# Contents



# 1 Highly Scalable Deep Learning Training System with Mixed-Precision- Training ImageNet in Four Minutes

## 1.1 Challenge

***Goals***

Build a high-throughput distributed deep learning training system. 1. improve training throughout * need faster computation and more efficient bandwidth utilization 1. improve the scaling efficiency * need more collective communication primitives * can handle a system with thousands of GPUs

***Challenge***

1. Large mini-batch size leads to lower test accuracy due to the generalization gap problem.
2. When using large clusters, it is hard to achieve near-linear scalability as the number of machine increase.

## 1.2 Optimizations

### 1.2.1 1. Mixed-Precision Training with LARS

***Mixed-precision training with LARS is one of the critical reasons that the proposed system could keep good scalability while increasing the mini-batch size to 64K***.

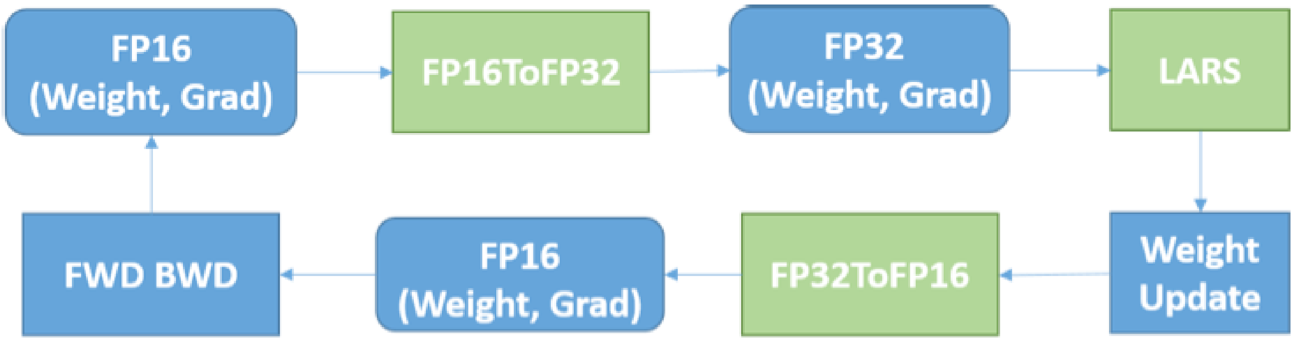
1. half-precision (FP16)
    - lower memory bandwidth pressure
    - increase arithmetic throughput
2. use LARS to enable large mini-batch training

***Problems by a naive implementation***

> ***using LARS directly on half-precision training will cause the computed learning rate to be out of the dynamic range of IEEE half-precision format (FP16), and thus cause the gradients to vanish and stall the training process***.

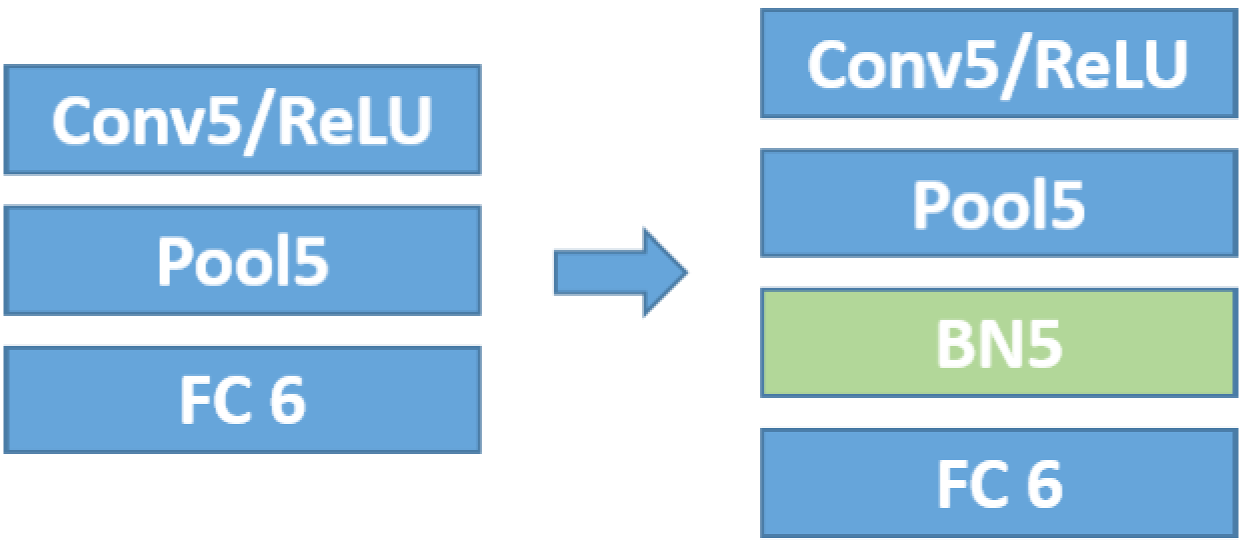
***Solution***

1. the operations in forward and backward propagation are performed in FP16.
2. weights and gradients are cast to single-precision (FP32) format before applying LARS
3. after applying LARS, weights, and gradients are cast back to FP16.




### 1.2.2 2. Model Architecture Improvments

1. ***eliminate weight decay on the bias and batch normalization***.
2. add proper batch normalization layers (after pool5) for AlexNet.




- In neural network training, it is a typical practice to penalize ***only the weights of the affine transformation*** at each layer and leaves the biases unregularized.

### 1.2.3 3. Communication Strategies

objective: ***maximize the throughput as well as reduce the latency***.

- Tensor fusion
  - challenges
    1. when training deep neural networks with multiple layers, the sizes of gradient tensors to aggregate vary a lot for different types of layers.
    2. sending too many small tensors in the network will not only cause the bandwidth to be under-utilized but also increase the latency.
  - solution
    * pack multiple small size tensors together before all-reduce to better utilize the bandwidth of the network.
    * set a threshold in backward phase. only send fused tensor when the total size is larger than the threshold.
- Hybrid all-reduce
  - challenges
    * tensor fusion increases the throughput but also increase latency.

* hierarchical all-reduce instead of ring-base all-reduce perform better for small tensor communication.

### 1.3 Experimental Results

1. For ResNet-50 training, LARS could improve the top-1 accuracy from 60.6% to 71.9%, but cannot reach the baseline accuracy yet. Eliminating weight decay on bias and batch normalization meets the baseline test accuracy.
2. Using mixed-precision training can speedup the single-node performance of ResNet-50 from 172 images/second to 218 images/second. (1.26)
3. scalability