Machine-Learning Approaches to Particle Track Reconstruction in the TrackML Dataset

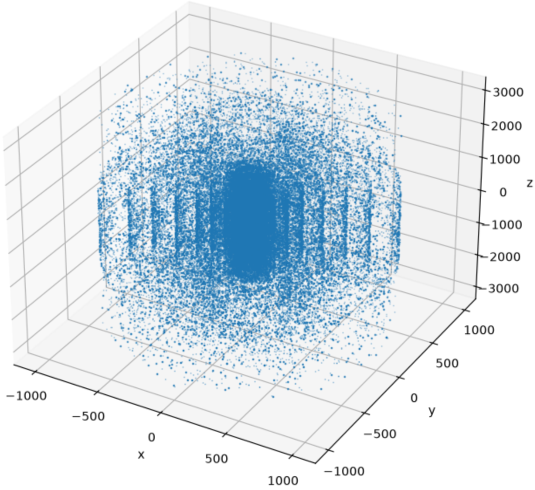
Machine Learning for Quantum Scientists

Konrad Beck

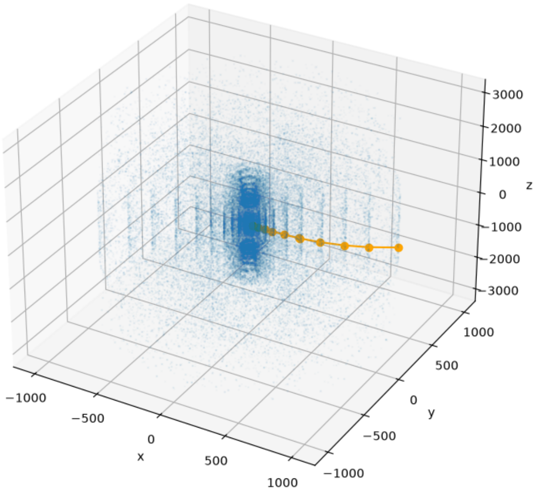
Rheinische Friedrich-Wilhelms-Universität Bonn

31.07.2026

The TrackML Challenge



The TrackML Challenge



Outline

Pair Classification Pipeline

$$(h_1, h_2) \to [0, 1]$$

Hit pair Preselection $\longrightarrow$ DNN $\longrightarrow$ Classifier $\longrightarrow$ Track Reconstruction

Candidate Edge Preselection

For every hit h:

- Nearest k neighbors are searched in next outer layer
- An edge $e = (h, k)$ is considered if the two hits obey the angular conditions

$$\boxed{|\Delta\phi| < 0{,}12\,\mathrm{rad} \qquad |\Delta\eta| < 0{,}20}$$

pseudorapidity:

$$\eta = -\ln\left[\tan\left(\frac{\theta}{2}\right)\right]$$

Pair Model: DNN Classifier

Input pair

$$\mathbf{x}_{ij} = \begin{pmatrix} \text{hit}_i \\ \text{hit}_j \end{pmatrix} \in \mathbb{R}^{52}$$

DNN shape

$$52 \to 8000 \to [4000]_{\times 3} \to [2000]_{\times 3} \to 1$$

Logit

$$s_{ij} = f_\theta(\mathbf{x}_{ij})$$

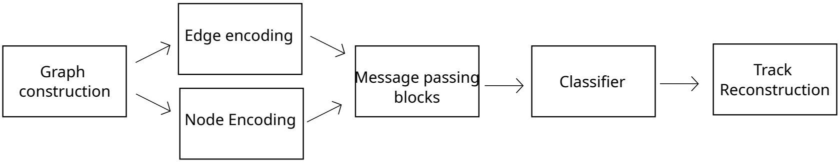
Probability

$$p_{ij} = \sigma(s_{ij}) = \frac{1}{1 + \exp(-s_{ij})}$$

Loss

$$\mathcal{L} = -\frac{1}{N} \sum_{n=1}^{N} [y_n \log \sigma(s_n) + (1 - y_n) \log(1 - \sigma(s_n))]$$

$$(V, E) \rightarrow [0,1]^{|E|}$$



GNN Graph Construction

Stored features

$$\mathbf{x}_i = \begin{pmatrix} r_i \\ \phi_i \\ z_i \end{pmatrix} \in \mathbb{R}^3$$

$$\mathbf{a}_{ij} = \begin{pmatrix} \Delta r_{ij} \\ \Delta \phi_{ij} \\ \Delta z_{ij} \\ \Delta \eta_{ij} \end{pmatrix} \in \mathbb{R}^4$$

$$X \in \mathbb{R}^{N \times 3}, \qquad A \in \mathbb{R}^{E \times 4}$$

Sparse graph storage

$$\texttt{source} = \begin{pmatrix} s1 & s2 & s3 & \cdots \end{pmatrix}$$

$$\texttt{destination} = \begin{pmatrix} d1 & d2 & d3 & \cdots \end{pmatrix}$$

$$\text{edge } e: \quad \texttt{source}[e] \rightarrow \texttt{destination}[e]$$

GNN Network

Encoding

$$X \in \mathbb{R}^{N \times 3} \xrightarrow{3 \to 42 \to 42} H^{(0)} \in \mathbb{R}^{N \times 42} \qquad A \in \mathbb{R}^{E \times 4} \xrightarrow{4 \to 42 \to 42} E^{(0)} \in \mathbb{R}^{E \times 42}$$

Message block

edge update: $\left[h_{s_e}^{(l)}, h_{d_e}^{(l)}, e_e^{(l)}\right] \in \mathbb{R}^{E \times 126} \xrightarrow{126 \to 42 \to 42} E^{(l+1)} \in \mathbb{R}^{E \times 42}$

aggregation: $M_{\mathrm{in}}, M_{\mathrm{out}} \in \mathbb{R}^{N \times 42}$

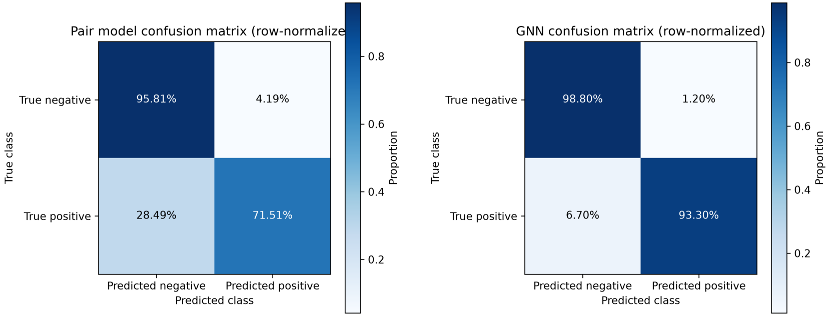
node update: $\left[H^{(l)}, M_{\mathrm{in}}, M_{\mathrm{out}}\right] \in \mathbb{R}^{N \times 126} \xrightarrow{126 \to 42 \to 42} H^{(l+1)} \in \mathbb{R}^{N \times 42}$

$l = 0, \dots, 3$ four message-passing blocks

Classification

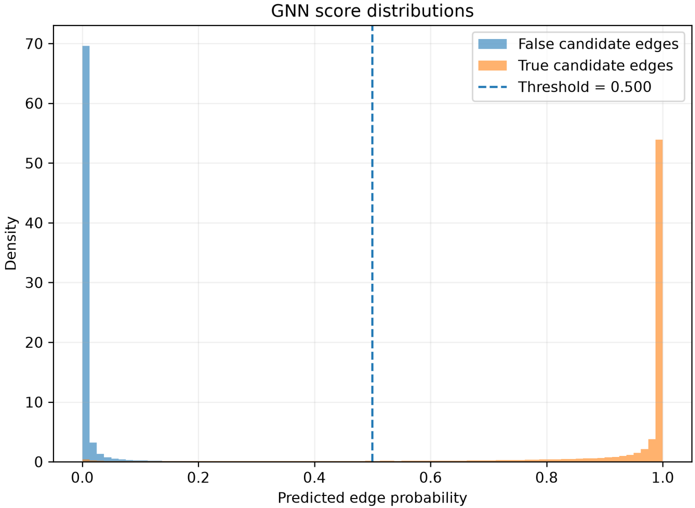
$$E^{(4)} \in \mathbb{R}^{E \times 42} \xrightarrow{42 \to 42 \to 1} s \in \mathbb{R}^{E \times 1} \qquad p = \sigma(s) \in \mathbb{R}^{E \times 1}$$

Normalized Confusion Matrices

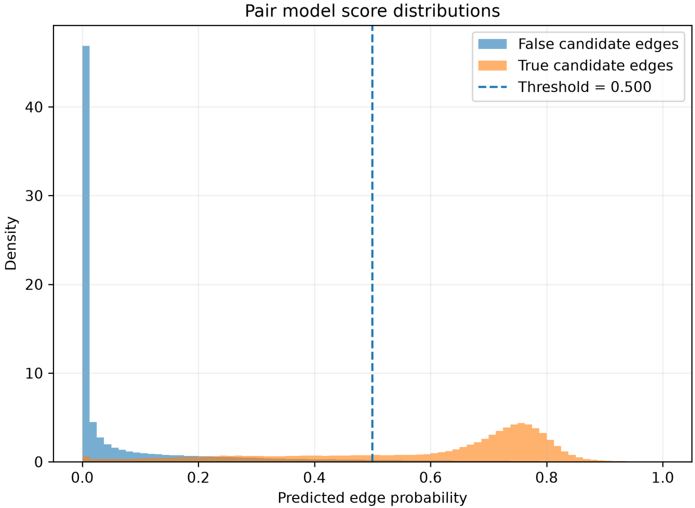


Pair model GNN

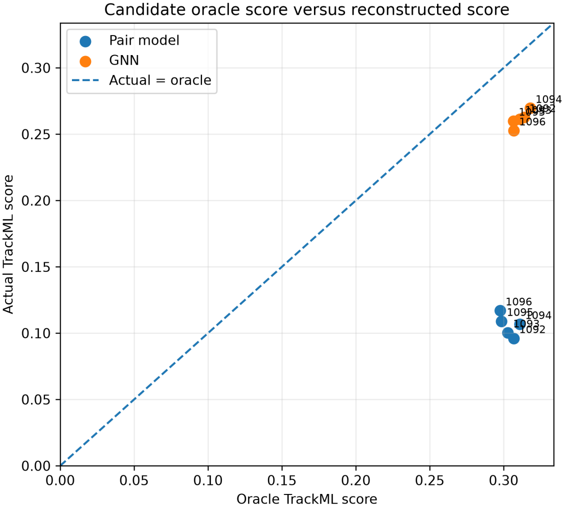
GNN Score Distributions



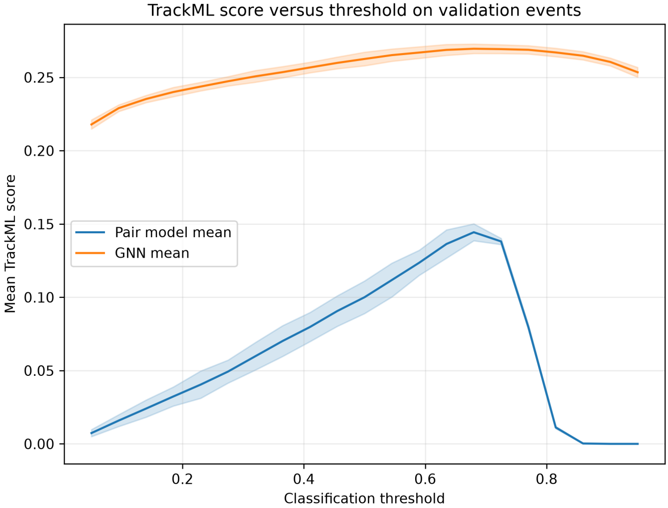
Pair Model Score Distributions



Candidate Oracle vs. Reconstructed Score



TrackML Score vs. Threshold



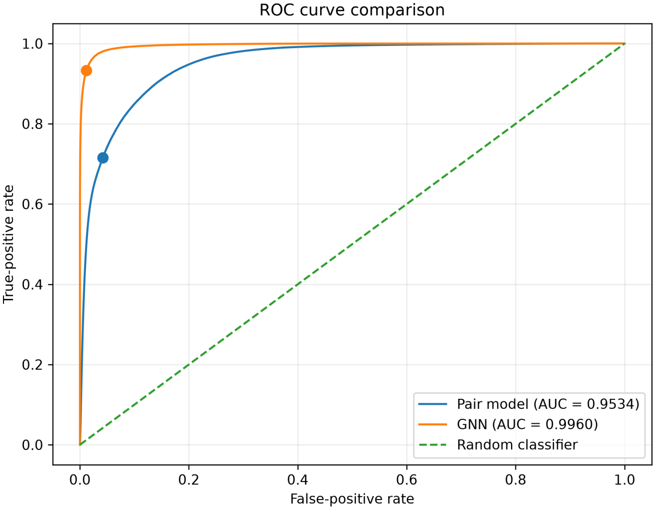
Conclusion and Possible Improvements

- Conclusion:
  - both models can get scores > 0
  - The GNN score is close to the oracle score
  - The main problem for both approaches seems to be the preprocessing and filtering of the data

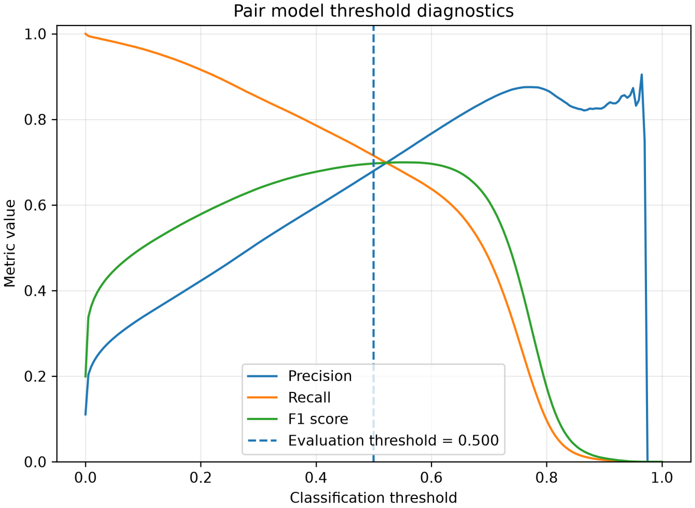
Conclusion and Possible Improvements

- Conclusion:
  - both models can get scores > 0
  - The GNN score is close to the oracle score
  - The main problem for both approaches seems to be the preprocessing and filtering of the data
- Possible Improvements:
  - Optimize threshold
  - Redesign the preprocessing pipeline
  - Increase GNN complexity and optimize hyperparameters
  - Fit curves through the tracks and recluster the hits based on the fits

ROC Curve


ROC curve comparison
True-positive rate
False-positive rate
0.0
0.2
0.4
0.6
0.8
1.0
Pair model (AUC = 0.9534)
GNN (AUC = 0.9960)
Random classifier

Pair Model Threshold Data



GNN Threshold Data