IEICE Proceeding Series

Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns—Association Ability for Sequential Patterns including Common Terms—

Junpei AMANO, Yuko OSANA

Vol. 1 pp. 632-635
Publication Date: 2014/03/17
Online ISSN: 2188-5079

2012 International Symposium on Nonlinear Theory and its Applications
NOLTA2012, Palma, Majorca, Spain, October 22-26, 2012

Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns — Association Ability for Sequential Patterns including Common Terms —

†Junpei AMANO and ‡Yuko OSANA

†Graduate School of Bionics, Computer and Media Sciences, Tokyo University of Technology
‡School of Computer Science, Tokyo University of Technology
1404-1 Katakura, Hachioji, Tokyo, 192-0982, Japan Email: †amano@osn.cs.teu.ac.jp, ‡osana@cs.teu.ac.jp

Abstract—In this paper, we examine the association ability for sequential patterns including common terms in the Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns (VFKFMAM-AR-SP). In this model, the connection weight fixed and semi-fixed neurons are introduced, and the patterns that have already been learned are not destroyed and a new pattern can be memorized. Moreover, when unknown patterns are given, neurons can be added in the map layer if necessary. In this research, we examined the association ability for sequential patterns including common terms and confirmed that the association ability for sequential patterns depends on the weighting factor β in the recurrent difference vector.

1. Introduction

In the real world, it is very difficult to get all information to learn in advance. So we need the model which can realize successive (additional) learning. However, most of the conventional neural network models can not realize successive learning. As the model which can realize successive learning, some models based on the Kohonen Feature Map (KFM) associative memory[1] have been proposed[2]-[5]. These models can learn sequential patterns including common terms successively, and have enough robustness for noisy input and damaged neurons.

Recently, we have proposed the Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns (VFKFMAM-AR-SP)[5]. This model is based on the Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Analog Patterns[4] and the Variable-sized Kohonen Feature Map Associative Memory with Refractoriness based on Area Representation (VKFMAM-R-AR)[6]. In this model, the connection weight fixed and semi-fixed neurons are introduced, and the pattern that has already been learned is not destroyed and a new pattern can be memorized. Moreover, when unknown patterns are given, neurons can be added in the map layer if necessary.

In this paper, we examine the association ability for sequential patterns including common terms in the Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns.

2. Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns

Here, we explain the Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns (VFKFMAM-AR-SP)[5].

2.1. Structure

The VFKFMAM-AR-SP has two layers; (1) input / output layer and (2) map layer, and the input / output layer is divided into two parts; (1) input part and (2) output part (See Fig.1). In this model, the map layer is treated as torus. In the VFKFMAM-AR-SP, neurons can be added in the map layer if necessary, so the distance between neurons in the map layer is not equal.

2.2. Learning Process

In the VFKFMAM-AR-SP, if enough area corresponding to the learning pattern can not be taken, some neurons are added in the map layer.

(1) In the initial network with the map layer composed of $x_{max} \times y_{max}$ neurons, the connection weights are initialized randomly.

(2) The initial recurrent difference vector is set to $\boldsymbol{y}_i(0) = \boldsymbol{0}$.

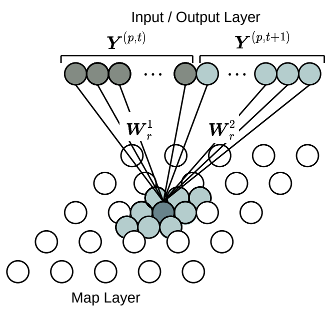


Figure 1: Structure of VFKFMAM-AR-SP.

(3) The recurrent difference vector of the neuron i in the map layer for the pattern t in the sequential pattern p $\boldsymbol{X}^{(p,t)}$, $\boldsymbol{y}_i(t)$ is calculated by

$$y_{ik}(t) = \begin{cases} \sum_{m=1}^{t}(1-\beta)^{t-m}X_k^{(p,m)} - \sum_{m=0}^{t-1}(1-\beta)^m W_{ik}, & (k \leq M/2) \\ X_k^{(p,t)} - W_{ik}, & (\text{otherwise}) \end{cases} \quad (1)$$

where M is the number of neurons in the input / output layer, β ($0.5 < \beta < 1$) is the weighting factor, W_{ik} is the connection weight between the neuron i in the map layer and the neuron k in the input / output layer. If the minimum of the norm of the recurrent difference vector is smaller than the threshold θ^l, the input vector $\boldsymbol{X}^{(p,t)}$ is regarded as the known pattern, and go to (14). Otherwise, go to (4).

(4) If there is no weight-fixed neuron, the neuron c is selected as the center of the learning area as follows:

$$c = \underset{i}{\operatorname{argmin}} \left\| \boldsymbol{y}_i(t) \right\| \quad (2)$$

and go to (12). Otherwise go to (5).

(5) Whether the area corresponding to the input pattern $\boldsymbol{X}^{(p,t)}$ can be taken without overlapping to the areas for stored patterns is checked. For the weight-fixed neurons z,

$$H_{i,z}^{area(1)} = \frac{1}{1+\exp\left(-(d_{iz} - (d^{max} + d_z^{min})D)/\varepsilon^h\right)} \quad (3)$$

is calculated, and if $H_{i,z}^{area(1)}$ is larger than the threshold θ^c for all weight-fixed neurons, the neuron i can be a center of the learning area. Here, θ^c is the threshold, ε^h is the steepness parameter, D is the constant which decides area size, d_{iz} is the distance between the neuron i and the weight-fixed neuron z, d^{max} is the maximum distance between neurons ($d^{max} = 1$) and d_z^{min} is the minimum distance from the weight-fixed neuron z to the nearest neuron. The neurons that satisfy this condition for all weight-fixed neurons are selected as the candidate of the center of the learning area. If there are some candidate neurons, go to (9). Otherwise, go to (6).

(6) Whether the area corresponding to the input pattern $\boldsymbol{X}^{(p,t)}$ can be taken without overlapping to the areas for stored patterns when the distance between neurons in the area for stored patterns is reduced to $\phi_n(d_z^{min})$ is checked. Here, $\phi_n(\cdot)$ is given by

$$\phi_n(d) = \begin{cases} d/2^n, & \left(d/2^n > d^{min}\right) \\ d, & (\text{otherwise}) \end{cases} \quad (4)$$

where n is the number of check in (6). d^{min} is the minimum distance between adjacent neurons. In the area for the input pattern, the distance between adjacent neurons is set to $\phi_{n-1}(d^{max})$.

For the weight-fixed neuron z,

$$H_{i,z}^{area(2n)} = \frac{1}{1+\exp\left(-(d_{iz} - (\phi_{n-1}(d^{max}) + \phi_n(d_z^{min}))D)/\varepsilon^h\right)} \quad (5)$$

is calculated, and if $H_{i,z}^{area(2n)}$ is larger than the threshold θ^c for all weight-fixed neurons, the neuron i can be a center of the learning area. The neurons that satisfy the condition for all weight-fixed neurons are selected as the candidate of the center of the learning area. If there are some candidate neurons, go to (9). Otherwise, go to (7).

(7) Whether the area corresponding to the input pattern $\boldsymbol{X}^{(p,t)}$ can be taken without overlapping to the areas for stored patterns when the distance between neurons in the area for stored patterns is reduced to $\phi_n(d_z^{min})$ and the distance between neurons in the area for the input pattern is set to $\phi_n(d^{max})$ is checked.

For the weight-fixed neuron z,

$$H_{i,z}^{area(2n+1)} = \frac{1}{1+\exp\left(-(d_{iz} - (\phi_n(d^{max}) + \phi_n(d_z^{min}))D)/\varepsilon^h\right)} \quad (6)$$

is calculated, and if $H_{i,z}^{area(2n+1)}$ is larger than the threshold θ^c for all weight-fixed neurons, the neuron i can be a center of the learning area. The neurons that satisfy the condition for all weight-fixed neurons are selected as the candidate of the center of the learning area. If there are some candidate neurons, go to (9). Otherwise, go to (8).

(8) In (6) and (7), if

$$\phi_{n+1}(d^{max}) \leq d^{min} \quad (7)$$

is satisfied, back to (6). Otherwise, it judges that the input pattern can not be learned as a new pattern.

(9) From the neurons which are selected as the center candidates of the learning area in (5)~(7), the neuron c whose norm of recurrent difference vector is minimum is selected.

(10) If the center candidates are selected in (6) or (7), the distance in the areas for stored patterns is reduced, and some neurons are added.

If the center candidates are selected in (6), for the area whose center is the neuron z which satisfies

$$\frac{1}{1+\exp\left(-(d_{cz} - (\phi_{n-1}(d^{max}) + \phi_{n-1}(d_z^{min}))D)/\varepsilon^h\right)} < \theta^c \quad (8)$$

the distance between neurons in the area is reduced, and neurons are added. The neurons which satisfy

$$\frac{1}{1+\exp\left(-(d_{iz} - d_z^{min}D)/\varepsilon^h\right)} < \theta^c \quad (9)$$

are generated as new neurons. The neuron i' corresponding to the neuron i ((x_i, y_i)) is generated at $(x_{i'}, y_{i'})$. Here, $x_{i'}$ and $y_{i'}$ are given by

$$x_{i'} = (x_i - x_z)\phi_n(d_z^{min}) + x_z \quad (10)$$

$$y_{i'} = (y_i - y_z)\phi_n(d_z^{min}) + y_z. \quad (11)$$

If the neuron exists at $(x_{i'}, y_{i'})$, no neuron is added there. The weight vector of the neuron i' $\boldsymbol{W}_{i'}$ is set as

$$\boldsymbol{W}_{i'} = \boldsymbol{W}_i. \quad (12)$$

If the center candidates are selected in (7), the new neurons are added in the area whose center z that satisfy

$$\frac{1}{1+\exp\left(-(d_{cz}-(\phi_n(d^{max})+\phi_{n-1}(d_z^{min}))D)/\varepsilon^h\right)} < \theta^c \quad (13)$$

(11) If the center candidates are selected in (6) or (7), new neurons are added in the area for the new pattern $\boldsymbol{X}^{(p,t)}$.

If the center candidates are selected in (6) and $n > 1$, the neurons are added in the area whose center is the neuron c. The neurons which satisfy

$$\frac{1}{1+\exp\left(-(d_{ic}-\phi_{n-1}(d^{max})D)/\varepsilon^h\right)} < \theta^c \quad (14)$$

are generated. The neuron i' corresponding to the neuron i (at $(x_i,\ y_i)$) is generated at $(x_{i'},\ y_{i'})$. Here, $x_{i'}$ and $y_{i'}$ are given by

$$x_{i'} = (x_i - x_c)\phi_{n-1}(d^{max}) + x_c \quad (15)$$

$$y_{i'} = (y_i - y_c)\phi_{n-1}(d^{max}) + y_c. \quad (16)$$

If the neuron exists at $(x_{i'},\ y_{i'})$, no neuron is added there. The weight vector $\boldsymbol{W}_{i'}$ is generated randomly.

If the center candidates are selected in (7), new neurons are added in the area whose center is the neuron c. The neurons which satisfy

$$\frac{1}{1+\exp\left(-(d_{ic}-\phi_n(d^{max})D)/\varepsilon^h\right)} < \theta^c \quad (17)$$

are generated.

(12) The input pattern $\boldsymbol{X}^{(p,t)}$ is trained in the area whose center is the neuron c. The connection weights which are not fixed are updated by

$$W_{ij} \leftarrow \begin{cases} \sum_{m=1}^{t}(1-\beta)^{t-m}X_k^{(p,m)} \Big/ \sum_{m=0}^{t-1}(1-\beta)^m, & (\theta_1^{learn} \le H(d_{ci}) \text{ and } k \le M/2) \\ X_k^{(p,t)}, & (\theta_1^{learn} \le H(d_{ci}) \text{ and } M/2 < k) \\ W_{ik} + H(d_{ci})\left(\sum_{m=1}^{t}(1-\beta)^{t-m}X_k^{(p,m)} \Big/ \sum_{m=0}^{t-1}(1-\beta)^m\right), & (\theta_2^{learn} \le H(d_{ci}) < \theta_1^{learn} \text{ and } H(d_{i^*i}) < \theta_1^{learn} \text{ and } M/2 < k) \\ W_{ik} + H(d_{ci})X_k^{(p,t)}, & (\theta_2^{learn} \le H(d_{ci}) < \theta_1^{learn} \text{ and } H(d_{i^*i}) < \theta_1^{learn} \text{ and } k \le M/2) \\ W_{ik}, & (\text{otherwise}) \end{cases} \quad (18)$$

where θ_1^{learn} and θ_2^{learn} ($\theta_1^{learn} > \theta_2^{learn}$) are the thresholds. And $H(d_{ci})$ and $H(d_{i^*i})$ are given by

$$H(d_{ij}) = \frac{1}{1+\exp\left((d_{ij}-d_i^{min}D)/\varepsilon^f\right)}. \quad (19)$$

(13) The connection weights of the neuron c, $\boldsymbol{W}_c$ are fixed.

(14) (3)~(13) are iterated until $t = t_p - 1$.

(15) (2)~(14) are iterated when a new pattern set is given.

2.3. Recall Process

When the pattern $\boldsymbol{X}^{(1)}(=(\boldsymbol{Y}^{IN},\boldsymbol{0})^T)$ is given, the output of the neuron i in the map layer at the time t, $x_i^{map}(t)$ is given by

$$x_i^{map}(t) = \begin{cases} 1, & (i = r) \\ 0, & (\text{otherwise}) \end{cases} \quad (20)$$

where r is selected randomly from the neurons which satisfy

$$\| \boldsymbol{y}_i(t) \| \le \theta^{map} \quad (21)$$

where $\boldsymbol{y}_i(t)$ is the recurrent difference vector of the neuron i in the map layer at the time t which is given by

$$y_{ik}(t) = \begin{cases} (1-\beta)y_{ik}(t-1) + \beta(X_k^{(t)} - W_{ik}), & (k \le M/2) \\ 0, & (\text{otherwise}) \end{cases} \quad (22)$$

In Eq.(21), θ^{map} is the threshold of the neuron in the map layer.

The output of the neuron k in the input / output layer at the time t, $x_k^{io}(t)$ is given by

$$x_k^{io}(t) = W_{rk}. \quad (23)$$

3. Computer Experiment Results

3.1. Relation between Association Ability and The Number of Common Terms

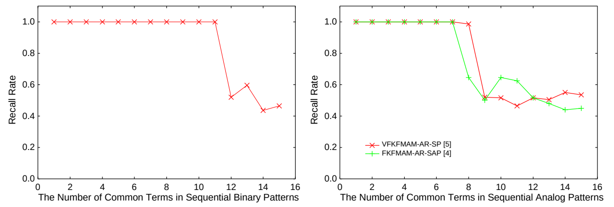
Here, we examined the relation between the association ability and the number of common terms in the VFKFMAM-AR-SP[5]. In this experiment, we used the network composed of 800 neurons in the input / output layer and 100 neurons in the initial map layer and two or three sequential patterns whose first and last patterns are common terms were memorized. Figures 2 and 3 show the relation between the association ability and the number of common terms in the VFKFMAM-AR-SP. For analog patterns, the results in the FKFMAM-AR-SAP[4] are also shown for reference.

3.2. Relation between Association Ability and Weighting Coefficient β

Here, we examined the relation between the association ability and the weighting coefficient β in the VFKFMAM-AR-SP[5]. In this experiment, we used the network composed of 800 neurons in the input / output layer and 100 neurons in the initial map layer and two sequential patterns whose first and last patterns are common terms were memorized. Figure 4 shows the relation between the association ability and the weighting coefficient β in the VFKFMAM-AR-SP. As shown in this figure, the VFKFMAM-AR-SP can recall sequential patterns which includes many common terms when β is small.

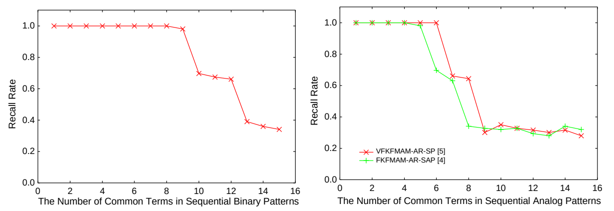
3.3. Robustness for Damaged Neurons / Noisy Input

Here, we examined the robustness for damaged neurons and noisy input of the VFKFMAM-AR-SP[5]. In these experiments, five random pattern sequences composed of four



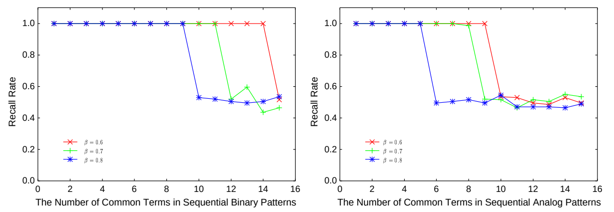
(a) Binary (b) Analog

Figure 2: Relation between Association Ability and The Number of Common Terms (1)



(a) Binary (b) Analog

Figure 3: Relation between Association Ability and The Number of Common Terms (2)



(a) Binary (b) Analog

Figure 4: Relation between Association Ability and Weighting Coefficient β

patterns were memorized. Figures 5 and 6 show the robustness of the VFKFMAM-AR-SP. As shown in these figures, the VFKFMAM-AR-SP has enough robustness for damaged neurons and noisy input as similar as the FKFMAM-AR-SAP[4].

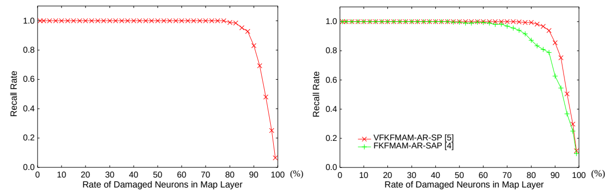
4. Conclusions

In this paper, we examined the association ability in the Variable-sized Fast Kohonen Feature Map Associative Memory using Area Representation for Sequential Patterns[5]. We carried out a series of computer experiments and confirmed that the VFKFMAM-AR-SP has following features.

(1) It can learn sequential patterns successively and neurons can be added in the map layer if necessary.

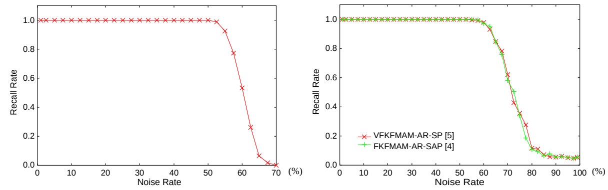
(2) It can deal with sequential patterns including common terms.

(3) The association ability depends on the weighting coefficient β.



(a) Binary (b) Analog[5]

Figure 5: Robustness for Damaged Neurons.



(a) Binary (b) Analog[5]

Figure 6: Robustness for Noisy Input.

(4) It has robustness for noisy input.

(5) It has robustness for damaged neurons.

References

[1] H. Ichiki, M. Hagiwara and M. Nakagawa: "Kohonen feature maps as a supervised learning machine," Proceedings of IEEE International Conference on Neural Networks, pp.1944–1948, 1993.

[2] N. Sakurai, M. Hattori and H. Ito: "SOM associative memory for temporal sequences," IEEE and INNS International Joint Conference on Neural Networks, pp.950–955, Honolulu, 2002.

[3] T. Shiratori and Y. Osana : "Kohonen feature map associative memory with area representation for sequential analog patterns," Proceedings of IEEE and INNS International Joint Conference on Neural Networks, Hong Kong, 2008.

[4] H. Midorikawa and Y. Osana : "Fast Kohonen feature map associative memory using area representation for sequential analog patterns," Proceedings of International Conference on Neural Information Processing, Sydney, 2010.

[5] J. Amano and Y. Osana : "Variable-sized fast Kohonen feature map associative memory using area representation for sequential patterns," Proceedings of IEEE International Conference on System, Man and Cybernetics, Anchorage, 2011.

[6] T. Imabayashi and Y. Osana : "Variable-sized KFM associative memory with refractoriness based on area representation," Proceedings of IEEE International Conference on System, Man and Cybernetics, San Antonio, 2009.