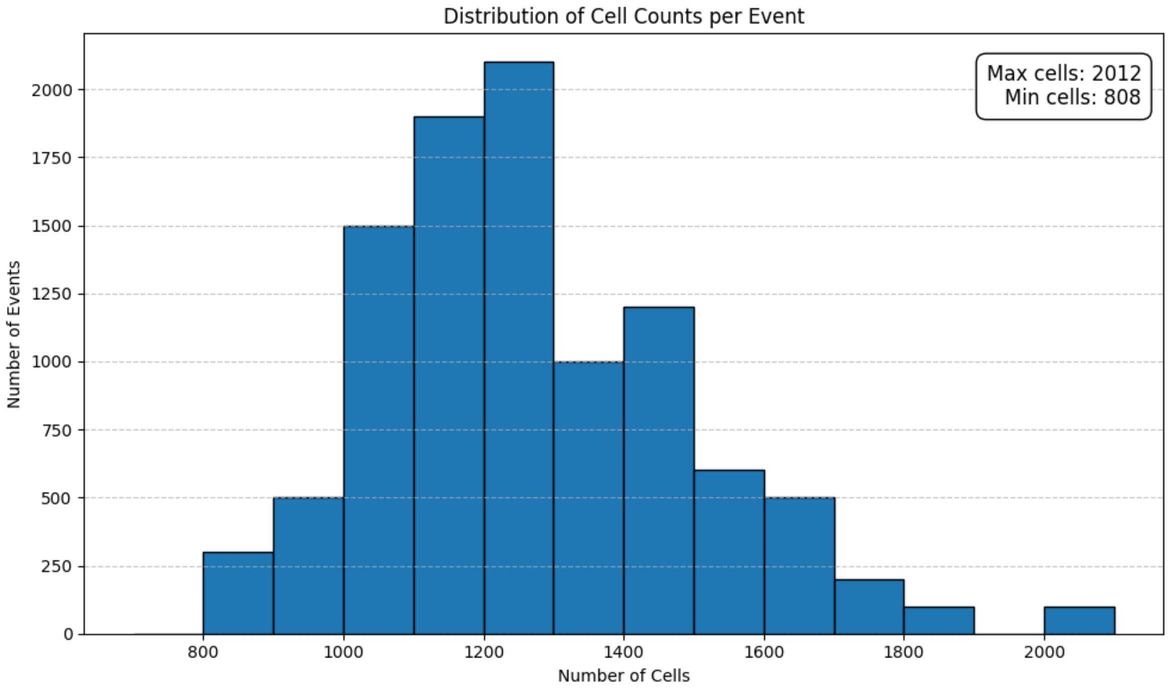
Generative Model for CEPC Detector Simulation

- Dataset
- Model Construction
- Loss Function
- Parameter Setting and Optimization
- Training Result
- Evaluation
- Future Work

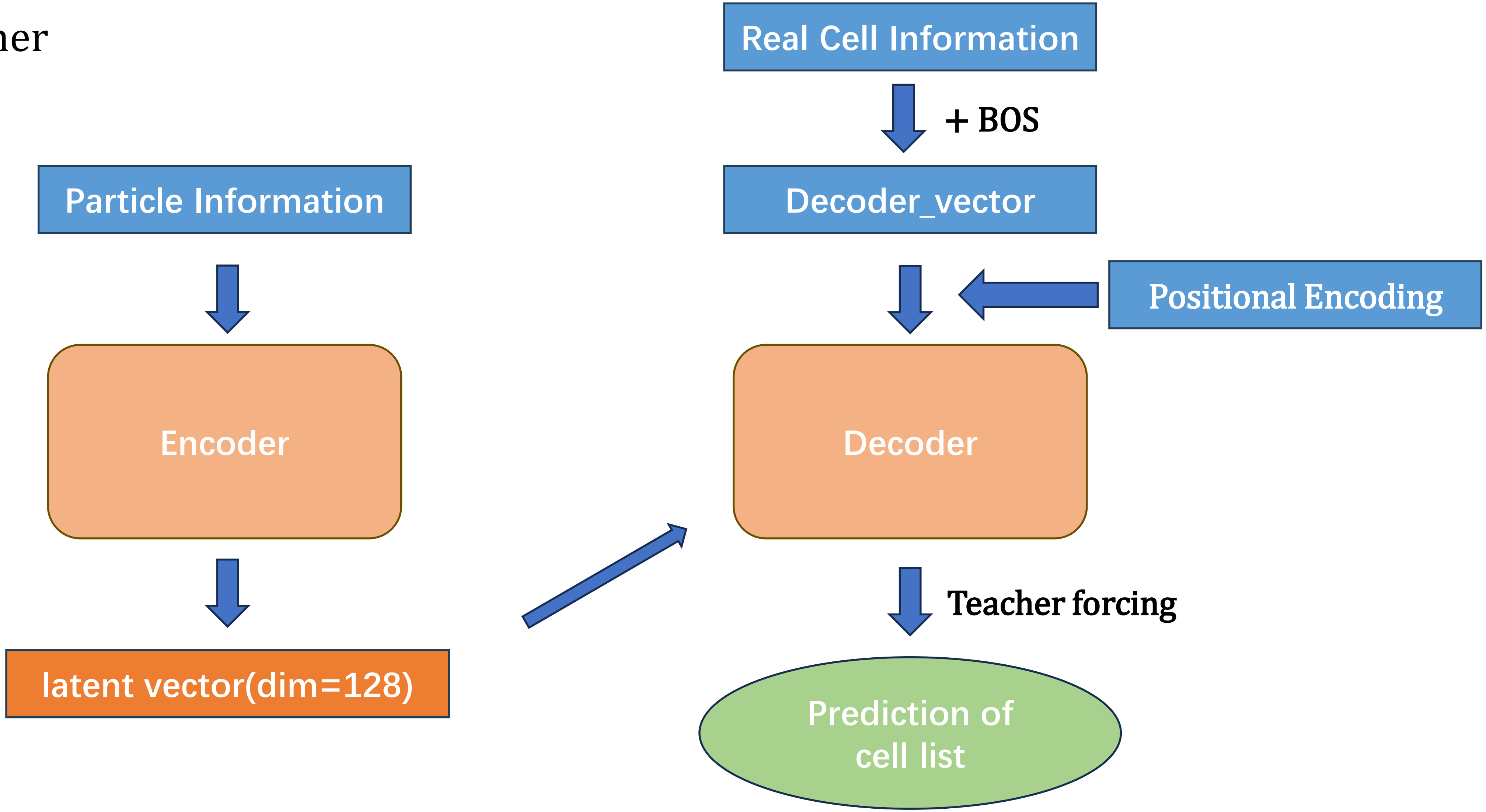
Dataset

- The dataset include 10000 events of $\gamma(E\epsilon[30,50], \theta\epsilon[73.6,106.4])$
- Every event includes $p_4\ of\ \gamma$ and a cell list.
- The number of the cells of every event is almost in [800,2100]



Model Construction

- Transformer



About teacher forcing:

Teacher forcing means that during training, the model generates each step based on the ground truth cells, rather than relying on the cells it has already generated.

The BOS means “the beginning of sequence”. A cell is represented as a 5-dimensional vector: energy, x, y, z, EOS, where EOS is a binary flag (0/1) indicating whether to continue or stop generation. BOS token is an initial vector represented as 0,0,0,0,0.

Step	Input	Prediction
1	BOS	Gen_cell 1
2	BOS, Real_cell 1	Gen_cell 2
...	...	...
n	BOS, Real_cell 1 ... Real_cell n-1	Gen_cell n-1

Loss Function

- **Loss of JS(x,y,z,energy)**: Measures the similarity between prediction and target.

$$\text{js\_loss} = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{4}\sum_{d=1}^{4}\text{JS}(\text{hist\_pred}_{b,d}, \text{hist\_target}_{b,d})$$

- **Loss of Total Energy**: MSE between the total predicted energy and the ground truth total energy.

$$\text{energy\_loss} = \frac{1}{B}\sum_{b=1}^{B}(\text{pred\_E}_{\text{sum},b} - \text{target\_E}_{\text{sum},b})^2$$

- **Loss of Barycenter**: Distance between the barycenter of prediction and target.

$$\text{center\_loss} = \frac{1}{B}\sum_{b=1}^{B}\text{MSE}(\text{pred\_center}_b, \text{target\_center}_b)$$

- **Loss of EOS**: Make the model know when to end the generation.

$$\text{eos\_loss} = \frac{1}{B}\sum_{b=1}^{B}\text{BCE}(\text{pred\_eos}_b, \text{target\_eos}_b, \text{weights}_b)$$

EOS Loss with Binary Cross Entropy (BCE)

For each generated cell, the model outputs a raw **logit** as the initial EOS value. This **logit** can be converted into a stop probability **p** using the sigmoid function. If $\mathbf{p < 0.5}$, then $\mathbf{EOS = 0}$; if $\mathbf{p > 0.5}$, then $\mathbf{EOS = 1}$.

The EOS loss is calculated using Binary Cross Entropy (BCE), which measures the difference between the predicted probability and the ground truth label.

$$\boldsymbol{BCE(p, y) = -[y * \log(p) + (1 - y) * \log(1 - p)]}$$

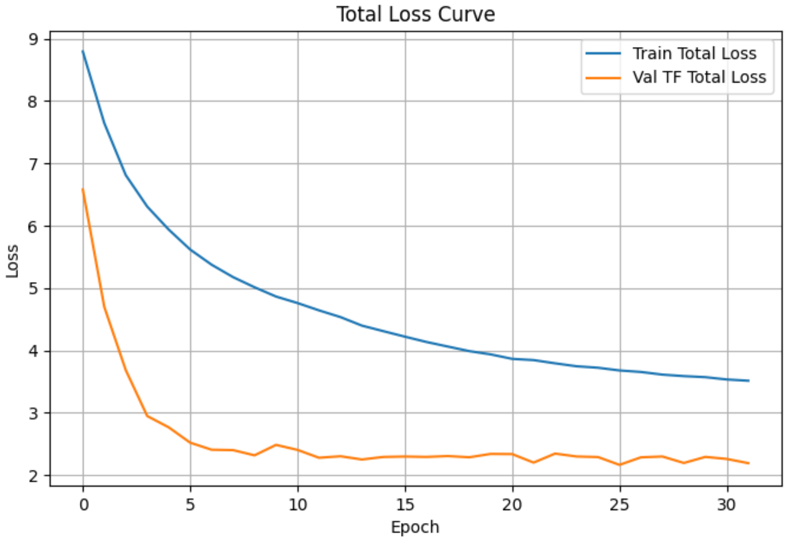
where the y is the EOS(0/1) in the real cell.

Parameter Setting and Optimization

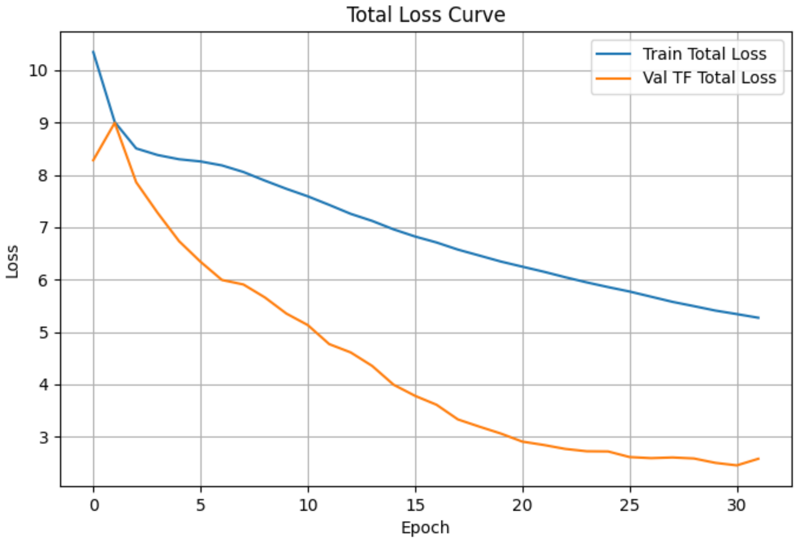
- Epoch and Batch_size(32&64 64&32)
- Dimension of the Embedding Latent Vector (dim= 128)
- Number of Heads($N_{head} = 8/4$)
- Number of Layers($N_{layer} = 5/4$)
- Loss Weights($Weight\ of\ JS\ loss\ is$ 2, others are all 1)
- Learning Rate($lr = 10^{-4}$)
- Max Sequence Length($Max_{sequence} = 512$)

Training Results

This training uses loss of JS, loss of centroid and loss of EOS but **without loss of total energy** because there is some instability needed to be find out.



$$Epoch = 32 \quad batch\_size = 64$$
$$N_{head} = 4 \quad N_{layer} = 4$$



$$Epoch = 32 \quad batch\_size = 32$$
$$N_{head} = 4 \quad N_{layer} = 5$$

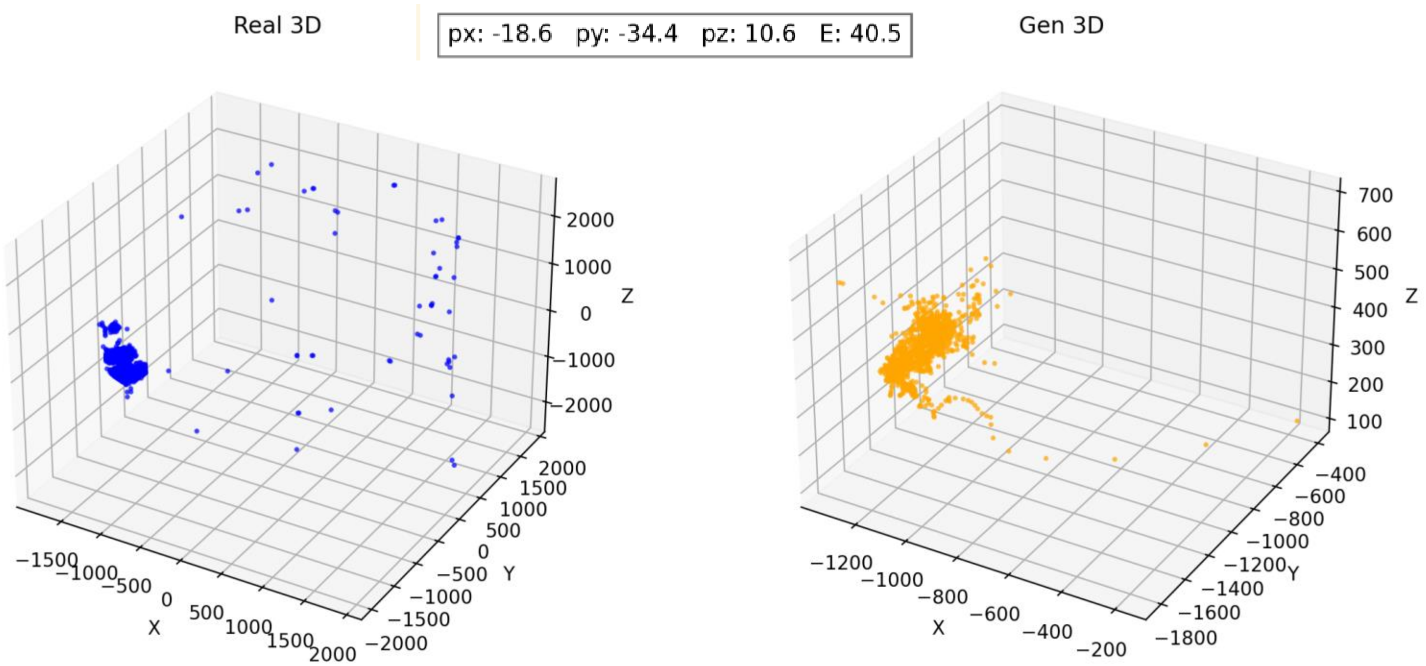
We can observe that the performance on the validation set is consistently better than on the training set. This is because both training and validation use the teacher forcing strategy. Additionally, during training, the data is shuffled within each batch, which may introduce some noise. Moreover, a dropout rate of 0.1 is applied during training, whereas dropout is disabled during validation.

Evaluation

- Visual Evaluation: 2D R-φ and R-z plot, also show a 3D cluster.
- Comparison of the distribution
- Barycenter of the cluster
- Total energy
- Resolution of the total energy

For a single event, we input the momentum and energy and generate the cell list. Here are some comparisons.

Comparison of the cluster(3D plot):



Barycenter:

Real:[-973.0, 1662.8, 461.1]

Gen:[-825.2, -1451.8, 255.8]

Cell Count:

Real: 1093, Gen: 1147

Total Energy :

Real: 37.0530, Gen: 43.6340

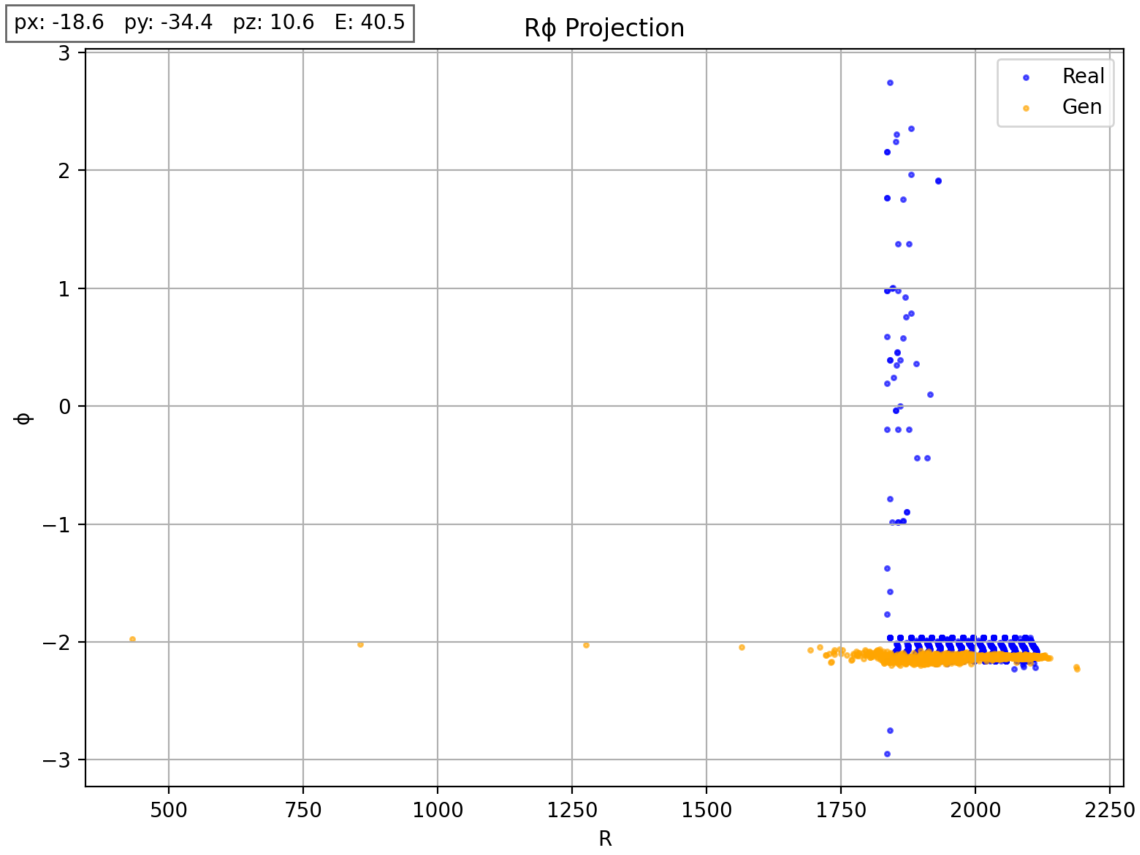
Resolution of energy :

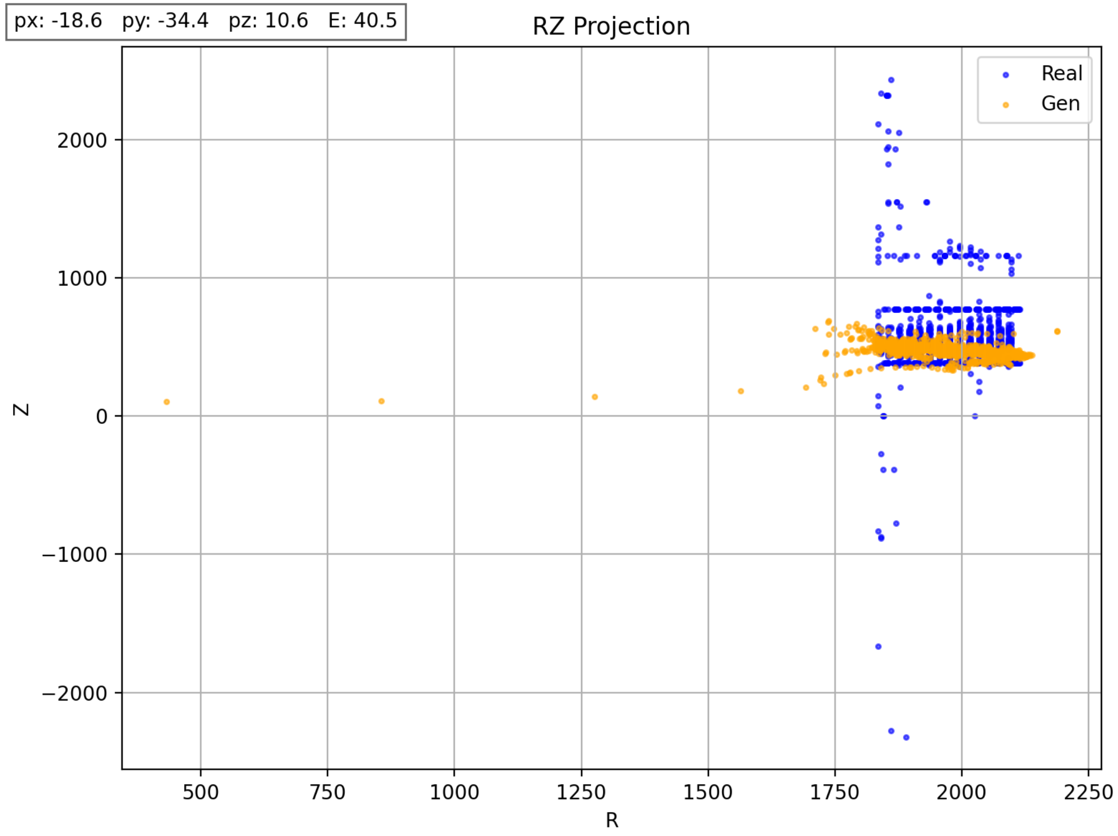
Real: -7.48%, Gen: 8.95%

Evaluation

For a single event, we input the momentum and energy and generate the cell list. Here are some comparisons.

Comparison of the cluster(projection in R-φ and R-Z):

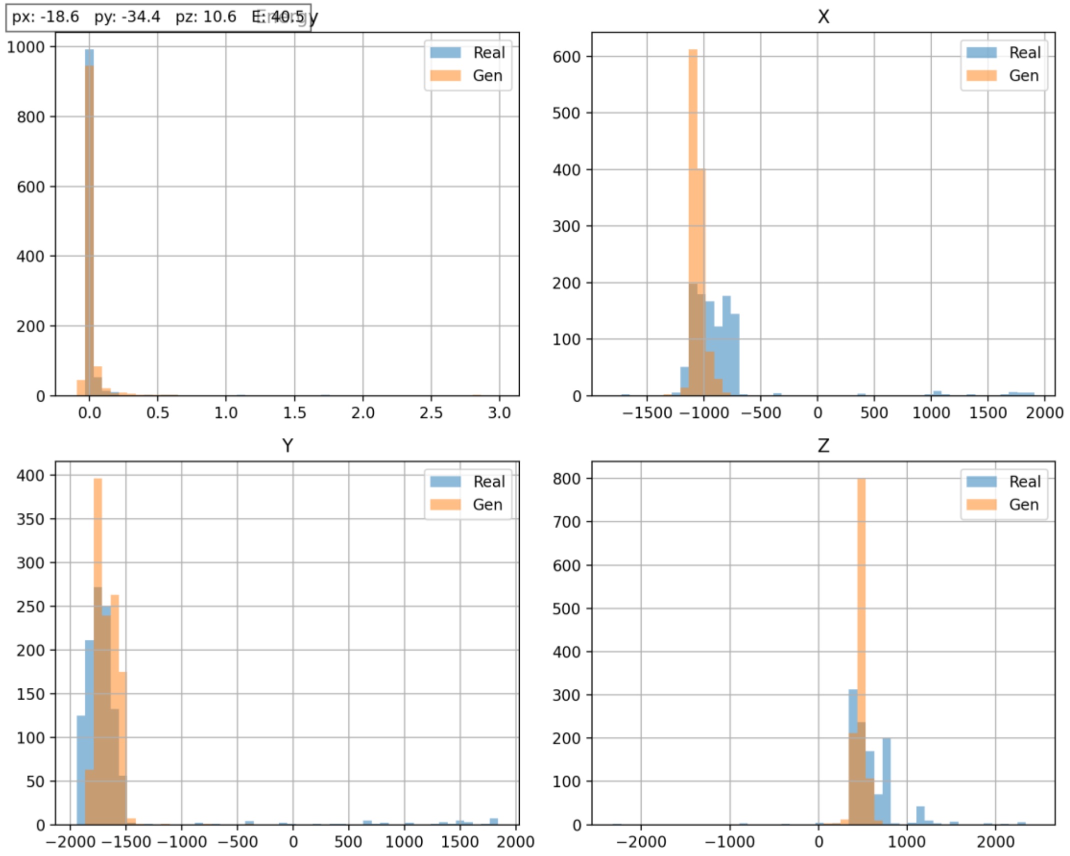




Evaluation

For a single event, we input the momentum and energy and generate the cell list. Here are some comparisons.

Comparison of the distribution of energy, x, y and z:



- We can see that the overall distribution fitting is still not ideal, which is related to the fact that we did not make any constraints on the detector structure.
- The energy distribution is highly peaked due to the large number of low-energy splashes. However, we observed that some of the generated energies are negative, which suggests that the current activation function may need to be revised.
- Also, the orange area is consistently larger than the blue area, indicating that the number of generated cells is always greater than the number of real cells. This is because the model hasn't effectively learned when to stop.

Future Work

- How to improve the quality of the model's generation
- How to teach the model when to stop ($EOS\_loss, Energy\_loss$)
- How to input the shape of the cluster into the model
- How to take the detector construction into account and incorporate it into the model's generation constraints

Future Work

About EOS_loss

Traditional EOS_loss: Only the last cell is assigned EOS=1 to indicate the end of generation. However, since the cell sequence is very long, the proportion of EOS=1 is extremely low, making little impact on training.

cell	cell 1	cell 2	cell 3	…	cell n-1	cell n
EOS	0	0	0	0	0	1

The improvement of EOS_loss:

- Increase the weight of the EOS loss.
- Assign EOS=1 to low-energy sequences at the end of the cell list to increase the EOS proportion.

cell	cell 1	…	cell t	cell t+1	…	cell n
EOS	0	0	1	1	1	1

Thank you for listening