Damaging Neurons of Affordable Neural Network for Pattern Recognition

Yoko Uwate
Institute of Neuroinfomatics,
University / ETH Zurich
Email: yu001@ini.phys.ethz.ch

Yoshifumi Nishio
Dept. Electrical and Electronic Eng.,
Tokushima University
Email: nishio@ee.tokushima-u.ac.jp

Ruedi Stoop
Institute of Neuroinfomatics,
University / ETH Zurich
Email: ruedi@ini.phys.ethz.ch

SUMMARY

Durability describes the ability of a device to operate properly in imperfect conditions. We have recently proposed a novel neural network structure called an "Affordable Neural Network" (AfNN), in which affordable neurons of the hidden layer are considered as the elements responsible for the robustness property as is observed in human brain function. Whereas earlier we have shown that AfNNs can still generalize and learn, here we show that these networks are robust against damages occurring after the learning process has terminated. The results support the view that AfNNs embody the important feature of durability. In our contribution, we investigate the durability of the AfNN is applied to pattern recognition when some of the neurons in the hidden layer are damaged after the learning process.

I. Introduction

Recently, studies on the brain have been carried out actively on various levels. On the neuroscience level, many researchers investigate how neuronal death occurs in the human brain. Neuronal death can be occurred by various but usual causes such as drinking a lot of alcohol, heavy smoking and concussions. Furthermore, some neurons die with apoptosis when the neuron can not receive any signals. Even if some neurons die in daily life, the human brain is still able to operate normally by creating new information processing from abundant experience and knowledge. This is because, the human brain has durability and flexibility.

One of most important application of neural networks is pattern recognition. Pattern recognition can be implemented by using a feedforward neural network that has been trained accordingly. During training, the network is trained to associate outputs with input patterns. When the network is used, it identifies the input pattern and tries to output the associated output pattern. The effectivenes of Back Propagation (BP) learning has been confirmed in a pattern recognition.

In our previous research, we have proposed a new network structure with affordable neurons in the hidden layer, for efficient BP learning [1]. We named this network "Affordable Neural Network" (AfNN). In this network, we prepare some extra neurons in the hidden layer. When the network executes operating, all of the neurons in the hidden layer are not used at every updating. By computer simulations, the AfNN has been confirmed to gain better performance for the BP learning on both convergence speed and learning efficiency. We can say that the AfNN has generalization ability. However, we believe that many advantages of the AfNN are still veiled. We consider that the AfNN is able to self-generate network durability.

In this study, we investigate the durability of the AfNN when some of the neurons in the hidden layer are damaged, after the learning process for a pattern recognition. Where the durability means strong, tough and tenacious for any damages from outside. Namely, the durability shows ability that network can still operate as well as before the neurons are damaged. We study the effect of two types of damaging. First, the connections to the output layer of the damaged neurons are cut. Namely, the damaging neuron does not operate (cutting-off damage). Second, the connections to the output layer of the damaged neurons produce random values (random performance damage). The learning example of the pattern recognition is 4 patterns of alphabet (**B, U, C, S**). Each pattern is composed of 7x5 neurons. By computer simulations, we confirm that the AfNN keeps its efficiency of recognition rate. We conclude that the affordable neurons exert an important influence on durability when the AfNN is applied to the pattern recognition.

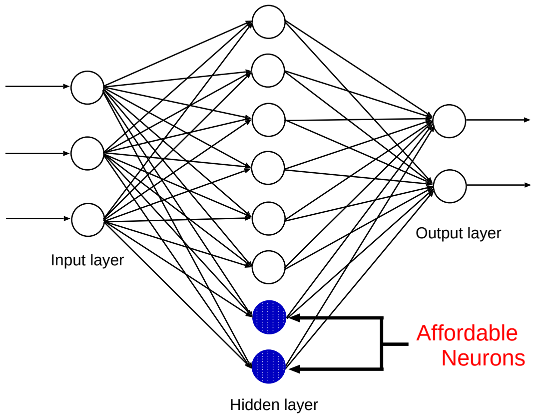
II. Affordable Neural Network

For our investigations, we use [1] the standard three-layered feedforward neural network. The output characteristics of the neurons are implemented by a sigmoid function and a subset of the neurons are tagged as affordable neurons. For convenience, the network model and the operation of the affordable neurons in the hidden layer are shown in Fig. 1(a) and Fig. 1(b), respectively.

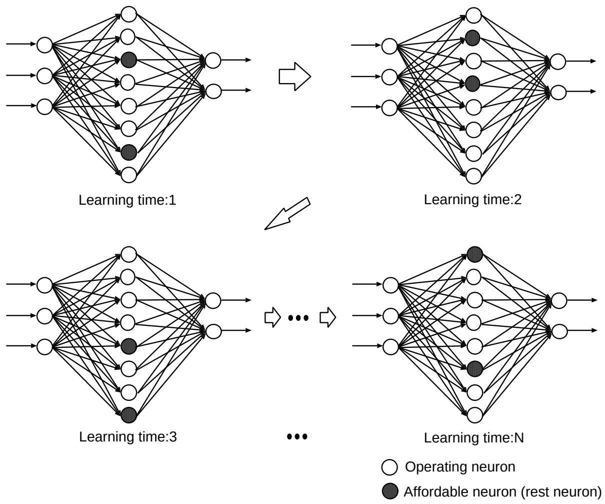
In the proposed BP learning network, at every input pattern, a subset of the affordable neurons is randomly chosen and silenced by cutting off their output. The temporal evolution that this network now undergoes is exhibited in Fig. 2, where the hidden layer consists of eight neurons. Out of these eight neurons, two are randomly selected and are silenced at each learning step (dots in Fig. 2).

In this study, we use the batch BP learning algorithm. This learning algorithm can be cast in a formula similar to the standard [2] BP learning algorithm. The difference lies in the timing of the weight. Whereas the update of the standard BP

SSJW' 08
Kagawa



(a) Network model with affordable neurons.



(b) Operation of the affordable neurons. In the proposed BP learning network, at every input pattern, a subset of the affordable neurons is randomly chosen and silenced by cutting off their output. The temporal evolution that this network now undergoes is exhibited in Fig. 2, where the hidden layer consists of eight neurons. Out of these eight neurons, two are randomly selected and the silenced at each learning step (dots in Fig. 2).

Fig. 1. Affordable neural network.

is performed after each single input data, in the batch BP, the update is performed only after all input data has been processed once. The total error E of the network is defined as

$$E = \sum_{p=1}^{P} E_p = \sum_{p=1}^{P} \left\{ \frac{1}{2} \sum_{i=1}^{N} (t_{pi} - o_{pi})^2 \right\}, \qquad (1)$$

where P is the number of the input data, N is the number of the neurons in the output layer, t_{pi} denotes the value of the desired target data for the pth input data, and o_{pi} denotes the value of the output data for the pth input data. The goal of the learning is to set the weights between the neurons so as to minimize the total error E. In order to minimize E, the

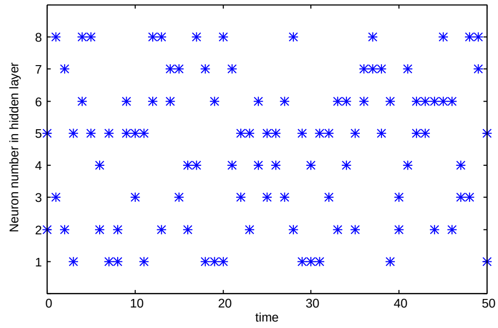


Fig. 2. Random selection.

weights are adjusted according to:

$$\begin{aligned} w_{i,j}^{k-1,k}(m+1) &= w_{i,j}^{k-1,k}(m) \\ &\quad + \sum_{p=1}^{P} \Delta_p w_{i,j}^{k-1,k}(m), \\ \Delta_p w_{i,j}^{k-1,k}(m) &= -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}}, \end{aligned} \qquad (2)$$

were $w_{i,j}^{k-1,k}$ is the weight between the ith neuron of the layer $k-1$ and the jth neuron of the layer k. m is the learning time, and η is a proportionality factor known as the learning rate. In this study, to the third line of Eq.(2) an inertia term was added, which leads to

$$\Delta_p w_{i,j}^{k-1,k}(m) = -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}} + \zeta \Delta_p w_{i,j}^{k-1,k}(m-1), \qquad (3)$$

where ζ denotes the inertia rate. The inertia term is introduced for efficient learning convergence.

III. Damaging Neurons

We now assume that after the BP learning phase, some neurons in the hidden layer are damaged according to the following two paradigms.

A. *Cutting-off damage*

In this paradigm, the connections of the damaged neurons to the output layer are temporally cut off, see Fig. 3.

B. *Random performance damage*

In this paradigm, the connections to output layer of the damaged neurons take random values from the unit interval $[0, 1]$. The diagram of this model is shown in Fig. 4. This situation is similar to the case where some neurons operate randomly, seemingly without any order or control.

IV. Simulated Results

In this study, we consider a pattern recognition task, where 4 diagrammic letters (**B**, **U**, **C**, **S**) are fed into the neural network for recognition (see Fig. 5).

In this case, the number of neurons in the input layer is 35 The number of neurons in output layer is fixed to 4, and we choose 16 hidden layer neurons. For recognition, a set

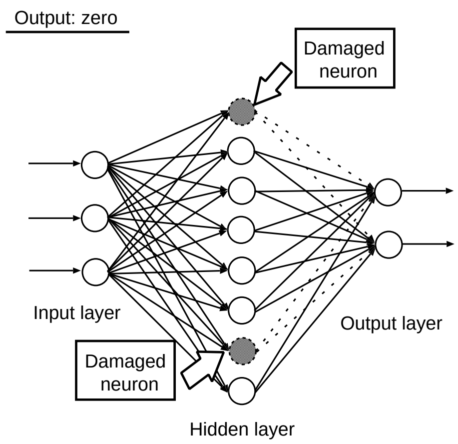


Fig. 3. Damaging neurons: cutting-off damage.

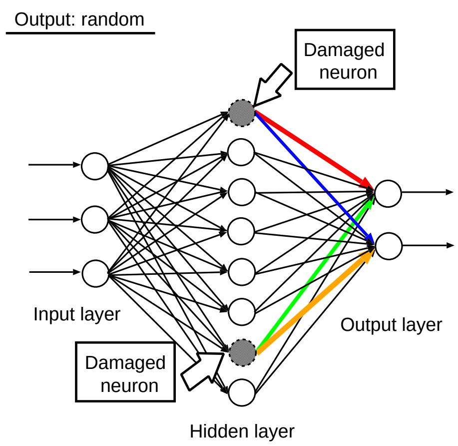


Fig. 4. Damaging neurons: random performance damage.

of 25 patterns shifted 1 bit from each original pattern was prepared, leading a set of 100 patterns to be recognized. The learning rate and the inertia rate were set to $\eta = 1.0$ and $\zeta = 0.8$, respectively, and the initial weights were uniformly randomly chosen from the interval $[-1, 1]$. The learning time was set to $m = 10000$ steps. By using 100 different network initial conditions, we obtained reliable averages characterizing the network performance in terms of the recognition rate R defined as

$$R = \frac{Number\ of\ sucsess}{Total\ number\ of\ input\ patterns}. \quad (4)$$

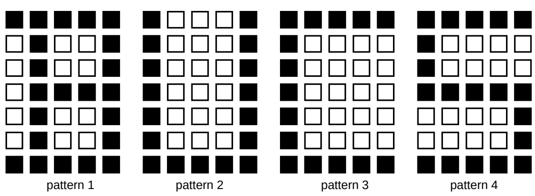


Fig. 5. Pattern recognition.

A. Cutting-off damage

Figure 6 displays the recognition rate for the AfNN and conventional BP network. From this figure, we infer that for this task, the recognition rates of the AfNN and the conventional network are similar, if only one neuron is damaged. Upon increasing the number of damaged neurons, the difference between the recognition rates, however, increases, leading to a noticeable higher recognition rate of the AfNN, where the difference of recognition rate between the both networks is larger than 10 percent.

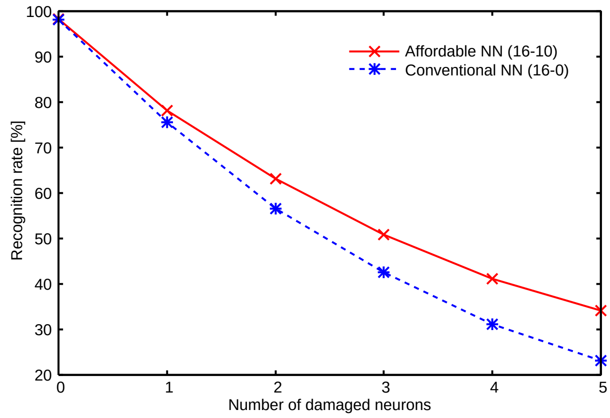


Fig. 6. Recognition rate (Damaging neurons: cutting-off damage).

B. Random performance damage

The results obtained for the pattern recognition problem in the case of "random performance damage" are shown in Fig. 7. Clearly, the influence of a damaged neuron in the AfNN is again smaller than that of a damaged neuron in the conventional BP network. Also for this task, the AfNN demonstrates a superior performance to that of the conventional BP network.

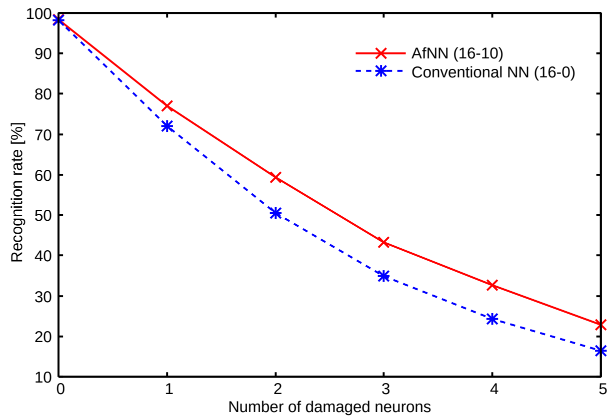


Fig. 7. Recognition rate (Damaging neurons: random performance damage).

V. Conclusions

For two rather different types of realistic neuron damages, similar overall performance was observed, where the performance of the AfNN (at least in the statistical sense) strongly outperformed the conventional BP network, which demonstrates the beneficial influence of affordable neurons for the reliable performance of neural networks under realistic working conditions. It is thus obvious that the affordable neurons exert an important influence on the durability of neural networks for pattern recognition task.

In future work, we will investigate the effects by chaotic selection methods, that more realistically model the selection of affordable neurons in the context of more complex problems. This view is motivated by investigations of the Hopfield network solving combinatorial optimization problems with the help of a chaotic input signal component, designed in order to avoid local minima. It appears, from computer simulations, that a chaotic input component may substantially enhance the capability of avoiding these local minima in comparison to random noise [3]-[5]. Hence, we believe that chaotic selection may be used to further enhance the efficiency of the proposed AfNN.

REFERENCES

[1] Y. Uwate and Y. Nishio, "Performance of Affordable Neural Network for Back Propagation Learning," *IEICE Transactions on Fundamentals*, vol. E89-A, no. 9, pp. 473-478, Nov. 2005.

[2] D.E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing*, vol. 1, MIT Press, MA, pp. 318-362, 1986.

[3] Y. Hayakawa and Y. Sawada, "Effects of the Chaotic Noise on the Performance of a Neural Network Model for Optimization Problems," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*,, vol. 51, no. 4, pp. 2693-2696, April. 1995.

[4] T. Ueta, Y. Nishio and T. Kawabe, "Comparison between Chaotic Noise and Burst Noise on Solving Ability of Hopfield Neural Networks," *Proc. NOLTA'97*, vol. 1, pp. 409-412, Nov. 1997.

[5] Y. Uwate, Y. Nishio, T. Ueta, T. Kawabe and T. Ikeguchi, "Performance of Chaos and Burst Noise Injected to the Hopfield NN for Quadratic Assignment Problems," *IEICE Transactions on Fundamentals*, vol. E87-A, no. 4, pp. 937-943, Apr. 2004.