https://doi.org/10.20965/jaciii.2015.p0225

Paper:

Neural Network Structure Analysis Based on Hierarchical Force-Directed Graph Drawing for Multi-Task Learning

Atsushi Shibata*, Fangyan Dong, and Kaoru Hirota***

*Department of Computational Intelligence & Systems Science, Tokyo Institute of Technology
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan
E-mail: {shibata, hirota}@hrt.dis.titech.ac.jp
**Education Academy of Computational Life Sciences, Tokyo Institute of Technology
J3-141, 4259 Nagatsuta, Midori-ku, Yokohama 226-8501, Japan
E-mail: tou@acls.titech.ac.jp

[Received June 21, 2014; accepted December 4, 2014]

A hierarchical force-directed graph drawing is proposed for the analysis of a neural network structure that expresses the relationship between multitask and processes in neural networks represented as neuron clusters. The process revealed by our proposal indicates the neurons that are related to each task and the number of neurons or learning epochs that are sufficient. Our proposal is evaluated by visualizing neural networks learned on the Mixed National Institute of Standards and Technology (MNIST) database of handwritten digits, and the results show that inactive neurons, namely those that do not have a close relationship with any tasks, are located on the periphery part of the visualized network, and that cutting half of the training data on one specific task (out of ten) causes a 15% increase in the variance of neurons in clusters that react to the specific task compared to the reaction to all tasks. The proposal aims to be developed in order to support the design process of neural networks that consider multitasking of different categories, for example, one neural network for both the vision and motion system of a robot.

Keywords: neural network, network structure, multi-task learning, clustering, visualization

1. Introduction

For multitask learning by neural networks with deep layers, encouraging classification accuracy is studied [1, 2], where lower layers close to the input layer of a neural network are formed via unsupervised learning for the purpose of feature extraction, and then the entire neural network is learned via a combination of those partially structured lower layers with higher layers. Some parameters, such as iteration times of unsupervised learning and number of neurons in each layer, have to be determined by two methods without any guidance from clear criteria. One is an empirical decision that chooses the most plausible number between the upper and lower layers. Another is a pruning method that deletes neurons connected to weak weights. The former requires an experiment and trial-and-error. The latter can determine a suitable size, which is lower than the initial size, through pruning, but it cannot identify the role of deleted neurons in multitask. Moreover, the high learning cost of deep neural networks and time required for trial-and-error result in significant computational costs for determining a suitable neural network structure. It is necessary to visualize the neurons in a task-related pattern as guiding criteria for adjusting network size.

Therefore, a hierarchical force-directed graph drawing is proposed for neural network structure analysis by creating neuron clusters in 2D Euclidian spaces, in which the placement of neurons is determined using connection weights between neurons. Explained in detail, an attractive force acts on neurons located on two adjacent layers, and a repulsive force acts on neurons in each layer. The force applied between neurons is used to update neuron lactations during iteration, and constructed neuron distributions are defined as the neuron cluster. The variance of neurons in clusters that react to input data is designed as the criteria for adjusting the number of neurons in each layer.

By applying our proposal, those with insufficient experience designing neural network structures can check the clusters of visualized neurons and use their variance as the criteria for determining the neural network structure. Given a specific task, the trial-and-error process for determining its network structure also benefits from a lower computational cost.

The proposal is evaluated by experiments with neural networks learned on the Mixed National Institute of Standards and Technology (MNIST) database that contains 70,000 handwritten digits of ten categories. In the first experiment, the average variance of the neurons in clusters that react to the input data is calculated in each layer to show its relationship with the status of the learning process. In the second experiment, the training data of one specific task (i.e., digit "1") is cut in half, and the variance of neurons in clusters that react to this task is calculated in the middle layer, which is adjacent to the input layer of

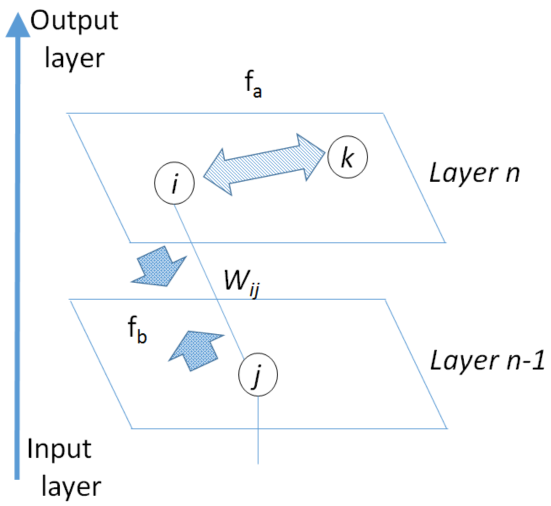


Fig. 1. INeuron cluster in 2D Euclidean representation of each layer in neural network.

the network, and the value is compared with the average variance of neurons in the cluster that corresponds to all tasks in the same layer.

The visualization method for showing neuron clusters in each layer and the evaluation criteria for network size are proposed in Section 2. Evaluation experiments that use handwritten digits to show the relationship between evaluation criteria with each task are covered in Section 3.

2. Neuron Cluster and Evaluation for Neural Network Structure

2.1. Visualizing Neurons into Clusters by Hierarchical Force-Directed Graph Drawing

To clarify the correspondence between neural network processing and its structure, the relationship between neurons in each layer is expressed in the 2D Euclidian space by assigning their location using a force-directed graph drawing [3], and the proposed visualization in **Fig. 1** is utilized to generate a clustering algorithm for a hierarchical structured network.

As a preprocessing step, the weights in the neural network are normalized. Then, the following two steps are iterated until the neuron movement velocity is reduced to a certain value.

(1) Forces related to neural network weights are calculated and applied to neurons.

(2) Neuron velocity and location are updated using applied forces.

First, for equitable comparison of neurons with different ratio between their weights and biases, the connection weight W_{ij} between neuron i and neuron j is normalized using bias b_i of neuron i as

$$W_{ij} := \frac{W_{ij}}{1+I(b_i)}, \quad (1)$$

$$I(b_i) = \begin{cases} b_i & \text{if } b_i > 0 \\ 0 & \text{otherwise} \end{cases} . \quad (2)$$

This normalization process is based on the idea that neuron firing is discriminated by whether its value is higher than the bias. If b_i is less than or equal to zero, neuron i fires regardless of the state of neuron j. However, if b_i is a positive value, the ratio of W_{ij} to b_i is important in deciding the state of neuron i.

Second, an attractive force is applied on neuron pairs in adjacent layers, and a repulsive force is applied on neuron pairs in the same layer. For two neurons on adjacent layers with weight W and distance d, their connection is regarded as a stretched spring. Attractive force F_a is calculated based on Hooke's law as

$$F_a = \begin{cases} -Wd & \text{if } d < d_1 \\ -Wd_1 & \text{otherwise} \end{cases}, \quad (3)$$

where d_1 is a limitation that prevents excessive increase in the velocity and divergence of neuron positions, and d_1 is set to one from experience. The repulsive force F_b applied on neuron pairs on the same layer is calculated in reference to the cubic function approximation of the van der Waal force as

$$F_b = \begin{cases} \left(\frac{5}{4}d^3 - \frac{19}{8}d^2 + \frac{9}{8}\right) & \text{if } d < 1 \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

Force F_b is designed as a local repulsive force that works in a radius range of one. If the distance between two neurons is over the range, the value of F_b is set to zero in order to avoid a negative value.

Finally, the velocity and position of all neurons are updated using Eqs. (3) and (4) by

$$V := \theta\left[V + dt\left(F_a + F_b\right)\right], \quad (5)$$

$$x := x + dtV, \quad (6)$$

where θ is a decay coefficient, and dt is a short period of time.

Neuron distributions are suitably constructed as clusters by iterating Eqs. (3) to (6) until velocity change decreases to a termination threshold. In this paper, θ, dt, and the termination threshold are set to 0.7, 0.1, and 0.01, respectively.

2.2. Neural Network Structure Evaluation from its Cluster

The size of a neural network and its learning process are estimated using neuron clusters in the 2D Euclidian space obtained in Section 2.1.

Neural network processing is supposed to be performed by firing the neurons that react to an input signal, and these neurons pass their information to the neurons in the upper layer via positive weights. At the same time, the information via negative weight is used to inhibit other neurons. Then, the weights converge to specific values during the learning progress.

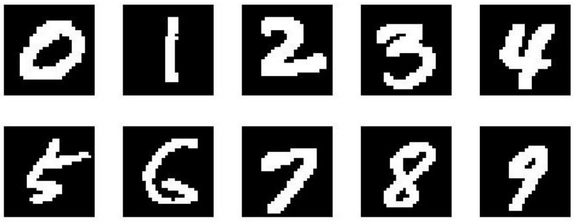


Fig. 2. Ten number image examples in MNIST dataset.

In the neural network that learned a task, the neurons that react to the task receive stronger information from a neuron in the lower layer, and these neurons are located in the vicinity and create clusters. For the same reason, the neurons not related to any tasks are connected by weak weights and are removed from the clusters. Therefore, the neural network size in each layer is estimated by the variance of the neurons in the clusters that react to the tasks. Furthermore, the neurons located sparsely around the clusters are marked as unnecessary in the estimation process.

3. Neural Network Estimation Experiments from Cluster Reaction

3.1. Preparation of MNIST Database and Neural Network Parameters

The proposal is evaluated by the clusters in a visualized neural network that already learned multitask, where multitask means the serial data, and each task is managed individually. To evaluate the proposed estimation methods for learning the progress and size of neural networks, we confirm that the neurons located sparsely around clusters are not necessary because these neurons do not react to any tasks, and the variance of neurons in a cluster that reacts to a task is reduced in accordance with the decrease in learning error. In addition, the relationship between each visualized cluster and its corresponding task is also confirmed by an experiment in which training data on one specific task is cut in half, and the variance of neurons in the cluster that reacts to that task is studied.

The MNIST database [4] commonly used in multitask learning is an image database of handwritten digits, as shown in **Fig. 2**. In the database, each example has a 784-pixel (28×28) binary image and number labels (from zero to nine). The database has a training set of 60,000 examples, and a test set of 10,000 examples. MINST is a subset of a larger set available from NIST. Each digit has been normalized in size and centered in an image with fixed size. In this paper, one task is characterized by identification of the number label to each input image, and 10,000 examples are retrieved from the training data and used as validation data to evaluate the variance of neurons in the 2D Euclidean space, and calculate the learning error that represents learning progress.

A denoising autoencoder [5] is used as the neural network model for multitask learning. In this paper, learning rate, learning decay, momentum, and mini-batch size are set to 1, 0.98, 0.5, and 50, respectively. The number of neurons is fixed to 784 in the input layer and ten in the output layer.

3.2. Neural Network Structure Estimation from its Cluster

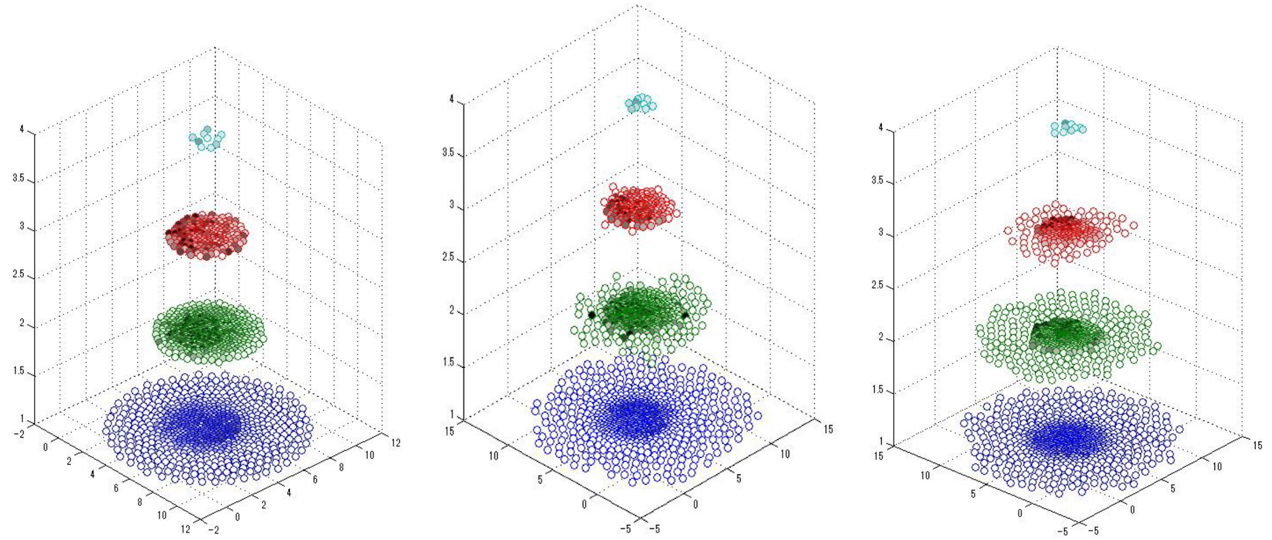
The neural network structure and its learning progress are evaluated through experiments confirmed by the reaction of the neuron cluster expressed in the 2D Euclidean space, where the four-layered neural network is used to verify correspondence between clusters in upper and lower middle layers.

Before evaluating the proposal, we show how a neural network is visualized during the training set learning in **Fig. 3**. In **Fig. 3**, three neural networks consist of the same size 784-500-200-10, and **Figs. 3(a)**, **3(b)**, and **3(c)** are in the different epochs of one, five, and 50 trial times where the neural network is learning the training set. The neurons are divided into four layers arranged in order from the input layer side, and the clusters are visualized circles filled in monochrome according to the firing frequency that reacts to the validation set. (In other words, neurons that do not react to any tasks are painted in white. Conversely, neurons that react independently of the task are painted in black.)

Through the learning progress shown in **Figs. 3(a)** to **3(c)**, specific neurons fire more frequently, which is observed by that the neurons in center is darker than other neurons in the surrounding. In addition, the weight between inactive neurons not related to any tasks becomes lower, which is observed by the whitecolored neurons located sparsely around the center, and their arrangement becomes uneven given that their weight is not sufficient to work as an attraction force (too weak). From these results, the trend of changes in the neural network state can be represented in the 2D Euclidean space.

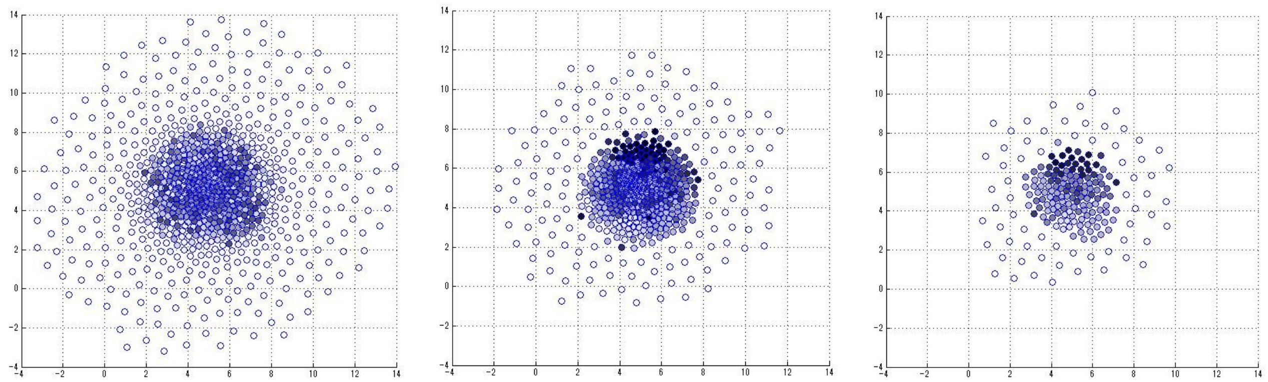
In the first experiment, visualization of the neural network is evaluated by visualizing the neural network according to the neuron reaction to all tasks (i.e., the digits from “0” to “9”). Furthermore, the relationship between the error rate and variance of the neurons on each layer (strongly related with suitable neural network size) is shown with a learning progress.

To discuss more details of the relationship between the visualized neural network and its size in each layer, three layers (one input layer and two middle layers) of the visualized neural network from **Fig. 3(c)** are shown in **Fig. 4** for clarity. **Figs. 4(a)**, **4(b)**, and **4(c)** are the input layer, lower middle layer, and upper middle layer, respectively. In the two middle layers, neurons with similar color depth are arranged collectively, and their cluster approximates a slightly distorted round shape. On the other hand, neurons in the input layer have uniform color in the center, and they are not constructed as a clear cluster, but as a nearly round form. The reason for this is that the neurons that correspond to the center of the input image react equally to any tasks and are connected evenly to the neu-



(a) (b) (c)

Fig. 3. Euclidean positions of neurons (a) epoch = 1, (b) epoch = 5, (c) epoch = 50.



(a) (b) (c)

Fig. 4. Clusters on two middle layers (a) input layer, (b) lower middle layer adjacent to output layer, (c) upper middle layer adjacent to input layer.

rons in the lower middle layer. In addition, the neurons that correspond to the corners of the input image are removed from the clusters.

To summarize, we confirm by the color gradation in clusters that neurons with similar firing frequency on tasks are arranged in a group as a cluster, and that neural network size is thus discriminated to be smaller in the given case.

Furthermore, to show the relationship between neuron clusters and the learning process, the average variance of the neurons in clusters that react to all tasks is calculated in the two middle layers for epochs one to 12, and the trend of the variance obtained is compared with that of the learning error calculated by supervised data based on the standard cross-validation method shown in **Fig. 5**. In **Fig. 5**, the variances of both middle layers are gradually reduced after an initial rise on their curves, and the learning error also decreases gradually with some local oscillation. However, the variance tendency decrease in the upper middle layer is more remarkable than that of the lower middle layer. This is attributed to the characteristic of the back propagation learning method, in which weight changes are sufficiently weak in layers farther from the output layer.

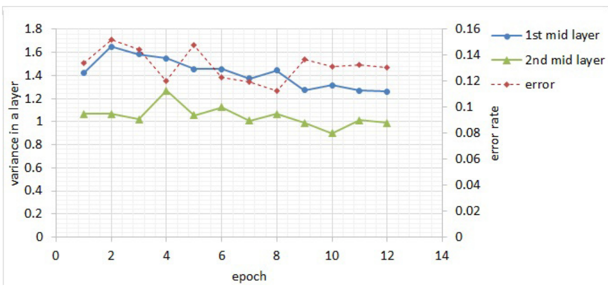
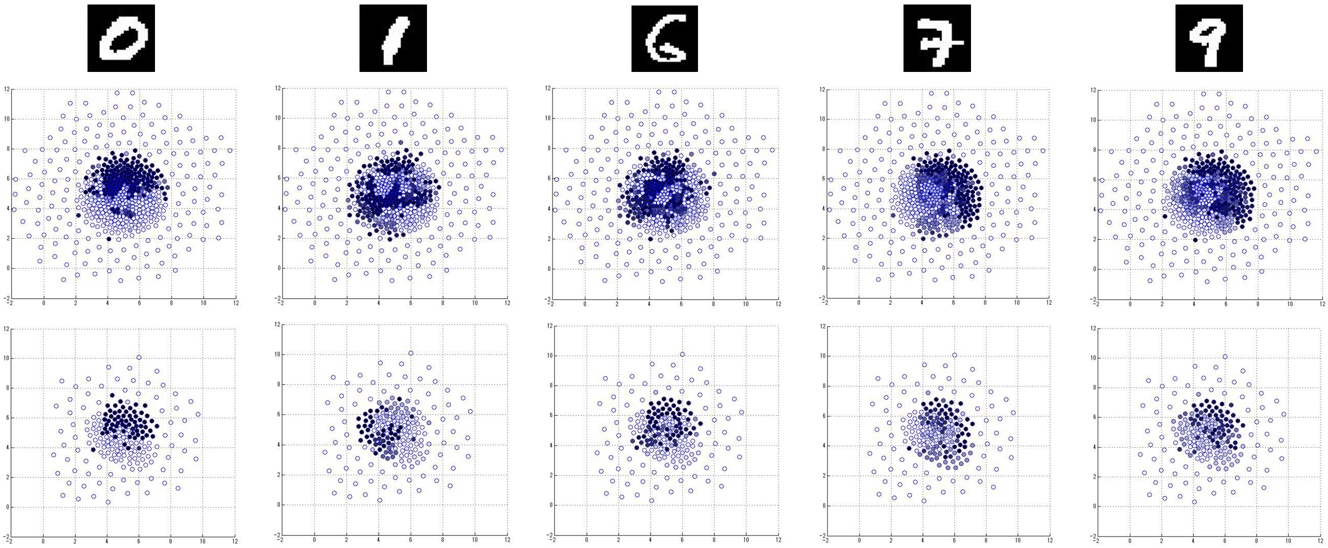


Fig. 5. Epoch variances of neurons in each layer compared with error rate based on cross-validation.

To summarize, the overall trend of the learning progress can be presented indirectly by the variances of neurons in each layer without help of supervised data, but only an approximate change in the learning progress can be obtained using variance as criteria.



(a) (b) (c) (d) (e)

Fig. 6. Neuron clusters in one neural network colored according to their reaction to each specific task from five tasks, five images in first row are examples of input images of each number task, five neuron clusters in second row are expressed reacting neurons to each task in upper middle layer, and five neuron clusters in third row are expressed reacting neurons to each task in lower middle layer. (a) task0, (b) task1, (c) task6, (d) task7, (e) task9.

In the second experiment, reaction of neurons to each task is evaluated by neuron clusters, where *task0* to *task9* correspond to identifications of the digit numbers from "0" to "9," respectively. Furthermore, neuron clusters that react to one specific task and their neuron variance trend are shown with reduced training examples by comparison of **(a)** and **(e)** in **Fig. 6**.

To observe the relationship between neuron clusters and tasks, the neurons that react to specific tasks in the two middle layers of **Fig. 3(c)** are shown in **Fig. 6**, where **Figs. 6(a)**, **6(b)**, **6(c)**, **6(d)**, and **6(e)** express the neurons that react to task0, task1, task6, task7, and task9, respectively; a higher reacting frequency is indicated with lower color depth. In addition, the learning error that corresponds to each task in **Fig. 6** is also listed in **Table 1**. By checking the mutual distances of reacting neurons in the lower layer in **Fig. 6**, the neurons that react to a specific task are arranged into the same cluster. In particular, the neurons that react to *taks0*, which share the lowest learning error in **Table 1**, are clearly divided by neuron color depth in **Fig. 6(a)**. On the other hand, the neurons that react to *task7*, which are of the highest learning error in **Table 1**, form a dispersive cluster in the 2D Euclidean space, especially in the upper middle layer.

The two neuron clusters that react to *task7* and *task9*, for which both handwritten digits are similar in their visual perception, share an overlapping part intermediately in the lower middle layer. On the other hand, the two neuron clusters that react to *task0* and *task1*, for which both handwritten digits are different in their visual perception, do not have a significant overlapping part. Thus, the *task6* cluster of reacting neurons in the lower layer is similar to a combined cluster by *task0* and *task1*, and it is consistent with the shape of the handwritten digits. However, in the upper layer, three clusters are not overlapped, which means that clusters in the upper layer are more specific to the corresponding tasks than in the lower layer. We also confirm that the neurons located sparsely around the center are unrelated to any tasks, and thus such neurons are possible candidates for deletion.

Table 1. Error rate in each task from **Fig. 6**.

task0	*task1*	*task6*	*task7*	*task9*
0.0245	0.0317	0.0428	0.1031	0.0763

We conclude that neuron clusters configured by learned neurons are constructed from lower to upper layers that correspond to the identification process of handwritten digits, and that neuron clusters configured by neurons under learning are detected when the variance of the neurons in the cluster becomes large, e.g., "7" in **Fig. 6**.

In order to check how neuron clusters change in accordance with the size of the training data for related tasks, the neuron cluster in the lower middle layers of two neural networks trained on the entire training set and lacking training set, respectively, are shown in **Fig. 7**.

Figures 7(a) and **7(b)** show the neuron clusters that react to *task0* and *task1* in the same neural network trained on the entire training set (50,000 images of handwritten digits) in 12 epochs. **Figs. 7(c)** and **7(d)** show the neuron clusters that react to *task0* and *task1* in the neural network trained on a reduced training set (47,500 images of handwritten digits with images from digit "1" cut in half, namely, 2,500 images of "1") of 12 epochs. By comparing **Figs. 7(a)** and **7(c)**, we can see that the neuron clusters show a similar cluster. However, in the comparison of **Figs. 7(b)** and **7(d)**, the neurons that react to *task1* have

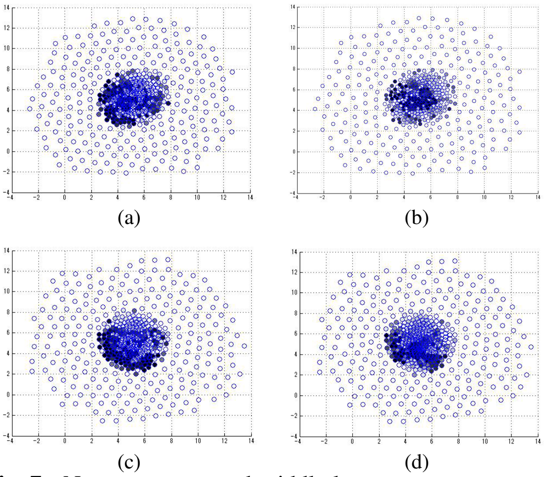


Fig. 7. Neurons on second middle layer response to particular task. (a) response to *task0*, learning with entire data, (b) response to *task1*, learning with entire data, (c) response to *task0*, learning with lacking *task1* data, (d) response to *task1*, learning with lacking *task1* data.

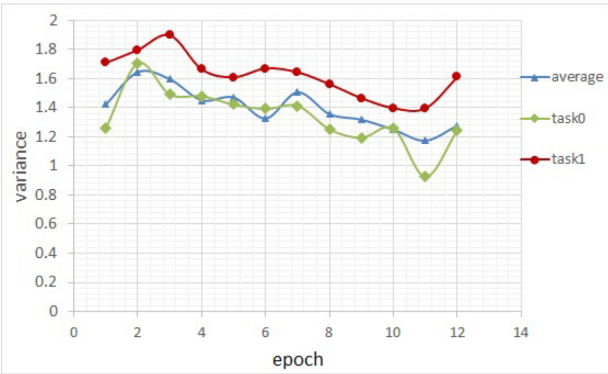


Fig. 8. Three neuron variances in each cluster react to **Figs. 6(a)**, **6(b)**, and ten tasks in each epoch.

a more dispersive cluster in **Fig. 7(d)** than in **Fig. 7(b)**. This is because the neurons that correspond to *task1* have not learned with sufficient data, given the data shortage, in comparison to *task0*. To show the change of the neuron clusters that react to all tasks compared with those that react to one specific task using the reduced training data set, the variance of the neurons in the clusters is calculated from a neural network that learned on a training set (47,500 images for ten digits) where the images for *task1*, namely digit "1," is cut in half (2,500 images), and it is repeated from epochs one to 12, as shown in **Fig. 8**. In **Fig. 8**, the average variance of neurons in the cluster that reacts to *task0* shares almost the same trend with that of neurons that react to all tasks. However, the variance of neurons in the cluster that reacts to *task1* is always greater than the average variance of neurons by approximately 15%. To summarize, a relationship between tasks learned by neural network is inferred by the variance of neurons in the clusters that react to specific tasks in the 2D Euclidean space.

3.3. Discussion Regarding Calculation Cost and Relationship with Network Structure

In the experiments, a period longer than 46,000 sec is required for the learning process of the neural network learning with a predefined size (784-500-250-10) under an environment where i7 CPU (3.5 GHz), 8GB RAM, and MATLAB are used. Furthermore, the calculation time for visualization is longer than 282 sec in the same environment. Therefore, there is a significant computational cost when finding a more suitable network size through trial-and-error. On the other hand, it is possible to reduce the computational cost using our proposal, which detects unnecessary neurons for learning during the learning process.

Incidentally, one of the difficulties for neural networks learning multitasking is caused by task processing conducted in a black box. When performing a task using a neural network, the task is processed in cooperation with a neuron simple calculation by the summation and activation functions. Therefore, the relationship between neuron and task is unclear, and it is difficult to identify the neuron role in the network process. If neurons are associated by their processing and tasks, they can realize more efficient learning. For example, a neuron is assigned to learn a specific task, and those neurons that have completed their learning are excluded from the ongoing learning process. Furthermore, our proposal aims to be applied so that the learning process is accomplished in less time by identifying the tasks that are not advancing, and learning intensively on such tasks.

4. Conclusion

In the identification task for handwritten digits using the MNIST database, we confirmed through the characteristics of neurons in the 2D Euclidean space that the neural network size in each layer is estimated as reasonable based on whether its clusters are sparse or dense, and that the learning progress for each layer is evaluated through variance reduction in the neuron clusters that react to tasks. Furthermore, we also verified that the neuron cluster represents the characteristics of each learned task by the neuron clusters that react to that task, and the variance of the neurons in the clusters that react to a specific task is 15% larger than the average variance when the training data for that task is cut in half.

From these results, we can state that in neural network learning experiments on general character identification, the proposal clarifies the network size, learning progress, and relationship among tasks. Thus, the proposal shows the criteria that make neural network structure design easier for those with insufficient experience designing the neural network structures, and the trial-and-error process for determining network structure also benefits from lower computational cost.

The proposal aims to be developed in order to support the design of neural networks that consider multitasking of different categories by managing neuron clusters related with each task. As a longterm objective, the proposal aims to be applied to multitasking identification in the real world, such as the visual and situation cognition tasks of a robot.

References:

[1] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochell, "Greedy Layer-Wise Training of Deep Networks," Advances in Neural Information Processing Systems 19, pp. 153-160, 2007.

[2] Y. Huang, W. Wang, L. Wang, and T. Tan, "Multi-task deep neural network for multi-label learning," 20th the Int. Conf. on Image Processing, 2013.

[3] T. Itoh, K. Inoue, J. Doi, K. Yasumasa, and I. Yuko, "An Improvement of Force-directed Graph Layout Method", Report of Information Processing Society of Japan, Vol.2001, No.35, pp. 7-12, 2001 (in Japanese).

[4] Y. LeCun, C. Cortes, and C. J. C. Burges, "THE MNITS DATABASE of handwritten digits," http://yann.lecun.com/exdb/mnist/ [Accessed August 2014]

[5] R. B. Plam, "Deep Learning Toolbox," https://github.com/rasmusbergpalm/DeepLearnToolBox [Accessed August 2014]

Name:
Atsushi Shibata

Affiliation:
Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

Address:
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan

Brief Biographical History:

2005-2009 B.E., Aoyama Gakuin University

2010-2012 M.E., Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

2012- Ph.D. Student, Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

Main Works:

- A. Shibata, J. Lu, F. Dong, and K. Hirota, "A Neural Structure Decomposition Based on Pruning and its Visualization Method," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.17, No.3, pp. 443-449, 2013.

Name:
Fangyan Dong

Affiliation:
Associate Professor, Education Academy of Computational Life Sciences, Tokyo Institute of Technology

Address:
J3-141, 4259 Nagatsuta, Midori-ku, Yokohama 226-8501, Japan

Brief Biographical History:

2006-2014 Assistant Professor, Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

2014- Associate Professor, Education Academy of Computational Life Sciences, Tokyo Institute of Technology

Main Works:

- F. Dong, K. Chen, E. M. Iyoda, H. Nobuhara, and K. Hirota, "Solving Truck Delivery Problems Using Integrated Evaluation Criteria Based on Neighborhood Degree and Evolutionary Algorithm," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.8, No.3, pp. 336-345, 2004.
- F. Dong, K. Chen, and K. Hirota, "Computational Intelligence Approach to Read-world Cooperative Vehicle Dispatching Problem," Int. J. of Intelligent Systems, Vol.23, pp. 619-634, 2008.

Membership in Academic Societies:

- Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
- The Japanese Society for Artificial Intelligence (JSAI)

Name:
Kaoru Hirota

Affiliation:
Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology

Address:
G3-49, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan

Brief Biographical History:

1982-1995 Professor, College of Engineering, Hosei University

1995- Professor, Tokyo Institute of Technology

Main Works:

- M. L. Tangel, C. Fatichah, M. R. Widyanto, F. Dong, and K. Hirota, "Multiscale Image Aggregation for Dental Radiograph Segmentation," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.16, No.3, pp. 388-396, May 2012.
- M. Dai, F. Dong, and K. Hirota, "Fuzzy Three-Dimensional Voronoi Diagram and its Application to Geographical Data Analysis," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.16, No.2, pp. 191-198, March 2012.

Membership in Academic Societies:

- The Institute of Electrical and Electronics Engineers (IEEE)
- International Fuzzy Systems Association (IFSA), Fellow, Immediate-Past-President
- Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT), Past-President