http://iset2017.iemek.org

International Symposium on Embedded Technology

ISET 2017

The 12th IEMEK Symposium on Embedded Technology

Intelligent Systems and Applications

Date May 18–19, 2017

Venue Haeundae Grand Hotel, Busan, Korea

■ Hosted by

IeMek Institute of Embedded Engineering of Korea

■ Organized by

ETRI 한국전자통신 www.etr

Cyber Physical Systems Global Center

영남대학교 정보통신연구소

경북IT융합산업기술원 경북차량융합부품사업단

YU BrainKorea21 PLUS ICT기반 스마트기기 창의인재양성 사업팀

■ Sponsored by

YES01 YOUNG-IL EDUCATION SYSTEM (주)영일교육시스템

IeMek Institute of Embedded Engineering of Korea

Deep Learning Training on Distributed Embedded Systems

Sihyeong Park Jemin Lee Hyungshin Kim*

Department of Computer Science Engineering
Chungnam National University, Daejeon, Republic of Korea
E-mail: { sihyeong, leejaymin, hyungshin }@cnu.ac.kr, *Corresponding Author

Abstract

Deep Learning is applied in various fields and shows exceptional performance. In particular, to improve the accuracy of the deep learning, the model should be trained considering the environment and characteristics. The phase of training the model is a complex computation process, it requires multiple GPUs or on a server. In a smart home system, however, sensitive personal information can be included. Therefore, there is a risk of leakage of personal information during transmission to the server. Since there are many built-in devices in the smart home system, the cluster environment can be constructed, so the model can be trained without using the server. In this paper, we build a distributed computing environment using Raspberry Pi to training deep learning.

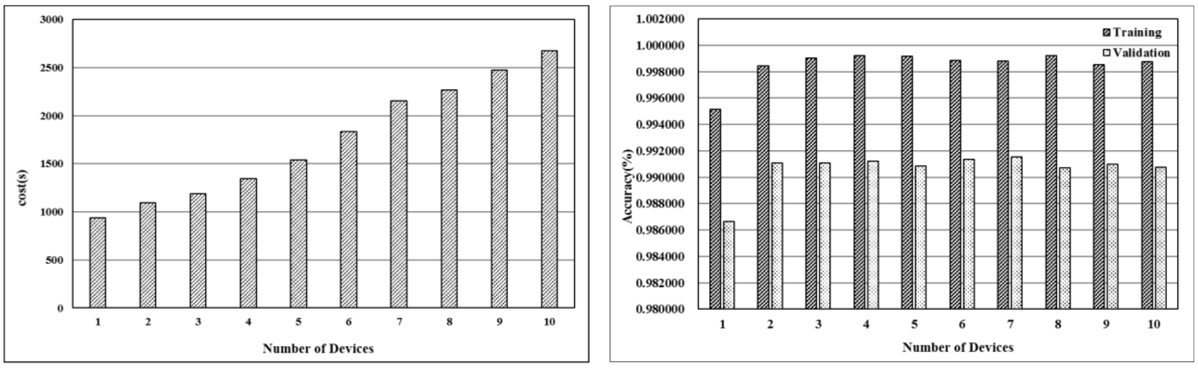
Keywords: Deep Learning, Distributed Computing, Embedded Systems.

1. Introduction

Deep learning, which is a machine learning based on multi-layer artificial neural network, have shown outstanding performance in a variety of domains such as computer vision, natural language processing, and speech recognition. Deep learning technology is usually divided into two steps: training and inference. Training is a process to generate a model with input data. In inference step, trained model is performed for certain tasks such as object detection. Since the training process consists of repeatedly calculating multi-layer, it is typically executed on a high-performance distributed system that supports multiple GPUs. Besides, several deep learning frameworks that support distributed computing have been studied[1-2]. However, all the prior works have running on distributed server environments. Those works have not yet considered full-training on embedded systems without cloud offloading due to limited hardware resources such as energy, memory, and computation power[3].

The availability of Internet of Things (IoT)-enabled devices has resulted in an increasing popularity of smart home systems[4]. Smart home systems make easier to implement a distributed processing system using embedded systems. In particular, smart home appliances using deep learning, high-performance embedded systems such as quad-core are frequently applied for the operation. Also for deep learning, devices collect data from their sensors and then send them to servers for training. However, the training on cloud servers raises privacy concerns because smart home systems incorporate an amount of sensitive information. Therefore, it is necessary to conduct training through distributed processing of embedded systems of the smart home system. Besides, online training is required to adapt environment changes including new furniture structures and joining new members.

In this paper, we use the distributed deep learning framework MXNet on Raspberry Pi 3 and measure the performance of the model training according to the change of the board number. For this, we used 11 boards to measure the time used for training and verify the accuracies. As a result, we show that deep learning is possible even in distributed embedded systems, also deal with problems that may occur. In section 2, we discuss the distributed deep learning framework, the experimental environment, and the result. Section 3 concludes the paper.



(a) (b)

Figure 1. Training results on distributed embedded systems. (a) Time costs, (b) Training and Validation Accuracy

2. Deep Learning Training on Distributed Embedded Systems

This section discusses the results of training through the LeNet[5] with MNIST dataset using the distributed deep learning framework, MXNet[1]. MXNet provides data and model parallel processing from a mobile device to a multi-GPU, multi-Device. Also, this framework uses a distributed key-value storage based on parameter servers for data synchronization for distributed computing. For the experiment, we installed MXNet on the Raspbian OS using 11 Raspberry Pi 3s. The MNIST dataset is a handwritten image of 0-9 and has a size of 32x32. For the evaluation, we set the layer size to 64, the learning rate to 0.05, the epoch to 10 and one Raspberry Pi was used as the master node, and the training and verification accuracy and execution time were measured from 1 to 10 devices. Figure 1 shows the experimental results. In each graph, the x-axis represents the number of devices and y-axis represents execution time and accuracy respectively. (a) is an average of the execution time of each epoch as the device is increased. Distributed processing should reduce the time required for each epoch. However, because of LeNet used in the experiment has a small layer size and batch size also small sample data, the CPU utilization of each device decreases to less than 20%, so the execution time seemed to increase gradually shown in Figure 1 (a). The execution time increased by 2.9 times when using 10 devices compared to 1 device. Figure 1 (b) shows the training accuracy and test accuracy. A Result can see that each use increases more than one.

3. Conclusions and Future Works

In this paper, we trained LeNet model in distributed embedded system and measured execution time and accuracy for a handwritten dataset. We shows that deep learning training is possible even in embedded distributed systems, and the efficiency of distributed processing is low when a small number of data and lightweight models are used. In the future, we plan to use a larger model and use heterogeneous embedded system.

Acknowledgement
This work was supported by the Human Resource Training Program for Regional Innovation and Creativity through the Ministry of Education and National Research Foundation of Korea(NRF-2014H1C1A1066721)

References

[1] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, Zheng Zhang, “MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems”, arXiv, arXiv:1512.01274, 2015.
[2] Hao Zhang, Zhiting Hu, Jinliang Wei, Pengtao Xie, Gunhee Kim, Qirong Ho, Eric Xing, “Poseidon: A System Architecture for Efficient GPU-based Deep Learning on Multiple Machines”, arXiv, arXiv:1512.06216, 2015.
[3] Nicholas D. Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, Lei Jiao, Lorena Qendro, Fahim Kawsar, “DeepX: A Software Accelerator for Low-Power Deep Learning Inference on Mobile Devices”, Information Processing in Sensor Networks (IPSN), 2016.
[4] Kelly, Sean Dieter Tebje, Nagender Kumar Suryadevara,, Subhas Chandra Mukhopadhyay, "Towards the implementation of IoT for environmental condition monitoring in homes.", IEEE Sensors Journal, Volume 13, Issue 10, 2013.
[5] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger, P. Simard, V. Vapnik, “Comparison of learning algorithms for handwritten digit recognition”, International conference on artificial neural networks, Vol.60, pp. 53-60, 1995.