Development of a framework based on PyTorch Lightning for training Graph Neural Networks (GNNs) for tracking

14ème Journées Informatiques IN2P3/IRFU (14-17 novembre 2022)

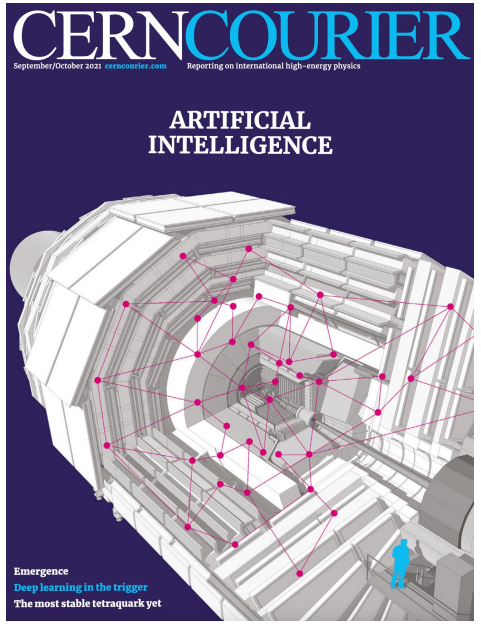
Sylvain Caillou

(On behalf of L2IT « Computing, Algorithms and Data » Team)

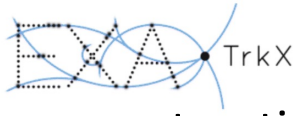
Laboratoire des 2 Infinis - Toulouse (L2IT-IN2P3), Université de Toulouse, CNRS, UPS

Context

⇒ Geometric ML have become one of the hottest fields of AI research
⇒ Increasingly popular for a large number of LHC physics tasks
⇒ GNNs perform pretty well to learn geometric pattern of particles tracks

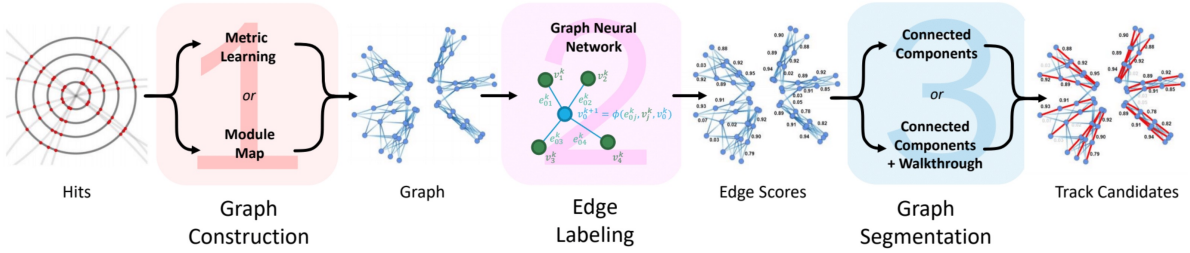


Collaboration L2IT & ExatTrkX

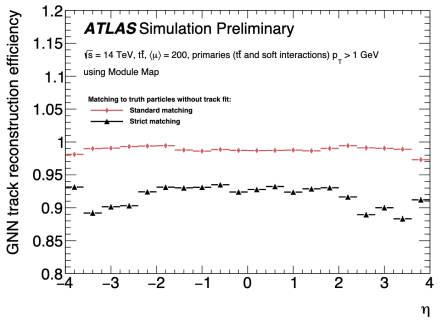


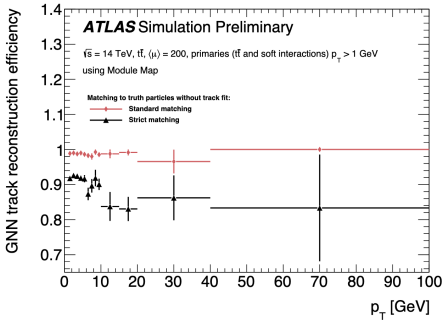
BERKELEY LAB

⇒ Construct a GNN-based track reconstruction algorithm for ATLAS Itk



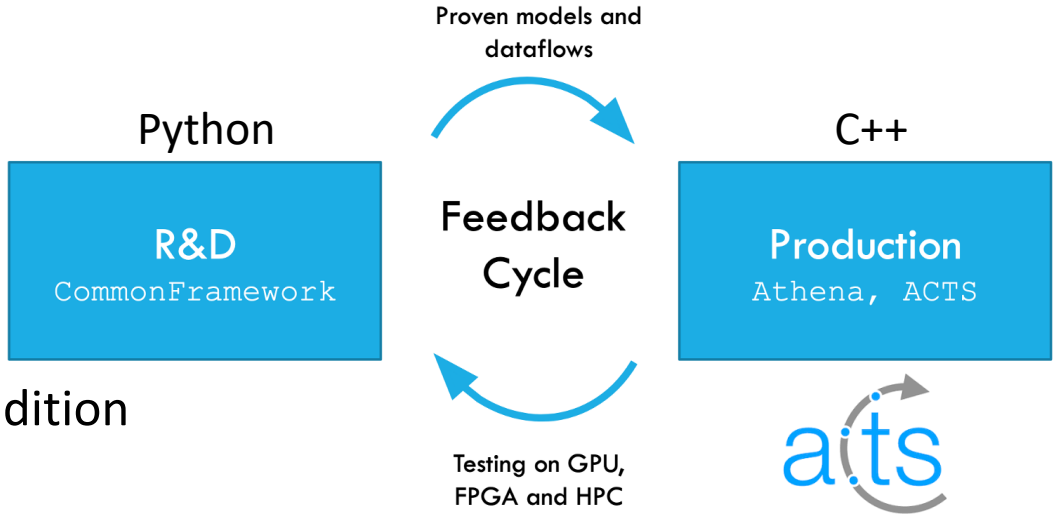
⇒ First results on ITk published in 2022 more than encouraging
⇒ GNN-based algorithms now appear as competitive solutions
⇒ Put into production (ACTS) for the HL-LHC





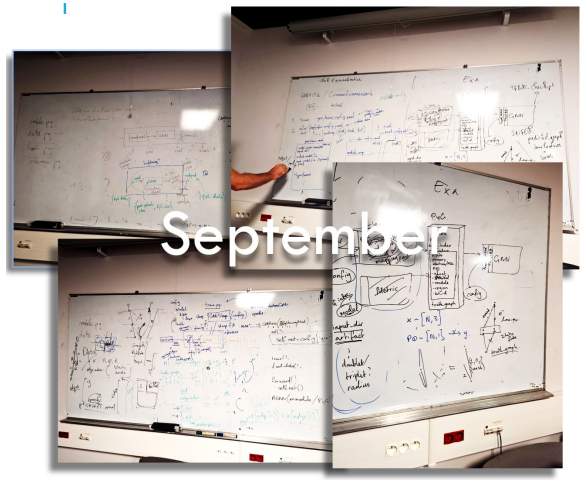
ATLAS ITk Track Reconstruction with a GNN-based pipeline, C.Rougier et al., CTD 2022
Graph Neural Network track reconstruction for ATLAS ITk , D. Murnane et al., IML 2022

Motivation



ML R&D in HEP take a LOT of time and (GPU) resources

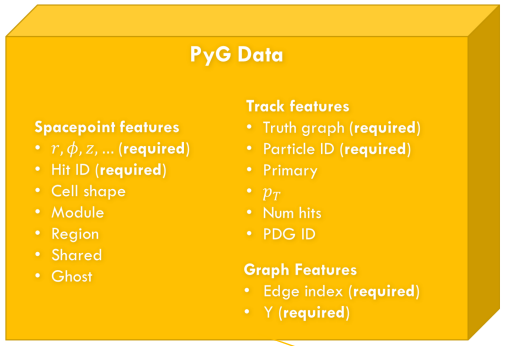
- ML R&D: Models and hyperparameters exploration
- Data: Complexity of the simulated detector data in HL-LHC condition
  (Pythia + GEANT4 + ITk geometry)
- (International) Collaboration:
  - Do we speak the same language (share the same semantic) ?
  - Data format
  - Experimental conditions
  - Performance metrics definition & implementation
- Reproductibility: Lack of traceability of experimental conditions
- Software quality: Messy idea implemented, no documentation, no test, etc…



Improve the process => save some R&D time
- ⇒ Get faster for research
- ⇒ Use less (GPU) resources and do more sober R&D

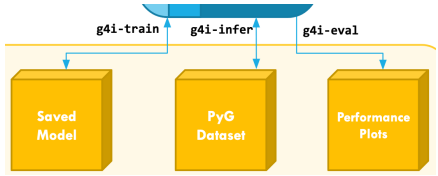
⇒Need to develop of a common R&D framework

gnn4ITk common framework (developped with D. Murnane, LBNL)

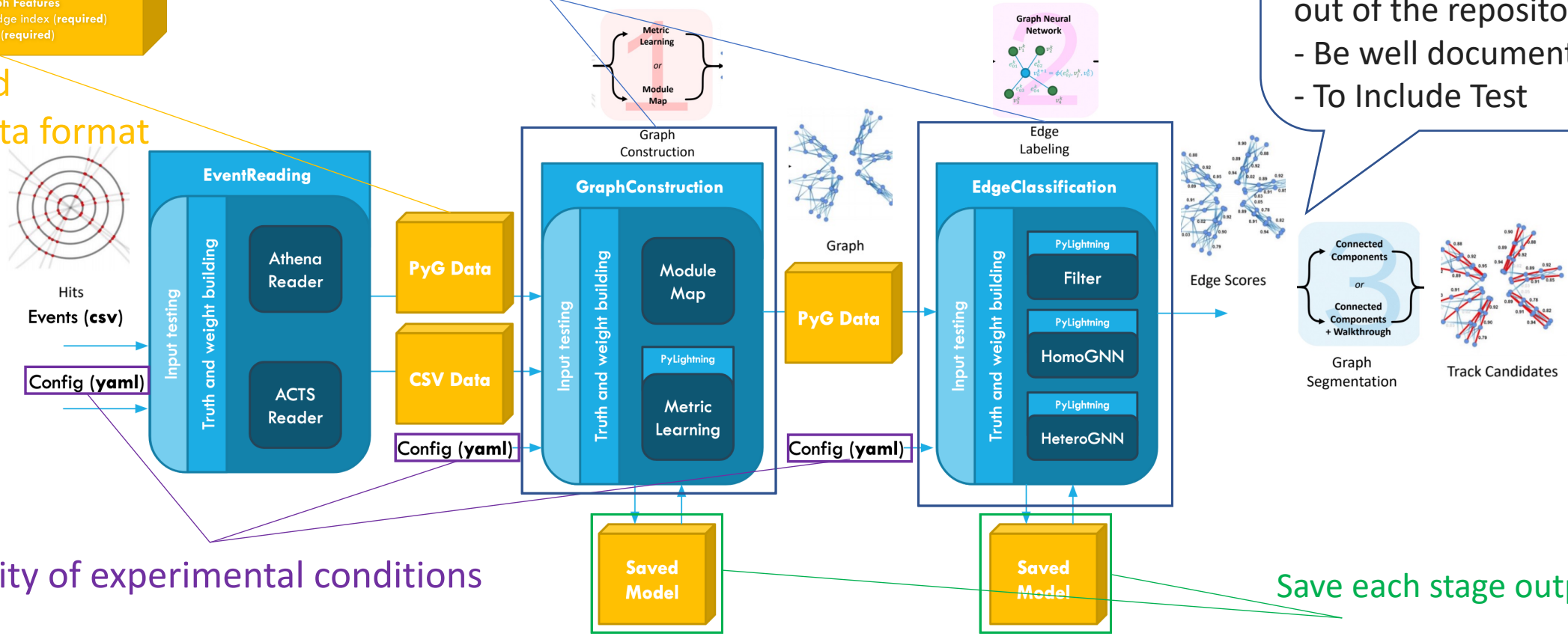


Well defined common data format

Modularity: Pipeline divided in stage
Each stage can include several models
It can be *train* or *infer* or *evaluate*
Share the same evaluation method



Also try to:
- Be Easy to use
- Be clean (keep messy dev & random ideas out of the repository)
- Be well documented
- To Include Test



Traceability of experimental conditions

Save each stage output

Next steps…

- Reproduce performance on approval dataset
- Use the framework (and no more home code) for the next studies
- Plublish it with open data demo
- Towards a common platform / tools for Geometric ML R&D in HEP ?