Evaluation of Distributed Processing of the Deep Learning Framework Caffe

Ayae Ichinose
Ochanomizu University

Atsuko Takefusa
National Institute of Informatics

Hidemoto Nakada
National Institute of Advanced Industrial Science and Technology (AIST)

Masato Oguchi
Ochanomizu University

1. INTRODUCTION

The spread of various sensors and Cloud computing technologies have made it easy to acquire life-logs and accumulate data. As a result, many life-log analysis applications, which transfer data from cameras and sensors to a Cloud and analyze them in the Cloud, have been developed. However, it is difficult to transfer raw data from sensors to a cloud because of the limitation of network bandwidth between sensors and a Cloud. In addition, sending raw sensor data to a Cloud may cause privacy issues. Therefore, we propose distributed deep learning processing between sensors and Clouds in order to reduce the amount of data sent to a Cloud and protect the privacy. In this paper, we split a deep learning processing sequence of the Caffe framework[1] and performs distributed processing between a client side and a Cloud side in a pipeline manner. We also investigate method processing times of classification varying a division point and its parameters using data sets, CIFAR-10[2] and ImageNet[3]. From the experiments, we show that the accuracies of deep learning with coarse-grain data are comparable to that with default parameter settings, and the proposed distributed processing has performance advantages in the cases of insufficient network bandwidth as an actual sensors and a Cloud environment.

2. EXPERIMENTS

Deep learning is a machine learning scheme using neural network with a large number of middle layers. The neural network is a information system in imitation of the structure which a human Cerebral cortex has. Even among them, Caffe construct a network architecture called convolutional neural network that is primarily applied for image recognization.

In this study, we propose a pipeline-based distributed framework shown in Figure 1. We define new layers to split the network and implement distributed processing in a client side and a cloud side. This approach makes it possible to protect privacy by sending not raw data but feature value, and reduce transferred data between a sensor and a Cloud for low bandwidth environments.

HPDC '16, May 31 - Jun 4, 2016, Kyoto, Japan

DOI: 10.1145/1235

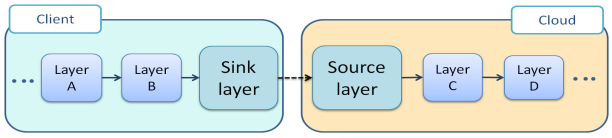

Figure 1. Distributed deep learning framework

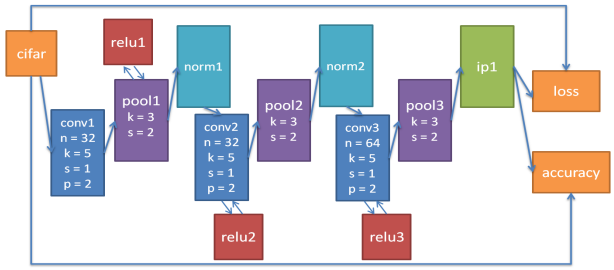

Figure 2. The CIFAR-10 network model

2.1 Accuracies with the different numbers of filters using CIFAR-10

Reduction of transferred data between the layers of neural network may decrease the accuracy of recognization. So we investigate the accuracies when we change the number of filters and reduce the amount of transferred data. We use CIFAR-10 whose network model of the data set is provided by the Caffe web site. The network of CIFAR-10 is shown in Figure 2.

Caffe stores and communicates data in 4-dimensional arrays; the batches size, the number of channels and two-dimensional image size. The channel parameters accords with the number of the filters in the convolution layer just before that. The amount of transferred data in network model for CIFAR-10 is (100 × 3 × 32 × 32) byte at the beginning, and it becomes (100 × 32 × 8 × 8) byte, which is smaller for the first time than the beginning after the pool2 layer. So we split the network between the pool2 layer and the norm2 layer. We investigate the accuracies varying the number of the filters at the pool2 layer in order to reduce the amount of data.

Table 1. Accuracies varying at the pool2

fileters	1	4	8	12	16	20	24	28	32
accuracy(%)	56.6	73.4	76.6	77.2	77.1	77.7	77.6	78.0	78.1

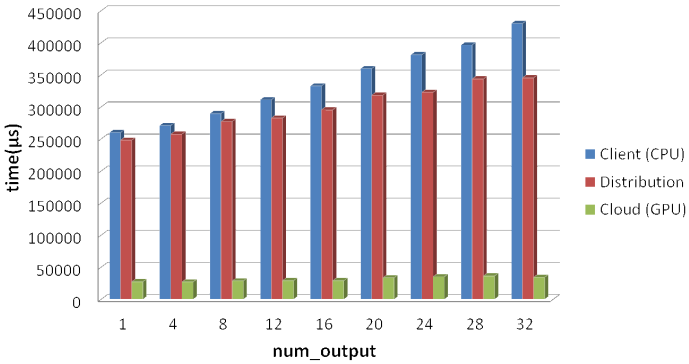

Figure 3. Processing times varying the number of filters at conv2 using CIFAR-10 (1Gbps)

The results are shown in Table 1. We can see that the accuracies converge when the numbers of filters are small as shown in Table 1. Even in the case of the number of filters being8, the accuracy keeps 76.6% and comparable to the result at default, 78.1% while the amount of transferred data in the case of 8 is a one-sixth of the raw data. In other words we can see that it is able to keep high accuracy even if we reduced the amount of transferred data by reducing the number of the filters of convolution layers.

2.2 Processing times using CIFAR-10

We show the effectiveness of the proposed method by measuring processing times of the identification of 1 batch using two machines as a client side and a Cloud side, respectively. We compare three cases; (1) performing all processing in the client side, (2) distributing processing in the both client and Cloud sides using proposed method, (3) performing all processing in the cloud side. No data is transferred between the client and the Cloud are in the case (1), while processed and filtered data are sent to the Cloud in the case (2), and the raw image data are sent to the Cloud in the case (3). In the cases (2) and (3), the client side and the cloud side synchronously work, so we use processing times measured in the client side including connecction times and waiting times. In all experiments, we use only CPU in the client side and use GPU in the cloud side. We set network bandwidth between the two machines to 1Gbps and 10Mbps.

Figure 3 and Figure 4 show the result. Client (CPU), Distribution and Cloud (GPU) mean the results in the cases (1), (2) and (3), respectively. Figure 3 show that the results of the case (3) are superior to the other cases in the 1Gbps high bandwith environment because deep learning processing for images takes more than the time of transmission between the client and the Cloud. However, in the 10Mbps environment considering communication environment between general homes and a Cloud, the results of the case (2) are faster than those of the case (3) because the amount of transferred data is large and the transmission times takes a longer for the 10Mbps environment. In addition, the case (1) is not realistic because of the limitation of the resources in a sensor side, so distributed processing is effective in an actual environment.

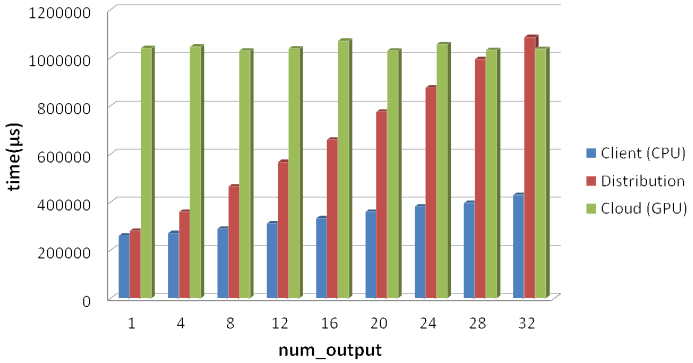

Figure 4. Processing times varying the number of filters at conv2 using CIFAR-10 (10Mbps)

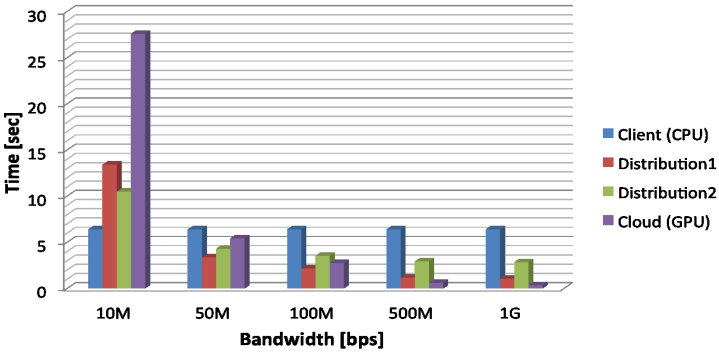

Figure 5. Processing times under different network bandwidth using ImageNet

2.3 Processing times using ImageNet

In the ImageNet network model, the amounts of data becomes a half after the pool1 layer, and a one-third after the pool2 layer. So we split the network between the pool1 layer and the norm1 layer as distribution1 and between pool2 layer and the norm2 layer as distribution2. We measure processing times changing network bandwidth between two machines.

Figure 5 shows that the proposed distributed processing method is again effective when network bandwidths are not large.

3. ACKNOWLEDGMENT

This paper is partially based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

4. REFERENCES

[1] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R. B., Guadarrama, S. and Darrell, T.: Caffe: Convolutional Architecture for Fast Feature Embedding, *CoRR*, Vol. abs/1408.5093 (2014).

[2] Alex, K., Nair, V. and Hinton, G.: The CIFAR-10 dataset. https://www.cs.toronto.edu/~kriz/cifar.html(accessed December 27, 2015).

[3] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. and Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge, *International Journal of Computer Vision (IJCV)*, Vol. 115, No. 3, pp. 211–252 (2015).