Classification using Spiking Neural Networks

Tharsini, V. and Charles, E. Y. A.

Department of Computer Science, University of Jaffna, Sri Lanka.

tharsini84@gmail.com, charles.ey@jfn.ac.lk

Introduction

Spiking Neural Networks (SNNs) are the third generation of neural networks composed of spiking neurons as processing elements. They derive their strength and performance from an accurate modeling of synaptic interactions between neurons, taking into account the time of spike emission. SNNs are capable to beat the computational power of neural networks made of threshold or sigmoidal units.

Objective

In this work, a novel supervised delay adaptation spike time based SNN learning model for classification that encodes the information in the connection delays is developed.

Background

The Neural Network

Neural Network is an information processing paradigm that mimics the operations of brain.

The Spiking Neural Network

The spiking neuron is an artificial neuron which is a mathematical model of a biological neuron. In addition to a weight in the ANN, a connection in the SNN could have a delay mechanism which postpones the arrival of an input spike at the other end of the connection as shown in Figure 1.

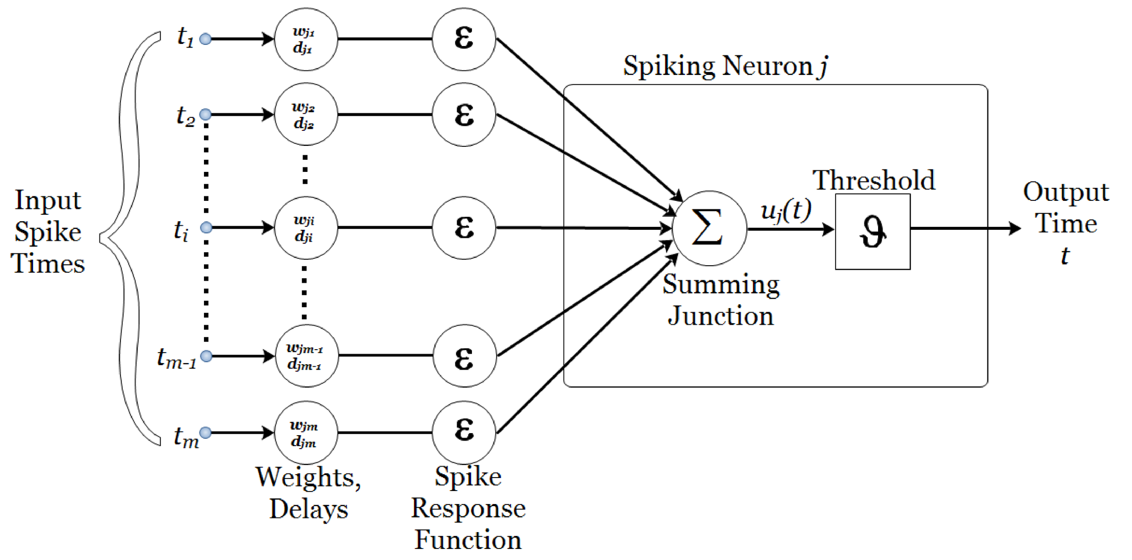
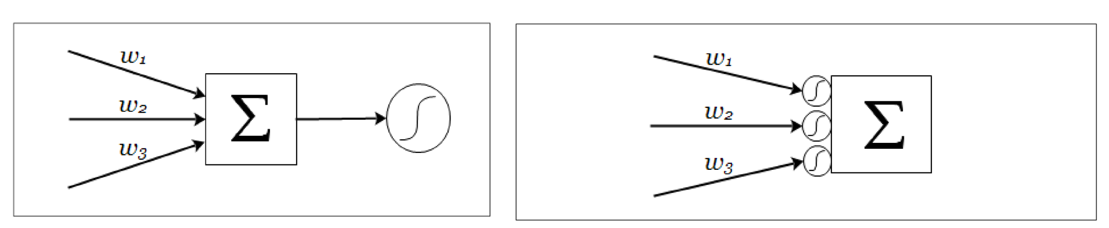


Figure 1: The Structure of Spiking Neural Network.

Comparison of Traditional and Spiking Model



(a): Standard Model. **(b):** Alternative Model.

Figure 2: Location of essential nonlinearity [2].

The Spike Response Model

The Spike Response Model(SRM) [3] is much popular SNN model which is used in this study. The state of neuron j at time t is specified by its potential $u_j(t)$ which can be computed using equation (1). At the time the potential reaches the threshold ϑ it fires a spike.

$$u_j(t) = \sum_i w_{ji}.\varepsilon(t - t_i - d_{ji}) \quad \text{.................. (1)}$$

$$\varepsilon(t) = (t/\tau).e^{(1-t/\tau)} \quad \text{.................. (2)}$$

where w_{ji} and d_{ji} are the connection weight and delay between neuron i *and* j respectively, t_i firing time of neuron i, $\varepsilon(t)$ is the spike response function which specifies the effect of an input spike at time t, given by the equation (2). τ is a time constant.

Coincidence Detection Property of Spiking Neurons

A delay adaptation method is used to achieve the objective of making certain input spikes to coincide at a given neuron. A simple scenario is illustrated in Figure 3. Since the inputs are nearly coinciding at the top output neuron there is a better chance for that neuron to generate an output spike than the bottom one.

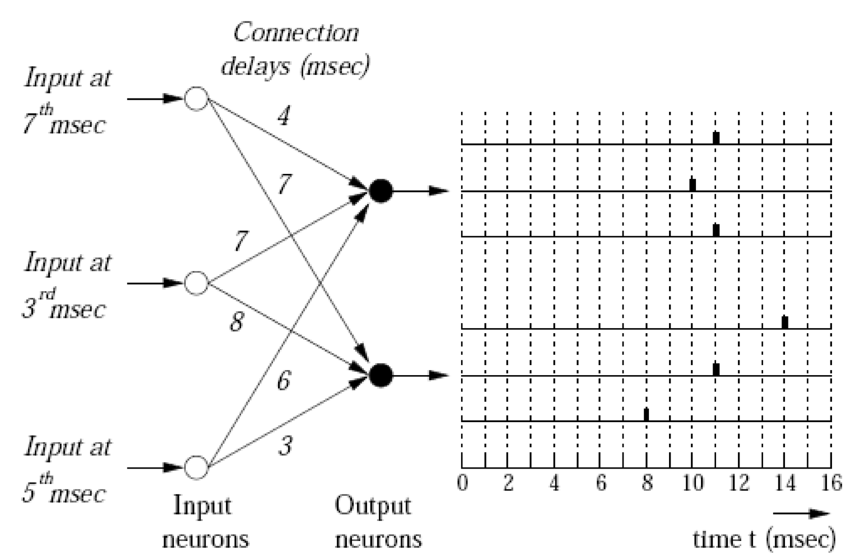


Figure 3: Coincidence Detection

Supervised delay adaptation learning model

A supervised delay adaptation learning model has been developed for training spiking neural networks where spiking neurons are realized as coincidence detectors. There are two ways in which the delay adaptation can be implemented computationally: Delay Selection, Delay Shift proposed in [4]. In this study, the Hebbian-based shifting rule is employed for this purpose. The rule is given in equations (3) & (4).

$$\delta d_{ji} = \delta t\left(\frac{e^{-\delta t^2/\tau^2_{stdp}}}{\tau_{stdp}}\right) - b \ \cdots \ (3) \qquad \delta t = t'_j - (t_i + d_{ji}) - s \ \cdots \ (4)$$

δd_{ji} - change in delay for connection between neurons j and i.

τ_{stdp} - time constant for learning rule, b - (+)ve bias, s - (+)ve term.

δt - time difference between the delayed input spike and output spike.

t'_j - firing time of output neuron j.

Proposed Methods

1) *Error Calculation:* Error term is calculated at the output layer based on the coincidence detection property of spiking neurons. The output will be high when that particular spiking neuron receives coinciding input signals. The latency between the time of input and the firings of spiking neurons can be calculated using the gradient of the graph as shown in Figure 4, which is given by the equation (5). We assume that response function rises approximately linearly or descends approximately linearly.

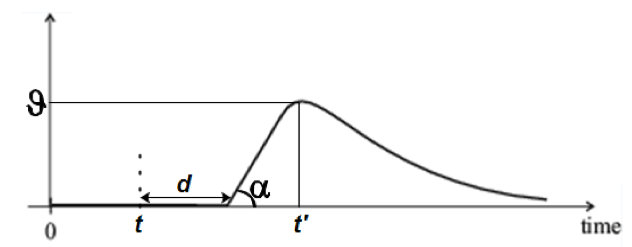


Figure 4: The latency in a single spike

$$\tan\alpha = \frac{\vartheta}{t' - (t+d)} \quad \text{.... (5)}$$

Neuron potential at coinciding time t'

When all pre-synaptic signals are coinciding, the output of spiking neuron is possibly earliest. If the coinciding time is t', then the internal state variable of spiking neuron at that time is given by the equation (6) which is equal to the threshold value given by equation (7).

$$u_j(t') = \sum_{i=1}^{m} w_{ji}\varepsilon(t') \quad \text{.... (6)} \quad \Longrightarrow \quad \varepsilon(t')\sum_{i=1}^{m} w_{ji} = \vartheta \quad \text{.... (7)}$$

m is the number of input neurons

But if the spikes are fired asynchronously, then the firing time t's will be different values. Those firing times can be calculated by simulation in a certain time window. Assume that t'' is the firing times of asynchronous inputs. Then the gradient of those signals is given by the equation (8).

$$\tan\hat{\alpha} = \frac{\vartheta}{t'' - (t+d)} \quad \text{....... (8)} \qquad error = \tan\alpha - \tan\hat{\alpha} \quad \text{....... (9)}$$

Proposed learning rule : $\delta d_{ji} = error \cdot \delta t\left(\frac{e^{-\delta t^2/\tau^2_{stdp}}}{\tau_{stdp}}\right) - b$

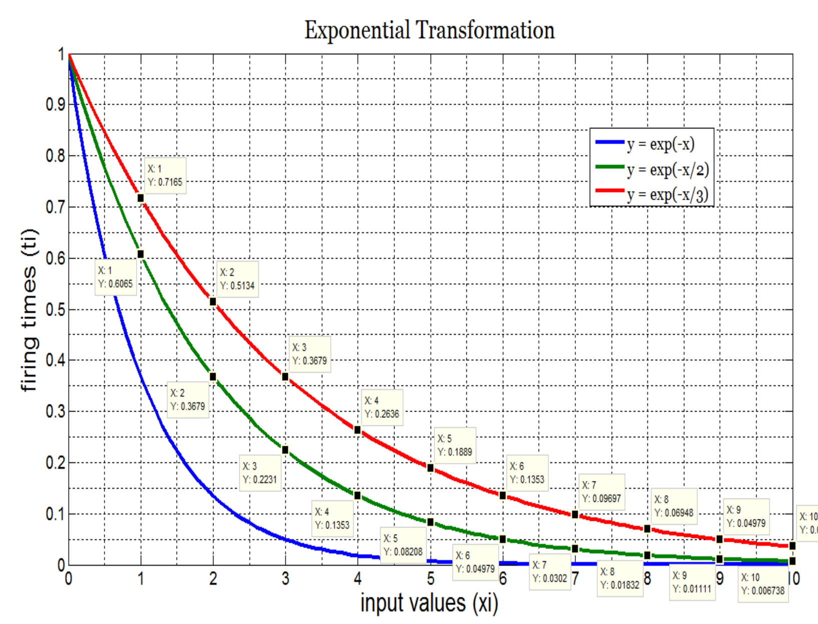
2) *Coding of the input patterns :*

Earlier method : $T' = T_o - x$ where T_o is a constant

- early spike for high input value, late spike for low input value.

Proposed method : $f(x) = \exp^{(-x/\alpha)}$ α is a (+)ve constant, $0 < \alpha < x$.

Figure 5: Exponential Transformation. It shows when α =1, 2, 3. Since α = 3 shows greater discrimination between the input values and fills the required time interval, α is chosen as three for input coding.



Dataset

Table 1: UCI benchmark dataset

Dataset	Number of				
	attributes	classes	samples	training	testing
Breast-Cancer	9	2	699	466	233

The training and testing were carried out using 3-fold cross validation. The parameters were set as [5].

Results

Table 2: Experimental Results

Fold	SNN		MLP	
	Training (%)	Testing(%)	Training (%)	Testing(%)
1	81.97	93.14	95.9	92.7
2	94.64	89.7	93.3	96.13
3	93.99	90.56	80.04	80.25
Average	**90.2**	**91.13**	**89.75**	**89.69**

- The model achieved considerably high accuracy with 17 training epochs. The results were compared with MLP which obtained much lower average accuracy for training and testing in 17 epochs.

ROC curves

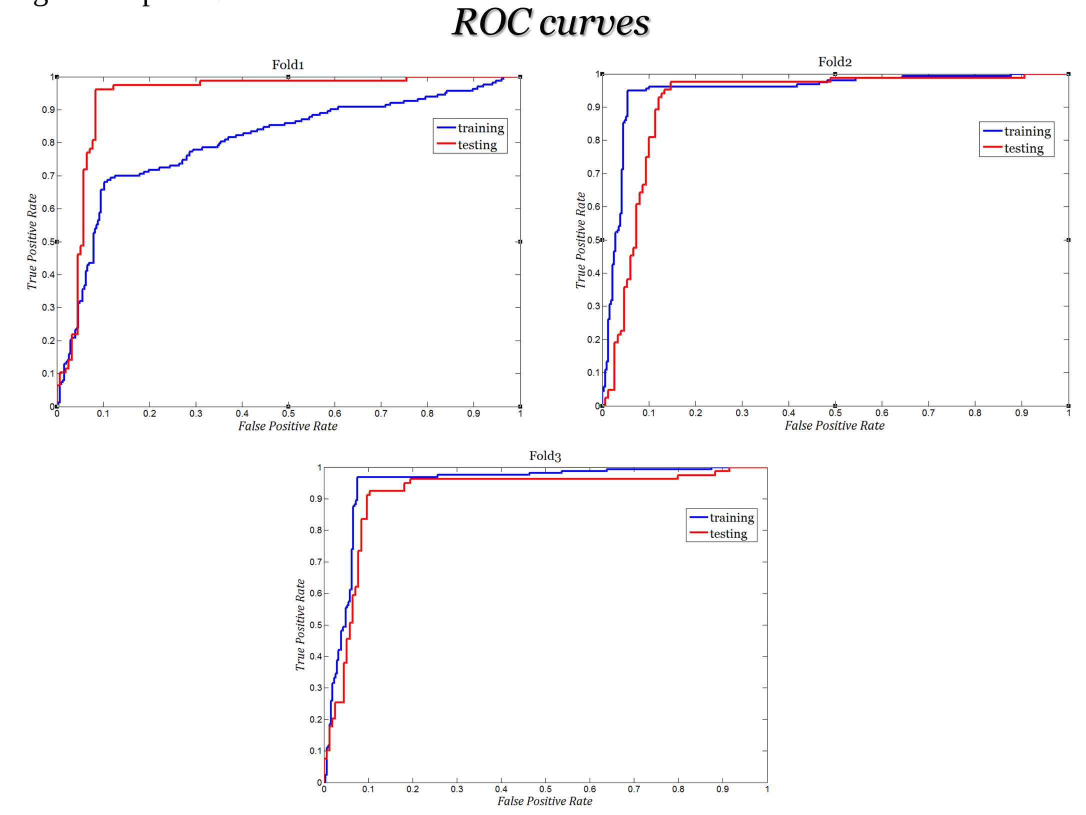


Figure 6: The ROC curves for three folds

Table 3: Testing results shown as area under the ROC curve(AUC).

Fold	Training	Testing
1	0.8030	0.9363
2	0.9493	0.9155
3	0.9428	0.9107
Average	**0.8984**	**0.9208**

Discussion & Conclusion

- The proposed model achieved considerably high accuracy with low number of epochs compared to MLP.
- Even though the performance of this model is good, the accuracy is expected to be improved by fine tuning the input coding and error estimating components.

References

[1]. D.J.Newman and S.Hettich and C.L.Blake and C.J.Merz, Wisconsin Breast Cancer Database, UCI repository of machine learning databases,University of California, Irvine, Dept.of Information and Computer Sciences, 1991, http://www.ics.uci.edu/ mlearn/MLRepository.htm, Online; accessed 05-May-2011.

[2]. A.M.Zador, The basic unit of computation, Nature America, volume 3, 2000.

[3]. W.Maass and C.M.Bishop, Pulsed Neural Networks, The MIT Press, Cambridge ,edition 1, 2001.

[4]. T.Tversky and R.Miikkulainen, Modeling Directional Selectivity Using Self-Organizing Delay-Adaptation Maps, Neurocomputing, volume 44-46, pages 679-684, 2002.

[5] D.T.Pham, M.S.Packianather, and E.Y.A.Charles. A Novel Self-Organised Learning Model with Temporal Coding for Spiking neural networks. MEC, Cardiff university, Cardiff CF24 3AA, UK.