

Machine Learning Frameworks for DUNE

Leigh Whitehead

Mini-Workshop on Portable Inference

04/12/20

Overview

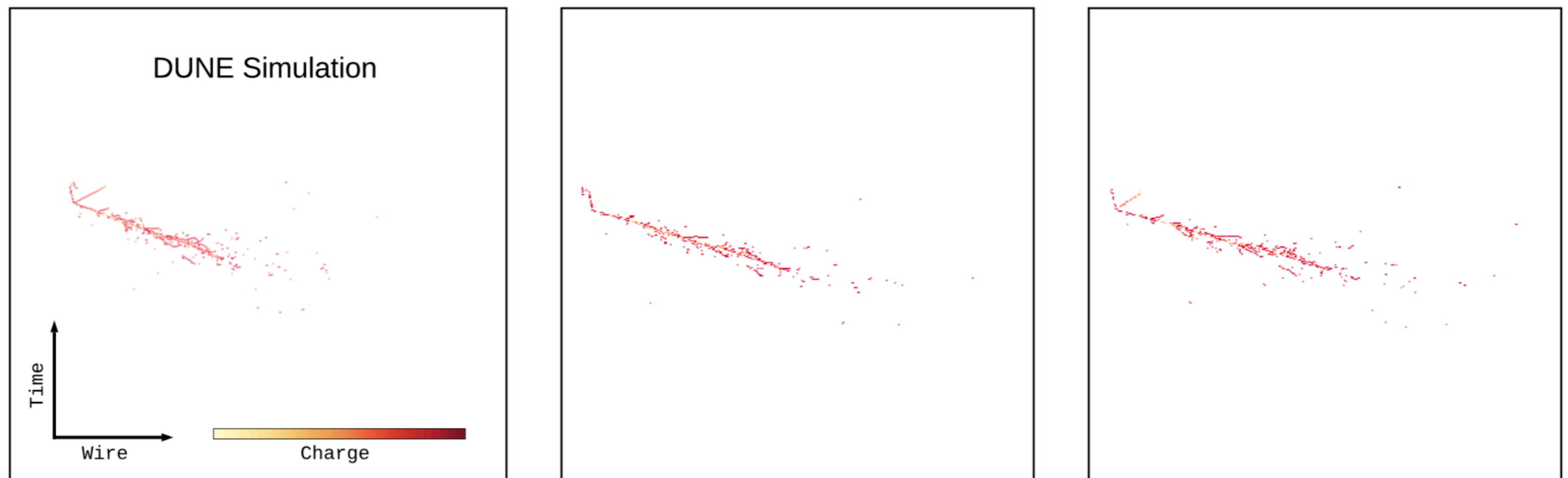
- I will briefly discuss the main machine learning techniques used in the DUNE experiment
 - DUNE Convolutional Visual Network (CVN)
 - ProtoDUNE hit-tagging CNN
 - Neutrino energy estimation
 - Track particle ID
- At the time of writing, all methods imbedded in the DUNE software run using interfaces to tensorflow

Typical workflow

- Write a C++ LArSoft module to dump the required information
 - For example, hit charges for all readout wires and time
- Generate a dataset in python using the above information
- Develop and optimise the deep learning algorithm in python
- Save the trained model and convert to .pb format
- Load the model into the DUNE software C++ tensorflow interface
- Run inference using a C++ LArSoft module

DUNE CVN

- The DUNE CVN^[1] is used for neutrino event classification
 - Operates on 3 x 500 x 500 pixel images of the neutrino interaction
- It runs after the reconstruction at the “analysis level”
 - Takes the reconstructed hit objects as input



[1] DUNE Collaboration, Neutrino interaction classification with a convolutional neural network in the DUNE far detector, Phys. Rev. D **102** 092003, 2020

DUNE CVN

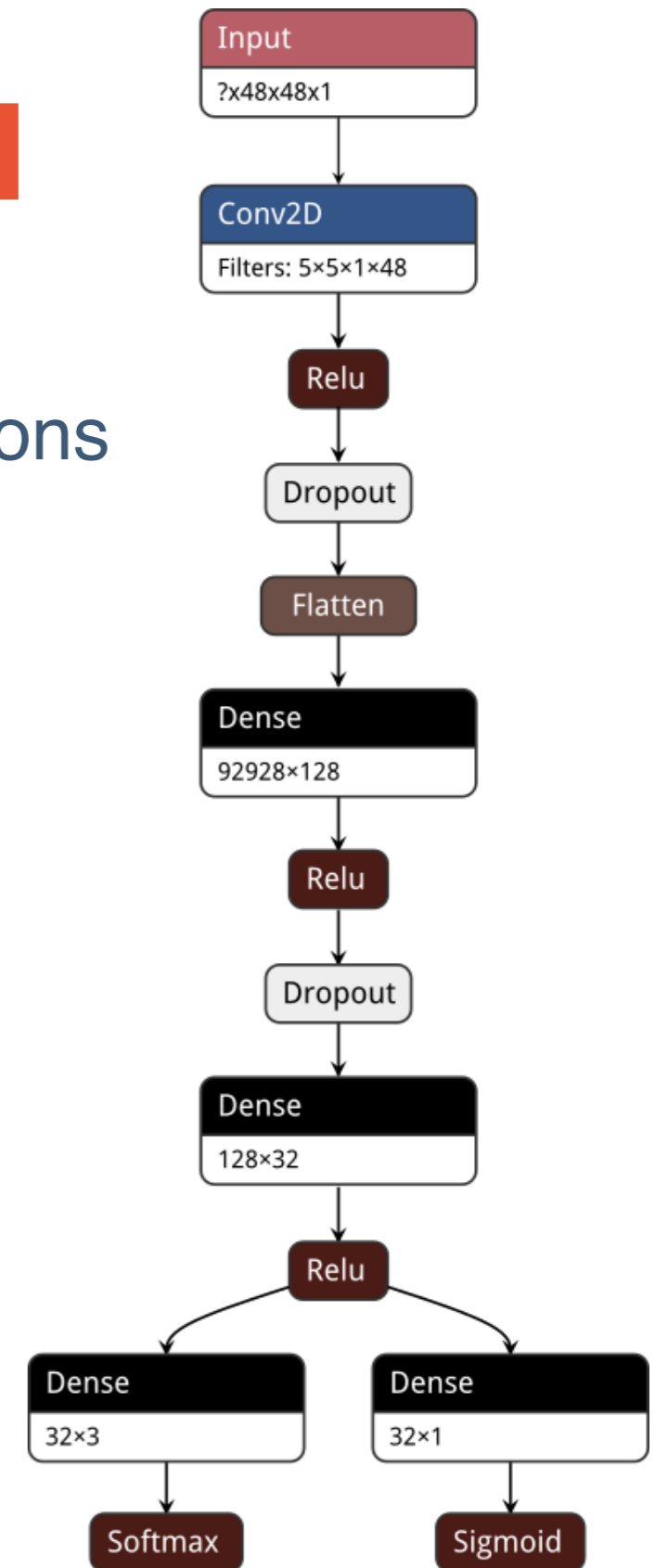
- The DUNE CVN^[1] is used for neutrino event classification
- Architecture based on the SE-ResNet
 - Early layers split over three branches, one for each input image
 - Total of 50 convolutional layers
- The inference time in LArSoft is less than a second
 - It only runs once per event
 - This is a small fraction of the total processing time
- Preliminary work has been done on running inference on an Edge TPU and FPGA^[2].

[1] DUNE Collaboration, Neutrino interaction classification with a convolutional neural network in the DUNE far detector, Phys. Rev. D **102** 092003, 2020

[2] M. J. Rodriguez, Fast inference using FPGAs for DUNE data reconstruction, EPJ Web of Conferences 245, 01030 (2020)

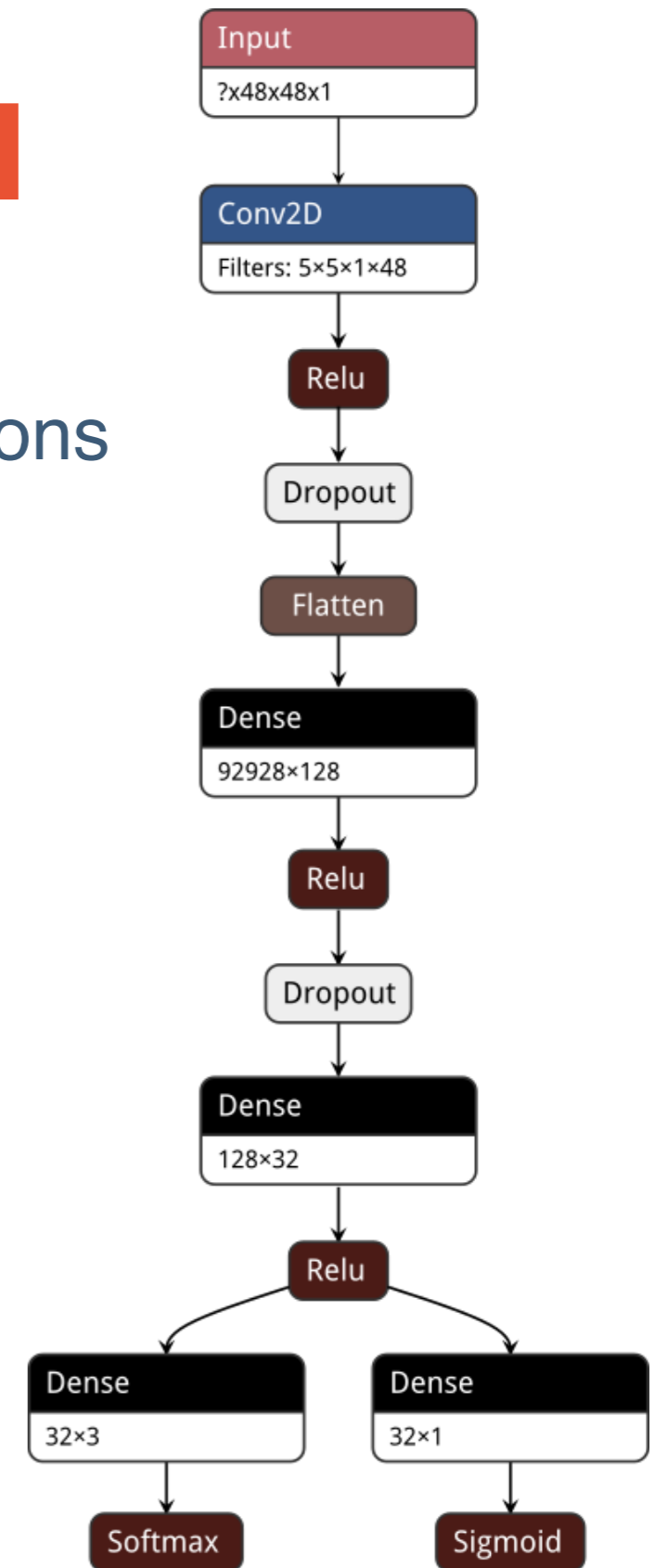
ProtoDUNE hit-tagging CNN

- Classifies hits at either track-like or shower-like
 - It also has an additional output for Michel electrons
- The CNN operates on 48 x 48 pixel images
- It needs to run many times per event
 - Runs separately on each detector readout view in each of the six TPCs
- The architecture is simple and uses only a single convolutional layer



ProtoDUNE hit-tagging CNN

- Classifies hits at either track-like or shower-like
 - It also has an additional output for Michel electrons
- One of the most time consuming parts of the ProtoDUNE-SP reconstruction chain
- Promising results obtained using the GPU as a service approach^[3]
 - Inference step sped up by a factor of 17
 - ProtoDUNE-SP reconstruction run-time improved by factor 2.7



[3] M. Wang, et al., GPU-accelerated machine learning inference as a service for computing in neutrino experiments, <https://arxiv.org/abs/2009.04509>

Other algorithms

- There is another CNN that measures the energy of neutrino interactions in DUNE
 - It uses a very similar framework to the CVN
 - Returns the energy under electron neutrino and muon neutrino hypotheses
- There is a 1D CNN that performs particle identification of track-like particles
 - Runs on each track in a neutrino interaction
 - Very lightweight due to a reasonably shallow (and 1D) architecture
 - Also runs with a very similar framework to the CVN

PyTorch

- There is a lot of interest in better coping with sparse data
 - Sparse Convolutional Neural Networks
 - Graph Neural Networks
- These are typically more straightforward (at the time of writing) to implement in PyTorch
 - PyTorch library now available in LArSoft
 - No interface developed within the DUNE software (yet)
- It has been incorporated into the Pandora reconstruction software

Summary

- There are (to my knowledge) four DUNE deep learning algorithms currently running inside the LArSoft framework
- Three of the algorithms only run once (or a few times) per event
 - These are not a burden on data processing
- The ProtoDUNE-SP hit-tagging CNN runs many times for each event
 - Significant speed-up obtained through GPU-as-a-service approach
- I'm sure that lots more deep learning algorithms will start appearing in the coming months and years