Structural Information Control for Flexible Competitive Learning

Ryotaro Kamimura*, Taeko Kamimura† Thomas R. Shultz‡

*Information Science Laboratory, Tokai University, 1117 Kitakaname, Hiratsuka, Kanagawa, 259-1292, Japan, ryo@cc.u-tokai.ac.jp †Department of English, Senshu University, 2-1-1 Higashimita, Tama-ku, Kawasaki, Kanagawa, 214-8580, Japan, taekok@isc.senshu-u.ac.jp ‡Department of Psychology, McGill University, Montreal, Quebec, H3A 1B1, Canada, shultz@psych.mcgill.ca

Abstract

In this paper, we propose a new information theoretic method called *structural information* to overcome fundamental problems inherent in conventional competitive learning such as dead neurons and deciding on the appropriate number of neurons in the competitive layer. @Our method is based on defining and controlling several kinds of information, thus generating particular neuron firing patterns. For one firing pattern, some neurons are completely inactive, meaning that some dead neurons are generated. For another firing pattern, all neurons are active, that is, there is no dead neurons. This means that we can control the number of dead neurons and choose the appropriate number of neurons by controlling information content. We applied this method to simple pattern classification to show that information can be controlled, and that different neuron firing patterns can be generated.

1 Introduction

The majority of unsupervised learning methods have been based upon competitive learning [1]. In competitive learning, neurons compete with each other and finally one neuron (winner-take-all) wins the competition for given input patterns. Though competitive learning is a simple and powerful method for unsupervised learning, several problems such as dead neurons and deciding on the appropriate number of neurons have been pointed out. For overcoming these problems, a number of heuristic methods have been proposed: for example, leaky learning [2], a conscience method [3], frequency-sensitive learning [4], rival penalized competitive learning [5] and lotto-type competitive learning [6]. Though each of these heuristic methods may be effective in eliminating one specific problem, none of them can solve the dead neurons problems and the appropriate number of neurons in a unified way.

In this context, we propose a novel approach called *structural information* to control the number of dead neurons. Our method is based upon information theoretic approaches to competitive learning. Information theoretic methods applied to neural computing have so far given promising results [7], [8]. One thing that these approaches have in common is that they have been exclusively concerned with the quantity of information obtained by learning. However, it is easily pointed out that the same amount of information can produce a number of unpredictable network final states or information representations. Consequently, current information theoretic methods are not as powerful as they might be. Structural information has been introduced to overcome these problems. In the structural information method, we can define several different kinds of information content, corresponding to different neurons activation patterns. By controlling this structural information, we can control competitive unit activation patterns. For example, by controlling structural information, we can create an activation pattern in which some neurons are always off, and thus not used for classification. This activation pattern can be used to deal with a situation where the number of neurons is redundantly larger than the expected classes. Thus, structural information control is used to control the number of dead neurons and to choose the appropriate number of competitive units.

2 Structural Information

In this paper, we are concerned not with information to be transmitted through information channels but with stored information in systems [9]. Thus, *information* can be defined as a decrease in uncertainty. Structural information is introduced to see this information content in a more detailed way.

We can distinguish different kinds of structural information. First, we consider simple structural information with two random variables to be immediately extended to n random variables. This structural information is defined by the uncertainty decrease from maximum to actually observed uncertainty. The maximum uncertainty H_0 is computed by $\log M$, where M is the number of elements in a system and actual uncertainty H_1 is described by first order entropy or uncertainty

$$H_1 = -\sum_j p(j) \log p(j), \tag{1}$$

where $p(j)$ denotes that the jth unit in a system occurs with probability $p(j)$. Thus, information independent of input patterns, that is, first order information is defined by

$$\begin{aligned} D_1 &= H_0 - H_1 \\ &= \log M + \sum_j p(j) \log p(j). \end{aligned} \tag{2}$$

First order entropy or uncertainty H_1 may be further decreased to second order uncertainty H_2, that is, the decrease in uncertainty after receiving input signals:

$$H_2 = -\sum_s \sum_j p(s)p(j \mid s) \log p(j \mid s), \tag{3}$$

where $p(s)$ represent the probability of input signals s. This uncertainty decrease, that is, second order information is defined by

$$\begin{aligned} D_2 &= H_1 - H_2 \\ &= -\sum_j p(j) \log p(j) \\ &\quad + \sum_s \sum_j p(s)p(j \mid s) \log p(j \mid s). \end{aligned} \tag{4}$$

Using the structural parameter α, second order structural information $SI^{(2)}$ is defined by

$$SI^{(2)} = \alpha D_1 + (1-\alpha) D_2, \tag{5}$$

where α is the structural parameter, ranging between 0 and 1. When the structural parameter α is 0, structural information is equivalent to second order information. On the other hand, when the structural parameter is 1, structural information is equal to first order information.

Finally, we should note that the structural information just explained can easily be extended to a more general n case as follows.

$$SI^{(n)} = \sum_n \alpha D_n, \tag{6}$$

where D_n denotes the nth order structural information, and $\sum_n \alpha_n = 1$.

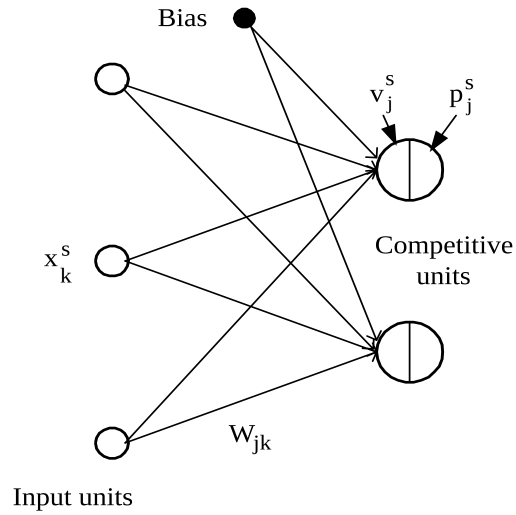

Fig. 1. Network architecture for defining structural information.

3 Application to Neural Learning

In this section, we attempt to apply the structural information method to a neural network architecture. As shown in Figure 1, the network architecture is composed of input units x_k^s and competitive units v_j^s.

Let us define information for competitive units and try to control competitive unit activation patterns. The jth competitive unit receives a net input from input units, and an output from the jth competitive unit can be computed by

$$v_j^s = f(\sum_k w_{jk} x_k^s), \tag{7}$$

where w_{jk} denotes a connection from the kth input unit to the jth competitive unit. In modeling competition among units, one of the easiest ways is to normalize the outputs from the competitive units as follows:

$$p_j^s = \frac{v_j^s}{\sum_m v_m^s}. \tag{8}$$

The conditional probability $p(j \mid s)$ is approximated by this normalized competitive unit output, that is,

$$p(j \mid s) \approx p_j^s. \tag{9}$$

Since input patterns are supposed to be uniformly given to networks, the probability of the jth competitive unit is approximated by

$$\begin{aligned} p(j) &= \sum_s p(s)p(j \mid s) \\ &\approx \frac{1}{S} \sum_s p_j^s \end{aligned}$$

$$= \quad p_j. \tag{10}$$

Using these approximated probabilities, first order information is approximated by

$$\begin{aligned} D_1 &= \log M + \sum p(j)\log p(j) \\ &\approx \log M + \sum_j p_j \log p_j. \end{aligned} \tag{11}$$

Second order information D_2 is approximated by

$$\begin{aligned} D_2 &= -\sum_j p(j)\log p(j) + \sum_s \sum_j p(s)p(j \mid s)\log p(j \mid s) \\ &\approx -\sum_j p_j \log p_j + \frac{1}{S}\sum_s \sum_j p_j^s \log p_j^s. \end{aligned} \tag{12}$$

As second order information is larger, specific pairs of input patterns and competitive units are strongly correlated. Second order structural information is approximated by

$$\begin{aligned} SI^{(2)} &= \alpha D_1 + (1-\alpha)D_2 \\ &\approx \alpha \log M + (2\alpha - 1)\sum_j p_j \log p_j \\ &\quad +(1-\alpha)\sum_s \frac{1}{S}\sum_j p_j^s \log p_j^s. \end{aligned} \tag{13}$$

Differentiating structural information with respect to input-competitive connections w_{jk}, we have the update rule:

$$\begin{aligned} \Delta w_{jk} &= \beta(2\alpha-1)\sum_s \left(\log p_j - \sum_m p_m^s \log p_m\right) Q_{jk} \\ &\quad +\beta(1-\alpha)\sum_s \left(\log p_j^s - \sum_m p_m^s \log p_m^s\right) \\ &\quad \times Q_{jk}, \end{aligned} \tag{14}$$

where

$$Q_{jk} = \frac{1}{S} p_j^s (1 - v_j^s)\xi_k^s, \tag{15}$$

and where β is the learning rate parameter.

4 Application to Simple Pattern Detection

In this experiment, we used artificial data to show how the structural information method can produce different types of activation patterns, depending upon the chosen structural parameter α. Figure 4(a) shows four input patterns used for this experiment. Because the number of input patterns is four, the number of competitive units is restricted to four. Even if the number of competitive units is larger than four, the results obtained will not change. The

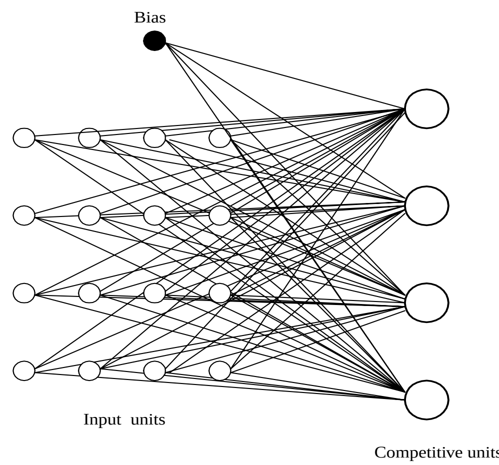

Fig. 2. A network architecture for artificial data in which the numbers of input and hidden units are 16 and 4, respectively. The bias is exclusively used to break the symmetry of competitive unit activation patterns.

number of input units was 16, as shown in Figure 2. The learning parameter β was always set to 1. The learning epochs were set to 1,000. No further adjustments were made to the learning process.

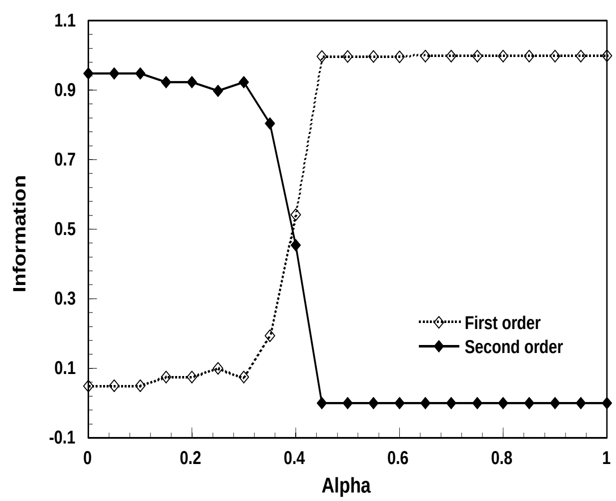

Fig. 3. Information as a function of the structural parameter α. First order and second information represent normalized information, that is, information divided by the maximum information. First and second order information are represented by a dotted and a solid line respectively.

Figure 3 shows information as a function of the structural parameter α. The information was averaged over ten different runs, and it was normalized for its values to range between 0 and 1. As can be seen in the figure, second order information represented by a solid line is close to the maximum point when the structural parameter α is 0. As the structural parameter is increased, second order information is decreased. Actually second order information reaches almost the zero level after the structural parameter α becomes 0.45. On the other hand, first order information represented by the dotted line is close to 0 when the structural parameter is 0. As the structural parameter is increased, first order information is increased, and it reaches approximately the maximum point after the structural parameter α becomes 0.45. When the structural parameter α is 0.4, first and second order information become almost equal to each other.

Figure 4 shows four input patterns (a) and different kinds of competitive unit activation patterns for three different parameter values (b), (c) and (d). In (b), (c) and (d), black and white squares represent competitive unit activation levels p_j^s close to 1 and close to 0, respectively.

As can be seen in (b), when the structural parameter is 0, the four different competitive units respond to different input patterns respectively. This means that we have completely specialized competitive units. In this case, we there are no dead neurons, and all neurons are equally used. In principle, conventional competitive learning can produce only this final state [2]. When the structural parameter is increased to 0.4(c), we have many different kinds of internal representations, depending upon different initial conditions. The activation pattern (c1) shows that the third competitive unit responds to three input patterns and that the fourth competitive unit responds to just one input pattern. The representation (c2) shows that only one competitive unit responds to all the input patterns. The representation (c4) illustrates that the network can classify four input patterns into two groups. The numbers of dead neurons in (c1), (c2), (c3) and (c4) are 2, 3, 2 and 2, respectively. Finally, as shown in Figure (d), when the structural parameter is 1, that is, when first order information is only used, just one competitive unit always responds to all four input patterns. The number of dead neurons here is always three. These results have shown that structural information can control the number of dead neurons and choose the appropriate number of competitive units by adjusting the structural parameter α.

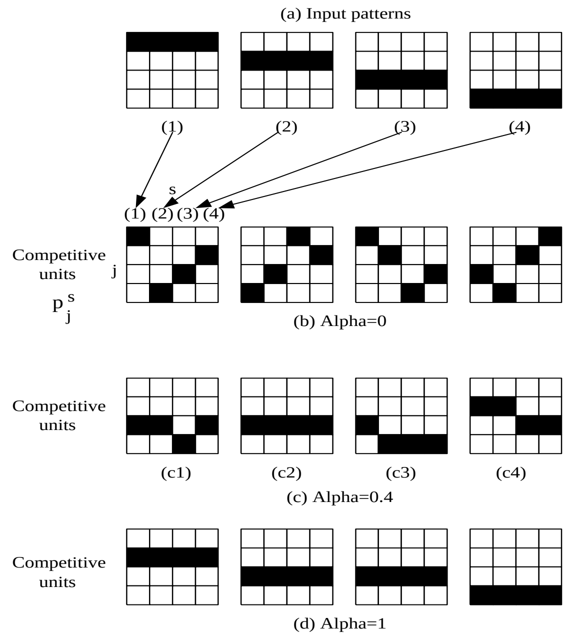

Fig. 4. Input patterns and final competitive unit activation patterns p_j^s for three different structural parameter values. Black and white squares in (b), (c) and (d) represent the activation values p_j^s close to 1 and 0, respectively. No intermediate activation levels could be seen in the experiments.

References

[1] S. Grossberg, "Competitive learning: from interactive activation to adaptive resonance," *Cognitive Science*, vol. 11, pp. 23–63, 1987.

[2] D. E. Rumelhart and D. Zipser, "Feature discovery by competitive learning," in *Parallel Distributed Processing* (D. E. Rumelhart, G. E. Hinton, and R. J. Williams, eds.), vol. 1, pp. 151–193, Cambrige: MIT Press, 1986.

[3] D. Diesieno, "Adding a conscience to competitive learning," in *Proceedings of IEEE International Conference on Neural Networks*, pp. 117–124, IEEE, 1988.

[4] S. C. Ahalt, A. K. K. andd P. Chen, and D. E. Melton, "Competitive learning algorithms for vector quantization," *Neural Networks*, vol. 3, pp. 277–290, 1990.

[5] L. Xu, "Rival penalized competitive learning for clustering analysis, rbf net, and curve detection," *IEEE Transaction on Neural Networks*, vol. 4, no. 4, pp. 636–649, 1993.

[6] A. Luk and S. Lien, "Properties of the generalized lotto-type competitive learning," in *Proceedings of International conference on neural information processing*, pp. 1180–1185, 2000.

[7] R. Linsker, "Self-organization in a perceptual network," *Computer*, vol. 21, pp. 105–117, 1988.

[8] R. Kamimura and S. Nakanishi, "Hidden information maximization for feature detection and rule discovery," *Network*, vol. 6, pp. 577–622, 1995.

[9] L. L. Gatlin, *Information Theory and Living Systems*. Columbia University Press, 1972.