Prong Segmentation using Point Set Transformers in Multiple View Neutrino Detectors

Jiaxi Liu, Alejandro Yankelevich, Dikshant Sagar,
Edgar Robles, Jianming Bian, Pierre Baldi, Wenjie Wu

University of California, Irvine

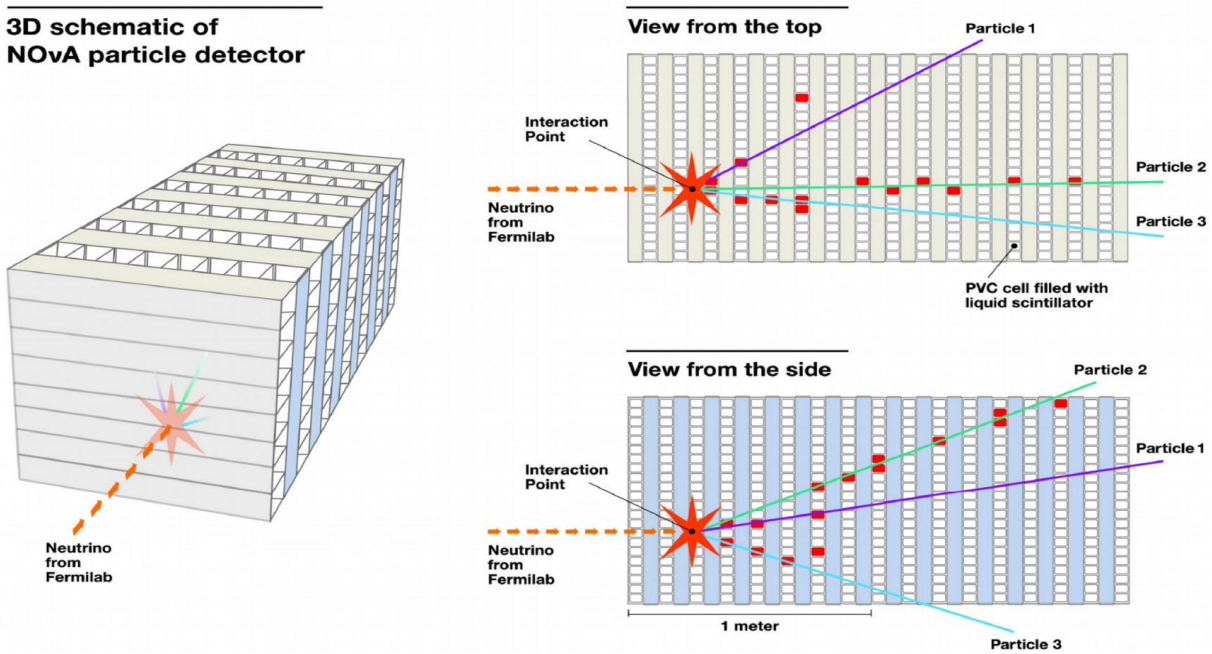
2026.06, NPML 2026

Outline

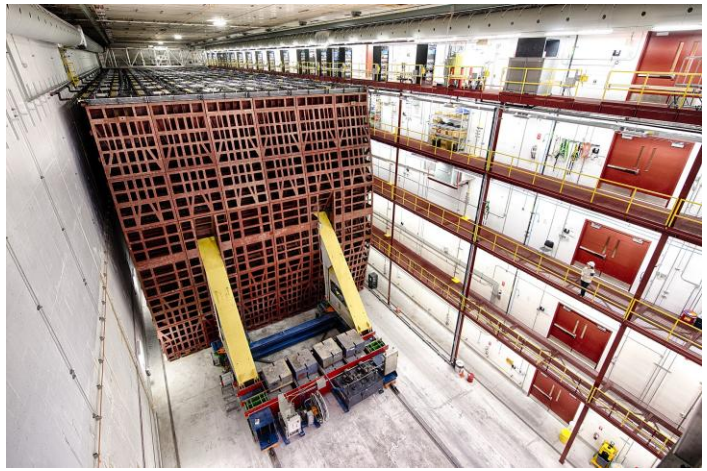
- NOvA Experiment
- Prong Segmentation in NOvA
  - Motivation for Point Set Transformers
  - PST Architecture & Enhancements
  - PST Performance Benchmarks
- PST Application in Other Multiple View Neutrino Detectors

NuMI Off-Axis ν_e Appearance (**NOvA**) Experiment

- Long-baseline neutrino oscillation experiment detecting neutrinos from Fermilab's NuMI beam, to precisely measure oscillation parameters and determine mass ordering.
- The detectors consist of 4cm×6cm PVC cells filled with liquid scintillator, arranged in planes of alternating orientation: view from the top (**XZ view**) & view from the side (**YZ view**).



Near Detector (at Fermilab)



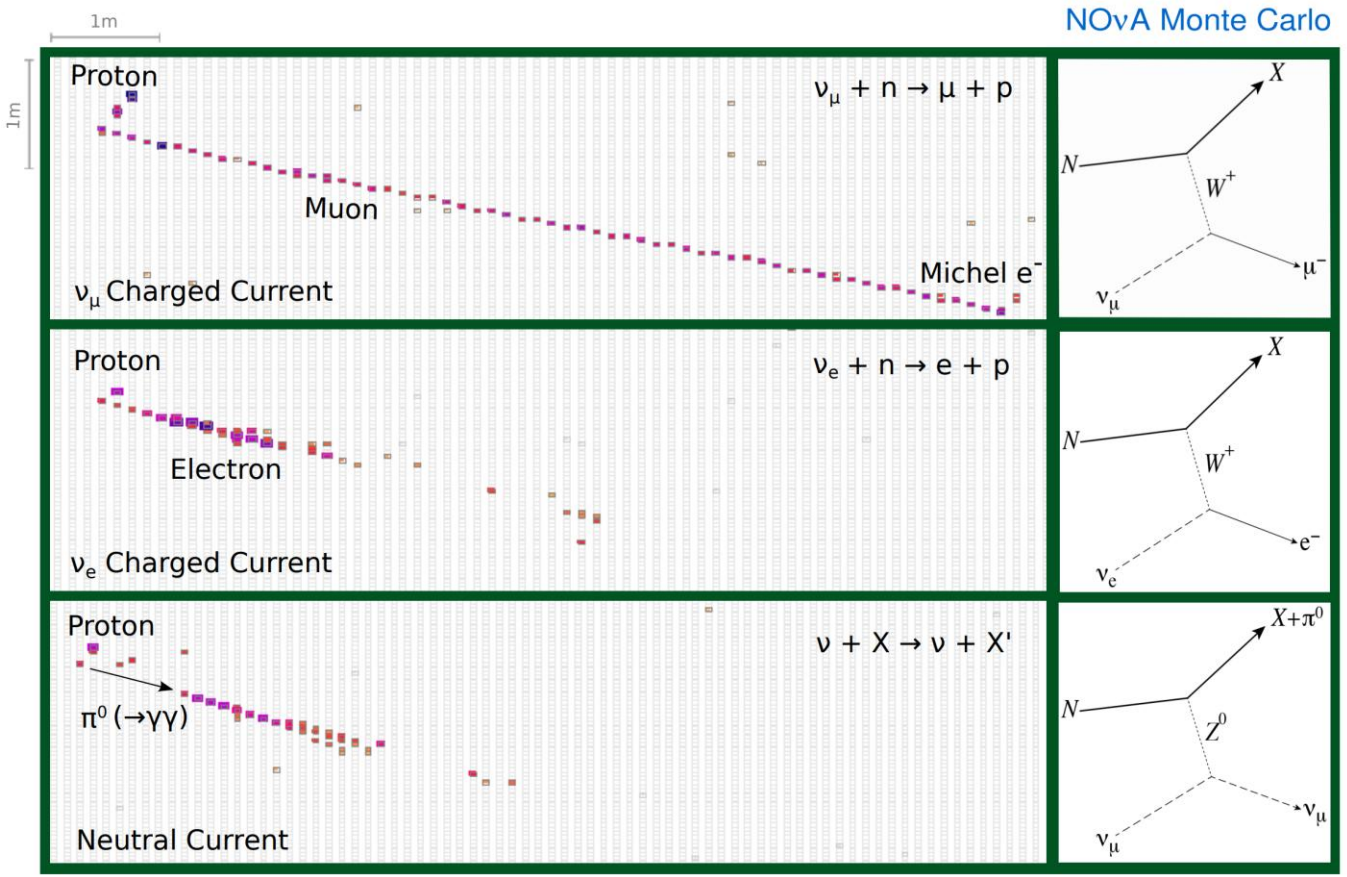
Far Detector (~810 km away)

NOvA Event Topologies

- For each event, NOvA output two 2D images that cannot be combined into one 3D image, because the third-coordinate information is lost in each view.

- Different neutrino flavor can be distinguished by the distinct behaviors of their final-state particles;

➢ Segmenting these individual particle tracks and showers (**prongs**) is critical for both flavor identification and energy reconstruction.

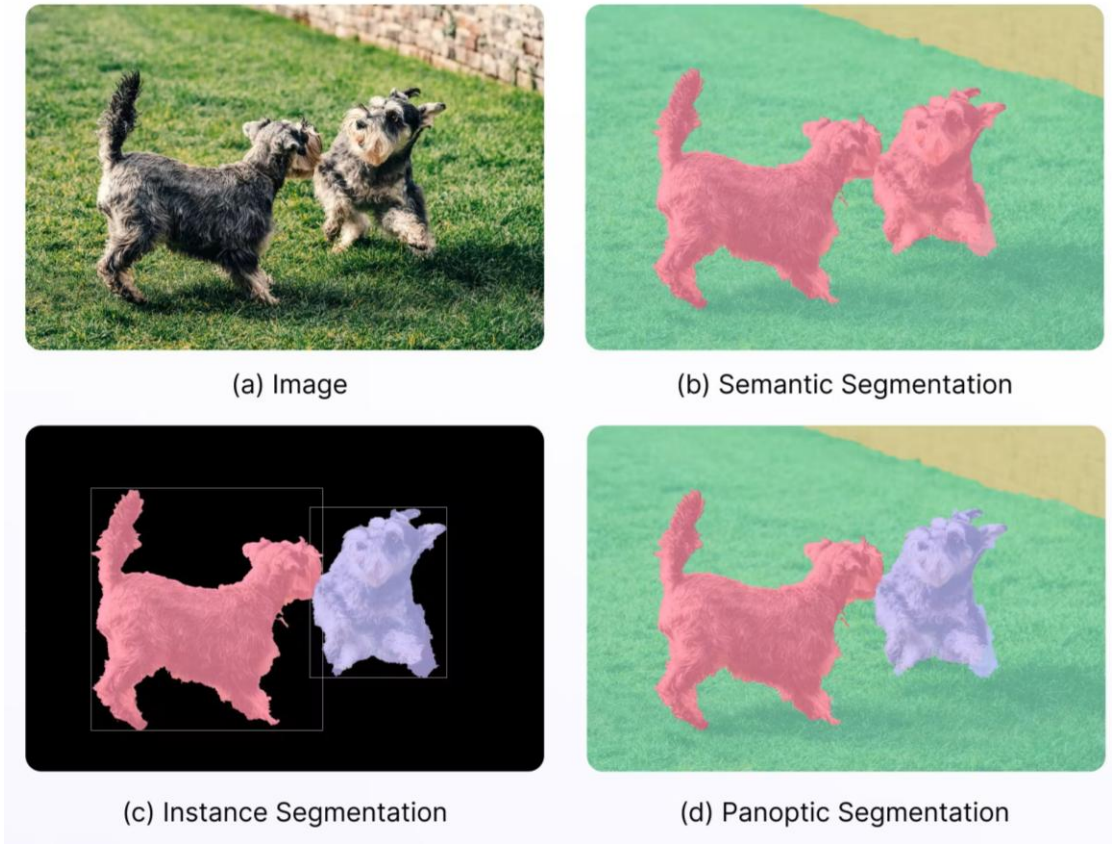


Prong Segmentation in NOvA

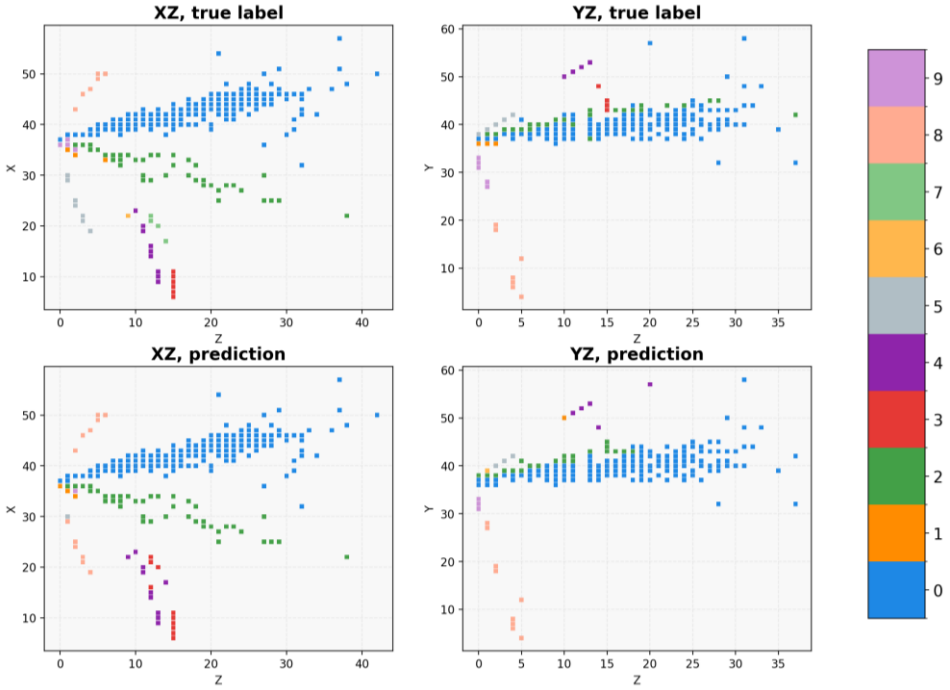
The prong segmentation task includes two aspects:

- **Semantic segmentation:**
  - Label the class corresponding to each point;
- **Instance segmentation:**
  - Cluster all points that come from the same object;

➢ **The combination of these two parts is called panoptic segmentation.**

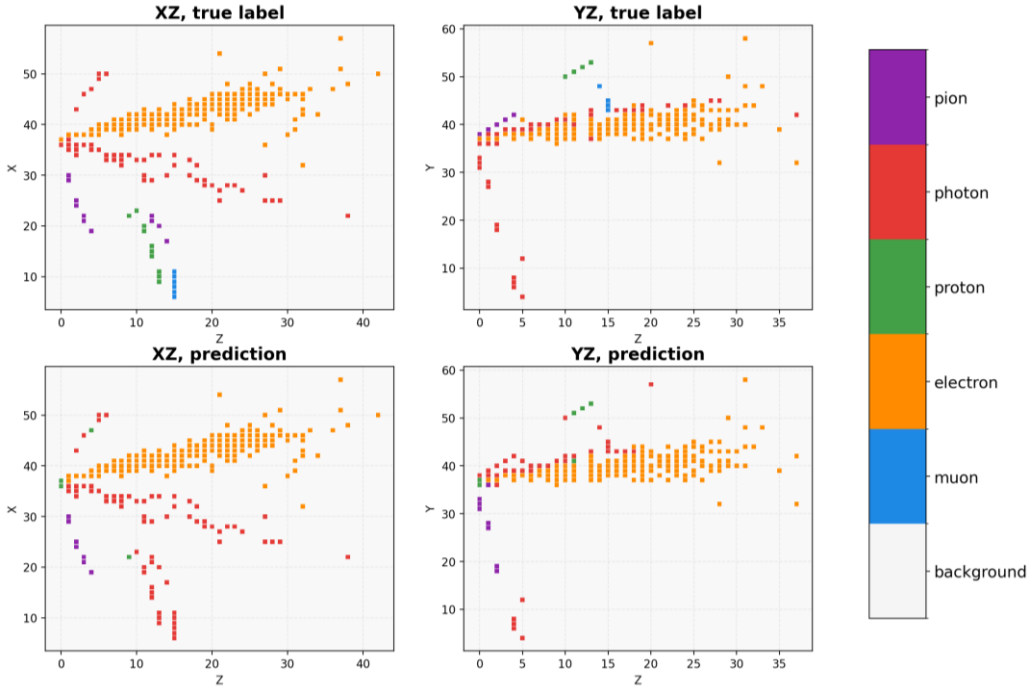


Prong Segmentation in NOvA



- **Instance segmentation:**

Cluster all hits that come from the same prong, label each prong from 0 to 9;
Labels are shared between two views.



- **Semantic segmentation:**

Label the particle type corresponding to each hit (muon, electron, proton, photon, pion, background);

Motivation for Point Set Transformers

- Current prong segmentation method: fuzzy k-means clustering algorithm, J. Phys. Conf. Ser. 664 072035 (2015).
- ➢ **Can deep learning methods further improve the performance under current computational resource limitations?**

Three main challenges for this task:

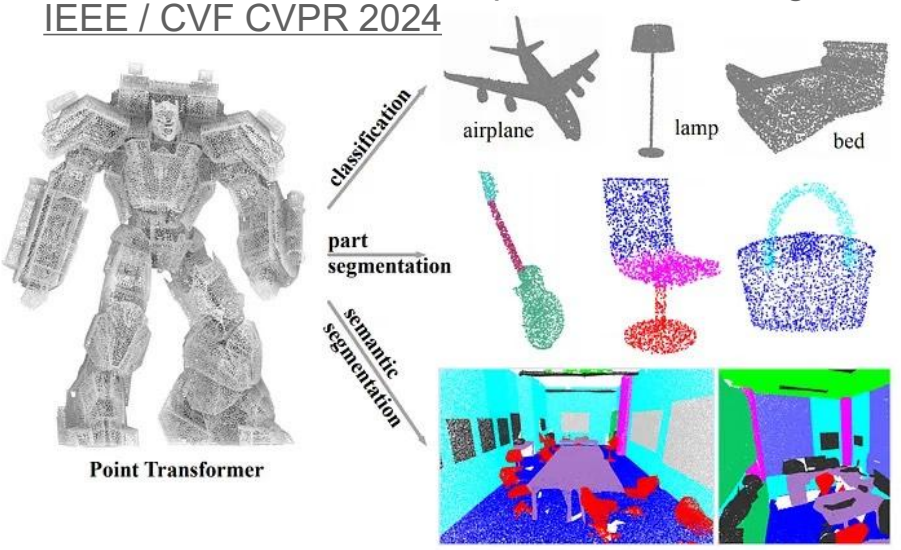
- **Output images are sparse → Traditional CNN model cannot capture meaningful features;**
- **Two views have mismatched coordinates→ Cannot be directly fused into one image;**
- **Current tradition method offers a computationally efficient baseline→ new method must operate under limited resources.**

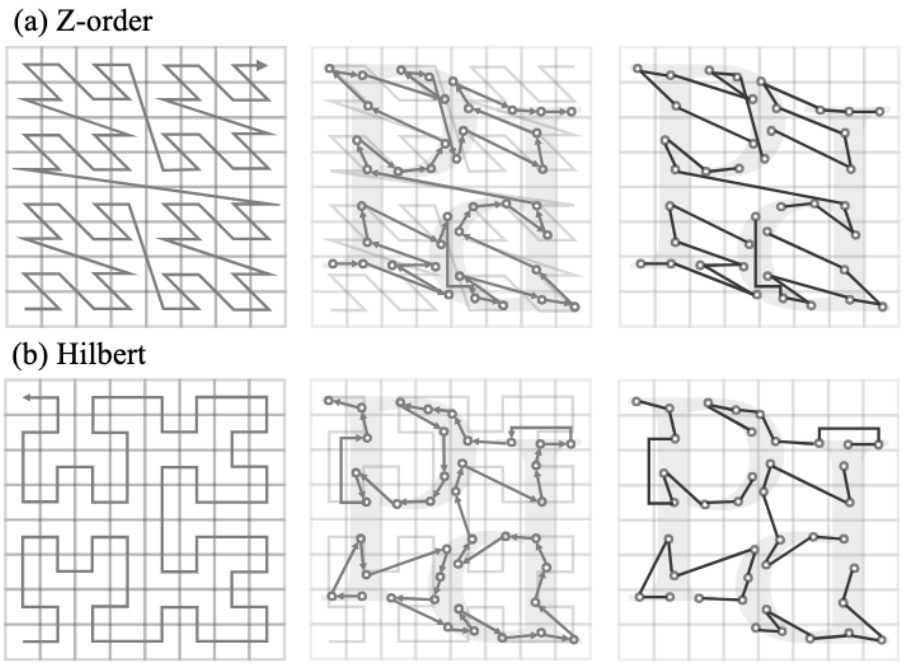
Point Set Transformer (PST)

We explored multiple models, including Region-based CNN (R-CNN) and Graph Attention Transformer (GAT), ultimately selecting **Point Set Transformer (PST)**:

- Treats inputs as point clouds, which naturally fits sparse data.
- Applies permutation-invariant self-attention across points, making it ideal for unordered point sets.
- Global attention has a complexity of $O(N^2)$, so attention operation usually limited to local k-NN neighborhoods (or via point serialization as in Point Transformer V3).

"Point Transformer" IEEE / CVF ICCV 2021
"Point Transformer V3: Simpler, Faster, Stronger" IEEE / CVF CVPR 2024



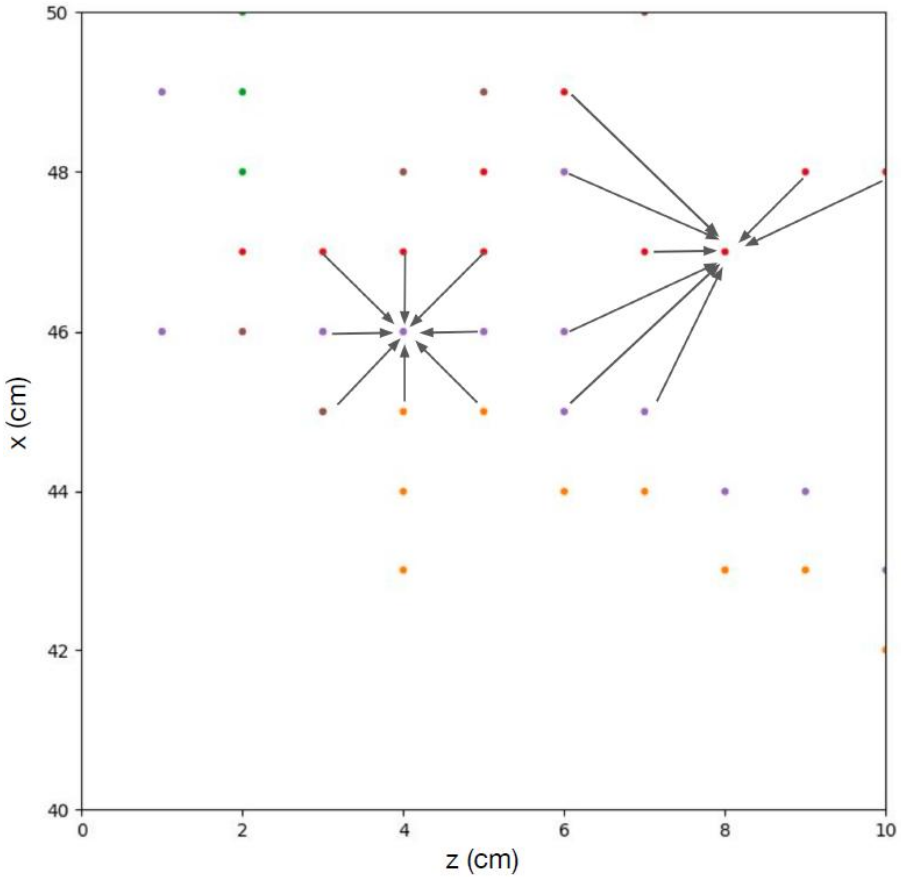


Point Serialization Method in Point Transformer V3

K-NN Attention in PST

Instead of global self-attention, we first employed k-NN attention to effectively process the point clouds:

- Reduces computational complexity from $O(N^2)$ to $O(N)$;
- Restricts model's focus to neighboring hits where actual particle interactions occur;
- Replace the traditional pixel grid with dynamic local neighborhood, adapting to non-uniform data densities.

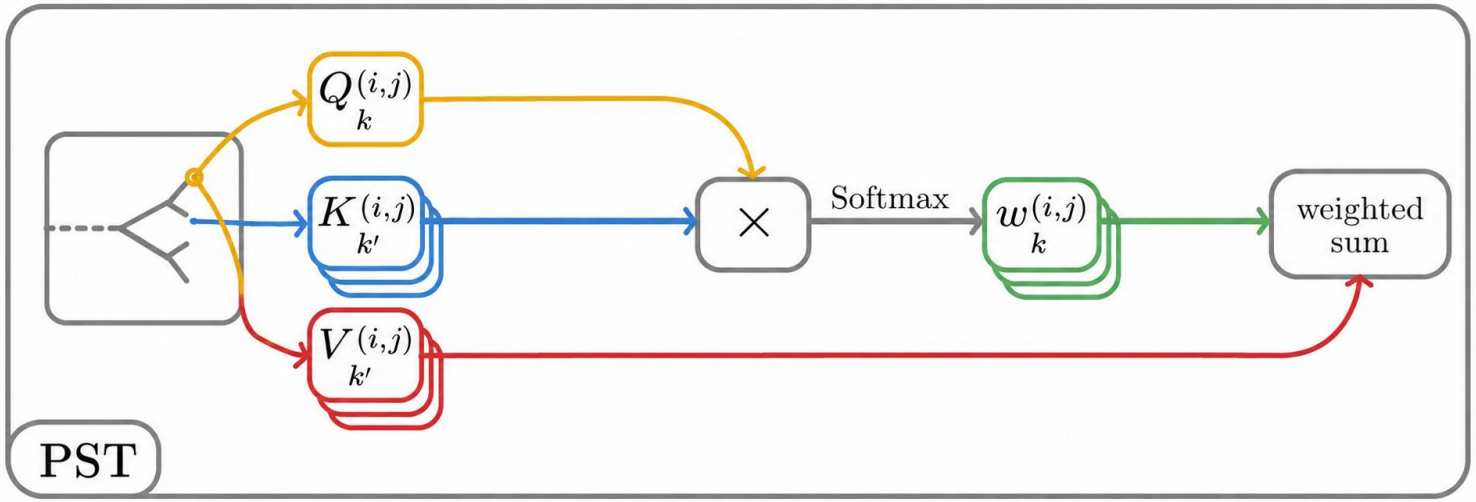


Example of 8-nearest neighbors

Heterogeneous Attention Mechanism

Intra-view Point Set Attention:

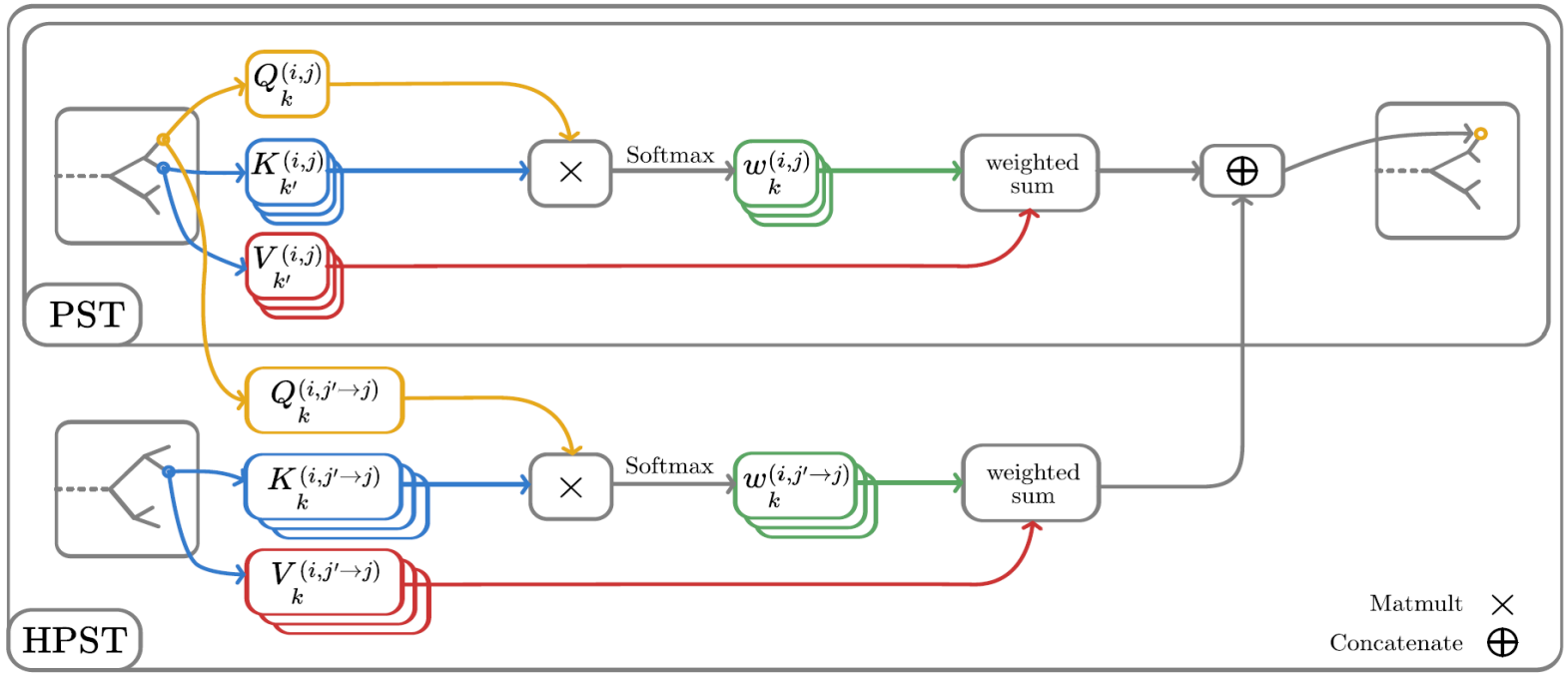
- Computes the weight from point k to k', adding a linear Relative positional Encoding (RPE) to preserve spatial relationships: $w_{kk'} = Q_k K_{k'} + RPE(x_k - x_{k'})$
- Normalizes the weights across all the k-NN neighborhood connected to point k: $h_k = \sum_{k'} \mathrm{Softmax}(w_{kk'}) V_{k'}$
- Attention is computed individually for each view; no information is shared across the views.



Heterogeneous Attention Mechanism

Inter-view Point Set Attention:

- Computes attention weights from point k (in view j) and point k' (in view j'): $w_{kk'}^{j\to j'} = Q_k^{j\to j'} K_{k'}^{j\to j'}$
- Determines the cross-view k-NN neighborhoods by using the shared Z-coordinate.
- Enables information to flow between the two views for a better global reconstruction of the event.
- Concatenates the bidirectional inter-view attentions with intra-view attention to form the final output.



HPST Performance

- Dataset: ~900k simulated NOvA events, including both ν_e-CC & ν_μ-CC.

- **HPST model can obtain a superior performance over the other models (R-CNN, GAT) while keeping a comparable memory usage.**

Model	Memory usage (MiB)	Time per sample (s)	OVR AUC	Segmentation accuracy
R-CNN	282.4 ± 37.43	265.3253 ± 2.012	0.732	0.343
GAT	29.8 ± 0.40	1.7381 ± 0.001	0.854	0.659
HPST (ours)	34.7 ± 1.00	7.0518 ± 0.001	0.968	0.835

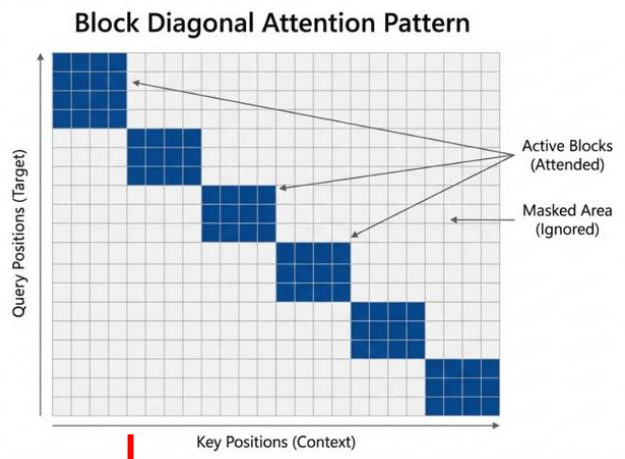
Source: Robles et al., 2025: Heterogeneous Point Set Transformers for Segmentation of Multiple View Particle Detectors

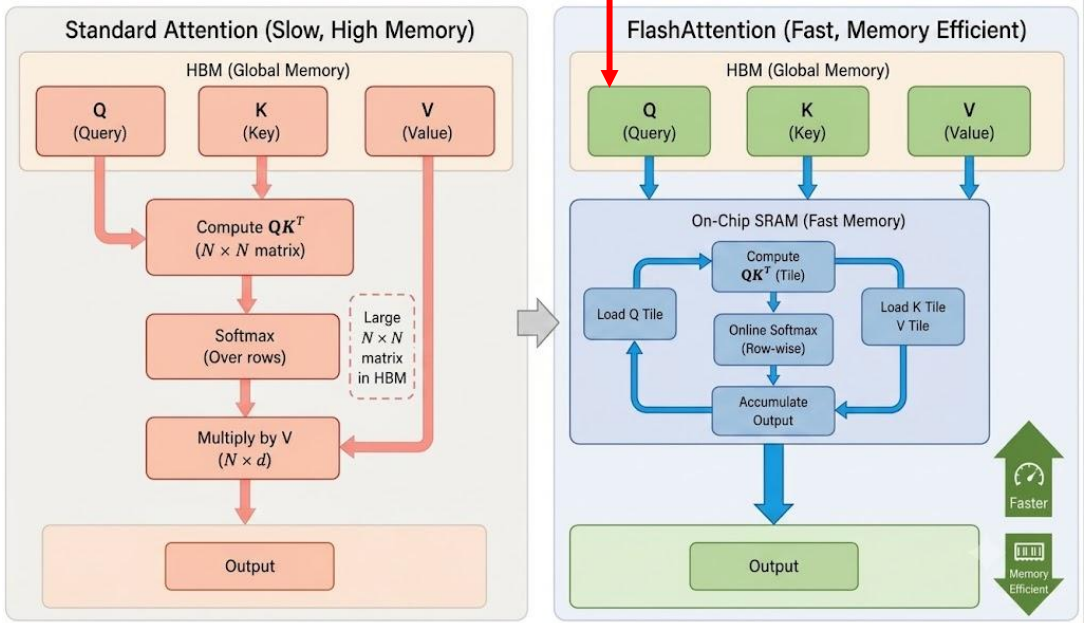
Heterogeneous PST: Architectural Enhancements

- As a resource-driven compromise, k-NN attention inherently limits the network from fully capturing long-range, global event topologies.

➢ **We integrated PyTorch's FlashAttention into our training pipeline to make event-level dense all-to-all attention computationally viable:**

- Effectively saves the vector of a whole batch;
- Tiles input matrices into blocks to process them inside ultra-fast Static RAM;
- Eliminates the need to store the full intermediate $N \times N$ attention matrix in global memory.



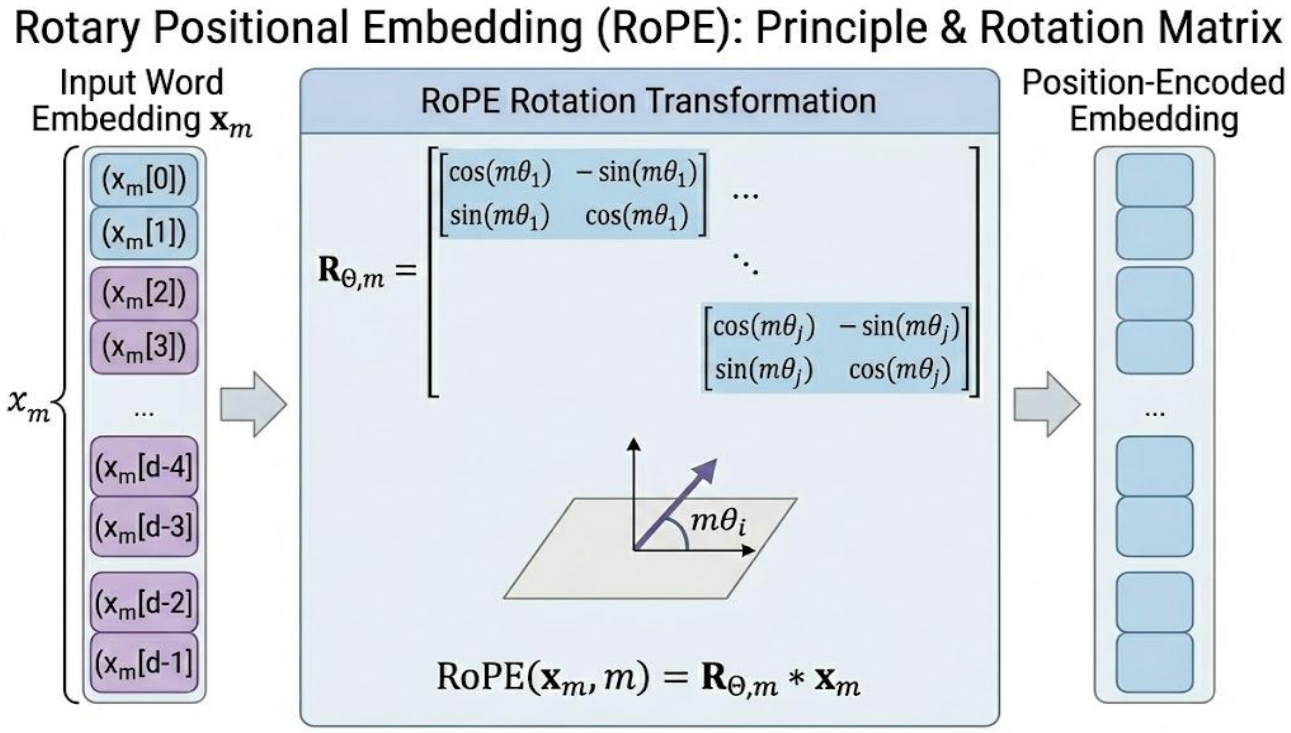


FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness

Heterogeneous PST: Architectural Enhancements

- Under a dense all-to-all attention framework, standard RPE incurs an expensive $O(N^2)$ complexity penalty.

➢ Upgrade this method to Rotary Positional Embedding (RoPE), effectively preserving relative spatial relationships without consuming additional computational resources.

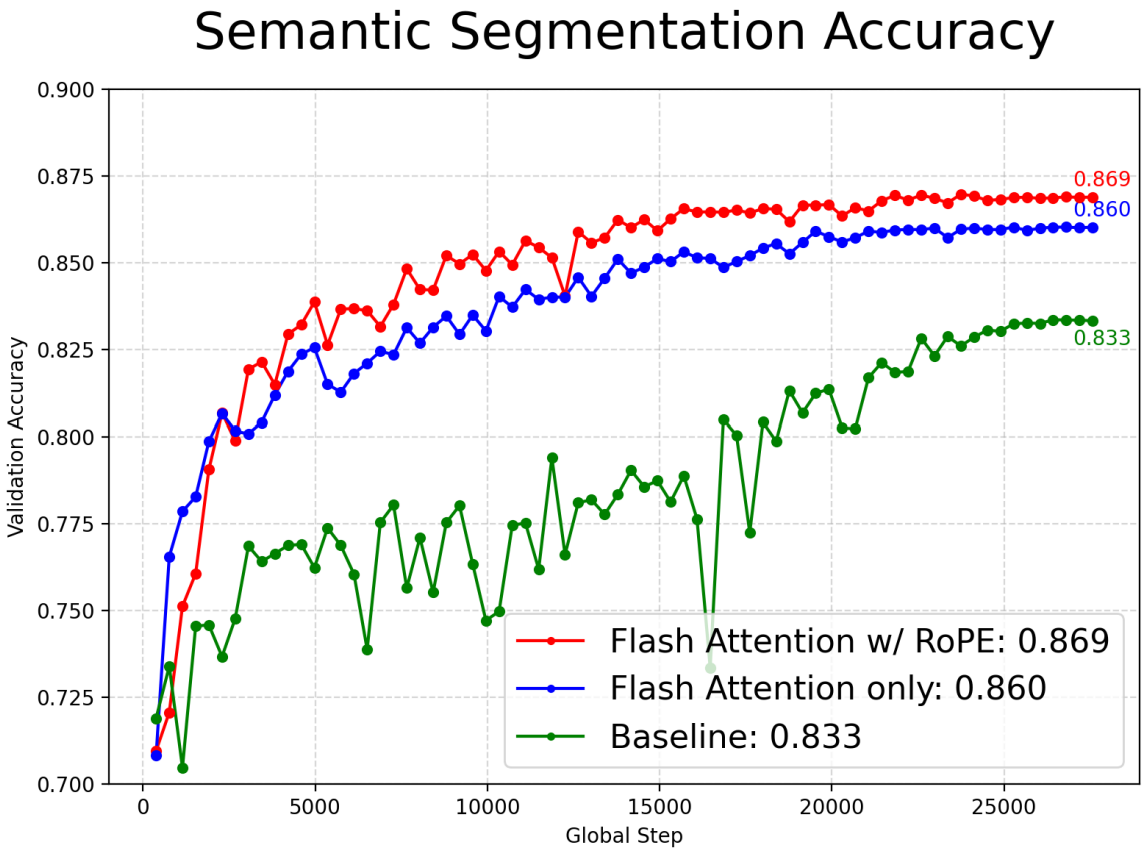


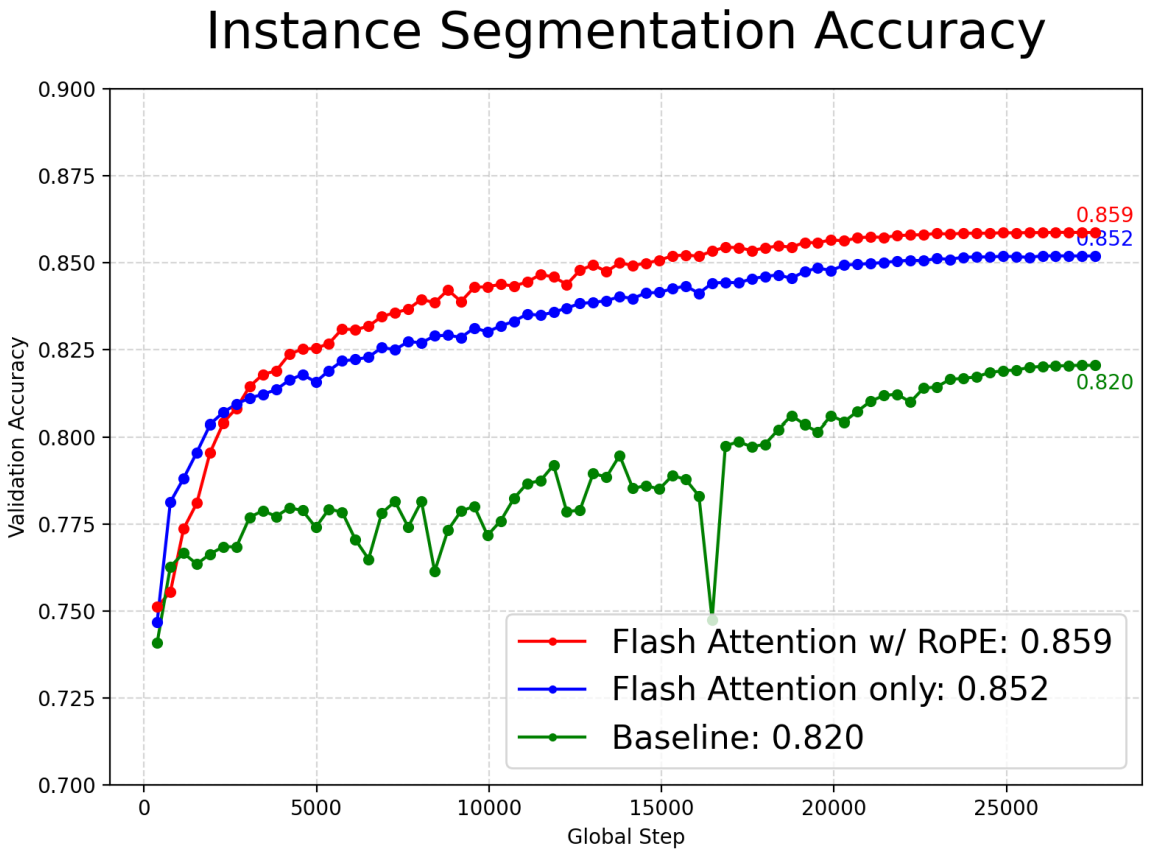
RoFormer: Enhanced Transformer with Rotary Position Embedding

- Embeds position information by rotating Q&K vectors, where a data point at coordinate m is rotated by a sequence of angles $\{m\theta_j\}$;
- The relative position between point m and n is naturally preserved as a series of 2×2 rotation matrices with rotation angles $\{(m-n)\theta_j\}$.

Performance Benchmark: Validation Accuracy

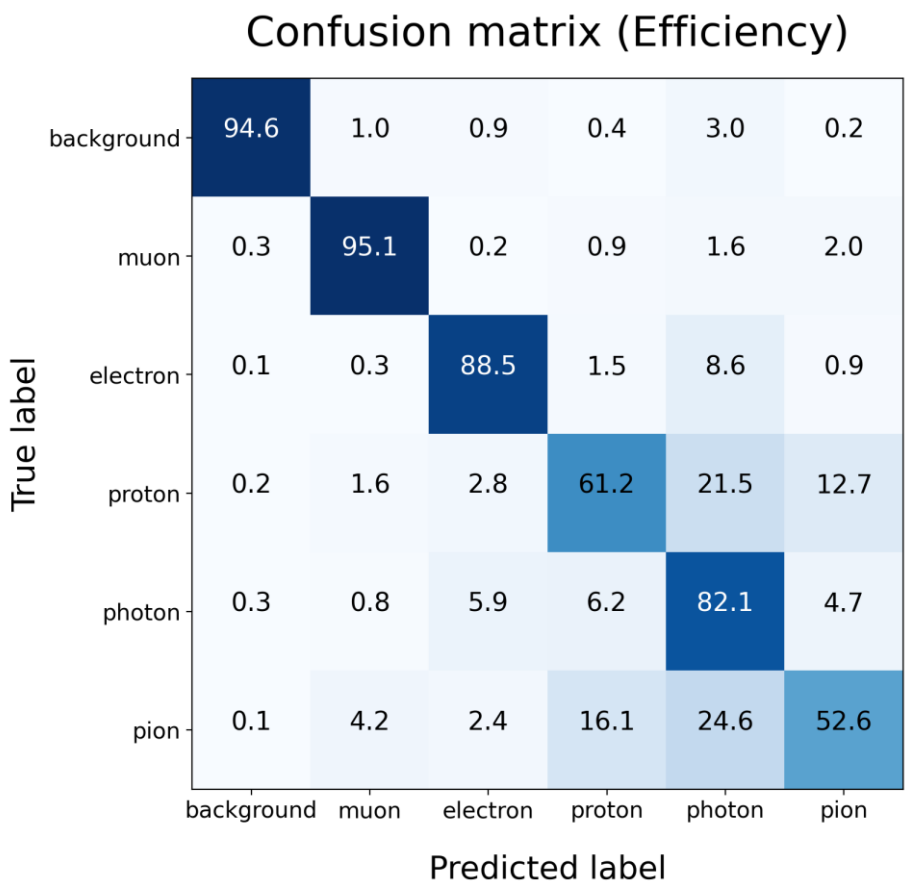
- Baseline & FlashAttention/RoPE integration validation accuracy results:



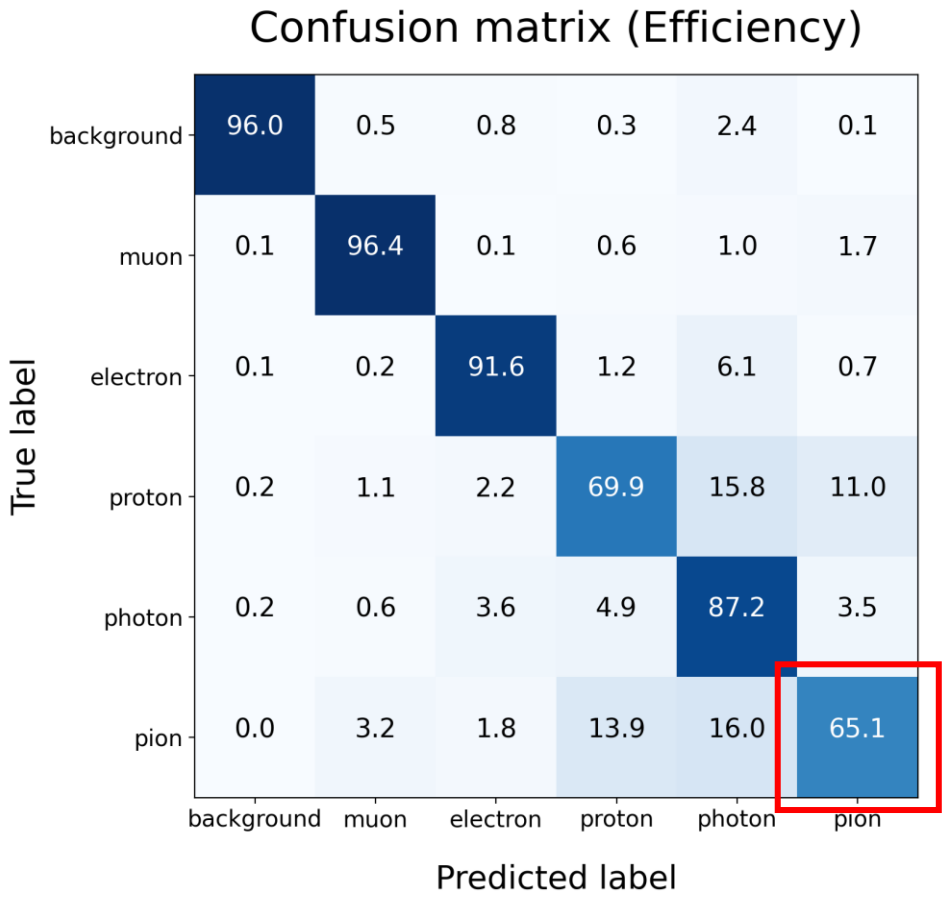


Performance Benchmark: Semantic Segmentation

- **Semantic segmentation performance**: hit-level confusion matrices for predicted & true labels



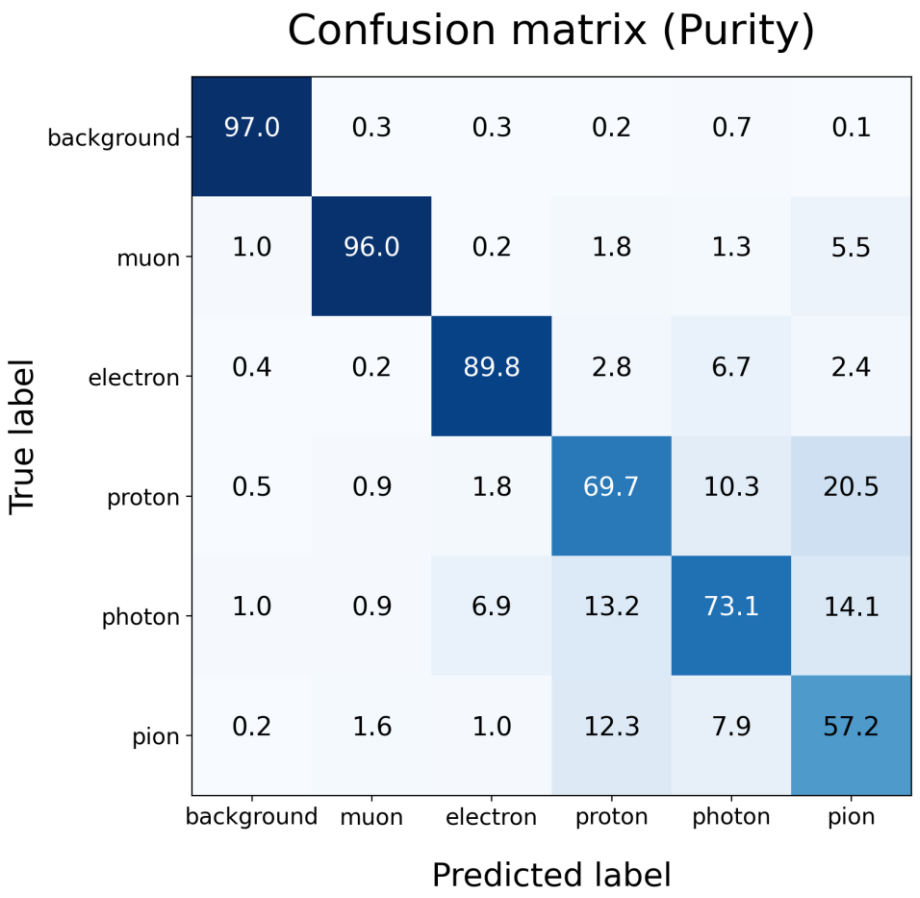
Baseline



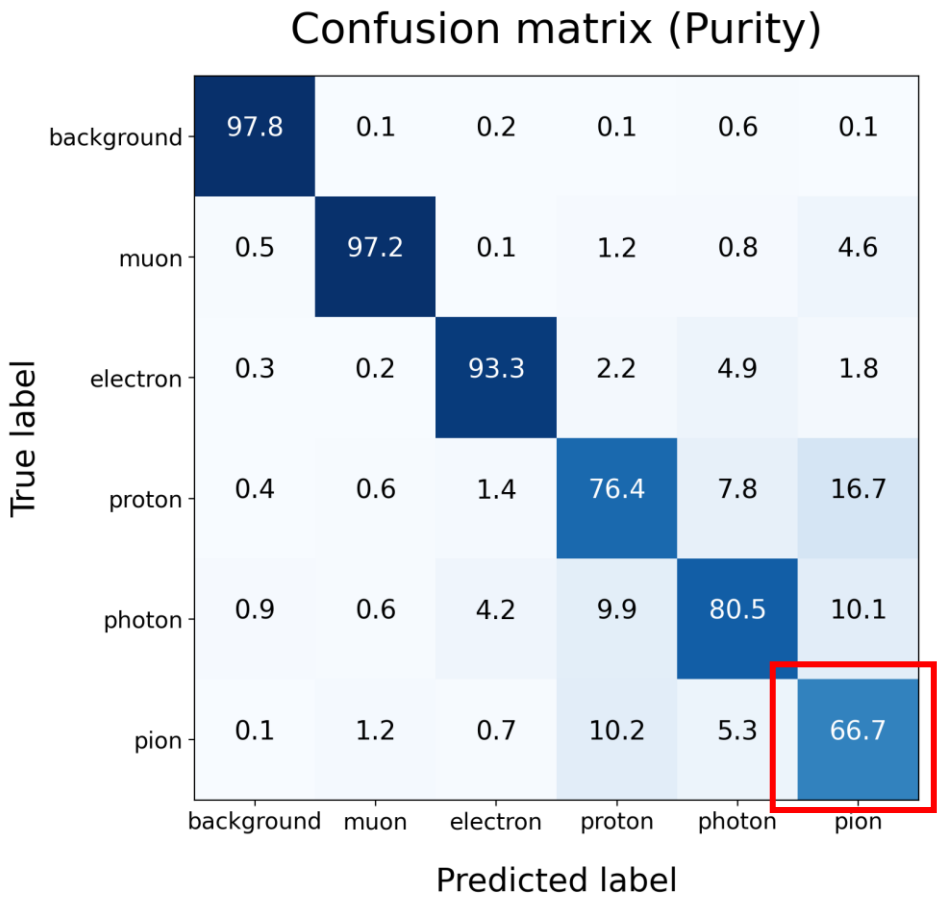
FlashAttention/RoPE

Performance Benchmark: Semantic Segmentation

- **Semantic segmentation performance**: hit-level confusion matrices for predicted & true labels



Baseline



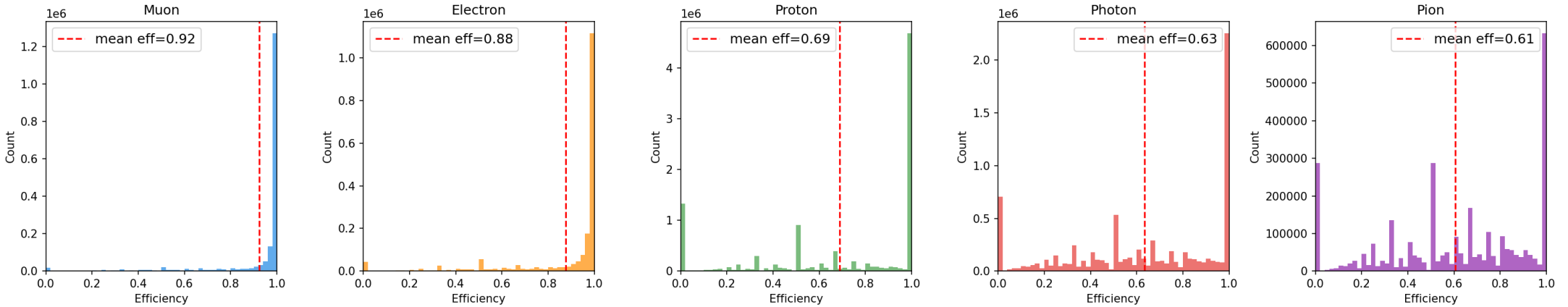
FlashAttention/RoPE

Performance Benchmark: Instance Segmentation

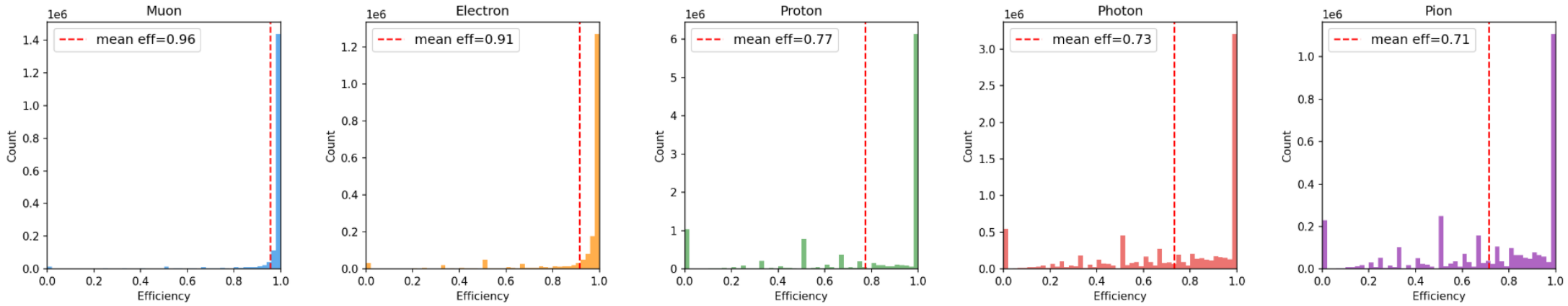
- **Instance segmentation performance**: prong-level efficiencies and purities

  Efficiency: % of hits in true prong predicted correctly.

Baseline



FlashAttention/ RoPE

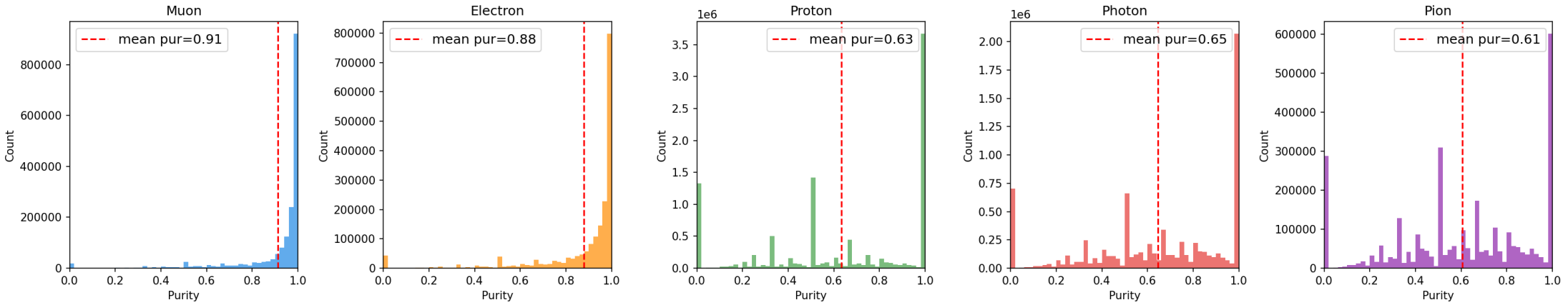


Performance Benchmark: Instance Segmentation

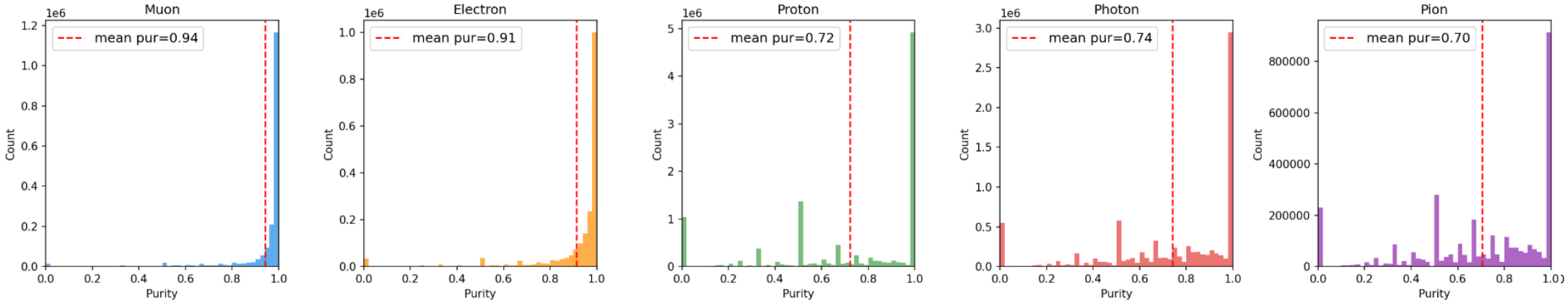
- **Instance segmentation performance**: prong-level efficiencies and purities

  Purity: % of hits in predicted prong assigned to correct prong.

Baseline

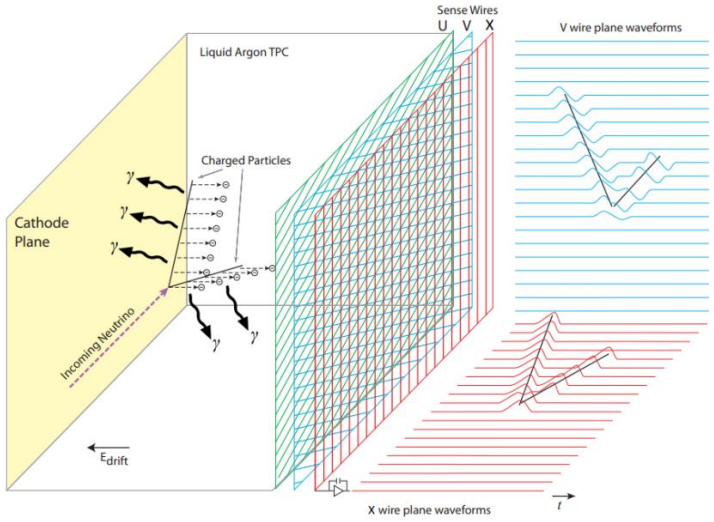


FlashAttention/ RoPE



PST Application in LArTPC

- This PST architecture can extend to other multi-view neutrino detectors, such as Liquid Argon Time Projection Chambers (**LArTPCs**).
- LArTPC event topologies: Similar to NOvA, events are recorded as multiple 2D projections; however, unlike NOvA, the native 3D spatial context remains uncompromised and can be fully reconstructed.



The Operating Principle of LArTPC

Model	Memory (MiB)	Time (s)	Semantic Segmentation AUC	Instance Segmentation Accuracy
2D R-CNN	440.50 ± 51.04	1.575 ± 0.091	0.526	0.518
2D GAT	88.60 ± 7.56	0.230 ± 0.025	0.833	0.659
3D GAT	506.10 ± 30.13	0.722 ± 0.060	0.859	0.727
2D HPST (ours)	**99.10** ± 7.39	**0.354** ± 0.019	0.936	0.779
2D PST (ours)	138.10 ± 11.29	**0.254** ± 0.021	0.949	0.827
3D PST (ours)	**170.20** ± 9.65	**0.140** ± 0.012	**0.982**	**0.889**

Source: Robles et al., 2025: Particle hit clustering and identification using point set transformers in liquid argon time projection chambers

Dataset: simulated 2m×7m×7m LArTPC with square 5mm pixel-based readout, ~200K events in total;

2D HPST surpasses other 2D models with fewer resources, while 3D PST has the best performance at the cost of increased memory usage.

Summary

- **Heterogenous Point Set Transformer (HPST) model was developed for the NOvA prong segmentation task:**
  - Heterogenous attention mechanism for efficient information flow between views;
  - ➢ Outperforming performance under limited computational resources.

- Architecture Updates:
  - FlashAttention: Enable dense all-to-all attention instead of k-NN attention;
  - RoPE: Embed spatial information in vectors, without extra computational cost.
  - ➢ These architectural updates deliver significantly enhanced performance.

- PST architecture also generalizes to other multi-view detectors, such as LArTPC.

Thank You!

Backups

Loss Function

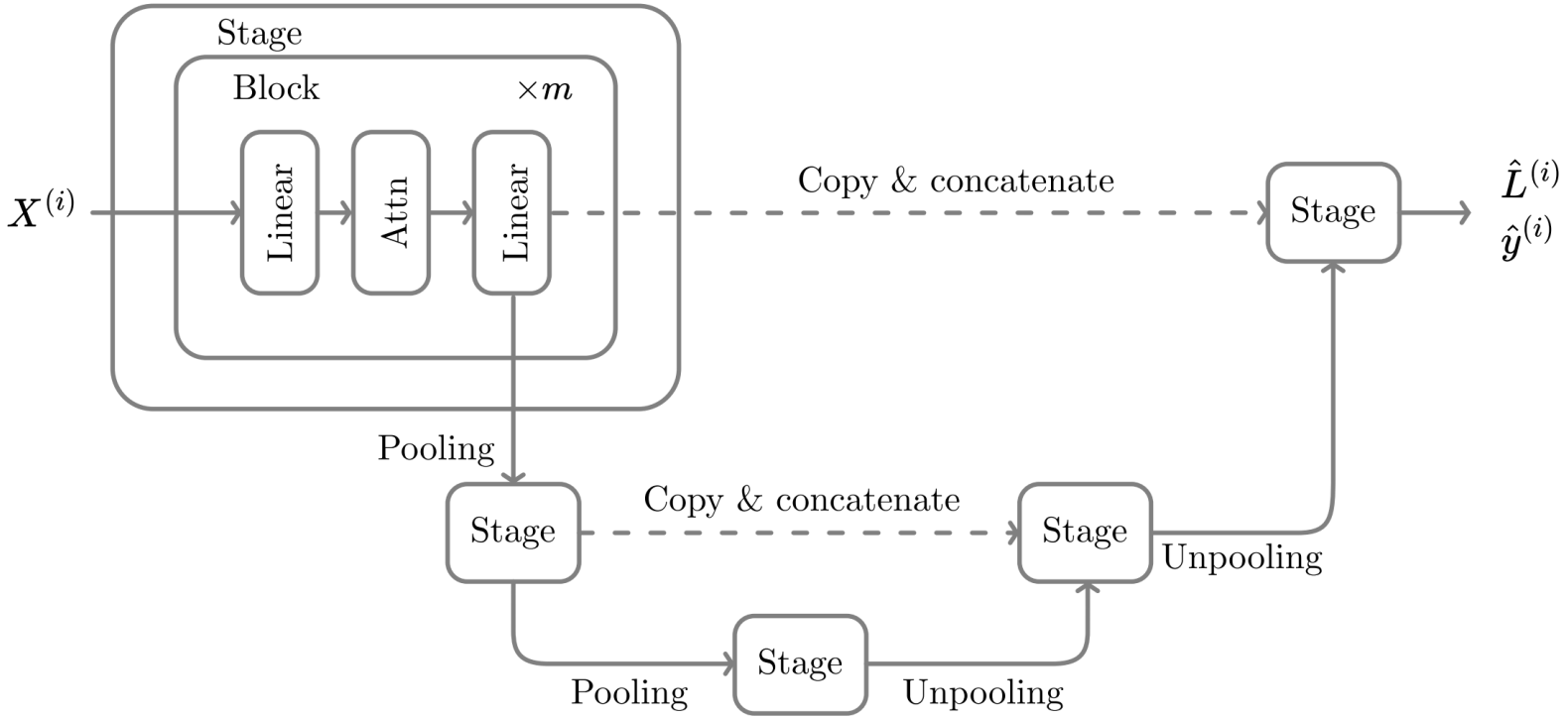
The loss function also consist of two parts: $\mathcal{L} = \lambda \mathcal{L}_{sem} + (1 - \lambda)\mathcal{L}_{ins}$

- **Semantic segmentation:**
  - **Multi-class cross-entropy loss**

- **Instance segmentation:**
  - **Multi-class cross-entropy loss with the best assignments of predicted and true labels**
  - The best assignment is calculated with the Hungarian algorithm.

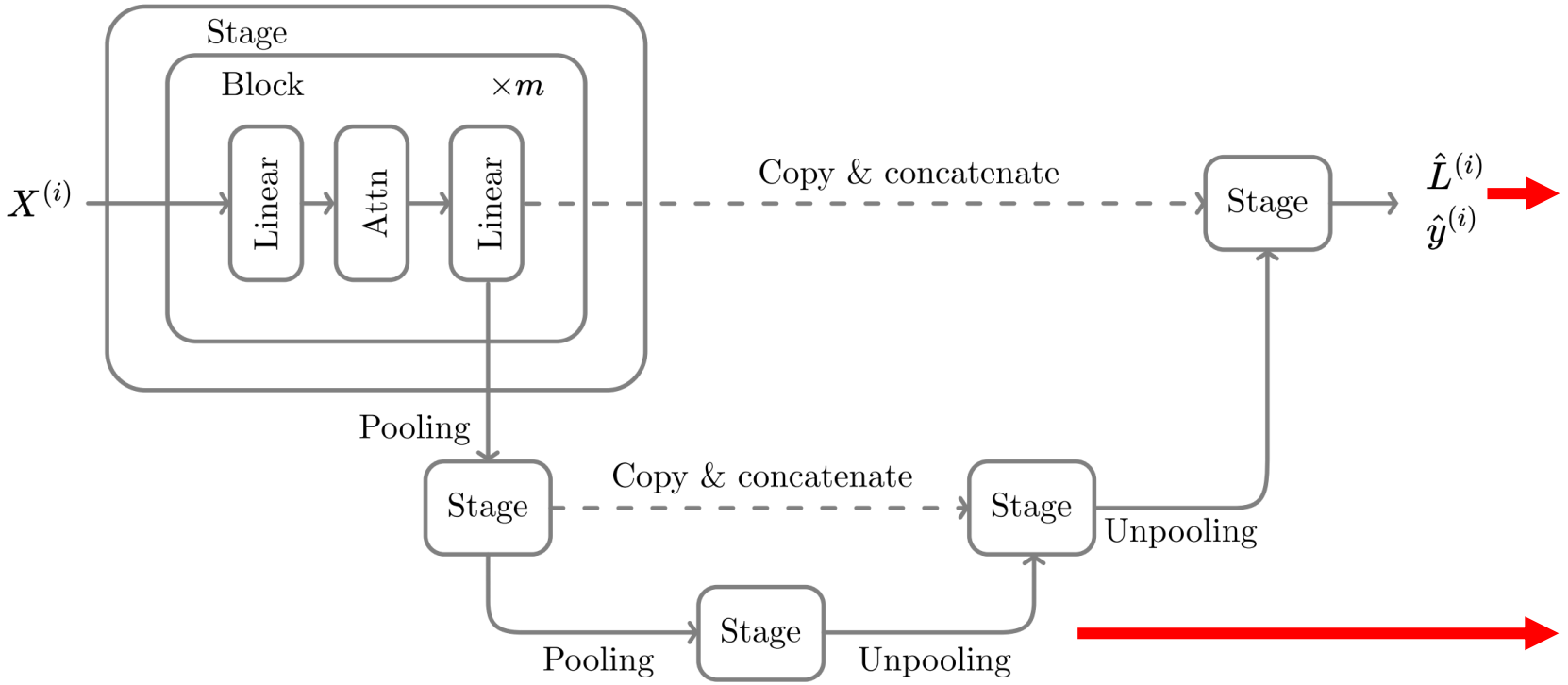
Heterogeneous PST Architecture

- **Overall architecture is U-Net-like with skip connections.**
  - In each pooling layer: create grid and average attention weights of all points in each square.
  - In each unpooling layer: copy back averaged attention back to hits originally in square.

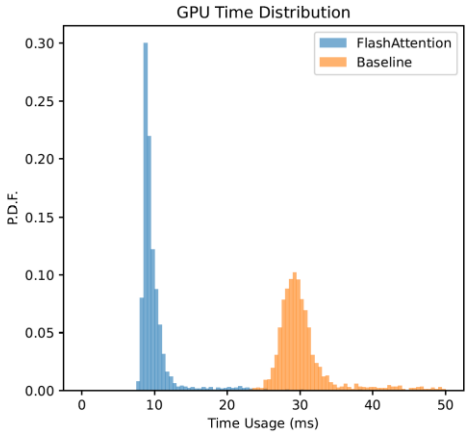


Performance Benchmark: Time Usage

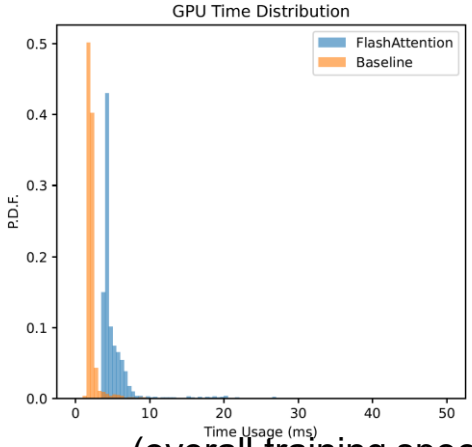
- In HPST's U-Net-like architecture, the first layer serves as the primary computational bottleneck.
- FlashAttention integration has different influences on different layers:



Overall HPST model architecture



First layer, Time usage

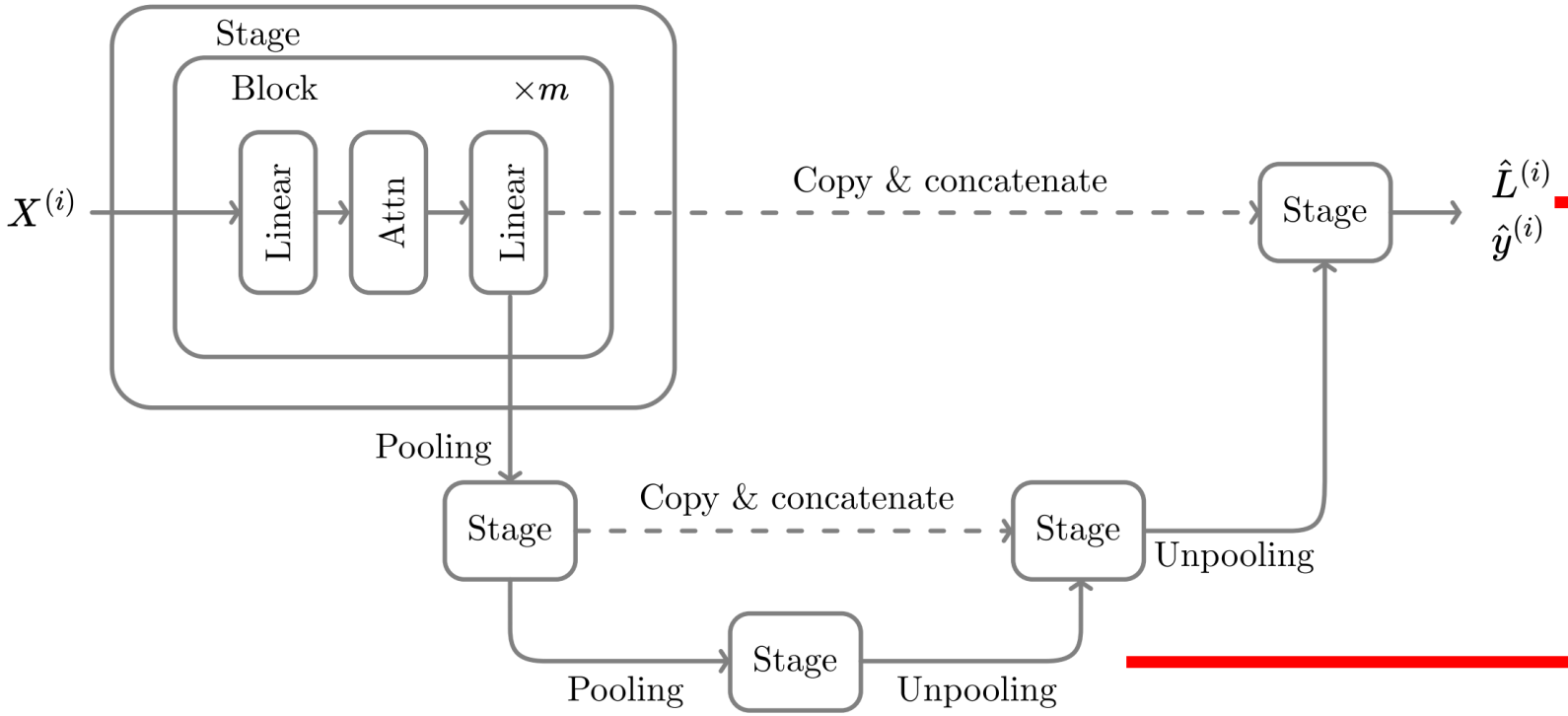


Last layer, Time usage

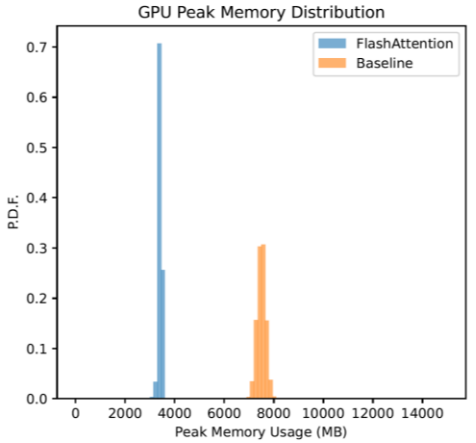
(overall training speed improved by 40%)

Performance Benchmark: Memory Usage

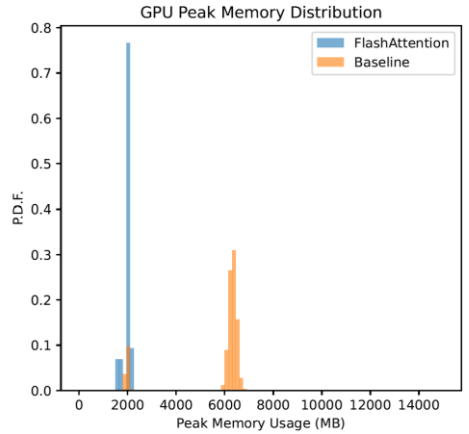
- In HPST's U-Net-like architecture, the first layer serves as the primary computational bottleneck.
- The integration of FlashAttention/RoPE has different influences on different layers:



Overall HPST model architecture



First layer, Memory usage



Last layer, Memory usage