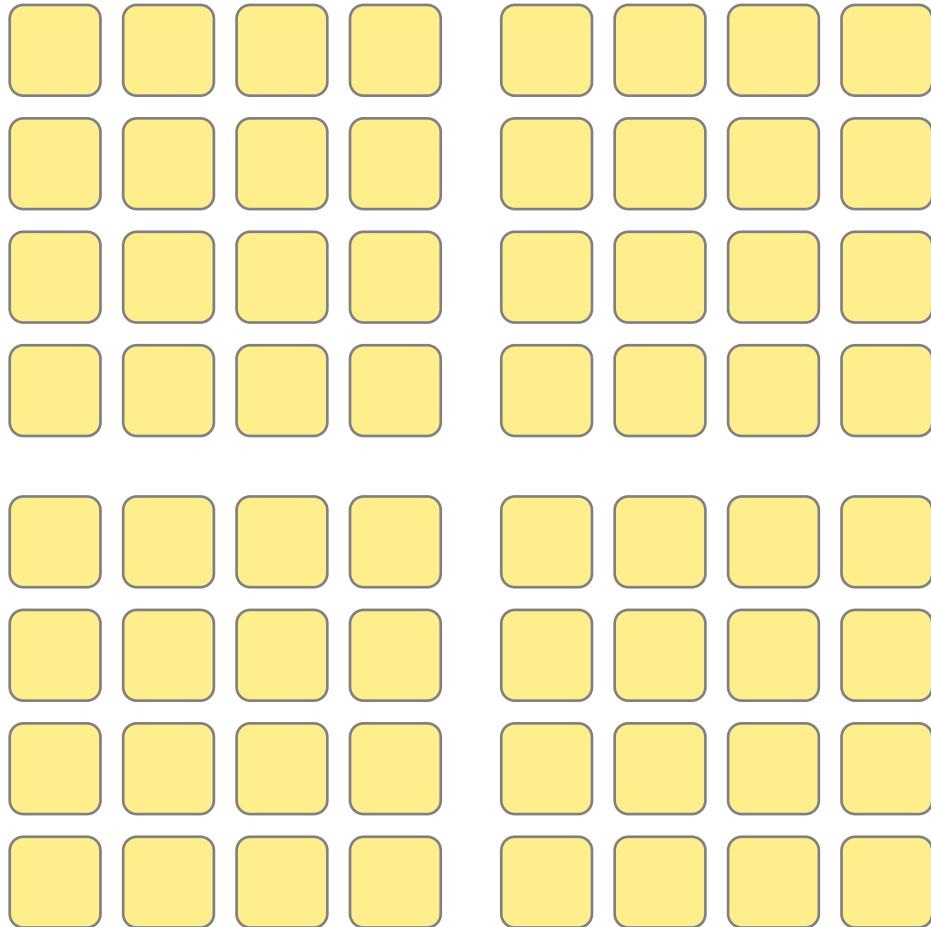


Figure 3: Proposed simplified layout of a SNN. The neurons are connected with hierarchical busses that allow for the network to be configured as a *small world network*

2.1 · NEURON MODELS

Leaky integrate and fire models seem the best bet, however complex dynamics like exponential decay and analog weights and potentials seem excessive, we might do without. Binary weights 1 for excitatory and 0 for inhibitory. Stronger weights can be modeled with multiple parallel synapses

2.2 · LEARNING

```
1 start with a collection of neurons with arbitrary connections
2 if a pre-synaptic neuron fires then
3   ↘ it has a chance to grow a synapse to a random post-synaptic neuron
4 if a post-synaptic neuron fires then
5   ↗ strengthen all connections to pre-synaptic neurons that fired before
6   ↗ weaken all connections to pre-synaptic neurons that did not fire, or fired after
① a neuron can be both pre-synaptic and post-synaptic
```

Algorithm 4: Unsupervised local learning rule for individual neurons. Based on STDP

```
1 probability of growing a synapse is inversely proportional to the amount it already
has
2 earlier firings should get a better chance to grow synapses, although this is
regulated by inhibitory action
```

Algorithm 5: Growing rules for synapses

2.3 · NETWORK

3 · RESULTS

4 · DISCUSSION

BIBLIOGRAPHY