Compression: Models are getting larger

Alexnet

2012 → 8 layers, 14 Gflop
~16% Error

2015 → 152, 22.6 Gflop

3.5% Error

The increase in size → increased accuracy

→ need for acceleration

- run a network faster (performance, runtime, inf/sec)
- run a network more efficiently (energy, monetary)

Objectives of acceleration

For inference (previous lecture) → replication (just feed into the network)

Training: → fast forward pass, fast back-prop (gradient)

Today's lecture: How do we reduce cost for all three operations?

1) Using reduced precision arithmetic

→ increase the amount of data we can fit into a given processing unit

and hence optimize util.
2) Compression -> pruning. (reduce the ops we are performing)

3) Better algorithm. (fastest alg possible) Low-rank approx. of fundamental ops in DNNs.

---

Reduced Precision

- We get reduced storage
- We get reduced energy (because we need less complex hardware "fewer transistors") when we use reduced precision.

→ Improved performance (speed)

→ Sometimes: has little effect on the accuracy.

Examples: \( \max(0, x) = x \), \( x < 0 \)

---

precision is less important
DNN: \( f \) relu, or other non-lin.

\[
b_i = f \left( \sum_j w_{ij} a_i \right)
\]

Output \( b \)

Param, \( a \)

Input vector

\( M \) weight matrix

\( \rightarrow \) multiplication

\( + \) addition
\[ b = f(\mathbf{W}a) \] forward.

\[ W_{ij} = W_{ij} + \Delta a_{ij} \text{ updates} \]

learning rate \rightarrow \text{ Full added} \\
\text{multiplication} \rightarrow \text{Addition}

Add, Mult units

\textbf{FP32} (full precision)
\textbf{FP16}
\textbf{SEU}

\begin{array}{c|c|c}
\hline
\text{Range} & -38 & 52 \\
\hline
10^{-8} & 6 \times 10^{-5} & 0.5 \\
10^{-4} & 6 \times 10^{-4} & 0.05 \\
0.125 & 6 \times 10^{-1} & 0.5 \\
\end{array}
Goal of low-prec arithmetic

Optimize the tradeoff between prediction accuracy of the network and

\[ \text{Cost} = f(\text{energy, runtime, ...}) \]

Add in FP32 $\rightarrow$ 0.9 pJ 4k\( \mu \text{m}^2 \)
FP16 $\rightarrow$ 0.4 pJ 1k\( \mu \text{m}^2 \)

- Mult, Add $\rightarrow$ operations

Parameters:
- weights, inputs/outputs
- either use the same precision
- mixed precision

Either use raw data to the intermediate results

\( \text{gradient, learning rate} \)
Mixed precision says:
use different arithmetic for 
non inputs/outputs/grads

weight et.

\[ x_1 = x_0 + ax \]

scaling.

(32b)

24b batch non.

Forward pass

46b \[ W \]

16b \[ a \]
Weight Update

$\Delta \rightarrow 16^{-5}$

low prec arithmetic

$\Delta w$?

$+\rightarrow W$

What is the effect of low precision here (assuming quantization/rounding)?

$10^{-5}$

very low learning rate.
To avoid $\Delta W = 0$ due to low-prec.

\textbf{Stochastic Rounding (SR)}

\[ \begin{align*}
\text{Input} & \rightarrow \text{Full prec} \\
\Delta W & \rightarrow \text{SR} \\
\Delta W' & \rightarrow \text{Output}
\end{align*} \]

I want to add 0.3 to 0, 100 times.

$E(\Delta W') = \Delta W$

Reg. rounding:

0.3 $\rightarrow 0$

0 $\neq 30$

\[ \text{Round}(x) = \begin{cases} 
Lx, \text{ w. prob. } 1 - (x - Lx) & \\
Lx + 1, \text{ w. } x - Lx
\end{cases} \]

\[ \text{Round}(0.3) = \begin{cases} 
0, & 70\% \\
1, & 30\%
\end{cases} \]

$E[\text{Sum}] = 30$
Choi, De Sa ISCA '17 discusses efficient stochastic rounding schemes.

Summary

- Reduced prec. -> save memory space
- Reduced central band

Important points

High prec. -> Batch norm

Stochastic Rounding during...

2. Pruning
after training.

W. Full
Train Connectivity

Prune Connection

Retrain

lot.
After pruning (whatever are left)

Quantize

Full connect:

Prune

Retrain

Pruning

Quantize

Original network

same case?

reduction size of myrrh
Want to group weight w. similar values into one weight.

2.09, 2.12, 1.92

Let's assign these params into a cluster k = 2.00

\[ v \]

\[ K \]
4x FP16 vs. 4x int8 + FP16

This requires changing how one performs updates and learning.

\[ W_{\text{Matrix}} = \begin{bmatrix} W_{\text{Matrix}} + 2 \end{bmatrix} \]

Cluster index centroids.

Now I only update the centroids and the indexes. They don't care about.
Some W's are changing frequently.

What does it mean that W is changing frequently? (we have not conveyed)

freq. weights → less bite
in freq. weights → increased bite

(Huffman Coding)
original model → **Pruning** → **Qualize** → Hullman → Cadini

- 10x
- 30x
- 50x

SqueezeNet (great paper)