High level reconstruction with deep learning at ILD full simulation

Taikan Suehara / 末原　大幹
(ICEPP, The University of Tokyo)
R. Tagami, T. Murata, W. Ootani, M. Ishino (ICEPP, UTokyo),
P. Wahlen (ETH/IP Paris/ILANCE UTokyo),
L. Gui (Imperial/Kyushu U.), T. Tanabe (MI-6 Ltd.)

Deep learning with Higgs factories

- Significant part of reconstruction is “pattern recognition”
  - Cut-based method should have limitation
  - DNN should take more information than human-tuning
- “Big data” detector for Higgs factories
  - Much more detector elements than before
  - Should fit with modern network with many learning weights
  - Also good for detector design
- Sensor → objects → physics
  should be more seamless with deep learning techniques
  - Event reconstruction is the heart of the chain

Today's topics

All works done with ILD full simulation (plus FCCee Delphes for comparison)

Flavor tagging with Particle Transformer (ParT)

- Modern DNN-based jet flavor tagging originally developed for LHC
- Much better performance than current algorithm (LCFIPlus(2013))
  - Reported by FCCee colleagues earlier, comparison done
- Big impact on Higgs studies
  - Including self coupling
- Strange tagging, under investigation

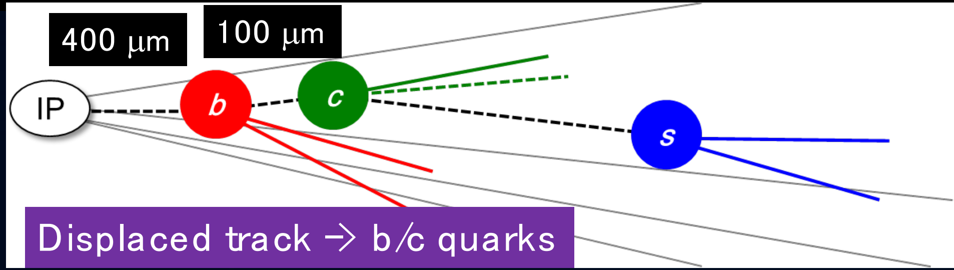
Particle flow with DNN

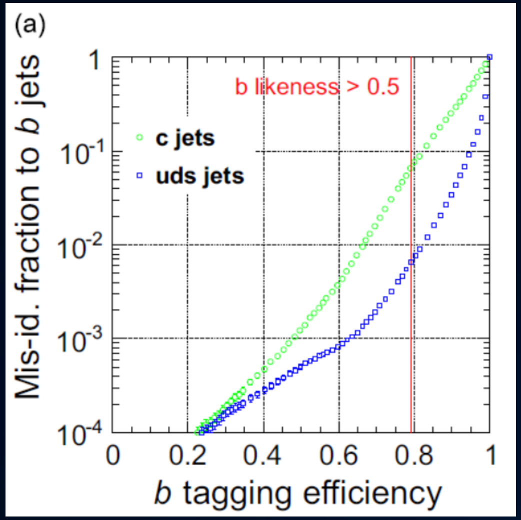
- GNN originally developed for CMS HGCAL clustering
  - GravNet / Object condensation
- Track-cluster matching implemented
- Promising initial results seen
  - Comparable with PandoraPFA
  - Still much rooms to improve
- Another trial with NLP-like architecture (Transformer)

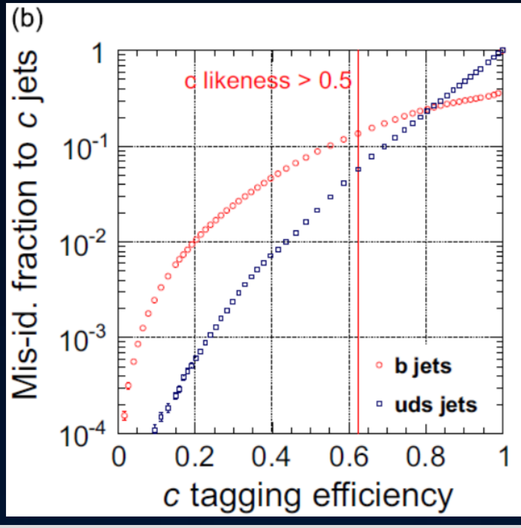
Flavor tagging for Higgs factories

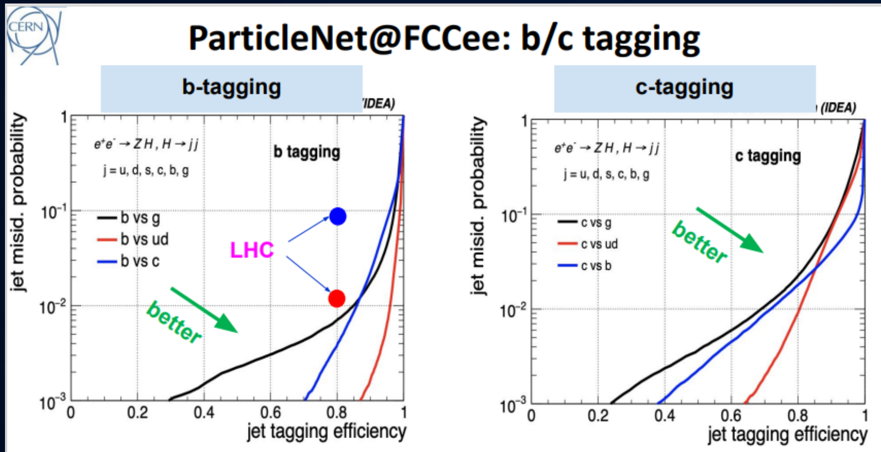
- Jet flavor tagging is essentially important for Higgs studies (including self coupling)
- LCFIPlus (published 2013)[1] was long used for flavor tagging
  - b-tag: ~80% eff., 10% c / 1% uds acceptance;
  - c-tag: ~50% eff., 10% b / 2% uds acceptance.
- Recently FCCee reported ~10x better rejection using ParticleNet (GNN)
  - To be confirmed with full simulation (with latest algorithm: Particle Transformer (ParT)

  → If good, consider to apply to physics analyses hopefully with common framework







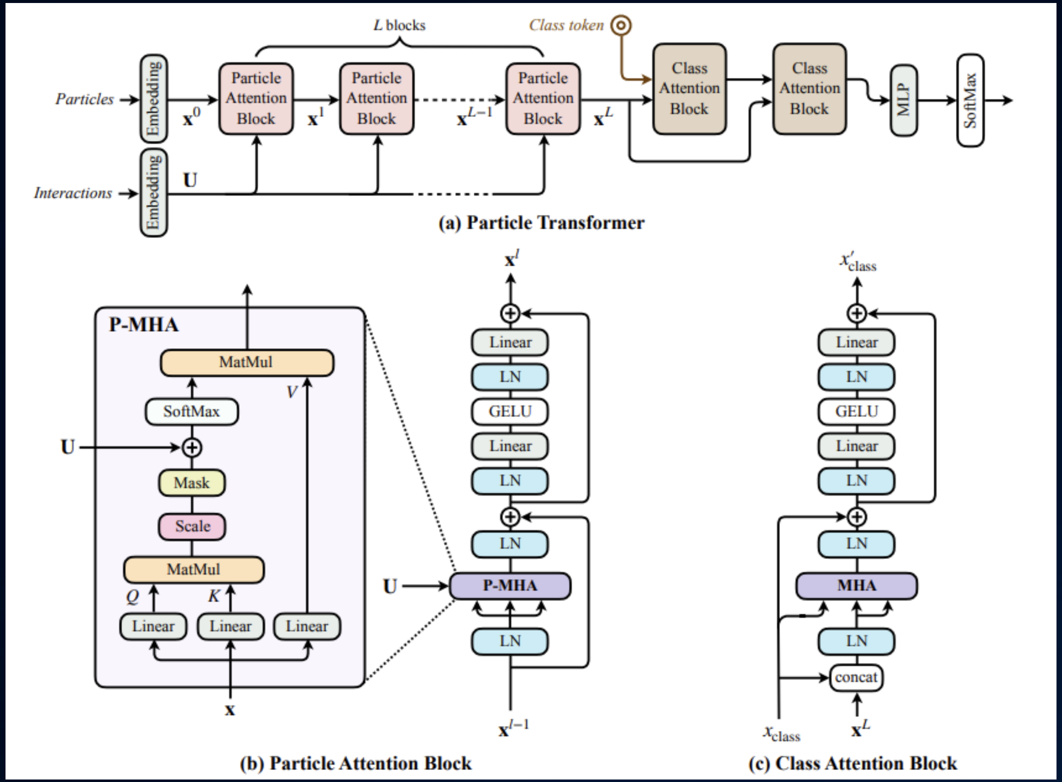


WP	Eff (b)	Mistag (g)	Mistag (ud)	Mistag (c)
Loose	90%	2%	0.1%	2%
Medium	80%	0.7%	<0.1%	0.3%

WP	Eff (c)	Mistag (g)	Mistag (ud)	Mistag (b)
Loose	90%	7%	7%	4%
Medium	80%	2%	0.8%	2%

Particle Transformer (ParT)

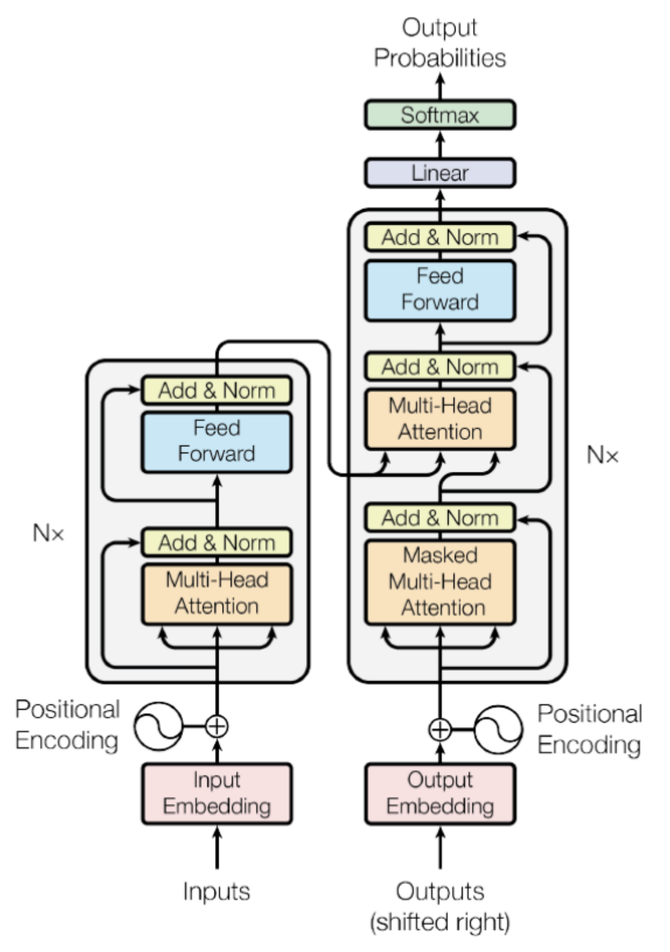
- Transformer: self-attention based algorithm intensively used for NLP (e.g. chatGPT)
  - Weak biasing: possible to train big samples efficiently (with more learnable weights) but demanding big training sample for high performance
- ParT is a new Transformer-based architecture for Jet tagging, published in 2022[2].
- Surpasses the performance of ParticleNet
  - ParticleNet (or other GNNs) only looks "neighbor" particles while Transformer judges where to look by training



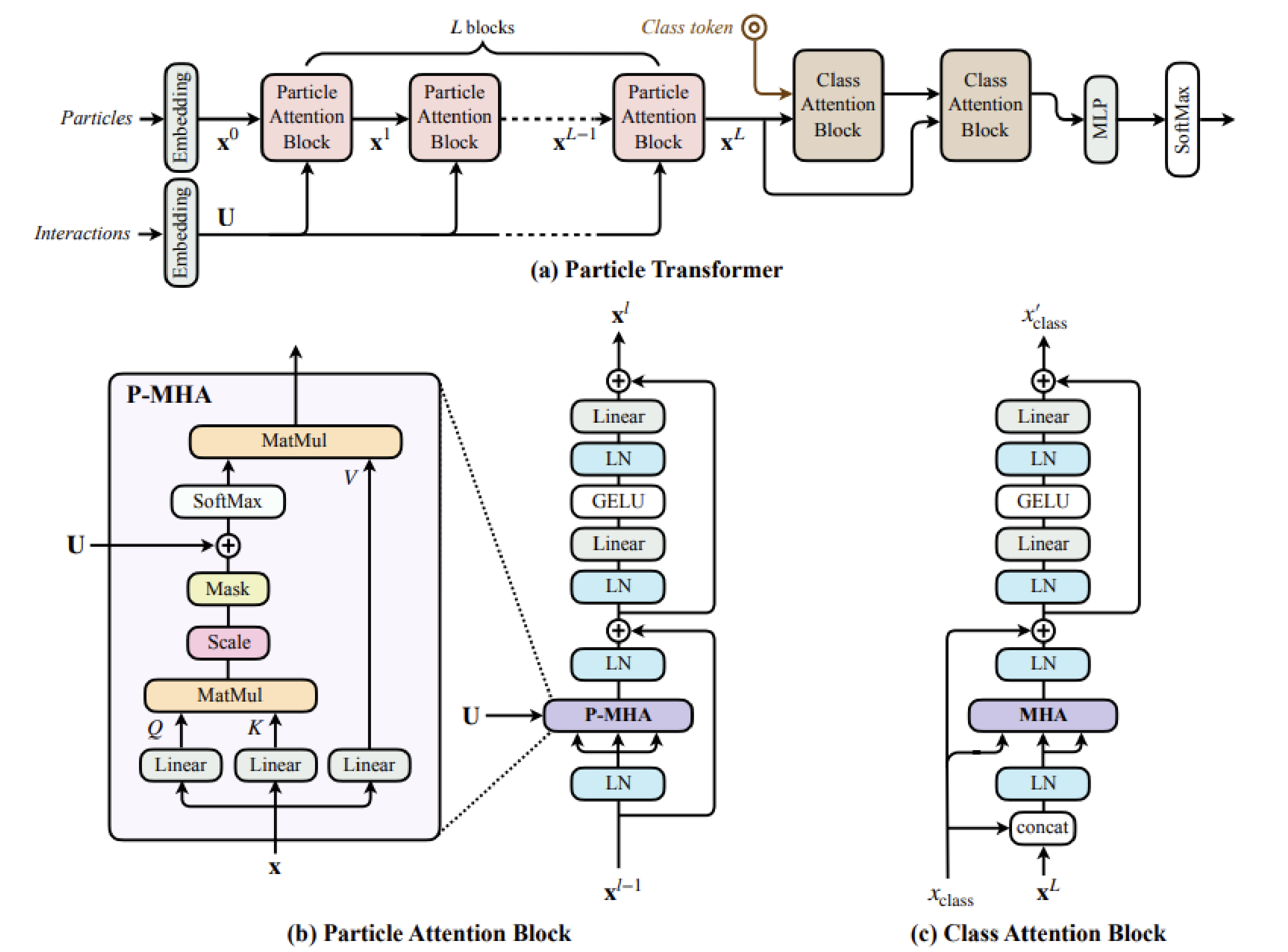
Performance on event categorization (ie. not direct flavor tagging but flavor information is essential for the categorization)

	All classes Accuracy	AUC	$H \to b\bar{b}$ Rej$_{50\%}$	$H \to c\bar{c}$ Rej$_{50\%}$	$H \to gg$ Rej$_{50\%}$	$H \to 4q$ Rej$_{50\%}$	$H \to \ell\nu qq'$ Rej$_{99\%}$	$t \to bqq'$ Rej$_{50\%}$	$t \to b\ell\nu$ Rej$_{99.5\%}$	$W \to qq'$ Rej$_{50\%}$	$Z \to q\bar{q}$ Rej$_{50\%}$
PFN	0.772	0.9714	2924	841	75	198	265	797	721	189	159
P-CNN	0.809	0.9789	4890	1276	88	474	947	2907	2304	241	204
ParticleNet	0.844	0.9849	7634	2475	104	954	3339	10526	11173	347	283
ParT	**0.861**	**0.9877**	**10638**	**4149**	**123**	**1864**	**5479**	**32787**	**15873**	**543**	**402**
ParT (plain)	0.849	0.9859	9569	2911	112	1185	3868	17699	12987	384	311

Comparison between regular Transformer and Particle Transformer



Regular Transformer



Particle Transformer

Note:
- MHA – MultiHeadAttention
- P-MHA – Augmented version of MHA by Particle Transformer that involves Interactions Embeddings instead of Positional Embeddings

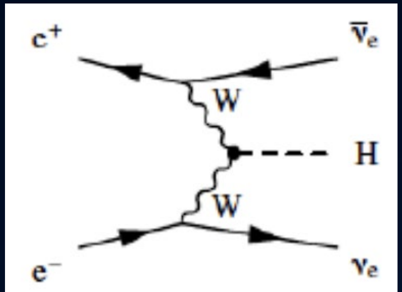
Data Used For Investigation

- ILD full simulation:

  1. e+ e- ⟶ qq (at 91 GeV)
     (DBD sample used for initial LCFIPlus study)
  2. e+ e- ⟶ ννH ⟶ννqq (at 250 GeV)
     (2020 production, process ID: 410001-410006)

  With 1M jets (500k events) each

$$\left\{ \begin{array}{c} \text{q = b,c,uds} \\ \text{ν = neutrino} \end{array} \right\}$$

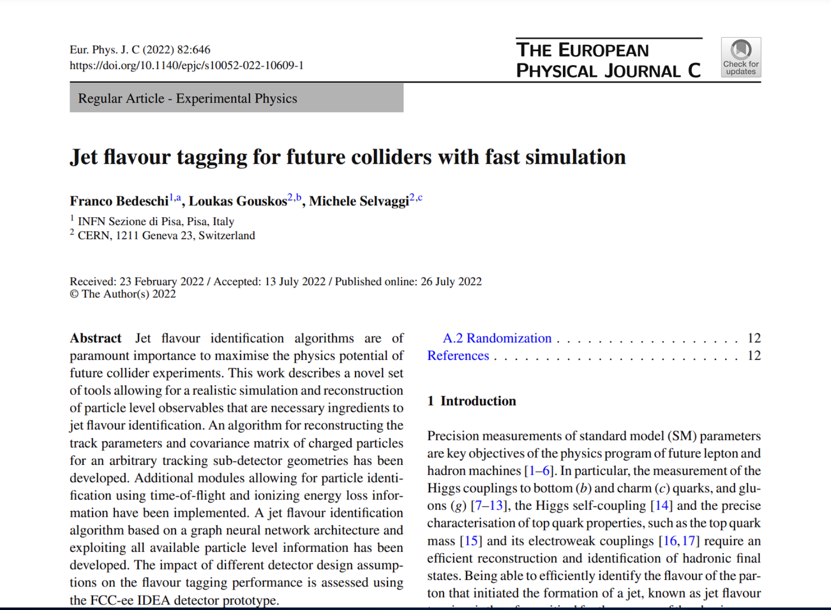


- FCCee fast simulation (Delphes with IDEA detector):

  e+ e- ⟶ ννH ⟶ννqq (at 240 GeV)

  With 10M jets (5M events) each

- 80% are used for training, 5% for validation, 15% for test



Eur. Phys. J. C (2022) 82:646
https://doi.org/10.1140/epjc/s10052-022-10609-1

The European Physical Journal C

Regular Article - Experimental Physics

Jet flavour tagging for future colliders with fast simulation

Franco Bedeschi[1,a], Loukas Gouskos[2,b], Michele Selvaggi[2,c]
[1] INFN Sezione di Pisa, Pisa, Italy
[2] CERN, 1211 Geneva 23, Switzerland

Received: 23 February 2022 / Accepted: 13 July 2022 / Published online: 26 July 2022

Abstract Jet flavour identification algorithms are of paramount importance to maximise the physics potential of future collider experiments. This work describes a novel set of tools allowing for a realistic simulation and reconstruction of particle level observables that are necessary ingredients to jet flavour identification. An algorithm for reconstructing the track parameters and covariance matrix of charged particles for an arbitrary tracking sub-detector geometries has been developed. Additional modules allowing for particle identification using time-of-flight and ionizing energy loss information have been implemented. A jet flavour identification algorithm based on a graph neural network architecture and exploiting all available particle level information has been developed. The impact of different detector design assumptions on the flavour tagging performance is assessed using the FCC-ee IDEA detector prototype.

A.2 Randomization 12
References 12

1 Introduction

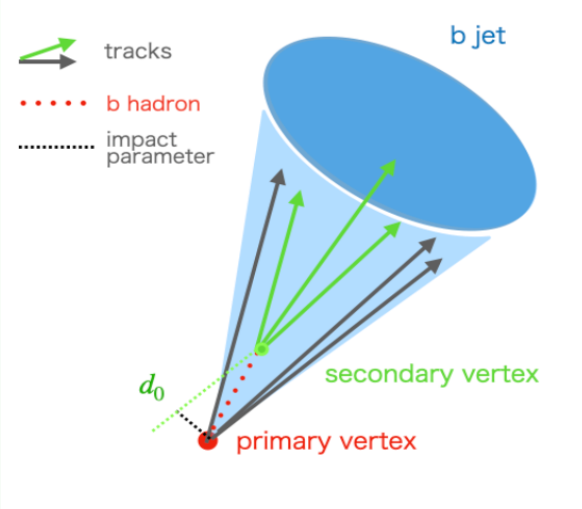
Precision measurements of standard model (SM) parameters are key objectives of the physics program of future lepton and hadron machines [1–6]. In particular, the measurement of the Higgs couplings to bottom (b) and charm (c) quarks, and gluons (g) [7–13], the Higgs self-coupling [14] and the precise characterisation of top quark properties, such as the top quark mass [15] and its electroweak couplings [16,17] require an efficient reconstruction and identification of hadronic final states. Being able to efficiently identify the flavour of the parton that initiated the formation of a jet, known as jet flavour

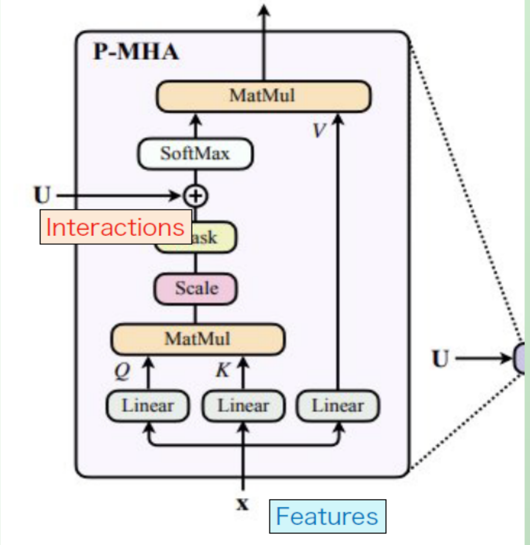
https://link.springer.com/article/10.1140/epjc/s10052-022-10609-1

Input variables

- **Features** (for each track/neutral)
  - Impact Parameter (6): Distance between primary vertex and track (2D/3D)
  - Particle ID (6) : Each particle's character is expressed as 0 or 1. (e, mu, charged hadron, gamma, neutral hadron)
  - Kinematic (4) : particle energy/jet energy etc.
  - Track Errors (15) : covariant matrix
  - Jet Distance (2) : Distance between jet axis and each track (2D/3D)

- **Interactions**
  - Kinematic variables (e.g. pt and mass) calculated from any pair of particles are added as interactions
  - Treated as bias to the attention





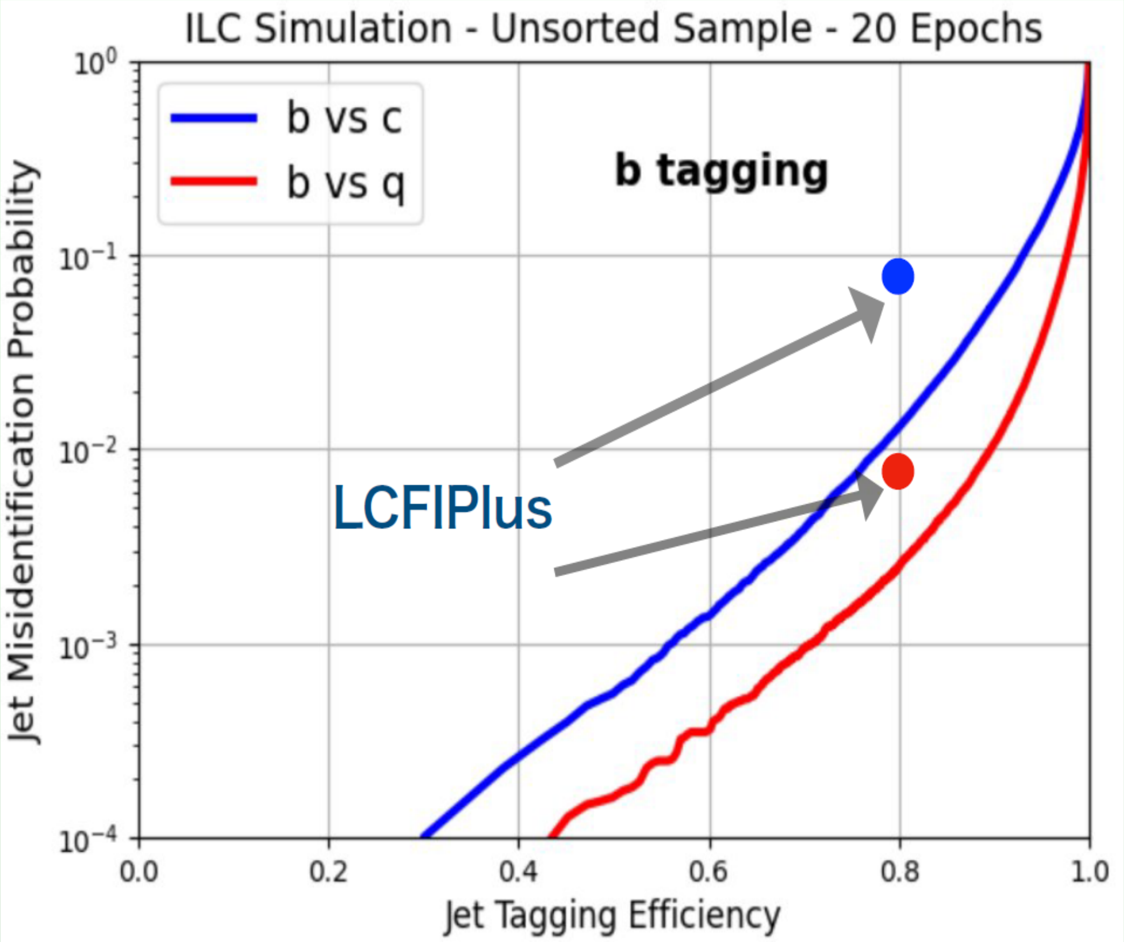
Compare LCFIPlus and ParT (ILD full simulation)

- 91 GeV data from ILD was used.
- The performance is greatly improved over LCFIPlus.

About 7.8 times

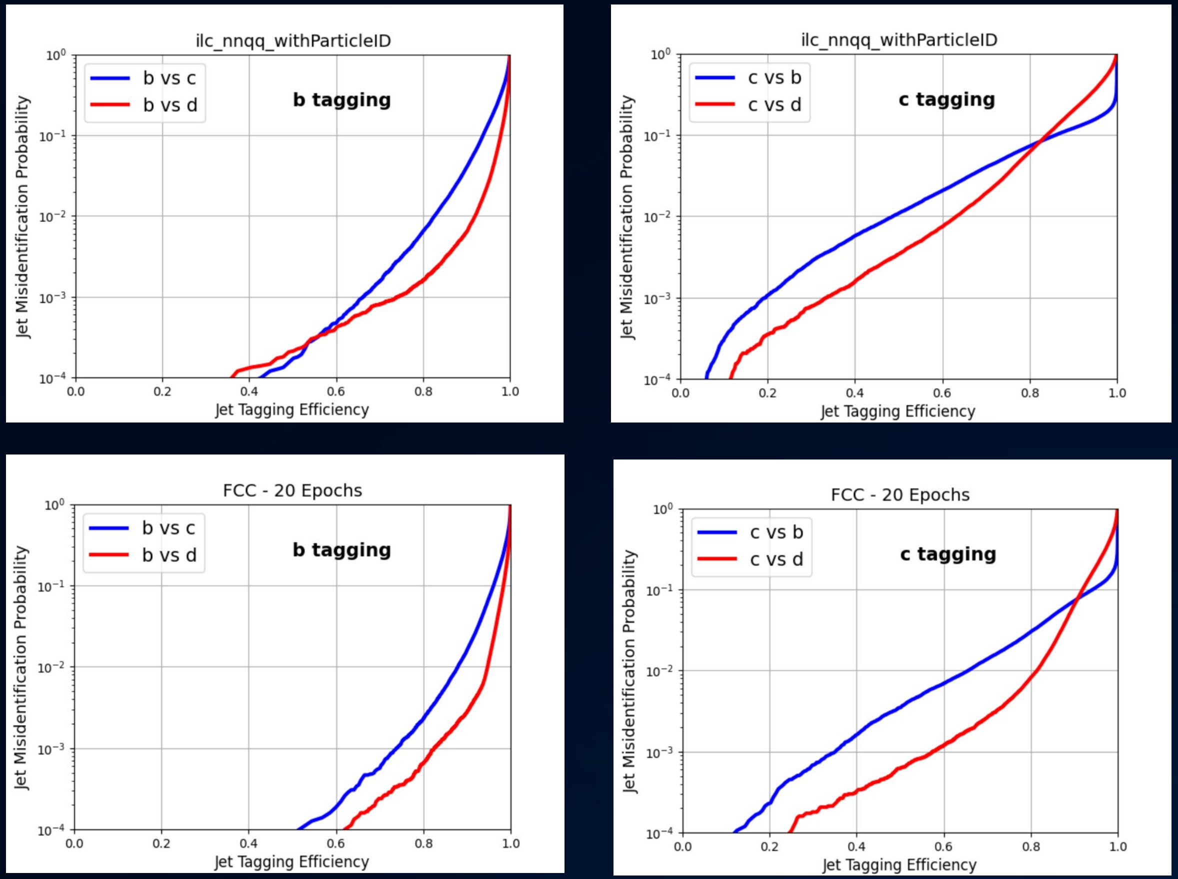
	b-tag 80% eff.		c-tag 80% eff.	
Method	c-bkg acceptance	uds-bkg acceptance	b-bkg acceptance	uds-bkg acceptance
LCFIPlus	10%	1%	10%	2%
ParT	1.29%	0.25%	1.02%	0.43%

Performance of ParT



Comparison with FCC data[3]

- Trained with same condition as ILD data for fair comparison. (800k data size, 20 epochs, etc.)
- FCC data has ~ 3 times the performance compared to ILD data.
- Possible cause of the difference:
  - Particle ID: too pessimistic for ILD
  - Definition of some variables
    - Theta, phi etc.
  - Difference on full and fast sim
    - Especially different on tails of distributions
  - Assumed detector resolution (?)



Data	Particle ID	Impact Parameters	Jet Distance	Track Errors	c-bkg acceptance @ b-tag 80% eff.	b-bkg acceptance @ c-tag 50% eff.
ILD (vvqq 250 GeV)	●	●	●	●	0.64%	1.09%
FCC	●	●	●	●	0.23%	0.35%

ILD (vvqq 250 GeV) vs. FCC with partial variables

800 kjet for training, 20 epochs

					c-bkg acceptance @ b-tag 80% eff.		b-bkg acceptance @ c-tag 50% eff.	
Plot Index	Particle ID	Impact Parameters	Jet Distance	Track Errors	ILD	FCC	ILD	FCC
(1)	●	●	●	●	0.64%	0.23%	1.09%	0.35%
(2)	✗	●	●	●	0.62%	0.47%	1.14%	0.64%
(3)	✗	●	●	✗	0.71%	0.24%	1.24%	0.35%
(4)	✗	●	✗	●	0.63%	0.75%	1.19%	0.80%
(5)	✗	●	✗	✗	0.79%	0.77%	1.28%	0.80%
(6)	✗	✗	●	●	9.69%	2.64%	6.91%	1.58%

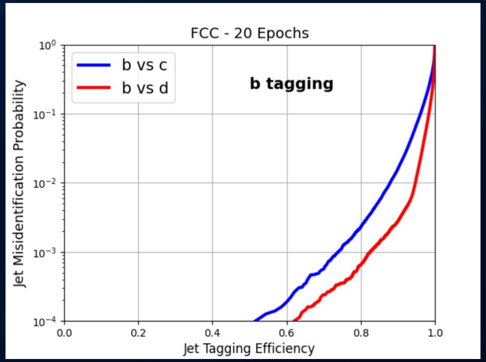
Observations:

1. PID gives significant effect on FCCee, not ILD (due to easy PID in ILD)
2. Track errors are rather harmful in FCCee
3. Difference on b-tag is small with only impact parameters (5), but still see difference in c-tag
4. (of course) significantly losing performance without impact parameter (but still ~ LCFIPlus)

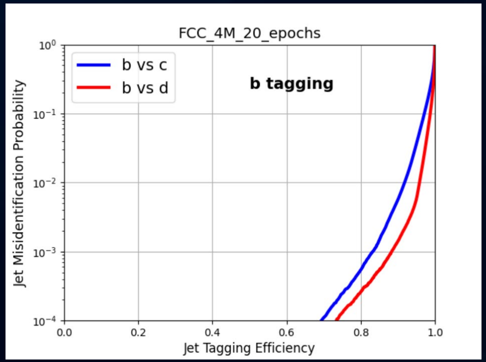
To be updated

Sample size affects performance (FCCee sample)

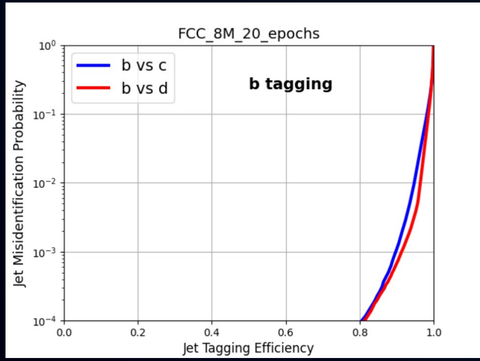
(1)



(2)



(3)



Plot Index	Particle ID	Impact Parameters	Jet Distance	Track Errors	Training Sample size	c-bkg acceptance @ b-tag 80% eff.	b-bkg acceptance @ c-tag 50% eff.
(1)	●	●	●	●	800k	0.23%	0.35%
(2)	●	●	●	●	4M	0.054%	0.20%
(3)	●	●	●	●	8M	Unreasonably good, TBC	

- Training performance significantly improved with bigger data sample size
- Training sample size change of FCC data:

  800k -> 4M : 4 times better performance (b-tagging)

  4M -> 8M: 5 times better performance (b-tagging)
- This non-linearity of increase in performance should be further investigated.
- Bigger data size of ILD should be obtained for better performance, as well as comparison with FCC data for further investigation on its behaviour.

Fine tuning

Two objectives
- Pretrained with fast sim and fine-tune with full sim
- Pretrained with large central production and fine-tune with dedicated physics samples in each analysis

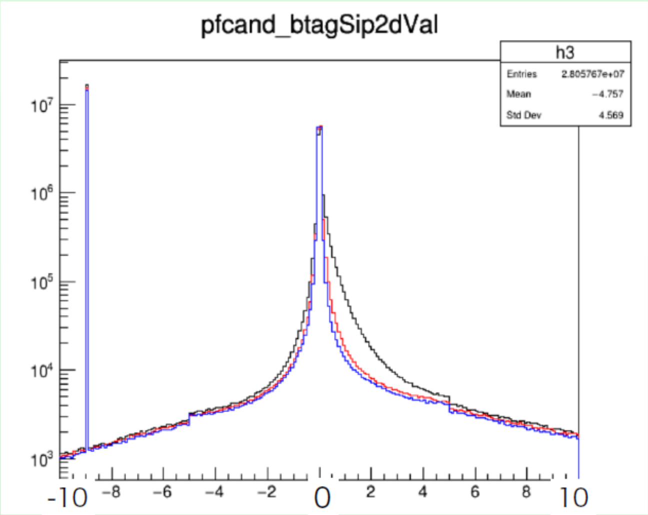
							c-bkg acceptance @ b-tag 80% eff.		b-bkg acceptance @ c-tag 50% eff.	
Particle ID	Impact Parameters	Jet Distance	Track Errors	Fine-Tuning Sample	Training Sample	Similar theta/phi ?	No Fine-Tuning	With Fine-Tuning	No Fine-Tuning	With Fine-Tuning
✗	●	●	●	FCC 240 GeV (8M)	ILD 250 GeV (800k)	✗	0.62%	1.37%	1.14%	1.95%
✗	●	●	●	FCC 240 GeV (8M)	ILD 250 GeV (800k)	●	1.77%	1.32%	2.22%	2.01%
●	●	●	●	ILD 250 GeV (800k)	ILD 91 GeV (80k)	●	4.49%	0.97%	3.79%	1.53%

- Use result of 8M FCC data to train ILD 800k data
- Improves performance only when setups are similar
- Training of same setup (pretrain ILD 91 GeV data with ILD 250 GeV data) gives best performance
- Further investigation should be conducted on how to maximise the outcome for fine-tuning between different data sets

Handling of neutral particles (input node)

- Neutral Particle has been set to -9 for track among the many features variables.
- To avoid embedding (linear, GELU) mixed with Track particles, we performed embedding separately before training, and observed a performance improvement of ~8%.

Neutral's data is gathered to -9

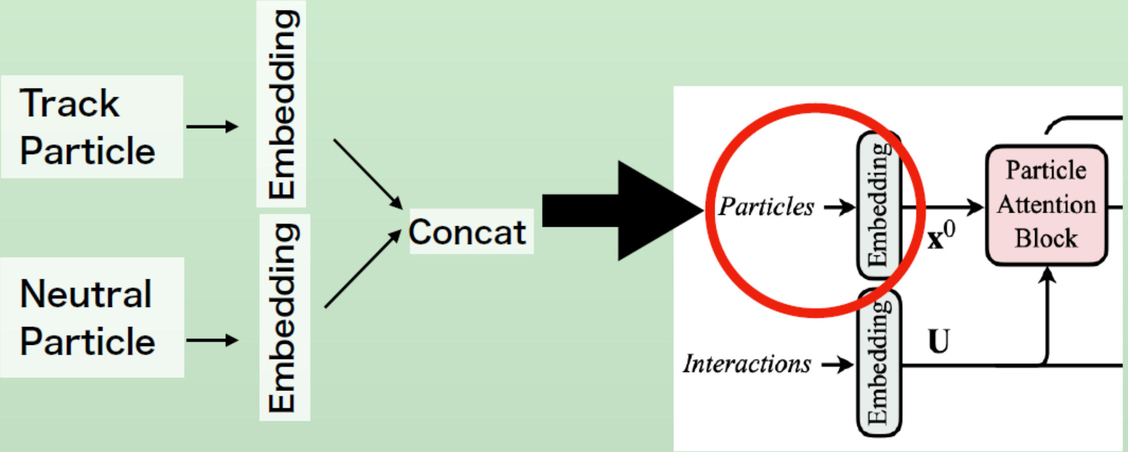


black ; b, red ; c, blue ; d

Learning for ILD data

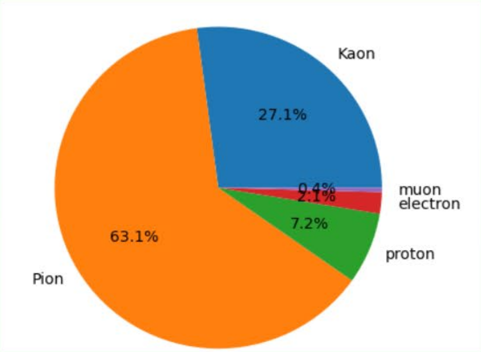
	b-tag 80% eff. c-bkg acceptance (%)	c-tag 80% eff. b-bkg acceptance (%)
Without dividing	0.518	6.60
Dividing and embedding	0.476	6.20

About 8%

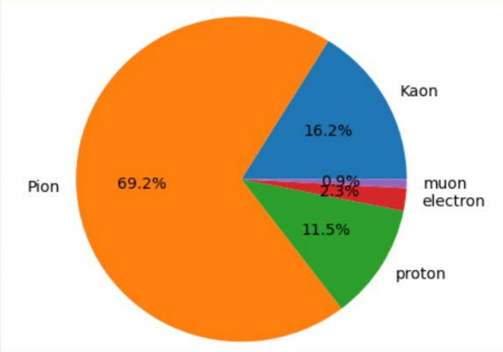


Strange tagging

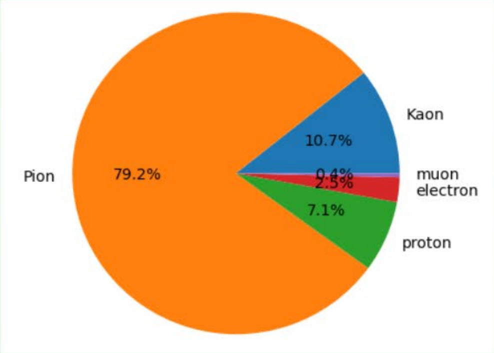
- Tagging high-momentum kaon in jet is a clue to strange jets
  - Contamination from g→ss give relatively low momentum
- dE/dx is essential for Particle ID in ILD
  - As well as ToF, but only effective in low energy tracks (which are less important in strange tagging)
- Using newly-developed comprehensive PID
  - Giving much better separation than previous PID



H->ss



H->gg

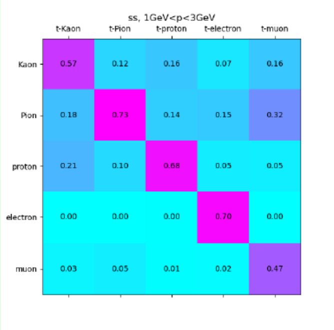


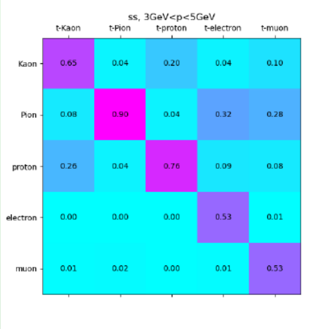
H->dd

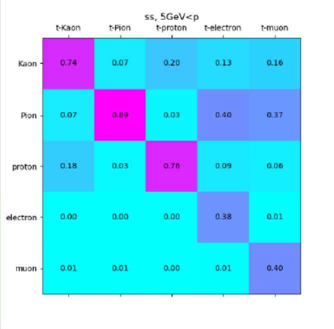
Particle ID (truth) ratio (p>5GeV)

- Strange jets have more Kaons
- Down jets have more Pions

11



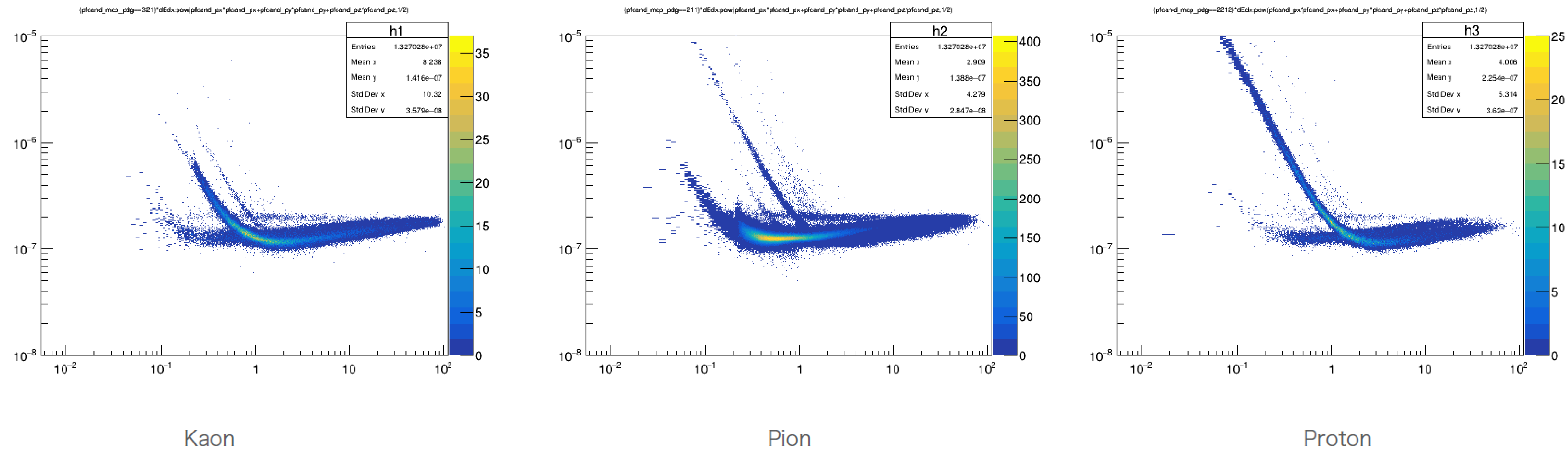




Progress in strange tag

	s vs c	s vs g	s vs u
0.8 efficiency	0.138	0.288	0.466

Current performance with ParT
(under investigation yet)

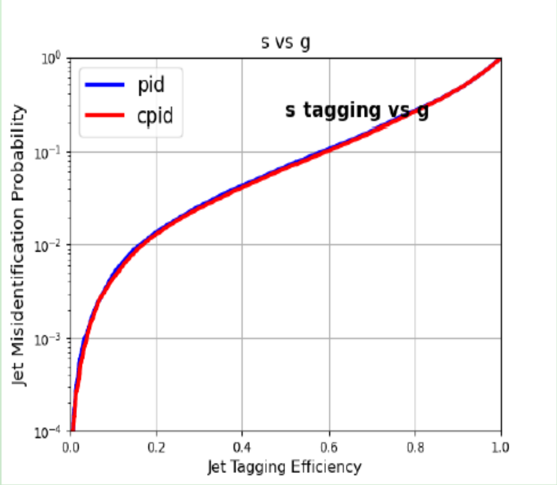


dE/dx inside strange jets (separated by MC PID)

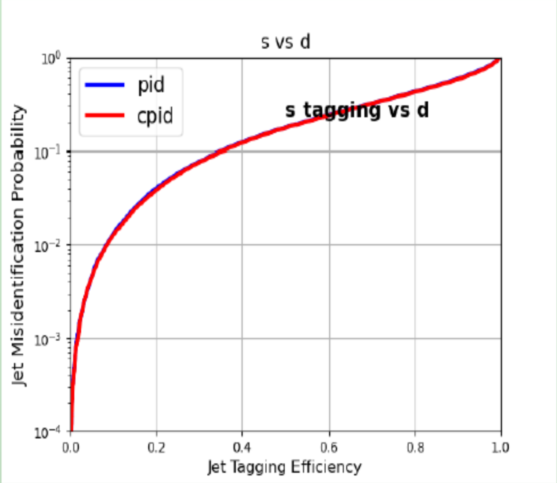
Strange tagging: initial results

- Current results gives significantly worse than FCCee results
  - FCCee@ s-tag 80% eff.: g eff. ~10%, light q eff. ~30%
  - Partially because of worse (realistic?) assumption of dE/dx performance at ILD
- Do not see any difference between old PID and CPID
  - PID performance significantly different so unreasonable
- Under investigation…

	s-tag 80% eff.	
Method	g-bkg acceptance (%)	d-bkg acceptance (%)
Previous PID	26.5%	42.8%
CPID	25.7%	42.7%



s vs g



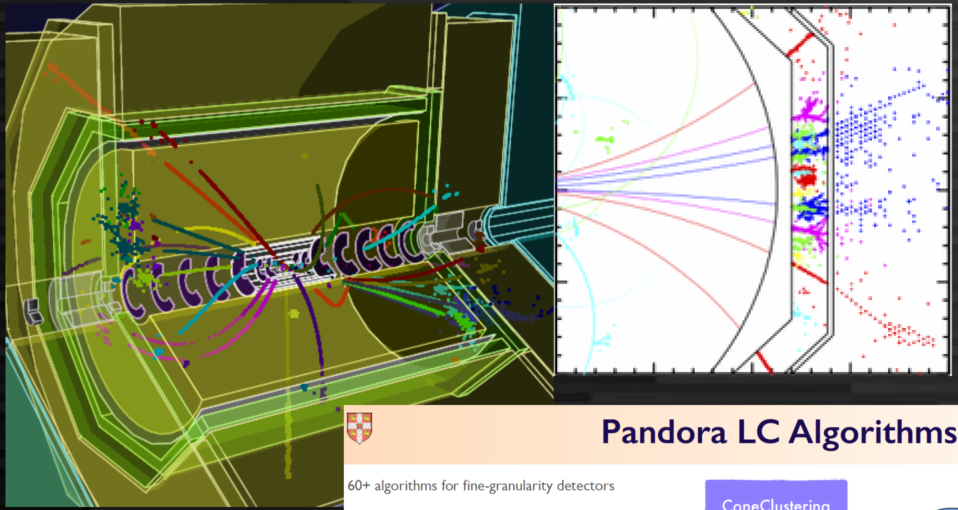
s vs d

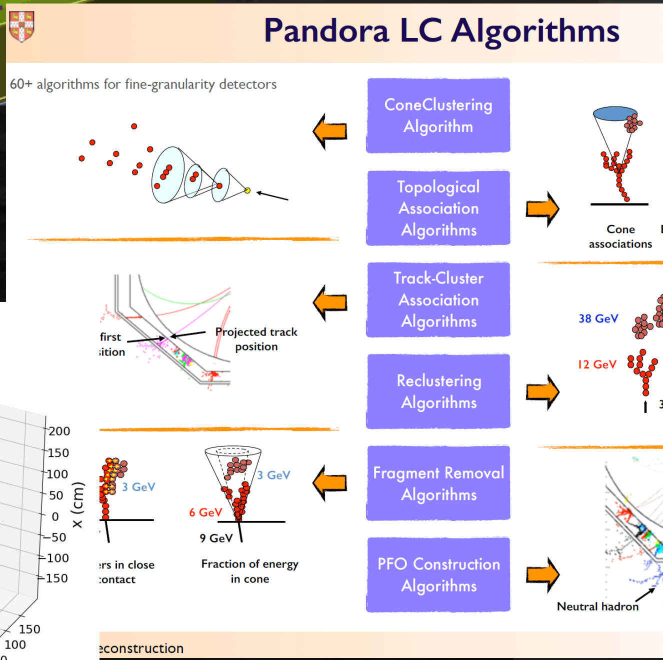
Flavor tagging: summary and plans

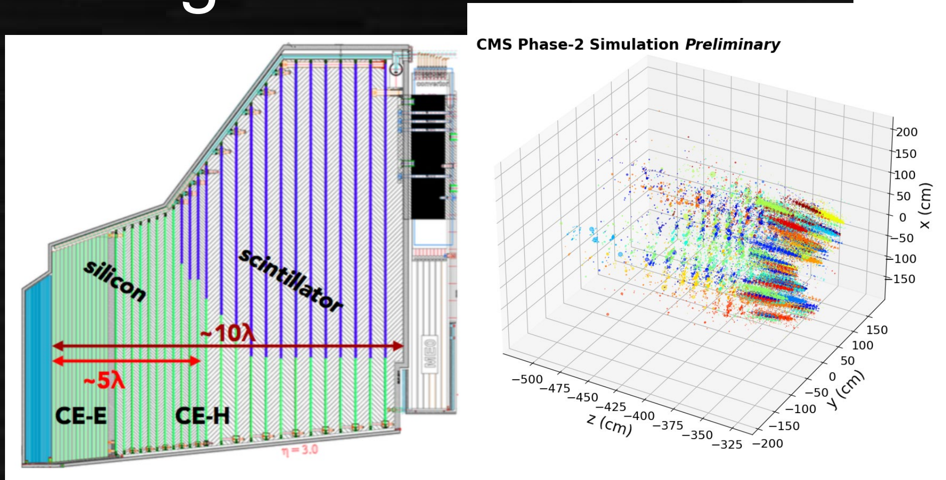
- Significantly better performance of flavor tagging with ParT
  - Implementation to the reconstruction framework foreseen to be applied to real physics analysis (time scale: this autumn)
  - Further optimization still possible
- Strange tagging under investigation
  - (Maybe technical problem) prevents high performance
  - To be fixed soon → to be used in H→ss for ECFA HF study
  - Dependence on PID performance to be investigated
    - Coming with various detector configurations

Particle flow with DNN: introduction

- Separation of cluster at calorimeter
  - Charged or neutral cluster
- Essential for jet energy resolution
- Current algorithm: PandoraPFA
  - Combination of various process
  - Not easy to optimize or adding more info
- CMS HGCal clustering
  - Similar to ILD calo
  - Good for starting point

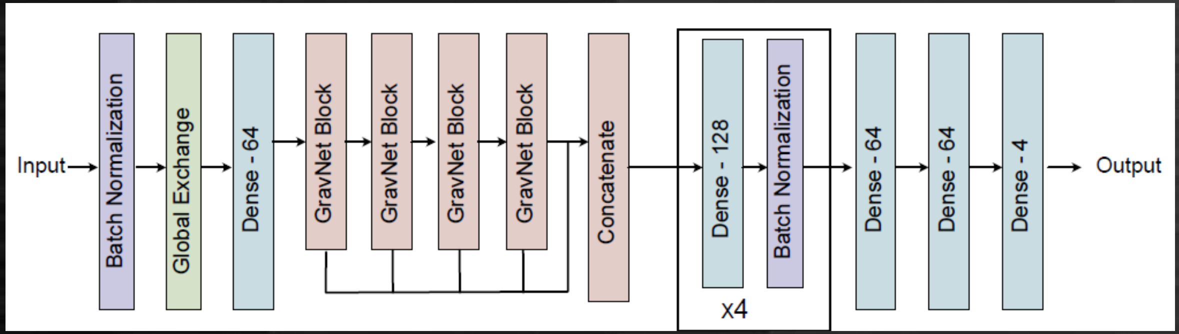






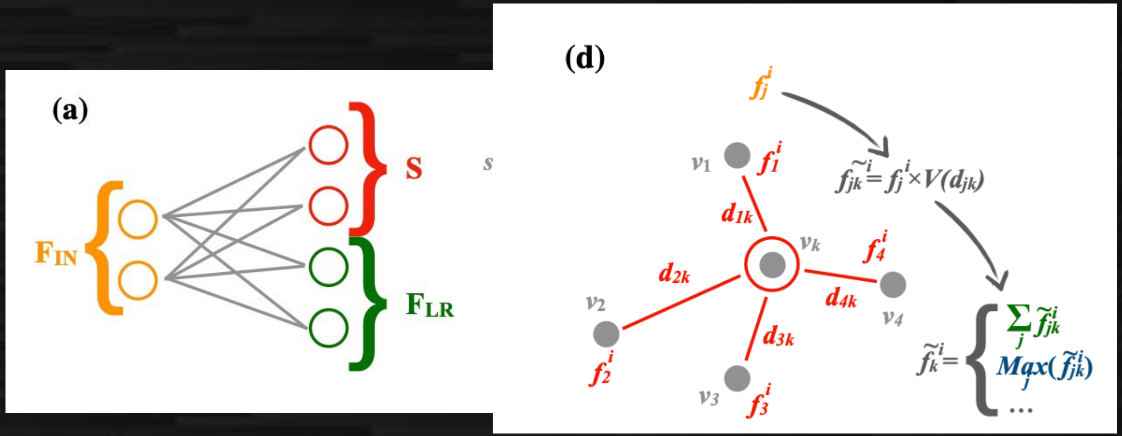
PFA: clustering algorithm

- Input: position/energy/timing of each hit
- Output: virtual coordinate and β for each hit



GravNet arXiv:1902.07987

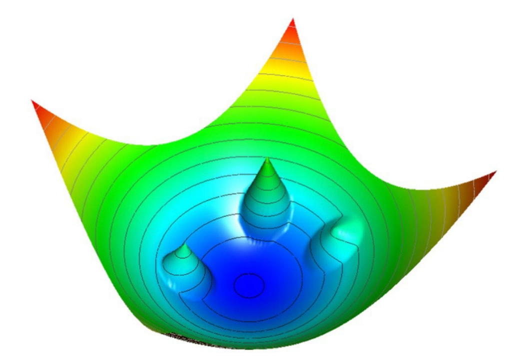
- The virtual coordinate (S) is derived from input variables with simple MLP
- Convolution using "distance" at S (bigger convolution with nearer hits)
- Concatenate the output with MLP



Object Condensation (loss function)

arXiv:2002.03605

$$L = L_p + s_C(L_\beta + L_V)$$



- Condensation point: The hit with largest β at each (MC) cluster
- L_V: Attractive potential to the condensation point of the same cluster and repulsive potential to the condensation point of different clusters
- L_β: Pulling up β of the condensation point
- L_p: Regression to output features

What we implemented: track-cluster matching

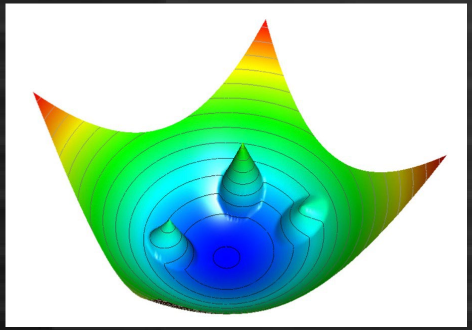
- PFA is essentially a problem "to subtract hits from tracks"
- HGCAL algorithm does not utilize track information
  - Only calorimeter clustering exists
- Putting tracks as "virtual hits"
  - Located at entry point of calorimeter
  - Having "track" flag (1=track, 0=hit)
  - Energy deposit = 0
- Modification on object condensation to forcibly treat tracks as condensation points (details next page)
  - Also modifying clustering algorithm to avoid double-track clusters

Current number of parameters: ~420K

Object condensation and our implementation

Object condensation loss function (the function to minimize)

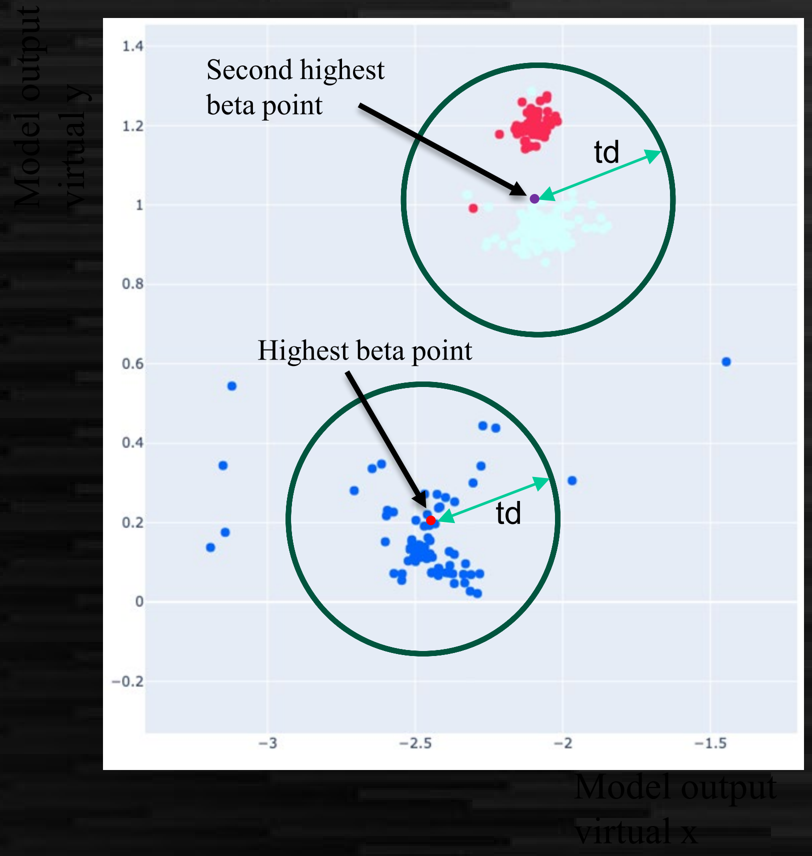
$$L = L_p + s_C(L_\beta + L_V)$$



- Condensation point: The hit with largest β at each (MC) cluster
  → For each MC cluster having a track,
  the track is forcibly the condensation point regardless of β
- L_V: Attractive potential to the condensation point of the same cluster and repulsive potential to the condensation point of different clusters (no modification)
- L_β: Pulling up β of the condensation point (up to 1)
  (no modification, but β of tracks become spontaneously close to 1)
- L_p: Regression to output features (energy etc.) → currently not used

Clustering algorithm

- Output of the network is position and β of each hit → need clustering
- Hits that are within a certain distance (td) from the highest β point assume as a cluster
- Continues clustering until all hits are clustered or β of remaining hits are below threshold (tbeta)

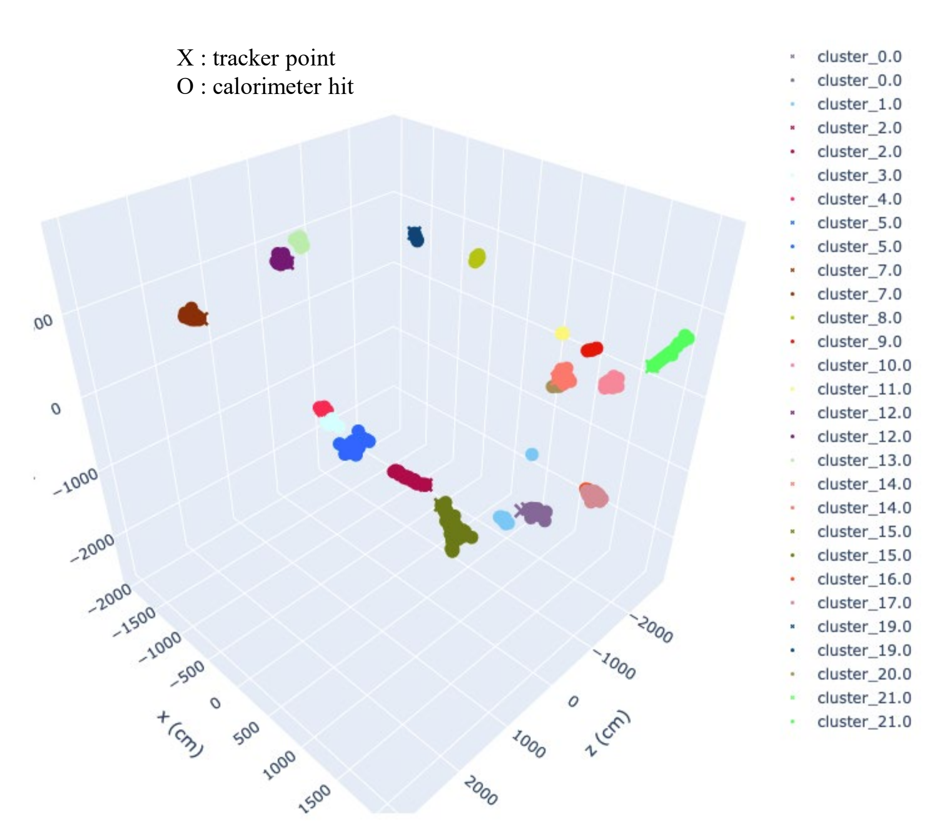


Our samples for performance evaluation

- ILD full simulation with SiW-ECAL and AHCAL
  - ECAL: 5 x 5 mm^2, 30 layers, HCAL: 30 x 30 mm^2, 48 layers
  - Taus overlayed with random direction
    - 100k events, 10 GeV x 10 taus / event → 1 million taus
    - 1M events with variable energies produced, to be tested
  - qq (q=u, d, s) sample at 91 GeV
    - ~75k events
    - Official sample for PFA calibration (other energies available)
  - Converted to awkward array stored in HDF5 format
    - A few 10 GB each

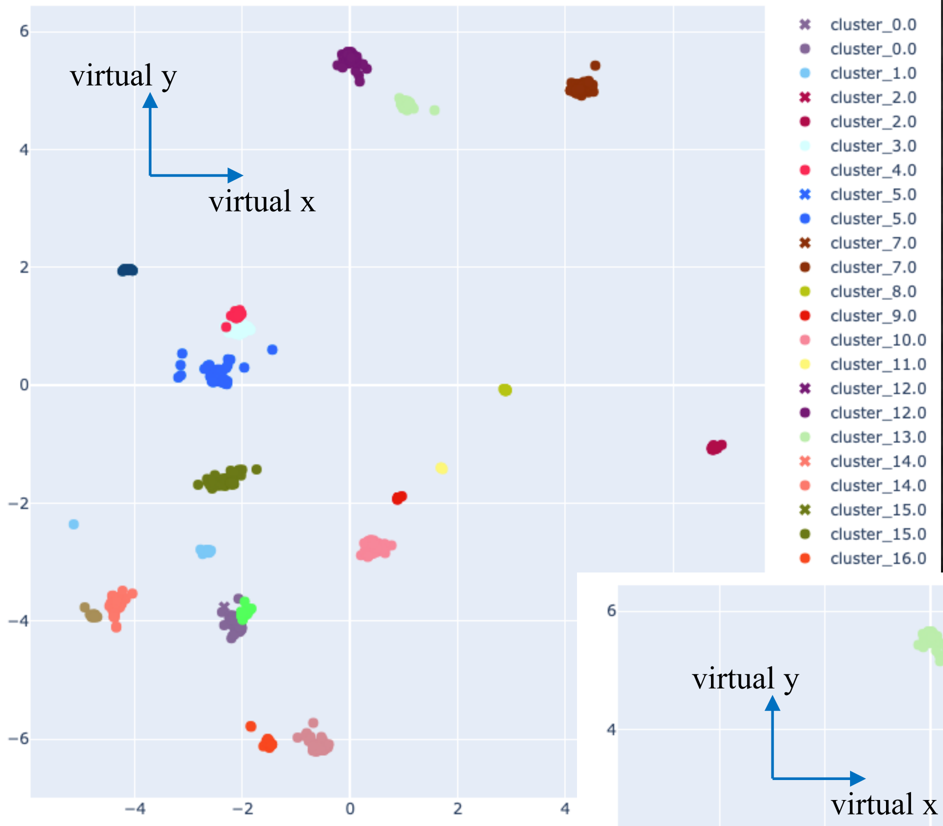
Taus: good mixture of hadrons, leptons and photons with some isolation
Good for training

Event display



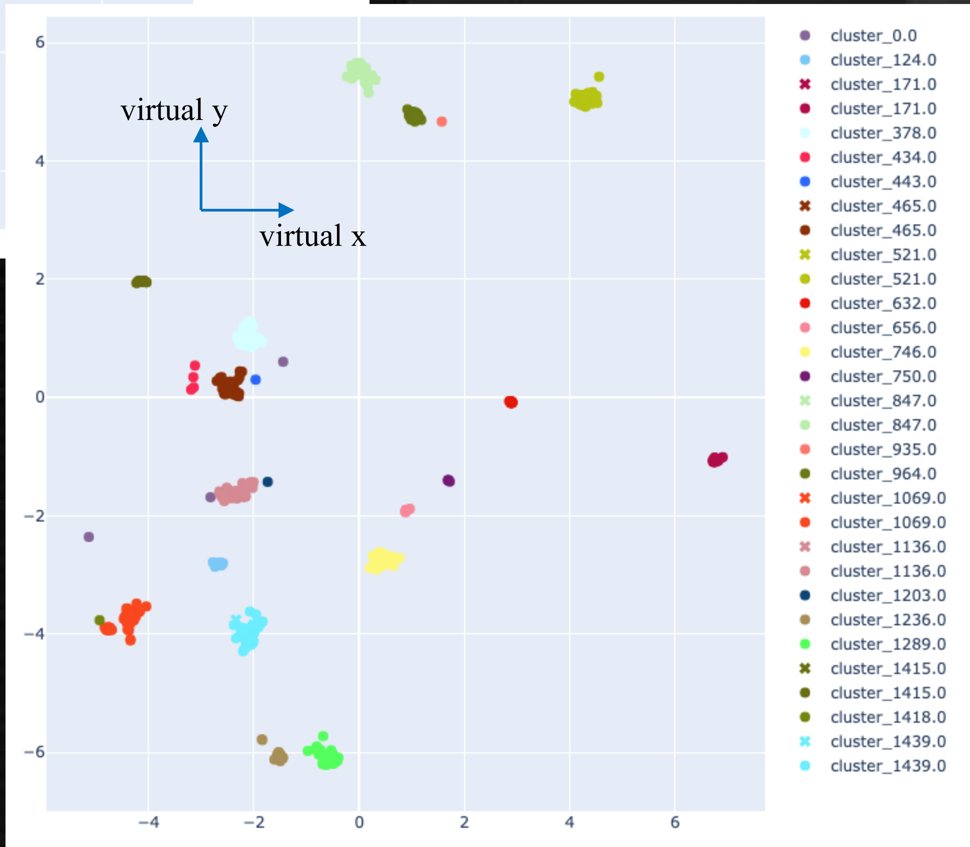
Input features
Real coordinate in detector

Colored by true clusters



Output features
Virtual coordinate

Colored by
true clusters



Colored by
reconstructed clusters

Taikan Suehara et al

Quantitative evaluation

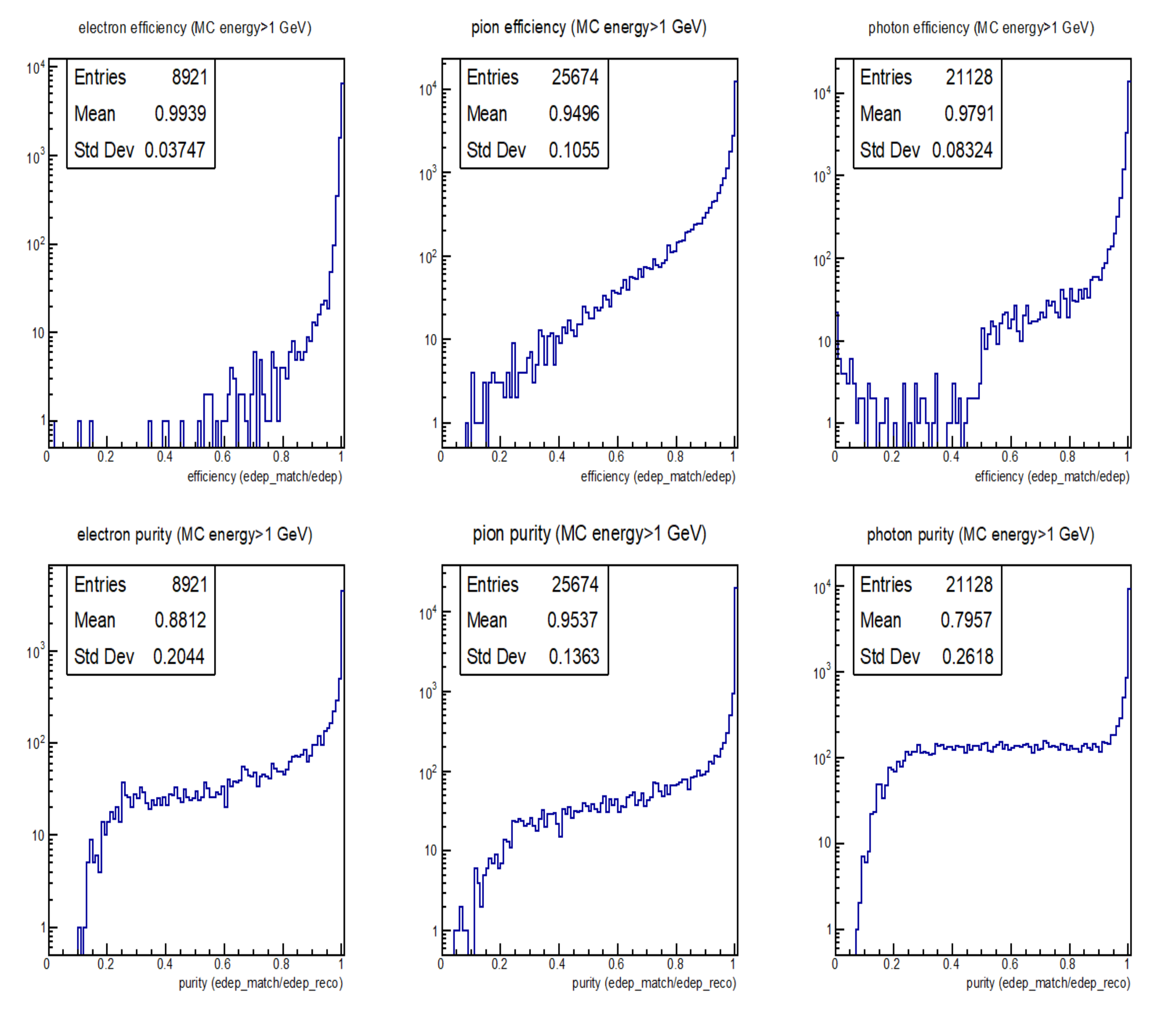
- Make 1-by-1 connection of MC and reconstructed cluster
  - Reconstructed cluster with highest fraction of hits from the MC is taken
  - Multiple reconstructed cluster may connect to one MC cluster
- Quantitative comparison with PandoraPFA
  - Compared "efficiency" and "purity" of particle flow
    - Efficiency : (reconstructed cluster energy that matches the MC cluster) / (MC cluster energy)
    - Purity : (reconstructed cluster energy that matches the MC cluster) / (reconstructed cluster energy)

Example results (ntau, GNN)

- Efficiency :
  over 90% for all particles
  slightly low in pions

- Purity :
  over 88% for all tracks
  79% for photons
  merged photons?

- Reasonably well reconstructed



Initial results (> 1 GeV)

Algorithm train/test	Electron eff.	Pion eff.	Photon eff.	Electron pur.	Pion pur.	Photon pur.
GravNet 10 taus/10 taus	99.4%	95.0%	97.9%	88.1%	95.4%	79.6%
GravNet 10 taus/jets	91.3%	88.1%	89.8%	62.2%	81.3%	64.4%
GravNet jets/jets	90.5%	89.7%	87.1%	65.6%	83.3%	70.9%
PandoraPFA 10 taus	99.3%	94.0%	99.1%	91.8%	94.6%	97.2%
PandoraPFA jets	80.2%	90.4%	79.0%	75.0%	90.6%	77.7%
PandoraPFA jets (ILCSoft)	96.7%	95.5%	96.4%	97.1%	90.4%	97.7%

Comparable performance on pion reconstruction on 10 taus

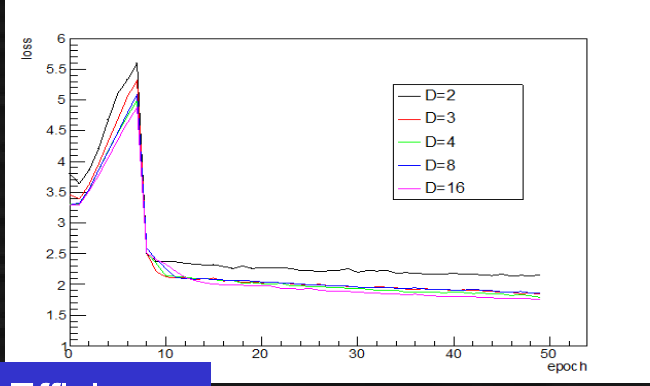
Still worse in photon reconstruction and reconstruction at jets

ILCSoft evaluation (using MC-cluster matching in ILCSoft) much better in PandoraPFA

Optimization of performance

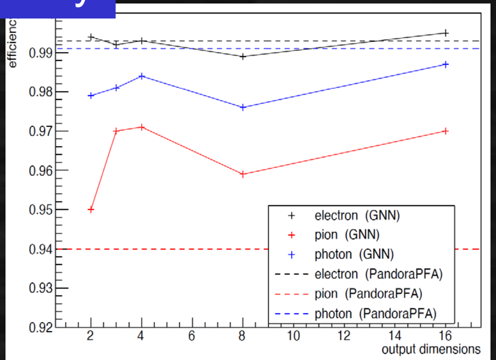
Output dimension of the coordinate

- The initial work done with output coordinate dimension D = 2 (for visibility)
- Tried D=3,4,8,16
    - D=3 much better than D=2
    - Slight improvements on D=4, 16
    - Degraded at D=8 (statistics?)

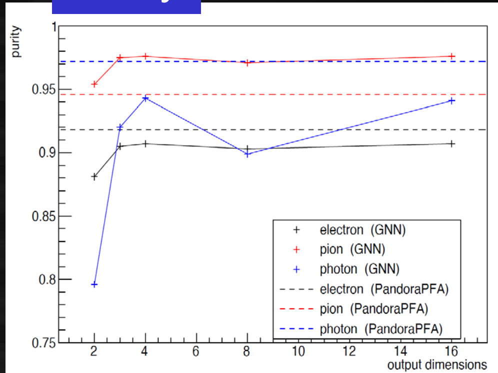


Loss function (training)

Efficiency



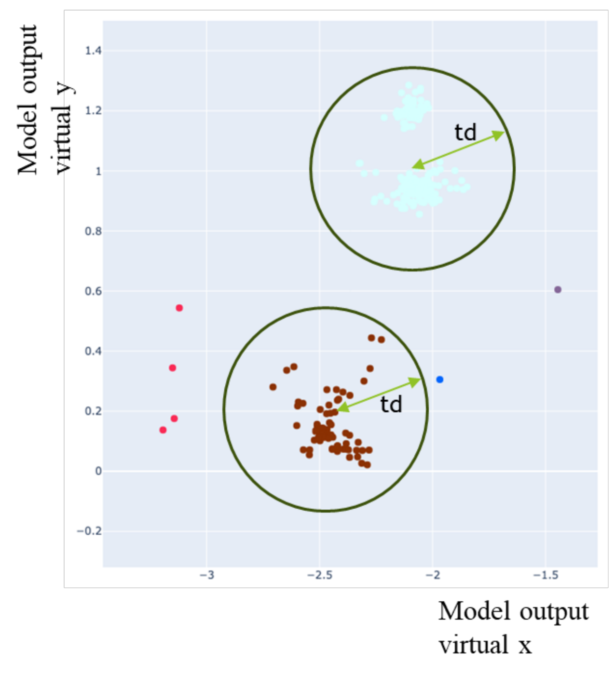
Purity

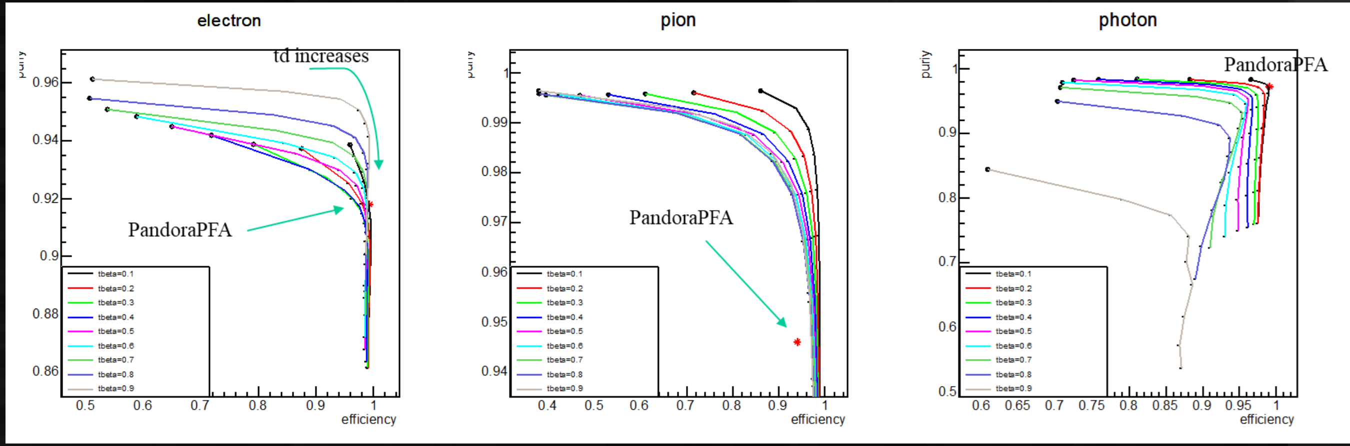


Clustering parameters (td, tbeta)

- td: radius which hits are treated as coming from the same cluster
- tbeta: threshold of beta to form clusters
- Scanning grid points (2D)

- tbeta = 0.1, td=0.3 would be taken (for ntaus)



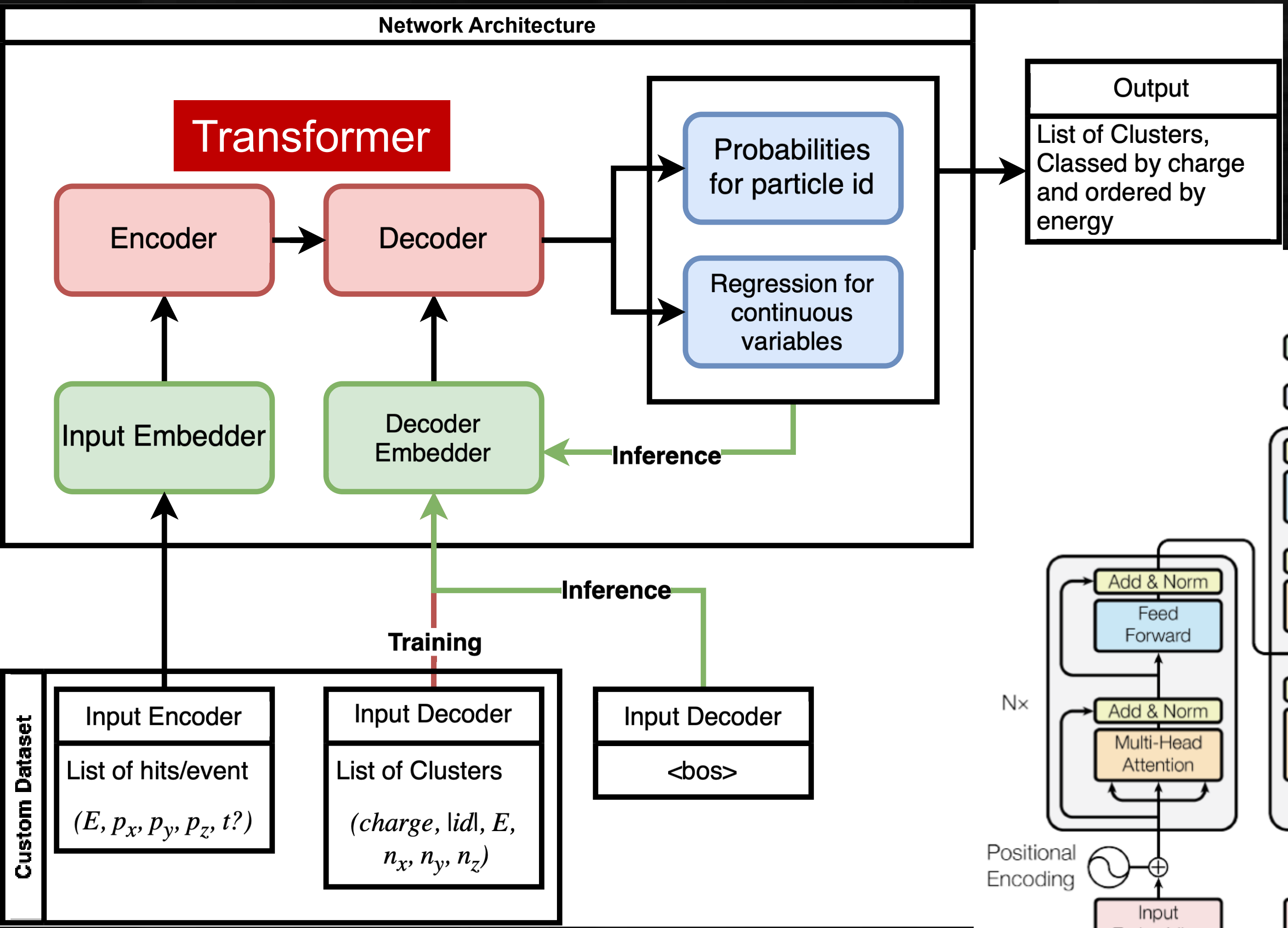


Optimized results (ntau only)

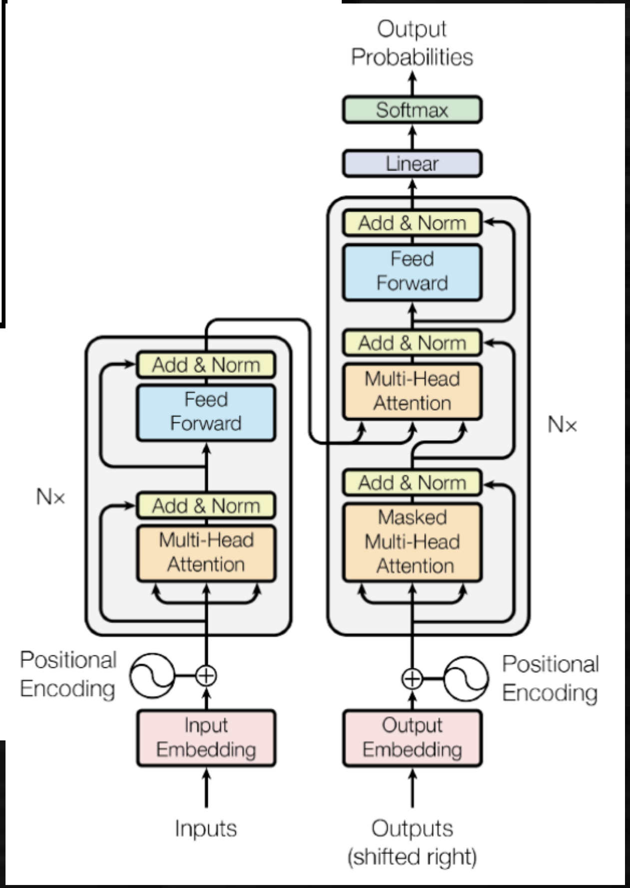
Algorithm train/test	Electron eff.	Pion eff.	Photon eff.	Electron pur.	Pion pur.	Photon pur.
GravNet (opt.) 10 taus/10 taus	99.1%	96.5%	99.0%	91.8%	98.9%	97.1%
GravNet 10 taus/jets						
GravNet jets/jets						
PandoraPFA 10 taus	99.3%	94.0%	99.1%	91.8%	94.6%	97.2%
PandoraPFA jets	80.2%	90.4%	79.0%	75.0%	90.6%	77.7%
PandoraPFA jets (ILCSoft)	96.7%	95.5%	96.4%	97.1%	90.4%	97.7%

Better performance on pion reconstruction while comparable performance on electron and photons
→ Promising! (more results will come)

More NLP-like model: transformer



Planned structure for PFA



Transformer

Transformer: training relation among elements (hits in PFA) with (multi-head) self-attention mechanism (used in GPT etc.)

Encoder: accumulate info of all hits/tracks by transformer

Decoder:

Input cluster info one by one

Output info of next cluster

(training) MC truth clusters

(inference) just provide <bos> to derive first cluster, using output as next input until <eos> obtained

(Inspired by translation NN)

Particle flow: summary and plans

- GNN-based particle flow has possibility to replace PandoraPFA
  - Performance seems exceeded for 10 tau events (tbc in jets)
  - Difference on MC-truth definition to ILCSoft to be investigated
    - (ILCSoft uses MCParticlesSkimmed while our method uses MCParticle collection)
- Regression of cluster energy to be tried
  - Necessary for complete PFA
  - Jet energy resolution would be compared with PandoraPFA
- Possible improvements
  - Momenta of tracks currently not used (improvements of clustering possible)
  - Incorporation of timing information etc.
- Another new idea to “ask network the next cluster” being tried
  - Still not competitive, starting from simple samples (1-2 photons)

Overall summary

- High level reconstruction @ ILD has a lot of room to incorporate with DNN to improve performance
  - Also easier to use for detector optimization
- Flavor tagging with ParT significantly better than LCFIPlus
  - To be applied to physics analysis
  - Strange tagging also under investigation
- Particle flow with GNN gives competitive performance
  - Still needs optimization
  - Hope to replace PandoraPFA in ~a few years
  - NLP-like method also being investigated