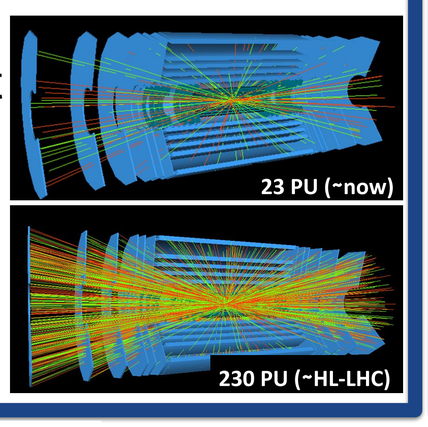
GNN-based Tracking and FPGA Acceleration

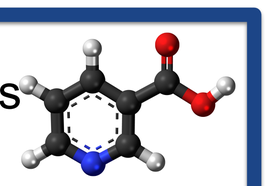
Team: M. Atkinson[1], G. DeZoort[2], P. Elmer[2], B. Galewsky[1], M.S. Neubauer[1], I.R. Ojalvo[2], S. Thais[2]
Institutions: [1]*University of Illinois at Urbana-Champaign*, [2]*Princeton University*

Tracking Challenge @ HL-LHC

The determination of charged particle trajectories in collisions at the Large Hadron Collider (LHC) is an important but challenging problem, especially in the high interaction pileup conditions present during HL-LHC running



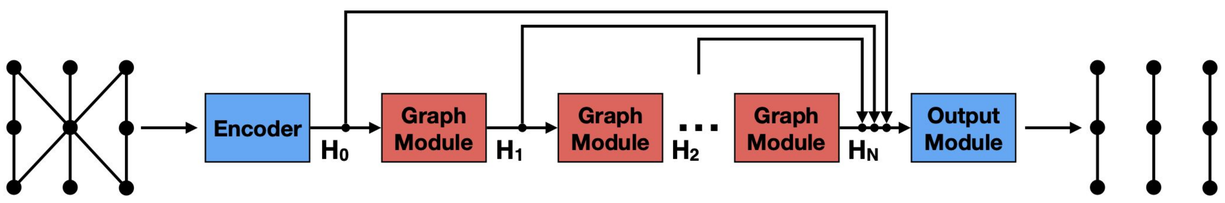
Graph Neural Networks



- Data relationships in many real-world applications can be naturally represented by graphs
- **Graph Neural Networks** (GNNs) are deep learning based methods that capture dependencies on graphs via message passing between the nodes of graphs [1]
- GNNs are well-suited to pattern recognition—a key element of reconstructing charged particles in tracking detectors [2]

Methodology

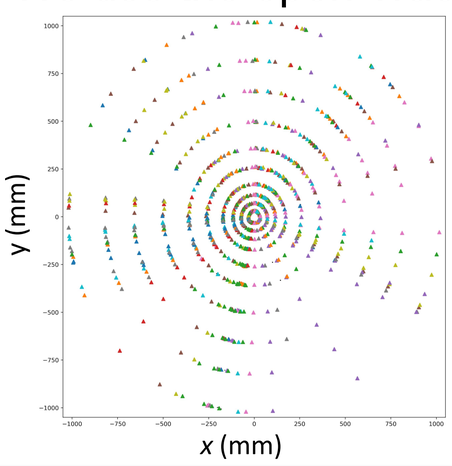
Our baseline is an approach described in [2]:

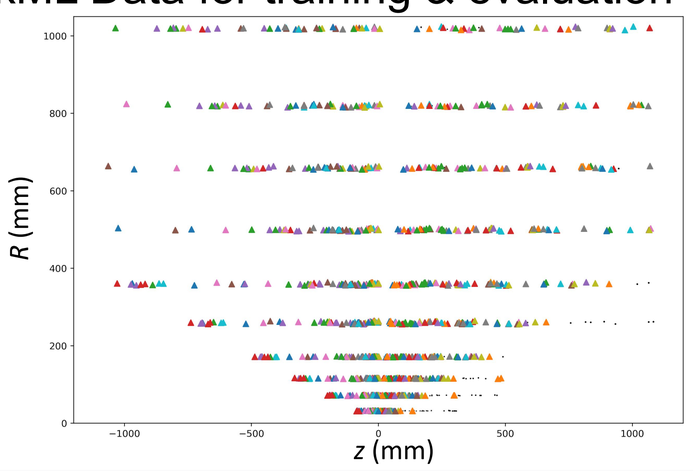


Tracking detector measurements are represented as graph nodes which are associated with one another by learned graph edges that represent the particle tracks

Data

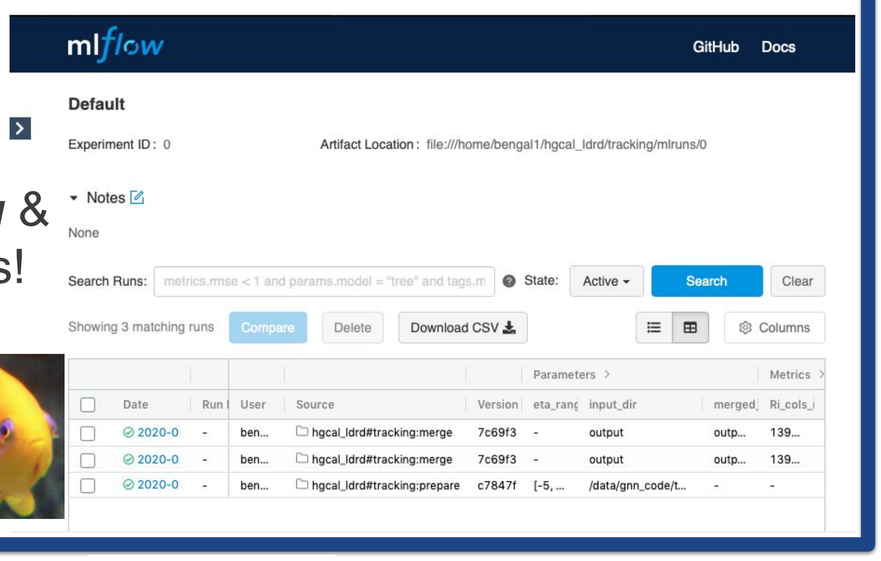
We use the open TrackML Data for training & evaluation



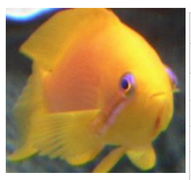


MLOps for Productivity & Reproducibility

We are using **MLFlow** to manage and track our ML training workflows
→ ***Provenance tracking*** & ***reproducibility*** of results!

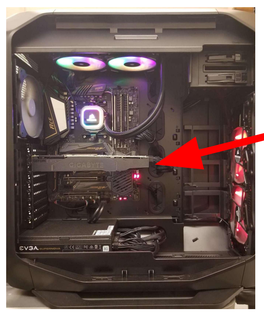


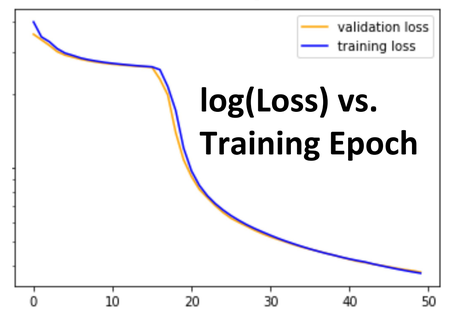
This is part of a broader effort on scalable CI for reproducibility in the NSF **SCAILFIN** Project



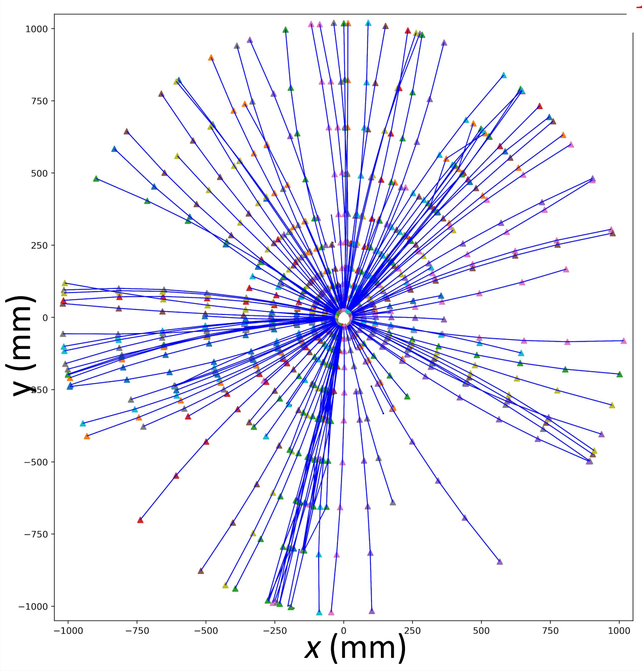
Training

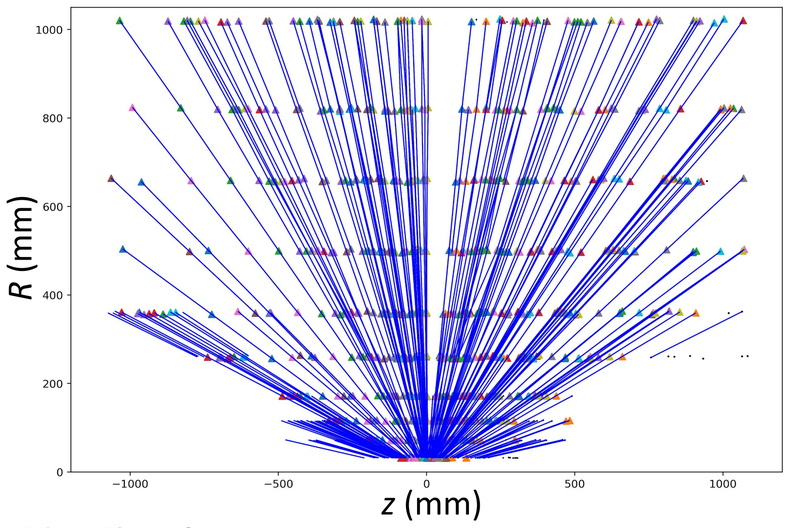
- Preprocessing for GNN models use HEP.TrkX libraries [3] with a truth particle $p_T > 2$ GeV cut
- Trained on 7080 evts / Validated on 1770 evts
- ~30 mins to train on the U. Illinois ML platform





Results





Edge Classification Summary:

Signal Efficiency: 98.6%	Noise Efficiency: 97.0%
Signal Fakes: 3.0%	Noise Fakes: 1.4%

FPGA Acceleration

- GNN-based inference can be implemented on FPGAs to accelerate computationally expensive parts of the event reconstruction such as calorimetry and tracking in the ATLAS or CMS High-Level (software) trigger
- Evaluate GNN tracking performance & utilization on Xilinx & Intel devices
- Coordinate with SSL and leverage resources in the Innovative Systems Lab @ NCSA

Plans

- Systematic study of tracking performance for architecture and hyperparameter variants, including addition of a regression NN
- Study performance (e.g. fake rates) when only including pixel hits
- Development & testing of GNN implementation (HLS) on FPGAs

References

[1] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner and G. Monfardini, "The Graph Neural Network Model," in *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61-80, Jan. 2009

[2] X. Ju, *et al*. "Graph Neural Networks for Particle Reconstruction in HEP Detectors", Workshop on ML for the Physical Sciences, NeurIPS19, Vancouver, CA

[3] https://github.com/esaliya/heptrkx-gnn-tracking

Acknowledgements: **We thank Lindsey Gray for his help and code as part of Exa.TrkX and Javier Duarte for his guidance on GNN FPGA implementations and discussions of his prior work within the HLS4ML group. The studies described in this document are motivated by the results of Exa.TrkX and LDRD L2019.017.**

This project is supported by National Science Foundation under Cooperative Agreement OAC-1836650. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.