Design on Distributed Deep Learning Platform with Big Data

Mikyoung Lee[1], **Sungho Shin**[1], **and Sa-Kwang Song**[1]
[1]Decision Support Technology Lab, KISTI, Daejeon, KOREA

Abstract - *In this paper, we design a distributed deep learning platform for model to predict typhoon track by analyzing typhoon satellite images. Recently, research on big data distribution processing and deep learning platform is actively being carried out. As the demand for deep learning study increases, it is necessary to study a distributed platform that can support large scale operation of training data when implementing deep learning model using massive data. Our platform uses Docker and Kubernetes to manage the distribution of server resources and Distributed TensorFlow and TensorFlow Serving to support distributed deep learning. We develop the wrapper libraries and modules needed for typhoon track prediction model.*

Keywords: Distributed Deep Learning Platform, Distributed Deep Learning, Distributed TensorFlow, Big Data Analytics

1 Introduction

In recent years, there has been a surge of interests in deep learning. Especially, using deep learning technology in big data analysis process improves the accuracy of analysis result. In order to increase the accuracy of the analysis, it is necessary to use a large number of computers because the size of the deep learning model becomes large and the amount of data to be computed becomes large. When using deep learning with big data, the scale of the deep learning model increases to increase the accuracy of the analysis and the amount of data to perform the calculation increases, so many computers are required. However, research on a distributed deep learning framework is an early stage. In this paper, we introduce a distributed deep learning platform that we designed. The distributed deep learning platform is designed for efficient installation, management and operation of the resources required to develop of the typhoon track prediction model. This makes it possible to training a deep learning model using big data in a distributed processing environment. Also, it is designed considering the stability, convenience, and portability of the deep learning platform.

2 Related Work

2.1 Deep Learning Framework on Spark

This is the use of deep learning distributed processing on SPARK, a well-established cloud. Developed by Seoul National University, DeepSpark allows distributed execution of Caffe deep learning jobs on Apache Spark™ cluster of machines. It is designed to make large-scale parallel distributed deep learning jobs easy and intuitive for developers and data scientists[1]. To support parallel operations, DeepSpark automatically distributes workloads and parameters to Caffe-running nodes using Spark. SparkNet is a framework for training deep networks in Spark. It includes a convenient interface for reading data from Spark RDDs, a Scala interface to the Caffe, and a lightweight multi-dimensional tensor library[3]. Both systems have advantages of using Spark which is excellent for big data processing and using Caffe Deep Learning Library, but there are restrictions on performance optimization when Spark and Caffe are interworked.

2.2 Distributed Deep Learning Framework

Singapore University's SINGA is a distributed deep-learning platform that provides different neural net partitioning schemes for training large models[4]. SINGA architecture supports asynchronous and synchronous training frameworks. Veles[5], developed by Samsung, is distributed machine learning platform. It supports artificial neural network learning and supports genetic algorithm as an integrated development platform for model development, training, and application. Microsoft CNTK[6] improves the performance of high-end parallel training using GPGPU. Petuum is a large-scale distributed machine learning[7]. It considers both data and model parallelism, has a key-value store and dynamic schedulers.

3 Design of Platform

3.1 Overview

We develop a distributed deep learning platform that helps to make deep learning model by training massive data. The data that we mainly deal with are large-scale typhoon observation satellite image data. This job must be performed in a large-scale distributed processing environment because of model parallel processing and data parallel processing. Because it performs a complex operation in a training process of processing a large amount of input data and generating a model. An overview of the platform is shown in Figure 1. The purpose of our platform is (1) to develop a platform capable of training and inferencing large scale deep learning models,

(2) to support efficient distributed deep learning using distributed TensorFlow and TensorFlow Serving, (3) helps to efficiently install, utilize, and operate necessary resources in a distributed computing environment. We want parallelization of data and model, asynchronous update of parameter server, and design as platform to support distributed environment efficiently. We design to make use of the open source libraries to make it easy to build the development environment, to construct a convenient environment to perform computation of big data.

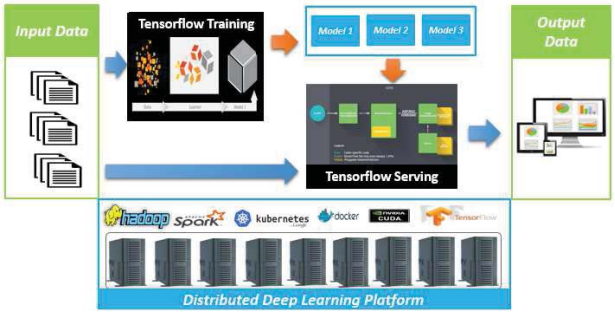

Figure 1. Distributed Deep Learning Platform Overview

3.2 Platform Configuration

The entire cluster node we have is composed of 10, with 128GByte of memory and two GPU accelerators per node. The platform is configured as shown in Figure 2. At the bottom is system OS (CentOS linux). It supports distributed parallel processing using Docker, a software container platform, and Kubernetes, a docker container orchestration tool. These help efficient installation, management, and operation of the distributed processing system. CUDA, is a parallel computing platform, to perform deep learning operations using the GPU. cuDNN is a GPU-accelerated library of primitives for Deep Neural Network. We use this to enhance single machines performance. The CUDA and cuDNN are libraries that make efficient use of the GPU when performing deep learning operations. We use a deep learning library, TensorFlow, to perform distributed deep learning. Distributed TensorFlow supports libraries, APIs, model parallelism, data parallelism, synchronized parallel deep learning, and asynchronous training framework to support distributed environments. It includes TensorFlow Serving to automatically deploy the traning model generated from TensorFlow, and deep learning wrapper libraries (Keras, TFLearn, TF-Slim etc.) that efficiently supports Deep Neural Network. Also, we develop a new KISTI Wrapper library to support the functions needed to operate the typhoon track prediction model and to improve the performance. This platform uses distributed TensorFlow to speed up training and inference of large scale data through data parallel processing, model parallel processing, and also makes distributed processing usefully by using Docker, Kubernetes, and so on.

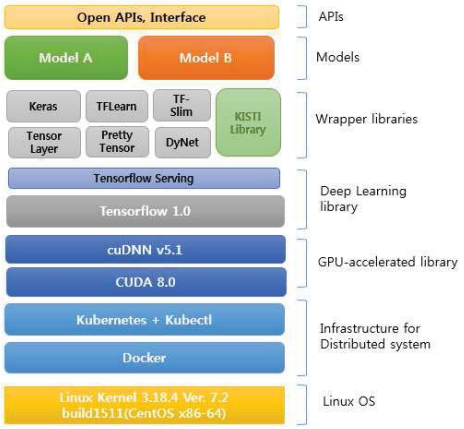

Figure 2. Platform Software Stack

4 Conclusions

Big data analysis using deep learning requires distributed processing technology using many computers. This is because there are many parameters to be learned and data for learning, which requires a considerable computation time for learning. In this paper, we describe about design on big data-based distributed deep learning platform. This platform uses Docker and Kubernetes to efficiently support distributed processing, and supports distributed deep learning using Distributed TensorFlow and TensorFlow Serving libraries. In the future, we will develop a deep learning wrapper library that supports large-scale CNN and RNN algorithms, which are mainly used in the typhoon track prediction model based on satellite image data, and will support training and inference of the typhoon track prediction model.

5 References

[1] H.Kim, J.Park, J.Jang, S.Yoon., "DeepSpark : A Spark-Based Distriuted Deep Learning Framework for Commodity Clusters"., arXiv :1602.08191v3, ACM KDD, 2016.

[2] Y.Jia, E.Shelhamer, J.Donahue, S.Karayev, J.Long, R.Gir shick, S.Guadarrama, T.Darrell, "Caffe:Convolutional Archit ecture for Fast Feature Embedding", MM'14 Proceedings of the 22nd ACM international conference on Multimedia, pp 67 5-678, Nov 2014.

[3]P.Moritz, R.Nishihara, I.Stoica, M.Jordan, "SparkNet: T raining Deep Networks in Spark", arXiv:1511.06051, ICLR, 2016.

[4] W. Wang et al, "SINGA: Putting Deep Learning in the H ands of Multimedia Users," Proc. 23rd ACM International Co nference on Multimedia, 2015, pp. 25-34

[5] Veles, https://github.com/Samsung/veles/

[6] CNTK, https://github.com/Microsoft/CNTK

[7] E. Xing et al, "Petuum: A New Platform for Distributed Machine Learning on Big Data", IEEE Transaction on Big Data, vol. 1, pp49-67, 2015.