Denoising Auto Encoder with Intermittency Chaos to Express Space Features

Ryuta Yoshimura, Shinsaburo Kittaka, Yoko Uwate and Yoshifumi Nishio

Dept. Electrical and Electronic Engineering, Tokushima University

{yoshimura,kittaka,uwate,nishio}@ee.tokushima-u.ac.jp

Recently, deep learning is used as commercial services and it becomes hot topic. It is difficult to learn whole network, because deep learning has complicated network. It is known that giving good initial values in advance is effective for learning the whole network. Auto encoder is used to give initial values.

Auto encoder has three layers. They are input, hidden and output layers. The goal of the auto encoder is to obtain the output values to match the input values. The number of neurons in the hidden layer is smaller than those of input and output layers. When input data are sent to the hidden layer, they are moved to a dimensional space. This process is called dimensional reduction. So we can think of input layer as an encoder because it compresses data. Then, output layer as a decoder is try to reconstruct the original data by using relation between the hidden layer and the output layer. Also we use input data with the noise to obtain more robust value. This method is called denoising auto encoder and we obtain good values for deep learning [1] . Equation (1)

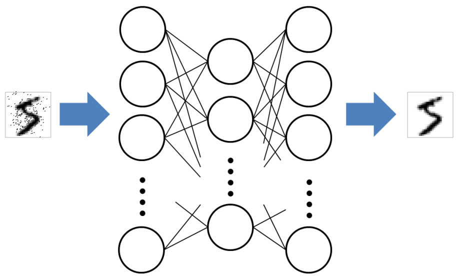


Figure 1: Denoising auto encoder.

shows encode and decode.

$$\begin{cases} y = s(Wx + b) \\ z = s(W'y + b') \end{cases} \tag{1}$$

x means input data. y means encoded information. z means reconstructing data from x. s means sigmoid function. W and W' mean weight. We define W_x equal W'_x by tied weight. We update the weight so that output z becomes closer to input x with calculating an error function. After updating the parameters W, b and b' the network calculates to minimize error function. The cross entropy (2) is used as error function.

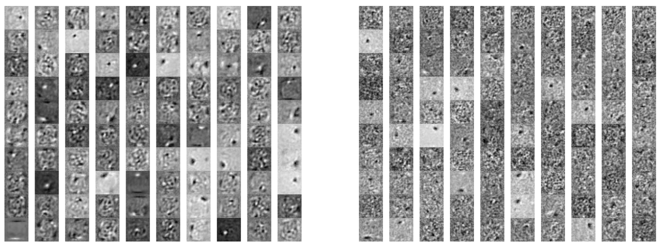
$$L_H(x, z) = -x \log z - (1 - x) \log(1 - z) \tag{2}$$

In this study, we use logistic map to make noise. Equation (3) shows logistic map.

$$f(x_{n+1}) \quad = \quad ax_n(1 - x_n) \tag{3}$$

Parameter a contrals the logistic map behavior. We set the parameter a as 3.828327 and use the intermittency chaos. We generate a random number by the logistic map and compare the number with the threshold. When it exceeds a threshold, it output 0. The output is multiplied by each pixel of the input data. The pixel which is multiplied 0 is painted black and becomes a noise. We use the logistic map for binarization. In this study, we forcus on weight in network due to difference of the number of neurons in hidden layer.

We set the number of neurons between 100 and 600 and visualize them in Fig. 2. As the number of neurons in hidden layer decrease, every neuron works and express features each other. Visualization of weight with 100 neurons has more space features than one with 600 neurons.



(a) (b)

Figure 2: Visualization of weight due to the number of neurons. (a) 100. (b) 600.

References

[1] P. Vincent, H. Larochelle, Y. Bengio and P. Manzagol, “Extracting and composing robust features with denoising autoencoders,” CML ’08 Proceedings of the 25th International Conference on Machine Learning, pp.1096-1103, 2008.