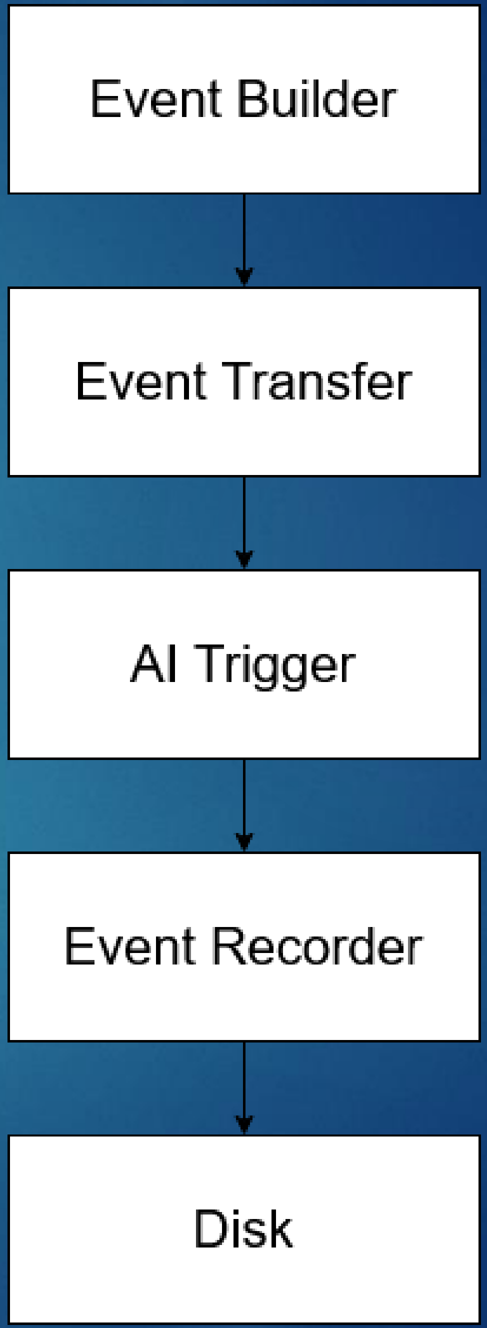
AI Level 3 Trigger

RICHARD TYSON

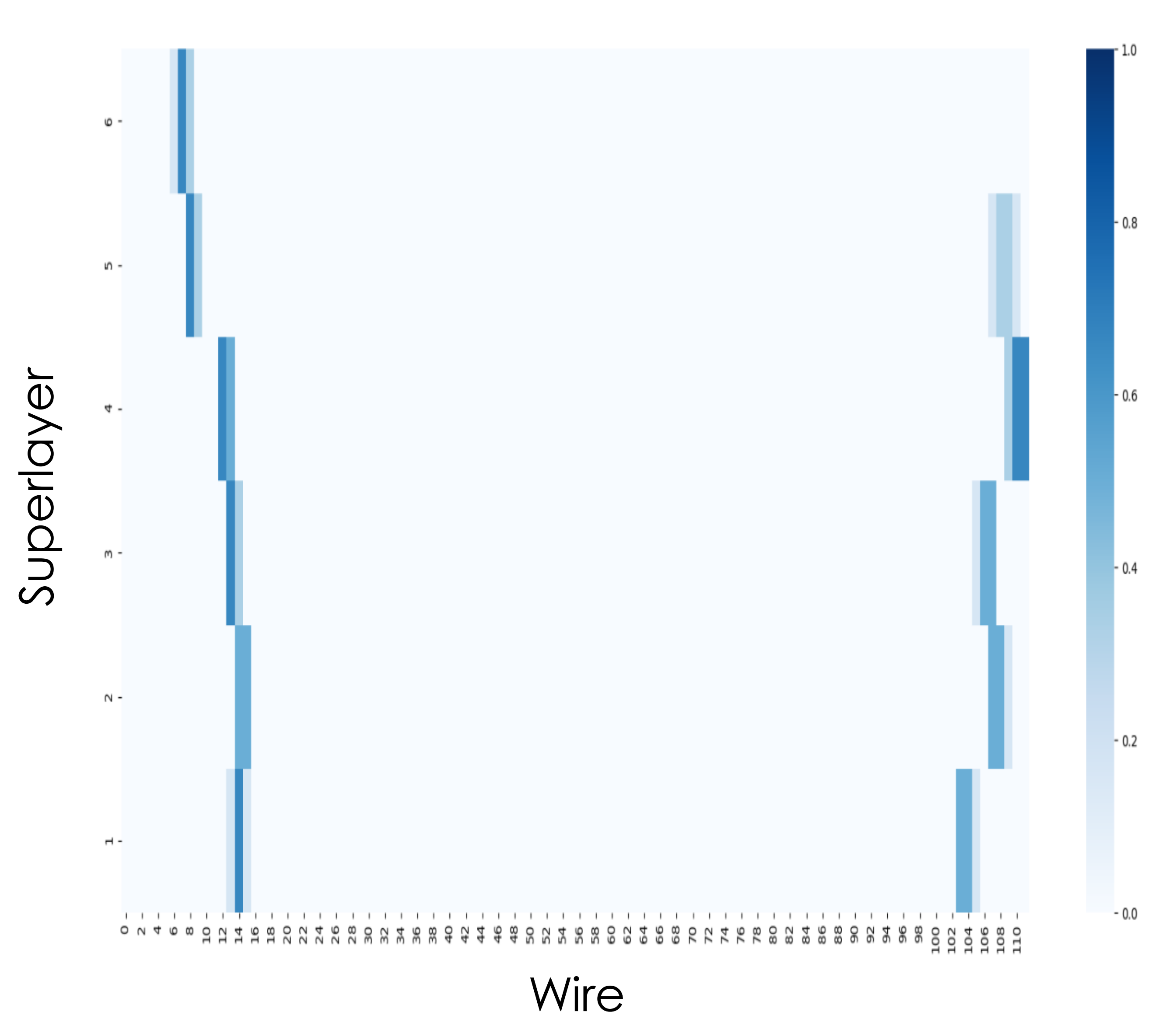
AI Level 3 Trigger

- The aim is to produce an (FD) electron trigger with a higher purity than the current electron trigger by using a neural network.
- It will learn to associate DC tracks with hits in the ECAL and recognize the energy deposition and momentum distributions characteristic of electrons.
- Once trained it can easily be loaded and deployed in the CLARA/COATJAVA workflow.
- The AI Trigger will sit between the Event Builder and the Event Recorder.



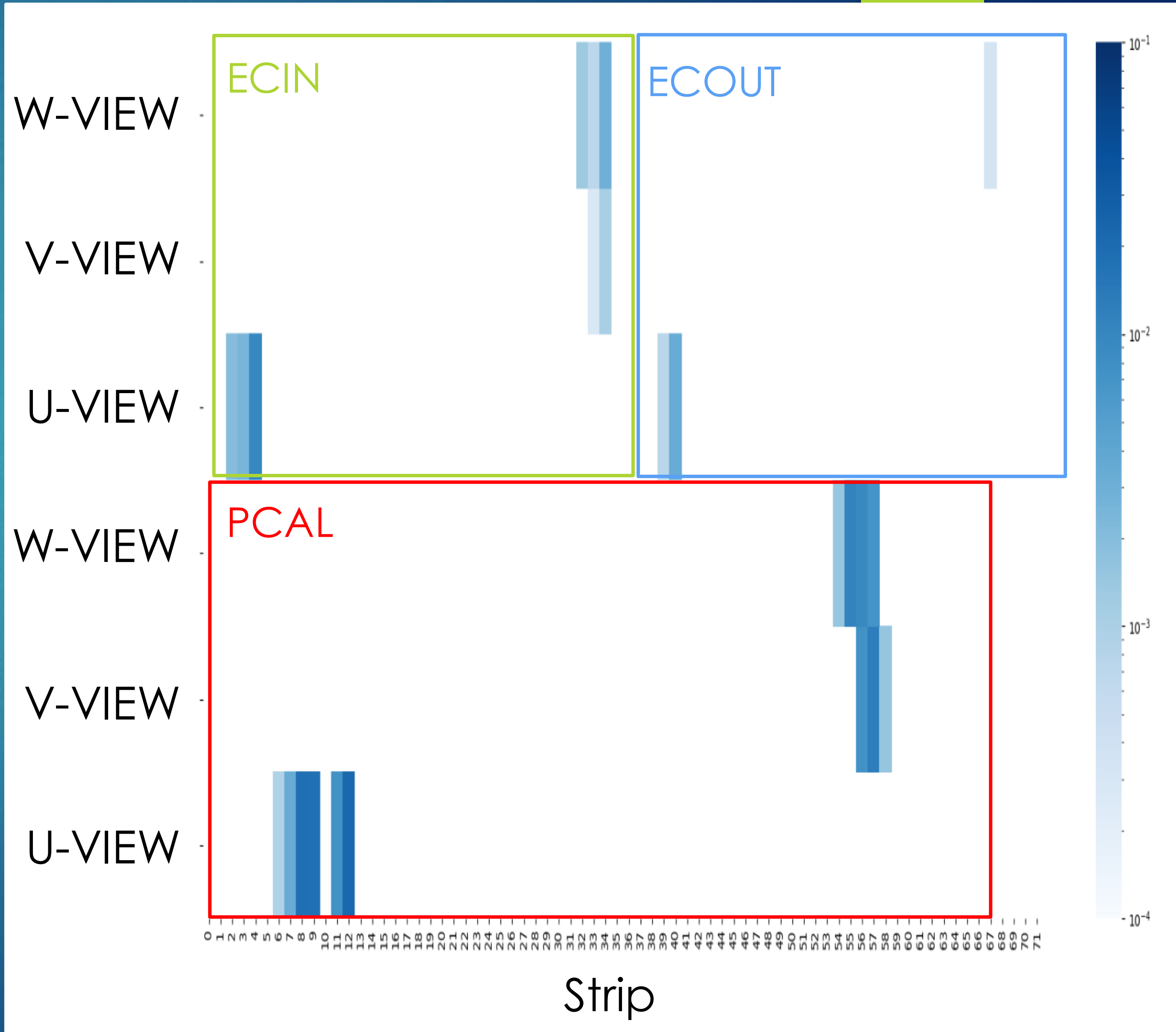
DC Tracking

- The DC is composed of 6 superlayers with 6 layers and 112 wires per layer.
- 6*112 array filled with 0 if a wire wasn't hit and 1/6 per hit per layer.
- Take tracks per sector, meaning we can have multiple tracks for each element of our training or testing samples.



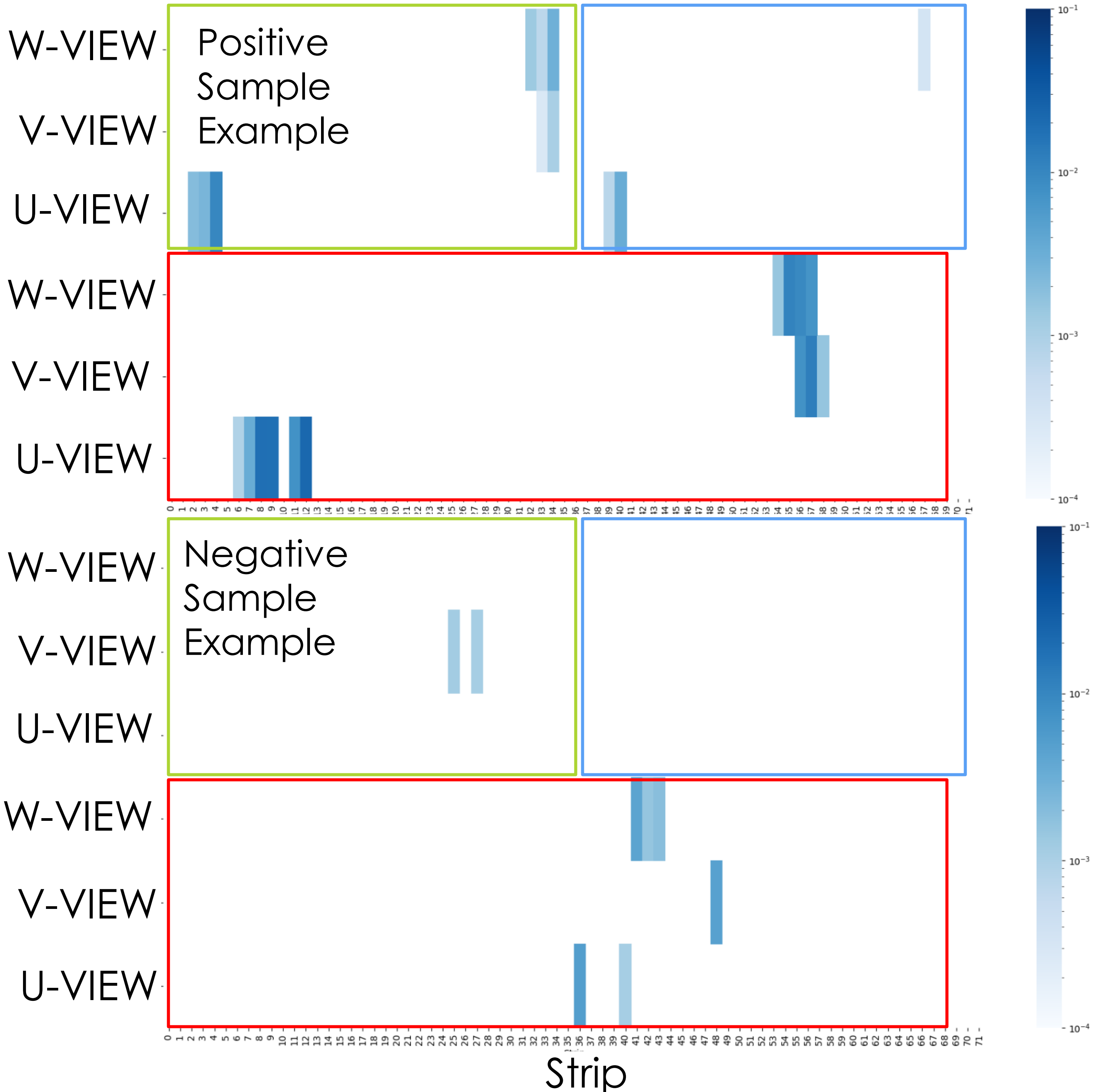
ECAL hits

- The ECAL has three views (U/V/W) for each of the PCAL, EC Inner, EC Outer.
- The PCAL has 68 strips in U and 62 in V/W. The ECIN and ECOUT have 36 strips in U/V/W.
- 6*72 array filled with the energy deposited in each strip.
- Take hits per sector, meaning we can have hits from multiple particles for each element of our training or testing samples.



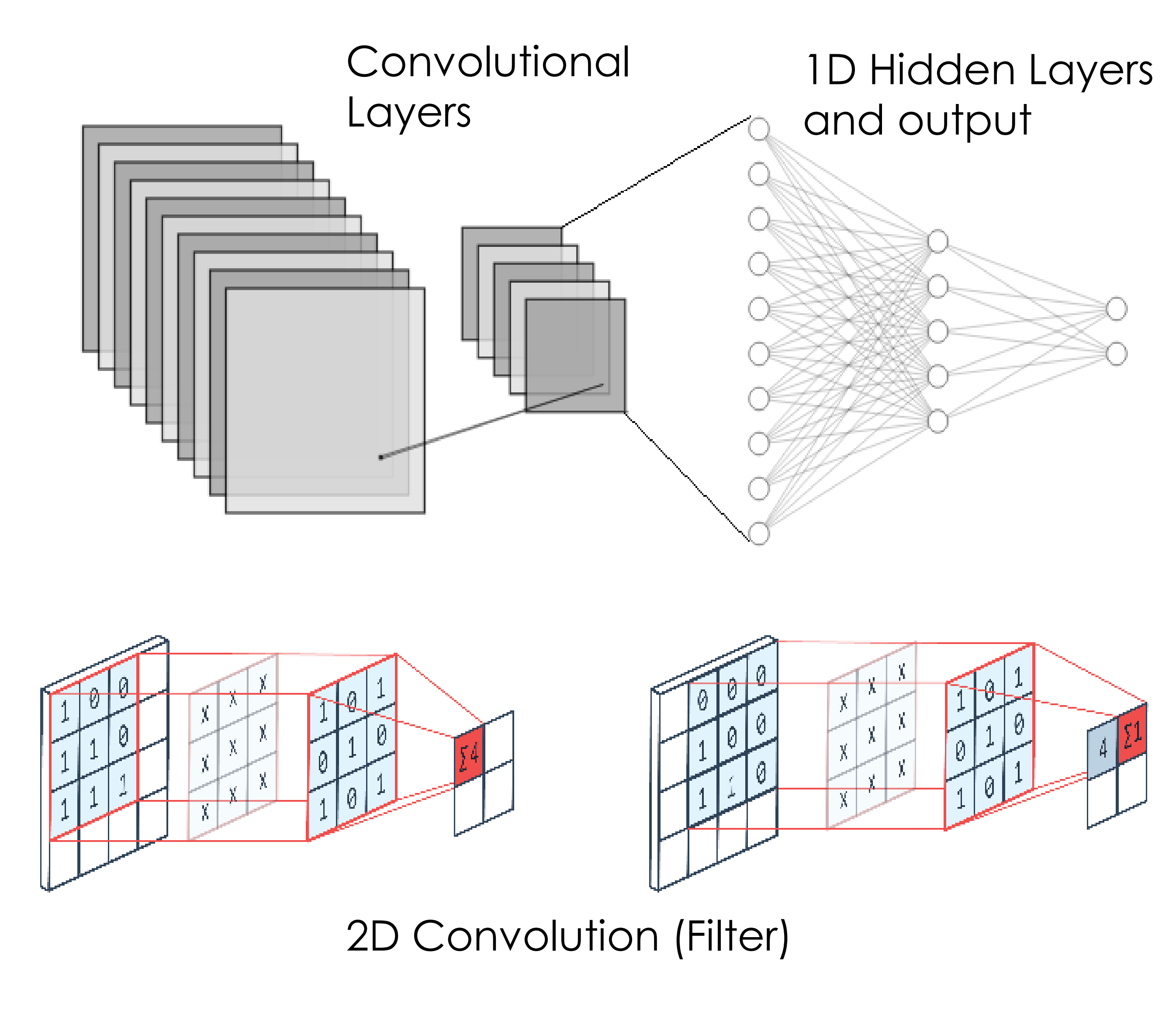
Training Sample

- Positive Sample:
  - Any DC tracks in the same sector as an electron.
  - Associate any hits in ECAL in same sector as the electrons.
- Negative Sample:
  - Any track and hits in a sector with no reconstructed electrons.
- The neural network was trained and tested only on inbending data. We need to expand this to both field configurations.
- Training is done in python with tensorflow.



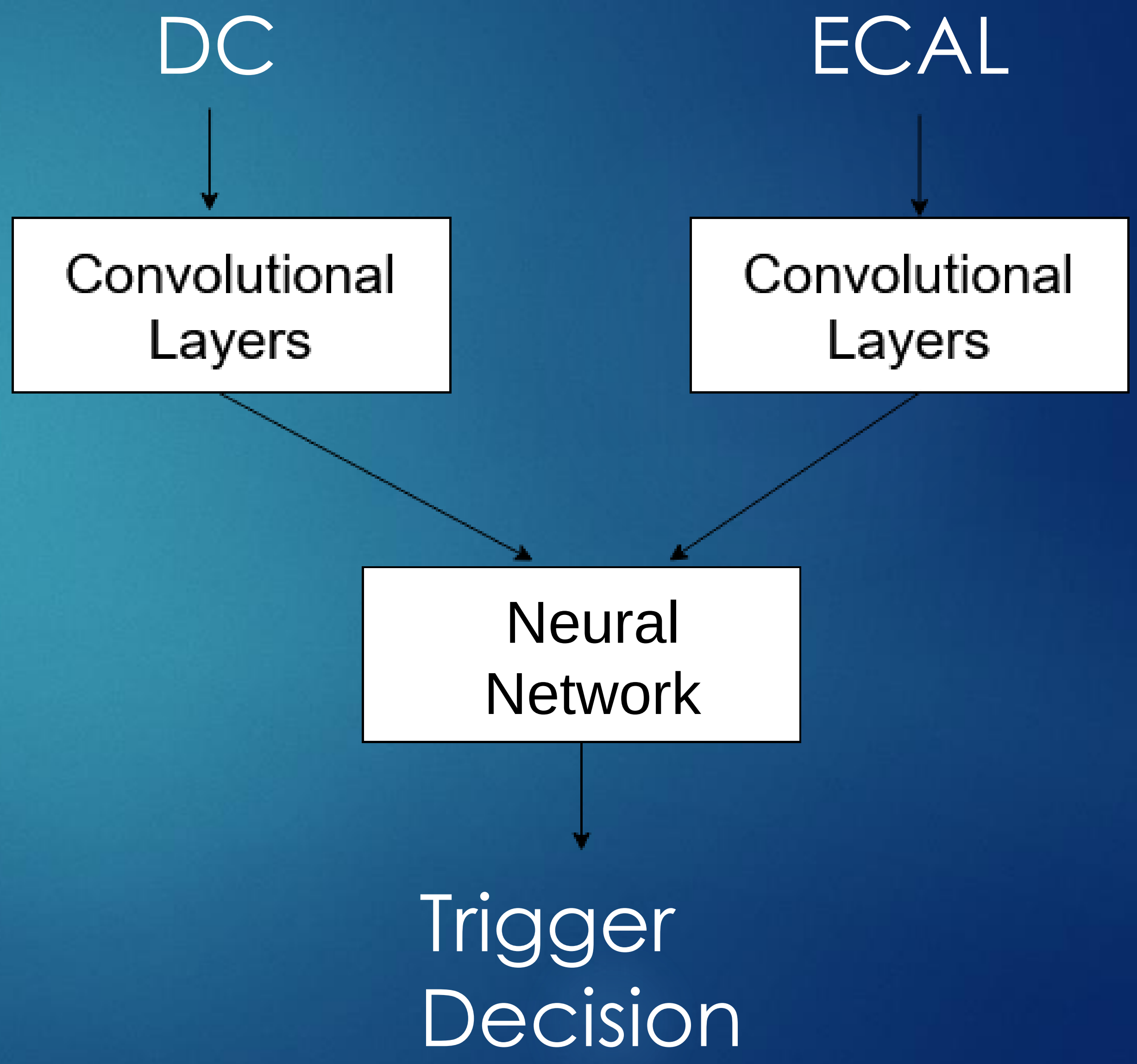
Convolutional Neural Networks (CNNs)

- The convolutional layers apply a given number of filters to an image. These filters enhance or remove certain features.
- The resulting image is flattened and passed to a neural network for classification.
- As it trains the classifier will learn what filters to apply, along with optimising the neural network weights.
- We can tune certain parameters of the convolutional and hidden layers to improve our classifier.



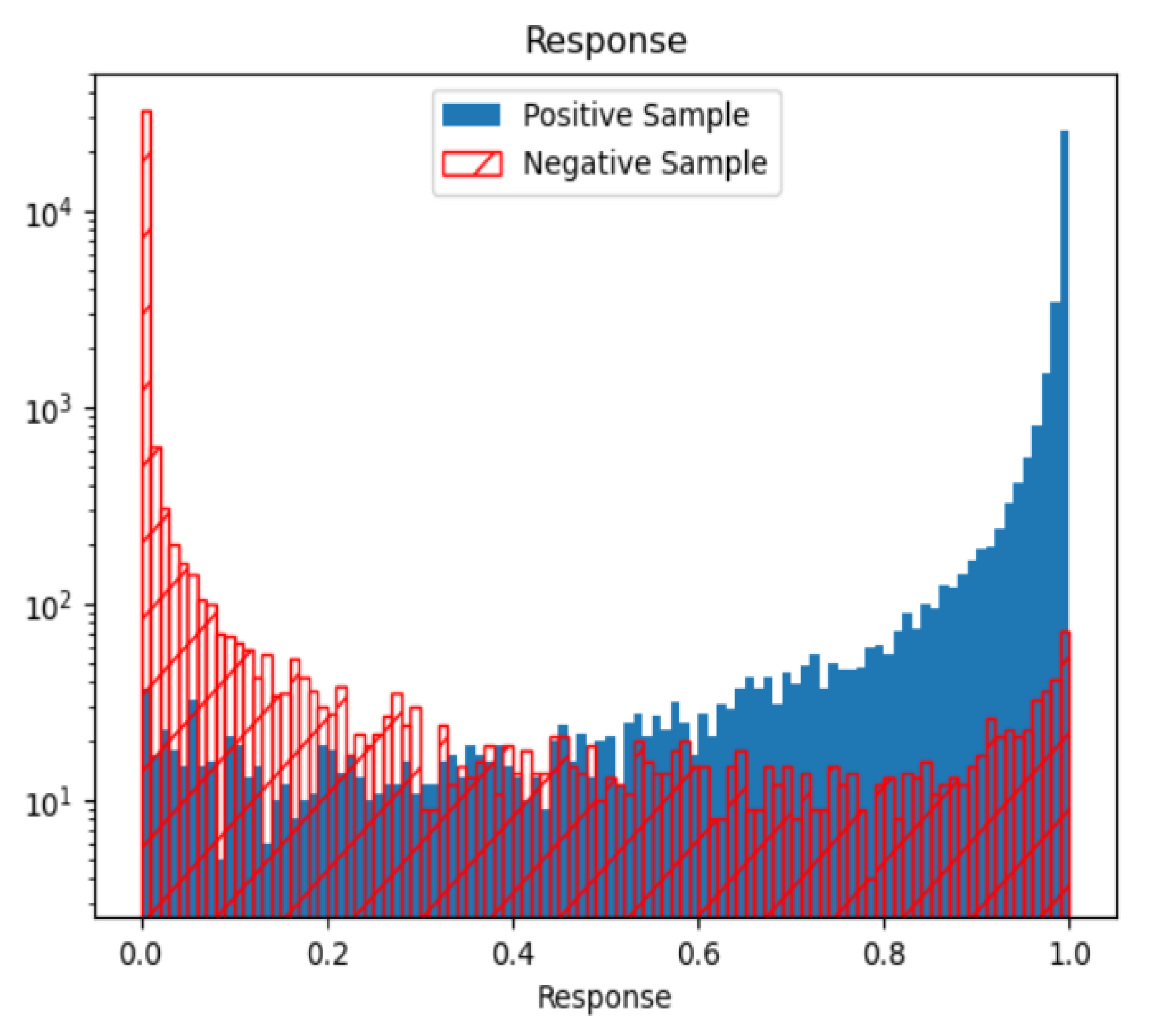
Network Architecture

- DC and ECAL images are passed separately to the neural network.
- The convolutional layers apply several filters to the DC and ECAL images to select the important features.
- The hidden layers then take their concatenated output and classes it as a trigger or no trigger event.



Response

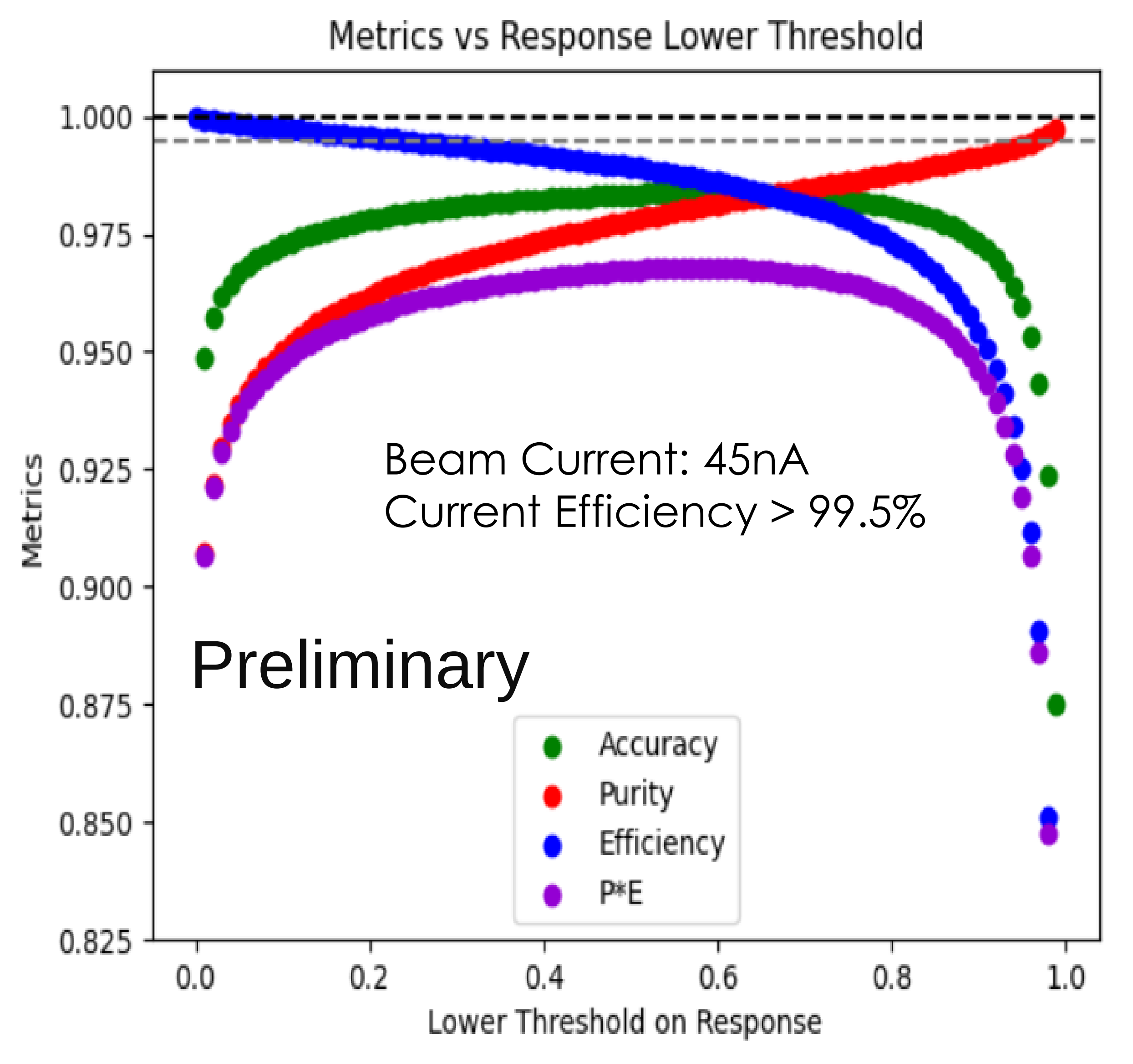
- The output of the neural network is given as a probability of being from the positive sample. We call this probability the response.
- A perfect classifier would assign a response of 1 to all positive events and a response of 0 to all negative events.
- The neural network effectively reduces the trigger decision down to a cut on the response.



Metrics and Response

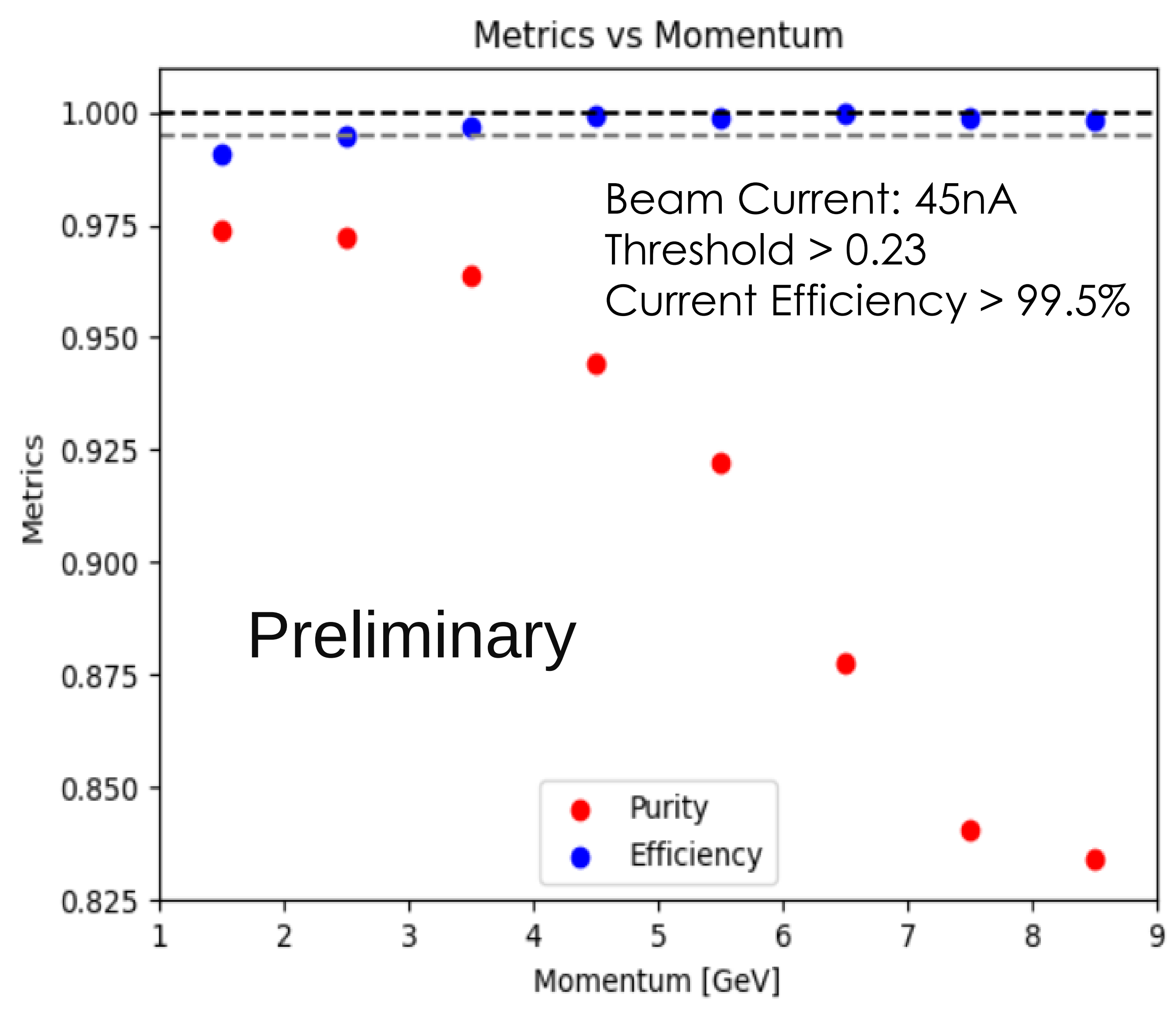
- The Accuracy measures the proportion of correctly classified events.
- The Efficiency measures the proportion of correctly classified positive events.
- The Purity measures the proportion of positive events in all trigger events.

Threshold	Purity	Efficiency	Accuracy
0.0012	0.841	0.9999	0.906
0.03	0.930	0.999	0.962
0.23	0.964	0.995	0.979
0.47	0.977	0.99	0.983



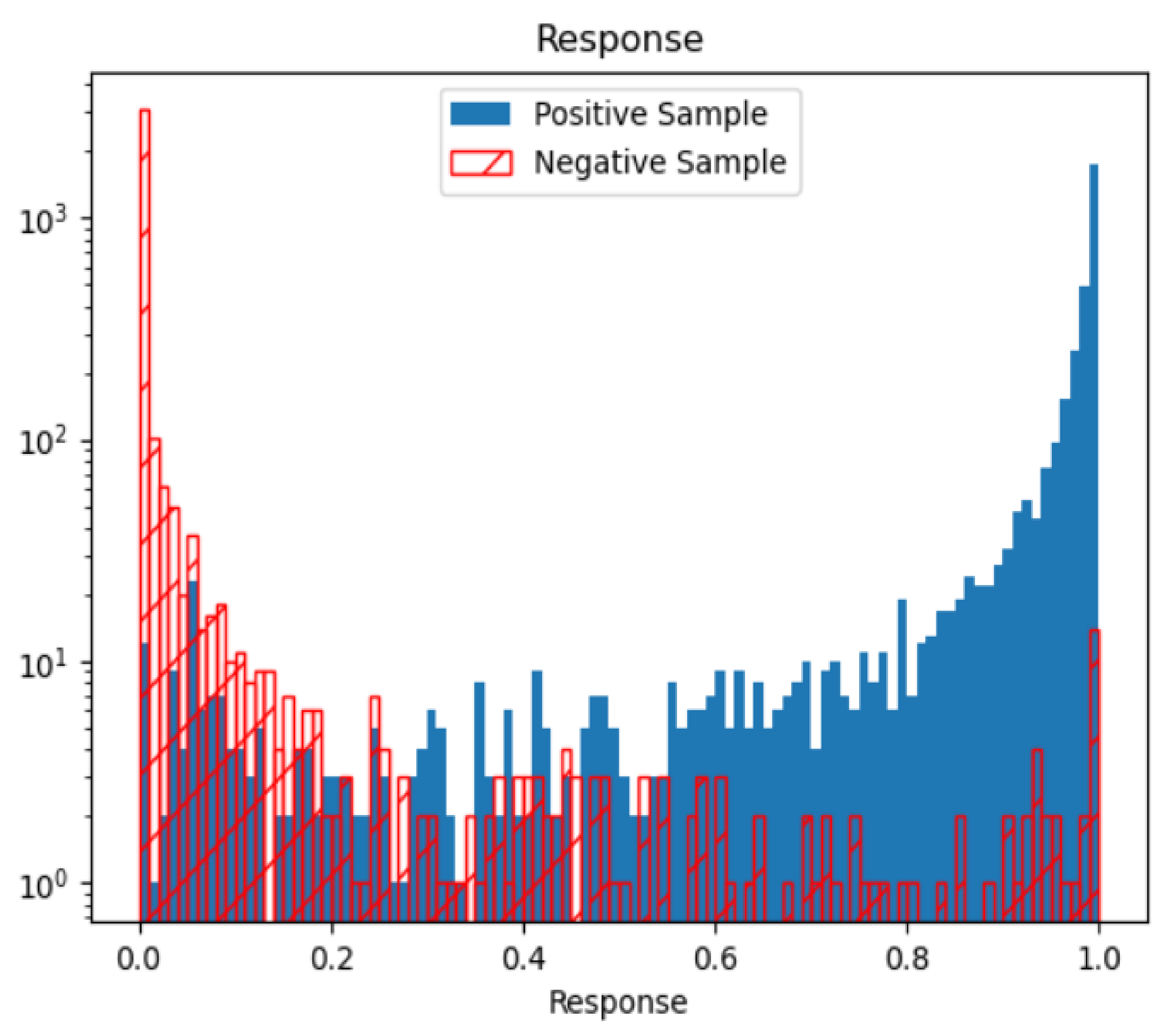
Momentum Dependence

- Sorted test data into 1 GeV bins in momentum.
- Higher momentum electrons are well identified by the neural network, and the efficiency increases with momentum.
- Higher momentum negative events are more often misclassified, so the purity decreases with momentum.
- These studies need more statistics.



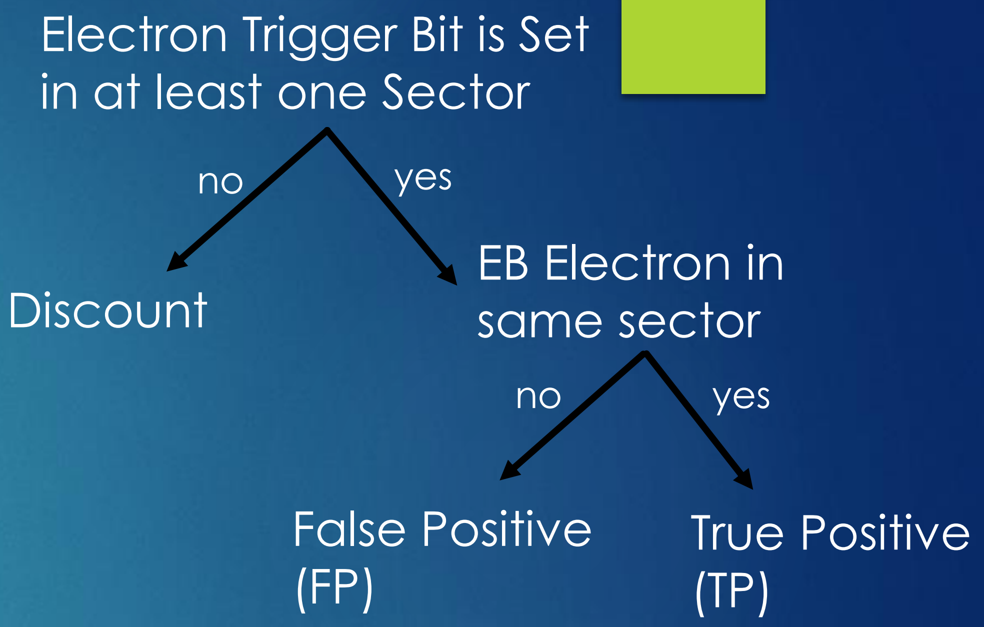
5 Superlayer Tracks

- The neural network can distinguish between positive and negative sample events even when these only have 5 superlayer segments.
- More in-depth studies on the effect of track quality on the response are on their way.



Current Trigger Purity

- The electron trigger requires geometrically matched:
  - HTCC clusters with at least two photoelectrons
  - PCAL clusters with high energy deposition in all calorimeters
  - DC track with at least 5 superlayer segments (3 layers in each superlayer)
- The current trigger efficiency is above 99.5%.
- At higher luminosities, higher occupancy means the trigger purity decreases.
- At 45 nA, for a 99.5% efficiency the AI Trigger purity is 96.4% purity ie ~70% data reduction.



Purity = TP/(TP+FP)

Beam Current	Purity TP/(TP+FP)
5 nA	0.43
40 nA	0.28
45 nA	0.29
50 nA	0.27
55 nA	0.23

Performance

- RG-A ran at a rate of 11-12*10^3 event/s, and there are plans to run at 20 kHz in the near future. The neural network needs to perform at a similar or higher prediction rate.
- 6 inferences per event, one for each sector. These can be parallelised as the trigger will be called on batches of 100 events (record).
- These estimates are bound to change once deployed, with an expected increase in performance on GPU with better specs (RTX 3090).

Processor	Prediction Rate inference/s	Prediction Rate events/s
CPU	17*10^3 inference/s	3 kHz
GPU	84*10^3 inference/s	14 kHz

The prediction rates were measured using tensorflow with the following specs:

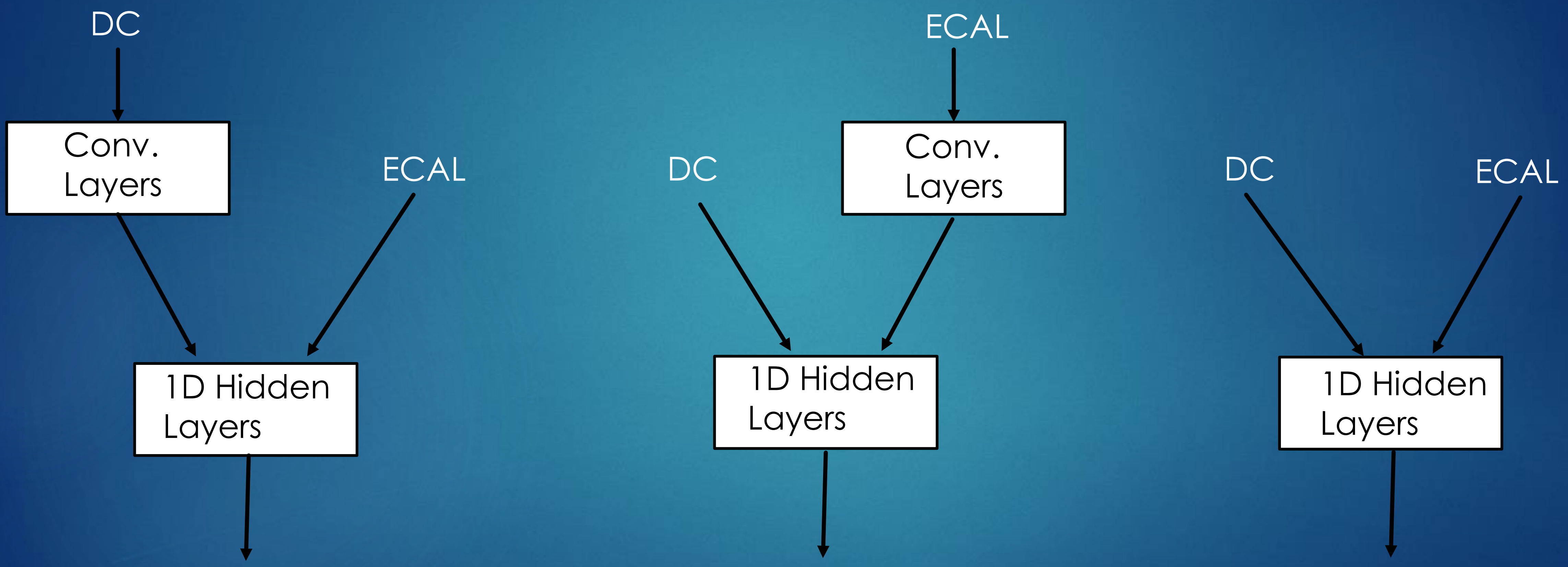
- CPU: 48 Xeon cores and 128G RAM.
- GPU: Nvidia GeForce RTX 2080 Ti graphics card with 11 GB GDDR6 RAM and 4352 CUDA cores.

Conclusion and Next Steps

- Preliminary results suggest the AI trigger can improve on the current electron trigger with a ~70% data reduction.
- We will further test with increased statistics how this varies with track quality, luminosity and both in and out bending data.
- The neural network's estimated 14 kHz (GPU) event rate matches that of past data taking.
- This will change once implemented in the CLARA/COATJAVA workflow and will be tested thoroughly.
- An analysis note is on the way.

Backup Slides

Network Architectures and Prediction Rates



Pred. Rate: 15 $*10^3$ event/s

Pred. Rate: 16 $*10^3$ event/s

Pred. Rate: 18 $*10^3$ event/s

Momentum Dependance of Response