

Ensembling SNNs with STDP learning on base of rate stabilization for image classification. Alexander Sboev^{1,2}

(Sboev_AG@nrcki.ru), Alex Serenko^{*1}, Roman Rybka¹, Danila Vlasov^{1,2}.

¹National Research Centre «Kurchatov Institute», Moscow, Russia,

²Moscow Engineering Physics Institute (MEPhI), Moscow, Russia.



Alexander Sboev



Alex Serenko



Roman Rybka



Danila Vlasov

ABSTRACT

In spite of a number of existing spiking neural network models for image classification, it still remains relevant from both methodological and practical points of view to develop a model as simple as possible, while at the same time applicable to classification tasks with various types of data, be it real vectors or images.

Our previous work [1] proposed a simple spiking network with Spike-Timing-Dependent-Plasticity (STDP) learning for solving real-vector classification tasks.

In this paper, that method is extended to image recognition tasks and enhanced by aggregating neurons into ensembles.

The network comprises one layer of neurons with STDP-plastic inputs receiving pixels of input images encoded with spiking rates.

This work considers two approaches for aggregating neurons' output activities within an ensemble: by averaging their output spiking rates (i. e. averaging outputs before decoding spiking rates into class labels) and by voting with decoded class labels.

Ensembles aggregated by output frequencies are shown to achieve a significant accuracy increase up to 95% (by F1-score) for the Optdigits handwritten digit dataset, and is comparable with conventional machine learning approaches.

INTRODUCTION

Of the methods for applying spiking neural networks (SNNs) to machine learning tasks, the majority mimic the structures of formal networks. Recently, however, especial relevance is shifting towards the problem of creating methods that would be based on Spike-Timing-Dependent Plasticity as a mechanism underlying learning, since STDP, being local, has some prospective potential of being hardware-implemented in energy-efficient memristive devices. A general goal of ours is the creation of network topologies simple and as generic as possible, utilizing characteristic phenomena of STDP.

Our previous work [1] proposed a learning algorithm based on the effect of stabilizing the mean output spiking rate of a neuron under STDP. In this algorithm, each neuron is trained on its own class and then is to distinguish its own class from the other classes. An advantage of the algorithm is its robustness in a wide range of neuron and STDP parameters, owing to the robustness of the underlying effect of spiking rate stabilization. In spite of using extremely few neurons, just one neuron per class, the algorithm has proved its efficiency on the benchmark classification tasks of Fisher's Iris and Wisconsin breast cancer.

The simplicity of the model, along with the low number of neurons used in the algorithm, leaves the potential for improving its accuracy by employing more neurons, while still keeping the numbers of adjustable parameters and neurons relatively low compared to other approaches.

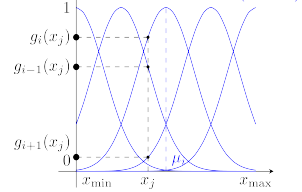
The aim of this work is to study on a more complex dataset, what accuracy increase an ensemble of several networks with rate-stabilisation-based STDP learning achieve in comparison with the accuracy of a single network.

METHODS

1. Rate encoding of input data

Vectors $\vec{x}$ of input data are preprocessed with Gaussian receptive fields:

$$g_i(x_j) = \exp\left(-\frac{(x_j - \mu_i)^2}{\sigma^2}\right)$$



and then encoded by mean frequencies $\vec{v}$ of Poisson input spike sequences:

$$v_{in}^i = v_{low} + v_{high} \cdot x_i$$

$$\begin{pmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{pmatrix} \rightarrow \begin{pmatrix} v_{in}^1 \\ \vdots \\ v_{in}^i \\ \vdots \\ v_{in}^n \end{pmatrix}$$

2. Spike-Timing-Dependent Plasticity

The learning algorithm we use is based on additive STDP with restricted symmetric nearest-neighbour spike pairing scheme.

Nearest pairs (see Fig. 1) of spikes incoming to i -th synapse (presynaptic) and outgoing (postsynaptic) at times t_{pre} and t_{post} correspondingly cause a change of the synaptic weight w_i to $w_i + \Delta w$:

$$\Delta w = \begin{cases} W_+ e^{-\frac{t_{post} - t_{pre}}{\tau_+}} & \text{if } t_{post} > t_{pre} \\ -W_- e^{-\frac{t_{pre} - t_{post}}{\tau_-}} & \text{if } t_{post} < t_{pre} \end{cases}$$

An additional constraint keeps w_i between 0 and $w_{max} = 1$:

$$\begin{cases} \text{if } w + \Delta w > w_{max}, & w \leftarrow w_{max}; \\ \text{if } w + \Delta w < 0, & w \leftarrow 0. \end{cases}$$

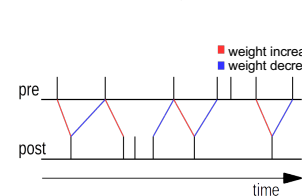


Figure 1. The restricted symmetric nearest-neighbour spike pairing scheme.

3. Leaky Integrate-and-Fire neuron model

The membrane potential V obeys

$$C_m \frac{dV}{dt} = \frac{V_{rest} - V(t)}{\tau_m} + I_{syn}(t)$$

The postsynaptic current is of exponential form:

$$I_{syn}(t) = \sum_i w_i \sum_{pre-spikes s} \frac{q_{syn}}{t_{syn}} e^{-\frac{t-s}{t_{syn}}} \Theta(t-s)$$

As soon as $V \geq V_{th} = -54$ mV, $V \leftarrow V_{rest} = -70$ mV, the neuron fires a spike, and during the refractory period $\tau_{ref} = 3$ ms the neuron is insensitive to its input.

4. Neural network learning algorithm

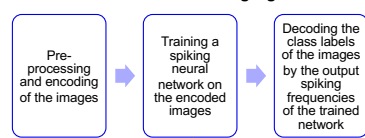


Figure 2. The algorithm for applying a single spiking network to a classification tasks.

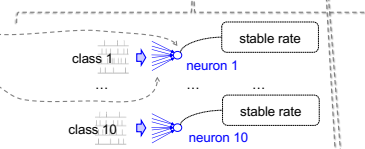


Figure 3. The scheme of the network consists of 10 neurons (by the number of classes in the Digits dataset), not connected to each other. At the training stage, each neuron receives vectors of its own class at its STDP-plastic input synapses.

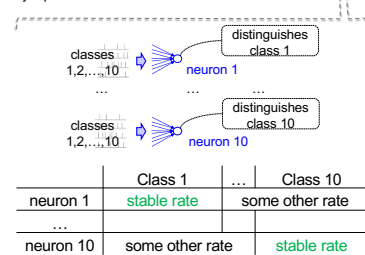


Figure 4. At the testing stage, class labels are decoded by the output frequencies of the neurons in response to an image, using the decoding rule based on the rate stabilization effect: the image is assigned to the class whose neuron spikes in response to it with the frequency the closest to the mean frequency of this neuron on its own class' training set vectors.

5. Ensembling technique

Two ways of ensembling are tried:

- the networks within an ensemble have all parameters identical as presented in Table 1 (the networks thus differ only by different random initial weights);
- each network has its own value of neuron membrane capacity C_m .

Aggregating outputs of trained networks within an ensemble is performed in two ways:

- by averaging output rates (Fig. 5): output spiking rates of neurons corresponding to the same classes are summed, and the resulting total rates are passed to the decoding procedure depicted in Fig. 4 as if it were outputs of a single network;

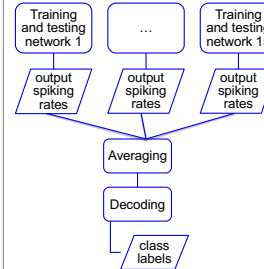


Figure 5. Aggregating networks in an ensemble by averaging output rates.

- by averaging class labels (Fig. 6): the output rates of each network are decoded into class labels, and the resulting label of an image is the label outputted by the majority of the networks in the ensemble.

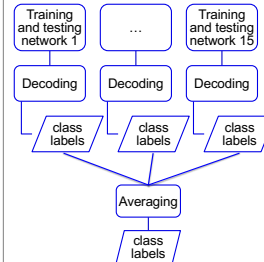


Figure 6. Aggregating networks by averaging class labels.

RESULTS

A single SNN with the optimal parameters from Table 1 achieves the F1-score of 85% to 88%, the deviation over cross-validation folds and over 15 independent runs being roughly the same.

Ensembles of up to 15 identical networks do not improve accuracy compared to a single network, as shown by the "Identical networks" curves in Fig. 7. However, ensembling networks with different C_m values increases the accuracy to **95 to 96%**, which is on par with the 98% $\pm 1\%$ result of the conventional Gradient Boosting classifier from the scikit-learn library, as well as with 98% reported by the dataset maintainers, achieved by the k -nearest-neighbour algorithm on the full version of the dataset.

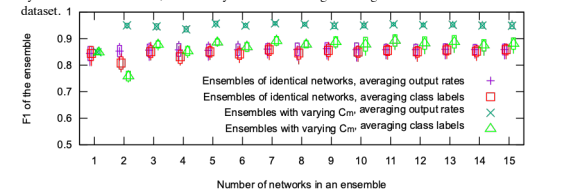


Figure 7. The F1-macro score of an ensemble in dependence of the number of networks in it, for the two ways of aggregating the outputs of networks within an ensemble (described in Fig. 5 and 6).

Notably, this increase is achieved by aggregating outputs of several networks, but not because of incorporating some very high-performing network, because the accuracy of each individual network remains roughly the same (see Fig. 8). Note, however, that the accuracy increase is only observed when aggregating network outputs is performed by *averaging output rates*, i. e. as in Fig. 6.

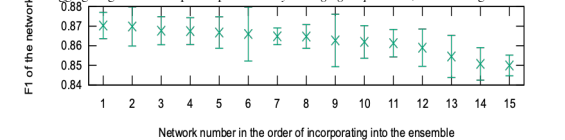


Figure 8. The F1-macro score of individual networks in the order in which they were incorporated into the ensemble. The deviation ranges show the minimum-to-maximum range of the score over cross-validation folds.

CONCLUSION

Aggregating networks with different values of neuron membrane capacity allows to improve the classification performance by about 9% by the F1-macro score.

The results obtained open a prospective possibility for simple spiking networks to solve not only vector classification tasks as shown earlier [1], but also image recognition tasks, and pave the way for solving more complex tasks.

Thus, this study is a step in the direction of developing spiking networks based on local STDP plasticity that would be implementable in memristive devices.

ACKNOWLEDGMENTS & REFERENCES

Numerical simulations have been conducted using the Neural Simulation Tool (NEST) library.

This work has been carried out using computing resources of the federal collective usage center Complex for Simulation and Data Processing for Mega-science Facilities at NRC "Kurchatov Institute", <http://ckp.nrcki.ru/>.

Detailed description of the learning algorithm based on the output rate stabilization effect of STDP: [1] A. Sboev, A. Serenko, R. Rybka, D. Vlasov: *Solving a classification task by spiking neural network with STDP based on rate and temporal input encoding*. Mathematical Methods in the Applied Sciences (2020). DOI 10.1002/mma.6241.