Temporal Pattern Recognition Using a Spiking Neural Network with Delays

Jeong-Woo Sohn[1], Byoung-Tak Zhang[1,2] and Bong-Kiun Kaang[1,3]
{jwsohn, btzhang}@scai.snu.ac.kr, kaang@plaza.snu.ac.kr
[1]Interdisciplinary Program in Cognitive Science
[2]Department of Computer Engineering
[3]Institute of Molecular Biology and Genetics
Seoul National University, Seoul, 151-742, Korea

Abstract

Spiking neural networks have been shown to have powerful computation capability. But most results have been restricted to theoretical work. In this paper, we apply a spiking neural network to a time-series prediction problem, i.e. laser amplitude fluctuation data. We formulate the time-series problem as a spatio-temporal pattern recognition problem and present a learning method in which spatio-temporal patterns are recorded as synaptic delays. Experimental results show that the presented model is useful for temporal pattern recognition

1. Introduction

Recently, spiking neural networks (SNNs) have been studied by many researchers [1]. This trend is due to several biological data indicating that spikes play an important role in biological information processing, which is in contrast with a view that the mean firing rate is important in biological information processing. On the other hand, the effort to apply SNN to computation has also increased. Several results show that SNNs are very powerful in computation: It appears that the model has at least the same computational power as neural networks like multi-layer perceptrons and sigmoidal neural networks of a similar size. Furthermore, there are concrete functions which require for their computation significantly fewer neurons in SNN than conventional neural network models [1].

The encoding type of SNNs is temporal, so SNN models fits to the analysis of time-structured data. In addition to this assumption, Hopfield presented a model for radial basis functions (RBFs) with delays [2]. The main idea is that the RBF neurons encode a particular input spike pattern in the delays that connects input neurons to RBF neurons. An input pattern being similar to the encoded pattern, the RBF neuron fires because the delays even out the differences of the firing times of the input neurons. This approach is supported from both computational and biological aspects [3].

All of this could make SNNs a good temporal pattern recognizer. But as far as we know, there are few applications of SNNs to real-life data. This is due to the lack of effective learning algorithms for SNNs. In this study, we apply a spiking neural network model to time series prediction. We formulate time-series prediction as a spatio-temporal recognition problem. The output value to predict is the result of the past history. We separate the previous information into spatial and temporal information. The temporal information is encoded into synaptic delays of network. The spatial information is encoded into the neuron configuration.

This paper is organized as follows. Section 2 describes the network architecture. Section 3 formulates the time-series prediction addressed in this paper. Section 4 represents the learning method. Section 5 and 6 provide experimental results and concluding remarks.

2. Recognition mechanism and architecture

Our neuronal network is based on the spike response model presented in [4]. If the sum of PSP (Post Synaptic Potential) reaches the threshold, the neuron fires. In our model, there are no synaptic weights. Instead, there are synaptic delays. The delay is given by the difference between the presynaptic firing time and the time the postsynaptic potential starts rising. The presynaptic potential is considered as a unit impulse function.

The sum of the PSP of output neuron v_j at time t $P_{v_j}(t)$ is given as

$$P_{v_j}(t)=\sum_{i=1}^{n}\varepsilon\left(t-\left(t_{u_i}+d_{ji}\right)\right) \quad (1)$$

where ε is the activation function, n is the number of input neurons, t_{u_i} is the firing time of an input neuron u_i and d_{ji} is the delay between neurons v_j and u_i. In our work, the activation function is given as

0-7803-5529-6/99/$10.00 ©1999 IEEE

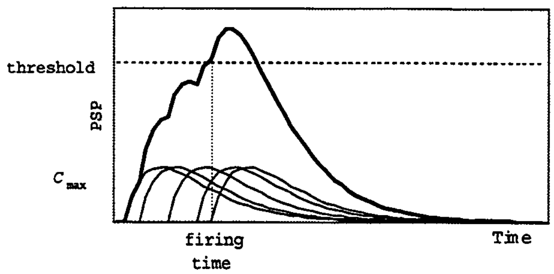

Figure 1. Recognition mechanism. Five incident signals are added temporally. The sum of postsynpatic potential reaches the threshold and the output neuron fires.

$$\varepsilon(t)=\begin{cases} C_{\max}\dfrac{t}{t_p}\exp\left(1-\dfrac{t}{t_p}\right) & \text{for } t>0 \\ 0 & \text{for } t\le 0 \end{cases} \tag{2}$$

where $C_{\max}$ is a maximum constant and t_p is a time constant. This equation is generally called the alpha equation.

Figure 1 shows the recognition mechanism. As can be seen, the size of the PSP depends on the time of arrival of action potentials to the output neuron. The less the time differences of arrivals are, the larger the PSP is attained and the earlier the firing time occurs. Therefore, the PSP of an output neuron is a temporal sum of input action potentials. Input neurons receive the time-varying signal and the signal reaches output neurons after specific delay. In the output neuron, the signal is temporally hold in shape of alpha function. If another signal arrives at the output neuron before the potential is inactivated, the PSP sum of the output neuron increases. If the sum of $P_{v_j}(t)$ reaches the threshold Θ, the output neuron v_j fires.

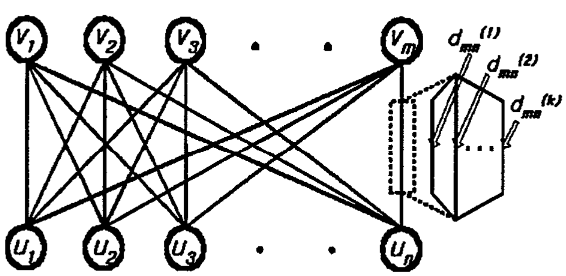

Figure 2. The network structure.

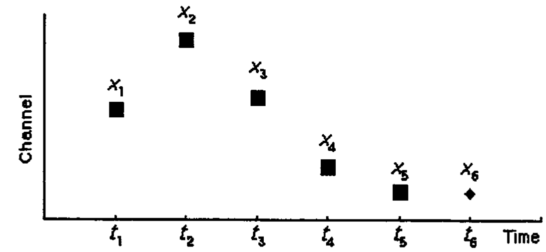

Figure 3. Time series prediction as a spatio-temporal pattern recognition problem.

The starting times of the PSP are represented as $t-\left(t_{u_i}+d_{ji}\right)$ in Equation 1. We need to adjust the delays to make a certain output neuron fire according to the input data so that our spiking neural network has adaptive delays (Figure 2). In addition, there are multiple delays between the input and the output neurons. The basic assumption underlying this approach is that there are several paths with different delays between each input neuron and output neuron.

Initially, there are no synaptic delays, but synapses grow as the output neuron fires. New synapses are added each time the output neuron fires. So, output neurons have different numbers of synapses depending on the data history.

There are n input neurons, where n is the number of spatial channels for spatio-temporal pattern recognition (Figure 3).

The output layer has m neurons, where m is the number of patterns to learn. The number of output patterns match that of input values in experiments in this paper. Each output neuron has a time integrator described in this section.

3. Time series prediction as spatio-temporal pattern recognition

Time series can be considered as a combination of temporal and spatial information. For example, given a time series $\langle x_1,x_2,x_3,x_4,x_5\rangle$, this vector can be encoded as X = ⟨(spatial information, temporal information)⟩ = $\langle(x_1,t_1),(x_2,t_2),(x_3,t_3),(x_4,t_4),(x_5,t_5)\rangle$ (Figure 3). The value of x_6 which should be predicted at time t_6 can be considered as the result of previous spatio-temporal pattern $\langle(x_1,t_1),(x_2,t_2),(x_3,t_3),(x_4,t_4),(x_5,t_5)\rangle$. And this information can be encoded separately. The temporal information is encoded as delays and the spatial information is encoded as the location of input neurons. Early-fired signals are manipulated to be delayed.

For instance, spatial information being $\langle x_1, x_2, x_3, x_4, x_5 \rangle$, input neuron u corresponding to the value of $\langle x_1, x_2, x_3, x_4, x_5 \rangle$ fires successively. And the delays of output neuron v corresponding to the value of x_6 are set to $\langle t_6 - t_1, t_6 - t_2, t_6 - t_3, t_6 - t_4, t_6 - t_5 \rangle$. Each delay is connected to the input neuron that successively fires. In such a configuration, early-fired input neuron signals are delayed with a relatively large time and recently-fired input signals are transferred immediately. So the differences of the firing times of the input neurons can be evened out. As described in Section 2, output neurons having proper delays with input neuron of current signals will fire because PSPs occurred simultaneously.

4. Learning

How can we modify delays according to input patterns? If our task were a purely pattern recognition problem, it would be possible to implement a system to find RBF's center cluster delays in an unsupervised fashion [5]. But there are no center cluster delays in our experiment. The previously fired neurons can be fired again in different histories. This is the reason why the output neuron has multiple delays.

When output neuron v_j is activated at time t_{v_j}, a new synapse is created between v_j and the input neuron u_j firing at time t_{u_i} within a certain time interval $t_v - t_u < T$ called the coding interval. The synaptic delay d_{ji} between output neuron v_j and input neuron u_j is set to $t_{v_j} - t_{u_i}$. Synapses are created in the manner at each firing of the output neuron.

The roles of coding interval are summarized as follows. First, theoretically all previous inputs can have an influence on current value but some recent inputs should be considered more important as the influence of earlier input diminishes rapidly. Second, it relieves the network of computational burden.

5. Experimental results

Experiments have been performed on the laser amplitude fluctuation data (Figure 4). This is an univariate time record of a single observed quantity, measured in a physics laboratory experiment [6].

The maximum value of used data was 255 and the minimum value of used data was 2. There are 254 spatial channels. So we used 254 input neurons and the same number of output neurons in network. Other constants used in the experiment are shown in Table 1.

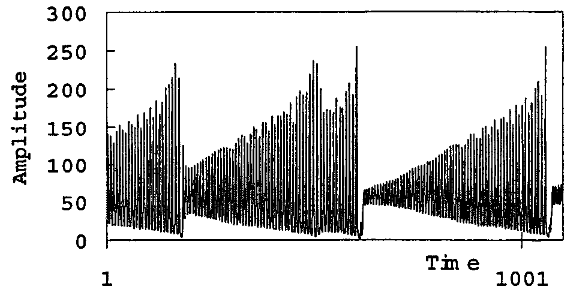

Figure 4. Laser amplitude fluctuation data [5].

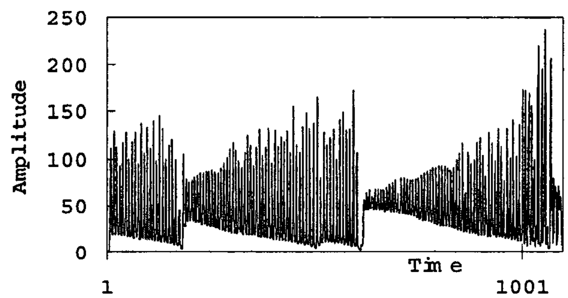

Figure 5. One-step ahead prediction result.

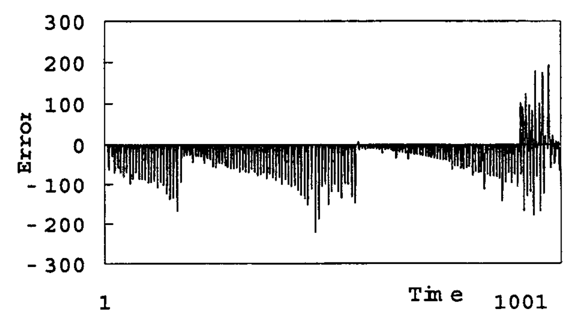

Figure 6. Prediction error.

Constant	C_{max}	t_p	Θ	T	n	M
Value	5	0.5	15	5	254	254

Table 1. The parameter values in the experiment.

Learning step

Delays of output neurons corresponding to each data point have been modified according to input time structure in the manner described in Section 4. We used the encoding interval of 5. That means the number of reference data is limited within five data points and the delay value can not exceed five. Each time an output neuron fires, a new synaptic delay is simply added. One thousand data points have been used in a learning step.

Recognition step

The data points used in the learning step have been given and another 100 validation data points have also been given. And the task is a one step ahead problem.

The prediction accuracy of the network was measured by the normalized mean squared error (NMSE):

$$\mathrm{NSME(N)} = \frac{\sum_{k \in N} (observation_k - prediction_k)^2}{\sum_{k \in N} (observation_k - mean)^2} \tag{3}$$

When only the mean value is predicted, the value of NMSE is one. The value of NMSE(1000) is 0.54185 in case of learning and the result of NMSE(100) is 1.556663 when validation data was given. It seems that the result is not so fantastic. But it doesn't explain whole situations.

In Figure 5 the output result pattern is similar to the original. It locally fluctuates and has some periodic behavior. Interestingly, the period of the fluctuation error pattern is the same as that of the original data (Figure 6).

What is the meaning of this? It may be explained as follows. Though the system is not very good at predicting the exact output values, it learned the trend of the overall pattern of the time series data. The model seems useful as a pattern recognizer.

6. Concluding remarks

We have presented a method for the application of SNNs to time series data. It has turned out that this method can be used as global temporal pattern recognizer. In biological systems, it is not well known how the synaptic delays are organized to reflect external stimuli. In this paper, learning was performed to memorize input time structure into delays. More realistic processes are expected to be more complex. It is possible that the key to solution lies in the activation functions which dynamically change to reflect the features of input data.

Acknowledgement

This work was supported in part by Korea Ministry of Science and Technology through KISTEP under Grant BR-2-1-G-06.

References

[1] Maass W., "Networks of spiking neurons: The third generation of neural network model," *Neural Networks*, Vol. 10, pp. 1659-1671, 1997.

[2] Hopfield J. J., "Pattern recognition computa5tion using action potential timing for stimulus representation," *Nature*, Vol.376, pp. 33-36 , 1995.

[3] Sejnowski T.J. , "Time for a new neural code?," *Nature*, Vol. 376, pp. 21-22 , 1995.

[4] Gerstner W., "Time structure of the activity in neural network models," *Physical Review E*, Vol. 51, pp. 738-758, 1995.

[5] Natschlager T. and Ruf B. "Spatial and temporal pattern analysis via spiking neuron," *Network: Computation in Neural Systems*, Vol. 9, pp. 319-332, 1998.

[6] Weigend A.S. and Gershenfeld N.A., "The future of time series," in: *Proceedings of the NATO advanced Research Workshop on Comparative Time Series Analysis*, Weigend A.S. and Gershenfeld N.A. (Eds.), pp. 1-70, 1992.