



Christoph Rieger, Bsc.

Hierarchical architectures for spiking Winner-Take-All networks

Master's Thesis

to achieve the university degree of

Master of Science

Master's degree programme: Biomedical Engineering

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl.-Ing. Dr.techn. Robert Legenstein

Institute of Theoretical Computer Science

Head: Univ.-Prof. Dipl.-Ing. Dr.techn. Robert Legenstein

Graz, August 2023

Kurzfassung

(spikenoden)

Diese Arbeit untersucht hierarchische Architekturen von gepulsten Winner-Take-All (WTA) Netzwerken, mit dem Fokus auf der Simulation von Rückkopplungsmechanismen, welche im visuellen Kortex gefunden wurden. Solche Rückkopplungsmechanismen hängen mit Aufmerksamkeit und einheitlichen Überzeugungen, über verschiedene Areale des visuellen Kortex hinweg, zusammen. Zunächst werden die biologischen und theoretischen Konzepte von gepulsten neuronalen Netzwerken und ihre Beziehung zur Bayesschen Inferenz gezeigt. Es wird verdeutlicht, wie die hypothetisierte probabilistische Natur des Gehirns mit dem Modell eines gepulsten WTA Netzwerks, welches Bayessche Inferenz durchführt, verknüpft werden kann. Die Arbeit umfasst eine Reihe von Experimenten, die entworfen wurden, um die Reaktion des Netzwerks auf visuelle Reize zu testen. Die Reaktion des Netzwerks auf mehrdeutige Bilder wird getestet und es wird gezeigt, dass das hinzugefügte Feedback einen entscheidenden Beitrag zur Interpretation solcher Bilder leistet. Der Effekt des visuellen Kortex, illusorische Linien zu sehen, wird reproduziert, indem dem Netzwerk Feedback gegeben wird, das den visuellen Reizen widerspricht. Außerdem wird der Zusammenhang zur Bayesschen Inferenz überprüft, indem die bedingten Wahrscheinlichkeiten von Neuronen berechnet werden und ihre synaptischen Gewichte von ihnen abgeleitet werden. Um ein besseres Verständnis des Netzwerkmodells zu gewinnen, wird der Einfluss von Hyperparametern des Netzwerks analysiert. Eine unerwartete Eigenschaft der Feuerraten der Eingangs- und Priorneuronen wird ~~entdeckt~~, welche die Wahrscheinlichkeitsverteilung des Ausgangs des Netzwerks verändert. Die Wiederverwendbarkeit von Hyperparametern von Netzwerken unterschiedlicher Größen wird getestet. Es wird gezeigt, dass die Hyperparameter nicht universell auf alle Netzwerkgrößen anwendbar sind. Die Qualität des Trainings des Netzwerks wurde mit dem analytischen Optimum verglichen und es wurde gezeigt, dass das Training es nicht erreichen konnte. Diese Experimente in Verbindung zeigen, dass die Einbindung von Feedback die Fähigkeit des Netzwerks visuelle Informationen zu verarbeiten verbessert und Verhaltensmuster widerspiegelt, die in biologischen Systemen beobachtet wurden.

Abstract

This thesis explores hierarchical architectures of spiking Winner-Take-All (WTA) networks, with the focus on simulating feedback mechanisms found in the visual cortex. Such feedback mechanisms are related to attention and consistent beliefs across different areas of the visual cortex. First, the biological and theoretical concepts of spiking neural networks and their relationship to Bayesian inference are established. It is demonstrated how the hypothesized probabilistic nature of the brain can be linked to the model of a spiking WTA network that performs Bayesian inference. The thesis involves a series of experiments designed to test the network's response to visual stimuli. The response of the network to ambiguous images is tested and it is shown that the added feedback makes a crucial contribution to interpreting such images. An effect of the visual cortex of seeing illusory lines is reproduced by feeding the network feedback that contradicts the visual input. Furthermore, the model's link to Bayesian inference is verified by calculating conditional probabilities of neurons and then deriving their synaptic weights from them. To gain a better understanding of the network model the impact of the different network hyperparameters is analysed. An unexpected property of the firing frequencies of input and prior neurons is discovered, which changes the output's probability distribution of the network. The reusability of hyperparameters to networks of different sizes is tested. It is found that the hyperparameters are not universal to all network sizes. The quality of the training of the network was compared to the analytical optimum and it was found that the training could not reach it. These experiments combined reveal that incorporating feedback enhances the network's ability to process visual information, reflecting behaviours observed in biological systems.

3 Theoretical background

3.1 Bayes' theorem

Bayes' theorem describes the probability of an event to occur, depending on the prior knowledge of conditions related to the event (Joyce, 2019)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad (3.1)$$

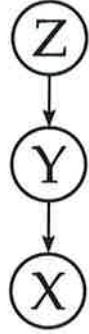
where $P(A|B)$ is the posterior probability of A, given that B is true. $P(B|A)$ is called conditional probability of B, given A being true. $P(A)$ is the prior probability, the probability of A occurring without any additional information. Finally $P(B)$ is the marginal probability of B, without any given condition. This theorem is often used for Bayesian inference, where it expresses how a belief, which is represented as probability, changes due to related evidence.

3.2 Bayesian inference

Bayesian inference is a process of data analysis to calculate the probability of a hypothesis depending on the available related evidence. As over time more and more evidence becomes available the probabilities can be updated, yielding a more sophisticated view on the hypothesis. It is given, according to Bayes' theorem, by

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)}, \quad (3.2)$$

where H represents a hypothesis and E some related evidence. $P(H|E)$ is the posterior probability, which signifies the probability of the hypothesis after the evidence was observed. $P(E|H)$ is called likelihood and it is the probability of observing the evidence, given the hypothesis. It is a measure of the compatibility of the observed evidence with the given hypothesis. $P(H)$ is the prior probability, which is the probability of the hypothesis before any evidence is considered or obtained. $P(E)$ is called marginal likelihood that represents the probability of the evidence being observed. However, it is independent of the chosen hypothesis. Thus, it is not factored in when comparing different hypotheses. Bayesian inference can be applied to the "face in shadow" example of Section 2.5. There a neuron in V_1 sees a small part of the visual field and signals if it sees an edge or not. At first it has the evidence of the observed pixels available and from it can calculate the likelihood that an edge is present. It has no prior knowledge of how probable an edge being present is, meaning that the prior probabilities for there being an edge or no edge are equal. With that likelihood and the prior probabilities it can conclude the posterior probability for the hypothesis "there is an edge" and decides that there is no edge. Later, the inferior temporal cortex determines that there is a face in the picture and feeds this information back to V_1 . This influences the prior probabilities and changes the posterior probability. Through that, the V_1 neuron starts to see the hypothesis "there is an edge" as more likely, in order to complete the contour of the face. We now define X as a binary random vector of the visual input, Y as a multinomial variable of the output of V_1 and Z as a multinomial variable of the feedback of the inferior temporal cortex. X is assumed independent of Z , when Y is given, ~~as the visual input is not influenced by the activity of the brain~~. This three node chain can be modelled as a generative probabilistic model and its visualization is given in Figure 3.1. In Section 3.2.2 of Korb and Nicholson (2011) it is shown how the chain rule can be applied to Bayesian Inference, when there are three nodes. When applying the chain rule and plugging the likelihood $P(X|Y)$, the prior $P(Y|Z)$ and the prior distribution $P(Z)$ into Equation 3.2, we get



*refer to figure
in the left.*

Figure 3.1: Generative probabilistic model of the 3 node chain, representing the interaction of the inferior temporal cortex, V1 and the visual input. It represents a multinomial mixture of 3 variables. X is a binary random vector of the visual input. Y is a multinomial variable of the output of V1 and Z is a multinomial variable of the feedback of the inferior temporal cortex. X is assumed independent of Z, when Y is given, ~~as the visual input is not influenced by the activity of the brain.~~

the posterior $P(Y|X, Z)$ as

$$\begin{aligned}
 P(Y = k|X = x, Z = j) &= \frac{P(X = x|Y = k)P(Y = k|Z = j)P(Z = j)}{\sum_{k'} P(X = x|Y = k')P(Y = k'|Z = j)P(Z = j)} \\
 &= \frac{P(X = x|Y = k)P(Y = k|Z = j)}{\sum_{k'} P(X = x|Y = k')P(Y = k'|Z = j)}.
 \end{aligned} \tag{3.3}$$

3.3 Network model

The network model used for the experiments in this thesis was taken from Nessler et al. (2013) and expanded by an additional layer to include hierarchical feedback information.

Network architecture Rectangular images in black and white were given to the network. These images had varying sizes in the experiments, determined by their width $image_{width}$ and height $image_{height}$. Each pixel of an image was

*$I(t)$ still
not mentioned
in the text.*

3 Theoretical background

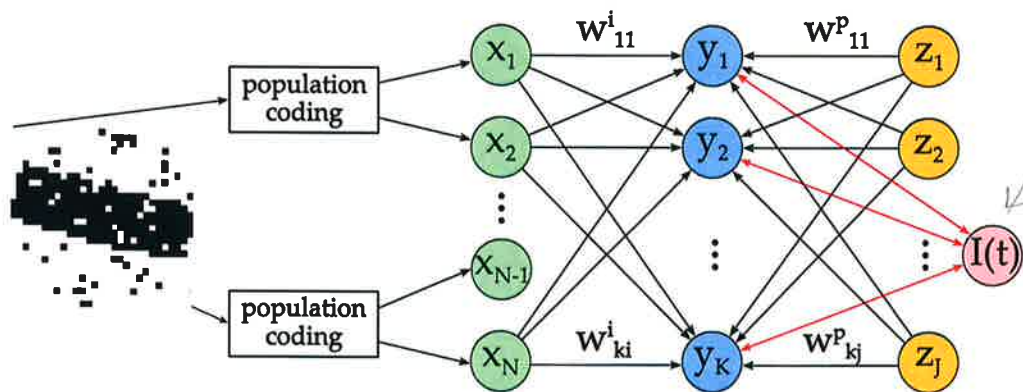


Figure 3.2: Architecture of the network.

connected to two neurons. The first of these neurons is in an active state when the pixel is black and in an inactive state otherwise. The second neuron expresses the opposite behaviour. As a consequence the network needs $image_{width} \cdot image_{height} \cdot 2$ excitatory input neurons x_1, \dots, x_N . These input neurons are fully connected to the excitatory output neurons y_1, \dots, y_K . This means that every input neuron x_i is connected to each output neuron y_k . To simulate the feedback from the inferior temporal cortex prior neurons, z_1, \dots, z_J , were added to the network. The prior neurons were also fully connected to the output neurons. The output neurons are modelled in a soft winner-takes-all (soft-WTA) circuit. The WTA behaviour was implemented via an adaptive inhibition signal. The adaptive inhibition is used to regulate the membrane potentials of the output neurons so that all of them together fire with a total firing rate $R(t) = 200\text{Hz}$ on average. Due to that, there never is a time window in which no output neuron may fire. However, it is unlikely for an output neuron to fire right after another output neuron has fired. A visualization of the network architecture can be seen in Figure 3.2.

*2.
unplan
waydown*

Neuron model As in Nessler et al. (2013) the input neurons x_1, \dots, x_N are firing according to a poisson process with an average firing rate f_{input} when active and with 0 Hz when in an inactive state. The input neurons receive binary input, which is either a black, or a white pixel of an image. Each spike an input neuron generates was modelled by a double exponential

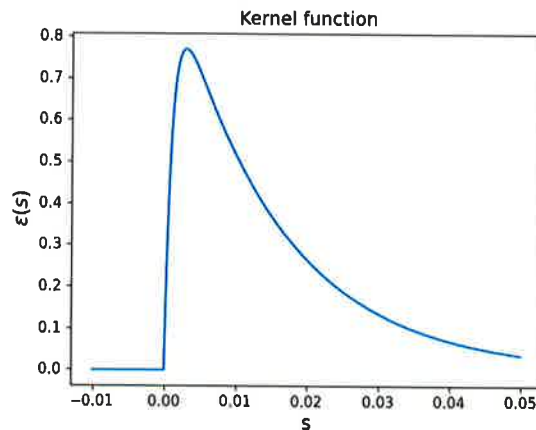


Figure 3.3: Visualization of the kernel function $\varepsilon(s)$ within the relevant time window. A double exponential kernel was used. To limit the kernel function to positive values of s a Heaviside step function was applied. The output of this function was used to calculate the unweighed membrane potential response.

kernel. As the signal is zero before the spike is generated, a Heaviside step function θ was applied to it, to limit the kernel to time ranges after the spike occurred. The Heaviside step function is given by

$$\theta(s) = \begin{cases} 1 & \text{if } s \geq 0 \\ 0 & \text{if } s < 0 \end{cases} \quad (3.4)$$

where s represents a time difference between the current simulation time and the time at which a spike occurred. Multiplying $\theta(s)$ with the double exponential kernel yields the kernel function

$$\varepsilon(s) = \theta(s) \cdot e^{-(s+\delta t)/\tau_{decay}} - e^{-(s+\delta t)/\tau_{rise}}. \quad (3.5)$$

The kernel function has a time constant for the rise of the signal $\tau_{rise} = 1ms$ and a time constant for the decay of the signal $\tau_{decay} = 15ms$. The time step size of the simulation is $\delta t = 1ms$. It had to be added to the double exponential kernel for numerical reasons, to evaluate the time at the end of the current simulation step, rather than at the beginning. A visualization of the kernel function can be seen in Figure 3.3.

space between value and unit, unit not italic. 1_{ms}
 \rightarrow correct everywhere.

3 Theoretical background

Several input neuron spikes can happen in a short time window, increasing the unweighed membrane potential response $x_i(t)$. Depending on the timing of the spikes, they contribute additively to $x_i(t)$. The unweighed membrane potential response is given by

$$x_i(t) = \sum_{t_i^{(f)}} \varepsilon(t - t_i^{(f)}), \quad (3.6)$$

with t being the current simulation time and $t_i^{(f)}$ the time at which the input spike occurred. Analogously, the unweighed membrane potential response of the prior neurons is given by

$$z_j(t) = \sum_{t_p^{(f)}} \varepsilon(t - t_p^{(f)}) \quad (3.7)$$

The membrane potential u_k of each output neuron is calculated by multiplying the unweighed membrane potential response of each input neuron times the input weight w_{ki}^I of the connection between them, plus the unweighed membrane potential response of each prior neuron times the prior weight w_{kj}^P

$$u_k(t) = \sum_{i=1}^N w_{ki}^I \cdot x_i(t) + \sum_{j=1}^J w_{kj}^P \cdot z_j(t). \quad (3.8)$$

In Nessler et al. (2013) each output neuron y_k also had an intrinsic excitability w_{k0} , which was learned for each neuron. For the experiments of this thesis it was omitted, as the different classes of input images were equally likely, thus the intrinsic excitabilities of the output neurons would all end up being equal to each other.

Nessler et al. (2013) defined that the firing probability of an output neuron y_k is exponentially proportional to its membrane potential u_k minus the received inhibition $I(t)$

$$p(y_k \text{ fires at time } t) \propto e^{u_k(t) - I(t)}. \quad (3.9)$$

3.4 Mathematical link between the spiking Winner-Take-All network model and Bayesian inference

also updated whenever an output neuron fires, in the same way as w_{ki}^I

$$\Delta w_{kj}^P = \begin{cases} \lambda \cdot (ce^{-w_{kj}^P} - 1) & \text{if } z_j \text{ fired in } [t^f - \sigma, t^f] \\ \lambda \cdot (-1) & \text{if } z_j \text{ did not fire in } [t^f - \sigma, t^f]. \end{cases} \quad (3.19)$$

3.4 Mathematical link between the spiking Winner-Take-All network model and Bayesian inference

Nessler et al. (2013) hypothesized that the ensemble of weights of a neuron can be understood as a generative model. They further claimed, that every synaptic weight, due to STDP-induced changes, converges stochastically to the log of the conditional probability that the presynaptic neuron has fired just before the postsynaptic neuron, given that the postsynaptic neuron fires. This connection is given by

$$w_{ki}^I = \log(p(x_i = 1 | y_k = 1)) \quad (3.20)$$

an will be analysed in Section 4.2. Furthermore, they claimed that in a Bayesian inference context, every input spike provides evidence for an observed variable and every output spike represents one stochastic sample from the posterior distribution over hidden causes, which are encoded in the circuit.

To explain the connection between the spiking Winner-Take-All network model and Bayesian inference it will be demonstrated, that the posterior probability of Bayesian inference given in Equation 3.3 is equal to the relative firing probability $q_k(t)$, given in Equation 3.17.

As explained in Section 3.2, the visual input, modelled by the input neurons, can be thought of as the Bayesian likelihood $P(X|Y)$ and the feedback, modelled by the prior neurons, as the Bayesian prior $P(Y|Z)$.

It is argued that $P(X = x | Y = k)$ equals $q_k^X(t)$. $q_k^X(t)$ is the conditional probability of an output neuron Y_k for the given input vector $X = x$. It can be calculated similarly to $q_k(t)$, but without the contribution of the prior

* siehe
①, ②

3 Theoretical background

neurons. Thus, the partial membrane potential, caused by the input neurons, is needed and given by

$$u_k^X(t) = \sum_{i=1}^N w_{ki}^I \cdot x_i(t). \quad (3.21)$$

Inserting $u_k^X(t)$, instead of $u_k(t)$ into Equation 3.17 yields

$$q_k^X(t) = \frac{e^{u_k^X(t)}}{\sum_{k'=1}^K e^{u_{k'}^X(t)}} = P(X = x|Y = k). \quad (3.22)$$

Similarly, it is argued that $P(Y = k|Z = j)$ equals $q_k^Z(t)$. $q_k^Z(t)$ is the conditional probability of an output neuron y_k , given the activity of a prior neuron z_j . Analogously to $u_k^X(t)$, the partial membrane potential, caused by the prior neurons, is given by

$$u_k^Z(t) = \sum_{j=1}^J w_{kj}^P \cdot z_j(t). \quad (3.23)$$

Inserting $u_k^Z(t)$, instead of $u_k(t)$, into Equation 3.17 yields

$$q_k^Z(t) = \frac{e^{u_k^Z(t)}}{\sum_{k'=1}^K e^{u_{k'}^Z(t)}} = P(Y = k|Z = j). \quad (3.24)$$

When inserting the likelihood and the prior into Equation 3.3 we get

$$\begin{aligned} P(Y = k|X = x, Z = j) &= \frac{e^{u_k^X(t)} \cdot e^{u_k^Z(t)}}{\sum_{k'=1}^K e^{u_{k'}^X(t)} \cdot e^{u_{k'}^Z(t)}} \\ &= \frac{e^{\sum_{i=1}^N w_{ki}^I \cdot x_i(t)} \cdot e^{\sum_{j=1}^J w_{kj}^P \cdot z_j(t)}}{e^{\sum_{i=1}^N w_{k'i}^I \cdot x_i(t)} \cdot e^{\sum_{j=1}^J w_{k'j}^P \cdot z_j(t)}} \\ &= \frac{e^{\sum_{i=1}^N w_{ki}^I \cdot x_i(t) + \sum_{j=1}^J w_{kj}^P \cdot z_j(t)}}{e^{\sum_{k'=1}^K e^{\sum_{i=1}^N w_{k'i}^I \cdot x_i(t) + \sum_{j=1}^J w_{k'j}^P \cdot z_j(t)}}} \end{aligned} \quad (3.25)$$

4.2 Experiment 2: Mathematical analysis and simulation of the network with 1-D images

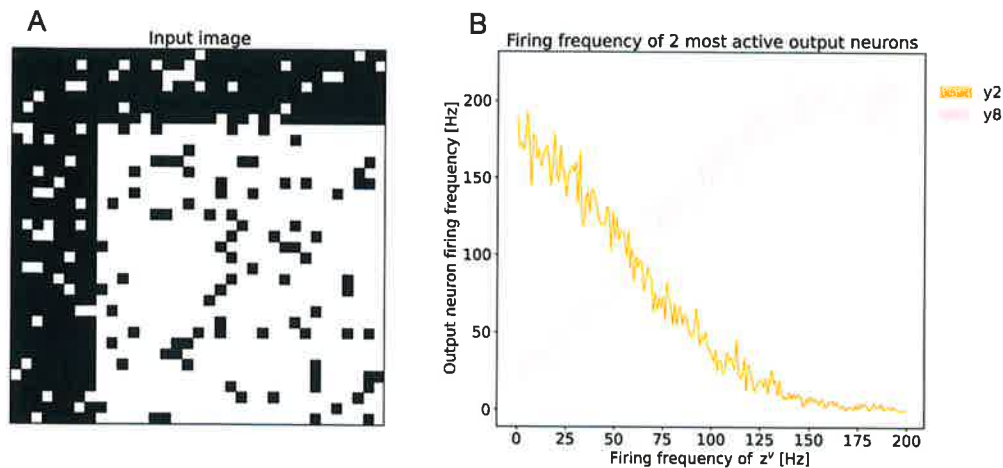


Figure 4.8: **Cross image with varying prior neuron activity.** **A** Validation cross image. **B** Firing frequency of the 2 most active output neurons, depending on the firing frequency of z^v . The output neuron firing frequencies were averaged over 10 runs to smooth the graphs.

relation

determined by utilizing the logarithmic connection between them and the corresponding conditional probabilities, as given by Equation 3.20. The conditional probabilities of the input and output neurons of the network were calculated and then used to determine the posterior probability of the network. After the simulation of the network, the distribution of the output spikes was used, to calculate the posterior probability. The posterior probabilities, of both the mathematical analysis and the simulation, were compared. It was tuned to approximate the analytical solution as closely as possible, by varying three network hyperparameters.

4.2.2 Methods

Input data The input image consisted of nine pixels in a horizontal line. Within those nine pixels, going from position 0 to 8, four output classes could be represented. Each output class had three pixels next to each other. This resulted in each output class overlapping its neighbour classes, by one pixel. Thus, the centers of the output classes were at position 1, 3, 5 and 7.

4 Experiments

Network architecture Only the amount of neurons was changed, compared to the architecture of the previous experiment. As there have to be 2 input neurons for each pixel, one neuron being active if the pixel is white and one if the pixel is black, the network had 18 input neurons. Furthermore, four prior neurons were implemented, of which only one is being active for one of the output classes at a time. Lastly, the network had four output neurons.

Mathematical Analysis The posterior probability of the network was calculated by using Equation 3.3. $P(X = x|Y = k)$ and $P(Y = k|Z = j)$ were derived, corresponding to the paradigm of the experiment. The calculation of $P(X = x|Y = k)$ was split into two parts. First, the contribution of the active input neurons was calculated by determining the matrix $P^{X|Y}$. This matrix is of size 4×9 and contains the conditional probabilities of each input neuron y_k being active, given that an output neuron x_i is active. These probabilities were calculated, by determining which input neurons are active depending on the output class and which input neurons are inactive, as dictated by the network architecture. Furthermore, the noise that was applied to the input neurons had to be taken into account. For input neurons, belonging to the output class, the conditional probability was determined as 1 and lowered by the noise level. For the input neurons outside of the 3-wide pixel block, which belongs to the output class, it was determined as 0 and raised by the noise level. According to this $P^{X|Y}$ is given by

$$P_{k,i}^{X|Y} = \begin{cases} 1 - \text{noise level} & \text{if } x_i \text{ is in the active area of } y_k, \\ 0 + \text{noise level} & \text{if } x_i \text{ is not in the active area of } y_k \end{cases} \quad (4.1)$$

After calculating all entries of $P^{X|Y}$ its rows were multiplied with the input image vector

$$P_1(X = x|Y = k) = P_{k,*}^{X|Y} \cdot x \quad (4.2)$$

resulting in a conditional probability that the input vector of the active pixels belongs to output class k . The input vector was given with entries of 1 for active pixels and with entries of 0 for inactive pixels. Next, to include the contribution of the input neurons that are spiking when a pixel is inactive, the conditional probability of the input neurons that are active for the