ROI-UNets into a Transformer: ROI-UNiTer

Jake Calcutt
Local ProtoDUNE WireCell Meeting

Introduction

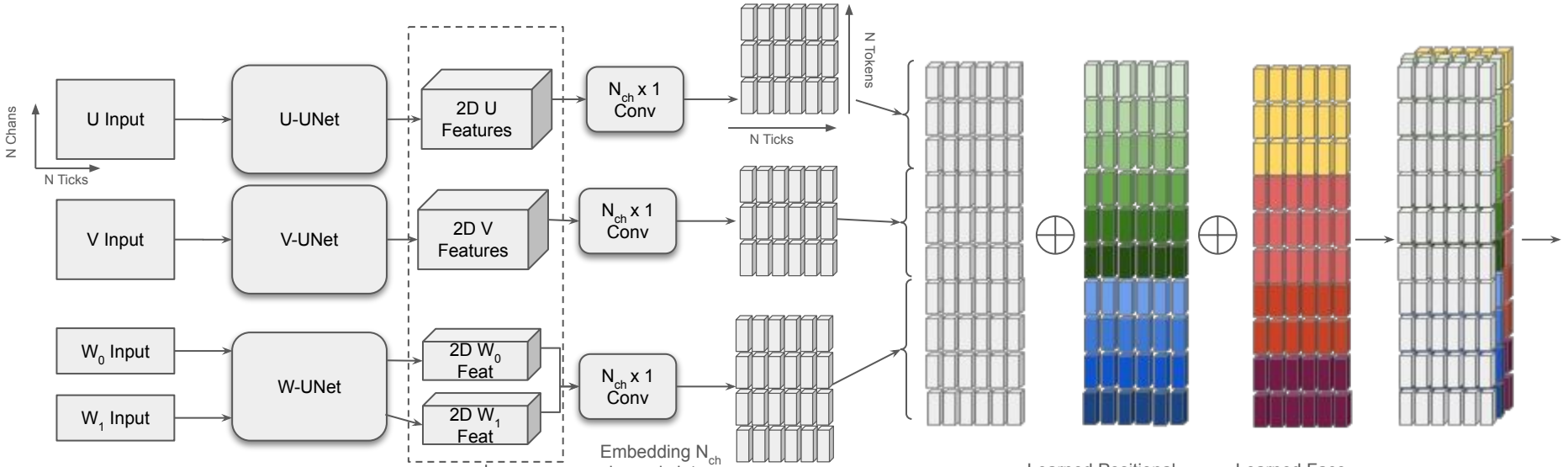
I’ve been exploring extensions/refinements of DNNROI

- Bottleneck in SPNG work is prelim ROI finding + MP2/3 creation
- I’ve made several attempts but they weren’t successful

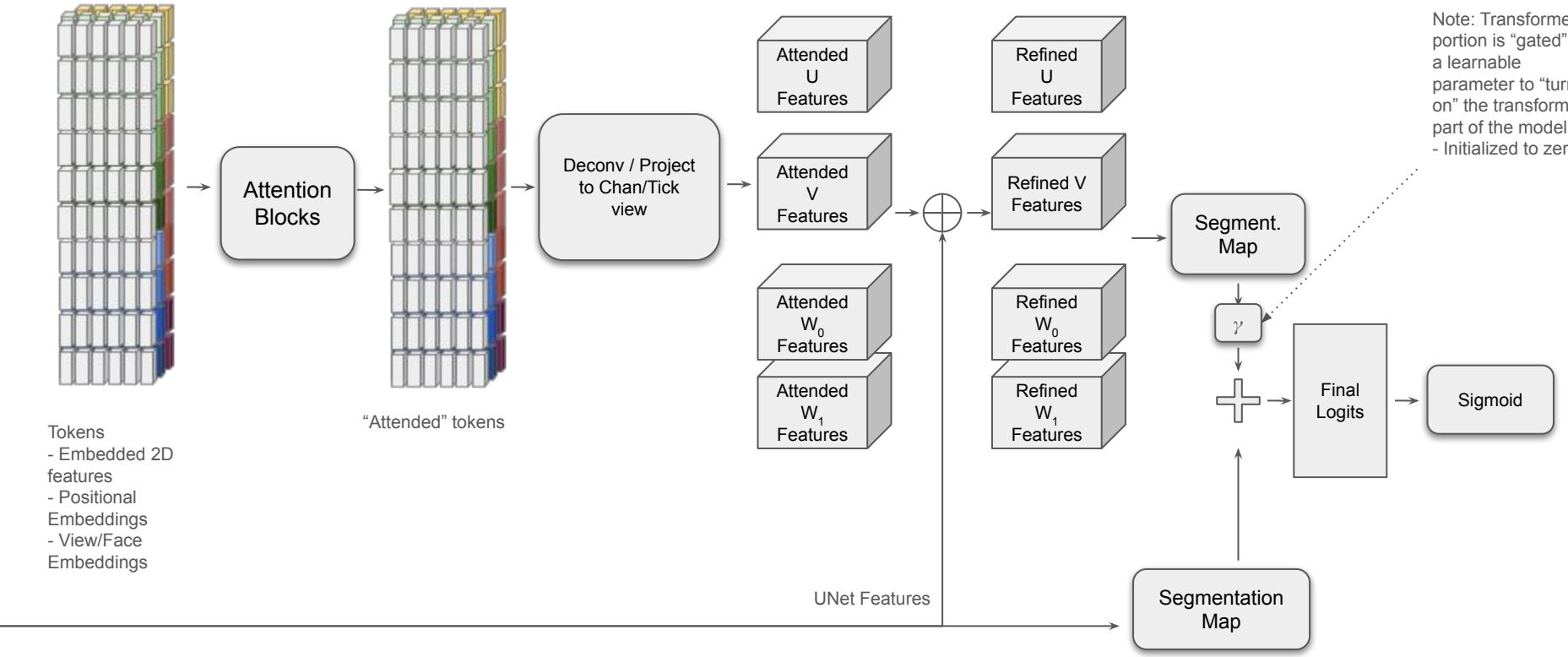
New network: ROI-UNiTer

- 3 separate UNets (w/out final segmentation layer)
  - 1 for each view type: U, V, W (separate W planes)
- Tokens
  - Embed every N channels (default 8) together separately for each tick
  - Learned Positional & (View x Face) Embeddings
- Send tokens through attention, several ticks at a time (default 3)
  - Several attention modes/styles
- Project back to chan-tick representation and combine with UNet features
- Segmentation

Model Architecture/Data Flow - 1



Model Architecture/Data Flow - 2



Attention - 1

Assume a token e_i: a vector of features (size f)

Create *queries*, *keys*, and *values* q_i, k_i, v_i from e_i with model parameters W_Q, W_K, W_V (matrices)

Expand for each token: matrices E, Q, K, V

- e_i forms cols of E, q_i forms cols of Q, etc.

$$\vec{q}_i = W_Q \cdot \vec{e}_i$$

$$\vec{k}_i = W_K \cdot \vec{e}_i$$

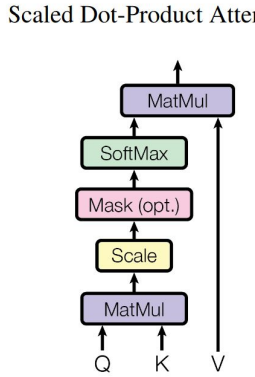
$$\vec{v}_i = W_V \cdot \vec{e}_i$$

$$Q = W_Q E$$

$$K = W_K E$$

$$V = W_V E$$

Note: practically, the W matrices are a single $d_Q \rightarrow f$ linear layer (i.e. torch.nn.Linear(d_Q, f)



Attention is All You Need

$$E \in \mathbb{R}^{f \times N}$$

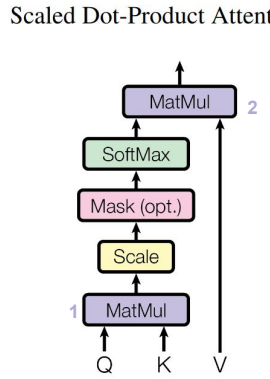
$$Q \in \mathbb{R}^{d_Q \times N} \quad W_Q \in \mathbb{R}^{d_Q \times f}$$

$$K \in \mathbb{R}^{d_K \times N} \quad W_K \in \mathbb{R}^{d_K \times f}$$

$$V \in \mathbb{R}^{d_V \times N} \quad W_V \in \mathbb{R}^{d_V \times f}$$

Attention - 2

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$



Attention is All You Need

For the multiplication to work: $d_K == d_V$
- Keys & Vals 1:1 → like a map

Each input token thus creates d_Q *queries* against the key-val map. Per transformer paper:

"The weight assigned to each value is computed by a compatibility function of the query with the corresponding key."

Softmax: $\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$.

QK^T → Compatibility matrix → Normalized by softmax

$$E \in \mathbb{R}^{f \times N}$$
$$Q \in \mathbb{R}^{d_Q \times N} \quad W_Q \in \mathbb{R}^{d_Q \times f}$$
$$K \in \mathbb{R}^{d_K \times N} \quad W_K \in \mathbb{R}^{d_K \times f}$$
$$V \in \mathbb{R}^{d_V \times N} \quad W_V \in \mathbb{R}^{d_V \times f}$$

Remember: Tokens are a set of N_{Chan} channels embedded together separately for each tick

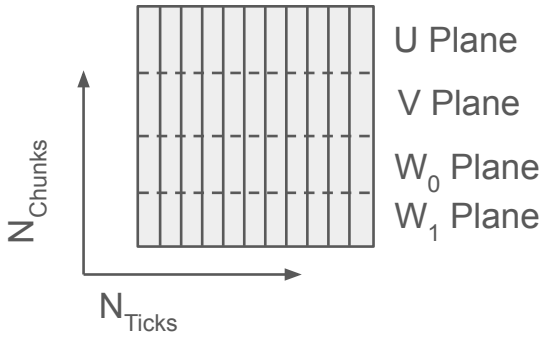
Attention in ROI-UNiTer - 1

Can't pass all ticks simultaneously

1. Memory scales as $N_{Token} \times N_{Token}$
   a. $N_{Token} = (N_{Chunks} \times N_{Ticks})$
2. Would include very-long-range correlation across time
   →Probably not useful

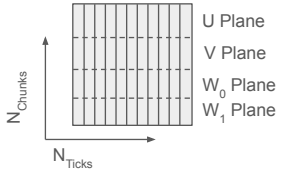
Solution: attention in bands of ticks

- Start with 1 Tick → Extend to N Ticks

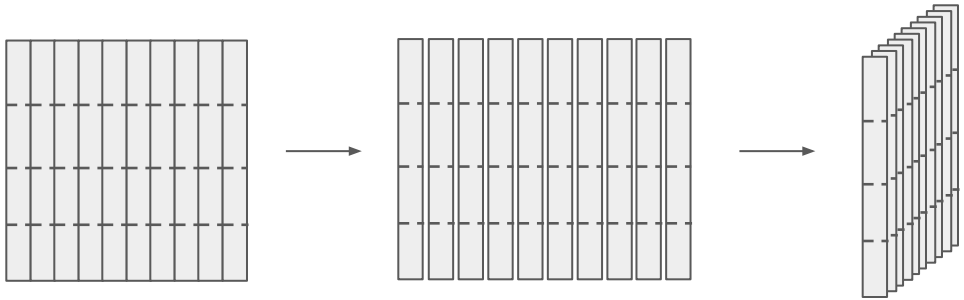


Attention in ROI-UNiTer - 2

Remember: Tokens are a set of N_{Chan} channels embedded together separately for each tick



Assume a band of width 1



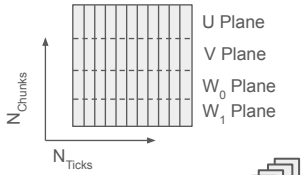
Put ticks as batch
→No cross-tick communication

I'll extend to tick bands > 1 tick later

Let's talk about attention "modes"

Attention in ROI-UNiTer - 3: Modes

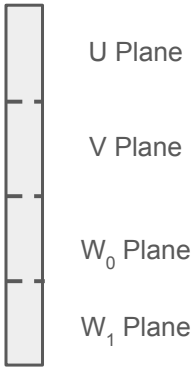
Remember: Tokens are a set of N_{Chan} channels embedded together separately for each tick



Consider one tick for now (consistent per tick-batch)

What do we want to communicate with each other?

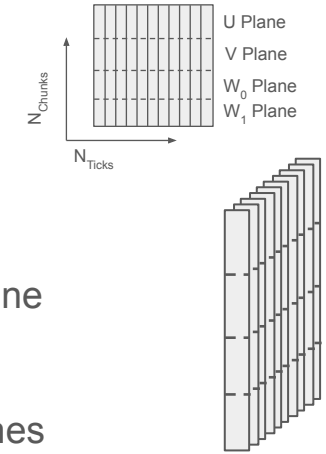
- Consider different “modes”
  - Useful for optimization/ablation studies



Ablation study: turning off/on specific parts of model to show their utility (or lack thereof)

Attention in ROI-UNiTer - 3: Modes (cont.)

Remember: Tokens are a set of N_{Chan} channels embedded together separately for each tick



Key

Query	U	V	W_0	W_1
U	✓✓✓	✓✓✓	✓✓✓	✓✓✓
V	✓✓✓	✓✓✓	✓✓✓	✓✓✓
W_0	✓✓✓	✓✓✓	✓✓✓	✓
W_1	✓✓✓	✓✓✓	✓	✓✓✓

1. Intra – Only sees within a plane
   a. No cross-plane sharing
   b. No cross-W
2. Inter – Only sees across planes
   a. No within-plane sharing
   b. No cross-W
3. All – Inter & Intra
   a. No cross-W
4. Full – No restriction

Q1: Does cross-plane information help? (Intra vs others)
Q2: Does in-plane information help? (Inter vs others)
Q3: Does sharing info across W planes hurt? (Full vs All)
Q4: What are the consequences on processing time?