Point Cloud Machine Learning for Cell-to-Track Association: Enhancing Event Reconstruction in High Energy Physics

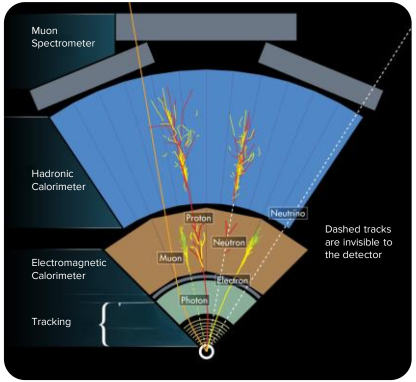
Luca Clissa, Joshua Himmens, Maximilian Swiatlowski, Iacopo Vivarelli
on behalf of the ATLAS collaboration

luca.clissa2@unibo.it

EUROPEAN AI FOR FUNDAMENTAL PHYSICS CONFERENCE EuCAIFCon 2025

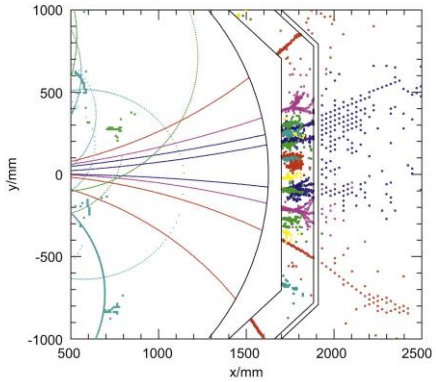
Particle flow in ATLAS

- **Problem:** particle identification & energy calibration
- Particularly challenging when we have jets/showers
- Key: exploit complementary components info:
  - tracker
  - calorimeters (calo)
- Particle flow (p-flow) algorithms reconstruct particle's trajectory and its energy deposit in detector components
- Inputs are tracks in the inner detector and topo-clusters in calorimeter
  - topo-clusters are groups of neighbouring cells
    ➡ useful to reconstruct showers in the calorimeter
- **Goal:** try to associate topo-clusters to tracks

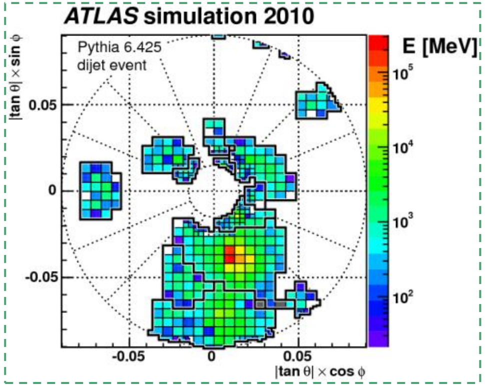


[ATLAS-OUTREACH-2021-052]

ILD experiment



[Nucl.Instrum.Meth.A611:25-40,2009]

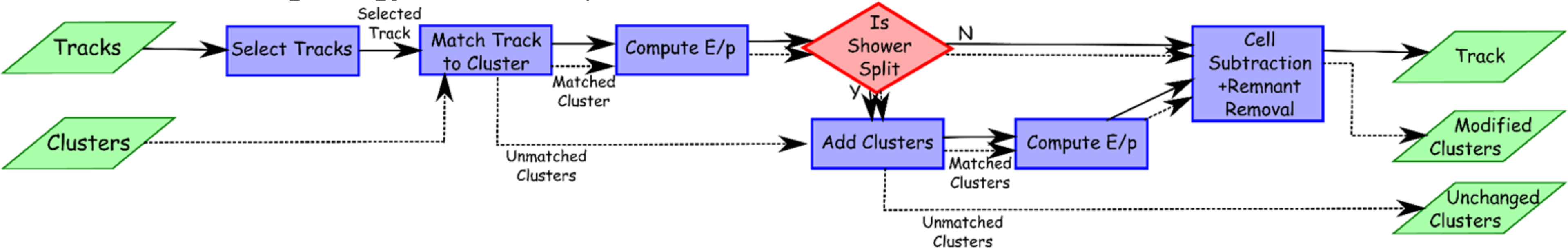


[Eur. Phys. J. C 77 (2017) 490]

ATLAS p-flow algorithm [Eur. Phys. J. C 77 (2017) 466]

For each track in descending pT:

1. associate closest topo-cluster based on angular distance $\Delta R' = \sqrt{\left(\frac{\Delta\phi}{\sigma_\phi}\right)^2 + \left(\frac{\Delta\eta}{\sigma_\eta}\right)^2}$
2. compute expected energy deposit based on the topo-cluster position and track momentum
3. if expected and measured energies differ significantly, associate more topo-clusters
4. subtract the expected energy by calo cells
5. if remaining energy lies within expected fluctuations, remove the remnants



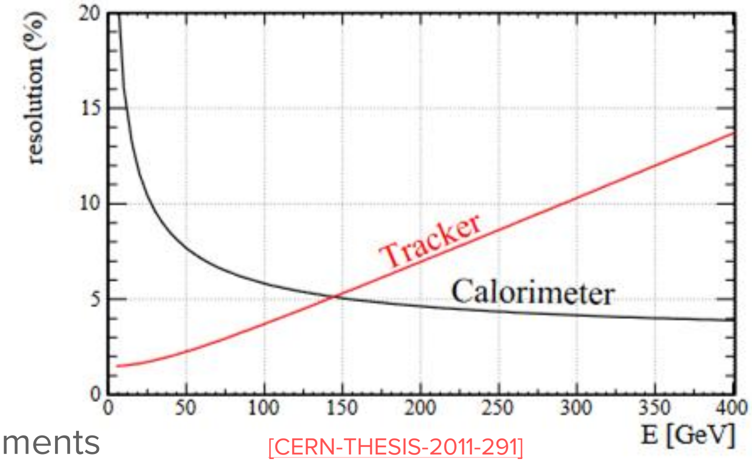
ATLAS p-flow algorithm: pros and cons

Existing ATLAS p-flow algorithm **strengths**:

- Calo + track information:
  ➜ improve energy resolution at low energy
- Good energy and angular resolution
- Pileup mitigation through “charged hadron subtraction”

Main **limitations**:

- Associate track to topo-clusters, not cells directly
  ➜ energy subtraction limited to fixed cluster boundaries
- No calibration currently available ➜ only detector measurements
- Tracker usage off above 100 GeV to avoid false matches

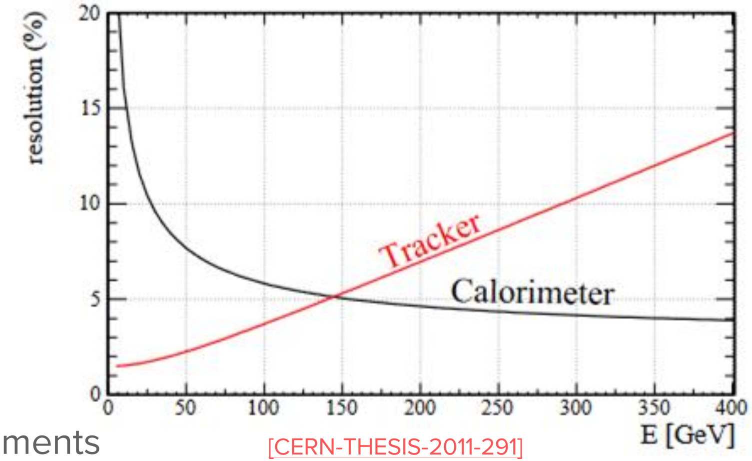


[CERN-THESIS-2011-291]

ATLAS p-flow algorithm: pros and cons

Existing ATLAS p-flow algorithm **strengths**:

- Calo + track information:
  ➡ improve energy resolution at low energy
- Good energy and angular resolution
- Pileup mitigation through “charged hadron subtraction”



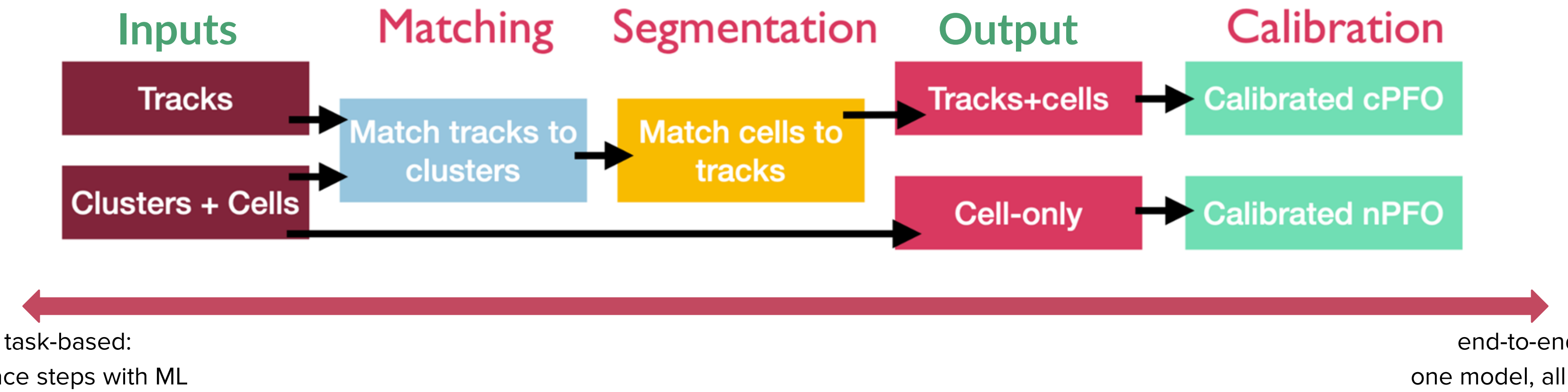
[CERN-THESIS-2011-291]

Main **limitations**:

- Associate track to topo-clusters, not cells directly
  ➡ energy subtraction limited to fixed cluster boundaries
- No calibration currently available ➡ only detector measurements
- Tracker usage off above 100 GeV to avoid false matches

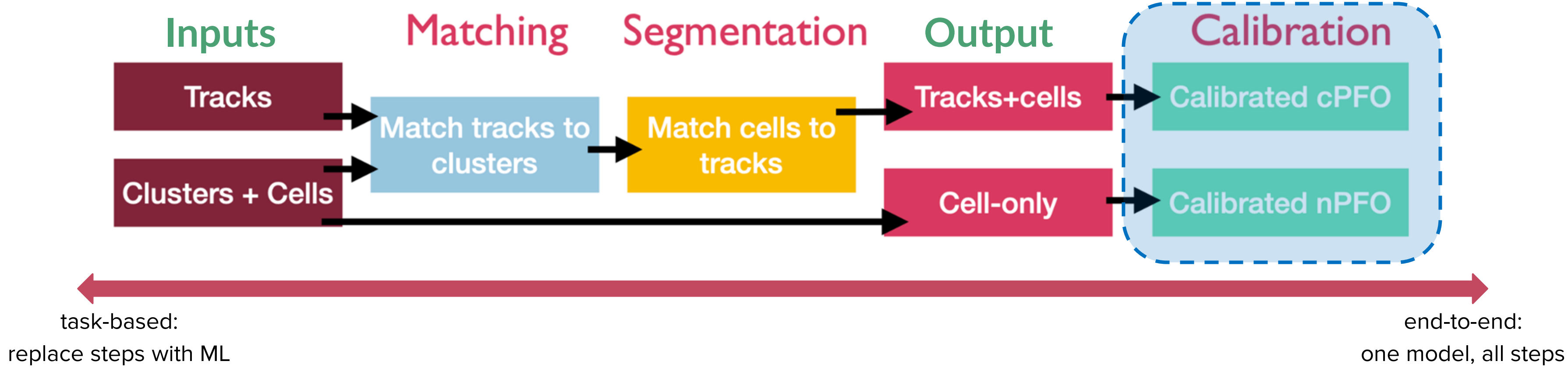
Can we do better? Maybe Machine Learning (ML) can help?

Machine Learning alternatives



- Machine Learning models have already shown promising results under various settings
  - HyperGraphs for end-to-end pflow [Eur. Phys. J. C 83 (2023) 596]
  - ongoing work on task-based solutions (matching, segmentation and calibration)
  - **image-based methods for calibration** [ATL-PHYS-PUB-2020-018] (central barrel reconstruction, $|\eta|<0.7$)

Machine Learning alternatives



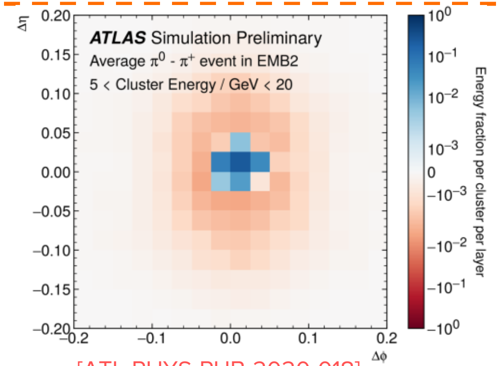
- Machine Learning models have already shown promising results under various settings
  - HyperGraphs for end-to-end pflow [Eur. Phys. J. C 83 (2023) 596]
  - ongoing work on task-based solutions (matching, segmentation and calibration)
  - **image-based methods for calibration** [ATL-PHYS-PUB-2020-018] (central barrel reconstruction, $|\eta|<0.7$)
    - ➔ **Outperform Local Hadronic Cell Weighting** (LCW) calibration
    - ➔ Work well for both identification and energy calibration
    - ➔ However, **inefficient representation** and **do not include tracking data**

Point cloud ML for p-flow [ATL-PHYS-PUB-2022-040, ATL-SOFT-PROC-2025-018]

- Focus on **pion identification and energy calibration**,
  - first step towards *hadronic shower reconstruction*
- Leverage **point cloud data**
  - only use actual hits, i.e. natural zero suppression
  - naturally handle varying granularity
  - naturally allow including tracking data
  - easily extend to including more information (momentum, hit confidence, …)
- Test 4 Deep Learning methods for point cloud data:
  - Graph Neural Network (GNN)
  - Deep Sets, Transformers, Merged Deep Fully Connected Network (DNN)
- Outline of extension to segmentation task

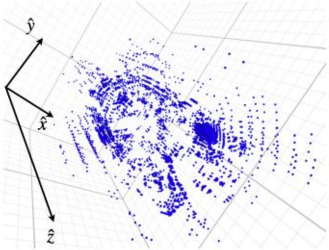
Why point cloud data?

- Image-based approaches are sub-optimal
  - different spatial granularity is difficult to render
  - only encode calorimeter information (**no tracker**)
  - irregular deposition geometries cause sparse images
    ➜ **inefficient representation**



[ATL-PHYS-PUB-2020-018]

- Point cloud representation has several advantages
  - represent hits as 3D points with properties
    ➜ complex 3D shapes instead of series of images
    ➜ features like energy, hit confidence
  - including tracker is straightforward
  - only uses actual hits
    ➜ efficient representation



ATL-PHYS-PUB-2022-040

Dataset

- **Hadronic showers** originate primarily from pions
  - π^0: decay promptly to photons ➔ EM calo
  - $\pi^{+/-}$: more fluctuation in energy deposit patterns ➔ EM + hadronic calorimeter
- Full ATLAS simulation using Geant4
- Uniform pion distributions in
  - azimuthal angle
  - pseudo-rapidity
  - log true energy
- 10M π^0, 5M π^+, 5M π^-
  - 3.5M training, 500k validation, 1M test after quality cuts
  - events with exactly 1 track

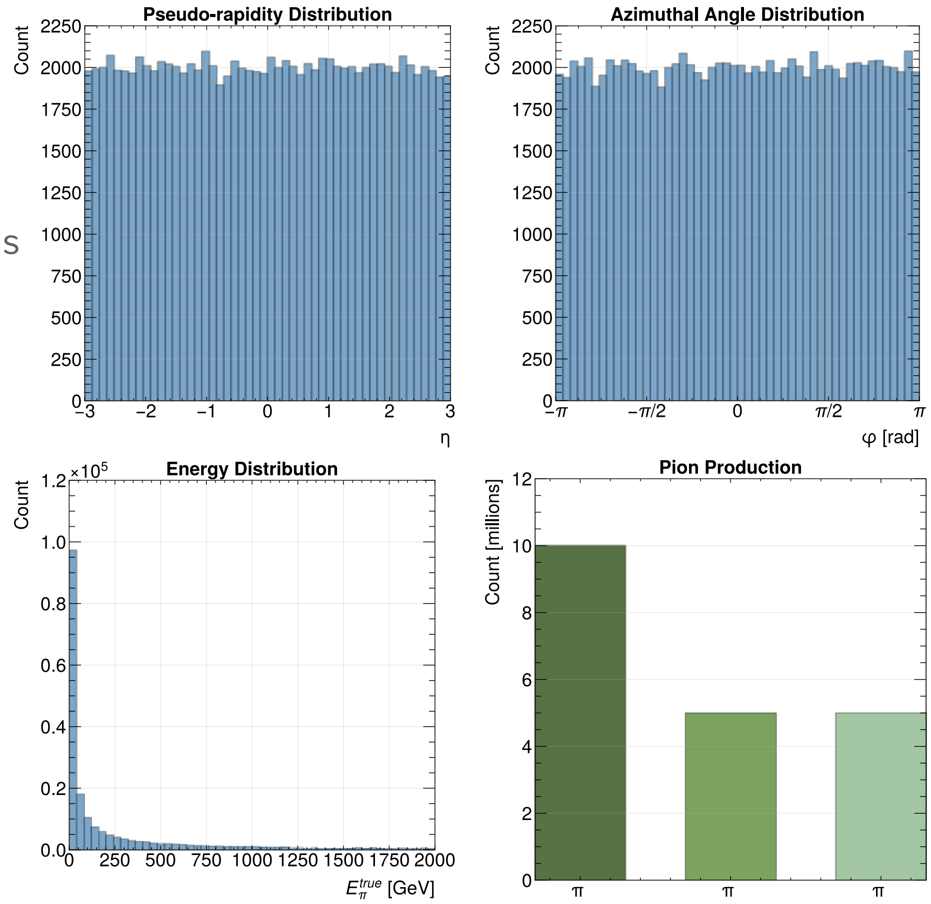
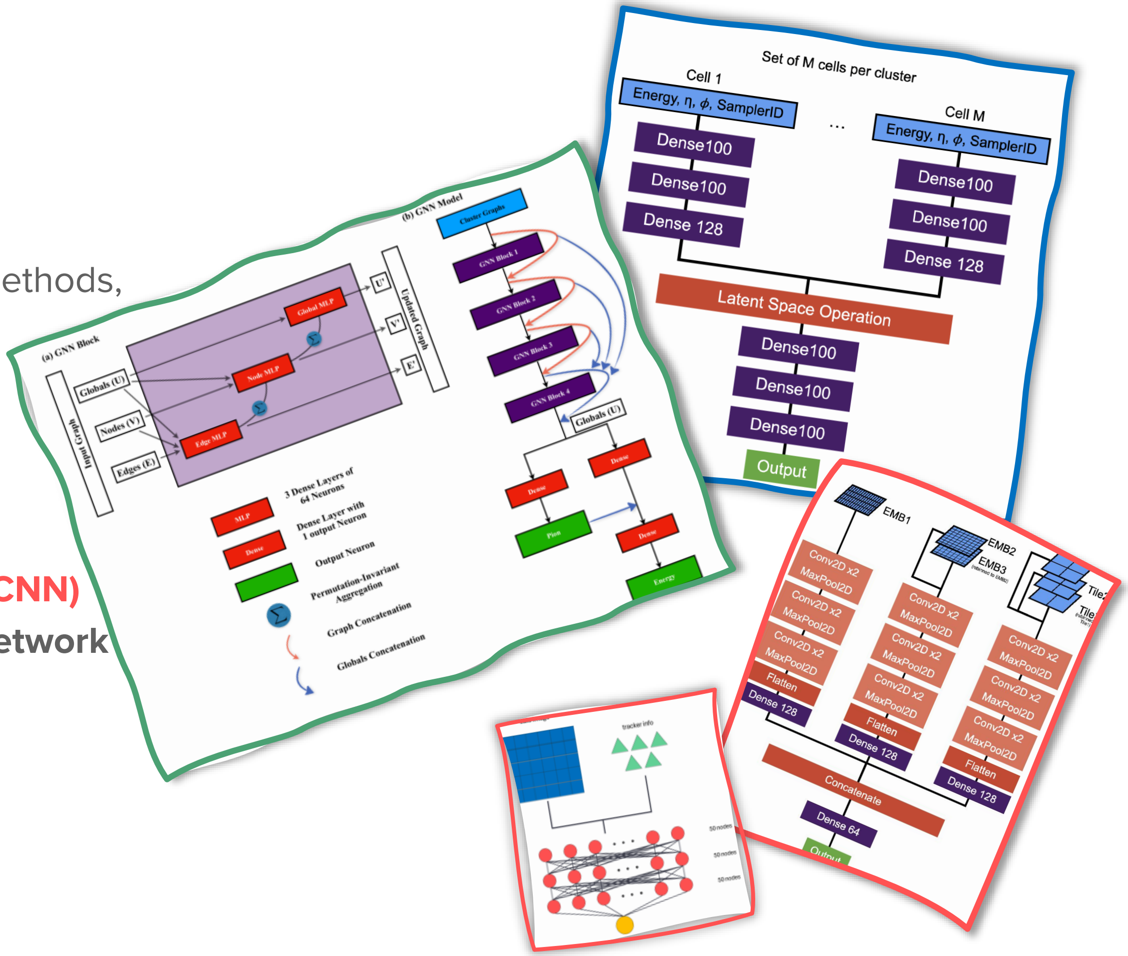


Illustration using non-official data (all plots)

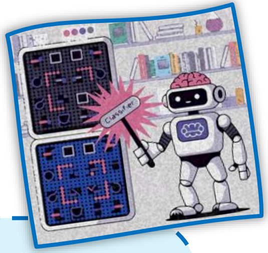
Deep Learning methods

We explored several Deep Learning methods, only some of them shown here:

- **Graph Neural Networks (GNN)**
- **Deep Sets**
- Transformers
- **Convolutional Neural Networks (CNN)**
- **Merged Deep Fully Connected Network (DNN)**
  ➔ image-based approaches

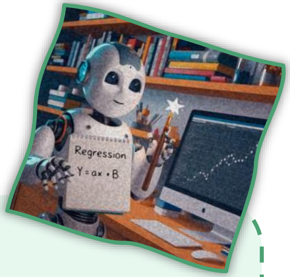


Learning tasks



Particle identification ➔ **classification: π^0 VS π^+/π^-**

- only calorimeter information
  ➔ adding tracks makes classification obvious
- input: one topo-cluster at a time



Energy calibration ➔ **regression: calibrated energy**

- **only calorimeter** information
- input: one topo-cluster at a time

- **calorimeter + tracker**
- input: one track + topo-clusters in ΔR<1.2

Results

We compare ML approaches against two baselines depending on the learning task:

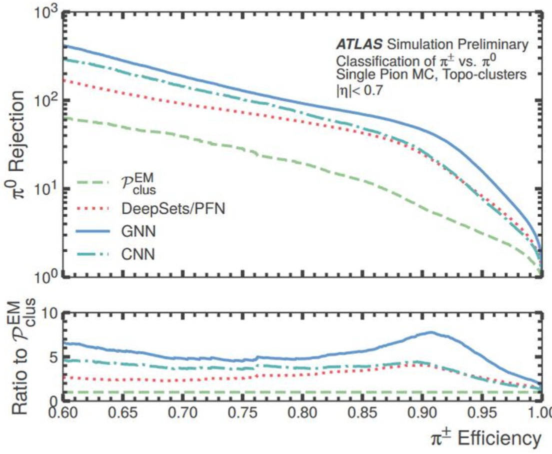
- classification
  → Electromagnetic (**EM**) scale + initial hadronic calibration step corrections: $\mathcal{P}^{\mathrm{EM}}_{\mathrm{cluster}}$
- regression
  → full Local Cell Weighting (**LCW**) calibration, i.e. $\mathcal{P}^{\mathrm{EM}}_{\mathrm{cluster}}$ + additional corrections: $E^{\mathrm{LCW}}_{\mathrm{cluster}}$

π^0 VS π^+/π^- classification: calo only

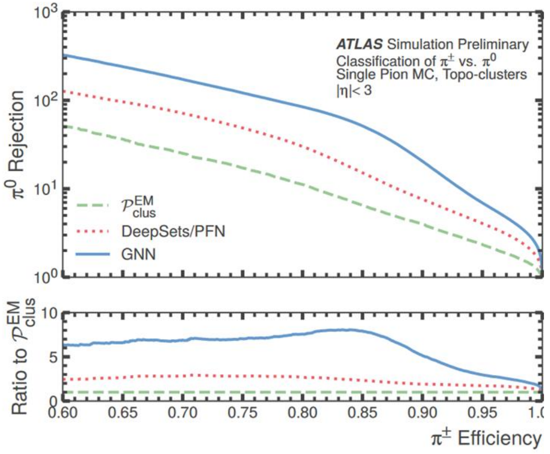
[ATL-PHYS-PUB-2022-040]

Metrics: $\pi^{\pm}$ efficiency VS π^0 rejection
- $\pi^{\pm}_{eff}$ = TP/(TP+FN)
- π^0_{rej} = 1 - FPR = TN/(TN+FP)

- ML methods outperform baseline $\mathcal{P}_{clus}^{EM}$
  - 4x to 8x background rejection in |η|<0.7
  - 2x to 6x in full η range
- **GNN performs best**
  ➜ 5x background rejection
- performance increases with higher topo-cluster energy



(a) $|\eta| < 0.7$



(b) $|\eta| < 3$

Model	Rej. @ 90% Eff. for $|\eta| < 0.7$	Rej. @ 90% Eff. for $|\eta| < 3$
CNN	26.584	-
GNN	46.419	20.500
Deep Sets	24.814	7.608
$\mathcal{P}^{EM}_{clus}$	6.123	3.977

[ATL-PHYS-PUB-2022-040]

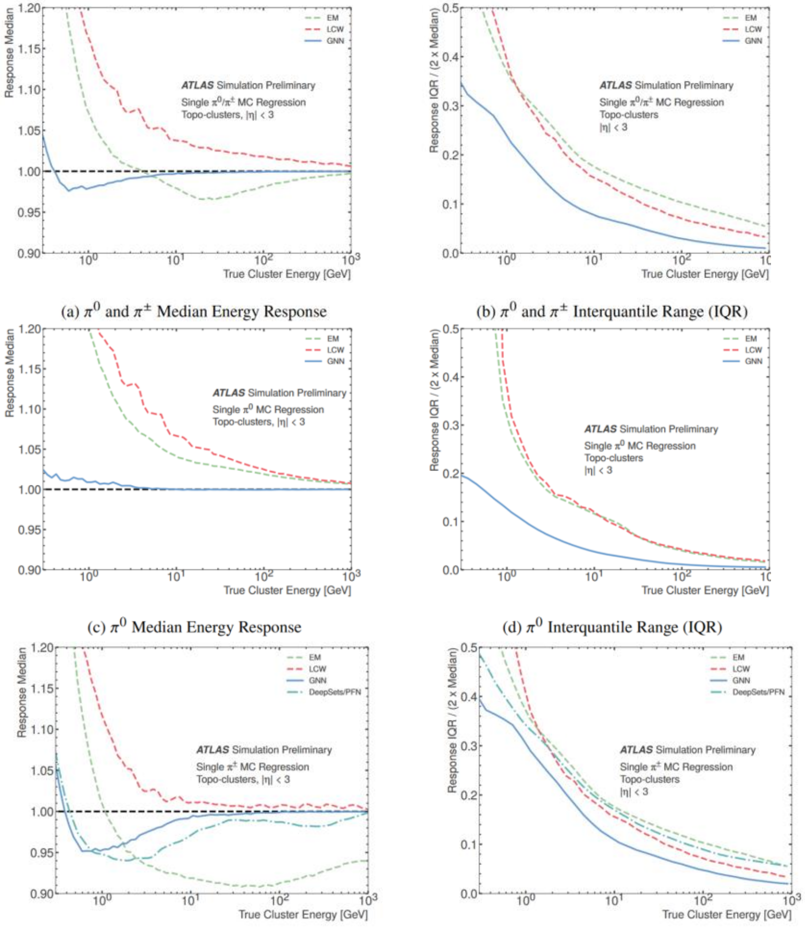
Energy regression: calo only

Metrics: median energy response and resolution

- energy response, $R = E_{pred}/E_{true}$
- resolution, IQR = median R $\pm$ 1σ (16-84%)

- ML significantly better than traditional calibrations across entire energy spectrum
  ➔ R closer to 1; lower IQR

- **GNN is best overall**
- **Deep Sets better than baseline for charged pions, especially at low-energy** (< 1 GeV)
  ➔ known weakness in conventional techniques
- ML mitigates long-standing calibration issues
  - high-energy $\pi^{\pm}$ underestimation
  - low-energy π^{0} overestimation



(a) π^0 and $\pi^{\pm}$ Median Energy Response
(b) π^0 and $\pi^{\pm}$ Interquantile Range (IQR)
(c) π^0 Median Energy Response
(d) π^0 Interquantile Range (IQR)
(e) $\pi^{\pm}$ Median Energy Response
(f) $\pi^{\pm}$ Interquantile Range (IQR)

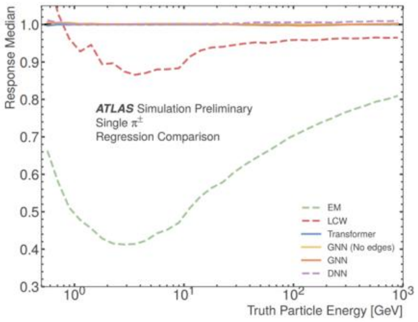
Energy regression: calo + tracker

[ATL-PHYS-PUB-2022-040]

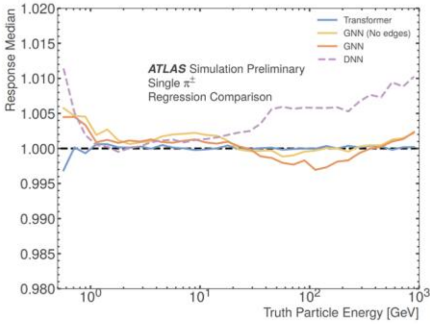
Metrics: median energy response and resolution

- energy response, $R = E_{pred}/E_{true}$
- resolution, IQR = median R $\pm$ 1σ (16-84%)

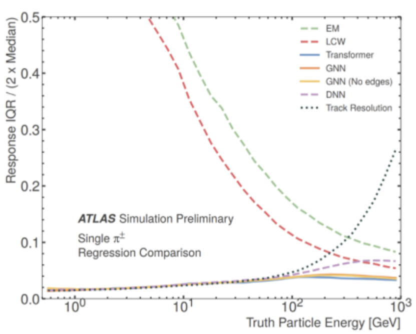
- Point cloud models VS baseline: significantly outperform EM and LCW calibration
  - better R and IQR across the full energy spectrum
- Point cloud VS image-based (DNN):
  - comparable median accuracy for E < 30 GeV
  - superior performance for E > 30 GeV
- Track information dramatically improves prediction
  ➔ IQR consistently below 0.1
  (VS 0.4 for cluster-only)
- Adding cell-level info further improves resolution, particularly at high energy (more in backup slides)



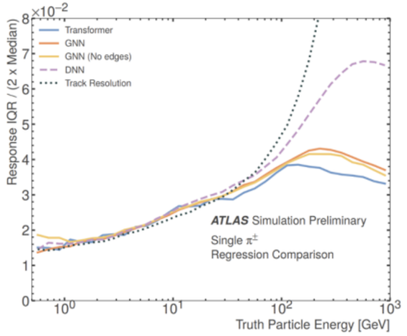
(a) With EM and LCW Baselines



(b) Without EM and LCW Baselines



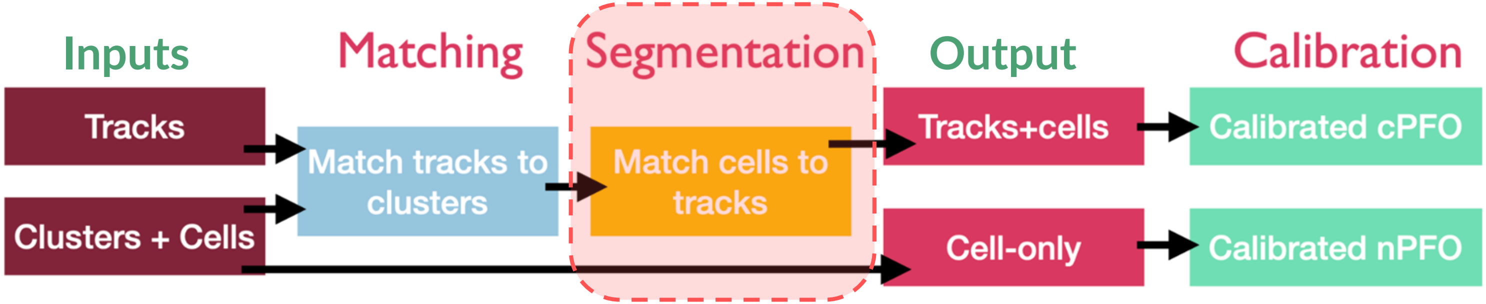
(a) With EM and LCW Baselines



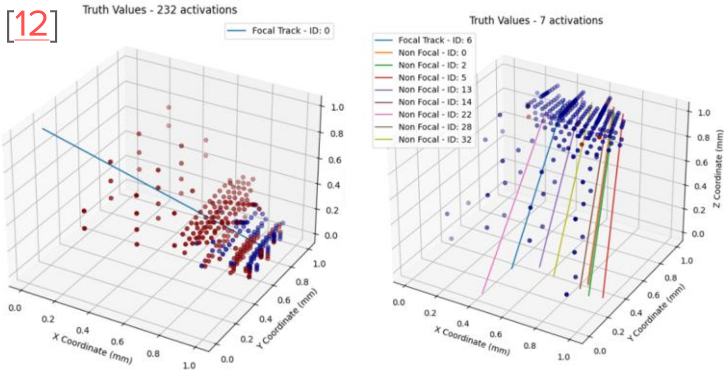
(b) Without EM and LCW Baselines

Next steps

Cells-to-track matching



- Extend point cloud methods to tackle cells-to-track matching [12]
  - one focus track at a time
  - all hits within ΔR=0.2 (tracker + calo) form point cloud (sample)*
  - associate hits with track contributing the most energy (>50%)
  - PointCloud architecture [6], attempt with MaskFormers [7]
- Promising results for simple ρ, Δ decays (~1 track per event)
- Trying to generalize to more challenging dijets scenarios



*technically, we need to pad events with less hits to ensure point clouds with same dimensions

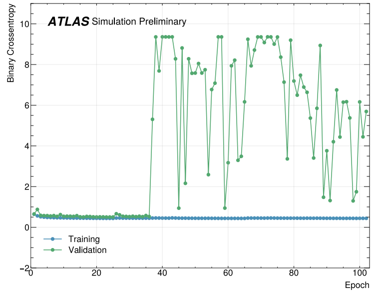
Illustration using non-official data (all plots)

Cells-to-track matching: lessons learned (cont'd)

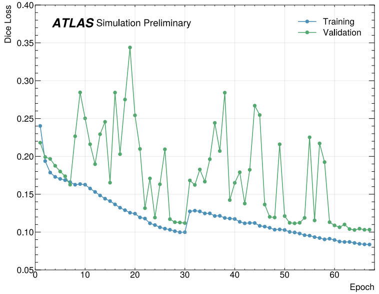
[ATL-COM-PHYS-2025-488]

- Complex task, many challenges:
  - Padding strategy affects results
  - Unstable training
  - Class imbalance

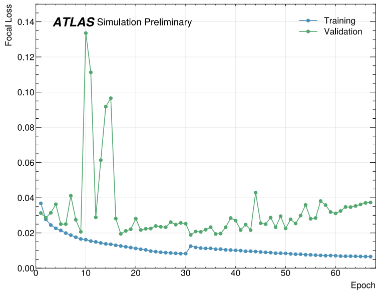
- Promising configs (more in backup):
  - Dice [10] and Focal [11] losses better than weighted BCE
  - Adam-W [12] produces better performance, also reducing instability
  - SGD [13] further stabilize training, but slower to converge
  - Cyclic learning rate [14] (with warm-up [15]) is key for convergence



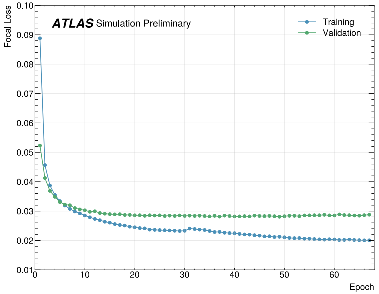
WBCE + Adam



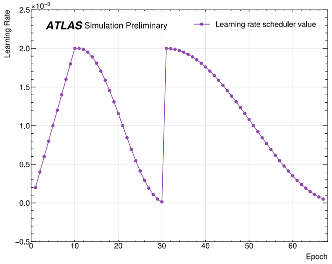
Dice + Adam-W



Focal + Adam-W



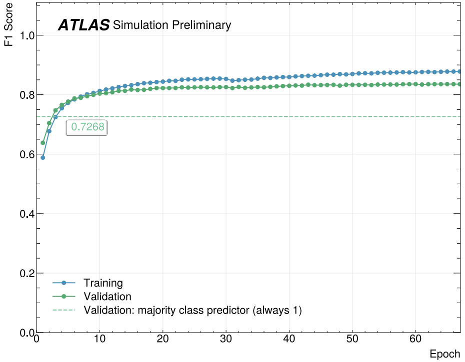
Focal + SGD

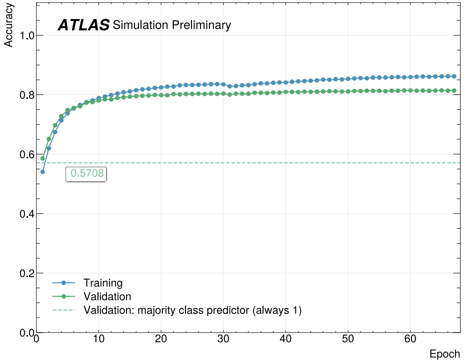


Cyclical LR w\ith warmup

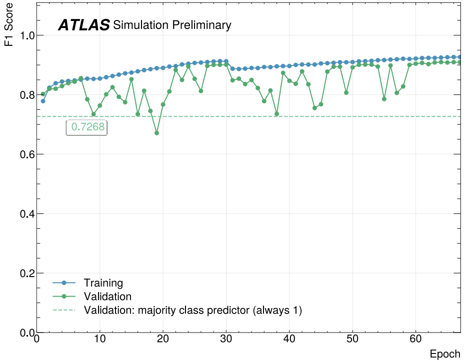
Cells-to-track matching: lessons learned (cont’d)

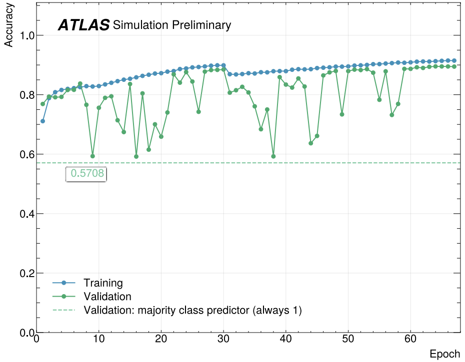
- Select loss/metrics wisely:
  - Masking is crucial
  - Accuracy typically misleading due class imbalance
    → F1 score more robust
  - Set meaningful baselines (e.g. trivial models for majority class)





Focal + SGD





Dice + Adam-W

[ATL-COM-PHYS-2025-488]

Conclusion

- Significant improvement in $\pi^0/\pi^\pm$ classification and energy regression
- Key findings from calorimeter-only regression:
  - GNN and Deep Sets outperform traditional calibrations across all energies
  - They mitigate long-standing calibration issues at the boundaries of energy values
  - **point cloud methods outperform image-based approaches**

    ➡ and **more efficient!**
- Combined calorimeter and tracker regression:
  - ML models surpass EM/LCW scales
  - Dramatic improvement in energy resolution (IQR/median < 0.1)
  - Pointcloud advantage increases at high energies (> 30 GeV)
  - Granular cell-level data further enhances results
- Outlook: promising step towards ML-optimized Particle Flow in ATLAS

References

[1] Aaboud, M., Aad, G., Abbott, B. et al. Jet reconstruction and performance using particle flow with the ATLAS Detector. Eur. Phys. J. C 77, 466 (2017). https://doi.org/10.1140/epjc/s10052-017-5031-2

[2] Di Bello, Francesco Armando, et al. "Reconstructing particles in jets using set transformer and hypergraph prediction networks." The European Physical Journal C 83.7 (2023): 596.

[3] Angerami, Aaron, and Piyush Karande. Deep Learning for Pion Identification and Energy Calibration with the ATLAS Detector. No. LLNL-JRNL-813169; ATL-PHYS-PUB-2020-018. Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2020.

[4] ATLAS collaboration. Point Cloud Deep Learning Methods for Pion Reconstruction in the ATLAS Experiment. ATL-PHYS-PUB-2022-040, CERN, Geneva, 2022.

[5] Thomson, M. A. "Particle flow calorimetry and the Pandora PFA algorithm." Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 611.1 (2009): 25-40.

[6] Qi, Charles R., et al. "Pointnet: Deep learning on point sets for 3d classification and segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[7] Van Stroud, Samuel, et al. "Vertex Reconstruction with MaskFormers." arXiv preprint arXiv:2312.12272 (2023).

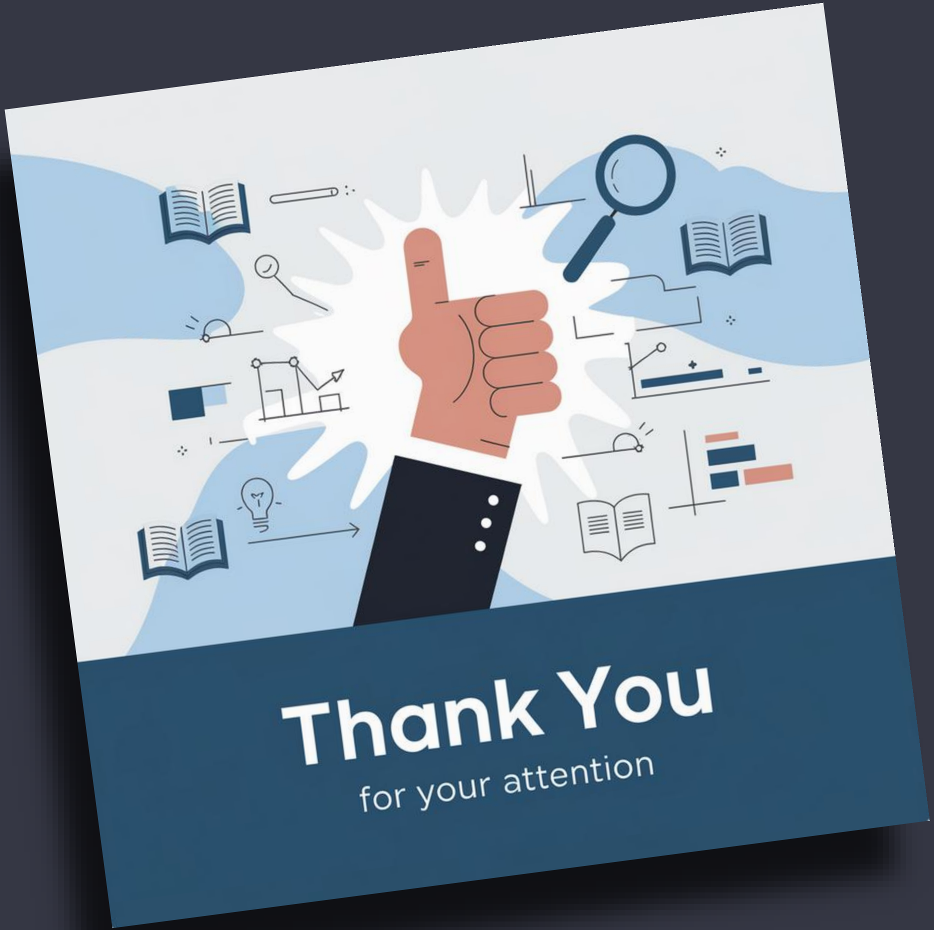
[8] Aad, Georges, et al. "Topological cell clustering in the ATLAS calorimeters and its performance in LHC Run 1." The European Physical Journal C 77.7 (2017): 1-73.

[9] Fleischmann, Sebastian. "Tau lepton reconstruction with energy flow and the search for R-parity violating supersymmetry at the ATLAS experiment." (2012).

[10] F. Milletari, N. Navab and S. -A. Ahmadi, "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," *2016 Fourth International Conference on 3D Vision (3DV)*, Stanford, CA, USA, 2016, pp. 565-571, doi: 10.1109/3DV.2016.79.

[11] T. -Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 2999-3007, doi: 10.1109/ICCV.2017.324.

[12] Clissa Luca, "Towards Machine-Learning Particle Flow with the ATLAS Detector at the LHC," *Proceedings of 27th International Conference on Computing in High Energy & Nuclear Physics, Kraków, Poland, 19 - 25 Oct 2024.*



Any questions?

Backup

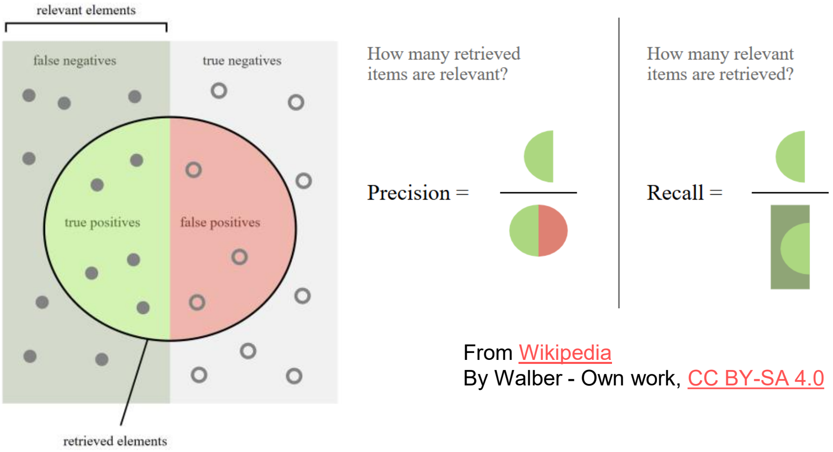
Loss and metrics for cell-to-track association

How do we measure performance?

- Use masking for selecting only **cell points**
- Loss and metrics are weighted by **energy**

Definitions:

- TP: true positives
- FP: false positives
- FN: false negatives



From Wikipedia
By Walber - Own work, CC BY-SA 4.0

Loss function (several attempts):

- Weighted Binary Cross-Entropy (wBCE)
- Weighted Focal loss
- Weighted Dice loss

Metrics

- Accuracy: (TP+TN) / (ALL)
- Precision (purity): P = TP / (TP + FP)
- Recall (signal efficiency): R = TP / (TP + FN)
- F1 score: 2 * P * R / (P + R)

Focal loss

- Slight variation of BCE:

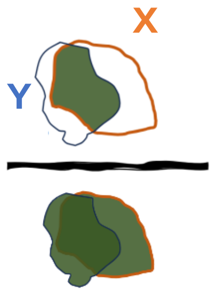
$$LOSS_{FOCAL}(\hat{p_t}) = \alpha_t (1 - p_t)^{\gamma} \ln{(p_t)}$$

where:

$$p_t = \begin{cases} \hat{p} & se\, y = 1 \\ 1 - \hat{p} & altrimenti \end{cases}$$ and $$\begin{cases} \alpha_t & peso\, classe\, t \\ \gamma & penalty \end{cases}$$

- Less weight to «easy data», more focus on difficult examples
- This mechanism helps mitigating issues with imbalanced datasets

Dice loss



- dice coefficient is a measure of “similarity”

$$DICE = \frac{2\,|X \cap Y|}{|X| + |Y|}$$

- dice loss

$$LOSS_{DICE}(y,\, \hat{p}) = 1 - \frac{2\, y \hat{p} + \epsilon}{y + \hat{p} + \epsilon}$$

0: perfect overlap

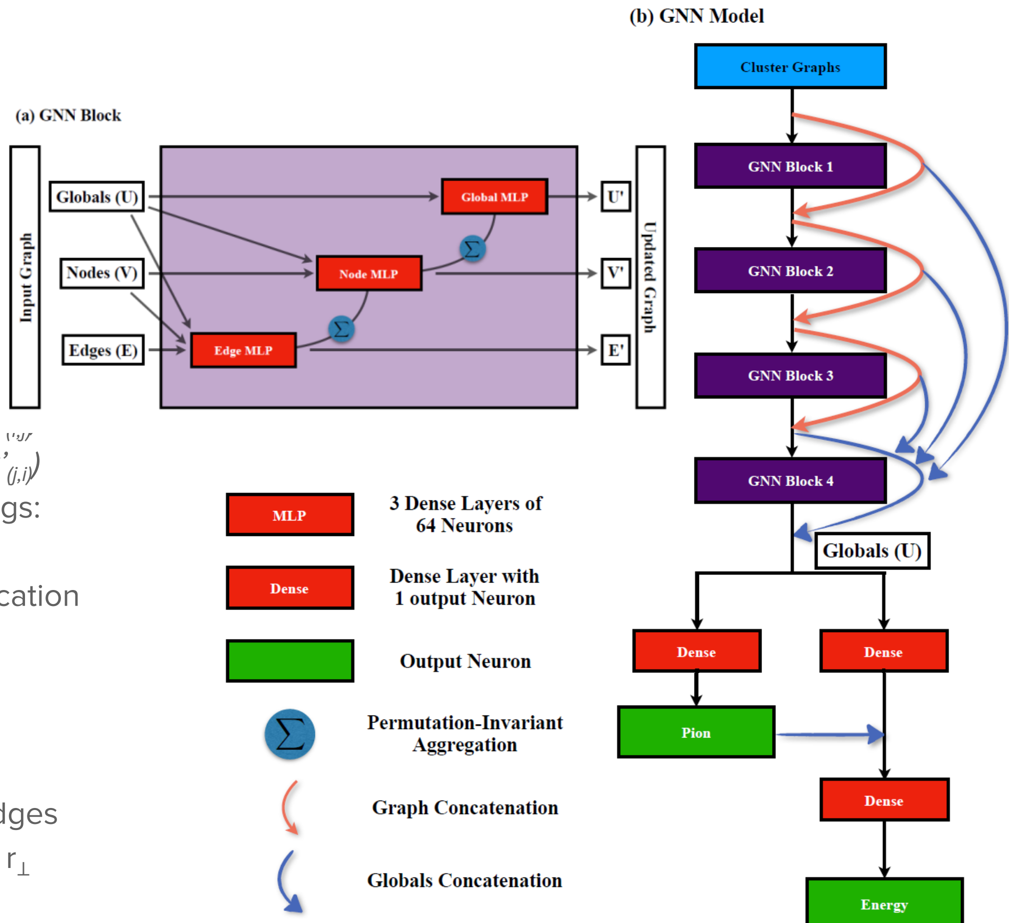
1: no overlap at all

- Specific for segmentation tasks

Graph Neural Network

Architecture

- 4 GNN blocks with Multi-Layer Perceptrons (MLP)
- Message passing to learn hidden representation
    - update edges: $x'_{(i,j)} = f_{edge}(x_i, x_j,$
    - update nodes: $x'_i = f_{node}(x_i, \Sigma_{j \in N_i} x'_{(j,i)})$
- Graph-level features as function of node embeddings:
  $g'_i = f_{global}(g, \Sigma_{i \in N} x'_j)$
- Global features concatenated with input for classification
- Simultaneous classification and regression tasks

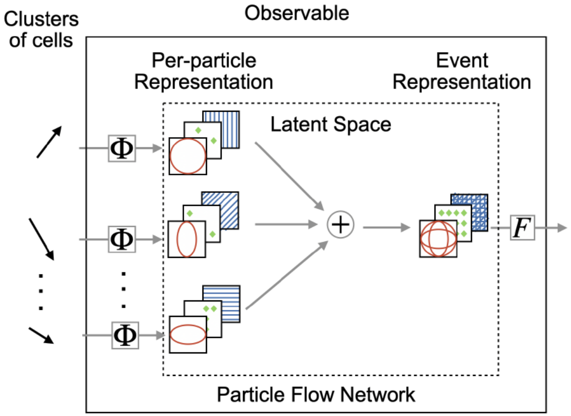


[ATL-PHYS-PUB-2022-040]

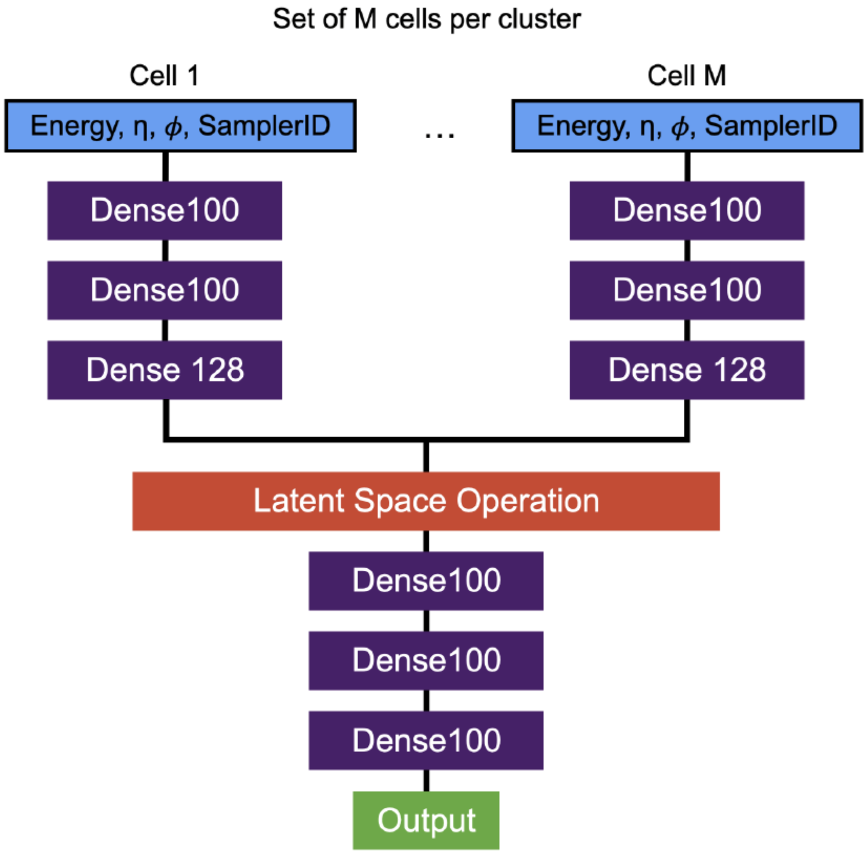
Components

- Cells are nodes, neighboring cells connected by edges
- Node features: energy sampling layer η, Δη, ϕ, $\Delta\phi$, $r_\perp$
- Edge features: type of connection

Deep Sets



Clusters of cells
Observable
Per-particle Representation
Event Representation
Latent Space
Φ
Φ
Φ
F
Particle Flow Network

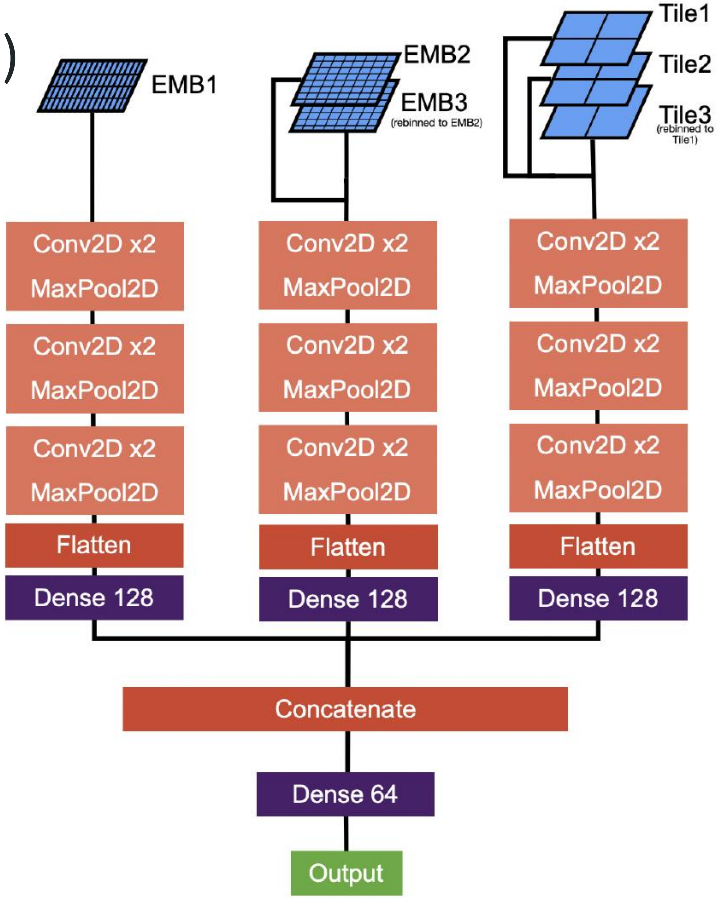


Set of M cells per cluster
Cell 1
Cell M
Energy, η, ϕ, SamplerID
...
Energy, η, ϕ, SamplerID
Dense100
Dense100
Dense 128
Dense100
Dense100
Dense 128
Latent Space Operation
Dense100
Dense100
Dense100
Output

Convolutional Neural Networks (CNN)

- pixels are bidimensional projections of cell baricenters
- pixel intensity reflects energy deposit
- considers calo layers separately to account for different granularity
    - EMB1 alone
    - EMB2, EMB3 together
    - Tile1, Tile2 and Tile3 together

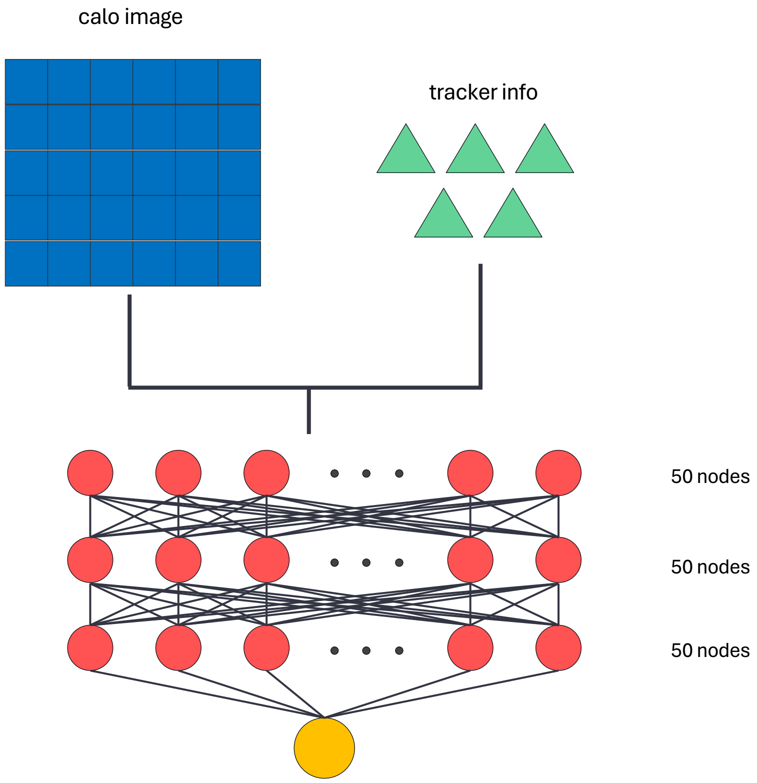
Calorimeter Layer	$(\Delta\eta, \Delta\phi)$ Granularity
EMB1	128×4
EMB2	16×16
EMB3	8×16
Tile1	4×4
Tile2	4×4
Tile3	2×4



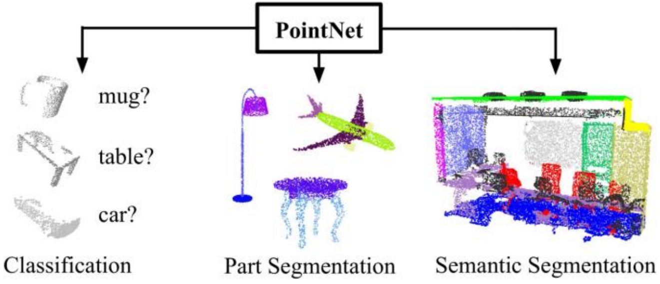
Merged Deep Fully Connected Neural Networks (DNN)

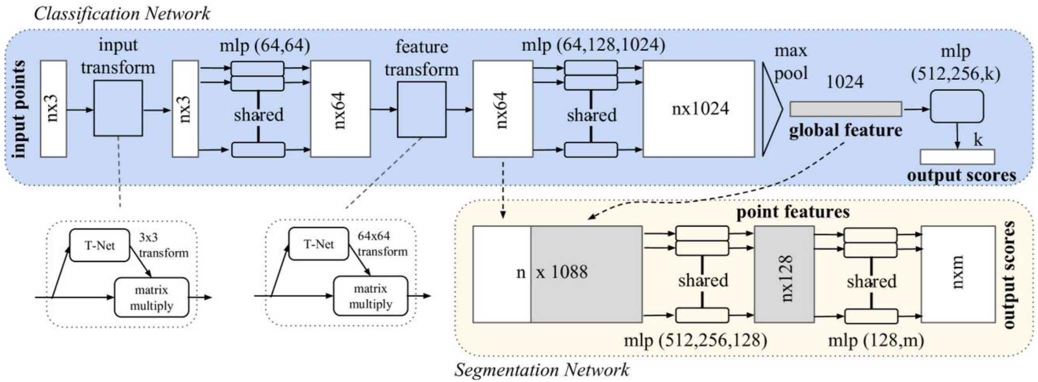
- image-based approach
  - EMB1 alone
  - EMB2, EMB3 together
  - Tile1, Tile2 and Tile3 together

- 3 fully connected hidden layers
- 50 nodes in each hidden layer
- outputs calibrated energy values



PointNet model



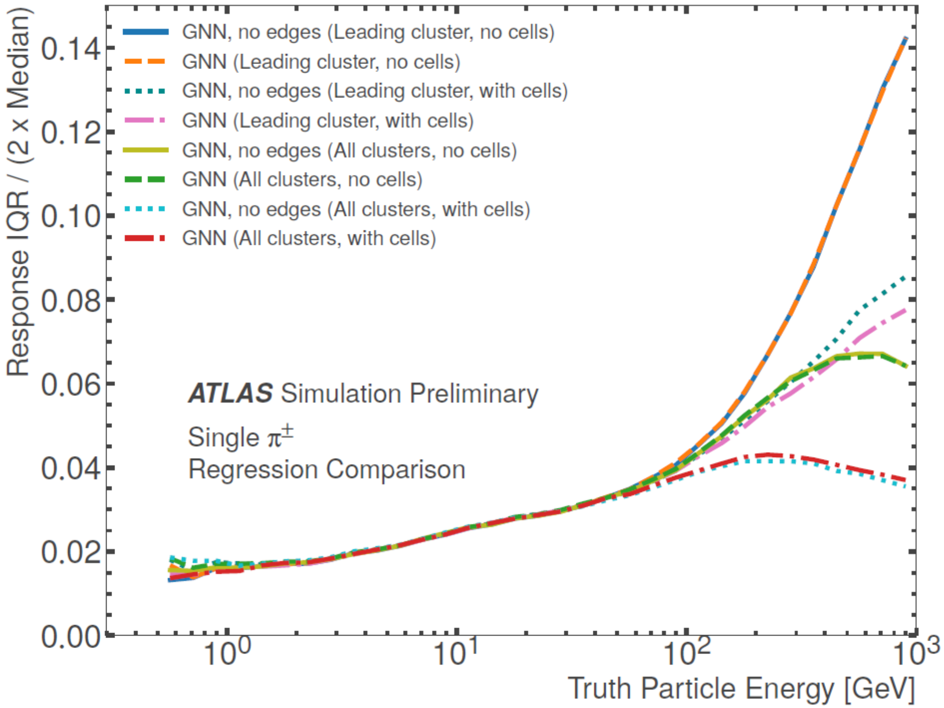


- Several learning tasks: classification, part segmentation, semantic segmentation
- permutation invariant
- transformation equivariance
- both shape classification & segmentation
- robust to data corruption → critical points

- no local context → global feature learning
- generalization to unseen scenes → global features depend on absolute coordinates
- no rotation/shape equivariance

Calo + track results using cell-level information

- Severall GNN configurations attempted
    - Leadining cluster only VS all clusters
    - With VS w/o edges
    - With VS w/o cell info
- GNN with cell-level data (red, light blue) improves resolution compared to versions trained without this information under several configurations

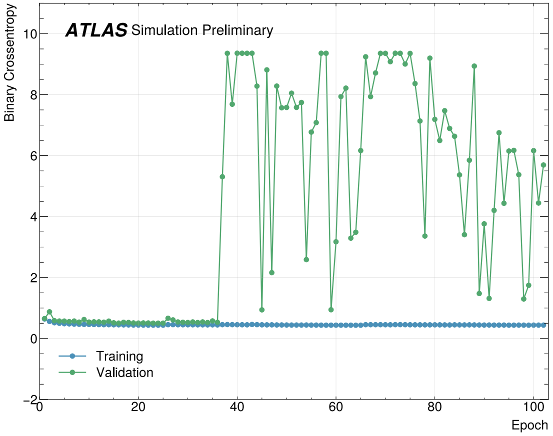


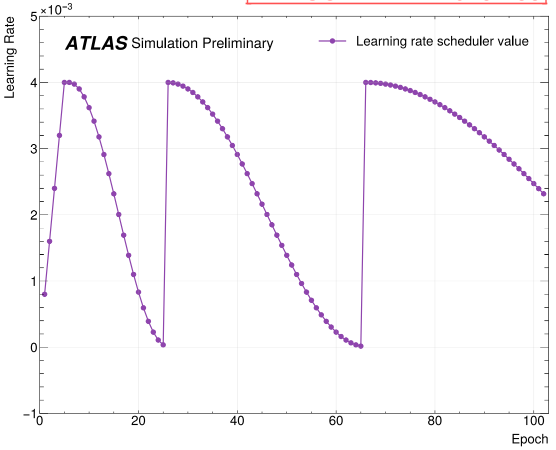
Results

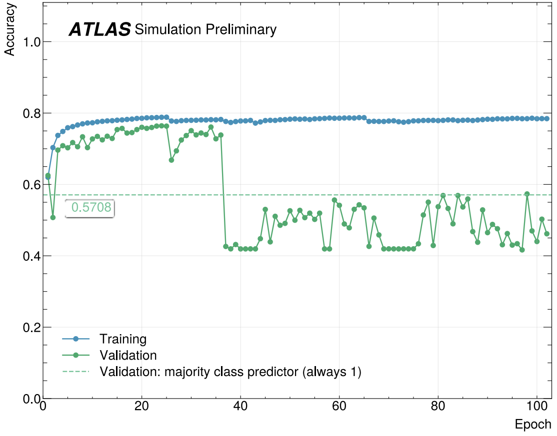
[ATL-COM-PHYS-2025-488]

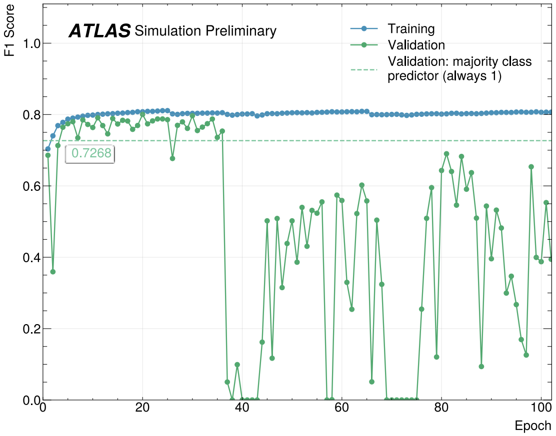
WBCE + adam + cosine annealing

- Very instable
- Training diverge
- Although initial metrics are satisfying, comparison with trivial baseline suggest model is just learning to predict majority class





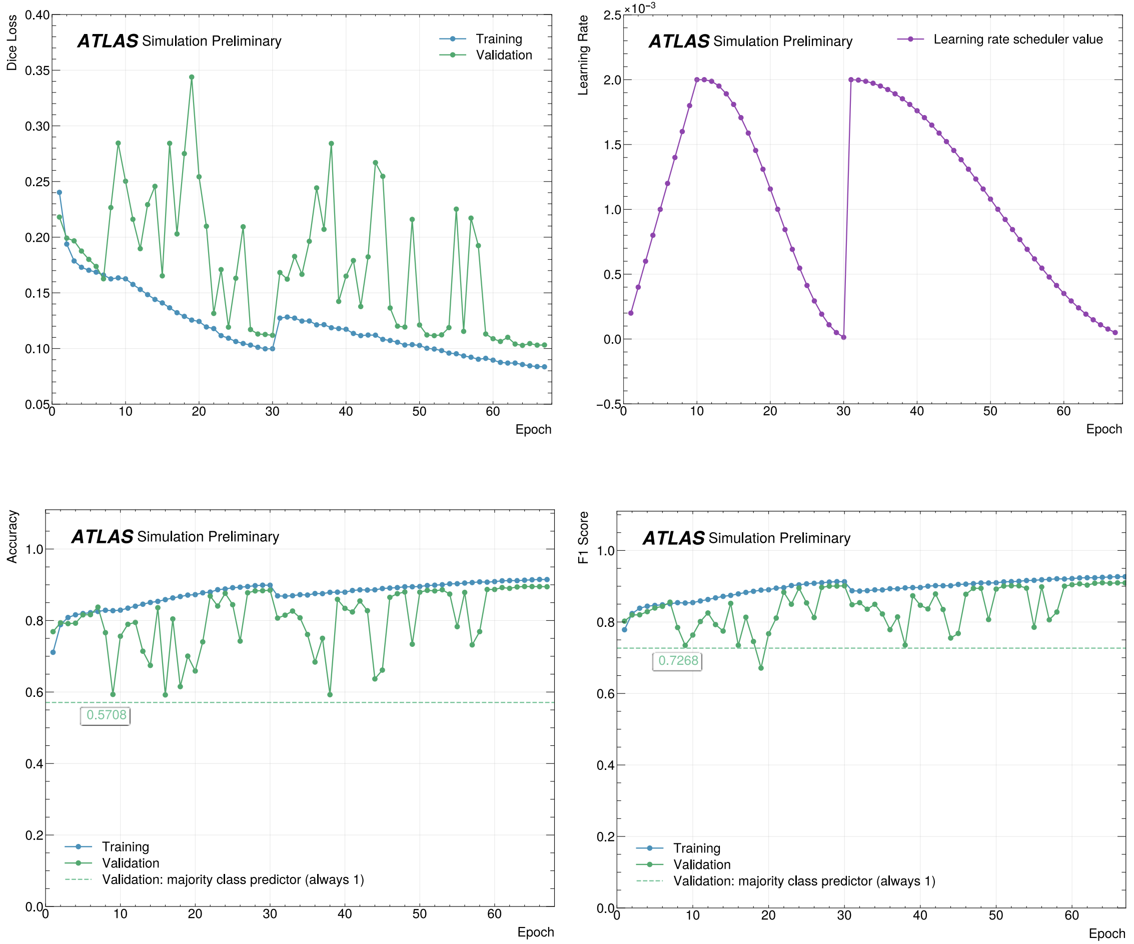




[ATL-COM-PHYS-2025-488]

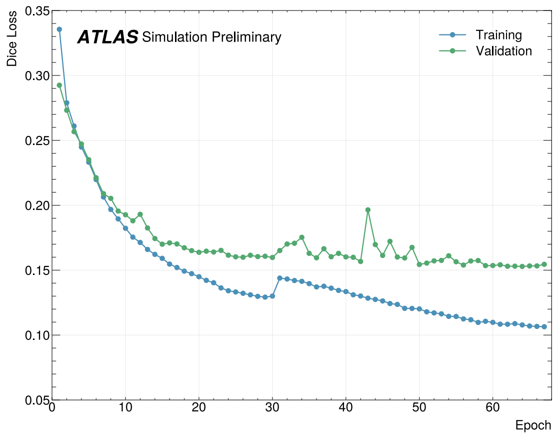
Dice + adamW + cyclical LR with warmup

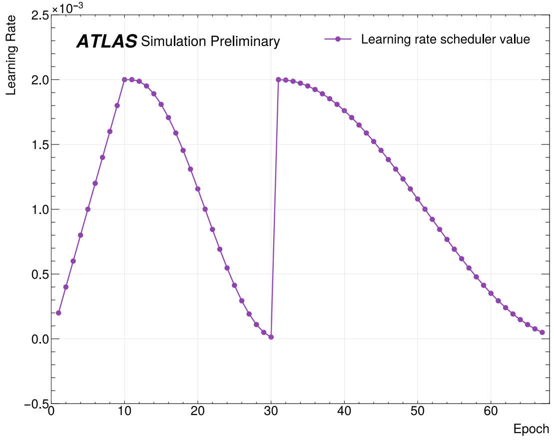
- More stable, although high variability in validation curves
- Sound training curves suggest little overfitting and potential to still improve
- F1 score close to 90%, much better than trivial baseline



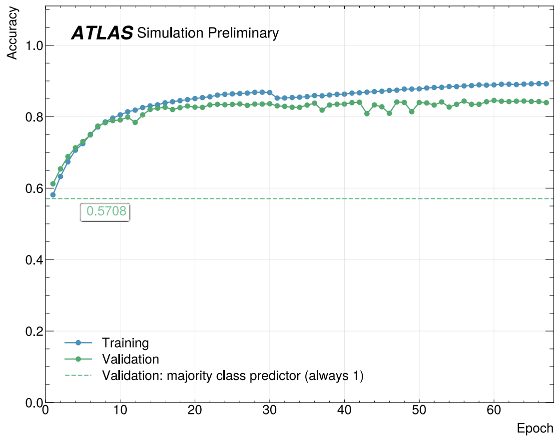
[ATL-COM-PHYS-2025-488]

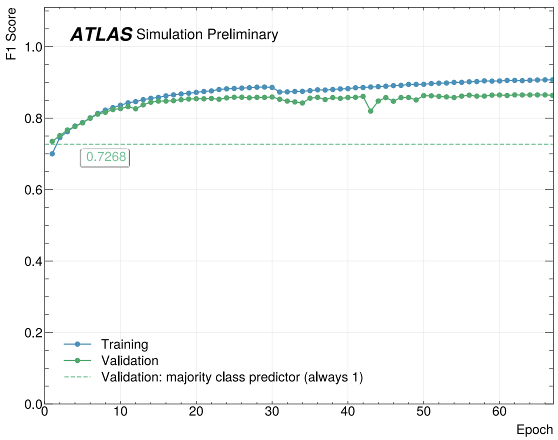
Dice + SGD + cyclical LR with warmup





- SGD stabilizes training, even validation curves are smoother
- Wider training/validation gap
- Flatten validation improvement at the end of training
- F1 score close to 85%, much better than trivial baseline but slightly worse than adamW results

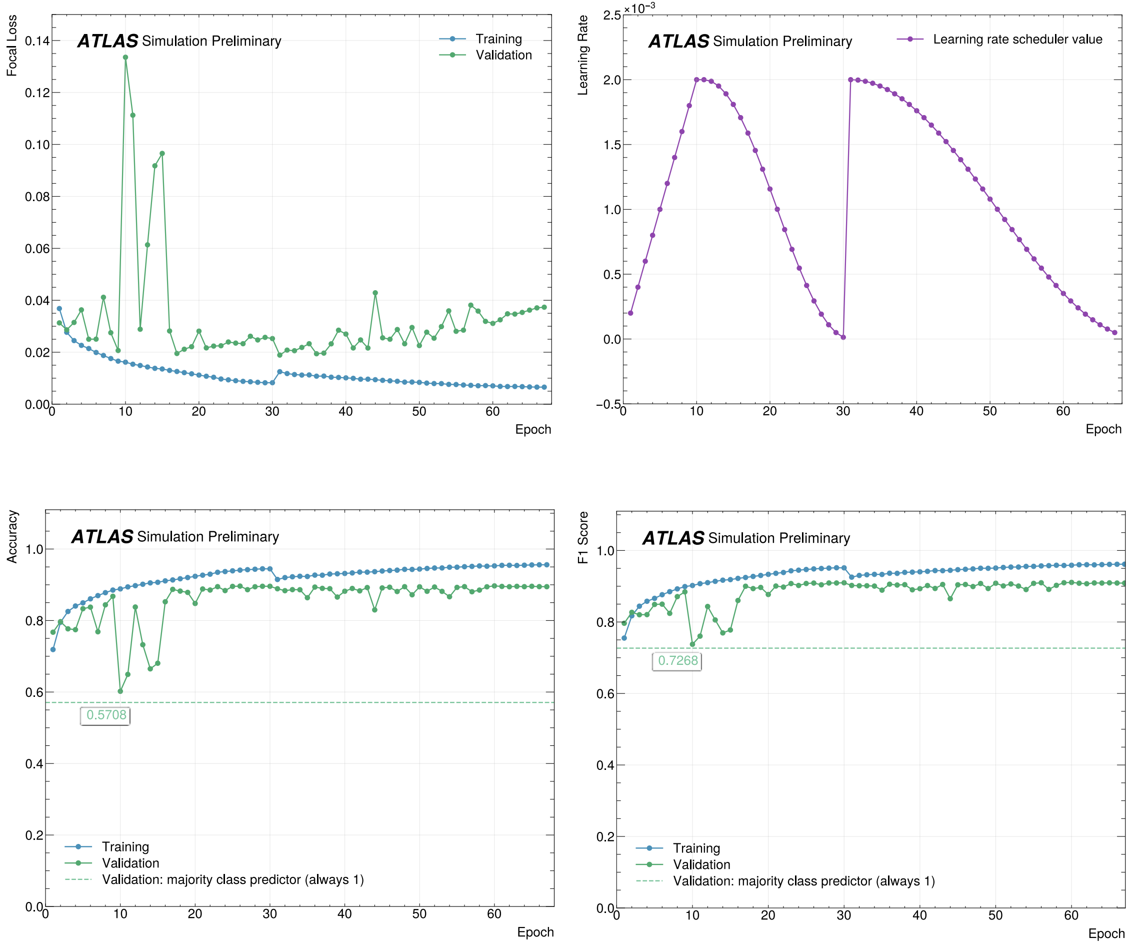




[ATL-COM-PHYS-2025-488]

Focal + adamW + cyclical LR with warmup

- More stable, although high variability in validation curves
- Increasing overfitting in final epochs
- F1 score close to 90%, much better than trivial baseline
- Comparable to Dice alternative (just slightly better)



[ATL-COM-PHYS-2025-488]

Focal + SGD + cyclical LR with warmup

- SGD stabilizes training, even validation curves are smoother
- Widening training/validation gap at end of training
- Flatten validation improvement in the end
- F1 score close to 90%, much better than trivial baseline and close to best performance obtained