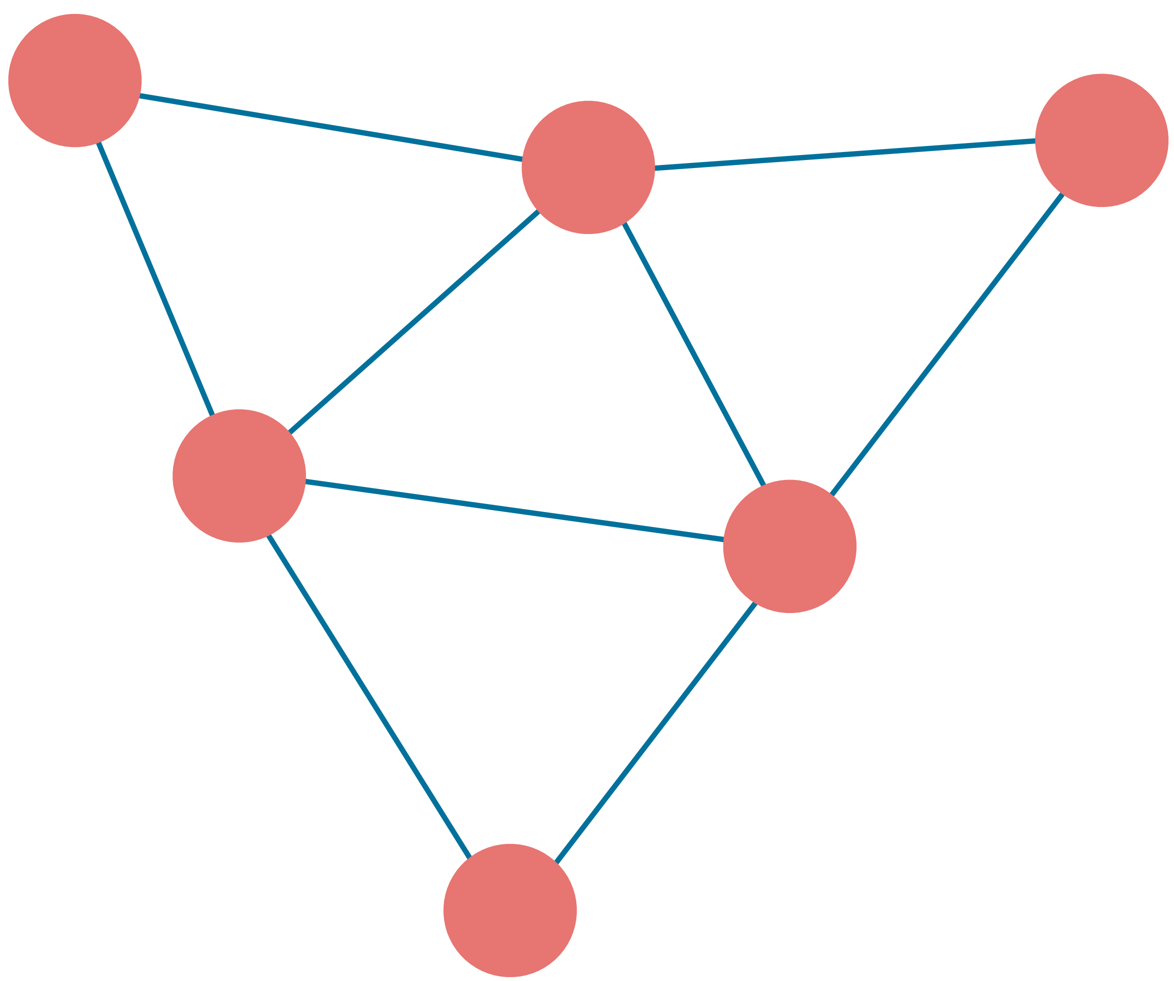
Multi-head attention network for DUNE reconstruction

Jeremy Hewes
DUNE reconstruction workshop
18th September 2020

Graph neural networks

- Describe information structure as a **graph** represented by **nodes** and **edges**.

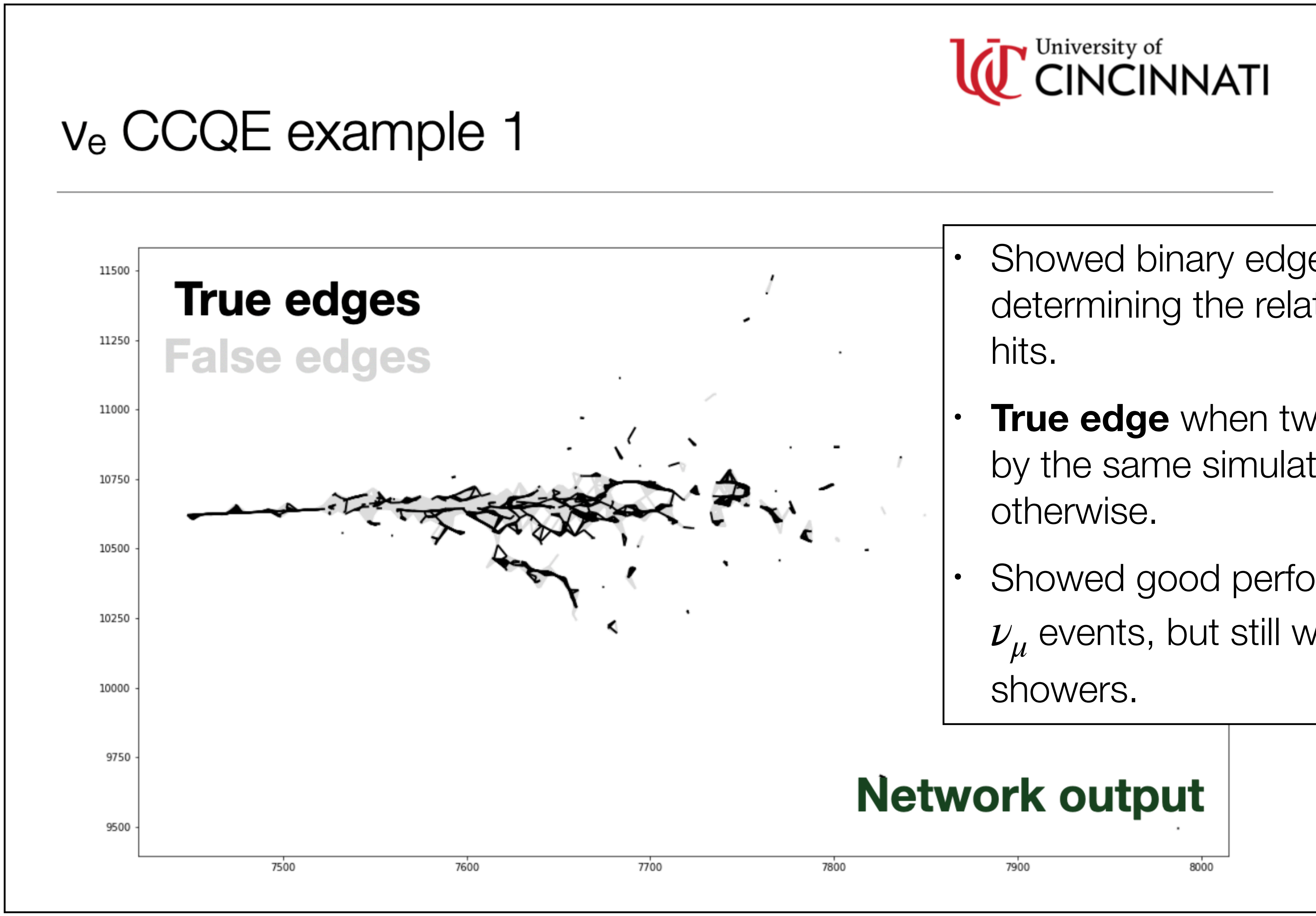


- **Nodes** are generalised as quantised objects with some arbitrary set of **features**.
- **Edges** describe the **relationships** between nodes.
- Perform convolutions on nodes and edges to learn relationships within the graph.
- Output is user-defined:
  - Classify nodes or edges.
  - Classify full graph.
  - Regression outputs.

Reminder

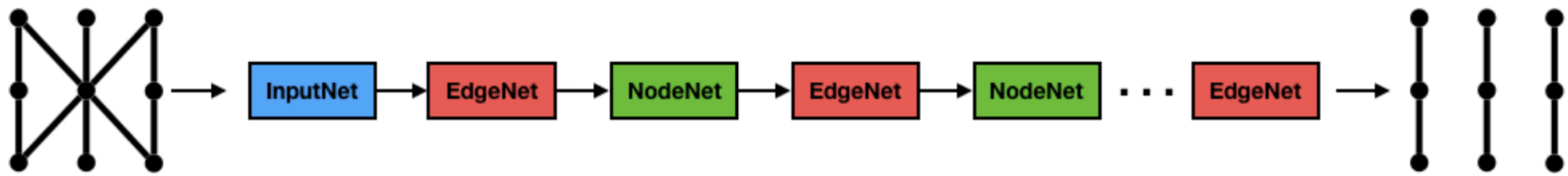
- Developing **graph neural network (GNN)**-based reconstruction in the **DUNE far detector**.
- Using **pytorch-geometric** toolkit for graph networks.
- Working with **CCQE beam neutrino simulation** in the full DUNE 10kt geometry.
  - 10,000 events each in nonswap and fluxswap configurations.
  - Approximately equal representation of ν_e and ν_μ interactions.
  - Standard simulation chain, low-level reconstruction (hit-finding).
  - Using pre-refactor Geant4, so enable shower daughters to retain information on EM system.
- Utilise Nvidia DGX GPU cluster with 8 Tesla V100 GPUs.
- Preprocess HDF5 inputs into .pt files which are used for training.

Reminder



- Showed binary edge classifier for determining the relationship between 2D hits.
- **True edge** when two hits were produced by the same simulated particle; **false** otherwise.
- Showed good performance for track-like ν_μ events, but still work to be done for showers.

Attention message-passing networks



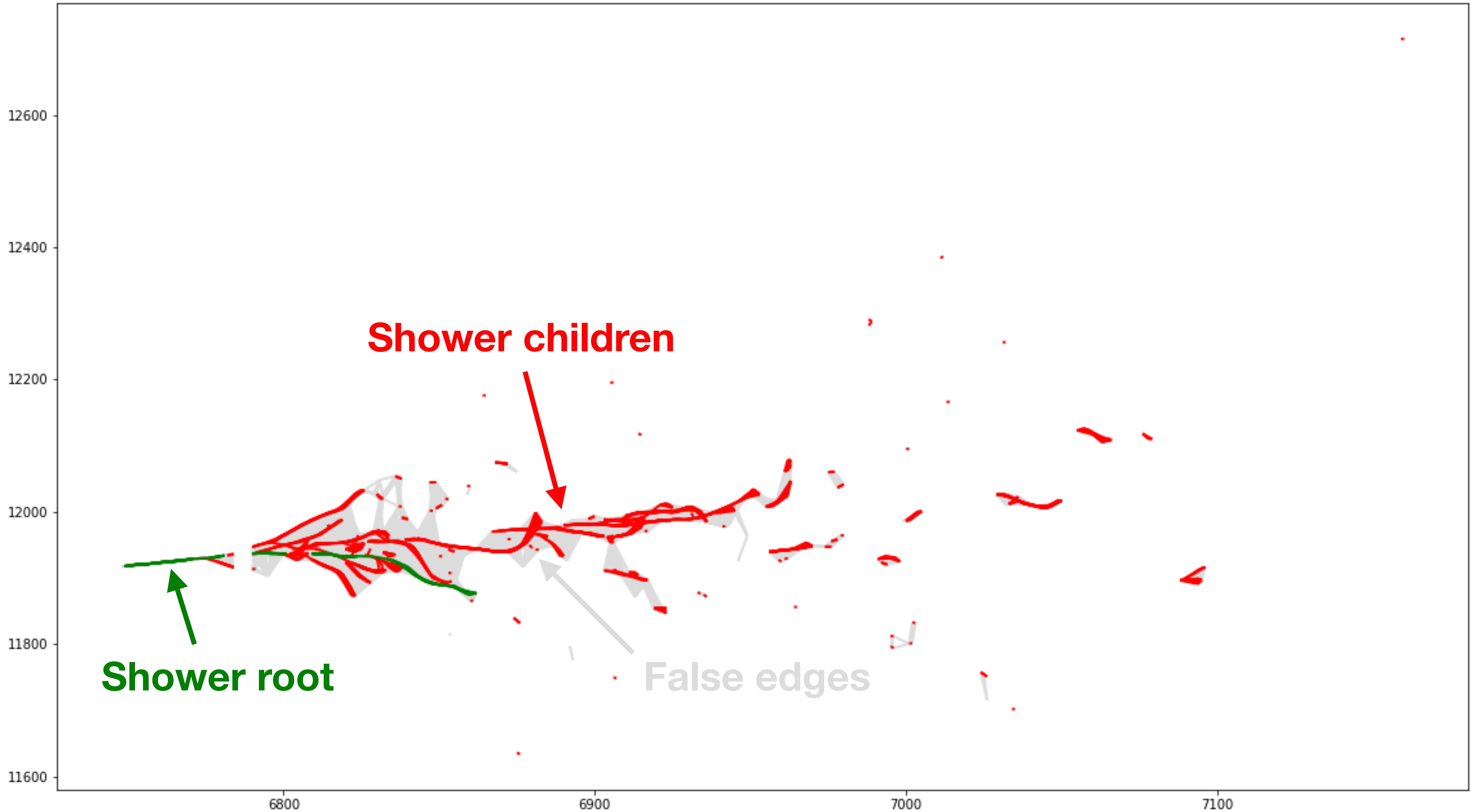
arxiv:1810.06111

- Message-passing network aggregates information from neighbouring nodes across edges to form new features on each node, utilising an attention mechanism to weight up useful edges.
- Repeat the same network multiple times in order for information to travel further across the graph over multiple iterations (the “message passing”).
- **Edge classifier:**
  - Input for each node is the features of incoming and outgoing nodes.
  - Two multi-layer perceptrons, using Tanh and sigmoid activations.
  - Outputs sigmoid score on each edge.
- **Node classifier:**
  - Uses edge score to aggregate each node’s features with incoming & outgoing edges as input.
  - Two multi-layer perceptrons with Tanh activation.
  - Produces new features for each node.

Flavoured edges

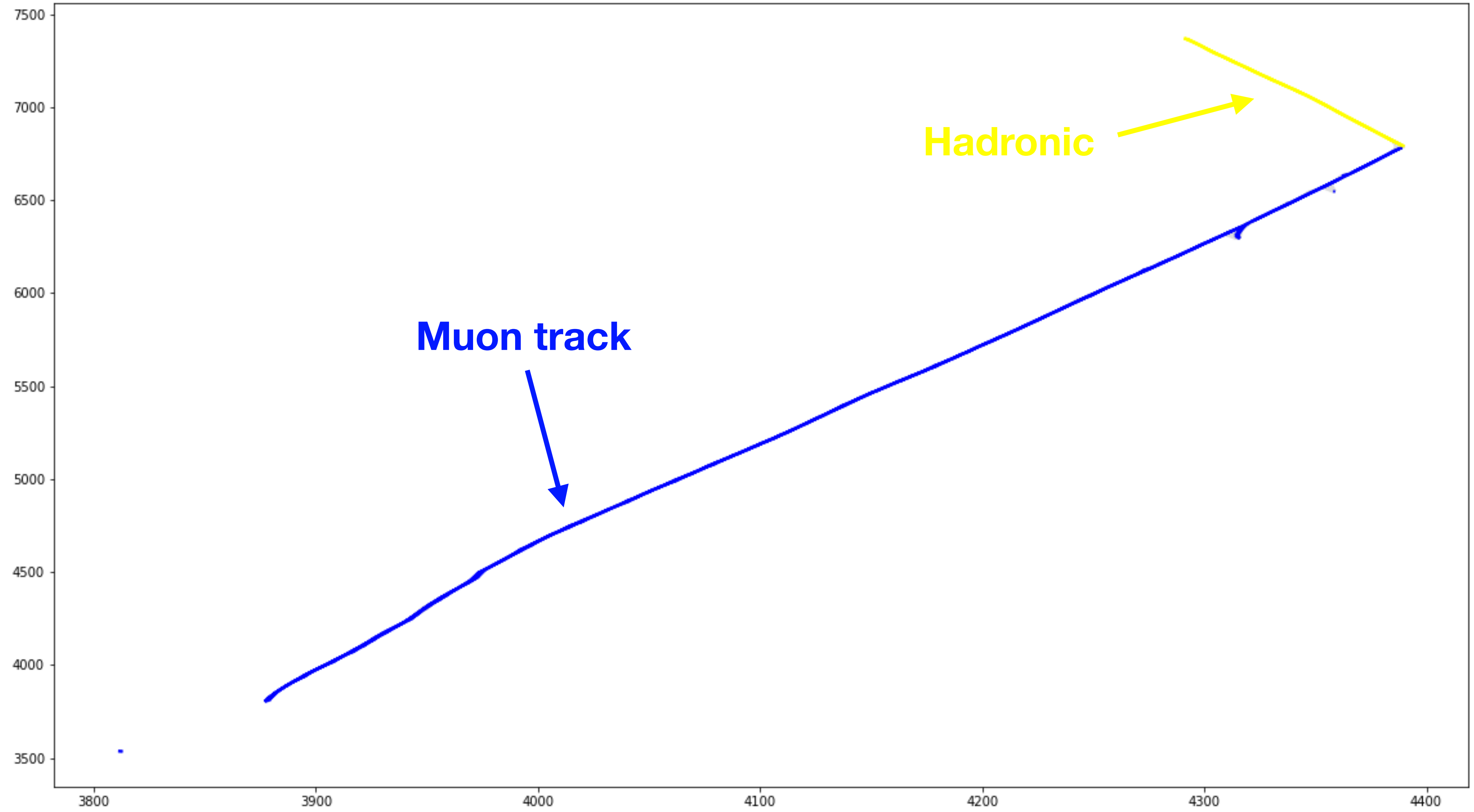
- For binary edge classification, observed that the network responded differently for tracks and showers.
- Track edges usually far from threshold, while shower edges very close to threshold.
- Try explicitly defining separate classes to help the network disambiguate.
- Expand binary truth (true/false edges) to flavour edges with different true scores.
  - True vs false definition remains the same.
  - True edges are now classified based on particle type:
    - Primary **muon** and descendents.
    - **Electron shower root**.
    - **Electron shower children**.
    - Everything else (for these CCQE events this means **hadronic**).

ν_e example

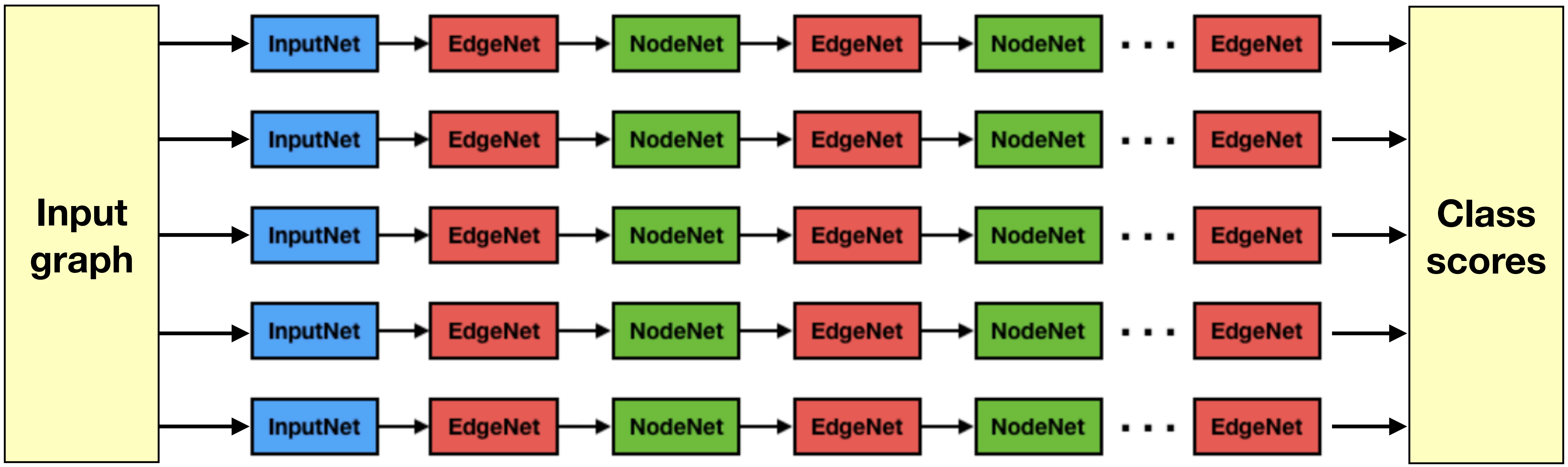


Shower children
Shower root
False edges
12600
12400
12200
12000
11800
11600
6800
6900
7000
7100

ν_μ example

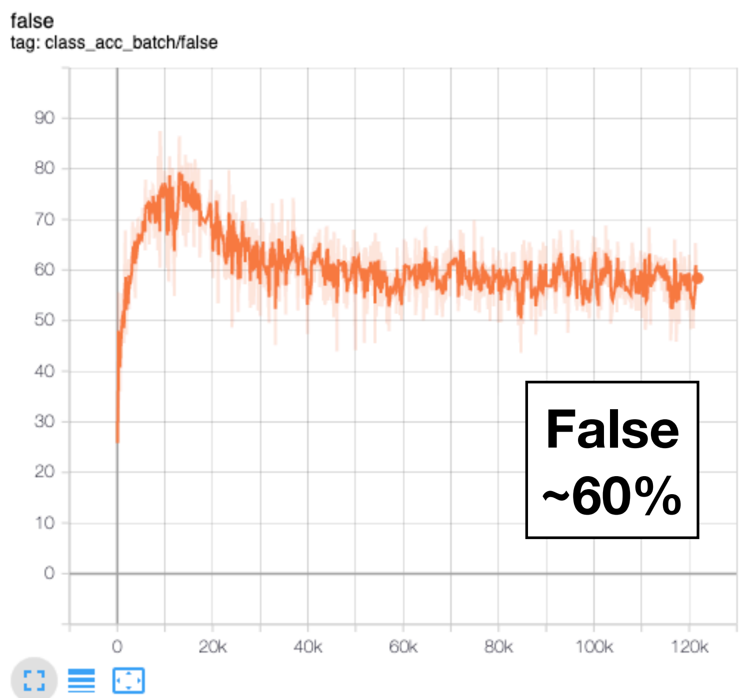


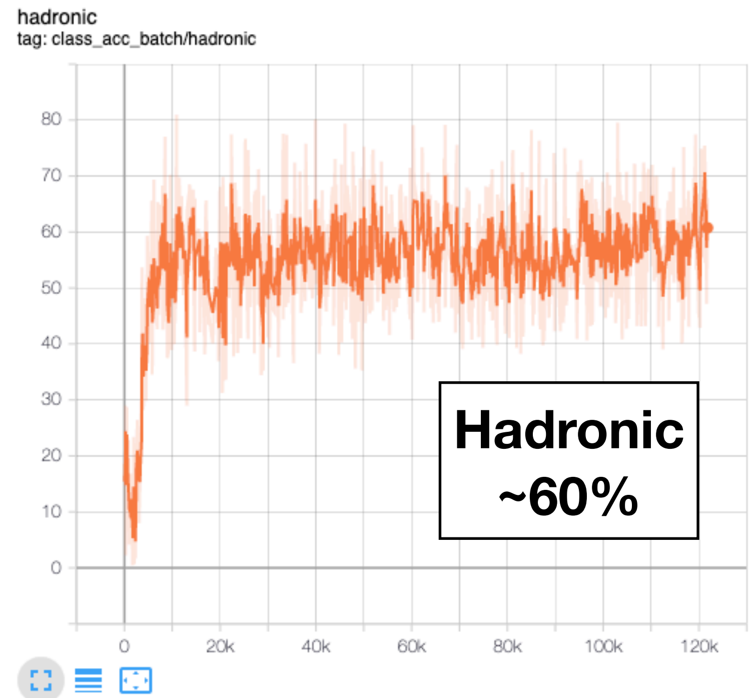
Multiclass attention network



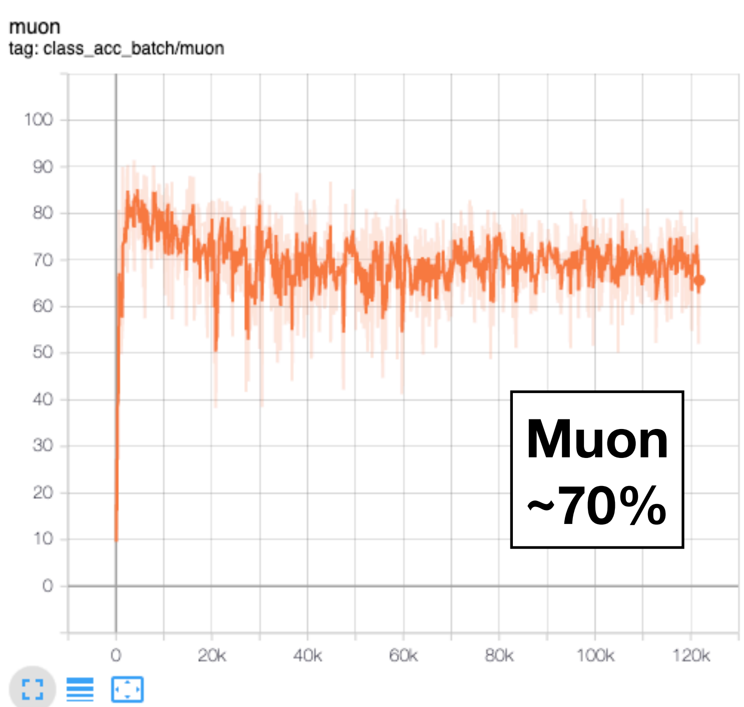
- Take our initial message-passing network and stack several of them in parallel.
- Each stack learns to classify a separate class.
  - Because each stack is disconnected, the mechanism for information flow can vary between different classes.
- Each stack outputs a single attention score on each edge.
- We can then take the softmax of these scores to get class probabilities for each edge.

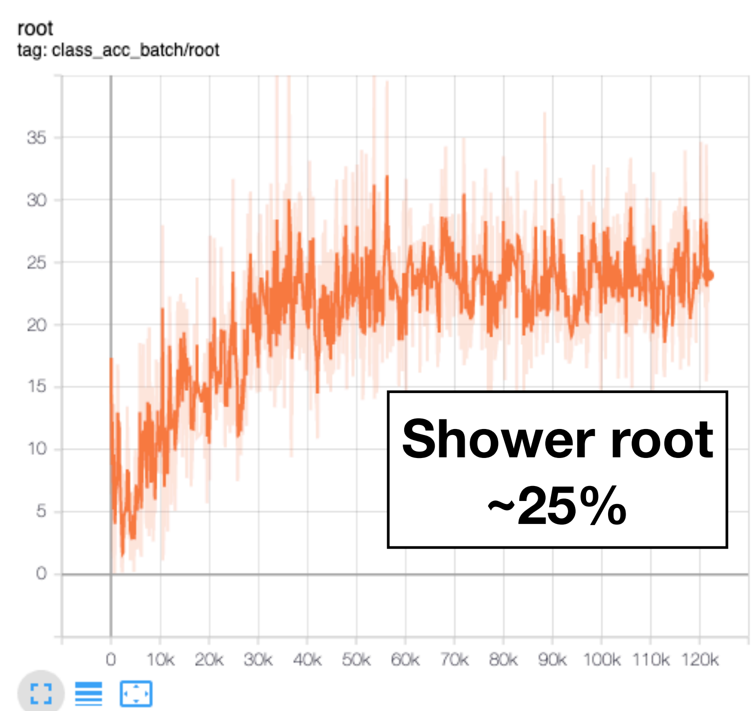
Network performance

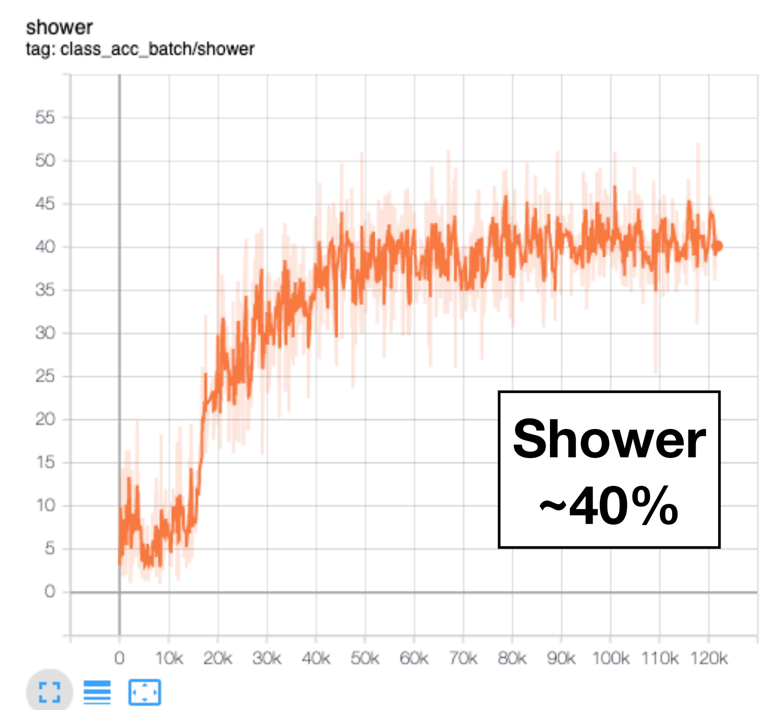




- Network appears to learn well for false and track-like classes.
- Accuracy for shower and shower root classes remains low.







University of CINCINNATI

Next steps

- Again, multi class network seems to show good promise for track-like particles, but more work required on showers.

- Next step is to define a simpler ground truth:
  - Combine shower root and shower children truths into a single object.
  - Rather than reconstructing the internal structure of the shower, just treat the whole shower as a single object with dense connections.

- Other to-do list items still outstanding from previous update:
  - Test on new inclusive sample I have produced.
  - Explore graph pooling techniques for hit clustering.