Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Self-Supervised Learning by Information Enhancement: Target-Generating and Spontaneous learning for Competitive Learning

Ryotaro Kamimura
IT Education Center, Tokai University,
1117 Kitakaname Hiratsuka Kanagawa 259-1292, Japan
ryo@cc.u-tokai.ac.jp

***Abstract*— In this paper, we propose a new self-supervised learning method for competitive learning as well as self-organizing maps. In this model, a network enhances its state by itself, and this enhanced state is to be imitated by another state of the network. We set up an enhanced and a relaxed state, and the relaxed state tries to imitate the enhanced state as much as possible by minimizing the free energy. To demonstrate the effectiveness of this method, we apply information enhancement learning to the SOM. For this purpose, we introduce collectiveness, in which all neurons collectively respond to input patterns, into an enhanced state. Then, this enhanced and collective state should be imitated by the other non-enhanced and relaxed state. We applied the method to an artificial data and three data from the well-known machine learning database. Experimental results showed that the U-matrices obtained were significantly similar to those produced by the conventional SOM. However, better performance could be obtained in terms of quantitative and topological errors. The experimental results suggest the possibility for self-supervised learning to be applied to many different neural network models.**

I. Introduction

A. Self-Supervised Learning

In this paper, we propose a new self-supervised learning in which a target is itself creased within a network. We suppose that a network can take two kinds of states, that is, an enhanced and a relaxed state. An enhanced state is one where units respond explicitly to input patterns, whereas a relaxed state is one where units respond almost equally to input patterns. The enhanced state is considered to be a goal to be attained by the relaxed state. Thus, this is a new learning model, in which targets are not given from the outside but are to be created by networks themselves. Though some attempts have been made to unify supervised and unsupervised learning or to include unlabeled data with label data in learning, for example, semi-supervised learning [1], [2], the concept of target-creating learning and spontaneous target-generation has not been much considered in neural networks nor in machine learning.

B. Relations to Competitive Learning

Self-supervised learning is the one most suited for competitive learning [3], because an enhanced state is considered to be one where one neuron wins the competition. Competitive learning has been one of the most important techniques used in neural networks, and it has been applied to several well-known models [4], [5], [6], [7], [8]. Many methods to refine competitive learning have been proposed [9], [10], [11], [12], [13], [14], [15], [16] [17], [18]. Our method of self-supervised learning is easily applied to competitive learning. In competitive learning, the winner-take-all algorithm picks up a best-matching unit, and connection weights only into the best-matching unit are updated [3]. We can immediately see that the state realized by this winner-take-all algorithm is considered to be an enhanced state to be attained by a relaxed state. In competitive learning, the winner-take-all algorithm is an outer operation to create, in our sense, an enhanced state. In self-supervised learning, an enhancement operation is considered to be an inner operation. Thus, an enhanced state is realized not by the outer winner-take-all algorithm, but by information enhancement with an inner enhancement parameter.

C. Information-Theoretic Interpretation

Self-supervised learning is considered to be a method to attain a state with larger information content on input patterns. We have seen that competitive learning is one of the typical examples of self-supervised learning. We have found so far that competitive learning has been realized by maximizing mutual information between competitive units and input patterns [19], [20], [21], [22], [23], [24]. Thus, it is possible to interpret self-supervised learning from an information-theoretic point of view. An enhanced state is considered to be a state with maximum information on input patterns; on the other hand, a relaxed state is one with minimum information. Using the concept of competitive learning, self-supervised learning can itself create a higher information state in which competitive units respond explicitly to input patterns and a lower information state in which competitive units respond to input patterns almost equally. Then, self-supervised learning tries to attain the higher information state from the lower information state.

D. Free Energy for Learning

To minimize the difference between an enhanced and a non-enhanced state, we introduce the cross entropy between two probabilities observed in two states. We must minimize this cross entropy in learning. Update rules obtained by directly

978-1-4244-2794-9/09/$25.00

differentiating the cross entropy become computationally complex. Thus, to simplify computation, we introduce a free energy similar to that used in statistical mechanics. The free energy was first (to the best of our knowledge) introduced by Rose, Gurewitz and Fox [25], [26]. The same type of deterministic annealing was proposed by Ueda and Nakano [27], [28] to overcome the local maxima problem associated with the EM algorithm [29]. In those methods, the free energy is to be minimized at each temperature point. Then, Gaepel, Burge and Obermayer [30] took the cooling or annealing process further to derive a soft topographic vector quantization to be applied to SOM. In addition to those approaches, Heskes [31] made explicit the relations between vector quantization and self-organizing maps in terms of the free energy. Kamimura attempted to formulate the free energy to simplify mutual information maximization[32], [33]. Thus, many attempts have already been made to apply the free energy to neural network research. In our model, the free energy is used to minimize the cross entropy between an enhanced and a relaxed state.

E. Collective Activations

One of the main characteristics of self-supervised learning is that it can incorporate many constraints observed in applications. In a basic form, self-supervised learning is suited for competitive learning, as already mentioned. However, we use the self-organizing maps to demonstrate the applicability of our method, because it is easy to demonstrate the good performance of the method intuitively, that is, by some visualization techniques. In the conventional SOM, the winner-take-all algorithm is used to select winners, and connections to the winners and to their neighboring neurons are updated [34], [7], [8]. In the SOM, objective functions are not explicitly given in the original approaches, and there have been many attempts to explicitly identify objective functions and to reformulate them in the framework of an information-theoretic approach, Bayesian approach, statistical mechanical approach and mixture models [35], [36], [31], [37], [38], [39], [32], to cite a few.

In our approach, the objective function is explicit, because the network must imitate an enhanced state. The difference between enhanced probabilities and non-enhanced probabilities is one to be minimized. To realize self-organizing maps, we should incorporate lateral interactions among neurons. Instead of the lateral interactions, we introduce a concept of collectiveness in a self-enhanced state. With this new term "collectiveness," we stress the importance of lateral interactions, and in our model, collectiveness governs competitive processes, while in the conventional model, lateral interactions are implicitly built in competitive processes. We suppose that all neurons collectively respond to input stimuli. This collectiveness is realized by summing all competitive unit activations. The summation is actually a weighted sum of all neurons. This enhanced and collective state should be imitated by self-supervised learning. In self-supervised learning, we do not use the winner-take-all algorithm usually used in the self-organizing maps; we suppose only the collectiveness of

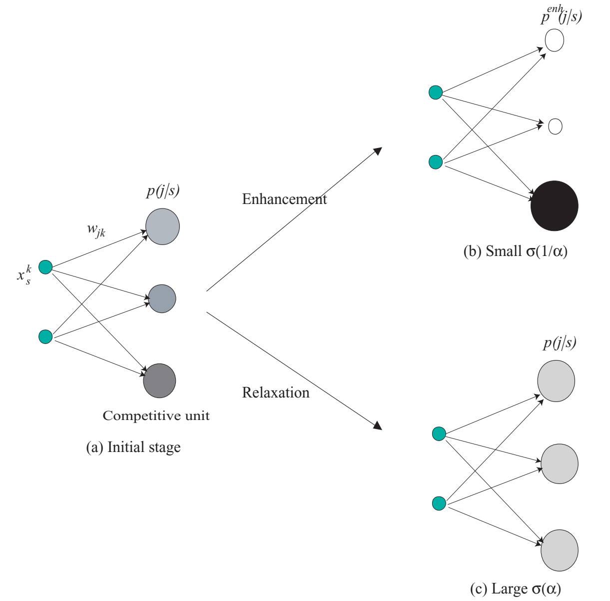

Fig. 1. Enhancement (b) and relaxation (c) of competitive units for a network itself.

competitive units. Thus, we think that our model is a more realistic and more biologically motivated computational procedure for competitive learning as well as self-organization. Similar methods were proposed by Linsker [40], [41], [42], [43] as an example of his maximum information principle. However, in Linsker's methods, only complicated learning rules were formulated, and they have not been used in practical applications, though some attempts have been made to formulate simple and local rules. Our methods are simple and practical and can be applied to large-scale applications.

II. Theory and Computational Methods

A. Enhancement and Relaxation

For introducing self-supervised learning, we have proposed information enhancement in which enhancement and relaxation are identified. Figure 1 shows a process of enhancement and relaxation for a network itself. Figure 1(a) shows an original situation obtained by competitive learning, in which three neurons are differently activated for input units. With enhancement, as shown in Figure 1(b), the characteristics of competitive unit activations are enhanced, and only one competitive unit is strongly activated. This means that obtained information in competitive units is larger. On the other hand, Figure 1(c) shows a state by relaxation, in which all competitive units respond uniformly to input units. Because competitive units cannot differentiate between input patterns, no information on input patterns is stored.

Now, let us compute competitive unit outputs for a network shown in Figure 1(a). A network is composed of the kth input unit x_k^s for the sth input pattern, and connection weights w_{jk} from the kth input unit to the jth competitive unit. The jth neuron output v_j^s for the sth input pattern can be defined by a

Gaussian-like function:

$$v_j^s \propto \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T \mathbf{\Sigma}^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}, \quad (1)$$

where $\mathbf{\Sigma}$ denotes the scaling matrix[1]. The enhanced scaling matrix can be defined by

$$\begin{aligned}\mathbf{\Sigma} &= \mathbf{\Sigma}_{enh}^{(\alpha)} \\ &= (1/\alpha)^2 \mathbf{I}, \end{aligned} \quad (2)$$

where $\mathbf{I}$ is an identity matrix and α is an enhancement parameter. When the relaxation is applied, we have

$$\begin{aligned}\mathbf{\Sigma} &= \mathbf{\Sigma}_{rel}^{(\alpha)} \\ &= (\alpha)^2 \mathbf{I}. \end{aligned} \quad (3)$$

B. Self-Enhancing

We have shown that an initial state can be split into an enhanced and a relaxed state. Then, we must decrease the gap between two states as much as possible. In the self-enhancing process, the difference between two probabilities should be as small as possible.

We now present update rules for self-supervised learning for a general case. At an enhanced state, competitive units can be computed by

$$v_{j,enh}^s \propto \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{enh}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}. \quad (4)$$

Normalizing this output, we have enhanced firing probabilities

$$p^{enh}(j \mid s) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{enh}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}}{\sum_{j=1}^{M} \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{enh}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}}. \quad (5)$$

Suppose that $p(j|s)$ denotes the probability of the jth neuron's firing in a relaxed state. Then, we should make these probabilities as close as possible to the probabilities of enhanced neurons' firing. Thus, we have an objective cross entropy function defined by

$$I = \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \log \frac{p(j \mid s)}{p^{enh}(j \mid s)}, \quad (6)$$

where M and S denote the number of competitive units and input patterns, respectively. In addition, we should decrease quantization errors, defined by

$$E = \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \|\mathbf{x}^s - \mathbf{w}_j\|^2. \quad (7)$$

It is possible to differentiate this cross entropy and the quantization error function to obtain update rules [44], [41]. However, the rules become complicated update rules with heavy computation required for computing conditional probabilities.

Fortunately, we can omit the complicated computation of conditional entropy by introducing the free energy used in statistical mechanics [27], [25], [45], [46], [47], [32], [33]. Borrowing from statistical mechanics, let us introduce free energy or a free energy-like function, defined by

$$\begin{aligned} F &= -2\alpha^2 \sum_{s=1}^{S} p(s) \log \sum_{j=1}^{M} p^{enh}(j \mid s) \\ &\quad \times \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{rel}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}. \end{aligned} \quad (8)$$

An optimal state specifies the output

$$\begin{aligned} p(j \mid s) &= p^{enh}(j \mid s) \\ &\quad \times \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{rel}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\} \\ &\quad \times (Z^s)^{-1} \end{aligned} \quad (9)$$

where

$$Z^s = \sum_{j=1}^{M} p^{enh}(j \mid s) \exp\left\{-\frac{1}{2}(\mathbf{x}^s - \mathbf{w}_j)^T (\mathbf{\Sigma}_{rel}^{(\alpha)})^{-1}(\mathbf{x}^s - \mathbf{w}_j)\right\}. \quad (10)$$

Then, putting $p(j \mid s)$ into the cross entropy, we have

$$\begin{aligned} F &= \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \|\mathbf{x}^s - \mathbf{w}_j\|^2 \\ &\quad + 2\alpha^2 \sum_{s=1}^{S} p(s) \sum_{j=1}^{M} p(j \mid s) \log \frac{p(j \mid s)}{p^{enh}(j \mid s)}. \end{aligned} \quad (11)$$

Thus, to decease the free energy, we must decrease the cross entropy and the corresponding error function. By differentiating the free energy, we have

$$w_{jk} = \frac{\sum_{s=1}^{S} p(j \mid s) x_k^s}{\sum_{s=1}^{S} p(j \mid s)}, \quad (12)$$

where $p(s)$ is set to $1/S$. We should note that, thanks to the excellent work of Heskes [31], the free energy can be interpreted in the framework of an EM algorithm.

In the above formulation, we have dealt with a general case of self-supervised learning. However, this is a self-supervised learning version of competitive learning. In application to competitive learning, an enhanced state is one where a winner takes all in the extreme case. This is realized by the enhancement parameter $1/\alpha$. On the other hand, a relaxed state is one where competitive units respond to input patterns almost equally, which is realized by setting the enhancement parameter to α. The self-supervised learning in terms of competitive learning tries to attain a state where the winner-take-all is predominant.

C. Collective Enhancement

To demonstrate clearly the performance of our method, we use self-organizing maps, because it is easy to interpret final results intuitively. In the previous section, we applied self-supervised learning to competitive learning. Thus, it is possible to apply it to self-organizing maps just by introducing lateral interactions in an enhanced state. Instead of lateral interactions, we introduce the concept of collectiveness in an enhanced

[1]We used the scaling matrix instead of the ordinary covariance matrix, because the output does not follow exactly the Gaussian function.

state. Then, we introduce collectiveness, as shown in Figure 2(c). Collectiveness means that all competitive units are related to each other. In the conventional approach [7], [8], competitive learning plays a central role in learning, and lateral interactions are introduced to include topological maps. In our approach, lateral interactions or collective activations play central roles, and these interactions and activations should be imitated by competition, that is, competitive learning. To stress the priority of lateral interactions in our model, we have introduced the concept of collectiveness.

Now, let us explain collective activations. Figure 2 shows a network architecture that is composed of a competitive and a collective layer. An enhanced output from an individual competitive unit can be computed by

$$v_{j,enh}^s \propto \exp\left\{-\frac{1}{2}(\mathbf{x}^s-\mathbf{w}_j)^T(\mathbf{\Sigma}_{enh}^{(\alpha)})^{-1}(\mathbf{x}^s-\mathbf{w}_j)\right\}. \quad (13)$$

Figure 2 shows that the jth neuron cooperates with all the other neurons on the map and responds to input patterns. We realize this cooperation by summing all the neighboring units' activities. Then, we have collected activations

$$V_j^s = \sum_{m=1}^{M} W_{jm} v_{m,enh}^s, \quad (14)$$

where W_{jm} denotes connection weights from the mth competitive unit to the jth competitive unit, and M is the number of competitive units. We can imagine many kinds of collectiveness on the map; however, we usually use distance between competitive units for expressing collectiveness. For example, when competitive units are closer to their neighbors, they should be linked to them more intensely. A distance function between two neurons can be defined by

$$W_{jm} = \exp\left(-\frac{1}{2}\|\mathbf{r}_j-\mathbf{r}_m\|^2\right), \quad (15)$$

where $\mathbf{r}_j$ denotes a position for the jth unit, and $\mathbf{r}_m$ denotes a position for the mth neighboring neuron. Thus, we have

$$V_j^s = \sum_{m=1}^{M} \exp\left(-\frac{1}{2}\|\mathbf{r}_j-\mathbf{r}_m\|^2\right) v_{m,enh}^s. \quad (16)$$

We can compute a normalized activity

$$p^{coll}(j \mid s) = \frac{V_j^s}{\sum_{m=1}^{M} V_m^s}. \quad (17)$$

This normalized activity is considered to represent collective firing rates. Then, we can introduce the free energy:

$$\begin{aligned} F &= -2\alpha^2 \sum_{s=1}^{S} p(s) \log \sum_{j=1}^{M} p^{coll}(j \mid s) \\ &\times \exp\left\{-\frac{1}{2}(\mathbf{x}^s-\mathbf{w}_j)^T(\mathbf{\Sigma}_{rel}^{(\alpha)})^{-1}(\mathbf{x}^s-\mathbf{w}_j)\right\}. \quad (18) \end{aligned}$$

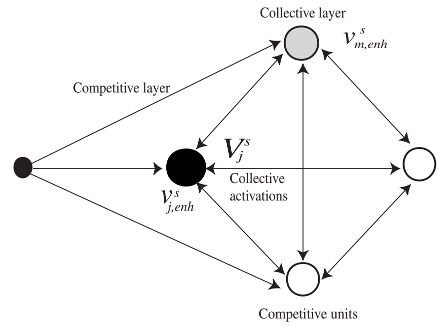

Fig. 2. A concept of self-enhanced collective activations.

III. Results and Discussion

In this section, we present the experimental results of using self-supervised learning. The results can be summarized by two points. First, self-supervised learning with a larger enhancement parameter can produce U-matrices and component planes similar to those produced by the conventional SOM. Second, self-supervised learning shows better results in terms of quantization and topological errors. For easy interpretation, comparison and reproduction, we used the SOM toolbox with default parameter values [48] where no special options were used for easy reproduction. All data in this experiment were normalized, with a range between zero and one. The number of competitive units was automatically determined by the software package. In self-supervised learning, no special techniques, for example, to accelerate learning, were used. Thus, experimental results presented here can easily be reproduced. We computed the quantitative measures for performance comparison: the quantization error and the topological error. The quantization error is simply the average distance from each data vector to its BMU. The topographic error is the percentage of data vectors for which the BMU and the second BMU are not neighboring units [48]. In addition, all the data except the first artificial data were taken from the well-known UCI machine learning database [2] for easy reproduction of the experimental results.

First, we applied the method to an artificial and symmetric data. Figures 3(a) and (b) show input patterns and quantization errors, respectively. As shown in Figure 3(b), quantization errors by the enhancement learning are gradually decreased as the enhancement parameter α is increased and finally the error becomes 0.428, while the error by the conventional SOM is 0.445. The topological errors are always zero by both methods. Although the difference is small, we can say that the enhancement learning shows better performance. Figures 4 (a) and (b) show the U-matrix and a map with labels by SOM and self-supervised learning, respectively. The almost same feature maps can be obtained. However, if we closely examine the maps, we can say that warmer-colored boundaries on the map by the self-supervised learning is more emphasized or more intensified than those by the conventional SOM. In addition,

[2]http://www1.ics.uci.edu/ mlearn/MLRepository.html

the labels by the self-supervised learning in Figure 4(b2) show the clearer and more natural ordering of the input data.

Figure 5 show quantization errors by the SOM in red and self-supervised learning in blue for the SPECTF heart database in the machine learning database where the numbers of input patterns and units are 80 and 22, respectively. Because the topological errors were zero by both methods, a figure for the topological errors was omitted. As shown in the figure, the quantization error is gradually decreased and finally the error is 1.121 when the enhancement parameter α is 14. On the other hand, by the conventional SOM, the quantization error is 1.133. Thus, the self-supervised learning shows slightly better performance in terms of quantization error. Figures 6(a) and (b) show U-matrices and labels by the SOM and the enhanced information, respectively. When we see labels, we can see that the maps can be divided into two parts: an upper and a lower part. However, in both U-matrices, we cannot see the clear boundaries separating two groups.

Figures 7(a) and (b) show quantization and topological errors by the SOM and self-supervised learning, respectively for the wine problem of the machine learning data base. The numbers of patterns and input units are 178 and 13, respectively. As shown in Figure 7(a), the minimum quantization error by the self-supervised learning is 0.364 when the enhancement parameter α is 16. On the other hand, the quantization error by the SOM is 0.373. Thus, the self-supervised learning shows slightly better results compared with the SOM. Figure 7(b) shows topological errors by self-supervised learning in blue and SOM in red. As can be seen in the figure, the topological errors become zero by using the self-supervised learning when the enhancement parameter α is increased beyond the level of 13. On the other hand, the topological error by the SOM is 0.028. Thus, the self-supervised learning shows much better performance in terms of topological errors. Figures 8(a) and (b) show U-matrices and maps with labels by the conventional SOM and the self-supervised learning, respectively. We can say that the same kind of U-matrices and labels can be obtained. However, the boundaries between groups by the self-supervised learning are clearer ones than those by the SOM.

We finally use a data of breast cancer also extracted from the machine learning data base and the numbers of input units and patterns are 10 and 699, respectively. Figure 9(a) shows quantization error by the SOM and self-supervised learning. As shown in the figure, by the self-supervised learning, quantization errors become smaller and reaches its lowest point of 0.272, when the enhancement parameter α is 25. Figure 9(b) shows the topological error by the SOM, and the error by the conventional SOM is 0.042. On the other hand, while the error by the self-supervised learning is far below the level by the SOM and the lowest point is 0.009 when the enhancement parameter is four.

These results show that the self-supervised learning shows slightly better performance in terms of quantization error, while the self-supervised learning shows significantly better results for two problems in terms of topological errors.

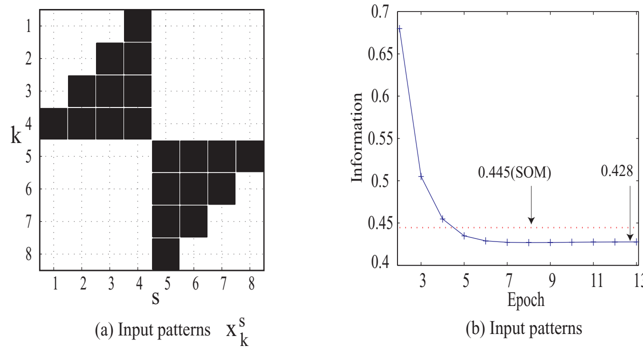

Fig. 3. Input patterns (a) and quantization errors (b) by self-supervised learning in blue and by SOM in red.

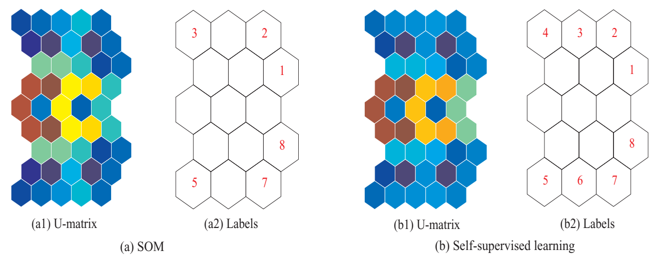

Fig. 4. U-matrix and labels by SOM (a) and self-supervised learning (b).

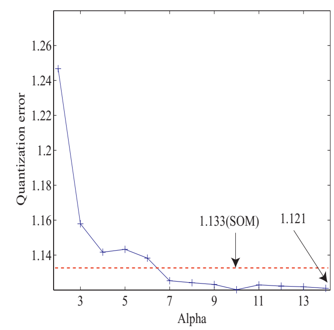

Fig. 5. Quantization errors as a function of the parameter α.

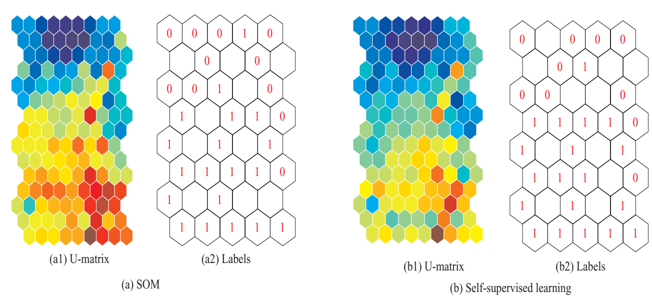

Fig. 6. U-matrix and labels by the conventional SOM (a) and self-supervised learning (b).

Fig. 7. Quantization errors (a) and topological errors (b) as a function of the parameter α.

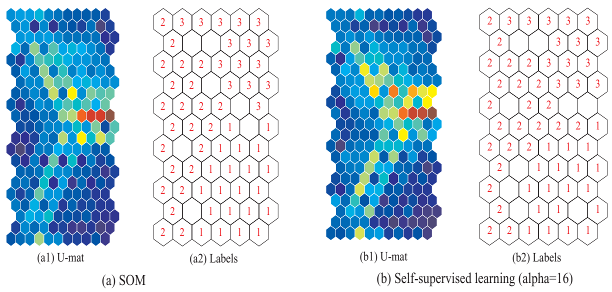

Fig. 8. U-matrix and labels by the conventional SOM (a) and self-supervised learning (b).

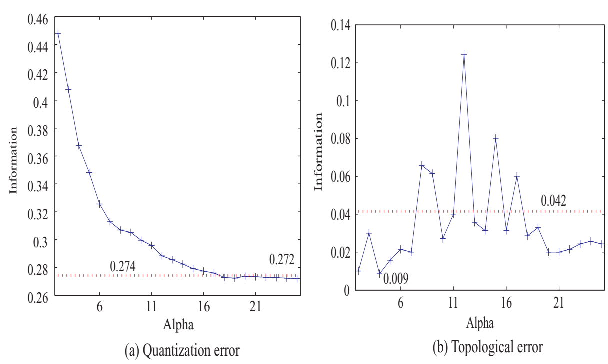

Fig. 9. Quantization errors (a) and topological errors (b) as a function of the parameter α.

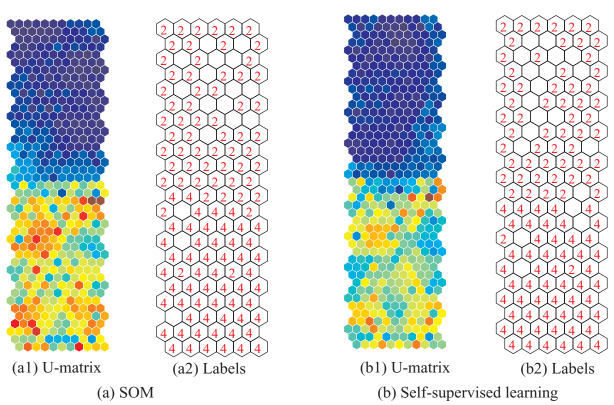

Fig. 10. U-matrices and labels by SOM (a) and self-supervised learning (b).

IV. Conclusion

We have proposed a new learning model of self-supervised learning. In the model, a state can be viewed as an enhanced or relaxed state. This relaxed state tries to attain the enhanced state. These states can be realized by changing the Gaussian wdith or the enhancement parameter α. The self-supervising learning processes have been realized by free energy minimization in which the cross entropy between two states is minimized, and at the same time, the errors between input patterns and connection weights are minimized.

We applied self-supervised learning to competitive learning, especially to the SOM, because it is easy to demonstrate the good performance of our method intuitively and quantitatively by its visualization techniques [49], [50], [51]. When applied the method to the SOM in which enhanced competitive units behave collectively. The self-supervised must imitate this collective behavior of the enhanced state. We applied the method to the artificial data and three databases from the well-known machine learning database. Experimental results showed that the U-matrices and labels obtained by self-supervised learning were significantly similar to those obtained by the conventional SOM when the enhancement parameter was large. However, better performance was obtained in terms of quantization and topological errors. Thus, our method has shown good potentiality for producing explicit self-organizing maps.

However, one problem, that of the optimal value of the enhancement parameter, should be solved, or we should develop a method in which we can easily determine the optimal value of the parameter. In addition, to better understand the results, we used the SOM, but this method has been primarily developed for competitive learning. Thus, comparison studies with competitive learning should be done in future studies. Though some problems can be solved for this method to be applicable to practical problems, our model surely opens up a new perspective for self-supervised learning.

Acknowledgment

The author is very grateful to Mitali Das for her valuable comments and suggestions.

References

[1] X. Zhu, "Semi-supervised learning literature survey," Tech. Rep. 1530, Compuer sciences, University of Wisconsin-Madison, 2005.

[2] O. Z. Chapell and B. Scholkopf, eds., *Semi-supervised learning*. MIT Press, 2005.

[3] D. E. Rumelhart and D. Zipser, "Feature discovery by competitive learning," *Cognitive Science*, vol. 9, pp. 75–112, 1985.

[4] K. Fukushima, "Cognitron: a self-organizing multi-layered neural network," *Biological Cybernetics*, vol. 20, pp. 121–136, 1975.

[5] K. Fukushima, "Neocognitron: a hierarchical neural network capable of visual pattern recognition," *Biological Cybernetics*, vol. 20, pp. 121–136, 1975.

[6] K. Fukushima, "Neocognitron: a neural network model for a mechanism of visual pattern recognition," *IEEE Transactions on Systems, Man, and Cyvernetics*, vol. 13, pp. 826–834, 1983.

[7] T. Kohonen, *Self-Organization and Associative Memory*. New York: Springer-Verlag, 1988.

[8] T. Kohonen, *Self-Organizing Maps*. Springer-Verlag, 1995.

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

[10] 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.

[11] A. Luk and S. Lien, "Properties of the generalized lotto-type competitive learning," in *Proceedings of International conference on neural information processing*, (San Mateo: CA), pp. 1180–1185, Morgan Kaufmann Publishers, 2000.

[12] M. M. Van Hulle, "Topographic map formation by maximizing unconditional entropy: a plausible strategy for 'on-line' unsupervised competitive learning and nonparametric density estimation," *IEEE Transactions on Neural Networks*, vol. 7, no. 5, pp. 1299–1305, 1996.

[13] M. M. V. Hulle, "The formation of topographic maps that maximize the average mutual information of the output responses to noiseless input signals," *Neural Computation*, vol. 9, no. 3, pp. 595–606, 1997.

[14] M. M. Van Hulle, "Faithful representations with topographic maps," *Neural Networks*, vol. 12, no. 6, pp. 803–823, 1999.

[15] Y. J. Zhang and Z. Q. Liu, "Self-splitting competitive learning: a new on-line clustering paradigm," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 369–380, 2002.

[16] H. Xiong, M. N. S. Swamy, and M. O. Ahmad, "Competitive splitting for codebook initialization," *IEEE Signal Processing Letters*, vol. 11, pp. 474–477, 2004.

[17] J. C. Yen, J. I. Guo, and H. C. Chen, "A new k-winners-take-all neural networks and its array architecture," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 901–912, 1998.

[18] S. Ridella, S. Rovetta, and R. Zunino, "K-winner machines for pattern classification," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 371–385, 2001.

[19] R. Kamimura and T. Kamimura, "Structural information and linguistic rule extraction," in *Proceedings of ICONIP-2000*, pp. 720–726, 2000.

[20] R. Kamimura, T. Kamimura, and O. Uchida, "Flexible feature discovery and structural information control," *Connection Science*, vol. 13, no. 4, pp. 323–347, 2001.

[21] R. Kamimura, "Information-theoretic competitive learning with inverse euclidean distance," *Neural Processing Letters*, vol. 18, pp. 163–184, 2003.

[22] R. Kamimura, "Teacher-directed learning: information-theoretic competitive learning in supervised multi-layered networks," *Connection Science*, vol. 15, pp. 117–140, 2003.

[23] R. Kamimura, "Progressive feature extraction by greedy network-growing algorithm," *Complex Systems*, vol. 14, no. 2, pp. 127–153, 2003.

[24] R. Kamimura, "Information theoretic competitive learning in self-adaptive multi-layered networks," *Connection Science*, vol. 13, no. 4, pp. 323–347, 2003.

[25] K. Rose, E. Gurewitz, and G. C. Fox, "Statistical mechanics and phase transition in clustering," *Physical review letters*, vol. 65, no. 8, pp. 945–948, 1990.

[26] K. Rose, E. Gurewitz, and G. C. Fox, "Vector quantization by deterministic annealing," *IEEE transactions on information theory*, vol. 38, no. 4, pp. 1249–1257, 1992.

[27] N. Ueda and R. Nakano, "Deterministic annealing variant of the em algorithm," in *Advances in Neural Information Processing Systems*, pp. 545–552, 1995.

[28] N. Ueda and R. Nakano, "Deterministic annealing em algorithm," *Neural Networks*, vol. 11, pp. 271–282, 1998.

[29] C. M. Bishop, *Neural networks for pattern recognition*. Oxford University Press, 1995.

[30] T. Graepel, M. Burger, and K. Obermayer, "Phase transitions in stochastic self-organizing maps," *Physical Review E*, vol. 56, no. 4, p. 3876, 1997.

[31] T. Heskes, "Self-organizing maps, vector quantization, and mixture modeling," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1299–1305, 2001.

[32] R. Kamimura, "Free energy-based competitive learning for mutual information maximization," in *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, pp. 223–227, 2008.

[33] R. Kamimura, "Free energy-based competitive learning for self-organizing maps," in *Proceedings of Artificial Intelligence and Applications*, pp. 414–419, 2008.

[34] T. Kohonen, "The self-organizing maps," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[35] G. J. Goodhill and T. J. Sejnowski, "A unifying objective function for topographic mappings," *Neural Computation*, vol. 9, pp. 1291–1303, 1997.

[36] S. P. Luttrell, "A Bayesian analysis of self-organising maps," *Neural Computation*, vol. 6, no. 5, pp. 767–794, 1994.

[37] B. Bakker and T. Heskes, "Clustering ensembles of neural network models," *Neural Networks*, vol. 16, pp. 261–269, 2003.

[38] A. Utsugi, "Hyperparameter selection for self-organizing maps," *Neural Computation*, vol. 9, no. 3, pp. 623–635, 1997.

[39] A. Utsugi, "Density estimation by mixture models with smoothing priors," *Neural Computation*, vol. 10, pp. 2115–2135, 1998.

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

[41] R. Linsker, "How to generate ordered maps by maximizing the mutual information between input and output," *Neural Computation*, vol. 1, pp. 402–411, 1989.

[42] R. Linsker, "Local synaptic rules suffice to maximize mutual information in a linear network," *Neural Computation*, vol. 4, pp. 691–702, 1992.

[43] R. Linsker, "Improved local learning rule for information maximization and related applications," *Neural Networks*, vol. 18, pp. 261–265, 2005.

[44] R. Kamimura, "Cooperative information maximization with gauissian activation functions for self-organizing maps," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 909–919, 2006.

[45] T. M. Martinez, S. G. Berkovich, and K. J. Schulten, "Neural-gas network for vector quanitization and its application to time-series prediction," *IEEE transactions on neural networks*, vol. 4, no. 4, pp. 558–569, 1993.

[46] D. Erdogmus and J. Principe, "Lower and upper bounds for misclassification probability based on renyi's information," *Journal of VLSI signal processing systems*, vol. 37, no. 2/3, pp. 305–317, 2004.

[47] K. Torkkola, "Feature extraction by non-parametric mutual information maximization," *Journal of Machine Learning Research*, vol. 3, pp. 1415–1438, 2003.

[48] J. Versanto, J. Himberg, E. ALhoniemi, and J. Parhankagas, "Som toolbox for matlab 5," Tech. Rep. A57, Helsinki University of Technology, 2000.

[49] S. Kaski, J. Nikkilä, and T. Kohonen, "Methods for interpreting a self-organized map in data analysis," in *Proceedings of ESANN'98, 6th European Symposium on Artificial Neural Networks, Bruges, April 22–24* (M. Verleysen, ed.), pp. 185–190, Brussels, Belgium: D-Facto, 1998.

[50] J. Vesanto, "SOM-based data visualization methods," *Intelligent-Data-Analysis*, vol. 3, pp. 111–26, 1999.

[51] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE-NN*, vol. 11, p. 586, May 2000.