Efficient Sample Exchanging for Large-Scale Training Distributed Deep Learning with Local Sampling

Truong Thao Nguyen, Yusuke Tanimura
{nguyen.truong,yusuke.tanimura}@aist.go.jp
National Institute of Advanced Industrial Science and Technology, Tokyo, Japan

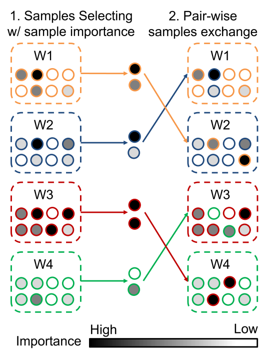

Figure 1: Proposed exchanging strategies.

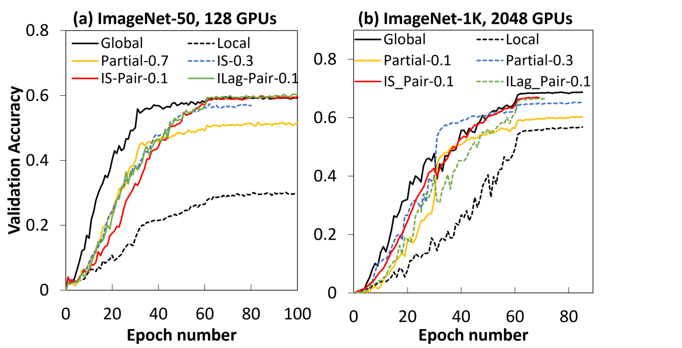

Figure 2: Top-1 accuracy of training ResNet50 model with (a) ImageNet-50 dataset on 128 GPUs and (b) ImageNet-1K dataset on 2048 GPUs.

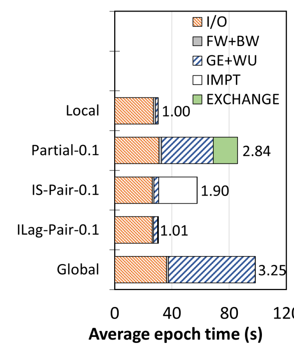

Figure 3: Breakdown of performance (2K GPUs, ImageNet-1K).

1 INTRODUCTION

Stochastic gradient descent (SGD) is the most prevalent algorithm for training Deep Neural Networks (DNN). SGD iterates the input data set in each training epoch processing data samples in a random access fashion (*global shuffling*). Because this puts enormous pressure on the I/O subsystem, the most common approach to distributed SGD in HPC environments is to replicate the entire dataset to node-local SSDs. However, due to rapidly growing data set sizes, this approach has become increasingly infeasible. In this context, an alternative way is to partition the dataset among workers, i.e., each worker uses the same part of the dataset for all the epochs (known as *local shuffling*). Our prior work [1] showed that the local shuffling could not achieve similar validation accuracy as the default global shuffling strategy in large-scale training. In this context, [1] proposed a novel *partial-local shuffling* strategy that randomly exchanges only a proportion of the dataset among workers in each epoch. Through extensive experiments on up to 2,048 GPUs of ABCI, we demonstrated that validation accuracy of global shuffling can be maintained when carefully tuning the partially distributed exchange. However, exchanging the samples randomly between workers leads to a personalized all-to-all communication pattern which is sensitive to network congestion when scaling up. In this study, we propose an exchange strategy that is scalable.

2 PROPOSED SAMPLE EXCHANGE STRATEGY

When training a model at a large scale, e.g., 1000s of GPUs, the number of samples in each local partition of the dataset becomes small. Thus, the distribution of worker's data does not represent the entire dataset. It leads to the reduction of accuracy (as shown in [1]). We propose to exchange the samples that provide more important information in training for balancing importance distribution among devices to further accelerate the accuracy of the training process. We define the importance of a sample by its inference loss. At a epoch t, a sample i is selected to exchange with a possibility of $p_i = \frac{I_i^t}{\sum(I_i^t)}$ where I_i^t is the importance of the sample i. We also avoid network congestion by managing the communication pattern in a pair-wise manner (instead of an all-to-all pattern). That is, we pair the worker that holds the most important sample with the worker that holds the least importance of samples, and so on (W3 and W4 in Figure 1, respectively). To reduce the overhead of computing the importance of samples in an epoch t, we propose to reuse the training losses in the previous epoch $t-1$ (lagging loss).

We evaluate our proposed strategy with the ResNet50 model and ImageNet dataset using the same setting as in [1]. The result in Figure 2 shows that, select samples based on their importance (*IS*) can achieve the same accuracy while exchanging a smaller number of samples (exchange fraction) than those of method in [1] (*Parital*). Pairing method (*IS-Pair*) helps to balance the data distribution among workers more easily leading to a further reduction in the exchange fraction, e.g., only 10% of samples vs. 30% and 70% as in *IS* and *Partial*. With the same exchanging fraction, *IS-Pair* reduces the communication time (WE+WU and EXCHANGE) significantly by avoiding the network congestion (Figure 3. Finally, Using lagging loss (*ILag-Pair*) removes the overhead of computing the importance (IMPT) while maintaining the same accuracy as *IS-Pair*.

3 ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI under Grant Number JP21K17751. This presentation is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

[1] Truong Thao Nguyen, , et al. 2022. Why Globally Re-shuffle? Revisiting Data Shuffling in Large Scale Deep Learning. In *IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 1085–1096.