Bio-inspired few-shot learning with spiking neural networks

Garibaldi Pineda García
g.pineda-garcia@sussex.ac.uk
School of Engineering and Informatics,
University of Sussex
Brighton, UK

Thomas Nowotny
t.nowotny@sussex.ac.uk
School of Engineering and Informatics,
University of Sussex
Brighton, UK

ABSTRACT

State of the art neural networks typically require large datasets and long training runs to achieve high classification accuracy. In contrast, animals, including humans but also more primitive animals such as insects, have to quickly learn to adapt to their environment. To address this few-shot learning problem, we investigate a spiking neural network whose architecture is inspired by insects' mushroom body (MB). To improve the ability of our network to rapidly learn to classify novel inputs we use a learning-to-learn methodology and modify parameters for both the network architecture and the learning rule. The model was designed to be compatible with the BrainScaleS accelerated neuromorphic platform so that, in the future, it could make use of the 1000x speedup for the costly neural simulation. In this work we are investigating meta-learning algorithms and parameters which best suit our problem.

CCS CONCEPTS

• **Computing methodologies → Bio-inspired approaches**; **Machine learning algorithms**; *Transfer learning*; *Semi-supervised learning settings*; *Neural networks.*

KEYWORDS

learning-to-learn, neural networks, unsupervised learning

ACM Reference Format:
Garibaldi Pineda García and Thomas Nowotny. 2020. Bio-inspired few-shot learning with spiking neural networks. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Recent successes of deep neural networks are partly based on large annotated data sets. Obtaining this data is usually expensive and time consuming, in particular as deep neural networks require many training examples (e.g. MNIST - 6000 per class [9], original ImageNet - on average 610 per class [3], Open Images - on average 1827 per class [4]). In addition, acquiring sufficient samples for every relevant class may be impossible if some classes are rare. In contrast, animals can identify novel or rare categories with few training examples. One striking example is route learning in ants and bees where individuals have to learn a route essentially from one example. The problem of recognizing categories from small labeled data sets is generally known as few-shot learning (FSL).

One of the reasons why animals can learn rapidly is that their sensory systems and learning centres have been honed by evolution for the learning tasks they are likely to encounter during their lifetime. For instance, desert ants have specialised brain networks for route learning, and when they eventually have to learn a specific route, they can do so very rapidly. In a similar vein, the FSL problem has typically been addressed in machine learning by pre-training a neural network and then freezing parts of it (thus transferring what it has learned). The actual FSL problem is then trained starting from the pre-trained network, leading to more rapid learning. This process can be refined using so-called meta-learning procedures. Learning to learn (L2L) [16] is such a procedure. In essence, it assumes that there is a family of related tasks and an algorithm that can be trained on them. The learning of the algorithm is parametrized by meta-parameters. By assessing the learning efficiency on a subset of tasks from the family, these meta-parameters can be optimized so that afterwards the algorithm can learn new tasks of the same family rapidly and reliably. This optimization of the meta-parameters is essentially the equivalent of what evolution is for animals.

In this paper we use an L2L approach to enable FSL for the Omniglot family of classification tasks. Omniglot is a data set of different alphabets (real and fictional languages) with few samples per character [8]. It is well-suited as a benchmark for L2L problems as each of the alphabets can be considered a different "task" for the "family" of character classification. Furthermore, each individual task constitutes a FSL problem as there are only a few samples for each character.

As the learning algorithm we use a shallow spiking neural network (SNN) inspired by the architecture of the insect mushroom body [6, 7, 13] but also closely related to support vector machines (SVMs [2]) [12] and extreme learning machines (ELMs [5]). One unique feature of our system is that we use an unsupervised learning rule similar to Nowotny et al. [13]. Attaching the names of characters is then a simple one-shot process after the system has identified the available classes in an unsupervised fashion.

2 NETWORK ARCHITECTURE

The architecture of the proposed SNN is inspired by insect anatomy and the network is composed of three layers (Figure 1a): An input layer, one hidden layer and an output layer.

Input layer (image encoding) – The input is a spiking representation of an image (characters in the Omniglot alphabets). The image-to-spikes transformation is based on the circuitry of the mammalian

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
NICE '20, June 03–05, 2018, Woodstock, NY
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

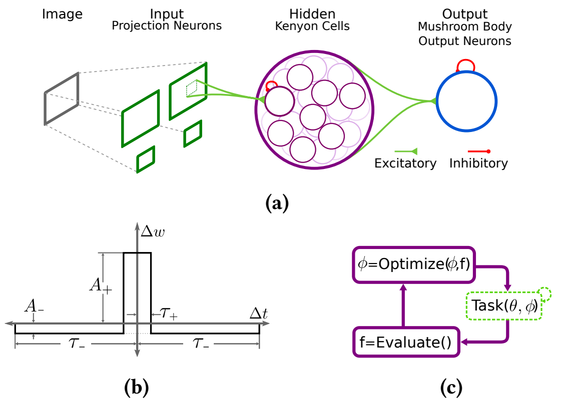

Figure 1: Network architecture and learning procedures.

retina [14]. The final representation is sparse and distributed, and it uses rank order coding-(ROC). Characters are presented to the network every 50ms and in a random order.

Hidden layer – The middle layer is inspired by the calyx of the insect mushroom body which is innervated by a large number of so-called Kenyon cells, and hence appears to implement a (non-linear) projection of the input representation into a high(er)-dimensional space. Little is known about the connectivity in the MB calyx other than that the anatomy would allow for almost any connections to be formed. Here, we split the hidden layer into groups of neurons that share an input region from which they can receive inputs. The number of groups is determined by how many receptive fields cover the input image and neurons within a group compete through lateral inhibition.

Output layer – The last layer corresponds to the mushroom body lobes and will provide the clustering (classification) of the input images. Neurons in this population compete with strong lateral inhibition to represent the inputs, forming a soft Winner-Takes-All (sWTA) circuit. Projections into the output layer are updated using a simple spike-timing-dependent plasticity (STDP) rule as illustrated in Fig. 1b.

The intuition behind the rule is that synapses are potentiated if pre- and post-synaptic spikes coincide and are depressed otherwise.

$$\Delta w_{pre,post} = \begin{cases} A_+ & |\Delta t| \leq \tau_+ \\ -A_- & \tau_+ < |\Delta t| \leq \tau_- \end{cases} \tag{1}$$

3 SIMULATION DETAILS

In our SNN, two types of neuron are used: standard leaky integrate-and-fire (LIF) for inhibitory neurons and LIF neurons with an adaptive threshold [11] for excitatory populations.

$$\tau_m \frac{dV_m(t)}{dt} = -(V_m(t) - V_{\text{rest}}) + R_m I(t) \tag{2}$$

$$\tau_m = R_m C_m \tag{3}$$

The neuron emits a spike when the membrane voltage (V_m) surpasses a threshold (V_s). For the adaptive type, the threshold voltage V_s adapts according to

$$\tau_s \frac{dV_s(t)}{dt} = -(V_s(t) - V_{s0}) + \omega_s \delta(t - t_{\text{post}}) \tag{4}$$

where τ_s is the timescale with which the threshold adaptation decays, V_{s0} is the base level for the threshold and $\omega_s = 1.8V_{s0}$ is the increment of the threshold for each spike.

4 LEARNING TO LEARN

We can think of a learning to learn procedure as a two level algorithm (Figure 1b). In this case, the first level executes a task with a SNN and the second level tunes meta-parameters. In our simulations the first level simulations were carried out with the GPU-accelerated GeNN simulator [18] and the second level is executed via the Learning-to-Learn framework developed by researchers from the Institute of Theoretical Computer Science at TU Graz and the Jülich Supercomputing Centre [15].

A key element for L2L is the evaluation of the performance of the algorithm to execute the task. Since the objective is to classify characters, a good classifier would activate an exclusive group of output neurons for inputs of each input class. To evaluate the performance of our SNN we investigate different distance measures between these groups of activated output neurons. The first two (Euclidian and Cosine distance) are well known but we have added a third one that we call overlap distance, defined as the number of neurons which spike for a single class (n) divided by the total number of activated neurons (m) ($f \propto n/m$).

Additionally, when the output population fails to spike for a character class, we add an additional cost term.

During the L2L outer loop we modify simulation parameters which can be arranged in three categories.

- Input to Hidden connectivity to obtain the optimal balance between receptive field size and random connectivity.
- Hidden to Output connectivity to optimize initialization parameters so Output neurons specialize appropriately.
- Learning rule to increase the efficiency of network learning.

5 CONCLUSIONS AND OUTLOOK

We presented a network for few-shot learning with various bio-inspired components and evaluate meta-learning algorithms (genetic algorithms, gradient ascent) and three different fitness functions. We find that the network is suitable for few-shot unsupervised clustering and identified suitable meta-learning methods and parameters. In the future, since the image encoding is done with rank-order encoding, we will investigate using ROC throughout the network. Decreasing the distance between same-class output vectors is shown to improve few-shot learning [1]; to reduce the distance we are considering to add lateral connectivity in the hidden layer in the future. New learning rules [17] could also allow to modify input connections and increase performance further.

ACKNOWLEDGMENTS

This research was funded by the European Union's Horizon 2020 research and innovation programme under grant agreement no. 785907 (HBP SGA2).

REFERENCES

[1] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. 2019. A Closer Look at Few-shot Classification. In *International Conference on Learning Representations*. https://openreview.net/forum?id=HkxLXnAcFQ

[2] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning* 20, 3 (01 Sep 1995), 273–297. DOI: http://dx.doi.org/10.1007/BF00994018
[3] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255. DOI: http://dx.doi.org/10.1109/CVPR.2009.5206848
[4] Google. Open Images v5. (????). https://storage.googleapis.com/openimages/web/index.html
[5] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. 2006. Extreme learning machine: theory and applications. *Neurocomputing* 70, 1-3 (2006), 489–501.
[6] Ramón Huerta and Thomas Nowotny. 2009. Fast and robust learning by reinforcement signals: explorations in the insect brain. *Neural computation* 21, 8 (2009), 2123–2151.
[7] Ramón Huerta, Thomas Nowotny, Marta García-Sanchez, Henry DI Abarbanel, and Mikhail I Rabinovich. 2004. Learning classification in the olfactory system of insects. *Neural computation* 16, 8 (2004), 1601–1640.
[8] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science* 350, 6266 (2015), 1332–1338. DOI: http://dx.doi.org/10.1126/science.aab3050
[9] Y. LeCun and C. Cortes. The MNIST dataset of handwritten digits. (????). http://yann.lecun.com/exdb/mnist/
[10] Qian Liu, Garibaldi Pineda García, Evangelos Stromatias, Teresa Serrano-Gotarredona, and Steve B. Furber. 2016. Benchmarking Spike-Based Visual Recognition: A Dataset and Evaluation. *Frontiers in Neuroscience* 10 (2016), 496. DOI: http://dx.doi.org/10.3389/fnins.2016.00496
[11] TimothÃľe Masquelier and Saeed R. Kheradpisheh. 2018. Optimal Localist and Distributed Coding of Spatiotemporal Spike Patterns Through STDP and Coincidence Detection. *Frontiers in Computational Neuroscience* 12 (2018), 74. DOI: http://dx.doi.org/10.3389/fncom.2018.00074
[12] T. Nowotny and R. Huerta. 2012. On the equivalence of Hebbian learning and the SVM formalism. In *2012 46th Annual Conference on Information Sciences and Systems (CISS)*. 1–4. DOI: http://dx.doi.org/10.1109/CISS.2012.6310939
[13] Thomas Nowotny, Ramón Huerta, Henry D. I. Abarbanel, and Mikhail I. Rabinovich. 2005. Self-organization in the olfactory system: one shot odor recognition in insects. *Biological Cybernetics* 93, 6 (01 Dec 2005), 436–446. DOI: http://dx.doi.org/10.1007/s00422-005-0019-7
[14] B. Sen Bhattacharya and S. B. Furber. 2010. Biologically Inspired Means for Rank-Order Encoding Images: A Quantitative Analysis. *IEEE Transactions on Neural Networks* 21, 7 (July 2010), 1087–1099. DOI: http://dx.doi.org/10.1109/TNN.2010.2048339
[15] Anand Subramoney, Sandra Diaz-Pier, Arjun Rao, Franz Scherr, Darjan Salaj, Thomas Bohnstingl, Jakob Jordan, Nikolaus Kopp, Daniel Hackhofer, Sinisa Stekovic, and et al. 2019. Learning to learn framework. (Mar 2019). DOI: http://dx.doi.org/10.5281/zenodo.2590760
[16] Sebastian Thrun and L. Pratt. 1998. Learning to Learn: Introduction and Overview. In *Learning To Learn*, S. Thrun and L. Pratt (Eds.). Kluwer Academic Publishers.
[17] James CR Whittington and Rafal Bogacz. 2019. Theories of error back-propagation in the brain. *Trends in cognitive sciences* (2019).
[18] Esin Yavuz, James Turner, and Thomas Nowotny. 2016. GeNN: a code generation framework for accelerated brain simulations. *Scientific reports* 6 (2016), 18854.

A ONLINE RESOURCES

Image conversion https://github.com/NEvision/NE15 [10]
Experiments https://github.com/chanokin/l2l-omniglot
GeNN simulator https://github.com/genn-team
Learning to learn framework https://github.com/IGITUGraz/L2L