Distributed Optimization of CNNs and RNNs

GTC 2015

William Chan
williamchan.ca
williamchan@cmu.edu

Electrical & Computer Engineering
Carnegie Mellon University

March 19, 2015

Outline

1. Motivation
2. Distributed ASGD
3. CNNs
4. RNNs
5. Conclusion

Motivation

- Why need distributed training?

Motivation

- More data $\rightarrow$ better models
- More data $\rightarrow$ longer training times

Example: Baidu Deep Speech

- Synthetic training data generated from overlapping noise
- Synthetic training data $\rightarrow$ unlimited training data

Electrical & Computer ENGINEERING

Motivation

- Complex models (e.g., CNNs and RNNs) better than simple models (DNNs)
- Complex models $\rightarrow$ longer training times

Example: GoogLeNet

- 22 layers deep CNN

Electrical & Computer ENGINEERING

GoogLeNet

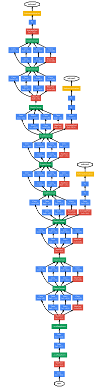
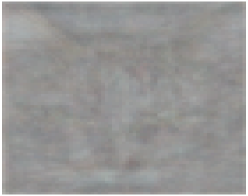


Figure 3: GoogLeNet network with all the bells and whistles

7

- Google Cats, DistBelief, 32 000 CPU cores and more...



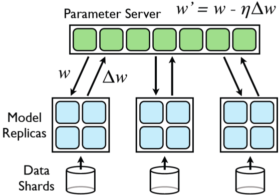


Figure 1: Google showed we can apply ASGD with Deep Learning.

Distributed Asynchronous Stochastic Gradient Descent

- CPUs are expensive
- PhD students are poor : (
- Let us use GPUs!

Electrical & Computer ENGINEERING

Distributed Asynchronous Stochastic Gradient Descent

Stochastic Gradient Descent:

$$\theta = \theta - \eta \nabla \theta \tag{1}$$

Distributed Asynchronous Stochastic Gradient Descent:

$$\theta = \theta - \eta \nabla \theta_i \tag{2}$$

Distributed Asynchronous Stochastic Gradient Descent

CMU SPEECH3:

- x1 GPU Master Parameter Server
- xN GPU ASGD Shards

Distributed Asynchronous Stochastic Gradient Descent

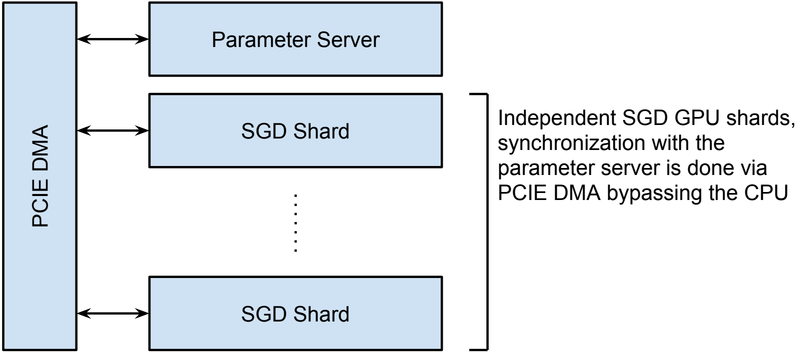


Figure 2: CMU SPEECH3 GPU ASGD.

Distributed Asynchronous Stochastic Gradient Descent

SPEECH3 ASGD Shard $\leftrightarrow$ Parameter Server Sync:

- Compute a minibatch (e.g., 128).
- If Parameter Server is free, sync.
- Else compute another minibatch.

- Easy to implement, < 300 lines of code.
- Works surprisingly well.

Distributed Asynchronous Stochastic Gradient Descent

Minor tricks:

- Momentum / Gradient Projection on Parameter Server
- Gradient Decay on Parameter Server
- Tunable max distance limit between Parameter Server and Shard.

CNNs

Convolutional Neural Networks (CNNs)

- Computer Vision
- Automatic Speech Recognition
- CNNs are typically $\approx 5\%$ relative Word Error Rate (WER) better than DNNs

Electrical & Computer ENGINEERING

CNNs

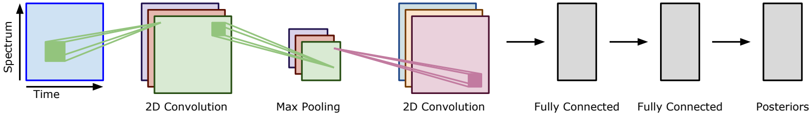
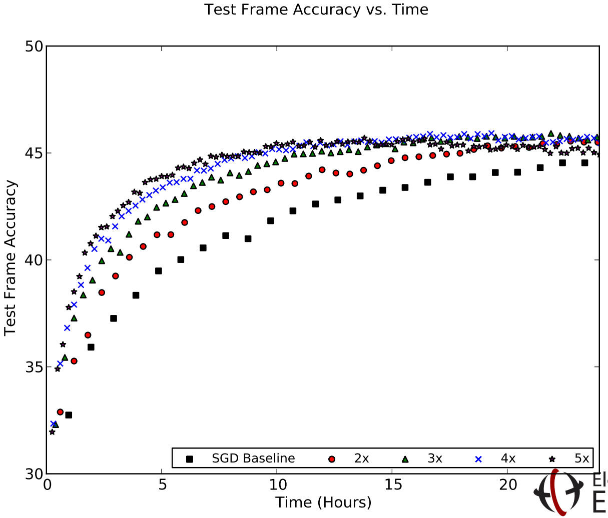


Figure 3: CNN for Acoustic Modelling.

Electrical & Computer ENGINEERING

CNNs



Electrical & Computer ENGINEERING

CNNs

Workers	40% FA	43% FA	44% FA
1	5:50 (100%)	14:36 (100%)	19:29 (100%)
2	3:36 (81.0%)	8:59 (81.3%)	11:58 (81.4%)
3	2:48 (69.4%)	5:59 (81.3%)	7:58 (81.5%)
4	2:05 (70.0%)	4:28 (81.7%)	6:32 (74.6%)
5	1:40 (70.0%)	3:49 (76.5%)	5:43 (68.2%)

Table 1: Time (hh:mm) and scaling efficiency (in brackets) comparison for convergence to 40%, 43% and 44% Frame Accuracy (FA).

Electrical & Computer ENGINEERING

RNNs

Recurrent Neural Networks (RNNs)

- Machine Translation
- Automatic Speech Recognition
- RNNs are typically $\approx$ 5-10% relative WER better than DNNs

Minor Tricks:

- Long Short Term Memory (LSTM)
- Cell activation clipping

Electrical & Computer ENGINEERING

RNNs

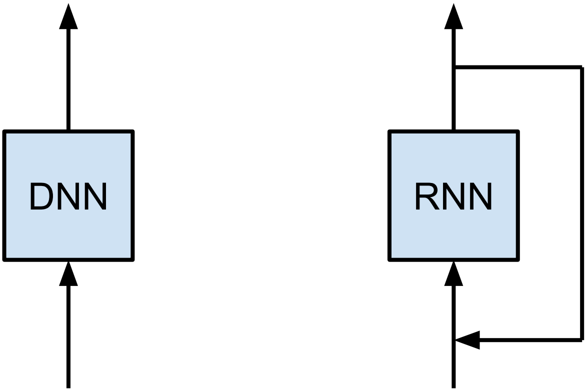


Figure 5: DNN vs. RNN.

RNNs

Workers	46.5% FA	47.5% FA	48.5% FA
1	1:51 (100%)	3:42 (100%)	7:41 (100%)
2	1:00 (92.5%)	2:00 (92.5%)	3:01 (128%)
5	-	-	1:15 (122%)

Table 2: Time (hh:mm) and scaling efficiency (in brackets) comparison for convergence to 46.5%, 47.5% and 48.5% Frame Accuracy (FA).

- RNNs seem to really like distributed training!

Electrical & Computer ENGINEERING

RNNs

Workers	WER	Time
1	3.95	18:37
2	4.11	8:04
5	4.06	5:24

Table 3: WERs.

- No (major) difference in WER!

Electrical & Computer ENGINEERING

Conclusion

- Distributed ASGD on GPU, easy to implement!
- Speed up your training!
- Minor difference in loss against SGD baseline!