CMS Jet Reconstruction with Quantile Neural Networks

Braden Kronheim

Mentor: Michelle Kuchera (Davidson College)

Duration: May 31- August 20

Background:

A key component of data analysis at the CMS experiment is correctly reconstructing the energies and directions of jets. Typically, this is done by reconstructing individual particles in the detectors using Particle Flow particles, clustering the hadrons together using the anti-kT algorithm, removing pileup through charged hadron subtraction, and correcting the final jet 4-vector based on what bin its 4-vector falls into. This method works well but has two areas which may be improved using machine learning. First, the algorithm has many components and steps. This is not inherently a problem, but it would be beneficial to have an algorithm that accomplishes the task in a single step. Second, this method does not consider the probabilistic nature of hadronization in detectors. The exact same incident jet will interact differently with detector every time it enters, and the same detector response could be caused by different jets, so a probabilistic reconstruction method is desirable.

Over the past two semesters of work on an honors thesis I developed an algorithm which has the potential to solve both of these problems using neural networks. It functions by accepting an input array of particle four vectors ordered by transverse momentum (pT). Through the use of deep networks which assign particles to jets, the algorithm is able to build a set of potential jets from the Particle Flow particles. A third network provides a final correction factor, accounting for parts of the jets which were incorrectly reconstructed by the detector. Additionally, by using a quantile loss function, these networks were trained to predict an output corresponding to a given quantile.

The use of this quantile loss function, initially discussed in the context of image fill-in in the referenced paper at the end of this proposal, has proven very effective so far. When tested on recovering the generator level jet (gen) from just the uncorrected reconstruction level jet (reco) in one step, the trained network was able to match the conditional distribution of gen given reco very accurately, even for very limited reco pT ranges. The full results for the previously described particle by particle approach with the quantile loss function are not yet available, though it has succeeded in predicting a range of possible jets when given a set of inputs.

Proposal:

I have a proof-of-principle version of this algorithm currently working, though there are still several areas which need to be investigated to optimize the algorithm and make it useful to physicists using CMSSW. These areas would be addressed through this fellowship. First, the algorithm needs to be implemented and documented for use within CMSSW. I have already implemented an older version which takes a particle-by-particle approach but is not probabilistic. The probabilistic work has been done entirely in python, and models have not yet been adapted for use in CMSSW. This will be the first deliverable.

The second research area is in creating an optimally trained network for accuracy. This will be examined through a hyper parameter search, an examination of the training data used, and an examination of the network architecture. These networks are trained using TensorFlow, which allows significant room for experimentation with hyperparameters and architectures. After

this optimization has been performed, I will then need to decrease the size of the network as much as possible and optimize the prediction code as much as possible to decrease the run time of the algorithm. Once these steps have been accomplished, a second version of the code will be released with the updates as the second deliverable.

Once this has been accomplished the network will be exhaustively tested on datasets containing jets created in various types of reactions and with different flavor contents. This will determine the efficacy of the algorithm and point out week points of its design, which will then be dealt with. After completing these adjustments, a final version of the algorithm will be released, along with documentation on how to use it and train the networks. Additionally, I plan to submit a paper on this work at that time.

Deliverables:

- June 11th: Implementation and documentation of model within CMSSW containing models already trained.
- July 9th: Implementation of a new version of the model with networks fine tunned for the highest possible accuracy and speed
- August 20th: Final implementation and documentation of model with adjustments for different jet sources and flavors, documentation of training process, and submission of a paper for publication on the model

Timeline:

Dates	**Task**
May 31 – June 11 (0.5 FTE Months)	Complete implementation of current model in CMSSW, determine typical time required to run the networks.
June 14 – June 25 (0.5 FTE Months)	Work on improving the general performance of the network through varying the form of the four-momentum vector predicted and the network architecture.
June 28 – July 9 (0.5 FTE Months)	Investigate ways to increase execution speed by decreasing the required number parameters in the network and improving the algorithm implementation.
July 12 – July 23 (0.5 FTE Months)	Procure additional jet datasets and determine efficacy of network on jets with different sources and flavors.
July 30 – August 6 (0.5 FTE Months)	Address limitations of approach exposed during previous two weeks.
August 9 – August 20 (0.5 FTE Months)	Complete documentation of how to use system with CMSSW, retrain networks, and procure datasets used. Complete paper detailing research done over the past year on the project.

Reference:

arXiv:1806.05575 **[cs.LG]**