HW evaluation for DNN execution

David Levinthal

Microsoft: Azure CSI

All kinds of HW

- CPU, GPU, TPU, FPGA, assorted DNN processor startups
  - Many versions of each
    - AVX512/MKL for Intel CPUs
    - K80s, M4/40, P4/40/100, volta from Nvidia
    - KNL, KML from intel
    - Vega10 from AMD
    - Altera and Xylinx FPGA systems
    - Google TPUv1/v2
    - Groq, Nervana, Graphcore, Wave Computing, Cerebras, Tenstorrent, etc

DNNs comes in many shapes and sizes

- CNN: Convolutional Neural Networks
  - Image recognition
- RNN: Recursive Neural Networks
  - Text recognition
  - Language translation
  - Voice recognition
- Reinforcement Learning
  - Games
- New ones all the time, very fast moving field

Many Programing Environments

- Dominated by Tensorflow
  - Caffe, cntk, mxnet,torch7, caffe2
  - Python, Lua and assorted framework specific sytaxes
- But all are relevant
- HW must have support for all

ml framework	github stars
tf	64154
cntk	11792
mxnet	10446
caffe	19146
caffe2	5213
torch7	7112
all but TF	53709

Benchmark issues

- Nature of benchmarks depends on the DNN structure
  - Mostly CNNs
    - Can be run with synthetic data making for short running tests
      - https://www.tensorflow.org/performance/benchmarks
  - But RNNs are actually executed at larger scale
    - Search, Edge interactions, Large data investigation
    - Real problems require real data
      - Inference then requires a converged Trained model
    - Does Fairseq change this situation?
- Strong HW specific performance dependence on:
  - Batch size (images per batch, word seq length)
  - Numerical precision (FP32, FP16/fp32 accumulate, int8 for inference)
    - Performance impact of numerical precision can depend on batch size

Dearth of public benchmarks

- Useful components
  - TF_CNN_Benchmarks: https://github.com/tensorflow/benchmarks/tree/master/scripts/tf_cnn_benchmarks
    - TF Only, Synthetic data, Training and Inference, wide set of models
  - DLBench: HKBU https://github.com/hclhkbu/dlbench most frameworks
    - Real Data, but only training, small problem sizes
  - PTB: https://github.com/tensorflow/models/tree/master/tutorials (only PTB works, TF only)
  - Seq2seq/Translate: https://github.com/google/seq2seq (requires patching, TF only)
  - DeepBench: https://github.com/baidu-research/DeepBench (Bash, C, Cuda)
  - Convnet: https://github.com/soumith/convnet-benchmarks (many frameworks)
  - https://github.com/facebookresearch/fairseq (Torch seq2seq with conv and lstm)

Issues

- Different distributions of a network are not equivalent
  - Alexnet(TF,dlbench) != Alexnet(convnet) != Alexnet(tf_cnn_benchmarks)
- Comparing performance across frameworks is difficult
  - When evaluating HW, frameworks must be evaluated independently
  - For cloud hosting ALL frameworks must work well
- Benchmarking across range of batch sizes and numerical precision
  - Inference has a need for batch size = 1->4
  - Varying numerical precision changes speed AND result accuracy
- TF Performance on RNNs?
  - http://dlbench.comp.hkbu.edu.hk/
- OMP support in frameworks is problematic
  - Required for effective MKL usage
- Lack of common intermediate representation
  - Ex: XLA
  - Raises difficulty on new HW supporting many frameworks
  - Increases difficulty of porting benchmarks
- Who is r2rt aka Spitis? ☺

Resources (mostly RNNs, they are more difficult)

- TF tutorials, models and benchmarks (https://www.tensorflow.org)
- CNTK Tutorials/examples https://www.microsoft.com/en-us/cognitive-toolkit/features/model-gallery/?filter=Tutorial
- http://www.wildml.com/2015/09/implementing-a-neural-network-from-scratch/
- http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/
- http://www.wildml.com/2016/08/rnns-in-tensorflow-a-practical-guide-and-undocumented-features/?utm_campaign=Revue%20newsletter&utm_medium=Newsletter&utm_source=The%20Wild%20Week%20in%20AI
- http://karpathy.github.io/2015/05/21/rnn-effectiveness/
- http://karpathy.github.io/neuralnets/
- https://github.com/spro/practical-pytorch/blob/master/seq2seq-translation/seq2seq-translation.ipynb
- http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://github.com/spitis
- https://r2rt.com/
- Deep Learning by Ian Goodfellow and Yoshua Bengio