Training Deep Learning Model based on Serverless Computing

Young Han Kim*, Ta Phuong Bac,
Soongsil University

younghak@ssu.ac.kr*, bactp@dcn.ssu.ac.kr

Abstract

Serverless computing brings a lot of efficiency and cost for event-driven applications. Besides, the serverless runtimes are limited to deploying applications that requiring lightweight computation and memory, such as training deep learning models. The scale and complexity of the deep learning pipeline make it hard to provision and manage resources a burden for deep learning practitioners that hinders both their productivity and effectiveness. In this study, we propose a method for training the deep learning model by split the deep learning workflows into functions to deploy on a serverless platform. The training process is completed on many distributed nodes then aggregated in the cloud. This solution addresses the general problem of data center resource management as well as provides the best performance and cost to train deep learning models.

Ⅰ. INTRODUCTION

As cloud computing increasingly becomes the platform of choice for commercial and scientific computing, serverless computing [1], has emerged in recent years. With the increased use of containers and the concept of microservices, serverless computing has shown particular promise for the deployment of applications and services. Serverless computing has been applied to the area of deep learning with mixed results in the distributed resource-constrained environment [2,3]. Which, deep learning users are currently faced with several challenges [4] that significantly hinder their productivity and effectiveness:

- users need to manually configure numerous system-level parameters, such as the number of servers, resource allocation (e.g CPUs), servers, system topology, etc.
- users need to specify numerous deep learning parameters (e.g training algorithms, convergence rate, neural network structure), that interact in non-obvious ways with the system-level parameters.
- deep learning workflows are comprised of multiple phases like data processing, training model, hyperparameter tuning, and pre-trained caching, each of which has different computational requirements that deep learning developers have to compute for.

The need to account for the heterogeneous resource requirements of each stage is a burden for the builder that also leads to resource provisioning. The current deep learning frameworks do not have the flexibility required for such workloads. The serverless computing model is a compelling approach to address the data center resource management problem. Therefore, leveraging the added-value of the serverless model in the cloud and edge environments, in this paper we proposing serverless computing structures for training deep learning model by leveraging data parallelism. In addition, we introduce an extendable approach that enables the realization of both deep learning tasks as functions, and of deep learning pipelines as a set of functions.

Ⅱ. THE PROPOSE METHOD FOR TRAINING DEEP LEARNING MODEL WITH SERVERLESS

In this method, the deep learning model is trained distributed to reduce resource problems in the cloud as well as to leverage the resources of distributed environments such as edge devices. A given dataset is partitioned into N multiple subsets, each of which is applied to train a complete model on a machine (called a node). All node shares the same network model and performs compute gradients with a subset of training data by using common training algorithms, such as stochastic gradient [5]. Then, gradients are collected by parameter server, where the network parameters are aggregated and update. The data parallelism by serverless computing is illustrated in Figure 1. Where, all the machines are serverless instances and the parameter server waits till gradients from all workers are received and then executes aggregated and updates the parameters.

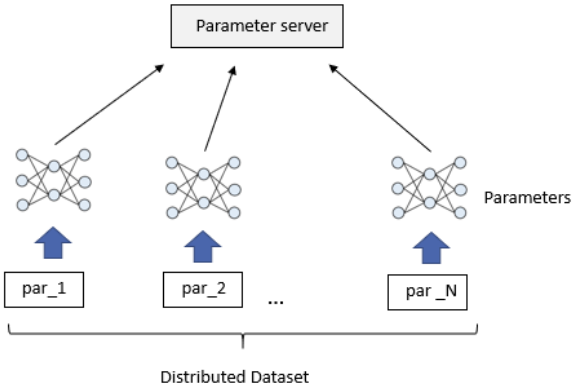

Figure 1. Distributed training

In this training scenario, each deep learning workflow is divided into 3 functions. Which incorporate diverse parts of a training pipeline, ranging from the data processing to trained deep learning model caching as shown in Figure 2.

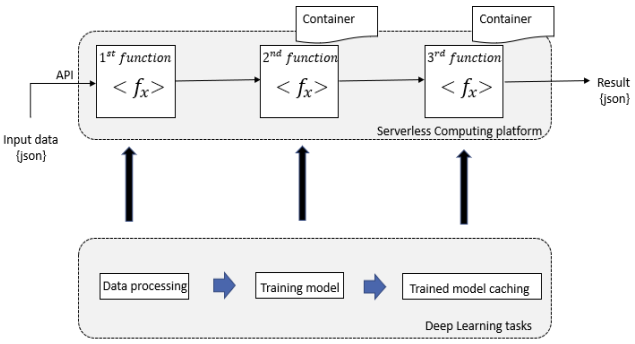

Figure 2. Deep learning workflow -Serverless approarch

The details of these function follow as:

- The *first function* refers to data processing procedures. It receives the parameters (in JSON format) and transforms their structure. Additionally, the function performs some tasks like feature extraction, handle missing values, normalization, and the output is fed into the training function.
- The *second function* the function receives the transformed data from the implemented first function's computations, and its main task is to perform training model by using third-party packages and libraries (python, pandas, keras, etc).
- The *third function* is dedicated to the trained model and caching. The trained model from the second phase is utilized and integrated into the mentioned action (classification, detection, prediction, etc). As the runtime is executed, it produces the final decision. The last action's output produces a message.

To overcome the constraints and limitations of the underlying platform, this method utilized external Docker containers to include the require deep learning libraries needed to complete each function task.

Ⅲ. EXPECTED RESULTS AND FUTURE WORKS

The proposed method for training deep learning model with serverless computing in a distributed environment is expected to leverage the serverless architecture for designing and implementing deep learning tasks to exploit them as a function. To provide the best performance and cost to train deep learning models, while taking advantage of data parallelism were explored.

The proposal will be implemented and evaluated. Then, this scheme will be integrated into a deep learning model deployment strategy in a limited resource environment. This challenge will be addressed in our future works.

ACKNOWLEDGMENT

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2021-2017-0-01633) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation)

REFERENCES

[1] Shafiei, H., Khonsari, A., & Mousavi, P. (2020, January 2). Serverless Computing: A Survey of Opportunities, Challenges and Applications. https://doi.org/10.31224/osf.io/u8xth

[2] C. Cicconetti, M. Conti and A. Passarella, "A Decentralized Framework for Serverless Edge Computing in the Internet of Things," in IEEE Transactions on Network and Service Management, doi: 10.1109/TNSM.2020.3023305.

[3] A. Christidis, R. Davies and S. Moschoyiannis, "Serving Machine Learning Workloads in Resource Constrained Environments: a Serverless Deployment Example," 2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA), 2019, pp. 55-63, doi: 10.1109/SOCA.2019.00016.

[4] V. Ishakian, V. Muthusamy and A. Slominski, "Serving Deep Learning Models in a Serverless Platform," 2018 IEEE International Conference on Cloud Engineering (IC2E), 2018, pp. 257-262, doi: 10.1109/IC2E.2018.00052.

[5] Herbert Robbins, Sutton Monro "A Stochastic Approximation Method," The Annals of Mathematical Statistics, Ann. Math. Statist. 22(3), 400-407, (September, 1951)