Machine Learning inference as a Service optimization in neutrino reconstruction

Neutrino experiments seek to answer fundamental questions about the nature of the universe, looking into the origin of matter, black hole formation, the unification of forces, and much else besides. The LHC also tackles these fundamental questions. In both cases, we have to perform some reconstruction on the raw sensor data. Traditionally, reconstruction has been done on the CPU. Recently we have begun to move to machine learning algorithms to do much of this reconstruction because it potentially provides a more accurate output and runs faster than traditional algorithms.

Despite these newer sophisticated algorithms running faster, they are still computationally expensive and require the coprocessor chip -whether it be GPU or FPGA- to be a direct part of the reconstruction workflow. The data being processed through these reconstruction chains are also rapidly increasing. One way to combat these issues is to provide the coprocessor as a web-based service instead of directly as exposed hardware. This allows for significant increases in speed without disrupting the native reconstruction workflow within LArSoft.

While GPUs are getting faster, because of the increasing data volume we still need to develop a better understanding of the resource requirements and reduce these requirements for machine learning applications. These resource requirements can only be optimized in the context of the exact hardware being used whether that be GPUs or FPGAs so we can leverage their unique architecture. This means that for this sort of optimization to be feasible, machine learning inference has to be available as a service. Currently, such a service is available in the Larsoft workflow. This project aims to work on machine learning reconstruction of neutrinos by developing a model based on the new Monte Carlo information that has become available. Furthermore, it will then optimize this model, on the inference side so that the queries can be answered quicker/with less compute resources using Nvidia's TensorRT suite. The TensorRT suite relies on breaking the model into the appropriate subgraphs and then seeing which ones can be optimized to run on Tensor RT engines and which ones should be left to the default TensorFlow engines. Furthermore, it allows us to combine nodes and subgraphs so that they can run more efficiently. Tensor RT optimization applies to any machine learning model, not just the ones used for neutrino reconstruction. This project will be carried out under the supervision of Jane Nachtman (Professor, Dept Physics, University of Iowa).

Background:

I am a second-year graduate student at the University of Iowa. I have graduated with a double major in Physics and Computer Science from Cornell College. I have prior experience with machine learning as well as LArSoft. My machine learning experience is both from classes I've taken as an undergraduate as well as models I have created in my own time. I have also attended multiple workshops on Machine learning that were conducted at fermilab. I am a member of the Fast ML team working on faster inference in HEP. I also have experience with neutrino reconstruction and have worked on the wire cell method of reconstruction.

Deliverables:

- An initial machine learning model - This would be a frozen graph that is specific to the neutrino reconstruction that I intend to work on
- Tensor RT optimization on the model - This would be a Tensor RT optimized graph which would also be specific to the neutrino reconstruction
- Documentation of instructions to apply Tensor RT optimization to any machine learning model - A generic set of instructions that anyone could follow to use Tensor RT to optimize their graph
- Final Report and Findings - A general report on what was done, what the lessons were and what future steps may be

Timeline:

Dates (m/dd)	**Planned Activity**
2/15-2/28	Gain understanding of current machine learning model and its implementation
3/01 - 4/24	Create and Train new preliminary model for the reconstruction
4/25-5/15	Gain better understanding of Tensor RT and how the optimization process works
5/16-7/23	Optimize the model inference with TensorRT
7/24-8/14	Create final report and documentation for optimization

Reference:

arXiv:2009.04509 **[physics.comp-ph]**

NVidia TensorRT Documentation. (n.d.). Retrieved November 30, 2020, from https://docs.nvidia.com/deeplearning/tensorrt/developer-guide/index.html