Advanced Learning Technologies for Deep Learning

Yasutoshi Ida, Sekitoshi Kanai, Yasuhiro Fujiwara, Satoshi Yagi, and Yasuhiro Iida

Abstract

NTT is focusing on artificial intelligence (AI) as a key technology in developing business strategies. The NTT laboratories have been engaged in research and development (R&D) of deep learning algorithms, which play a central role in AI. This article presents an overview of our R&D of novel algorithms that speed up and stabilize deep learning and touches on our collaboration with several partners to verify those algorithms.

Keywords: deep learning, algorithm, optimizer

1. Introduction

NTT places high priority on addressing various social issues and strengthening industrial competitiveness, and the use of artificial intelligence (AI) in our business endeavors in pursuit of these goals is an area of great importance to us. AI includes many technologies such as those used in statistical analysis, machine learning, and deep learning, which are applied to perform classification, regression, and prediction tasks with large amounts of data. Deep learning has been attracting a great deal of interest lately because it has achieved a practical level of accuracy in a variety of tasks. It has already been introduced to improve various business practices and is expected to become a driving force in the creation of new businesses.

Deep learning is a method to extract features from data hierarchically. A layered neural network is usually used as a model. Users select a suitable neural network according to a type of task or data. Convolutional neural networks are suitable for image data, while recurrent neural networks (RNNs) suit time-series data. In a neural network, the system makes predictions by giving weight to input data signals or by applying nonlinear transformation to the signals hierarchically and then having the signals propagate (**Fig. 1**). *Learning* is a process of adjusting weights to reduce the amount of error when using a large training dataset. In this process, the layered structure enables deep learning to achieve high accuracy in a variety of tasks.

However, the layered structure of deep learning models presents some problems such as an increase in learning time and destabilization of learning itself. Therefore, it is important to address this issue when making full use of deep learning techniques. To this end, the NTT laboratories have developed (1) an algorithm that improves learning efficiency and (2) an algorithm that stabilizes learning in an RNN [1, 2]. These algorithms are introduced in the following section.

2. Algorithm to improve learning efficiency

Learning of a layered neural network model is a process of adjusting weights to reduce the amount of error. This adjustment is performed gradually by applying a procedure as shown in **Fig. 2**. First, data are input into the model and a prediction result is obtained (forward propagation). Next, this prediction result is compared with the correct label, and the amount of error is calculated (error computation). The calculated amount of error is propagated to the model (back propagation), and the update direction of weights is calculated. The amount of updating is also

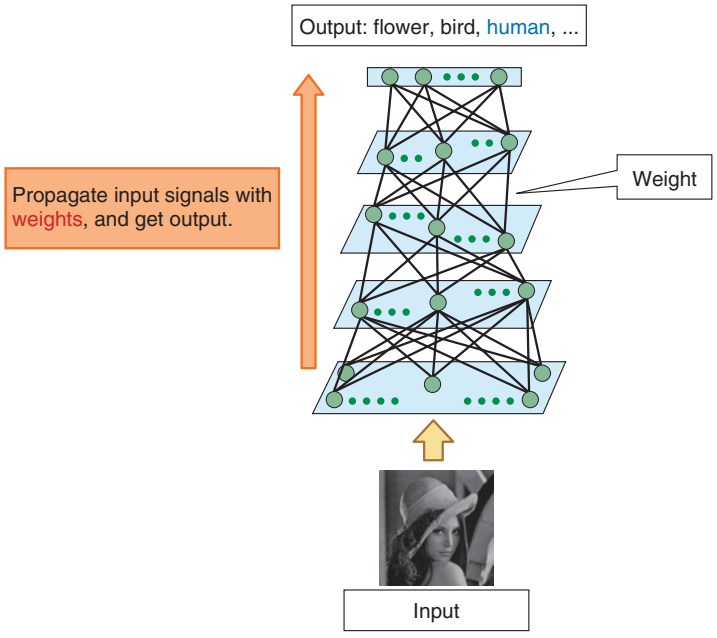


Fig. 1. Layered neural network.

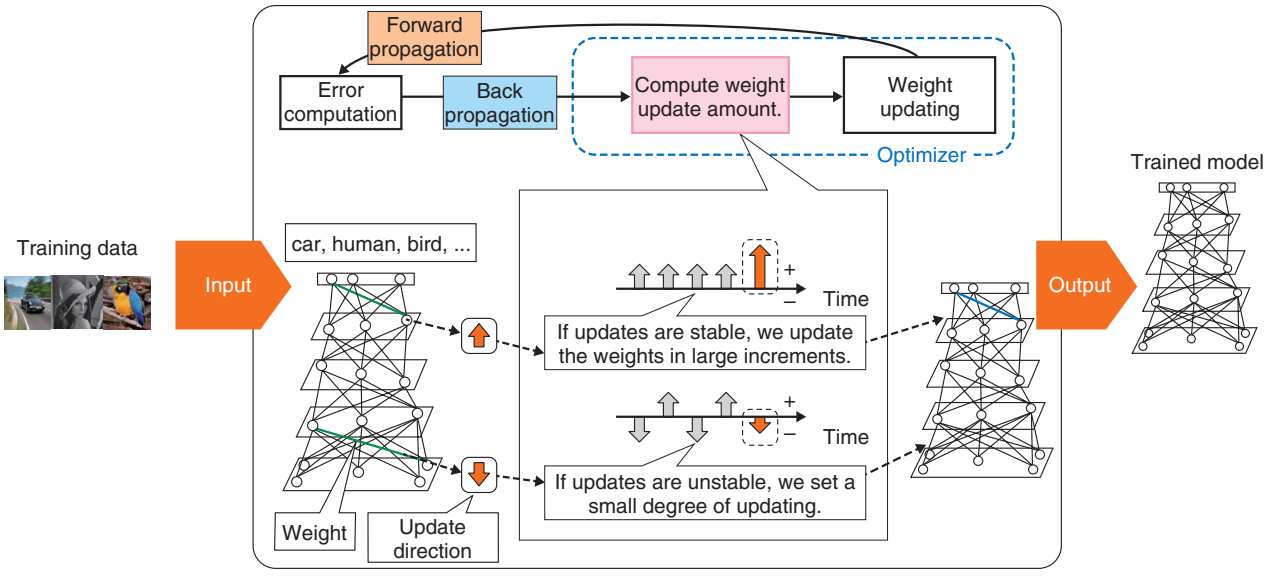


Fig. 2. Learning of a layered neural network model.

calculated (weight update amount computation). Finally, the weights are updated using this update amount (weight update). There are several approaches to calculate the weight update amount, and the learning efficiency varies depending on the approach used. In other words, the amount of error that can be reduced per loop depends on the approach used.

Several approaches adjust the update amount based on information about past update directions. For example, the widely used approaches RMSprop and Adam adjust the update amount based on statistical values calculated from the absolute values of the

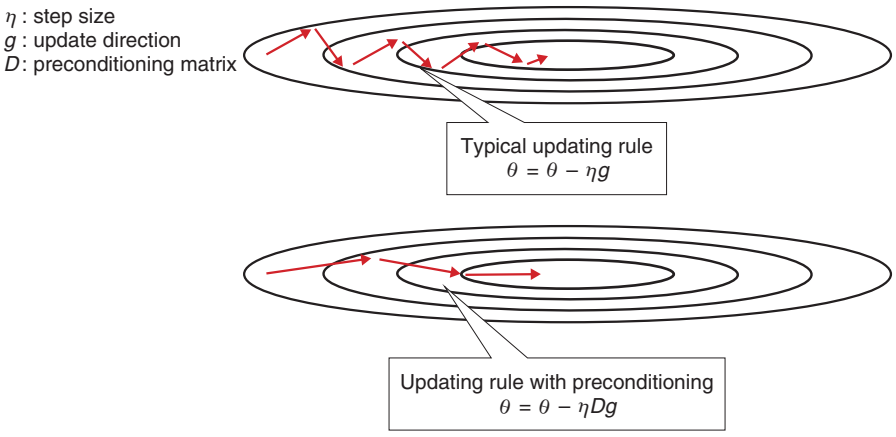


Fig. 3. Behavior of algorithm on loss function with respect to weights θ.

update directions. However, since these approaches use the absolute values of the update directions, they do not generally take variances of those values into consideration. This means that no matter how much variance there is in the update directions, the same update amount is set if the absolute amount is the same.

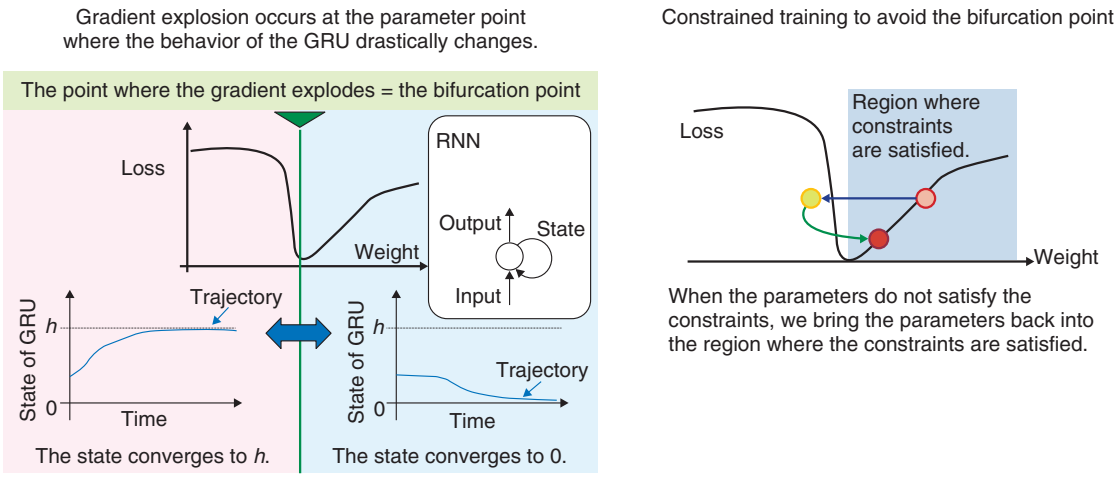
To address this problem, we have developed an algorithm that adjusts the update amount based on the variance in the update directions (Fig. 2). If the update directions vary significantly, it can be intuitively determined that learning is unstable. Therefore, the update amount is reduced to stabilize learning. If the variance of the update directions is small, the update amount is increased. This would increase the variance but would also increase the possibility of learning breaking away from a local solution, enabling it to further reduce the amount of error.

This algorithm is simple and can thus be easily implemented in numerous deep learning frameworks. It also has a theoretically interesting aspect. It can be regarded as a type of optimization with preconditioning. The preconditioning matrix that sets the optimization conditions becomes an approximation of the square root of the diagonal elements of a matrix calculated from gradients, which is called the Fisher information matrix. From the perspective of information geometry, which is a framework for visually understanding information processing problems, this fact suggests that an algorithm that repeats updates can converge faster than existing algorithms (**Fig. 3**).

3. Technique to stabilize training of gated recurrent unit (GRU)

A deep learning model called an RNN is used to handle time-series data such as those used in speech recognition or machine translation. The RNN memorizes information about past data as a state and calculates the output from this state and the current input. For example, consider a task of predicting the next word from a given text. The next word depends on the context of the preceding text. It can be predicted at a high level of accuracy by having the RNN memorize information about the past context as a state.

To process time-series data with a high degree of accuracy, it is necessary to preserve past information for the long term. Therefore, an important indicator of RNN performance is how long the RNN can memorize past information. A model structure called the long short-term memory (LSTM) was proposed in the late 1990s to achieve long-term memorization [3]. The LSTM has a structure called a memory cell that takes in past memories, and a gate structure that forgets unnecessary information so that old but important information can be saved while information that has become unnecessary can be forgotten. The LSTM has already been applied in several machine translation and speech recognition technologies. However, the LSTM has a complicated structure, so the simpler GRU, which combines the input gate and the forget gate of the LSTM into one, was proposed in 2014 [4]. The GRU has a simpler structure and requires less computation; however, it has been demonstrated empirically that the GRU achieves a level of accuracy almost equivalent to that of the



(a) Reveal the bifurcation point

(b) Avoid the bifurcation point

Fig. 4. Technique to stabilize training of GRU.

LSTM.

While the RNN can process time-series data at a high level of accuracy, its learning becomes unstable due to a phenomenon called gradient explosion [5]. This is a phenomenon whereby the gradient, which plays the main role in learning of neural networks, becomes so large as to cause learning to fail. An existing technique to counter this problem is gradient clipping, which clips gradients at a certain threshold value. This technique requires the threshold value to be adjusted through trial and error.

We are studying ways to solve the gradient explosion problem by analyzing the behavior of the GRU. From the perspective of dynamical systems, it can be said that a gradient explosion arises when a bifurcation occurs as a result of a change in parameters of the GRU (**Fig. 4**). A bifurcation is a phenomenon in which the behavior of the GRU changes dramatically when there is a small change in its parameters. NTT's algorithm for stabilizing the training of GRU involves analyzing the behavior of the GRU state to identify the point at which a bifurcation occurs. Furthermore, NTT has proposed a method of learning that efficiently avoids the bifurcation point. Thus, we are studying a method of learning that has a higher degree of accuracy and requires fewer trial-and-error attempts than gradient clipping [2].

4. Collaborations to refine NTT-developed advanced technologies

While devoting our energies to research, we also undertake activities to improve the effectiveness of deep learning technology in collaboration with a broad spectrum of users. For example, we have implemented our efficient learning technology to optimizers of multiple representative deep learning frameworks such as Chainer, TensorFlow, and Caffe, and we are evaluating our technology in collaboration with NTT Group companies. To date, we have examined the efficiency and accuracy of a model for an image recognition service in order to assess the model's feasibility. Through such examinations, we are accumulating know-how regarding the optimum sizes (batch sizes) of the input training dataset based on the data variety and the initial values of the learning rate and weights. We verified our learning stabilization technique for various types of time-series data such as audio data, language data, and sensor data so as to acquire know-how regarding the relationship between our technique when used in combination with optimizers as well as the degrees of stability and accuracy.

In addition, we are studying a wide range of possible approaches to promote collaboration with a broad spectrum of users, for example, providing technology for speeding up or stabilizing deep learning as open source. We are also studying the possibility of providing to users a one-stop service covering a deep learning

framework, tuning, and a distributed processing platform. This will be done by building the technologies for speeding up or stabilizing deep learning into an AI processing platform called the corevo Computing Infrastructure (CCI), currently being studied at the NTT laboratories [6]. By building deep learning technology into the CCI, we aim to enable even those users without specialized knowledge to perform advanced data analysis.

References

[1] Y. Ida, Y. Fujiwara, and S. Iwamura, "Adaptive Learning Rate via Covariance Matrix Based Preconditioning for Deep Neural Networks," Proc. of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017), pp. 1923–1929, Melbourne, Australia, Aug. 2017.

[2] S. Kanai, Y. Fujiwara, and S. Iwamura, "Preventing Gradient Explosions in Gated Recurrent Units," Proc. of the 31st Annual Conference on Neural Information Processing Systems (NIPS 2017), pp. 435–444, California, USA, Dec. 2017.

[3] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," Neural Computation, Vol. 9, No. 8, pp. 1735–1780, 1997.

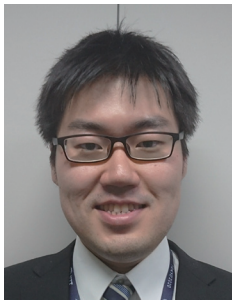
[4] K. Cho, B. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," Proc. of the 19th Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), pp. 1724–1734, Doha, Qatar, Oct. 2014.

[5] R. Pascanu, T. Mikolov, and Y. Bengio, "On the Difficulty of Training Recurrent Neural Networks," Proc. of the 30th International Conference on Machine Learning (ICML 2013), pp. 1310–1318, Atlanta, USA, June 2013.

[6] M. Kawashima, "Four Activities to Create Innovative Core Technologies that Support the Development of IoT/AI Services," Business Communication, Vol. 54, No. 12, pp. 8–13, 2017 (in Japanese).

Trademark notes

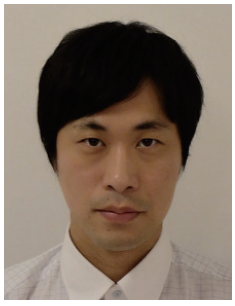
Yasutoshi Ida
Researcher, Distributed Computing Technology Project, NTT Software Innovation Center.
He received a B.E. and M.E. from Waseda University, Tokyo, in 2012 and 2014 and joined NTT in 2014. His research interests lie in the fields of Bayesian modeling and deep learning.



Sekitoshi Kanai
Researcher, Distributed Computing Technology Project, NTT Software Innovation Center.
He received a B.E. and M.E. in applied physics and physico-informatics from Keio University, Kanagawa, in 2013 and 2015. He joined NTT in 2015 and has been studying deep learning algorithms.

Yasuhiro Fujiwara
Distinguished Technical Member, NTT Software Innovation Center.
He received a B.E. and M.E. from Waseda University, Tokyo, in 2001 and 2003 and a Ph.D. from the University of Tokyo in 2012. He joined NTT Cyber Solutions Laboratories in 2003. His research interests include data mining, databases, and artificial intelligence. He has received several awards including the TAF Telecom System Technology Award from the Telecommunications Advancement Foundation, the IPSJ Nagao Special Researcher Award from the Information Processing Society of Japan (IPSJ), and the Commendation for Science and Technology from the Minister of Education, Culture, Sports, Science and Technology. He is a member of IPSJ, the Institute of Electronics, Information and Communication Engineers, and the Database Society of Japan.

Satoshi Yagi
Senior Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.
He received a B.E. and M.E. from Waseda University, Tokyo, in 2000 and 2002. He joined NTT Information Sharing Platform Laboratories in 2002. Since then, he has been studying digital identity, data mining, and optimization problems.



Yasuhiro Iida
Senior Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.
He received a B.S. and M.S. in applied physics engineering from the University of Tokyo in 1998 and 2000. He joined NTT Information Sharing Platform Laboratories in 2000. He is currently researching data mining and data management.