LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Improved Selective Background Monte Carlo Simulation at Belle II with Graph Attention Networks and Weighted Events

Boyang Yu [1], Nikolai Hartmann [1], Thomas Kuhr [1]
[1] *Ludwig-Maximilians-Universität München*
IDT-UM Collaboration Meeting, Feb 15th, 2022

ErUM-Data IDT

Bundesministerium für Bildung und Forschung

LMU TUM ORIGINS Excellence Cluster

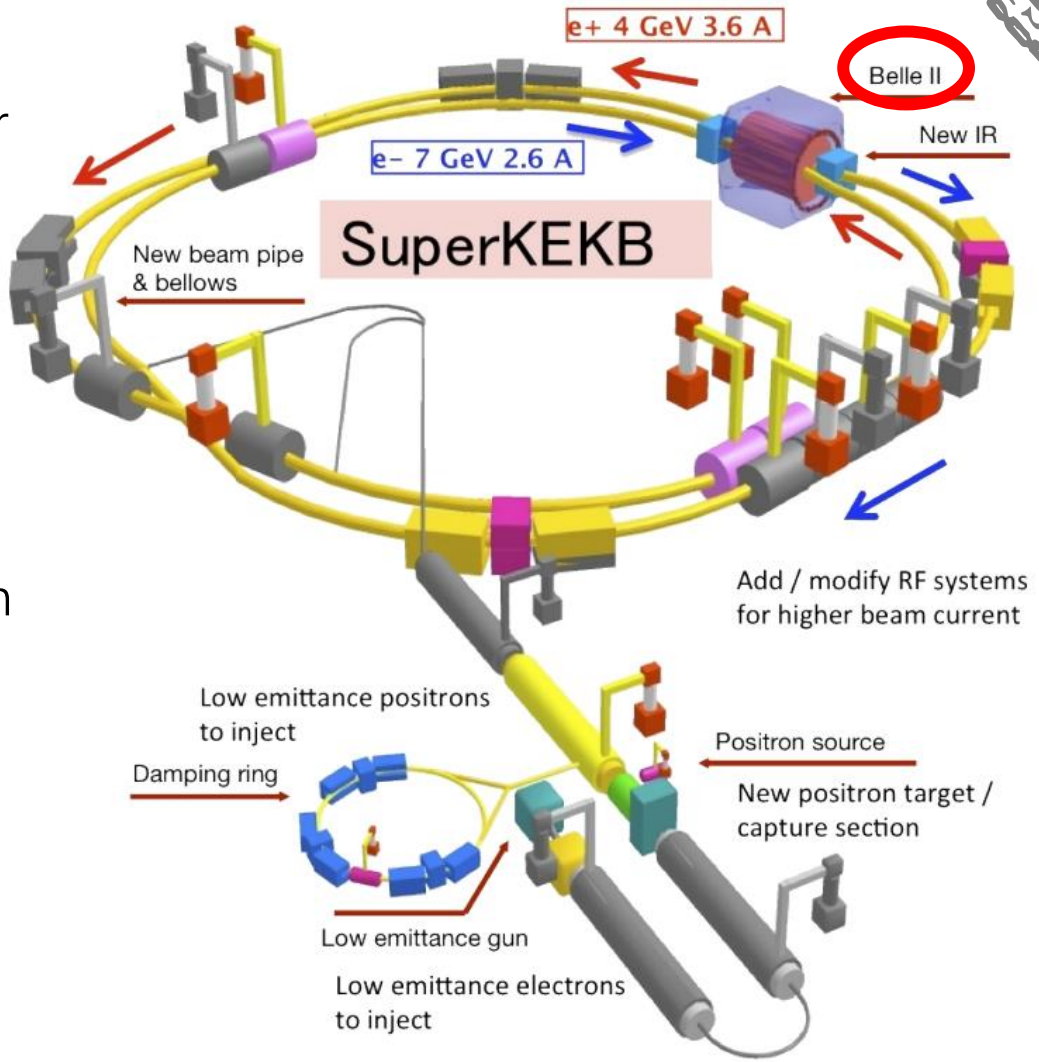
Belle II

SuperKEKB accelerator:

- Asymmetric energy electron positron collider at $\sqrt{s} = 10.6$ GeV
- Production of B meson pairs
- World´s highest luminosity reached end of 2021: $3.8 \times 10^{34}\ \mathrm{cm}^{-2}\ \mathrm{s}^{-1}$

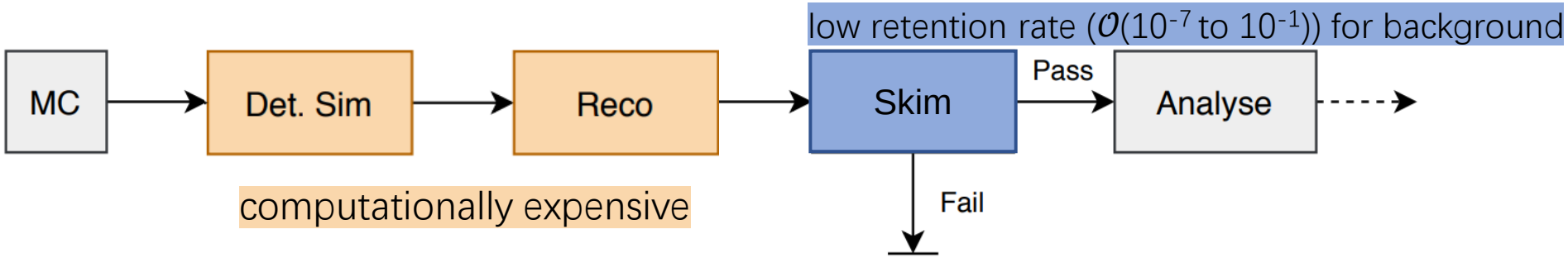
Belle II experiment:

- Detector for reconstruction and identification of charged and neutral particles
- Started taking physics data in 2019
- Designed to collect 50 times more data than Belle

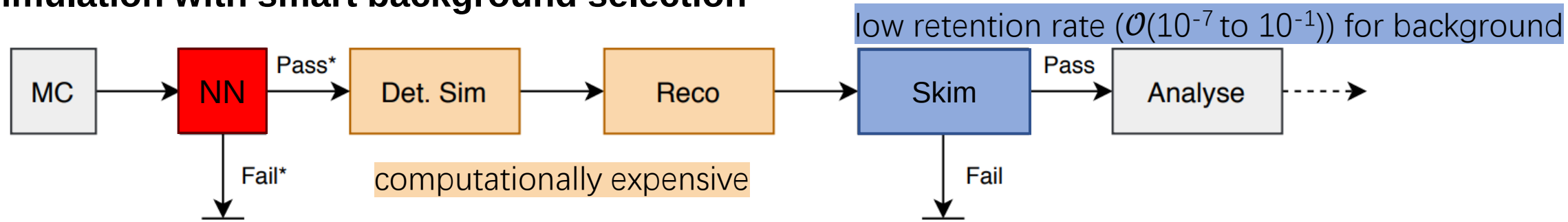


Motivations

Current Monte Carlo Simulation data flow



Simulation with smart background selection



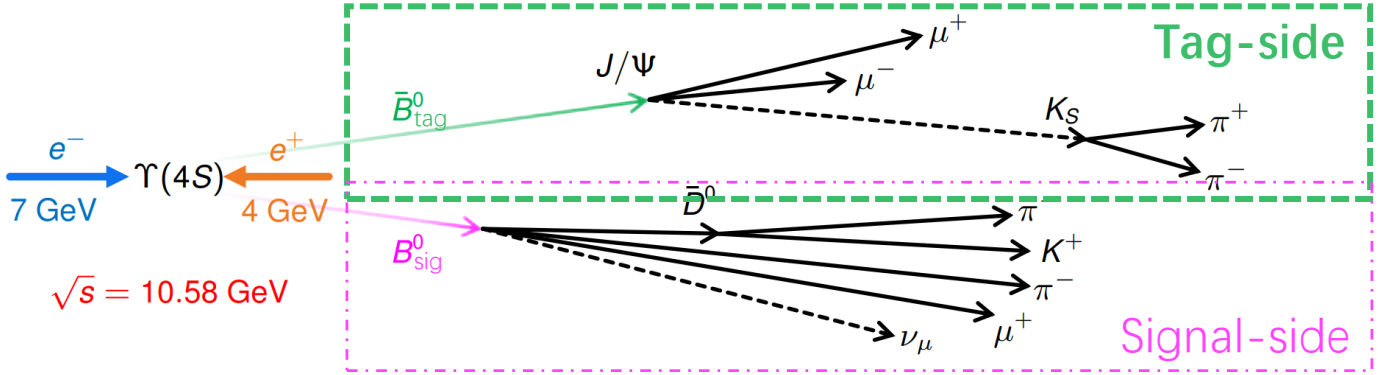
Previous Works:

- **PhD Thesis**: *Hadronic Tag Sensitivity Study of $B \to K^{(*)}\nu\bar{\nu}$ and Selective Background Monte Carlo Simulation at Belle II*, James Kahn, 2019
- **Talk**: *Selective background Monte Carlo simulation at Belle II*, James Kahn, CHEP 2019

Improved Selective Background Monte Carlo Simulation at Belle II with Graph Attention Networks and Weighted Events

Motivations

Tagging method:

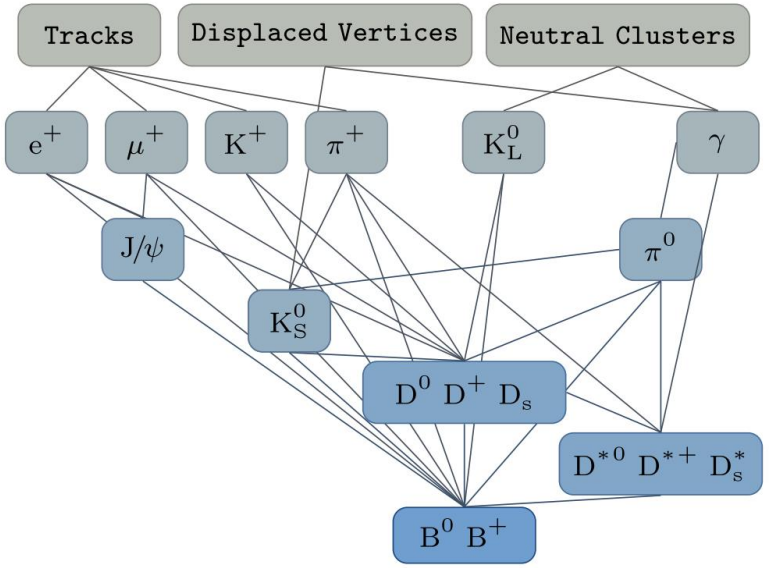


Process of FEI skim:

Events are kept if at least one of $\mathcal{O}(1000)$ considered decay chains is reconstructed.

Full Event Interpretation (FEI):

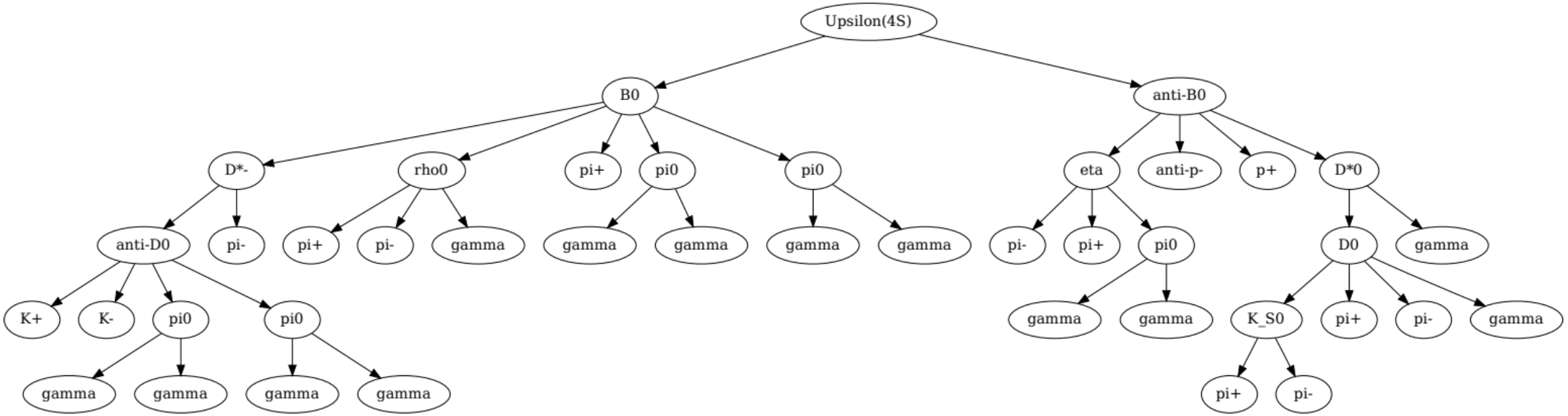
Hierarchically reconstruct from **tracks and clusters** to **intermediate particles** and finally to **B mesons**

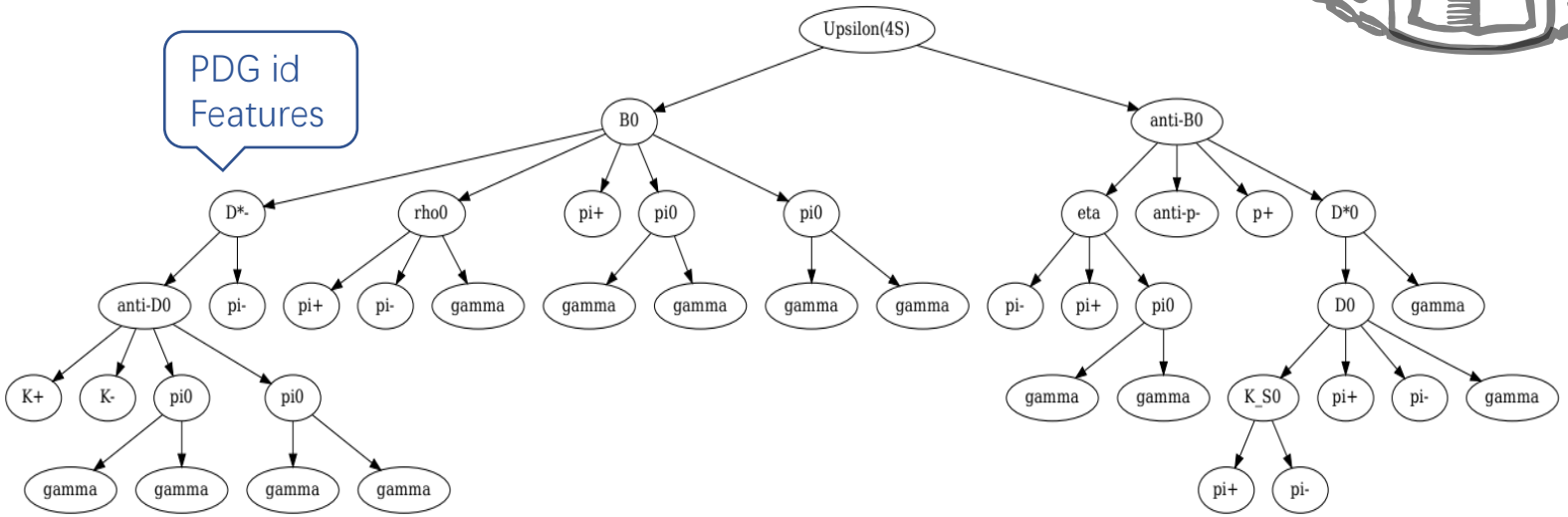


Retention rate after FEI skim:

	Hadronic B^+	Hadronic B^0
Mixed ($\Upsilon(4s) \to B^0\bar{B}^0$)	5.62%	**4.25%**

Tree Structures of Particle Decay inspire the use of Graph Neural Network



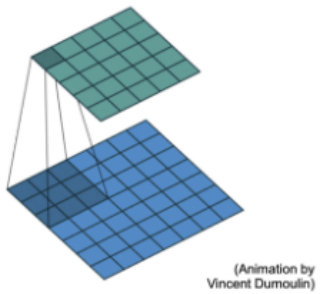


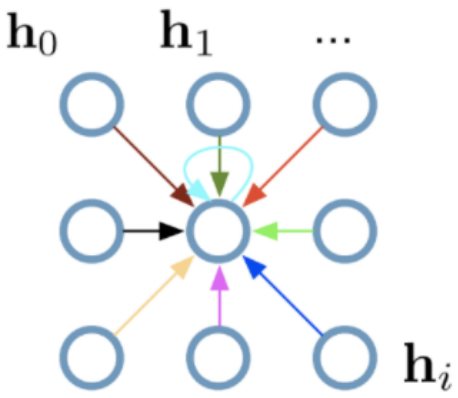
Dataset:

- Each event (each **Graph**):
  - ➢ Decay of $\Upsilon(4S) \rightarrow B^0\bar{B}^0$
  - ➢ 30-80 Particles (**Nodes**)
  - ➢ Decay relations (two way **Edges** + self loops)
  - ☐ Each particle (each **Node**)
    - ➢ **PDG** ids
    - ➢ 8 **Features**: **Production time**, **Energy**, **Position** (3d), **Momentum** (3d)
- **Labels** per event: Pass/Fail after the reconstruction of B decays (FEI skims)
- Other event level **attributions** for further analysis: e.g. $\boldsymbol{M_{bc}}$, $\boldsymbol{\Delta E}$, etc.

CNN vs GCN

Single CNN layer with 3x3 filter:



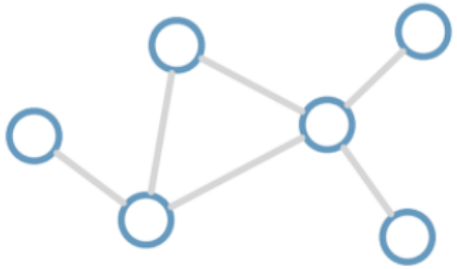


Full update:

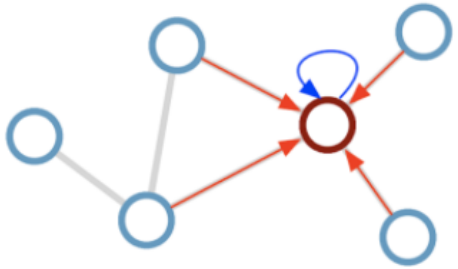
$$\mathbf{h}_4^{(l+1)} = \sigma\left(\mathbf{W}_0^{(l)}\mathbf{h}_0^{(l)} + \mathbf{W}_1^{(l)}\mathbf{h}_1^{(l)} + \cdots + \mathbf{W}_8^{(l)}\mathbf{h}_8^{(l)}\right)$$

Convolutional Neural Network (CNN)

Consider this undirected graph:



Calculate update for node in red:



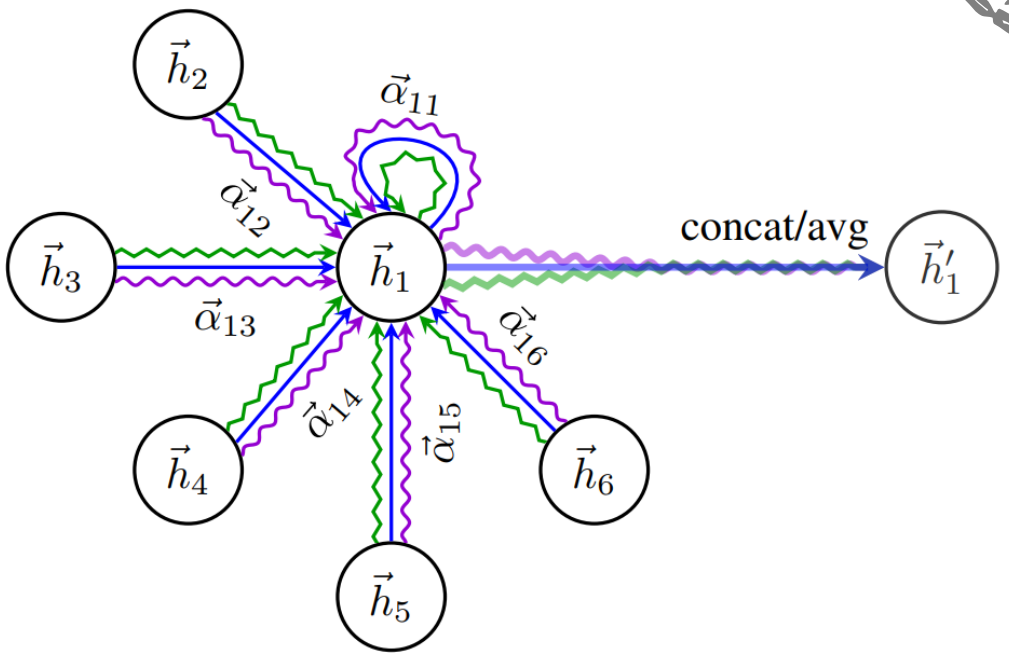
Update rule:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{h}_i^{(l)}\mathbf{W}_0^{(l)} + \sum_{j\in\mathcal{N}_i}\frac{1}{c_{ij}}\mathbf{h}_j^{(l)}\mathbf{W}_j^{(l)}\right)$$

Graph Convolutional Network (GCN)

Graph Attention Networks (GAT)

- Learn attention weights $\boldsymbol{\alpha}^{\boldsymbol{k}}$ of edges from features of adjacent nodes
- Allow n_{heads} = K different heads
- Concatenate outputs from all heads

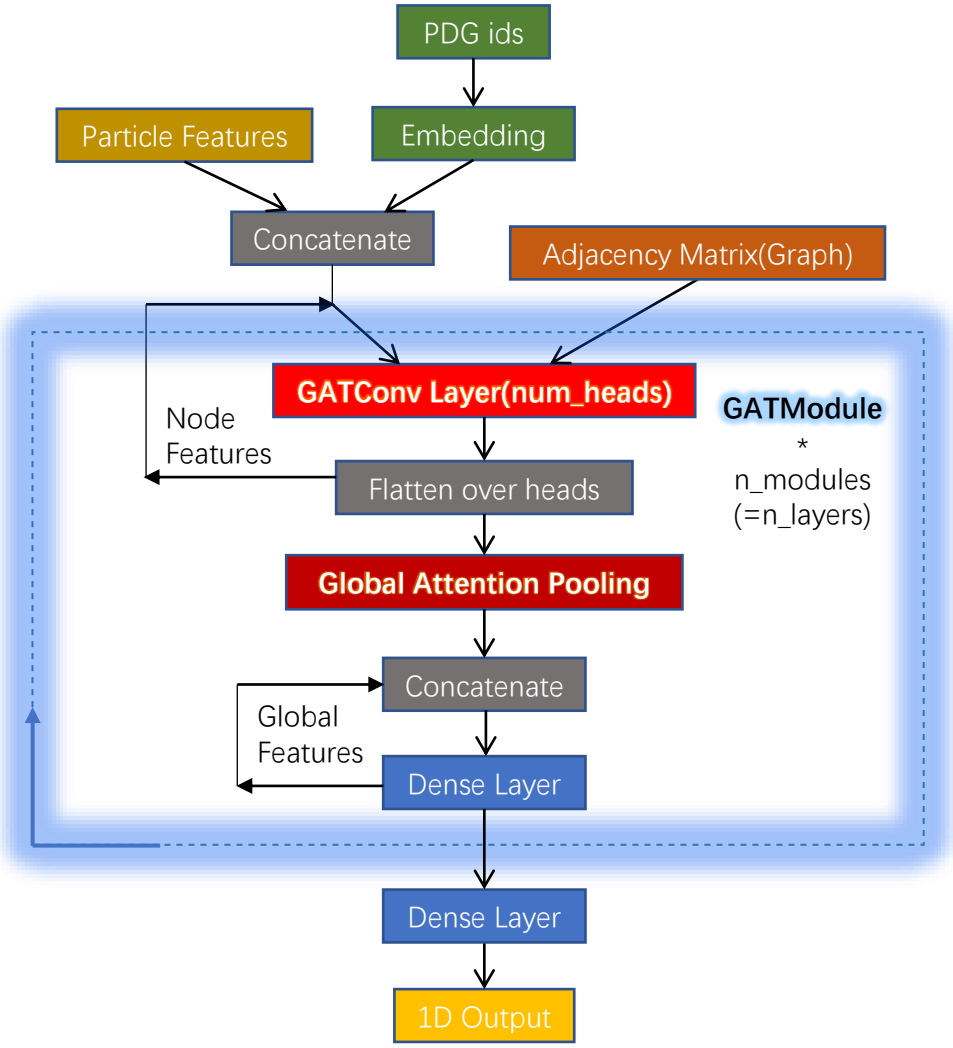


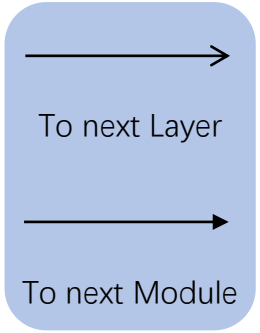
$$\vec{h}'_i = \Big\|_{k=1}^{K} \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^k \mathbf{W}^k \vec{h}_j\right)$$

Graph Attention Pooling (GAP)

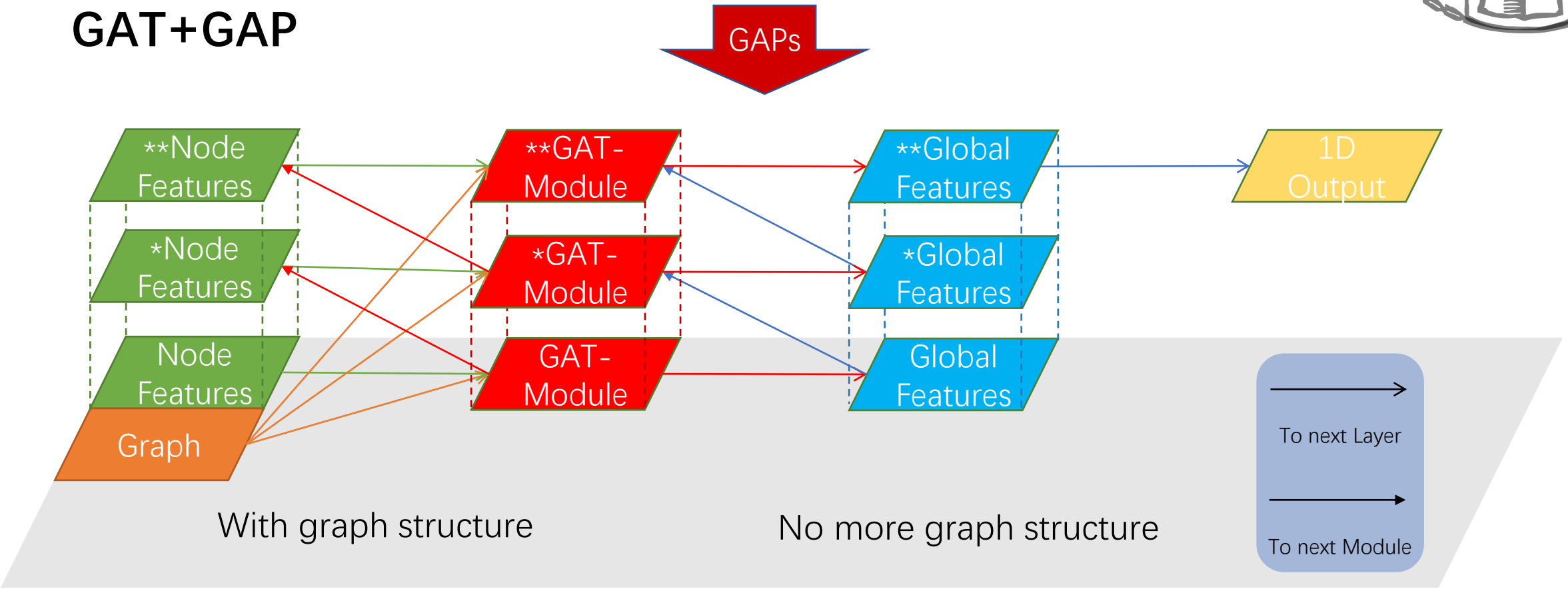
- Learn weights for aggregation into global features from node features

Final Architecture: GAT+GAP

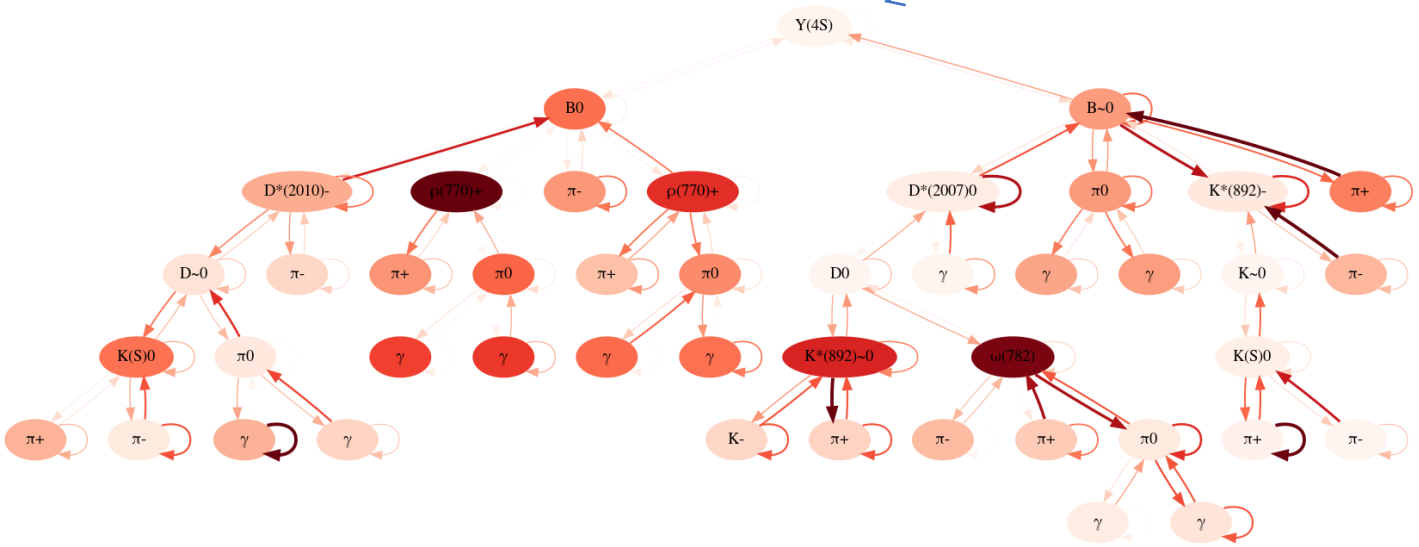


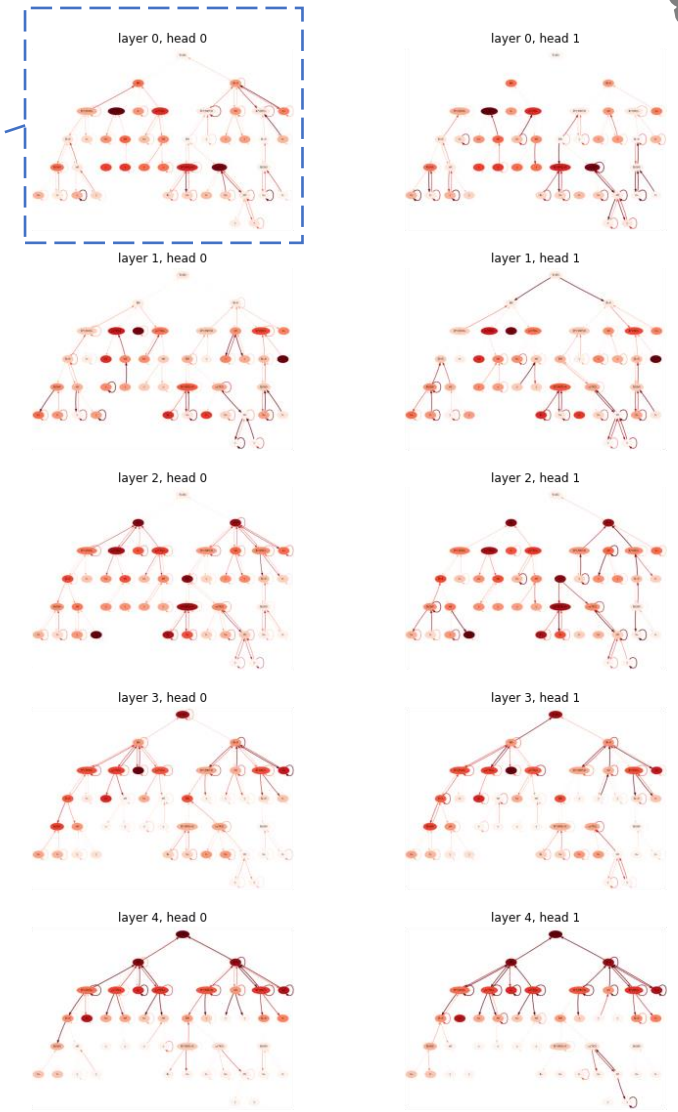


Final Architecture: GAT+GAP



Visualization of Node/Edge-Attentions for GAT+GAP





Heads

Modules

Hyperparameter Optimization

Model	AUC
GCN(sep)	0.9083
GAT(sep)	0.9094
GAT(gen)	0.9089
GATGAP(gen)	0.9122

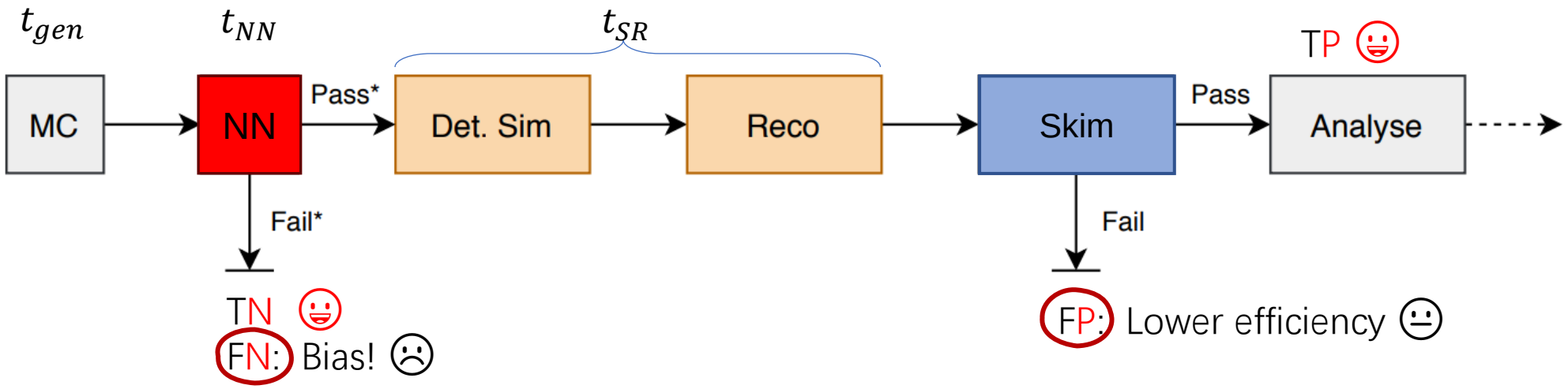
Batch Size	Number of Units	AUC	Training Time in s
128	16	0.9131	10940
512	32	0.9117	3568
128	128	0.9117	5205
1024	32	0.9115	1716
512	128	0.9115	2228
256	128	0.9115	2666
256	32	0.9115	4061

Number of Units	Number of Parameters
16	34,911
32	120,527
64	459,951
128	1,808,495

Final Configuration:

- GATGAP Model using PyTorch + Deep Graph Library (DGL)
- 6 GATModules with 4 attention heads each and 32 units for GAT output & global features
  -> ≈ 120k parameters
- Batch size 1024 (GPU training)

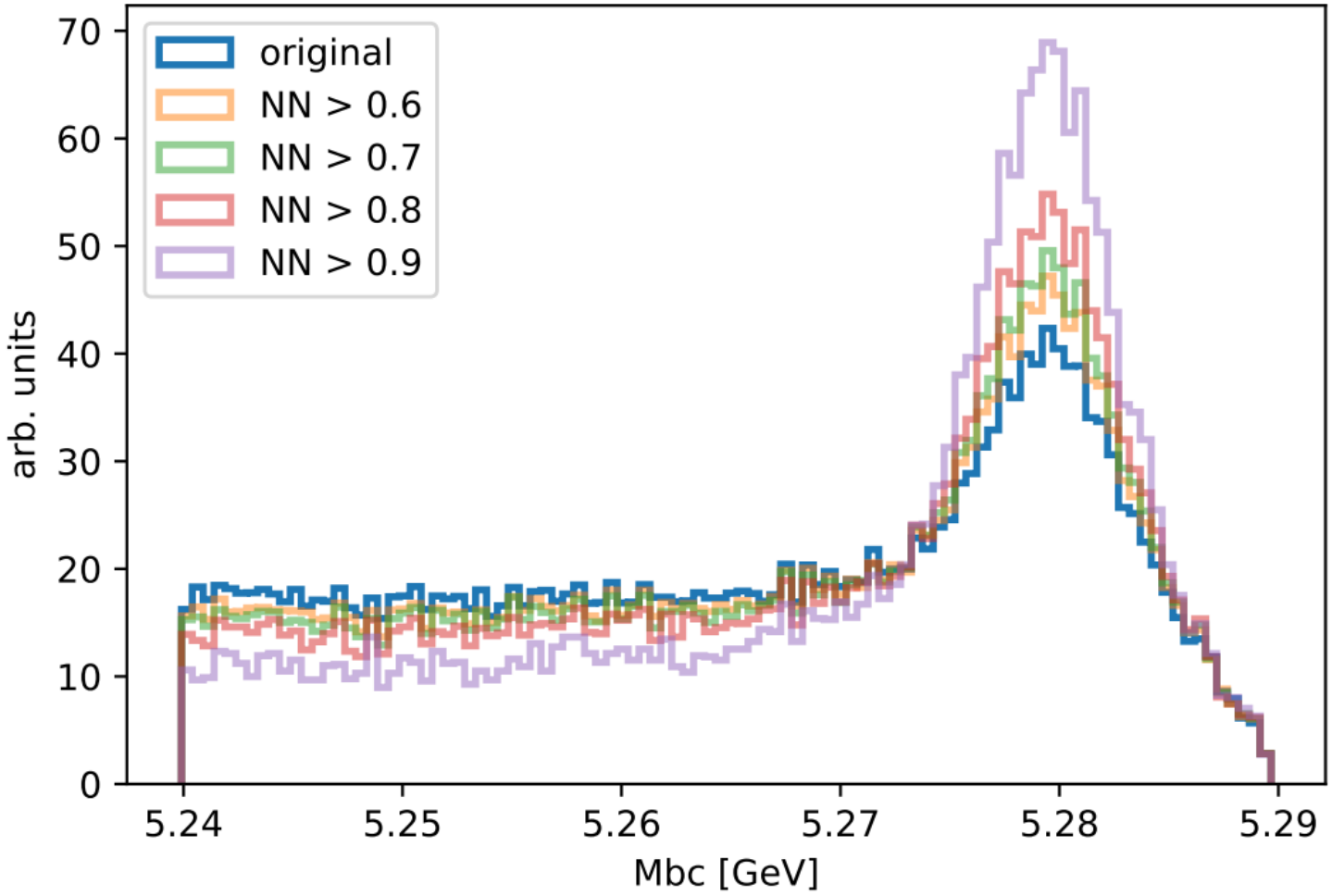
Implementation of NN filters



Skim \ NN	Positive	Negative
Pass	True-Positive (TP)	False-Negative (FN)
Fail	False-Positive (FP)	True-Negative (TN)

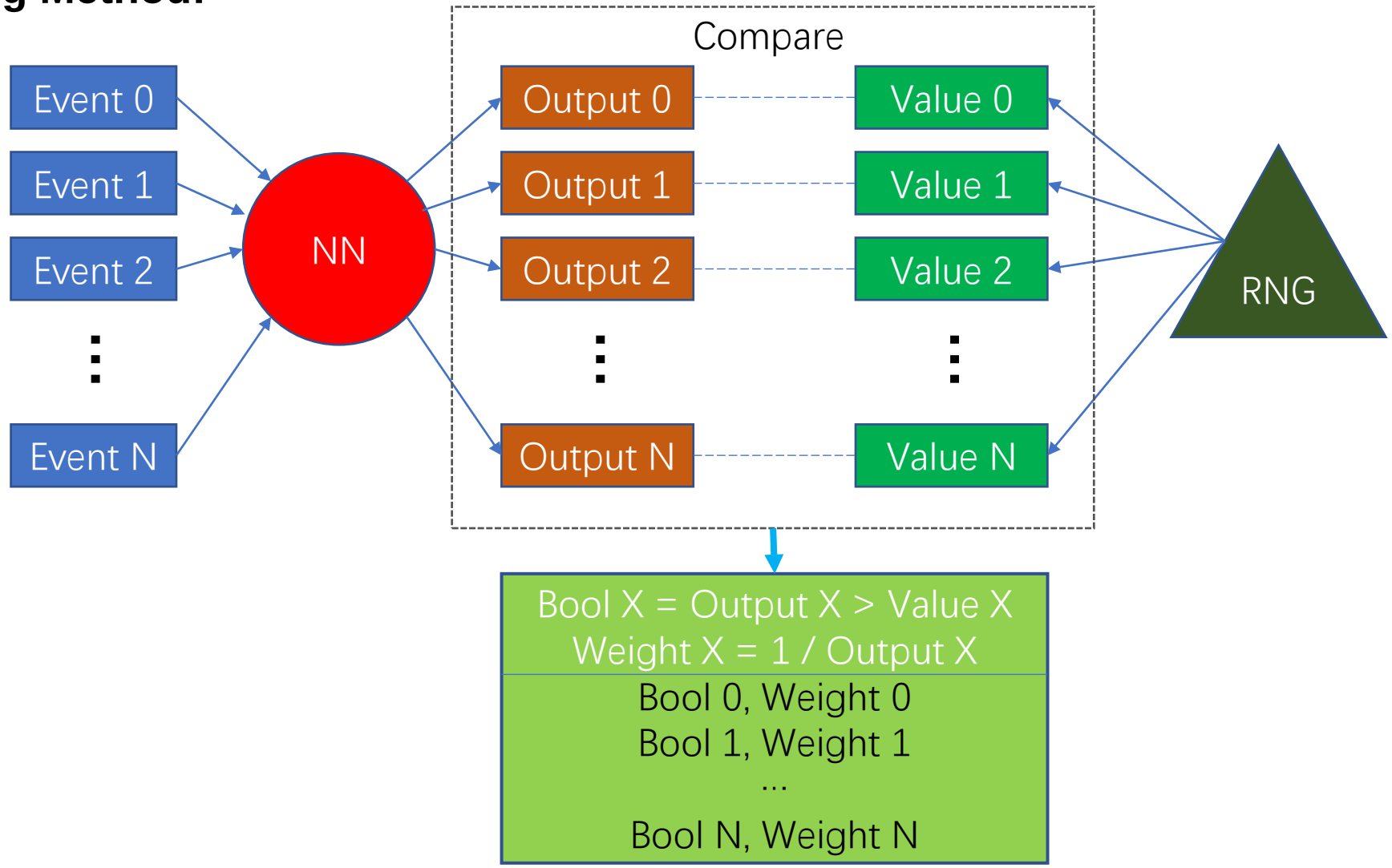
	t_{gen}	t_{NN}	t_{SR}	
TP	√	√	√	
FP	√	√	√	
FN	√	√		
TN	√	√		

Bias due to False-Negatives with Naive Filtering

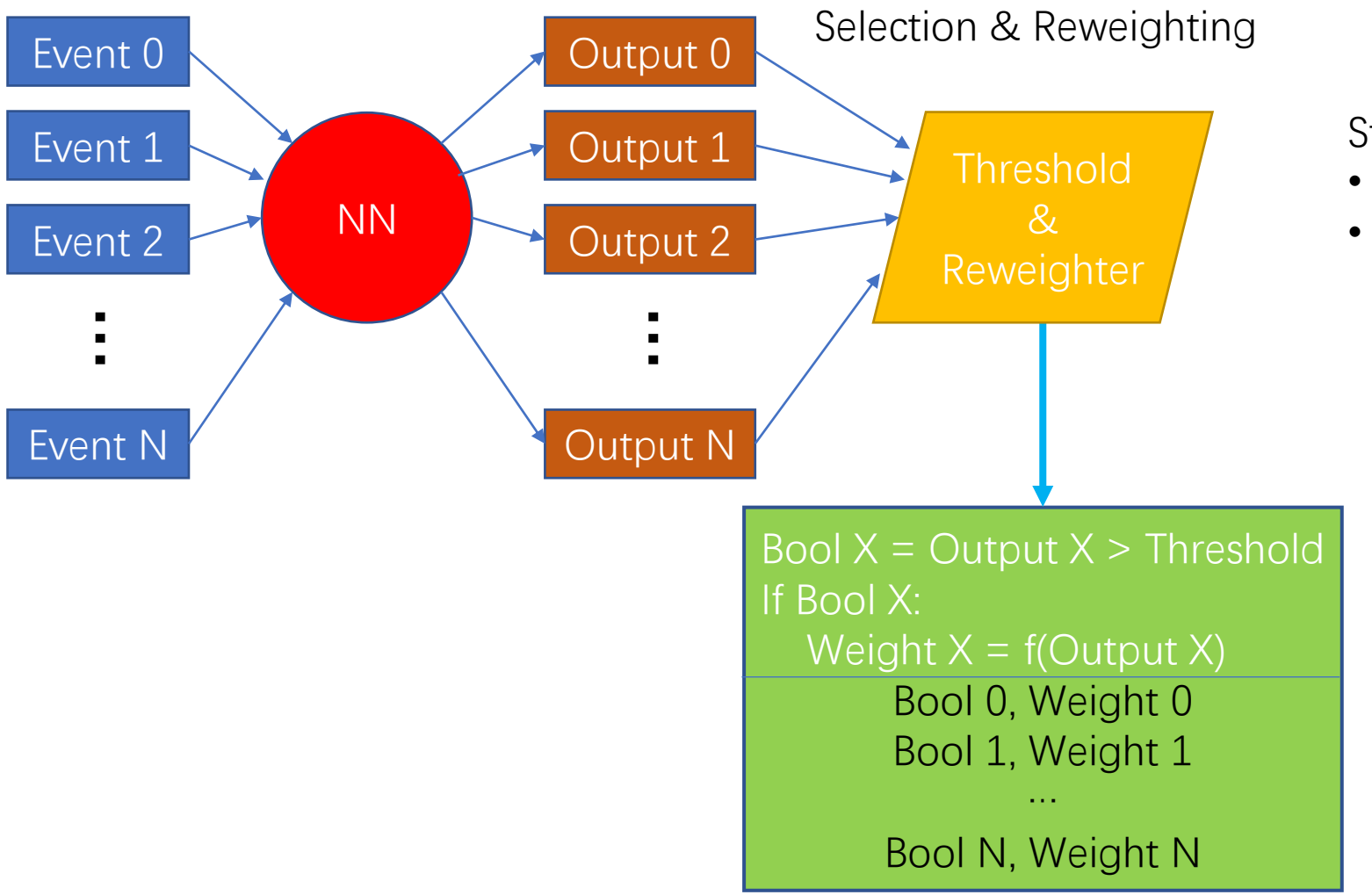


Event Weighting Methods

Sampling Method:



Reweighting Method:



Studied reweighters:

- GBDT Reweighting
- Histogram Reweighting

Reweighting Method:

- Train a Gradient Boosting Decision Tree (GBDT) classifier with some event level variables to distinguish between True-Positve events and False-Negative events
- GBDT Reweighting: use the outputs of the classifier directly:

$$w = \frac{1}{p_{clf}} = \frac{1}{p_{TP}/p_{TP+FN}} = \frac{p_{pass_skim}}{p_{TP}}$$

- Histogram Reweighting: compare the score histogram of all the events that can pass the skim (True-Positive + False-Negative) with the score histogram of True-Positives to give each bin of score a scaling factor:

$$w = w_{bin_i|p_{clf}\in bin_i} = \frac{H_{pass_skim,i}}{H_{TP,i}}\Big|_{p_{clf}\in bin_i}$$

Skim \ NN	Positive	Negative
Pass	True-Positive (TP)	False-Negative (FN)
Fail	False-Positive (FP)	True-Negative (TN)

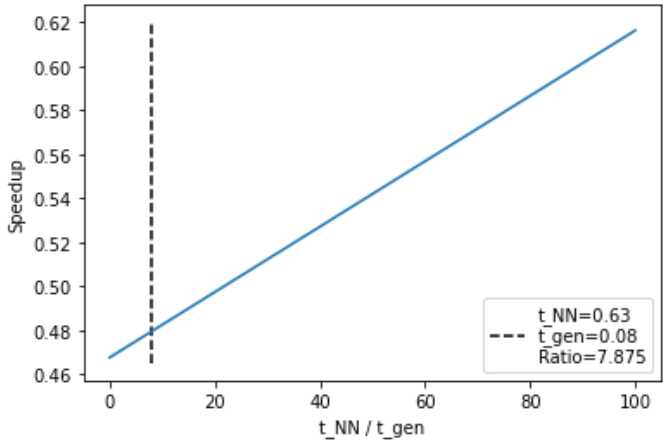
Metric: Speedup

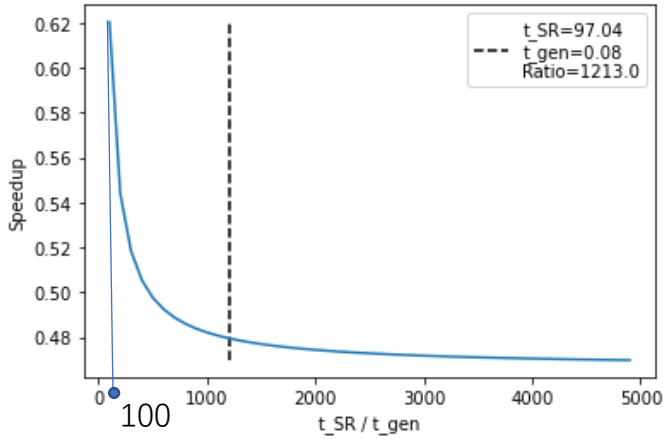
Variable	Formula	Remark
Skim retention rate	$r = 0.05$	Probability to pass the skim process
Times of different phases in s	$t_{gen} = 0.08$ $t_{NN} = 0.63$ $t_{SR} = 97.04$	Time for MC Generation, NN and Det. Sim + Reconstruction Taken from previous studies
(Inversed) Speedup	$R = \frac{t_{filter}}{t_{no_filter}} \sim {N_{eff}}^{-1} = \left(\frac{(\sum \omega_i p_i)^2}{\sum \omega_i^2 p_i}\right)^{-1}$	Ratio between time consumings of the whole work flows with and without NN filters for producing **the same effective sample size**

Weak dependency of **Speedup** on $\boldsymbol{t_{NN}}$ and $\boldsymbol{t_{SR}}$

↓

Safe to generalize

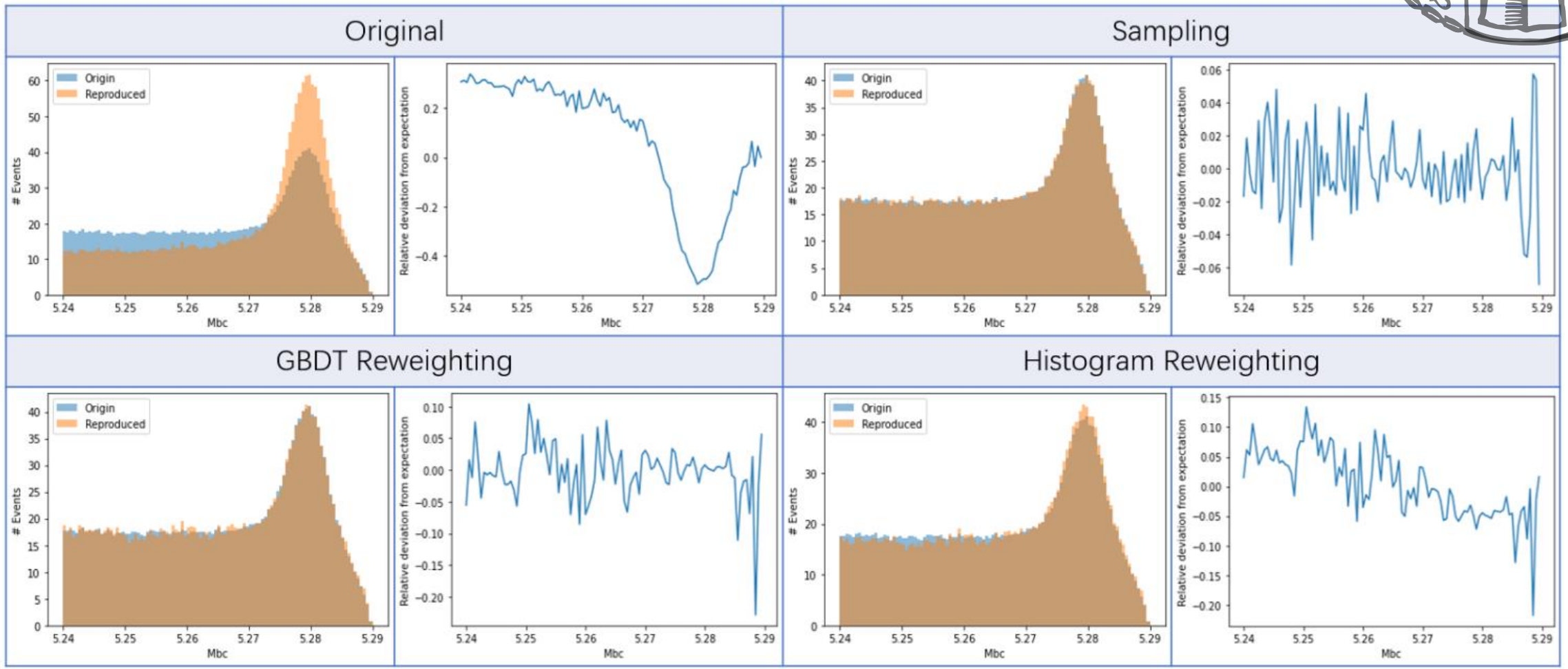




Compare

Sampling method	Reweighting method
Use NN output as **probability to keep event**	Use NN output as **score to cut on**
Weight events by inverse probability, like importance sampling	Reweight events to correct bias with the help of another classifier
No bias by construction	No bias for quantities included in reweighting
Use speedup as loss function in training	Use binary cross entropy in training
Best speedup: • 2.0	Best speedup: • GBDT Reweighting: 5.5 • Histogram Reweighting: 6.5
Highest bias of reweighted variables: (KS-Test) • All p-values above 0.5	Highest bias of reweighted variables: (KS-Test) • GBDT Reweighting: 8σ • Histogram Reweighting: 15σ

Reproduced M_{bc}



Results:

- Attention mechanism can improve NN performance for selective background monte carlo simulation
- Bias is avoided with sampling method while a speedup of factor 2 can still be maintained.
- Reweighting method can reach much higher speedup up to 6.5 but will still have some bias in the variables that are not used in the training of GBDT.

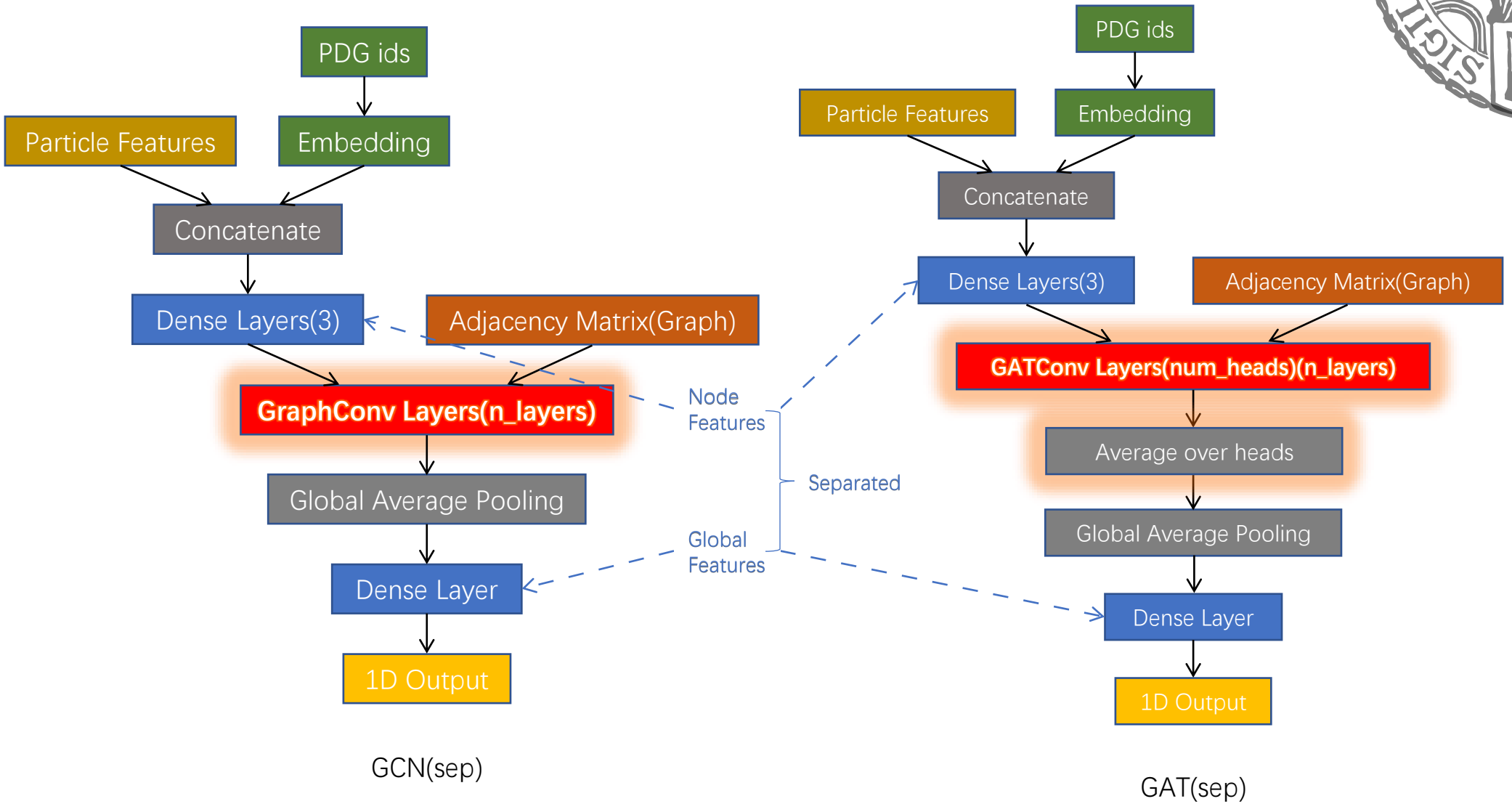
LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Thank You for your Attention

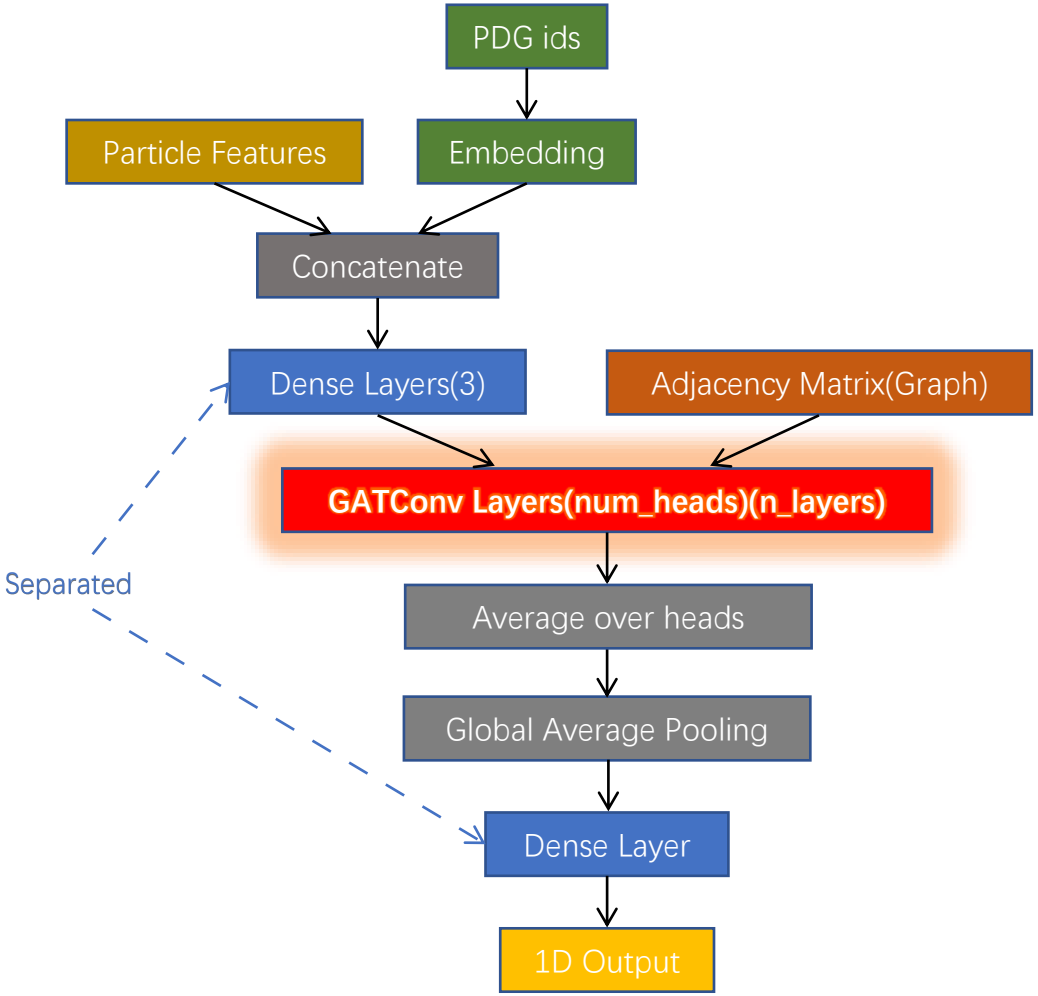
Boyang Yu [1], Nikolai Hartmann [1], Thomas Kuhr [1]
[1] *Ludwig-Maximilians-Universität München*
IDT-UM Collaboration Meeting, Feb 15th, 2022

Backup

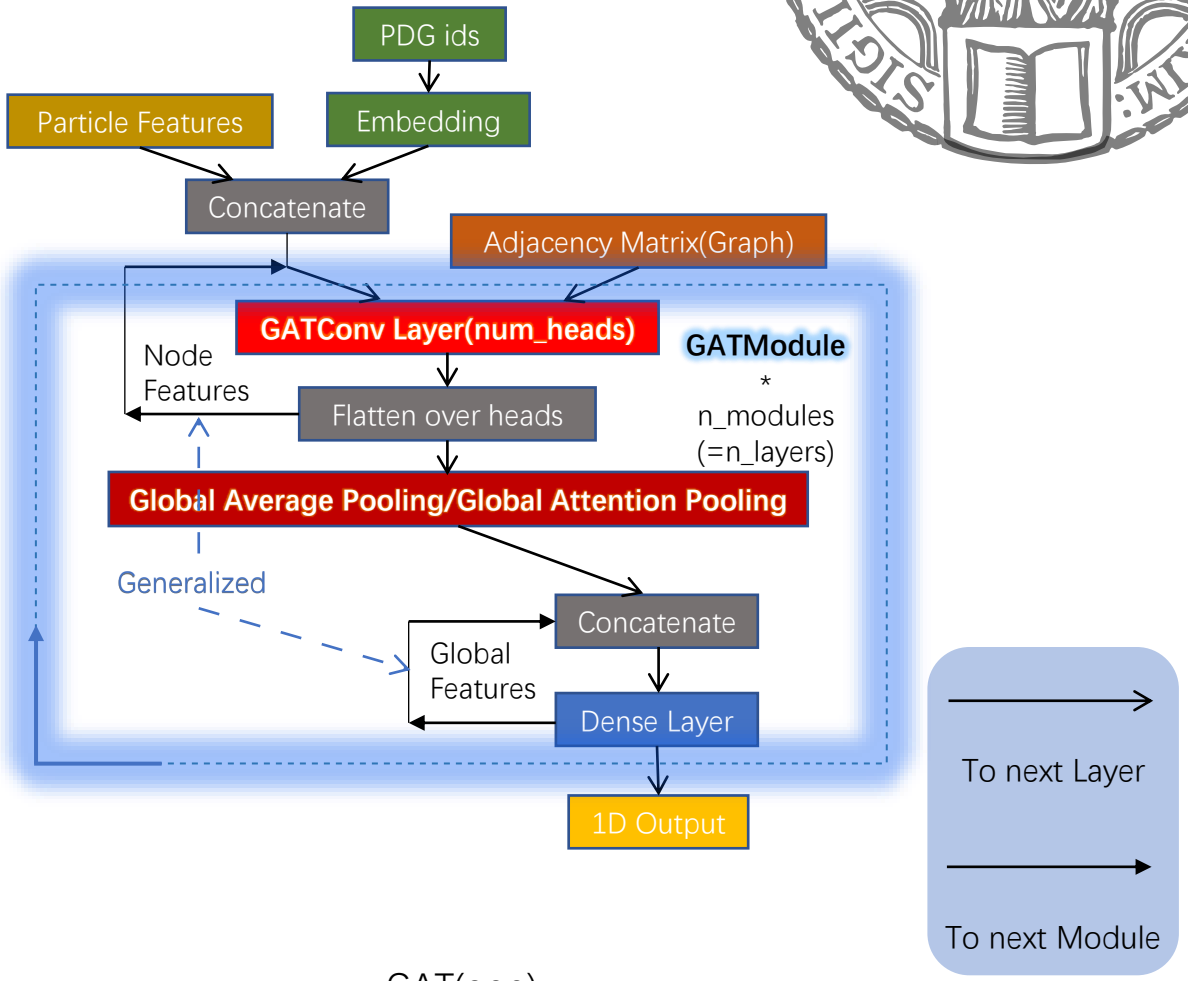
Backup: Network Structures



Backup: Network Structures



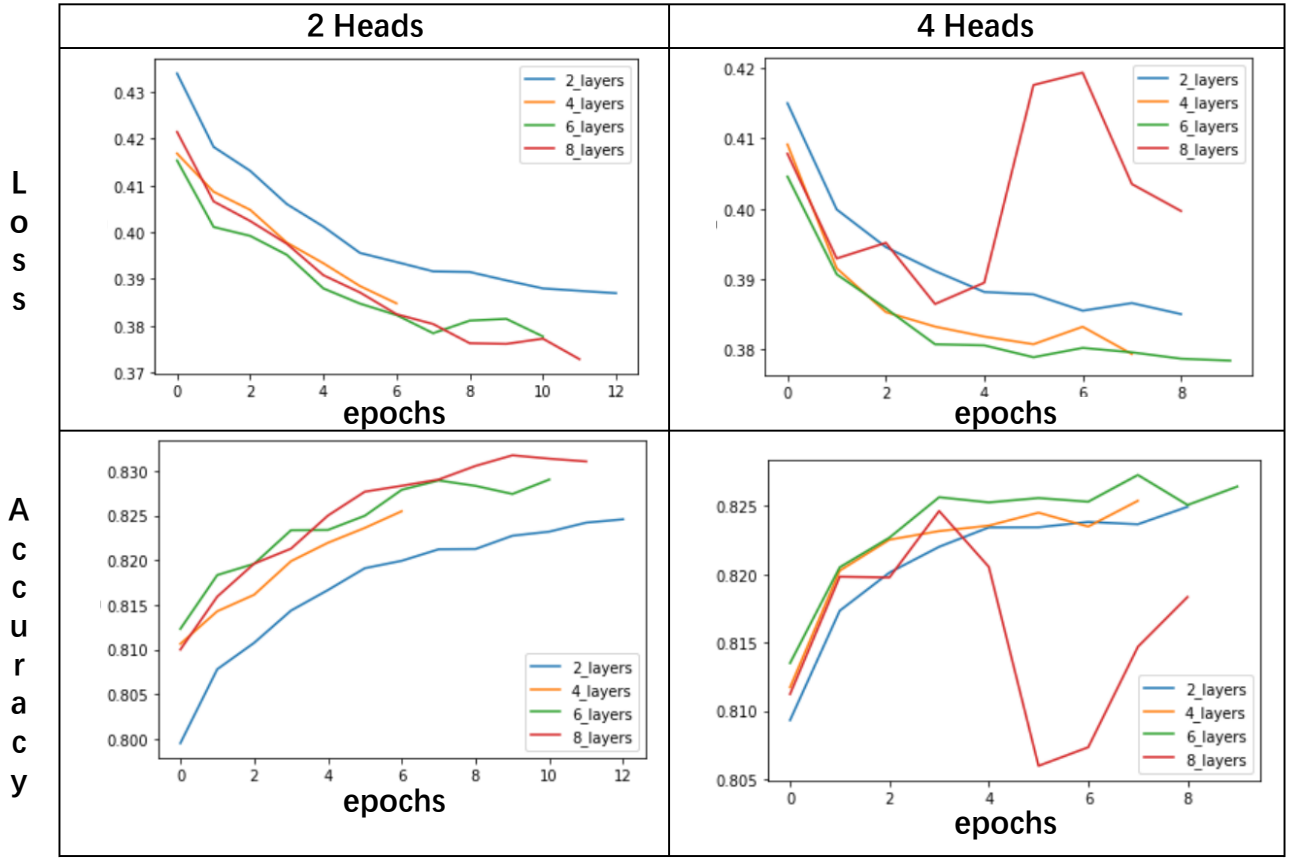
GAT(sep)

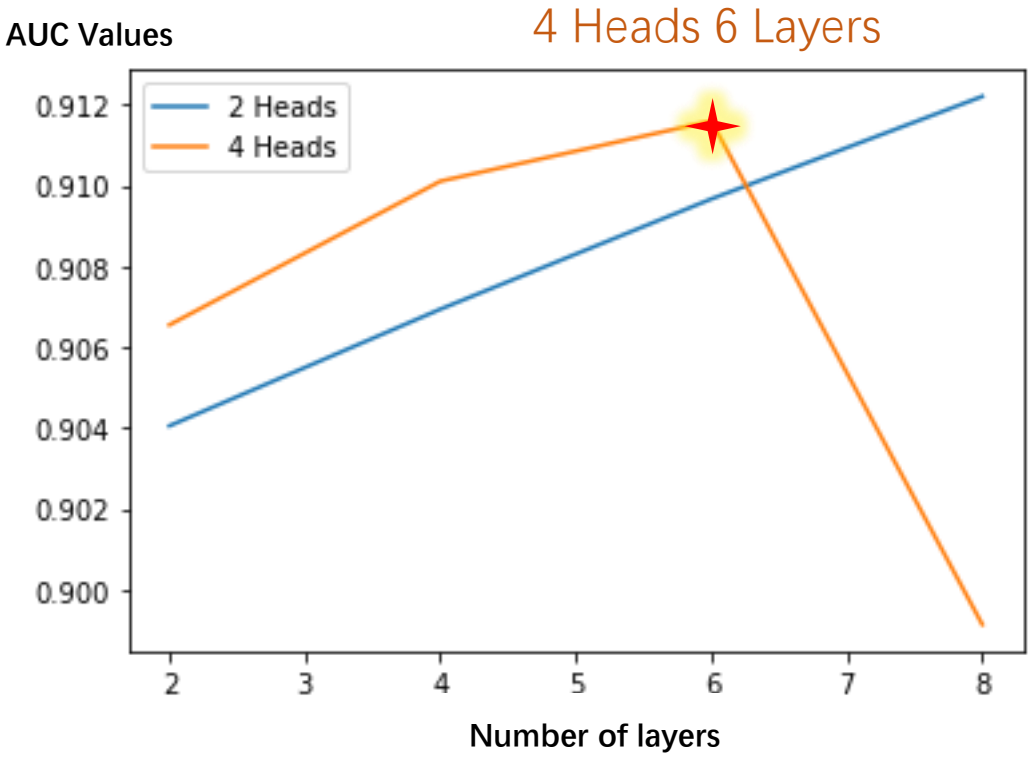


GAT(gen)
or
GAT+GAP(gen)

Backup: Network Structures

Quantitative Studies





Backup: Evaluations

Comparison

Parameters:

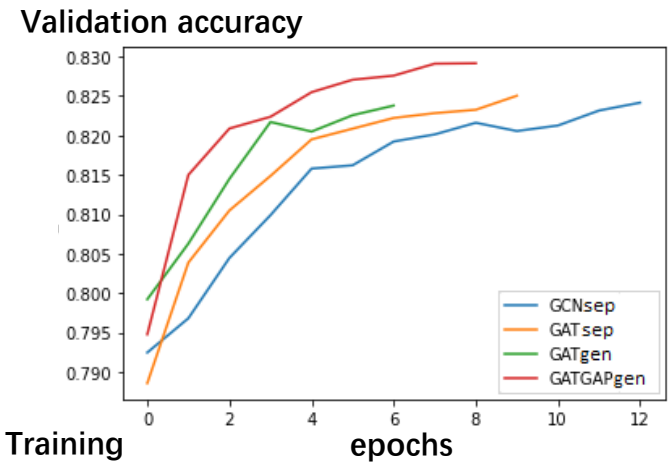
- n_heads = 4
- n_layers = 6
- n_units = 128
- batch_size = 128
- n_train = 0.9M
- n_val = 0.1M
- n_test = 0.5M

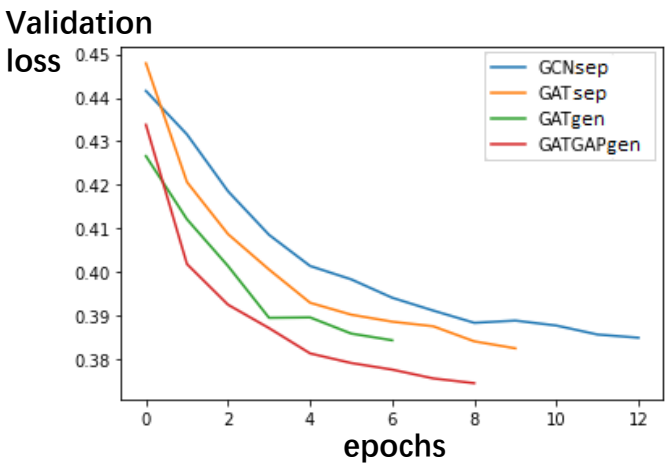
Loss:

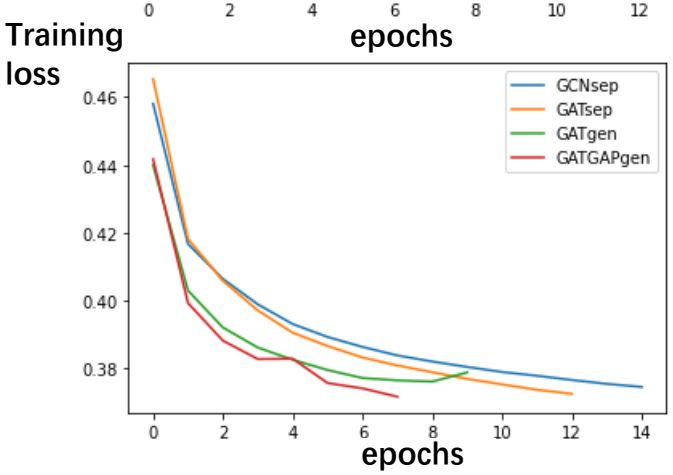
- Entropy

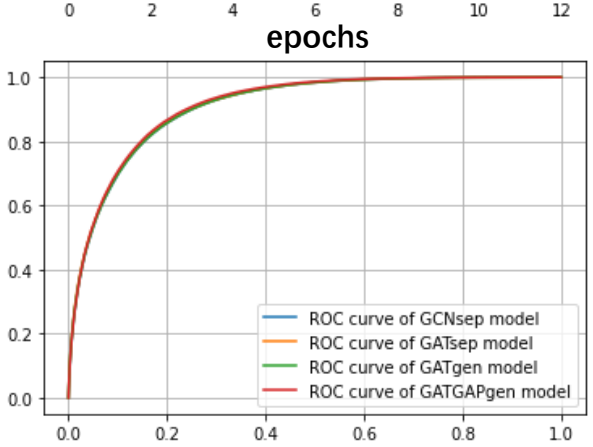
EarlyStopping:

- patience = 3
- delta = 1e-5









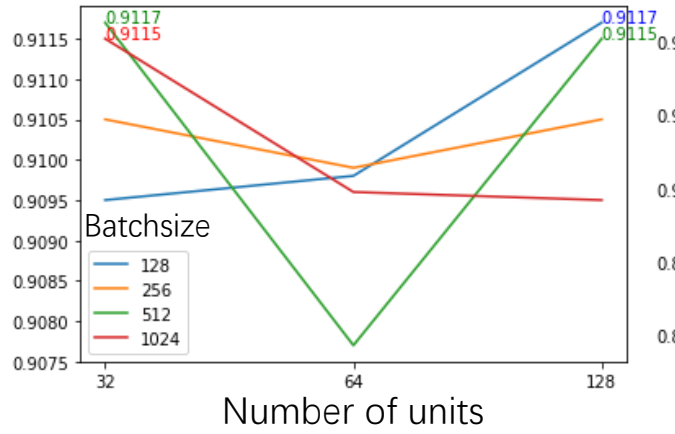
	GCN(sep)	GAT(sep)	GAT(gen)	GAT+GAP(gen)
TrainingTime	3619.46s	4047.47s	3471.48s	5049.81s
AUCValues	0.90831	0.90937	0.90891	0.91216

Improved Selective Background Monte Carlo Simulation at Belle II with Graph Attention Networks and Weighted Events

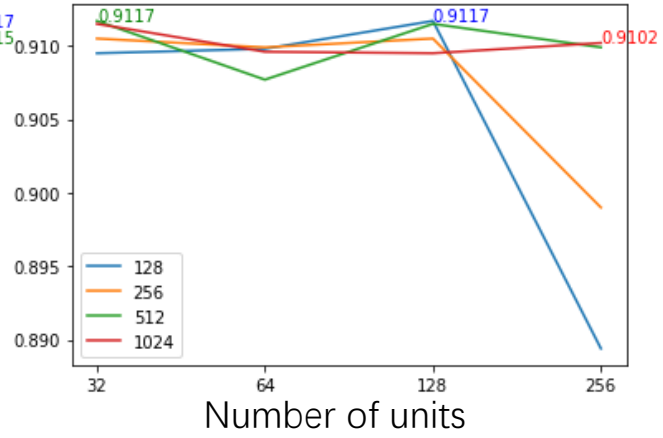
Backup: Evaluations

Grid Search

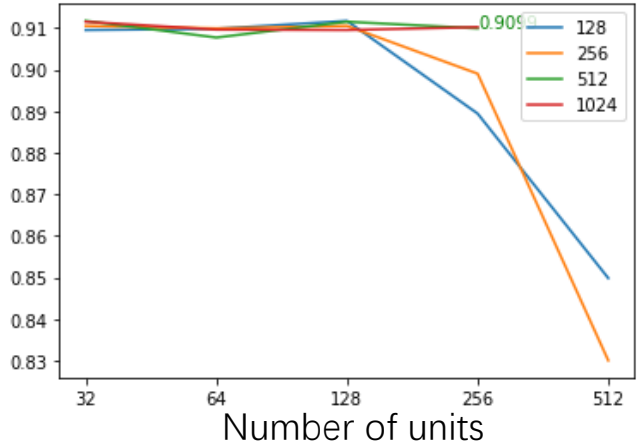
AUC



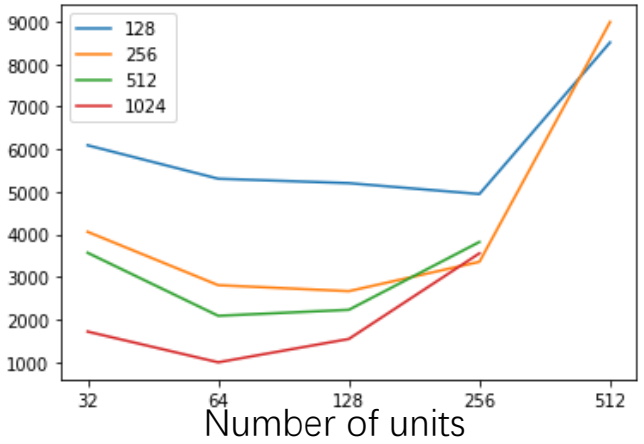
AUC



AUC



Training Time



Best Combinations

Batch-size	Number of units	AUC	Training Time
128	128	0.9117	5205
256	32	0.9105	4061
256	128	0.9105	2666
512	32	0.9117	3568
512	128	0.9115	2228
1024	32	0.9115	1716
1024	256	0.9102	3556

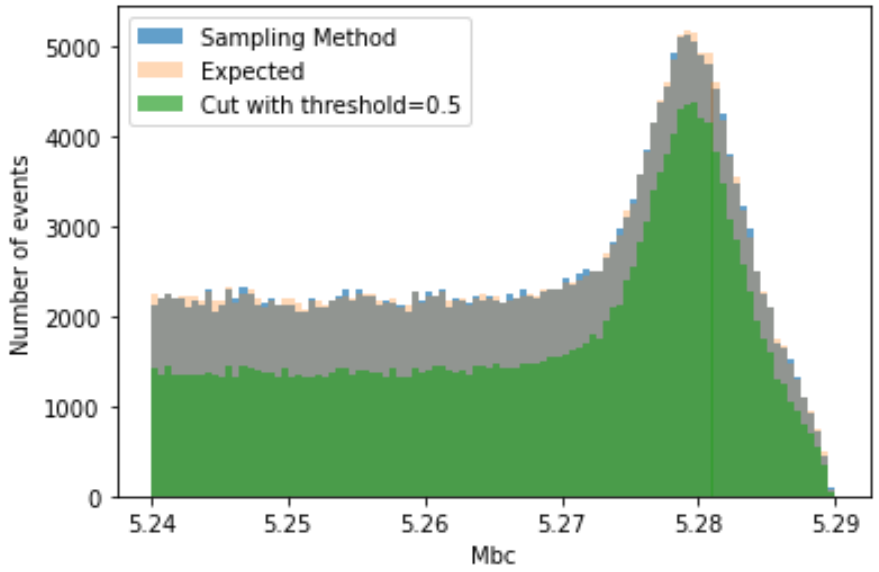
Network Sizes

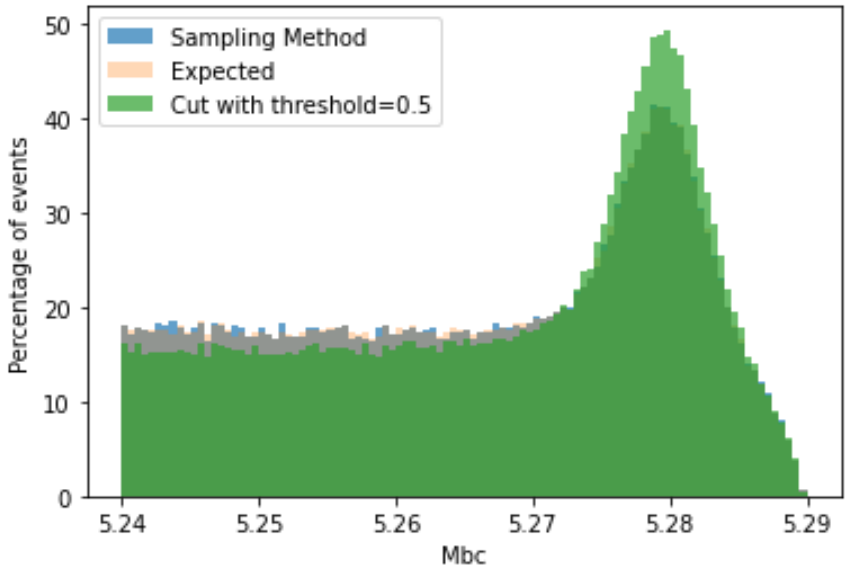
# Units	# Parameters
32	120,527
64	459,951
128	1,808,495
256	7,184,367
512	28,651,247

Sampling

- Previous method: Cut according to neural network outputs
- Problem: Inevitable bias

- Our method: Sampling with probability given by neural network outputs
- Problem: Statistical uncertainty





Relative statistical uncertainty and effective sample size

Variable	Formula	Remark
NN outputs / Probabilities to pass	$\{p_i\}$	'i' refers to each event in the whole sample (batch)
Weights	$\{\omega_i\} = \left\{\frac{1}{p_i}\right\}$	Infinities (at $p_i = 0$) are excluded and set to 0 Avoid the bias by construction
Relative statistical uncertainty	$S = \frac{\sqrt{\sum \omega_i^2 p_i}}{\sum \omega_i p_i}$	$\sum \omega_i^2 p_i = \sum \omega_i$ $\sum \omega_i p_i = N$ Here consider only passed events (label = 1)
Effective sample size	$N_{eff} = \frac{1}{S^2}$	Number of events needed to reach **the same statistical uncertainty without sampling**

Speedup rate

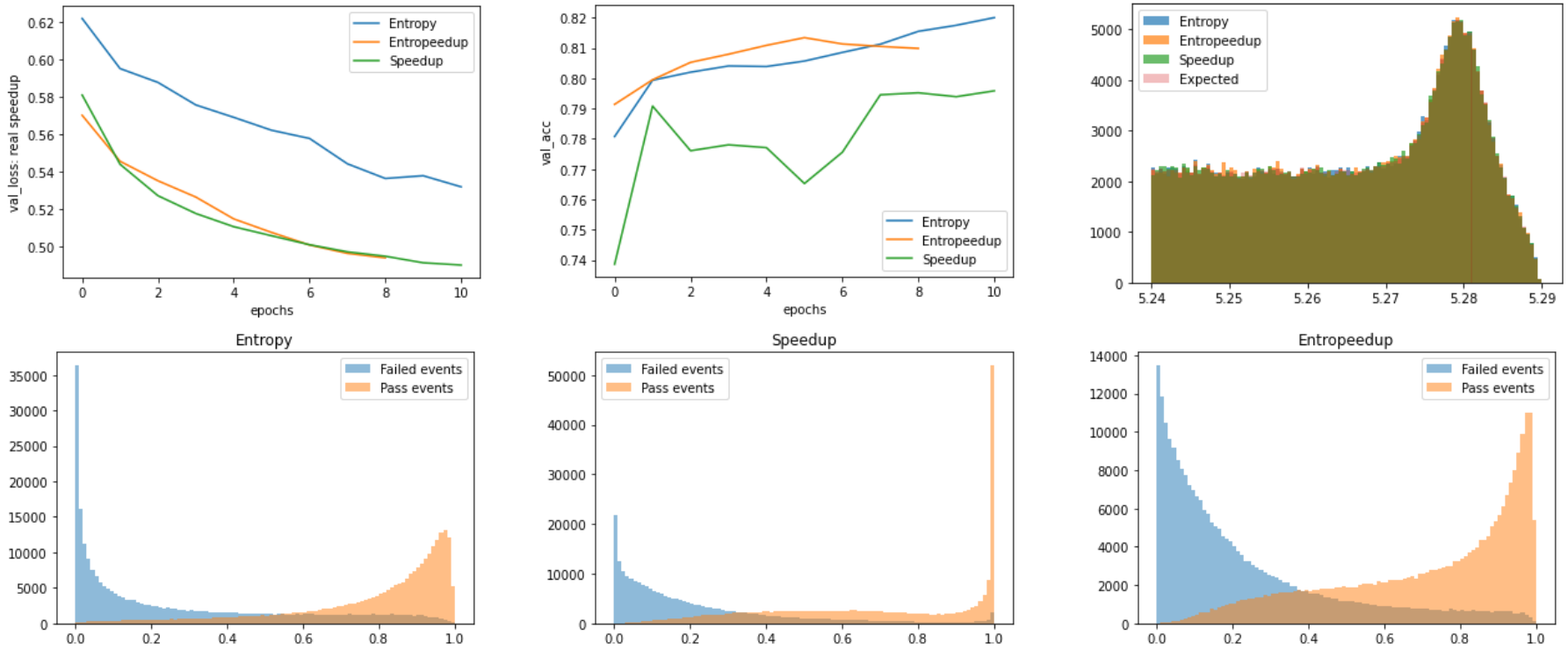
Variable	Formula	Remark
Skim retention rate	$r = 0.05$	Probability to pass the skim process
Times of different phases in ms	$t_{gen} = 0.08$ $t_{NN} = 0.63$ $t_{SR} = 97.04$	Taken from previous studies
Effective number of events after sampling	$n_+ = \sum p_i$ $n_- = \sum (1 - p_i)$	$\{p_i\}$ will be devided into two subsets where the events will/won't pass the skim process
Time consuming with NN filter	$t_+ = [n_{TP} r + n_{FP}(1 - r)](t_{gen} + t_{NN} + t_{SR})$ $t_- = [n_{FN} r + n_{TN}(1 - r)](t_{gen} + t_{NN})$	Positive/Negative: Result of sampling True/False: Result of sampling == skim process
Time consuming without NN	$t_0 = N_{eff}(t_{gen} + t_{NN})$	To reach the same statistical uncertainty
(Inverse) Speedup rate	$R = \frac{t_+ + t_-}{t_0}$	The lower the better

Performances of different training losses

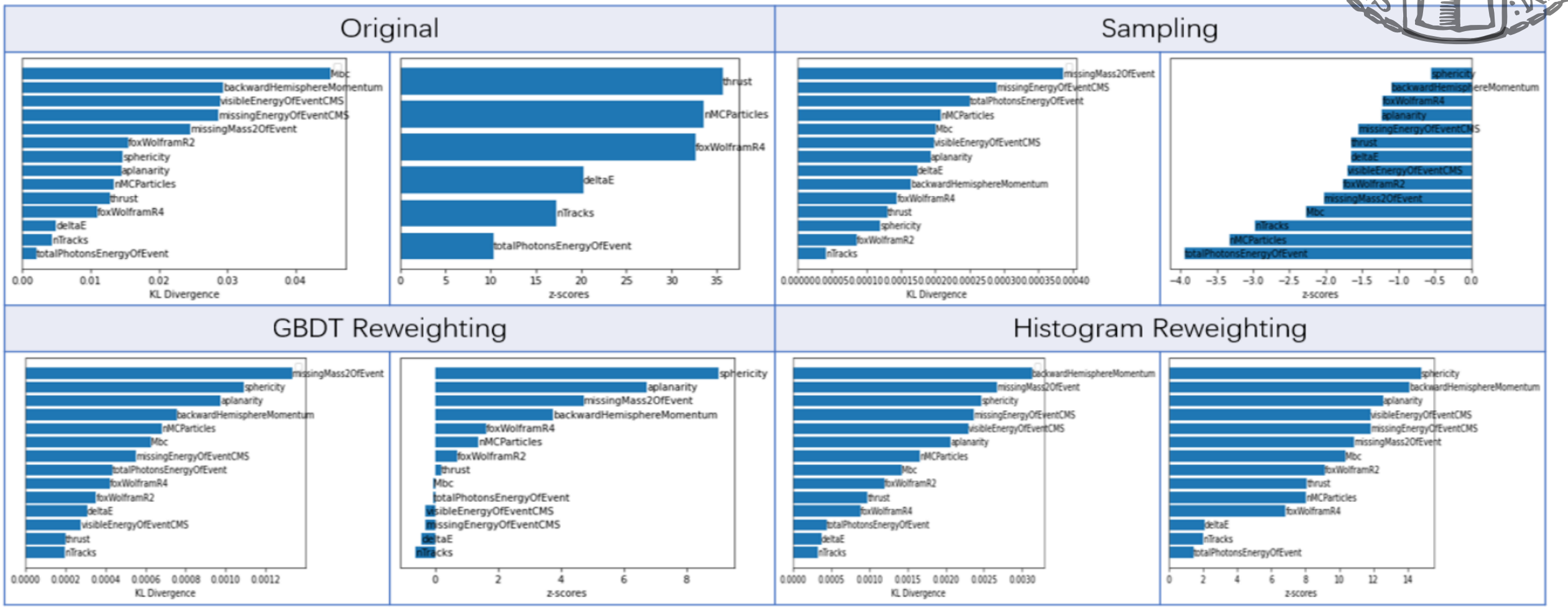
Speedup always as validation loss!

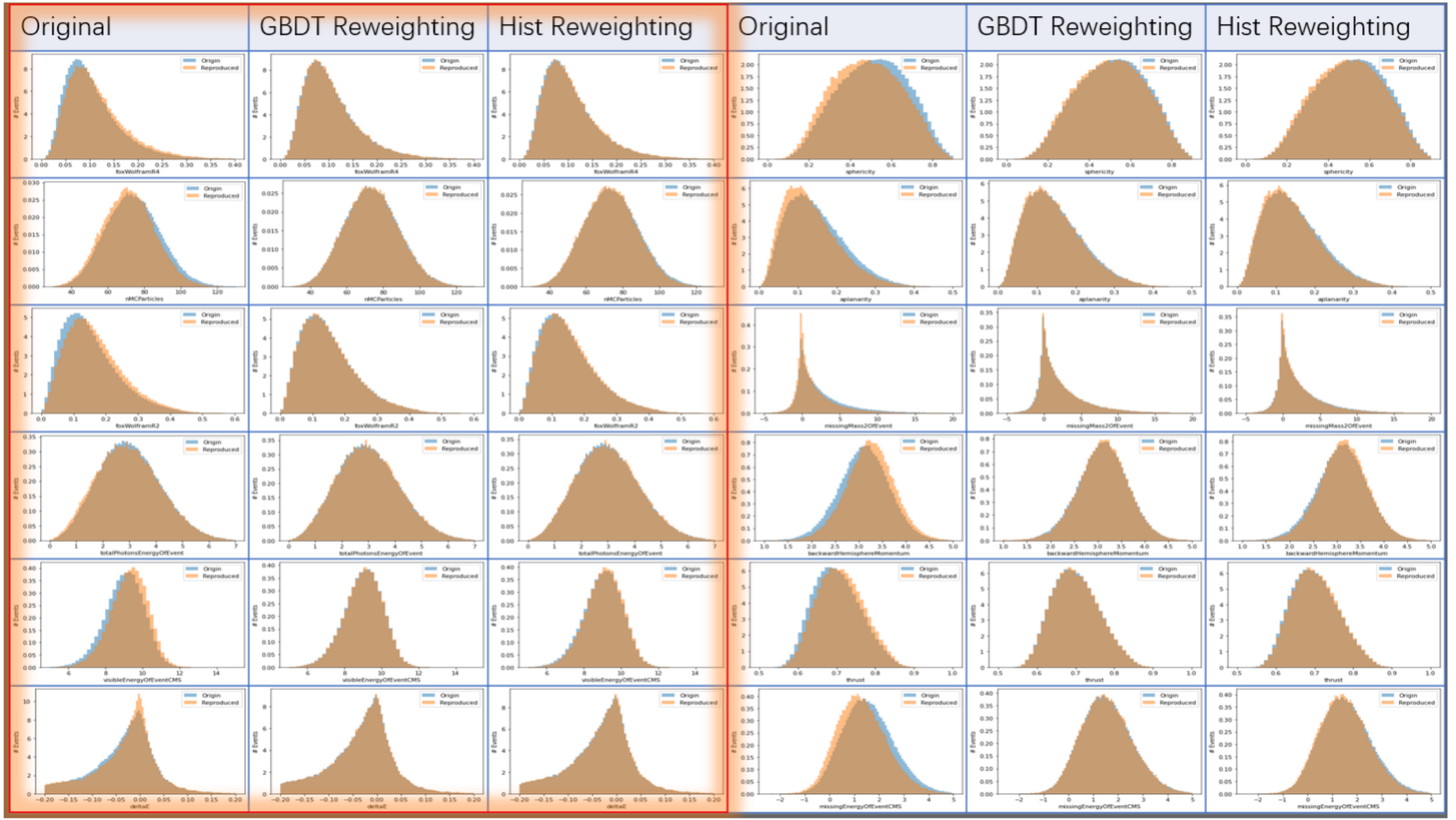
Entropy: Binary cross entropy between NN outputs $\{p_i\}$ and skim results $\{y_i \in \{0,1\}\}$
Speedup: (Inverse) Speedup rate R
Entropeedup: Training-epoch-dependent combination of Entropy and Speedup

$$\text{Entropeedup} = e^{-epoch/10} Entropy + Speedup$$



KS-Test


Original
Mbc
backwardHemisphereMomentum
visibleEnergyOfEventCMS
missingEnergyOfEventCMS
missingMass2OfEvent
foxWolframR2
sphericity
aplanarity
nMCParticles
thrust
foxWolframR4
deltaE
nTracks
totalPhotonsEnergyOfEvent
0.00 0.01 0.02 0.03 0.04
KL Divergence
thrust
nMCParticles
foxWolframR4
deltaE
nTracks
totalPhotonsEnergyOfEvent
0 5 10 15 20 25 30 35
z-scores
Sampling
missingMass2OfEvent
missingEnergyOfEventCMS
totalPhotonsEnergyOfEvent
nMCParticles
Mbc
visibleEnergyOfEventCMS
aplanarity
deltaE
backwardHemisphereMomentum
foxWolframR4
thrust
sphericity
foxWolframR2
nTracks
0.000000 0.00005 0.00010 0.00015 0.00020 0.00025 0.00030 0.00035 0.00040
KL Divergence
sphericity
backwardHemisphereMomentum
foxWolframR4
aplanarity
missingEnergyOfEventCMS
thrust
deltaE
visibleEnergyOfEventCMS
foxWolframR2
missingMass2OfEvent
Mbc
nTracks
nMCParticles
totalPhotonsEnergyOfEvent
−4.0 −3.5 −3.0 −2.5 −2.0 −1.5 −1.0 −0.5 0.0
z-scores
GBDT Reweighting
missingMass2OfEvent
sphericity
aplanarity
backwardHemisphereMomentum
nMCParticles
Mbc
missingEnergyOfEventCMS
totalPhotonsEnergyOfEvent
foxWolframR4
foxWolframR2
deltaE
visibleEnergyOfEventCMS
thrust
nTracks
0.0000 0.0002 0.0004 0.0006 0.0008 0.0010 0.0012
KL Divergence
sphericity
aplanarity
missingMass2OfEvent
backwardHemisphereMomentum
foxWolframR4
nMCParticles
foxWolframR2
thrust
Mbc
totalPhotonsEnergyOfEvent
visibleEnergyOfEventCMS
missingEnergyOfEventCMS
deltaE
nTracks
0 2 4 6 8
z-scores
Histogram Reweighting
backwardHemisphereMomentum
missingMass2OfEvent
sphericity
missingEnergyOfEventCMS
visibleEnergyOfEventCMS
aplanarity
nMCParticles
Mbc
foxWolframR2
thrust
foxWolframR4
totalPhotonsEnergyOfEvent
deltaE
nTracks
0.0000 0.0005 0.0010 0.0015 0.0020 0.0025 0.0030
KL Divergence
sphericity
backwardHemisphereMomentum
aplanarity
visibleEnergyOfEventCMS
missingEnergyOfEventCMS
missingMass2OfEvent
Mbc
foxWolframR2
thrust
nMCParticles
foxWolframR4
deltaE
nTracks
totalPhotonsEnergyOfEvent
0 2 4 6 8 10 12 14
z-scores


Original
GBDT Reweighting
Hist Reweighting
Original
GBDT Reweighting
Hist Reweighting
Origin
Reproduced
Events
foxWolframR4
nMCParticles
foxWolframR2
totalPhotonsEnergyOfEvent
visibleEnergyOfEventCMS
deltaE
sphericity
aplanarity
missingMass2OffEvent
backwardHemisphereMomentum
thrust
missingEnergyOfEventCMS