Fermilab

Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

Machine/deep learning in HEP

Gabriel N. Perdue
SAC Computation Working Group
2017 / April / 25

Learning with machines

- Machine learning has a long and vibrant history in HEP. For certain types of problems, we have well-understood workflows and a lot of institutional knowledge.
- The modern field of "deep learning" is a bit different.
  - Training data sets are generally much larger than what we typically handle when, for example, training a BDT on a set of high-level variables in an ntuple. Deep learning algorithms prefer data in a form much closer to raw data (e.g., image data) and these data are often orders of magnitude larger than analysis inputs.
  - Our data formats (e.g., ROOT ntuples) are generally not compatible with deep learning frameworks (prefer, e.g., HDF5).
  - Algorithmically, deep learning is evolving rapidly, and the most important researchers are driven by very different motivations than ours.

A different type of data locality problems

- Training and evaluation are often done in very different environments.
- Access to GPU clusters and/or HPC facilities can be a challenge (and relatively few "mid-tier" facilities where we can learn how to scale applications).
- Can we piggy-back on data locality solutions being employed for other problems (e.g. at HPC facilities)?
- How do we make evaluation as efficient as possible? One approach is to train on a GPU cluster and evaluate on a grid node (CPU), but even this can be slow, and can benefit from very hardware-specific optimizations.
- Should we be trying to rent or use cloud-based "ML as a service" APIs/providers?

File formats and algorithms

- One of the first, universal problems faced in using a deep learning framework is translating the data from (usually) ROOT to another format (HDF5, LMDB, etc.).
- This is not practical to do on the fly, so we have to build tools to make the translation at a large scale in advance.
  - Currently, everyone is building their own tools and workflow for this.
- What is the optimal way to approach this problem?
- Algorithmically, deep learning is evolving quickly and driven by industry. Our needs are different (we often care more about the uncertainty associated with algorithm performance than absolute performance).
- How do we contribute to development? What makes us special?
  - High fidelity MC and understanding biases
  - Wide variety of experiments and use cases (cross pollination)
  - Applications of AI/ML to scientific questions

Back-up

What is deep learning?

- "Deep learning" is the (now buzz-wordy) phrase used to describe machine learning using neural networks with many hidden layers (many layers being "deep").
- Revolutionary success in image recognition (cats on the internet) and natural language processing tasks (machine translation, voice processing), plus reinforcement learning (e.g., playing Go or Starcraft).
  - Success based on new hardware (GPUs) and computer science advances ("tricks").
- It has become a key focus for many companies in industry (Google, Amazon (Echo), Apple, Uber (self-driving cars), etc.) and is an extremely active and rapidly changing field of academic research.
- In HEP (neutrinos) we are mostly operating in the context of image recognition problems (but have some plans for treating particles as a "grammar", etc. as they have begun to do in some collider contexts).

Why deep learning?

- ImageNet is an image database (14.2 million images) organized into a classified hierarchical tree.
  - http://image-net.org
- They run an annual competition for classification. How have people done over the years in the classification challenge?
  - For many years, ~70%
  - 2010: 71.8%
  - 2011: 74.3%
  - 2012: 84%
  - 2013: 88.2%
  - 2014: 93.3%
  - 2015: 96.4%
- Humans: about 95%*
- What happened in 2012?

* e.g., Which is the Siberian Husky? Which is the Eskimo Dog?

Deep learning

- Prior to 2012, progress was very slow. ImageNet doesn't publish data from previous competitions, but comments in the literature (e.g., LeCun, Bengio, Hinton, nature14539 and refs within) indicate that beating 70% was very challenging and improvements were small.
- Neural networks are an old idea (neocognitron 1980s, LeCun's "LeNet" 1990s, etc.) that were given up in the 2000s because they were simply too slow to train.
- But cheap, prevalent GPUs have really changed the game and deep networks are finding application everywhere (driverless cars, Siri, Amazon Echo, winning at Go, etc.)
- Recent algorithmic breakthroughs have helped deal with over-fitting.
- Deep nets seem to remove the need for heavy feature engineering (true "machine learning").