NANJING UNIVERSITY

Machine learning in high energy physics at LHC

张 瑞 (南京大学)

2025年超级陶粲装置研讨会（湖南科技大学）

2025.7.2—2025.7.5

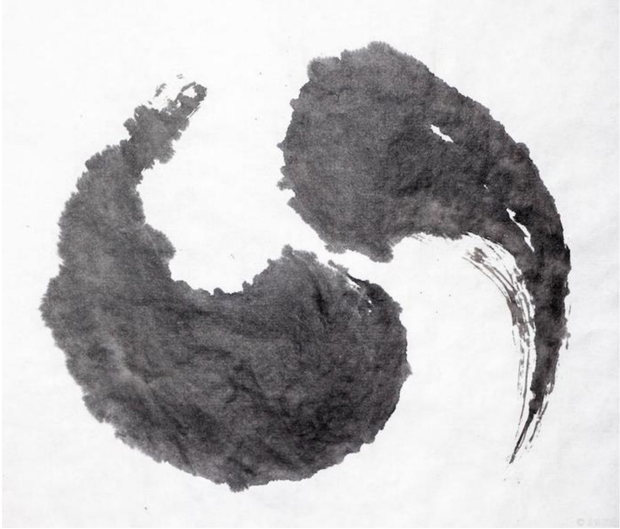
Contents

- Motivation to use machine learning (ML)
- How to map a HEP problem to a ML problem
- Examples of recent ML applications
- Summary

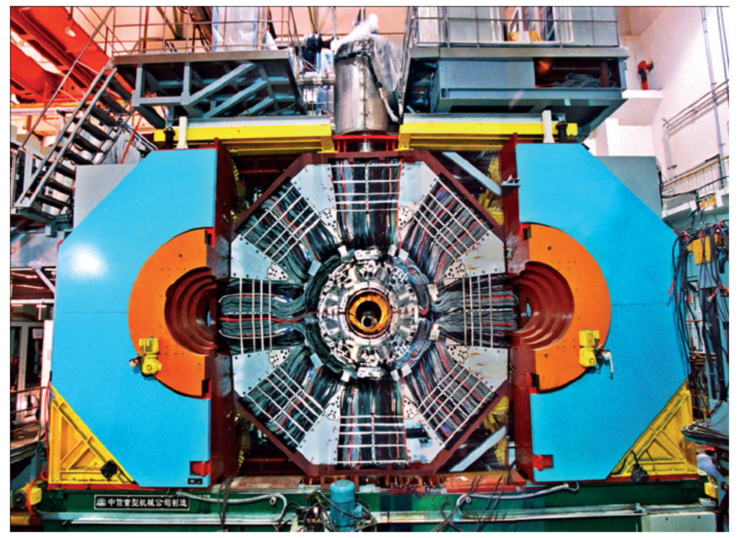
Physics analysis workflow

Experiment from reality

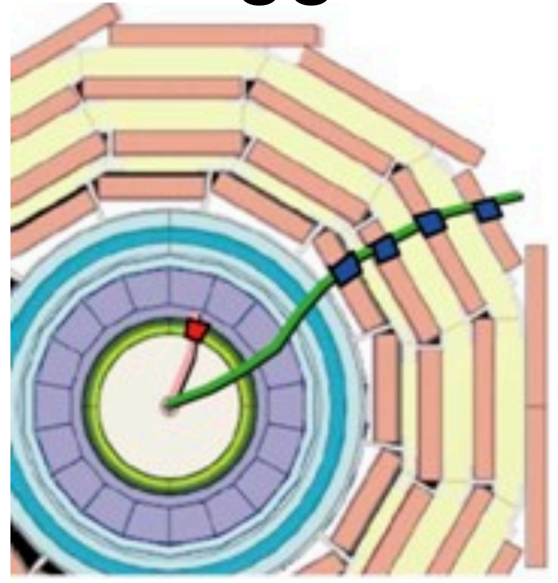
Nature



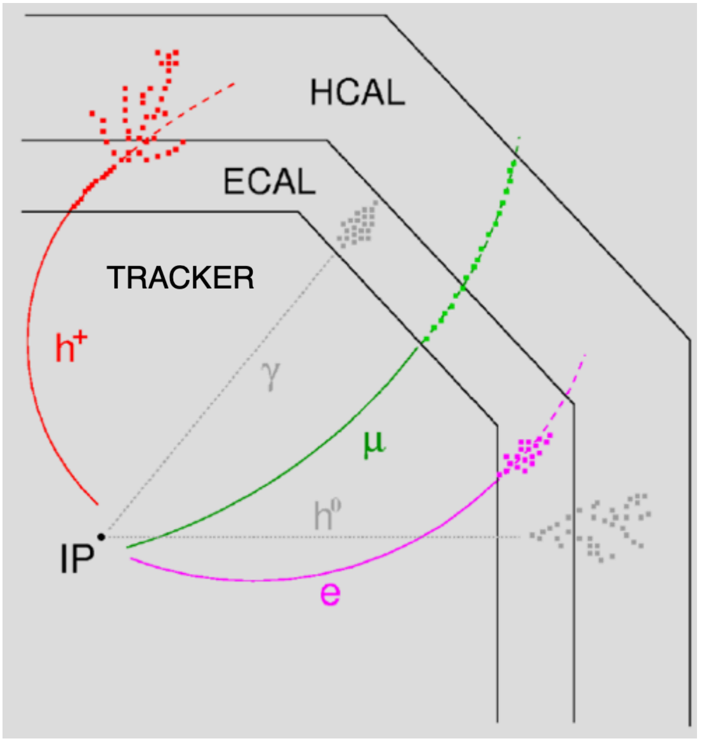
Experiment



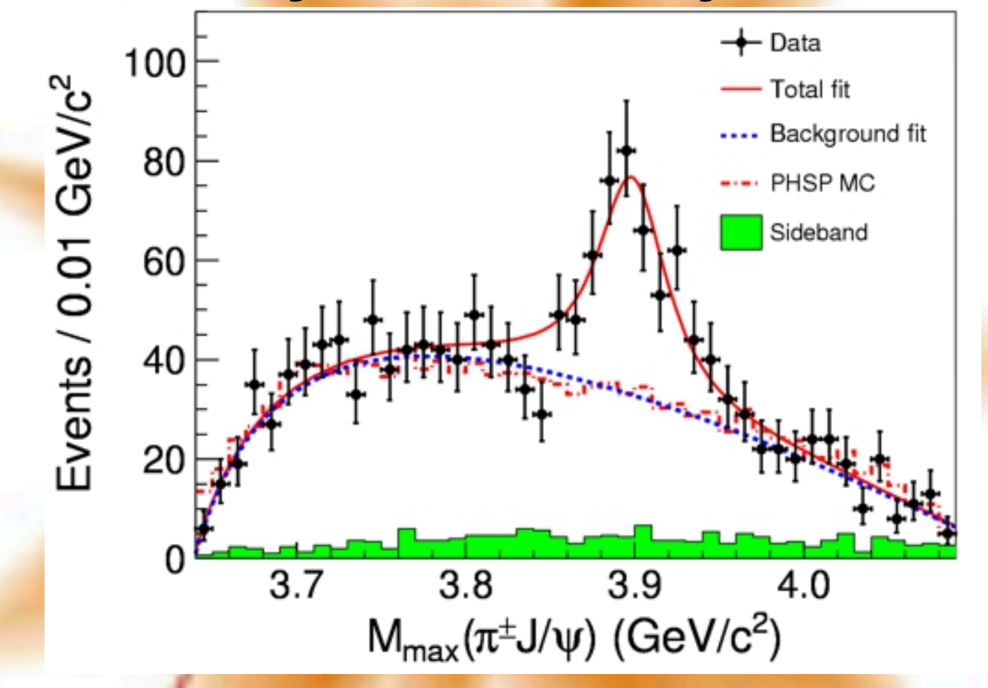
Trigger



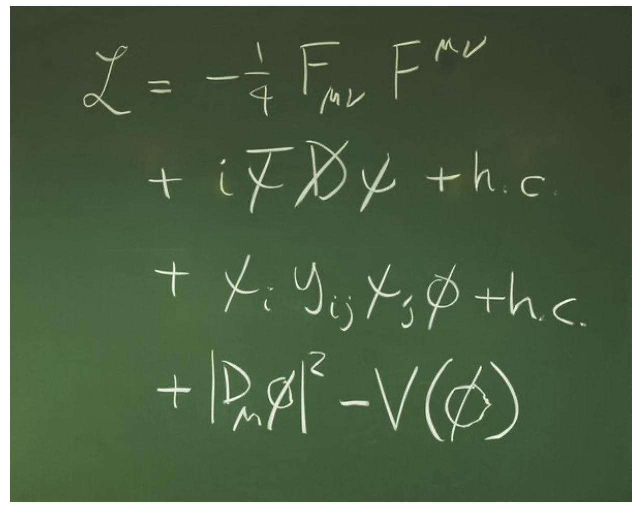
Physics object reconstruction



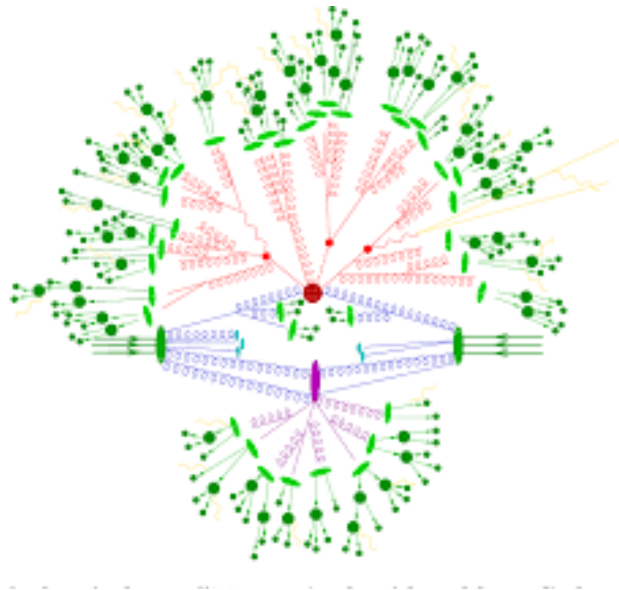
Physics analysis



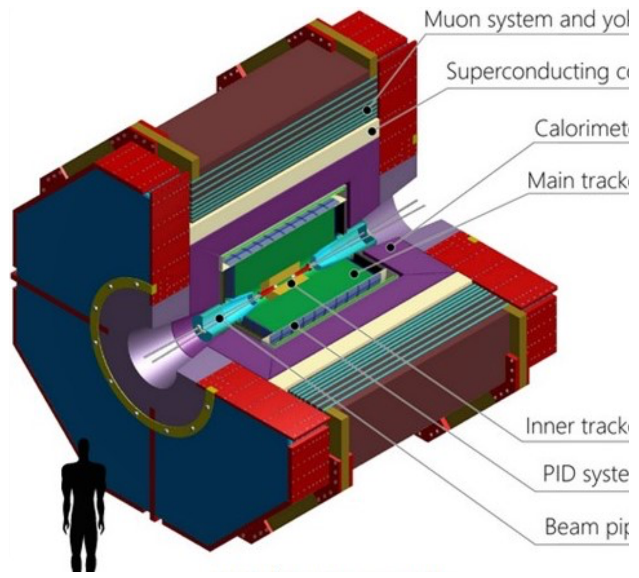
Theory of physics



Event generation



Detector response simulation



Simulation from knowledge

From data (dimension) perspective

Experiment from reality

Nature

无穷维自由度体系
O(∞)

Experiment

丰富物理学信号
O(billion)

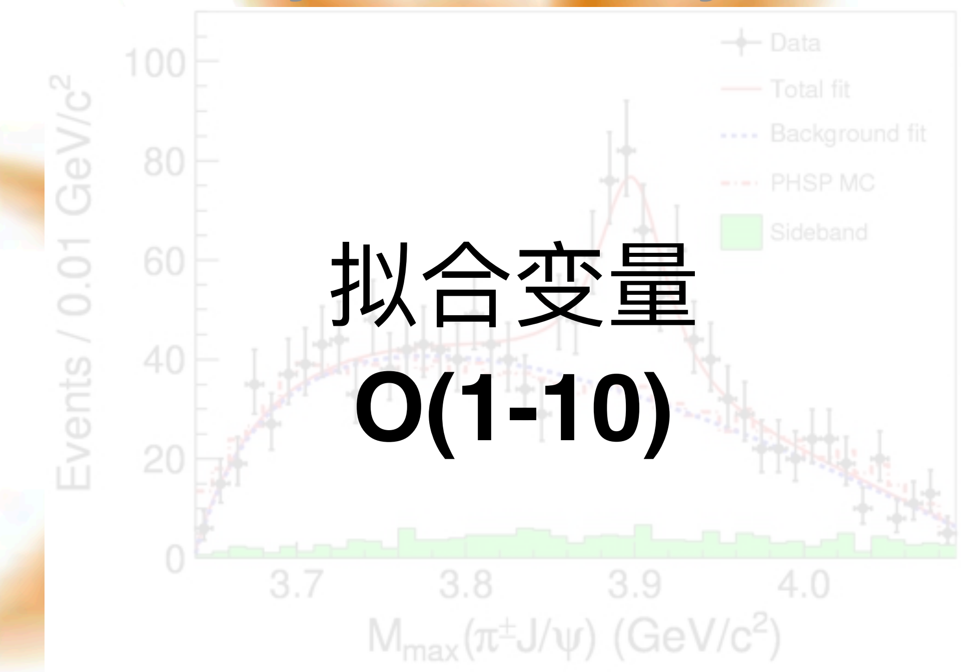
Trigger

有限电子学通道
O(million)

Physics object reconstruction

末态粒子信息
O(10-10³)

Physics analysis



拟合变量
O(1-10)

Theory of physics

第一性原理、
有效理论
O(1-10)

Event generation

丰富的末态粒子
O(10-10³)

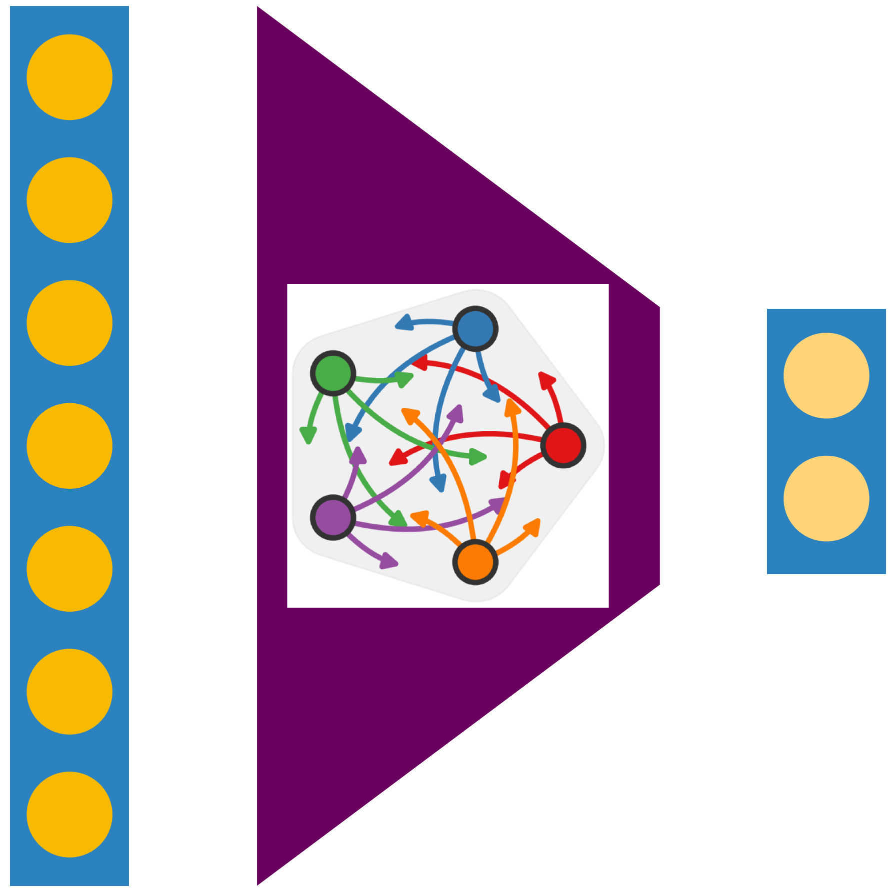
Detector response simulation

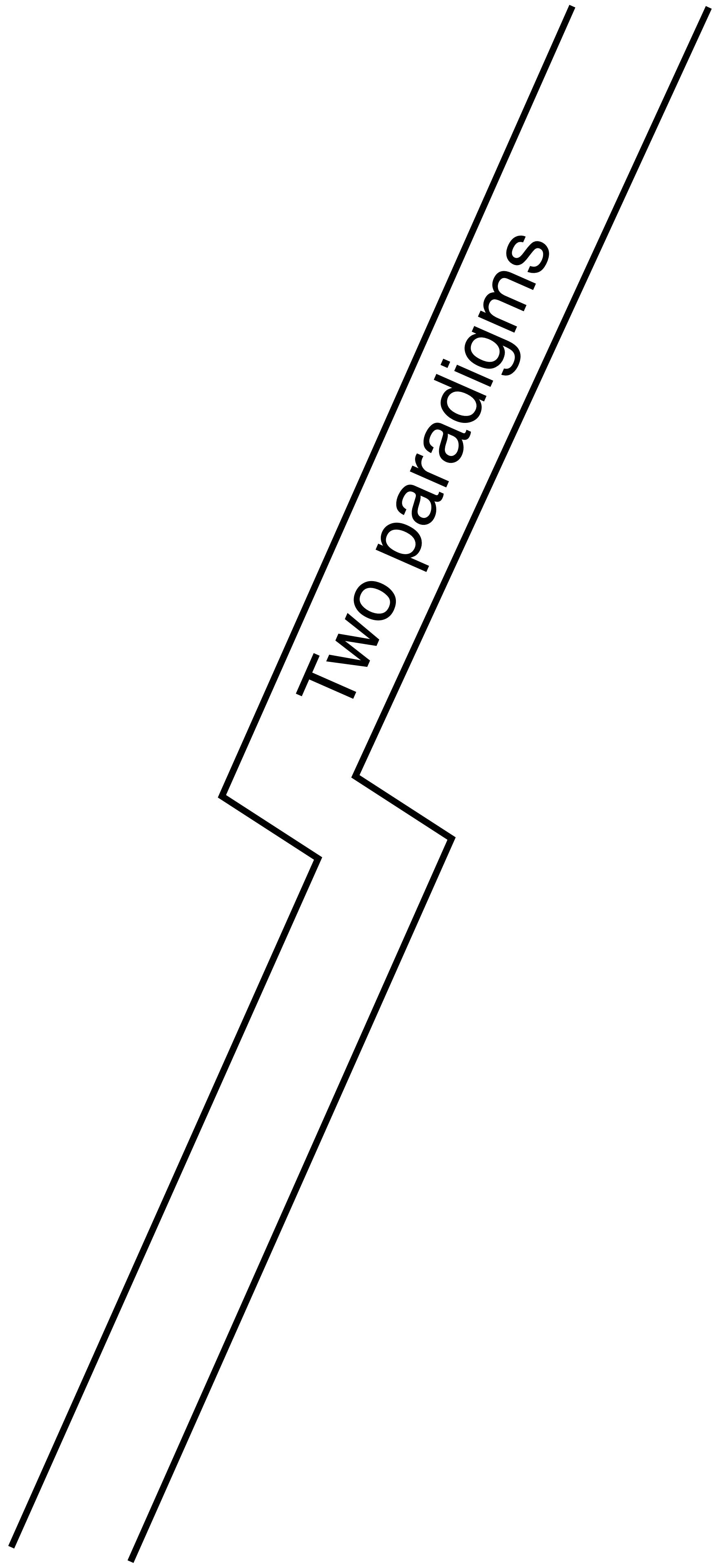
模拟探测器响应
O(million)

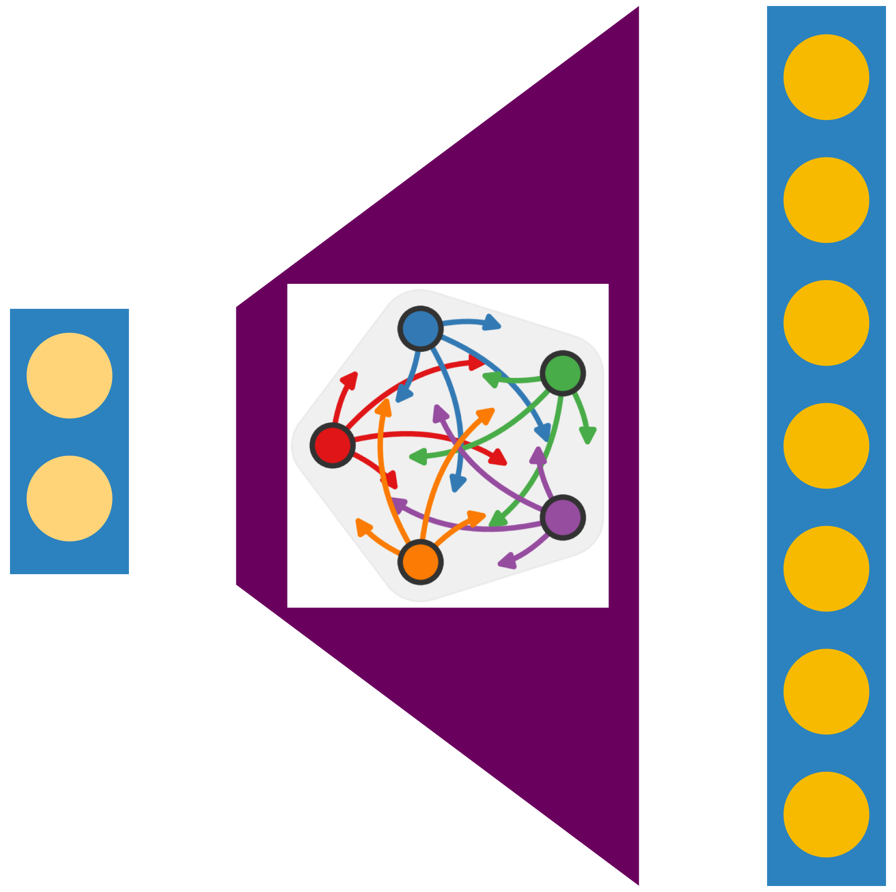
Simulation from knowledge

Can machine learning help?

- Changes of dimensionality of data is condensing / inflating information







Dimension reduction:

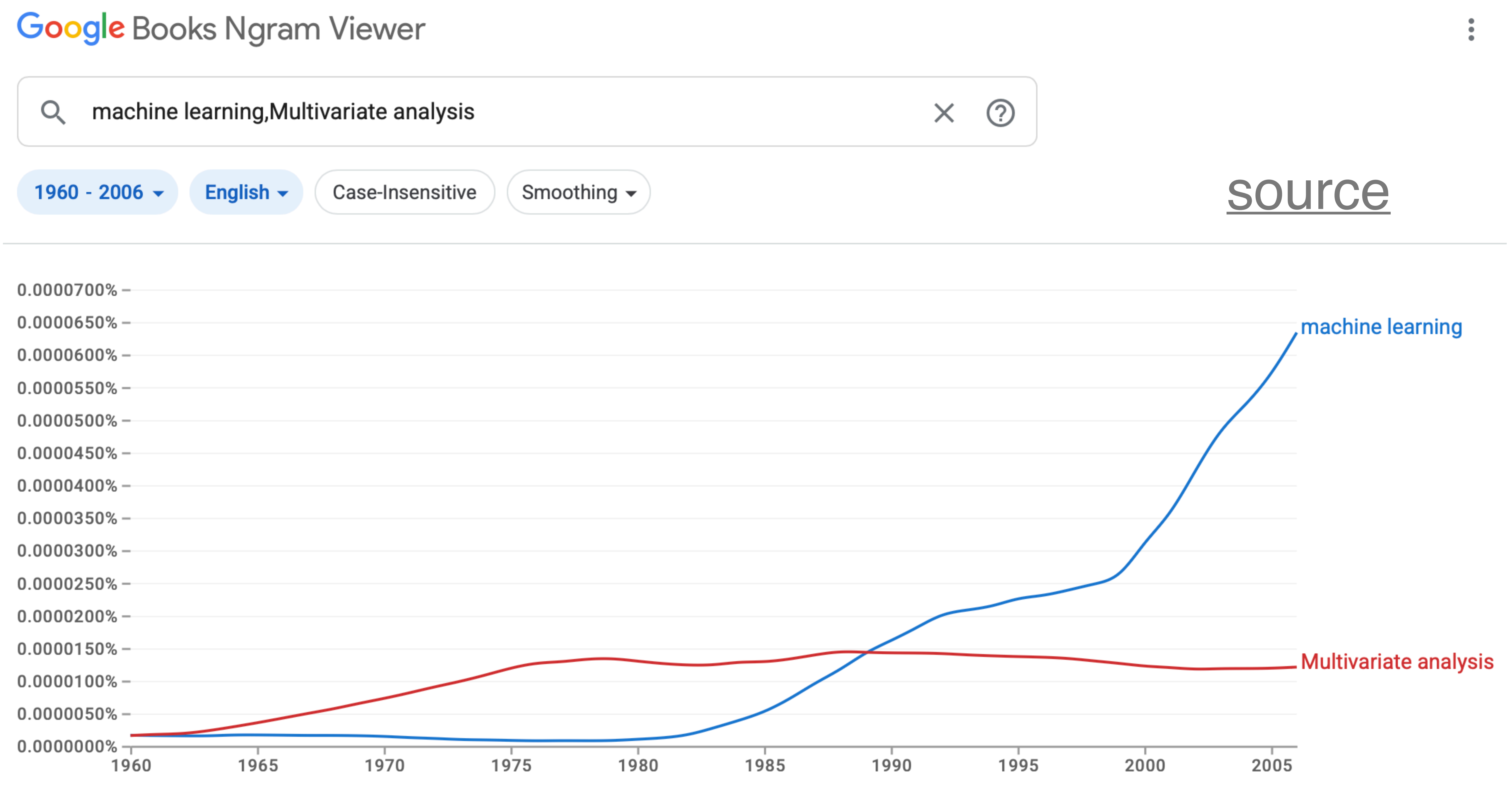
- *Trigger decision*
- *Reconstruction*
- *Sig-Bkg separation*
- *…*

Dimension increase:

- *Evt generation*
- *Detector simulation*
- *…*

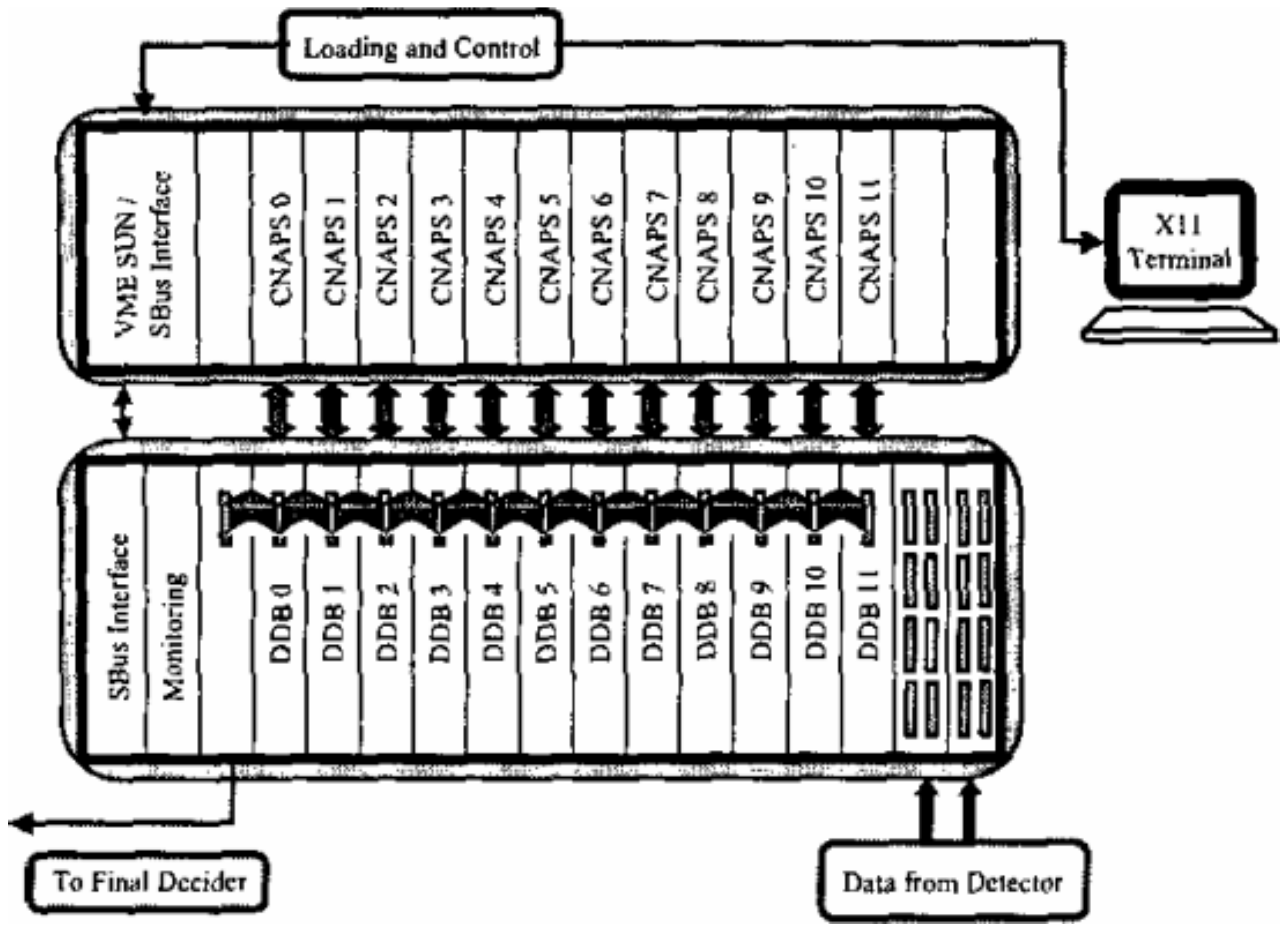
Did machine learning help?

- Machine learning (ML) is a modern term; in HEP we used to know something called "Multivariate analysis"

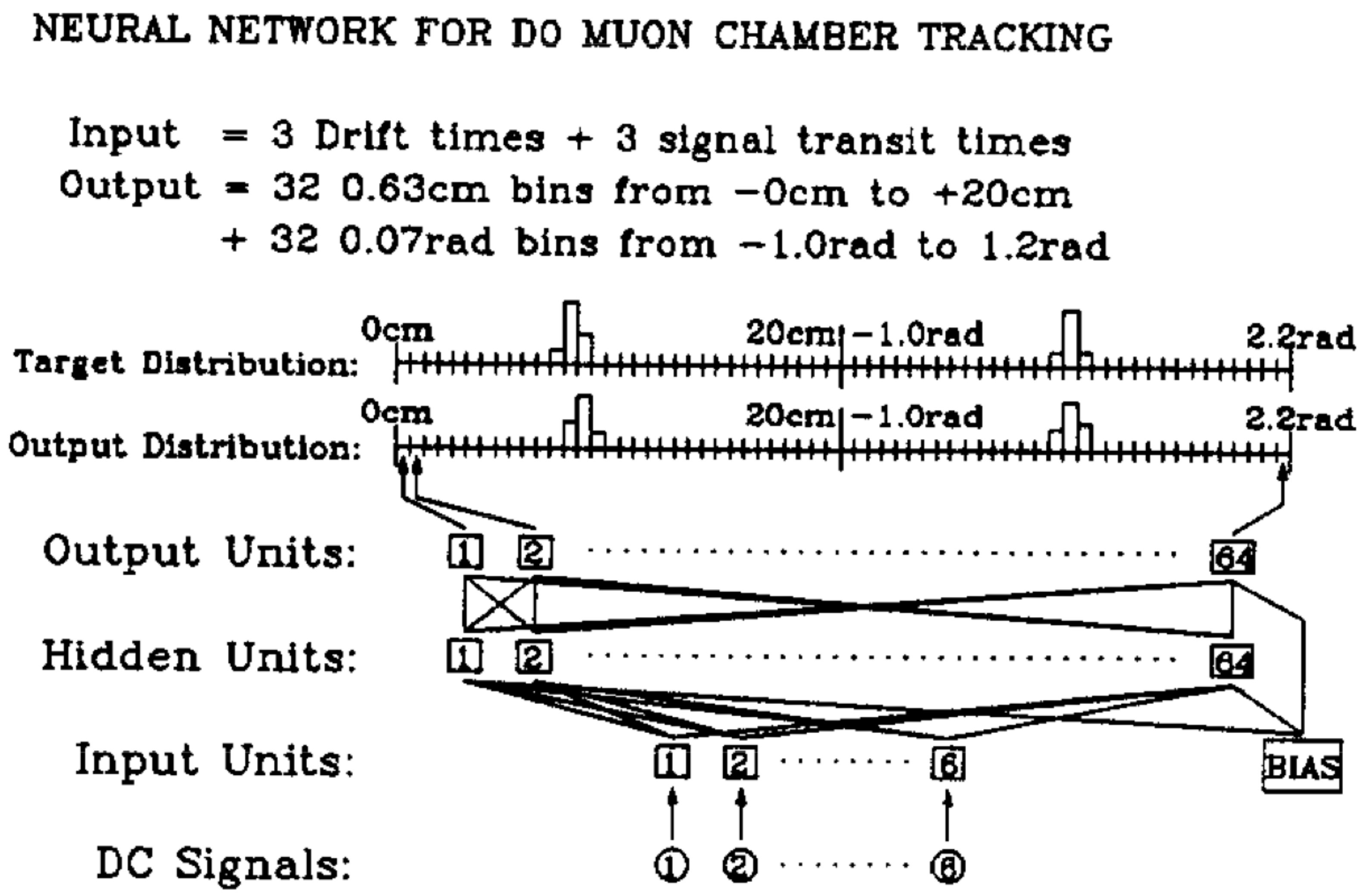


- We know for a long time more variables together could provide stronger distinction power, thus the term multivariate analysis

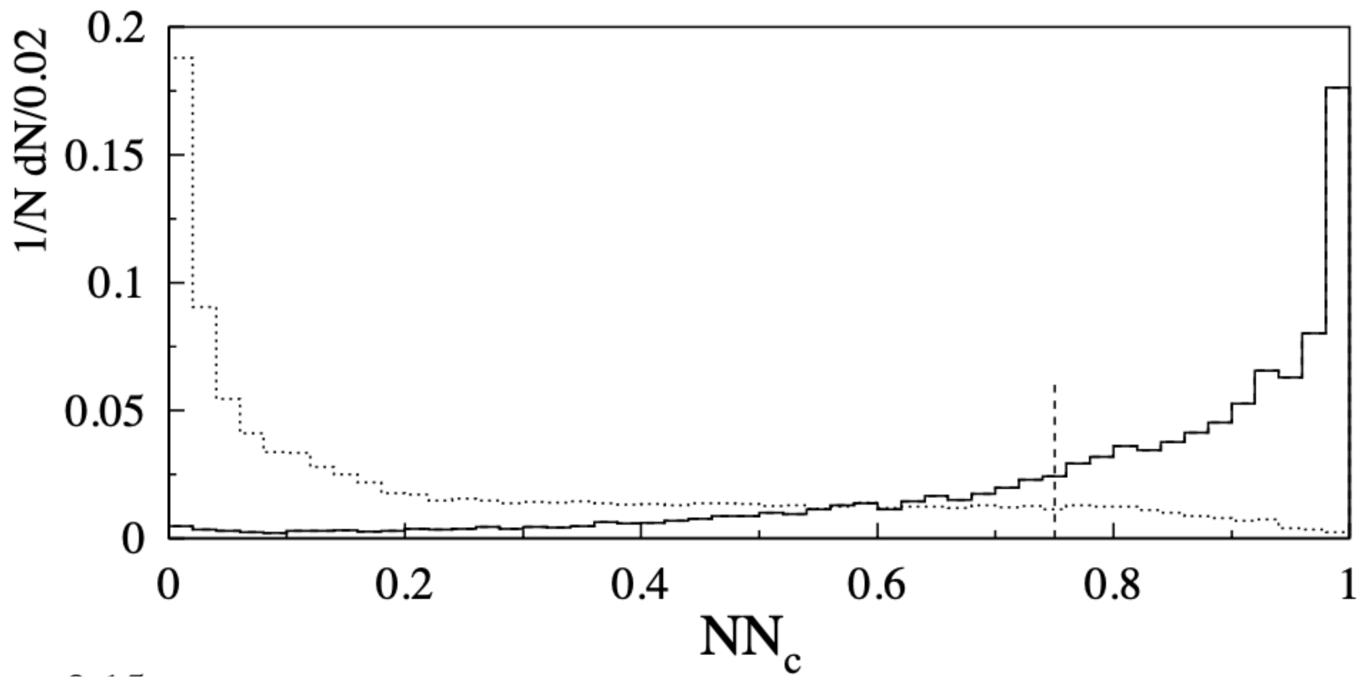
Examples of early ML applications



Connected network of adapted processors system (CNAPS) used for H1 trigger system, 2003: source



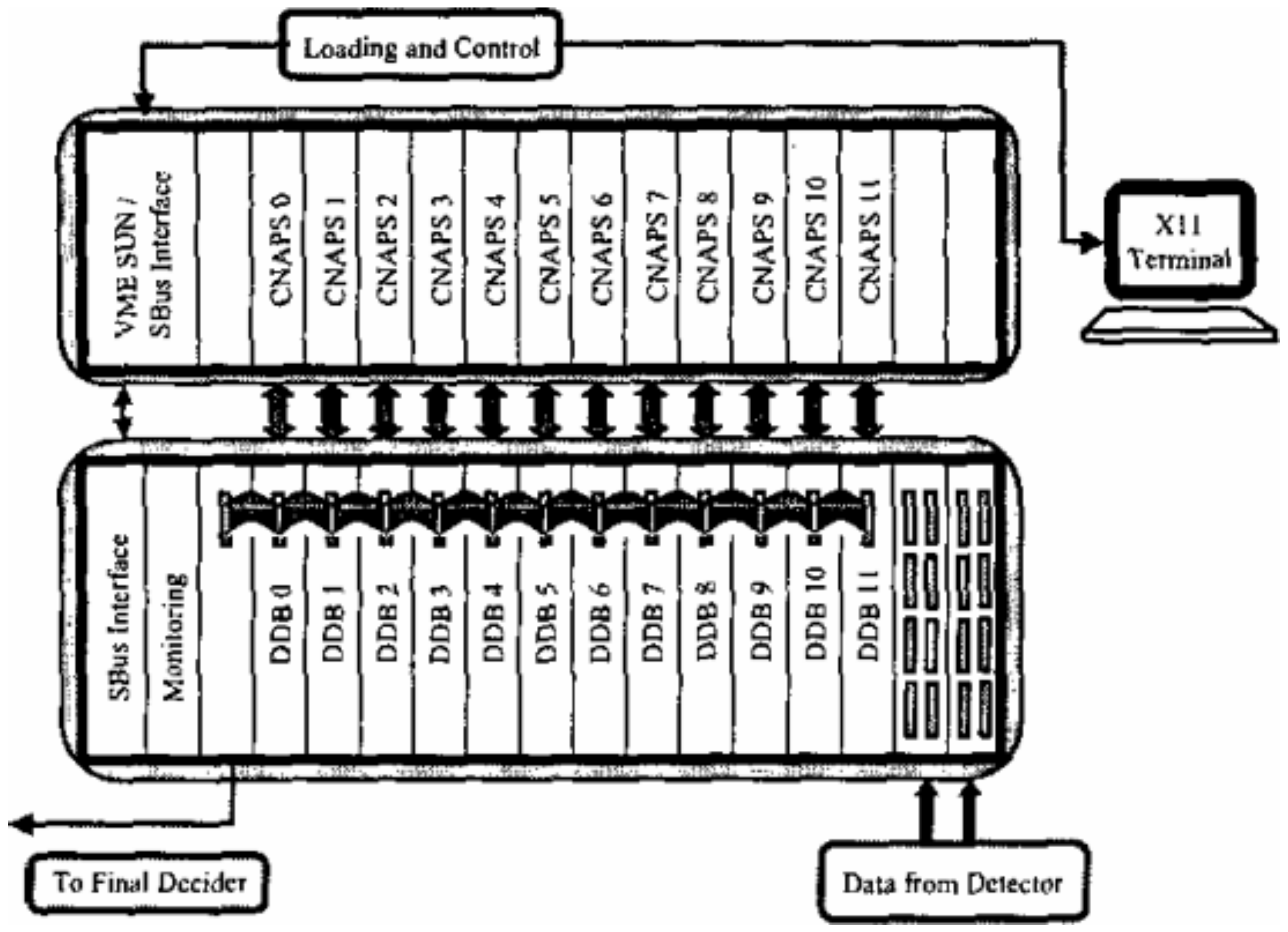
Primary vertexing based on the fired wires at E735, Fermilab, 1991: source



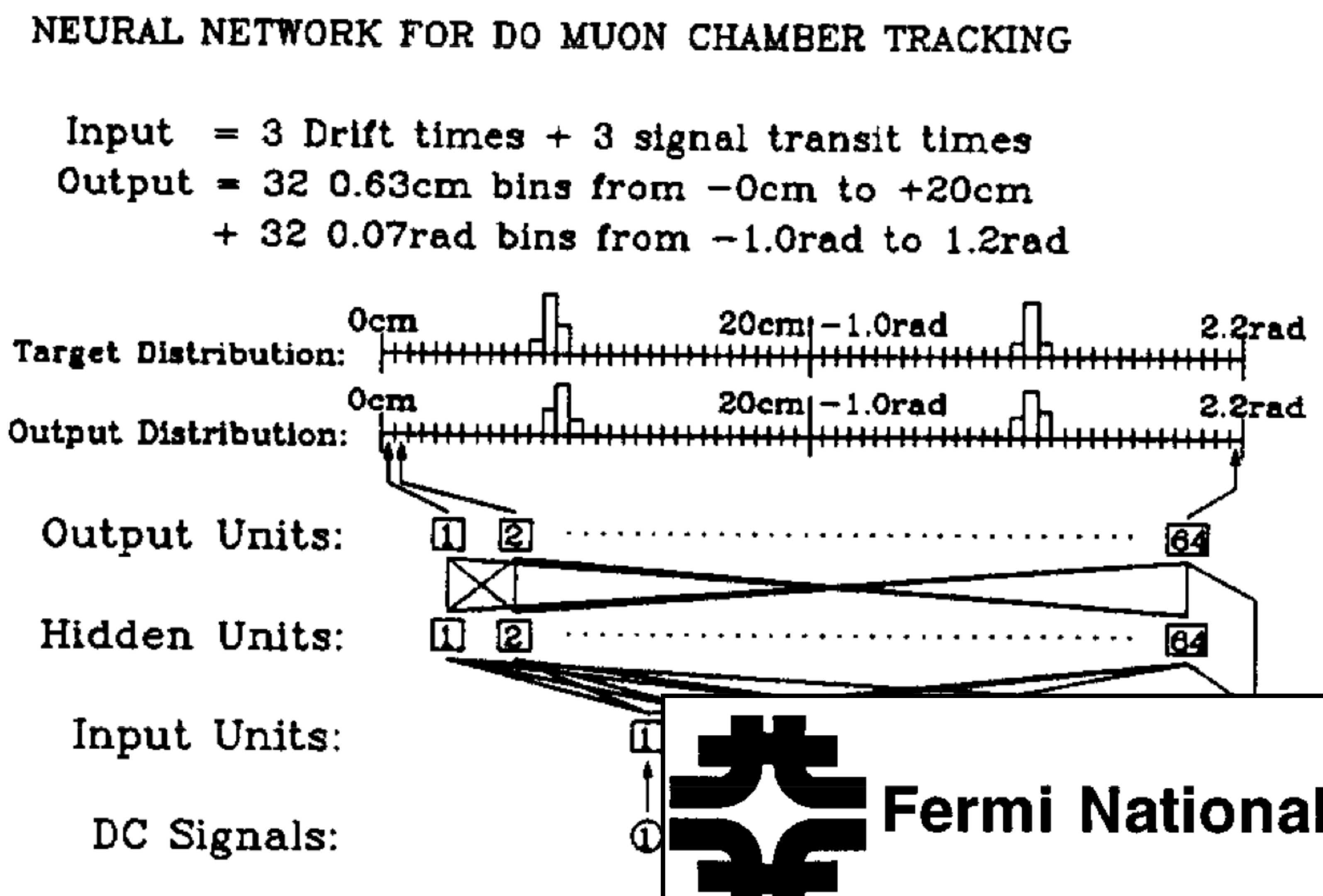
Selection of b hadrons at ALEPH, 1999: source

- ML has been used in HEP for a long time

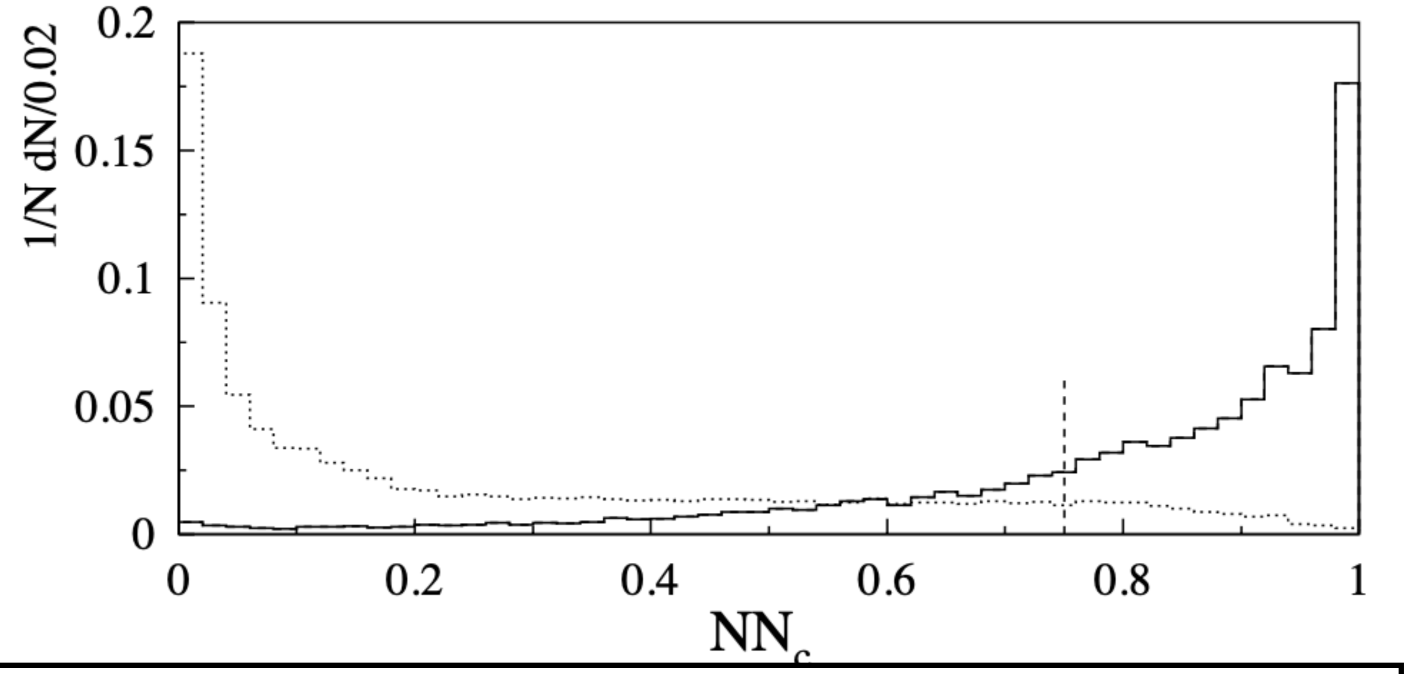
Examples of early ML applications

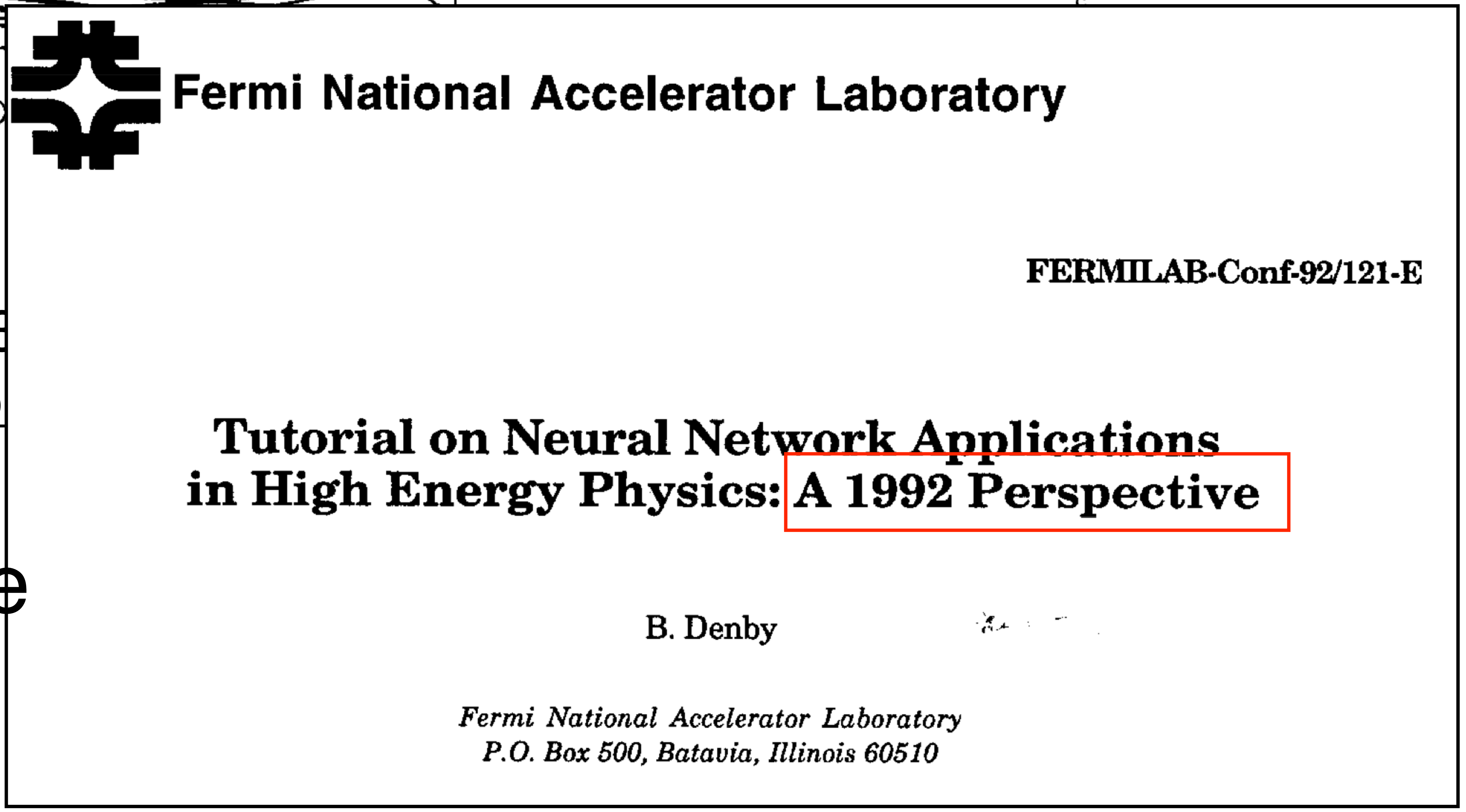


Connected network of adapted processors system (CNAPS) used for H1 trigger system, 2003: source



Primary vertexing
fired wires at E73
1991: so





Fermi National Accelerator Laboratory

FERMILAB-Conf-92/121-E

Tutorial on Neural Network Applications in High Energy Physics: A 1992 Perspective

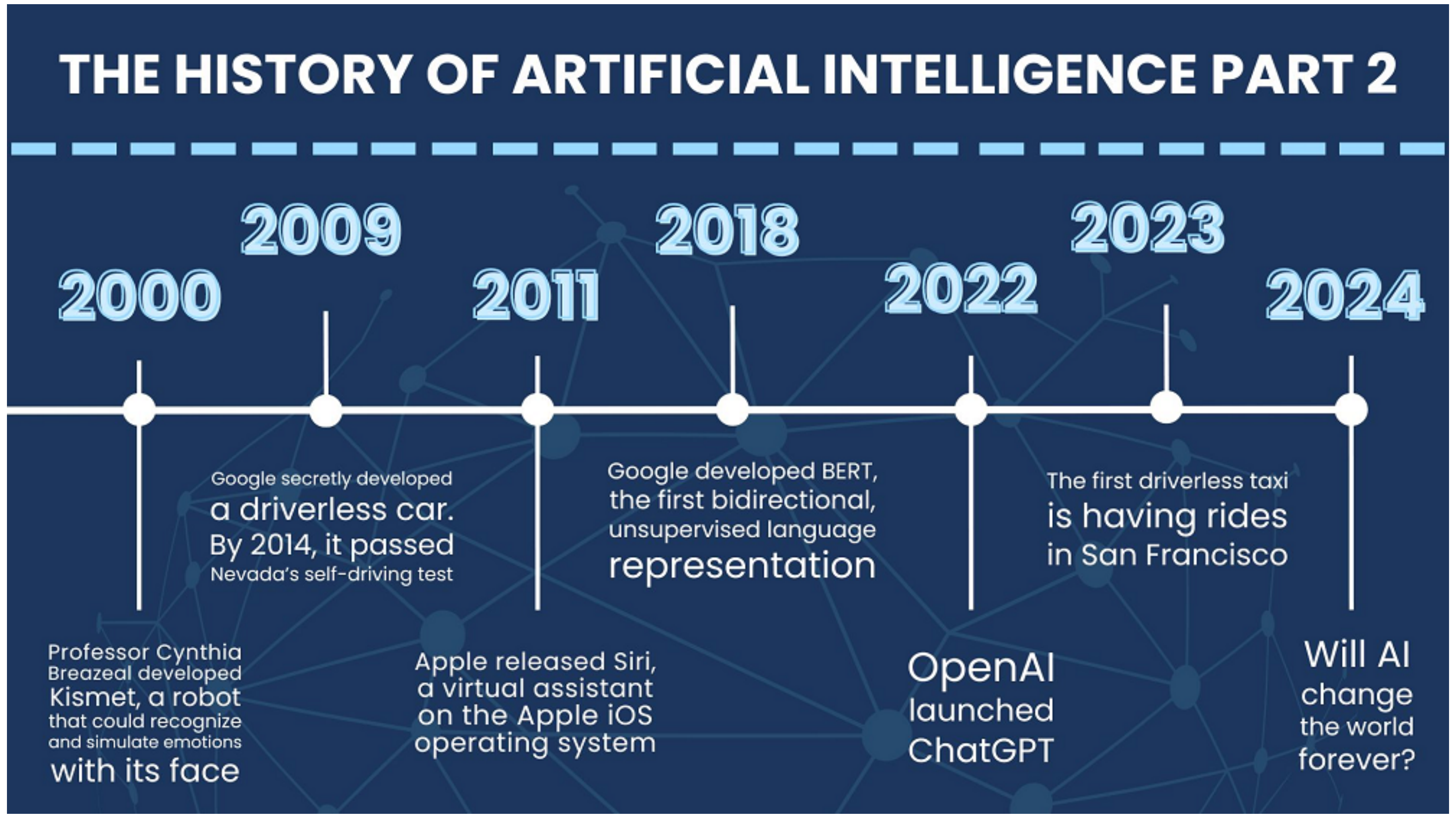
B. Denby

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

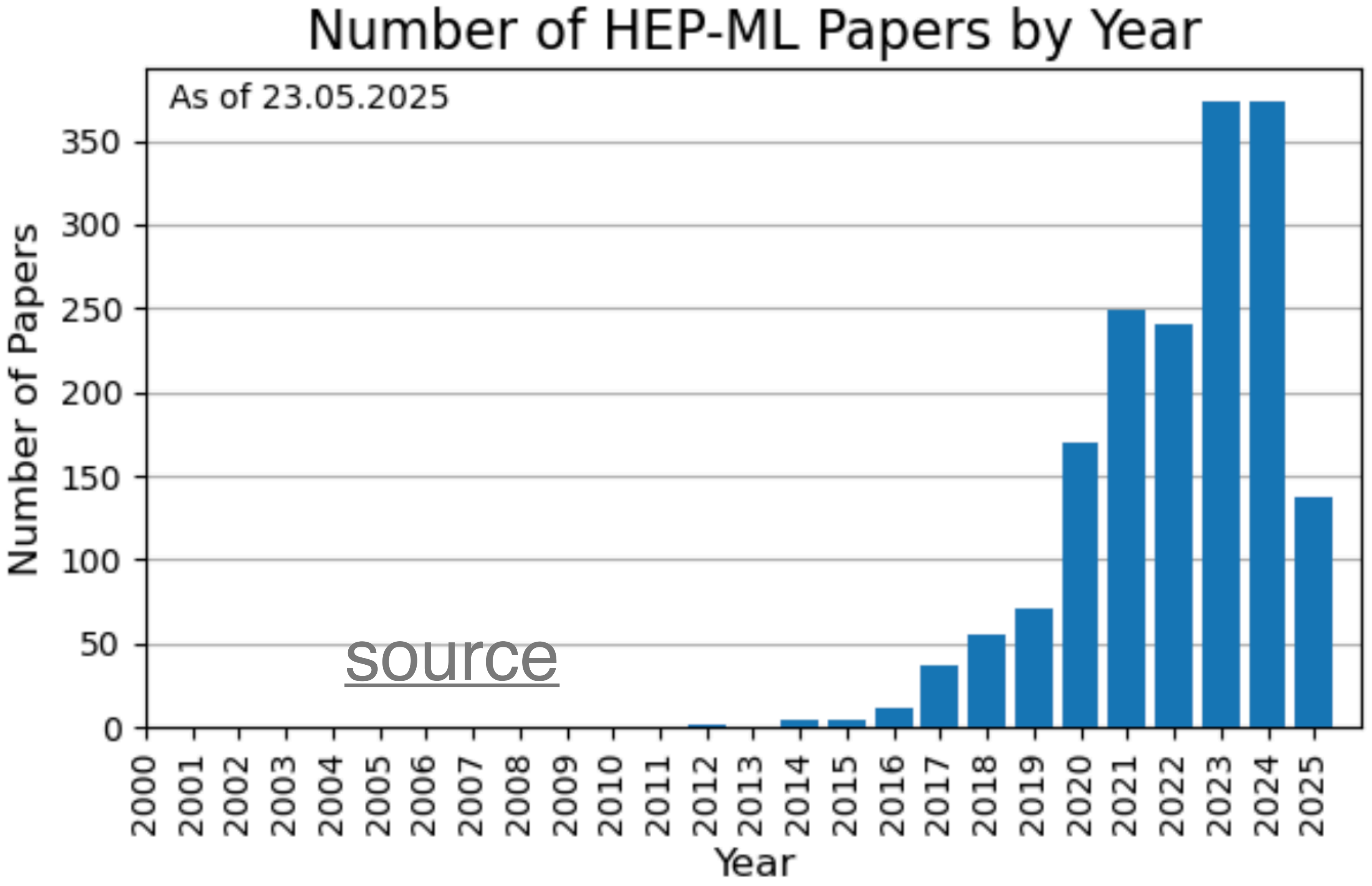
source

- ML has been used in HEP for a long time

Rapid development of ML technology

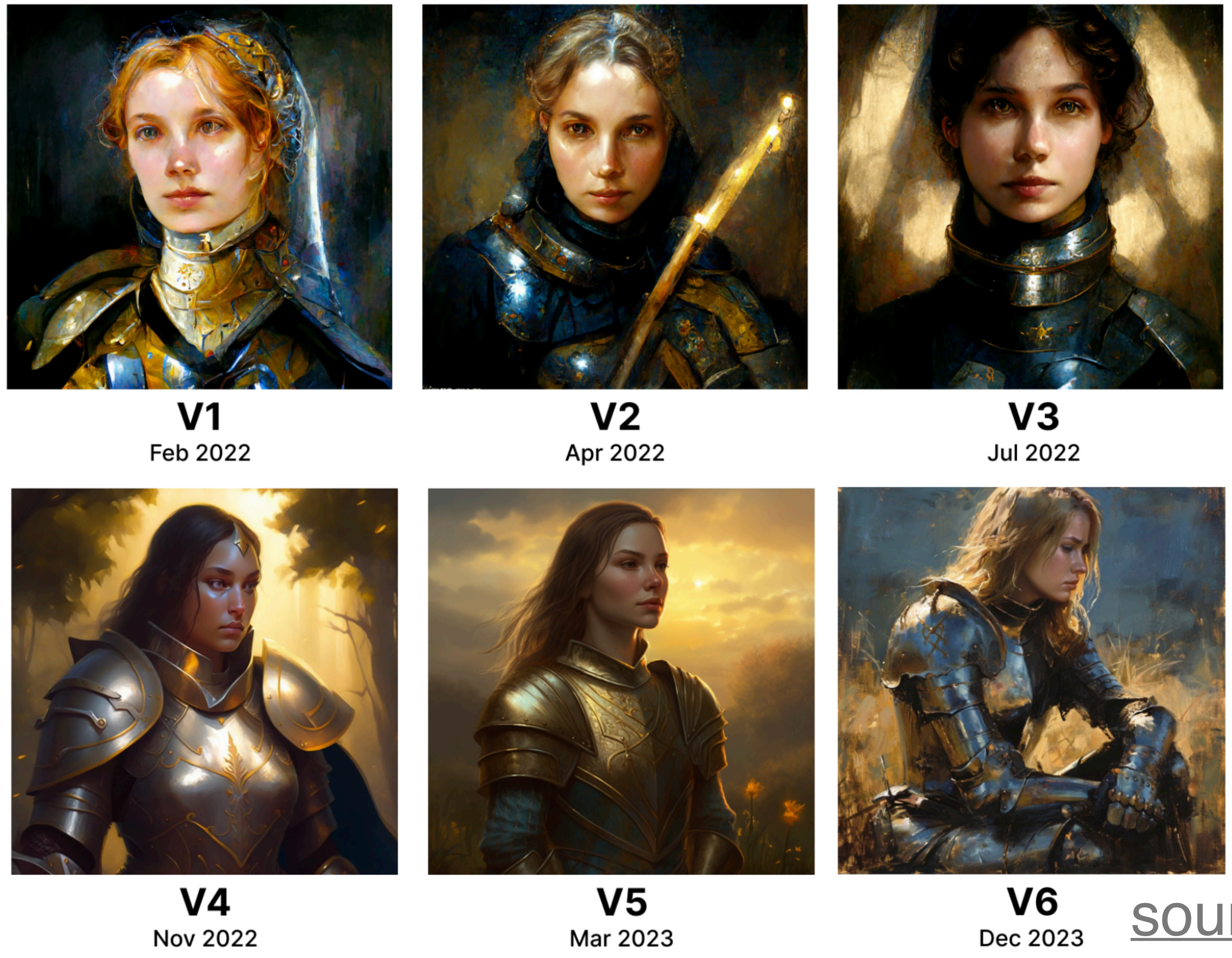


source



source

Images made by different MidJourney versions

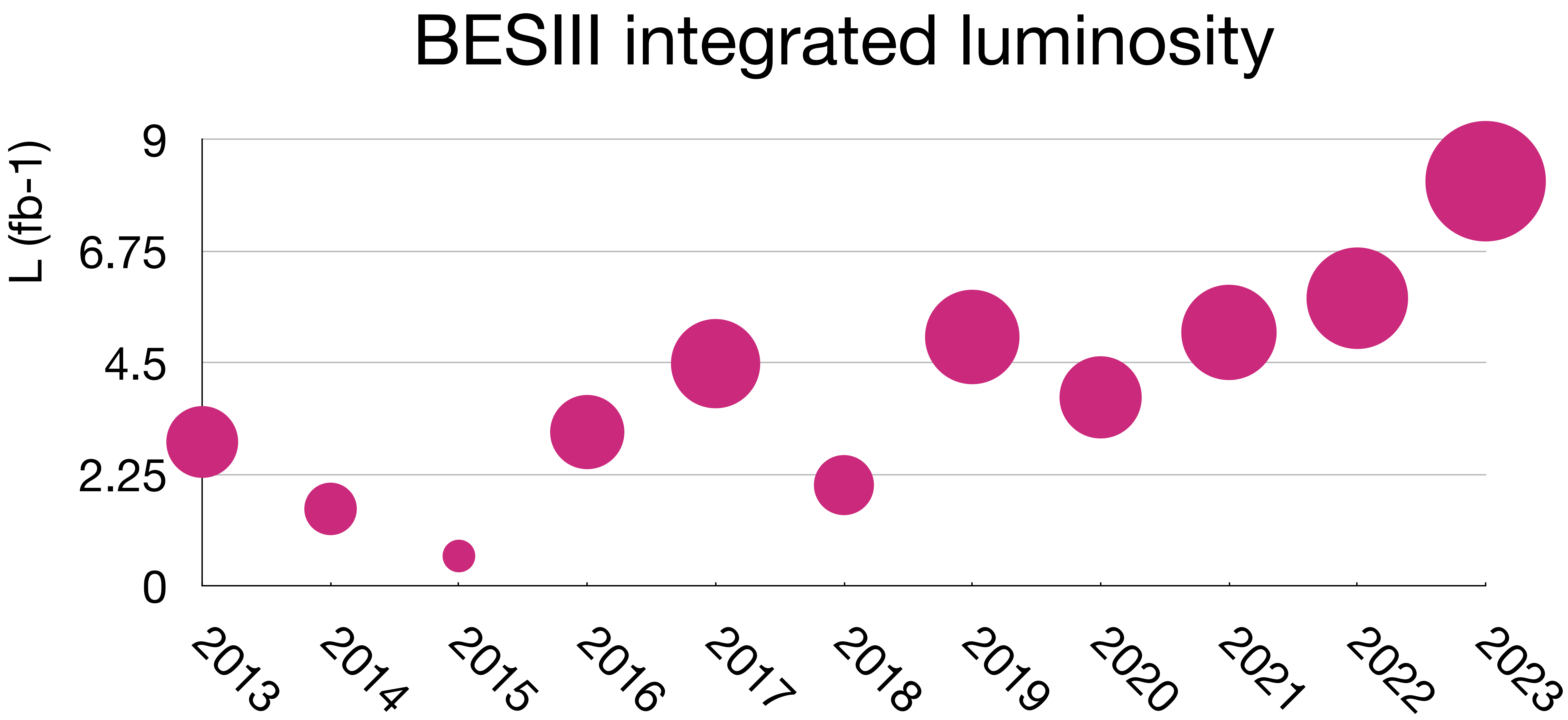


source

- Now that ML technology is used in daily life and everywhere thanks to the large dataset and powerful machine to train

Data volume

- HEP is known as data science



~1PB raw data

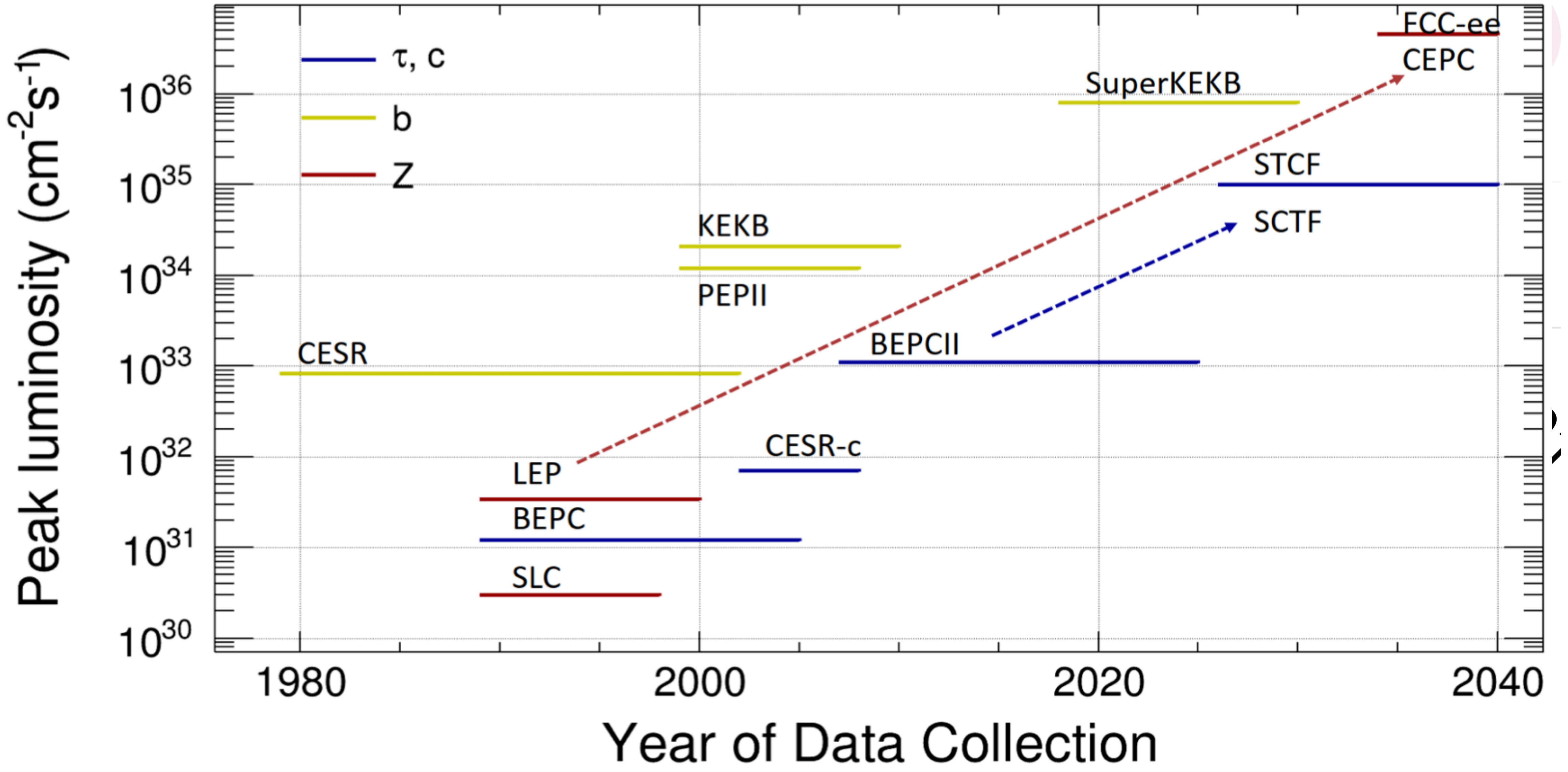
~1PB DST data

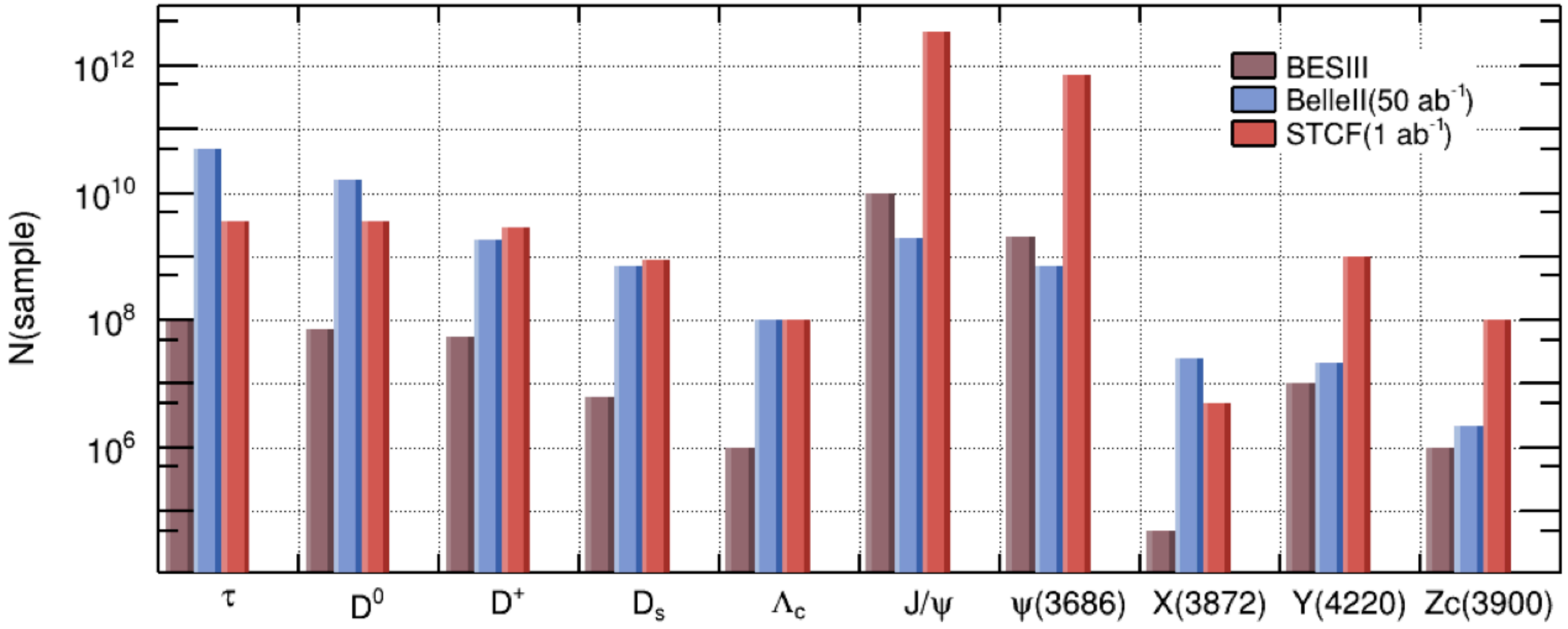
source

Data volume

BESIII integrated luminosity

STCF expected luminosity

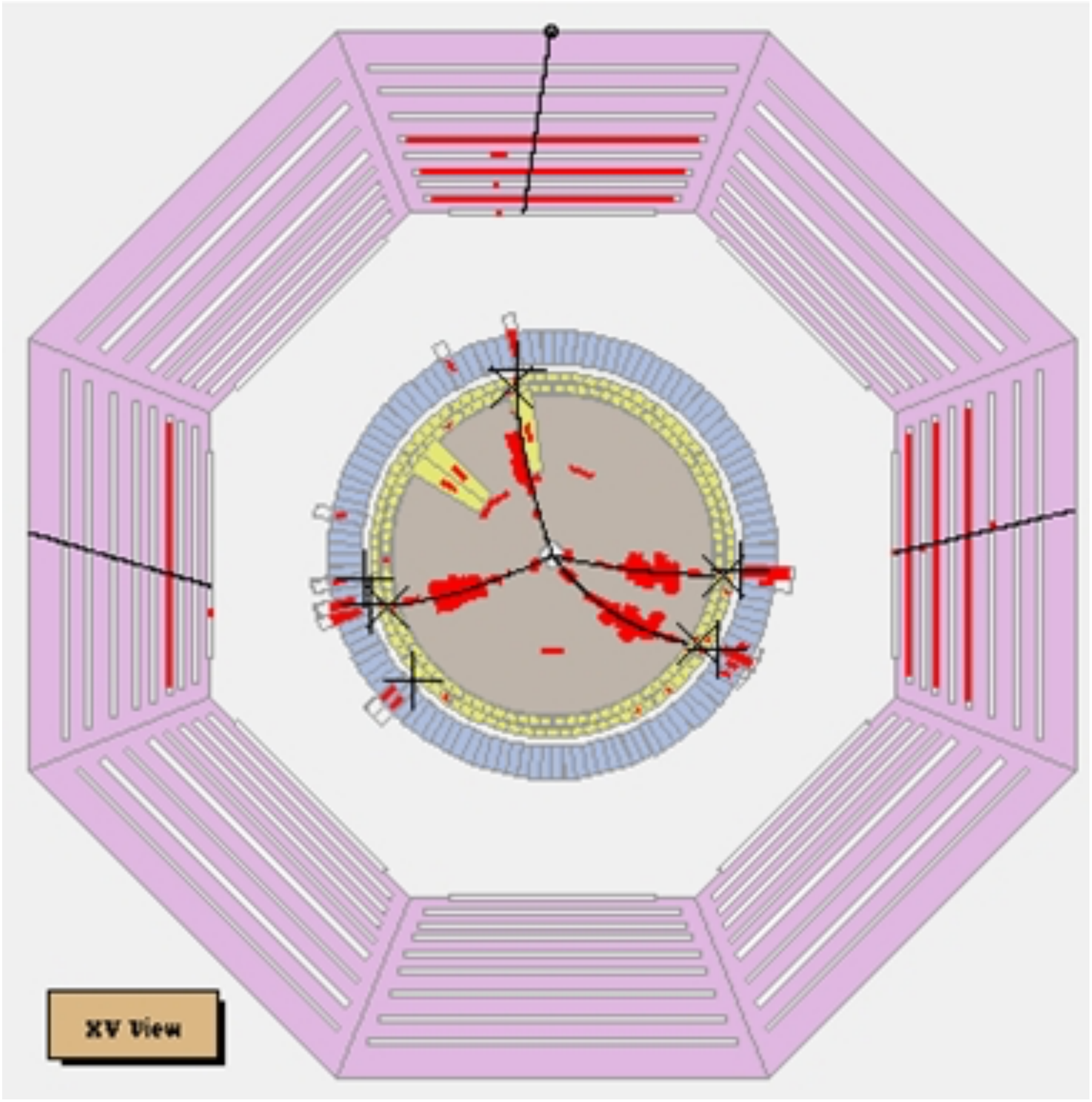




Contents

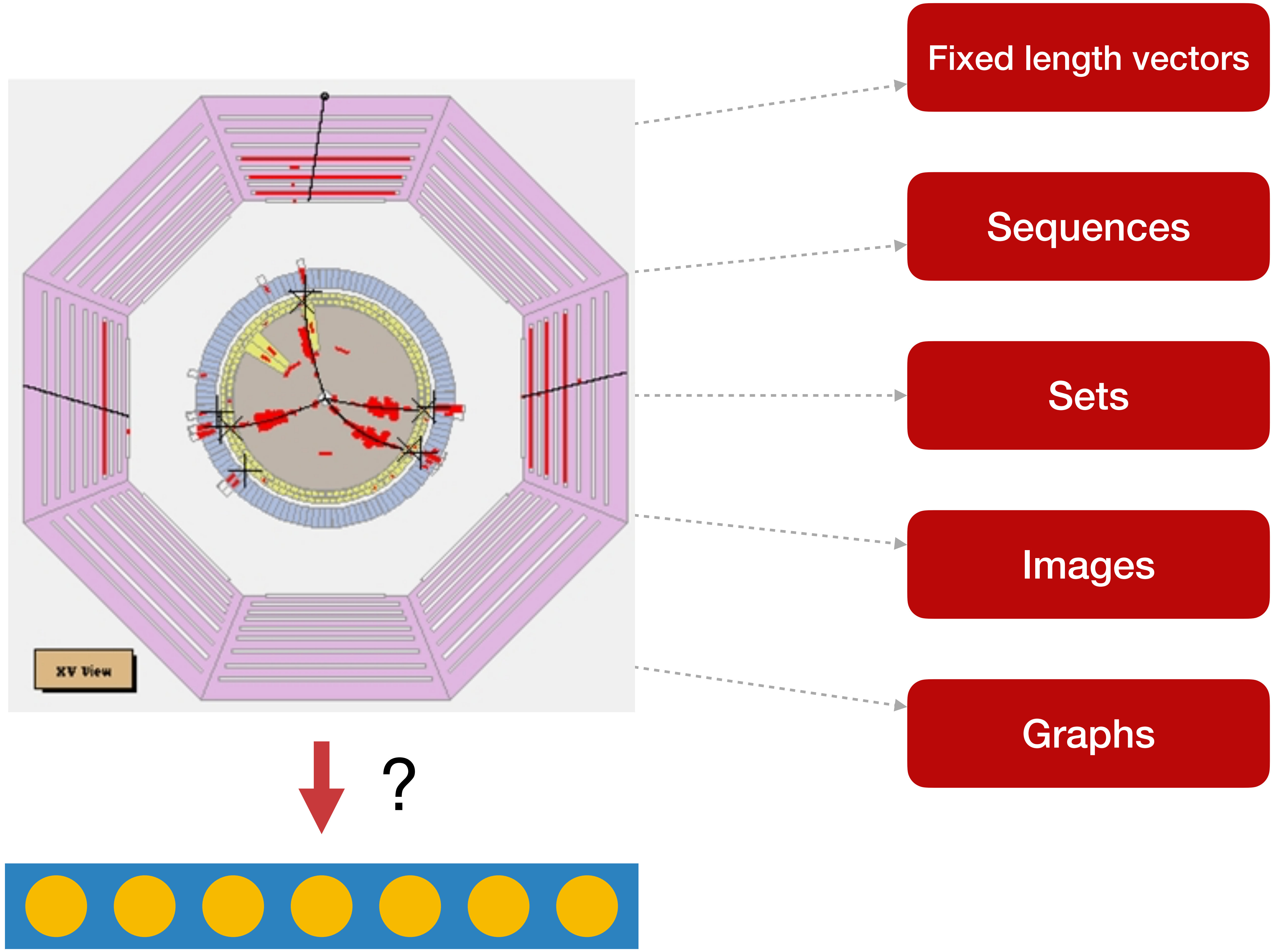
- Motivation to use machine learning (ML)
- How to map a HEP problem to a ML problem
- Examples of recent ML applications
- Summary

From Raw HEP Data to ML-Ready Formats

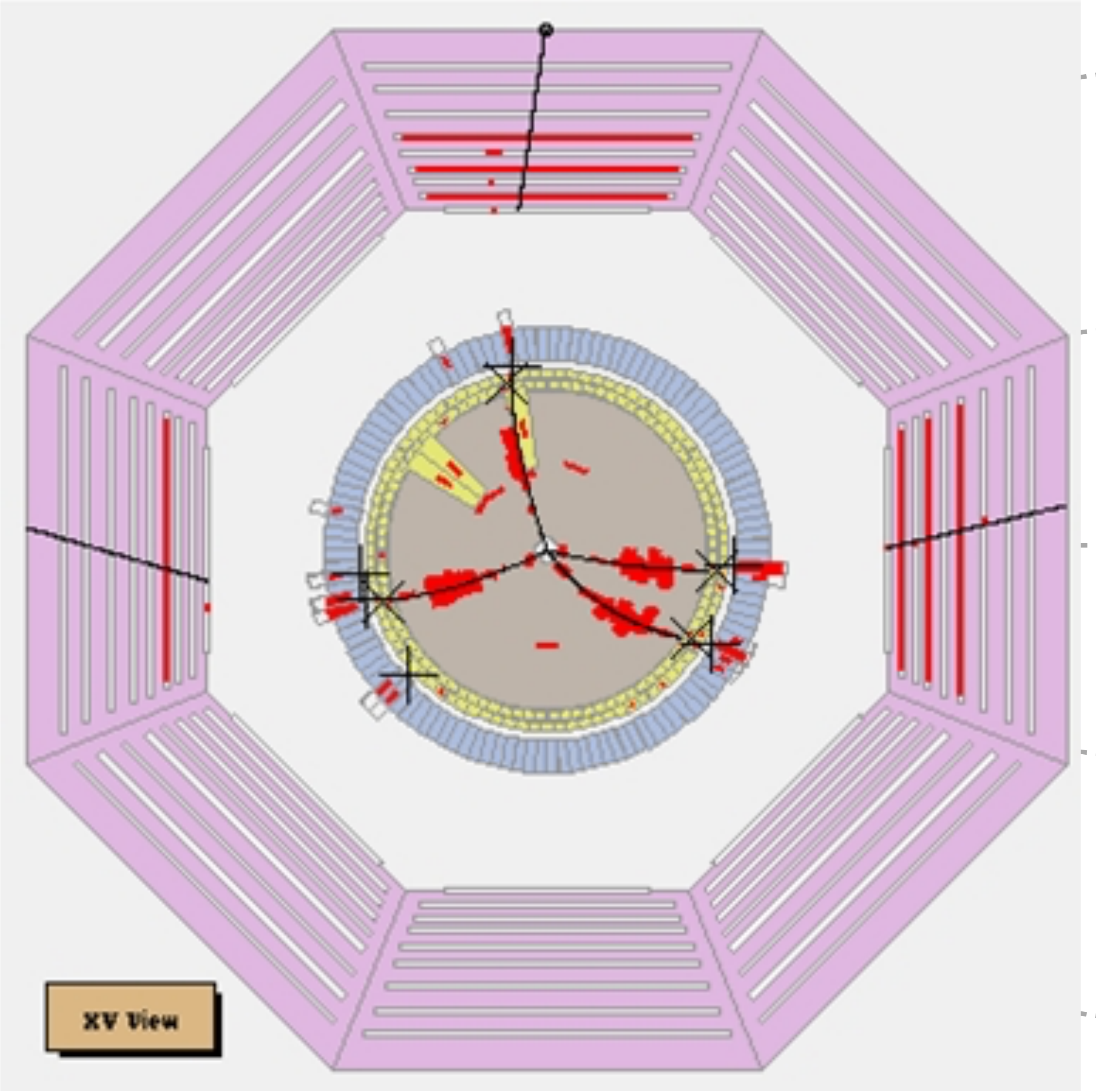


?

From Raw HEP Data to ML-Ready Formats



From Raw HEP Data to ML-Ready Formats



?

Fixed length vectors

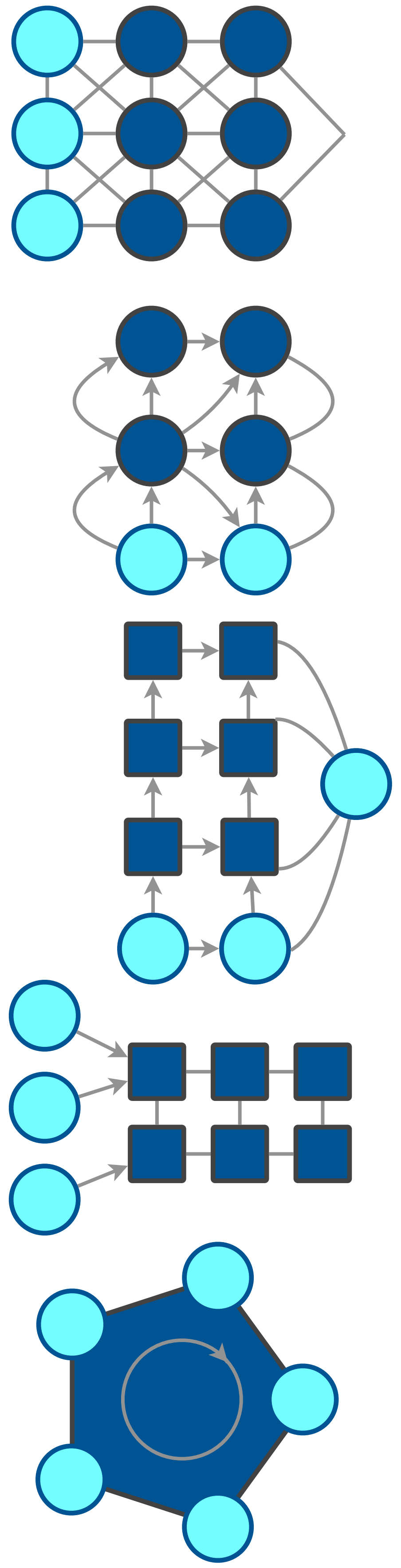
Sequences

Sets

Images

Graphs

...



DNN

RNN

Transformer

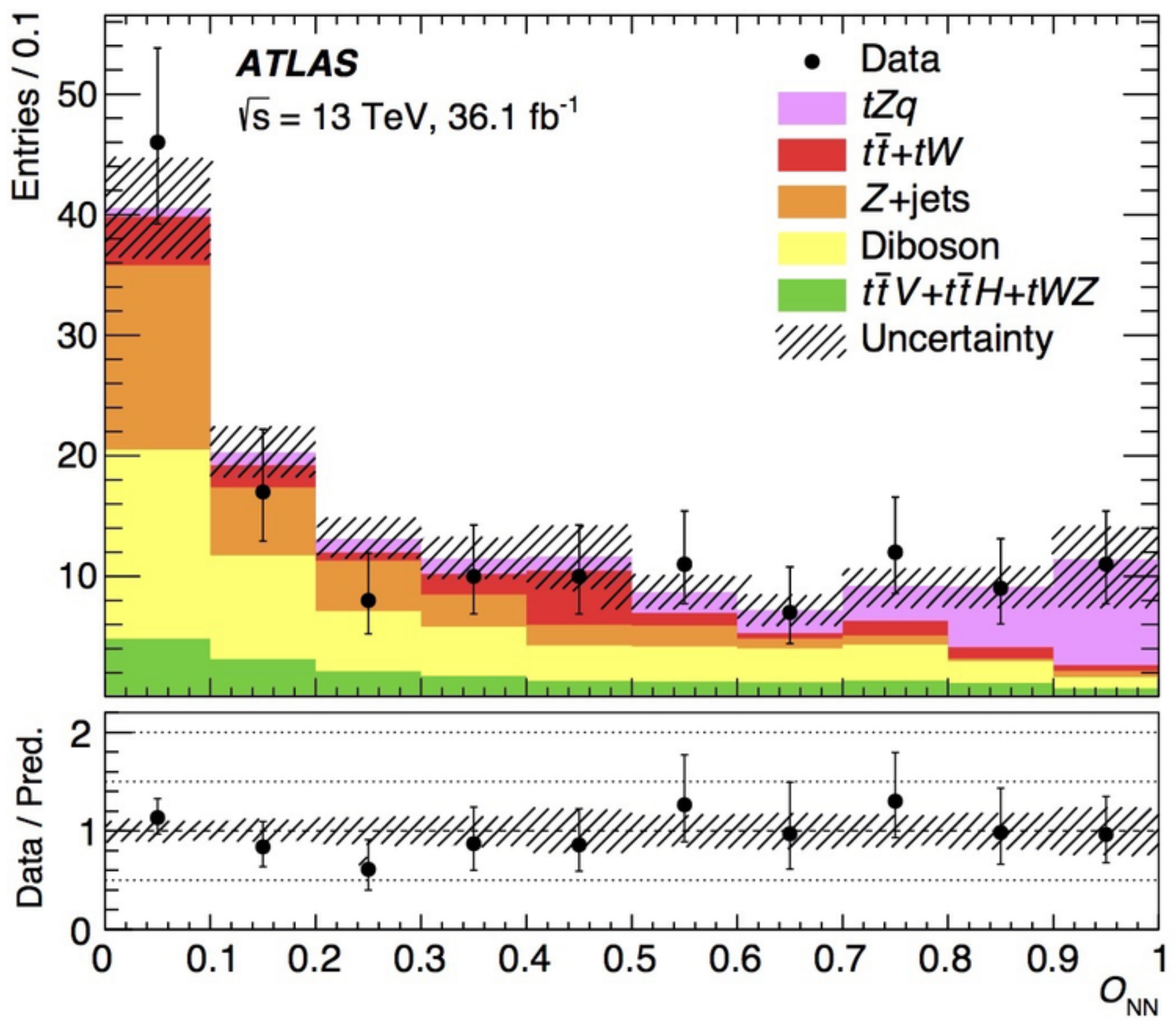
CNN

GNN

Fixed length vectors and DNN applications

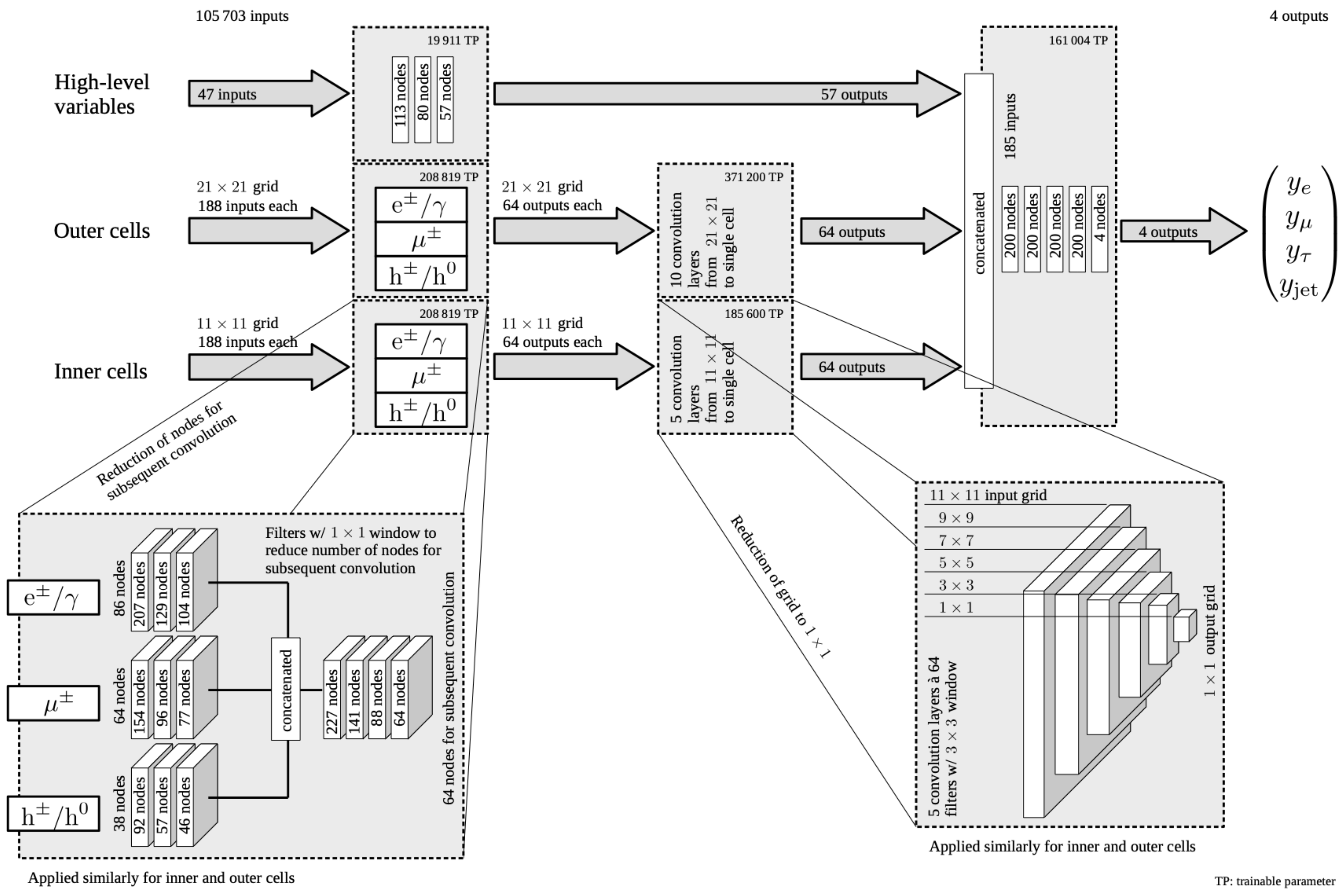
- Decide variable list for training in advance and train a deep neural network (DNN)

A typical signal extraction using NN



Phys. Lett. B 780 (2018) 557

CMS tau ID deep network

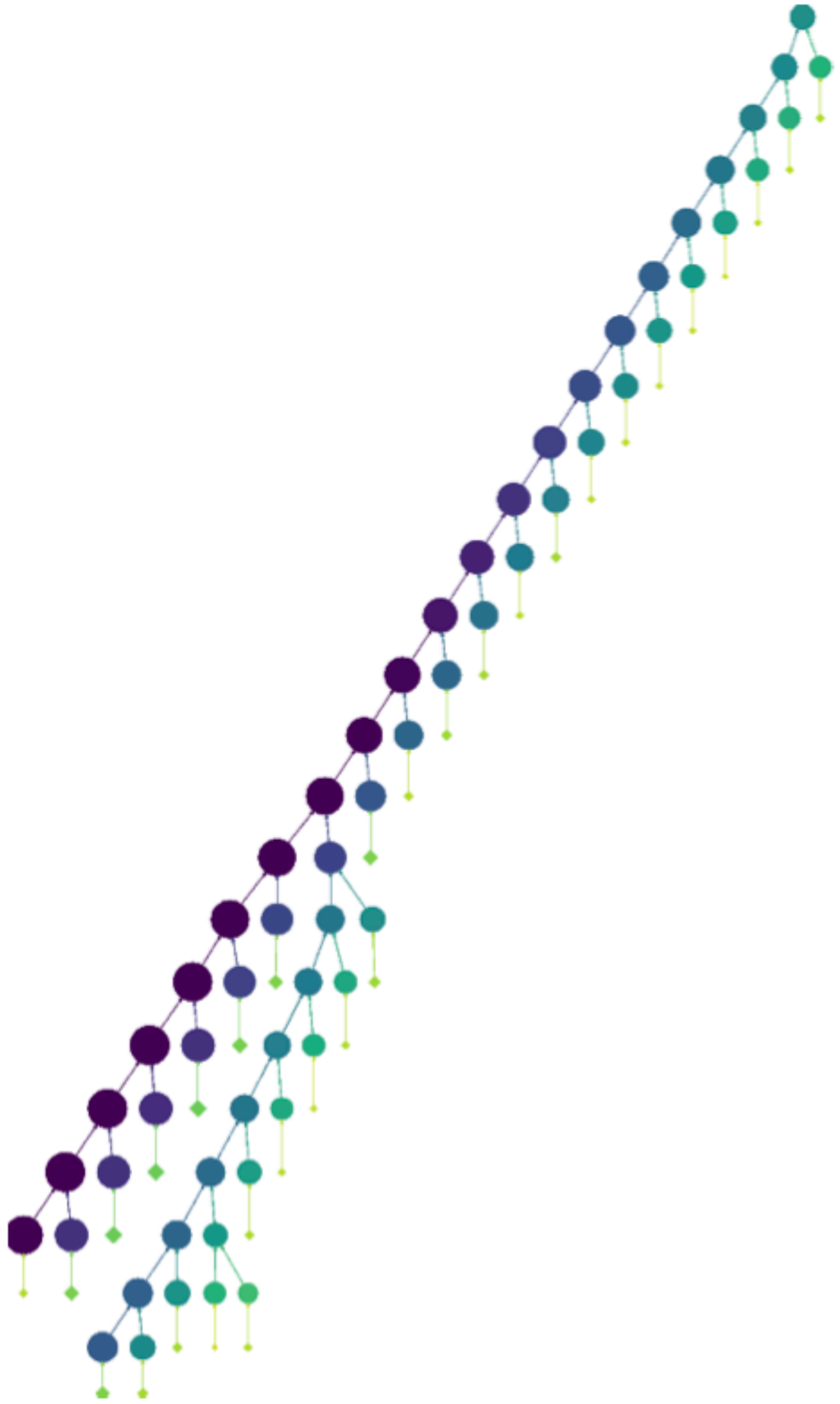


JINST 17 (2022) P07023

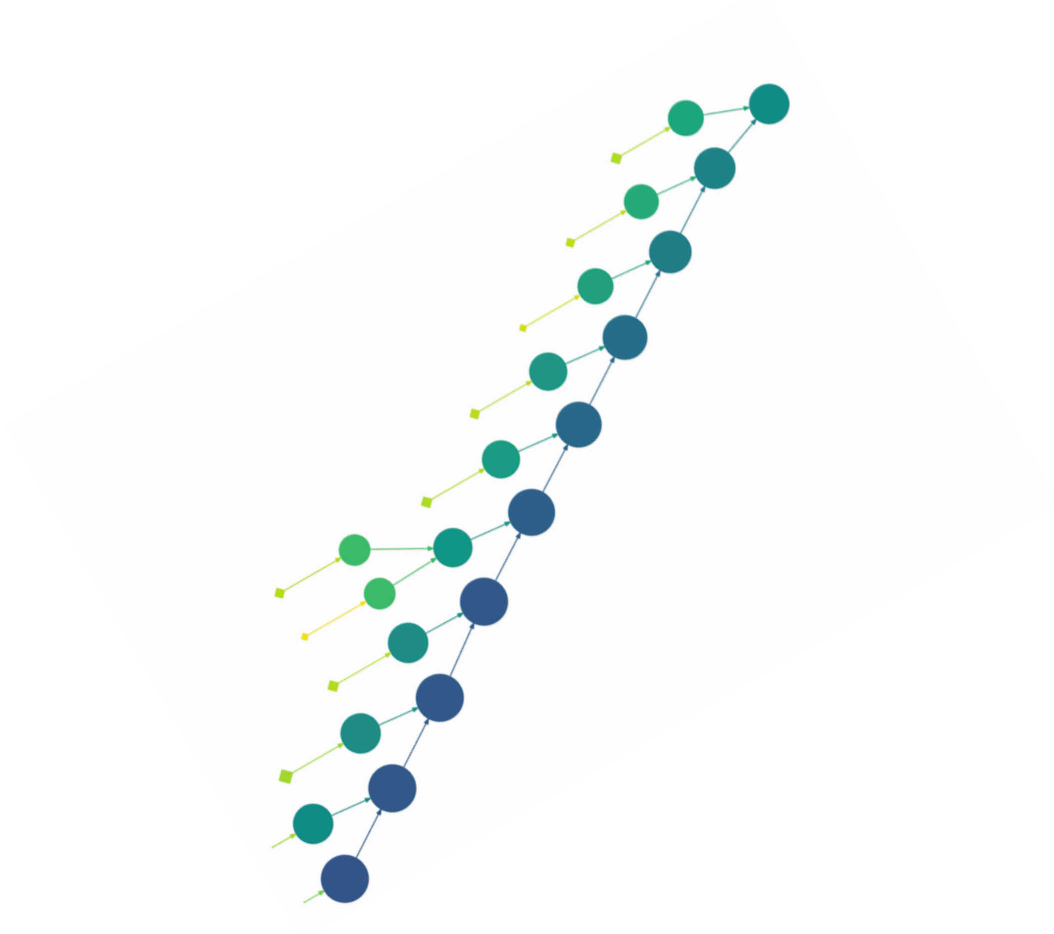
Sequences

- Sometimes fixed length vector is not applicable
  - e.g. Jets contain a variable number of particles
  - Recurrent Neural Networks shows great performance for Natural Language Processing tasks
  - Information across the entire sequence can be accumulated and used

A gluon jet

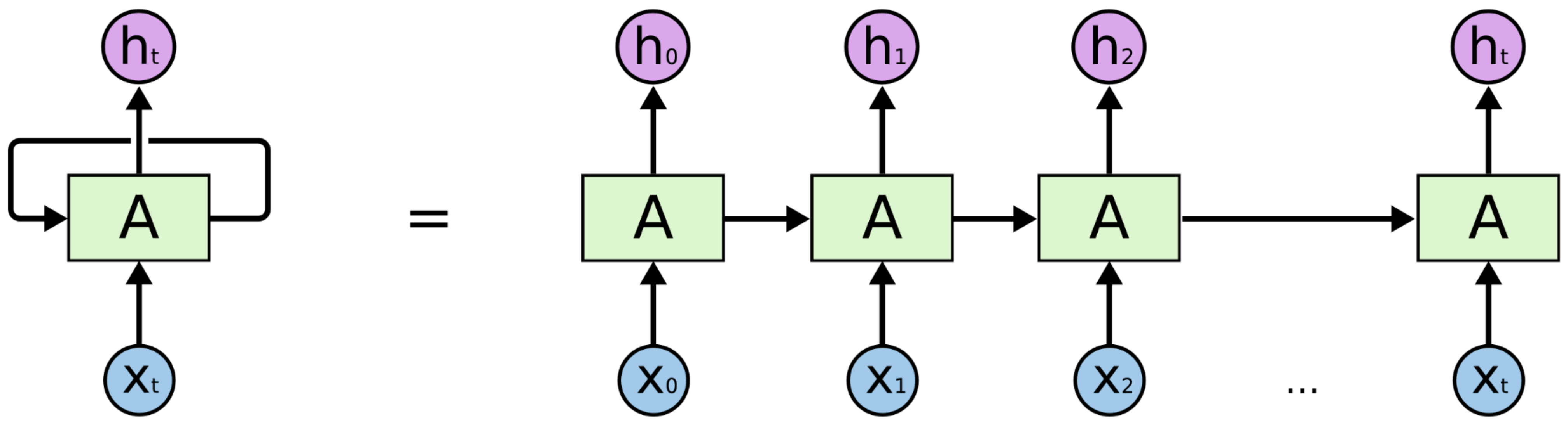


A quark jet



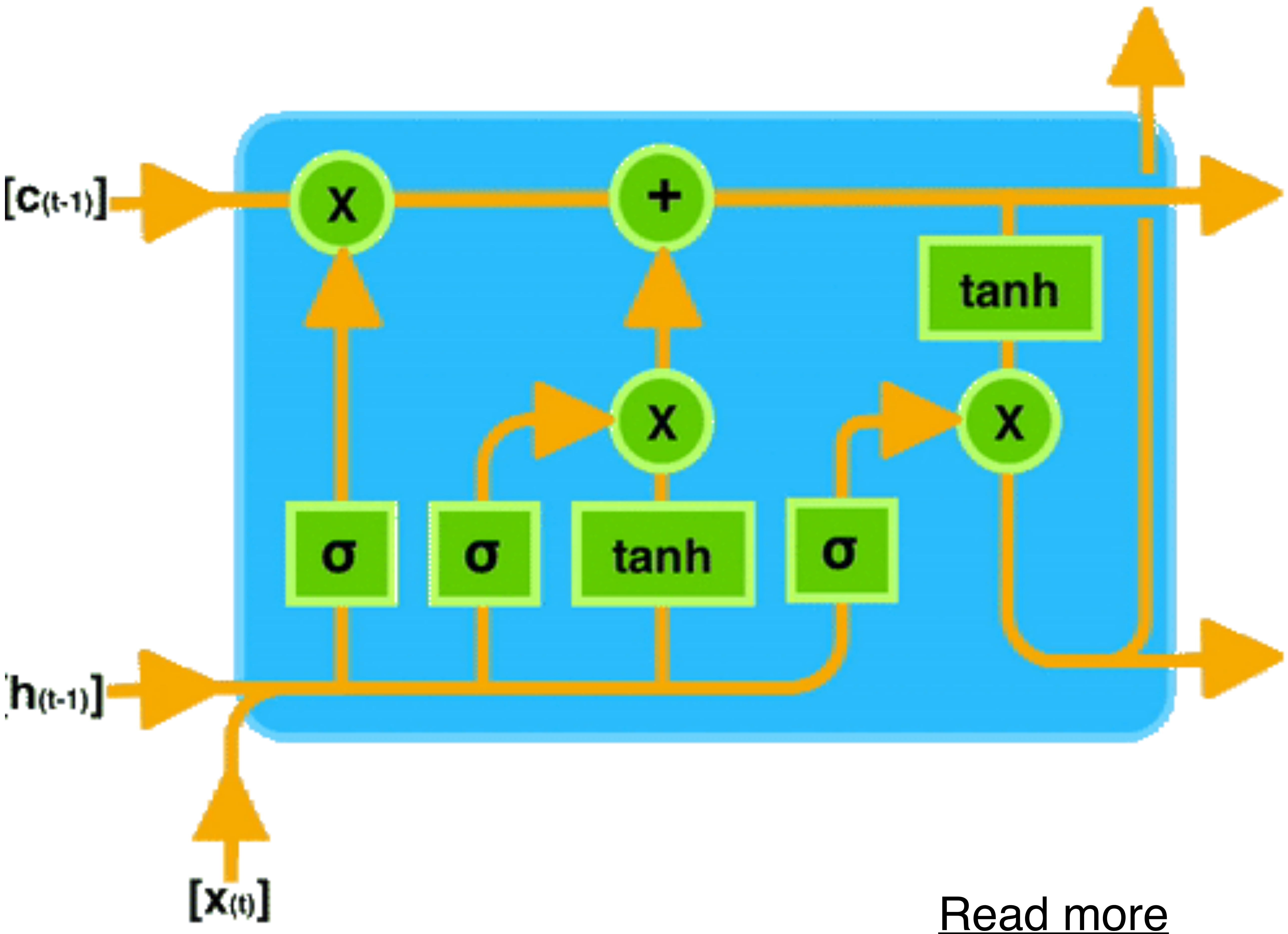
Recursive Neural Networks in Quark/Gluon Tagging

Recurrent Neural Networks (RNN)



An unrolled recurrent neural network.

Long Short-Term Memory (LSTM) network



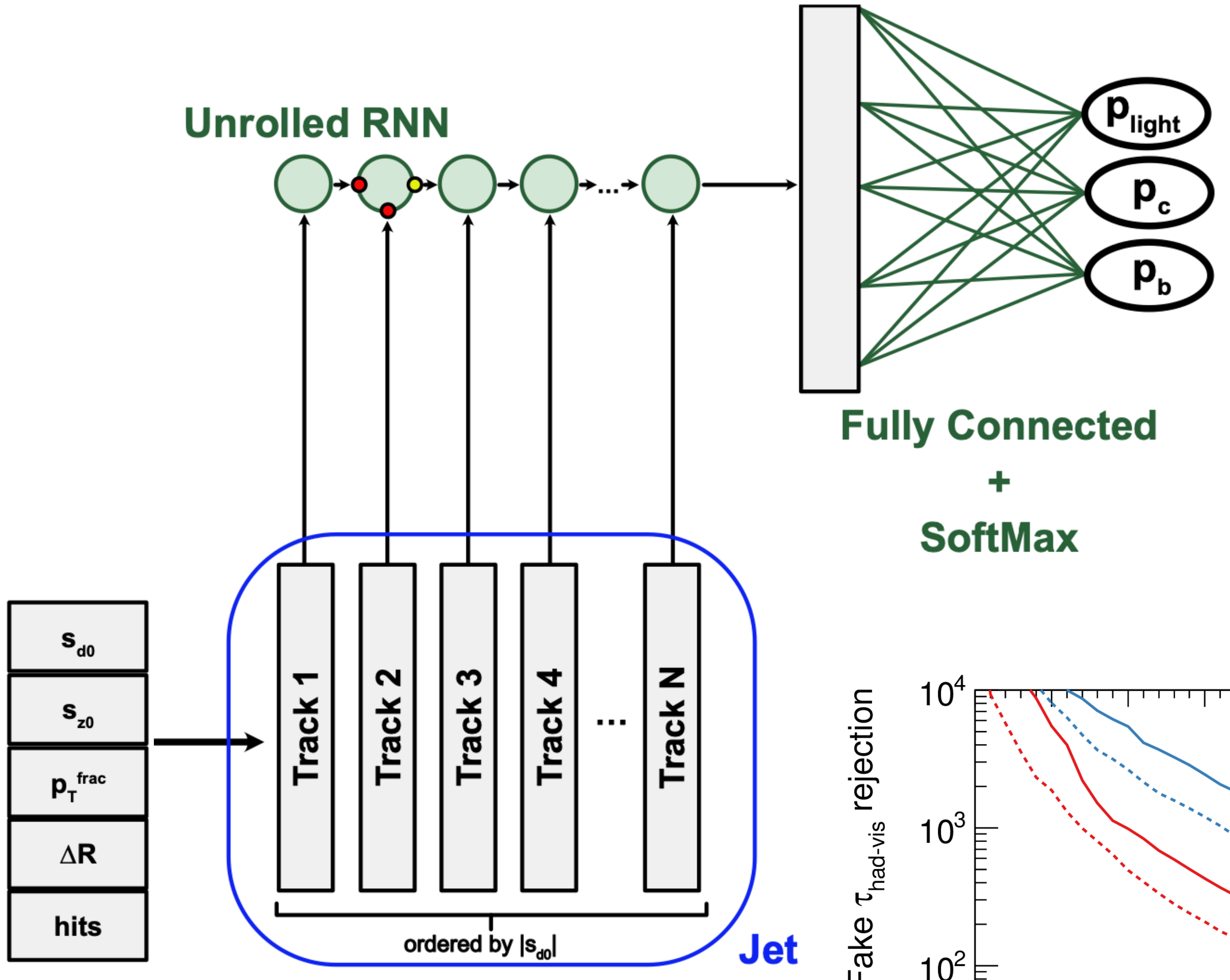
Success in applying RNNs to a variety of problems:

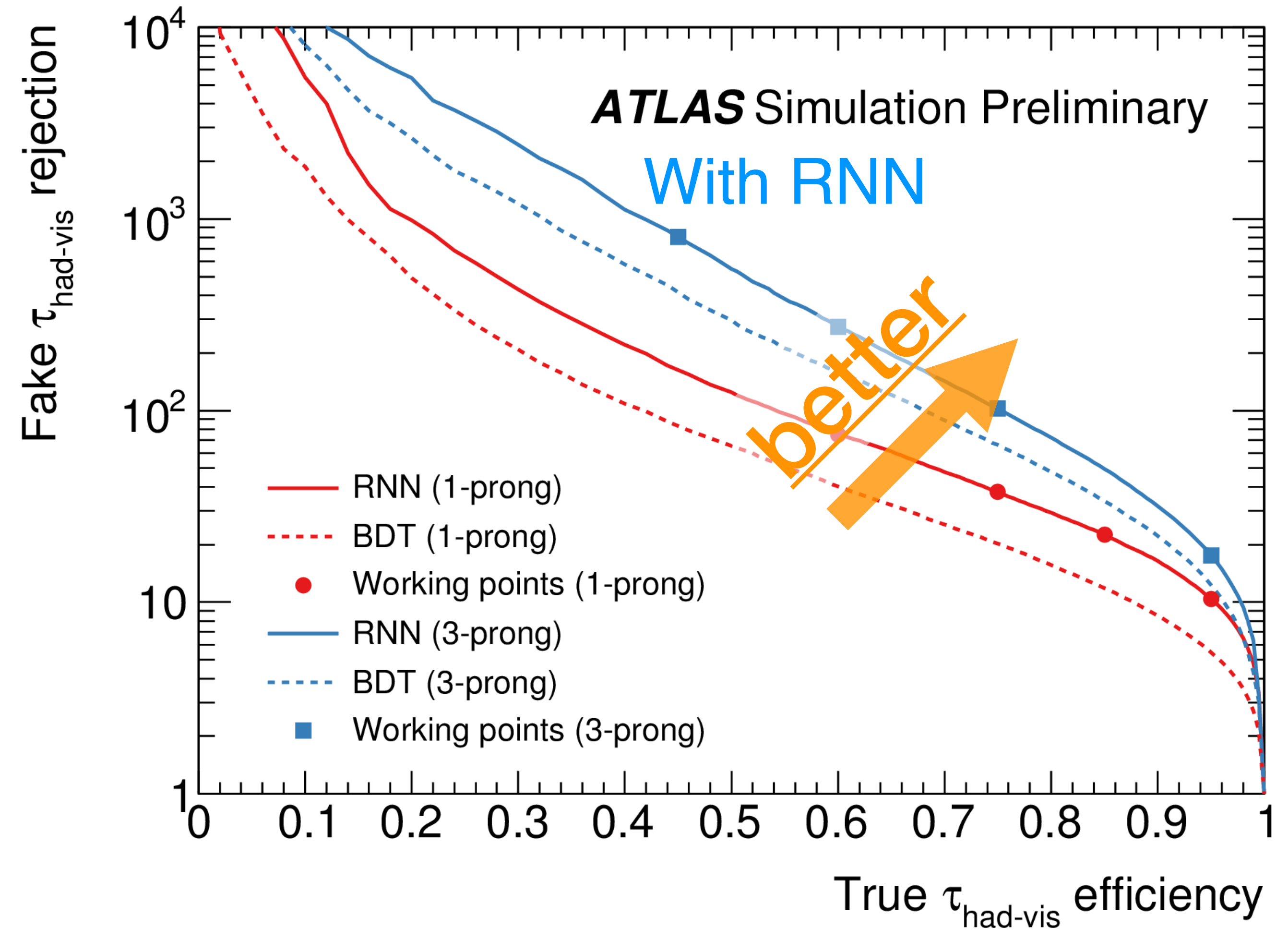
- Speech recognition
- Language modeling
- Translation
- Image captioning

Read more

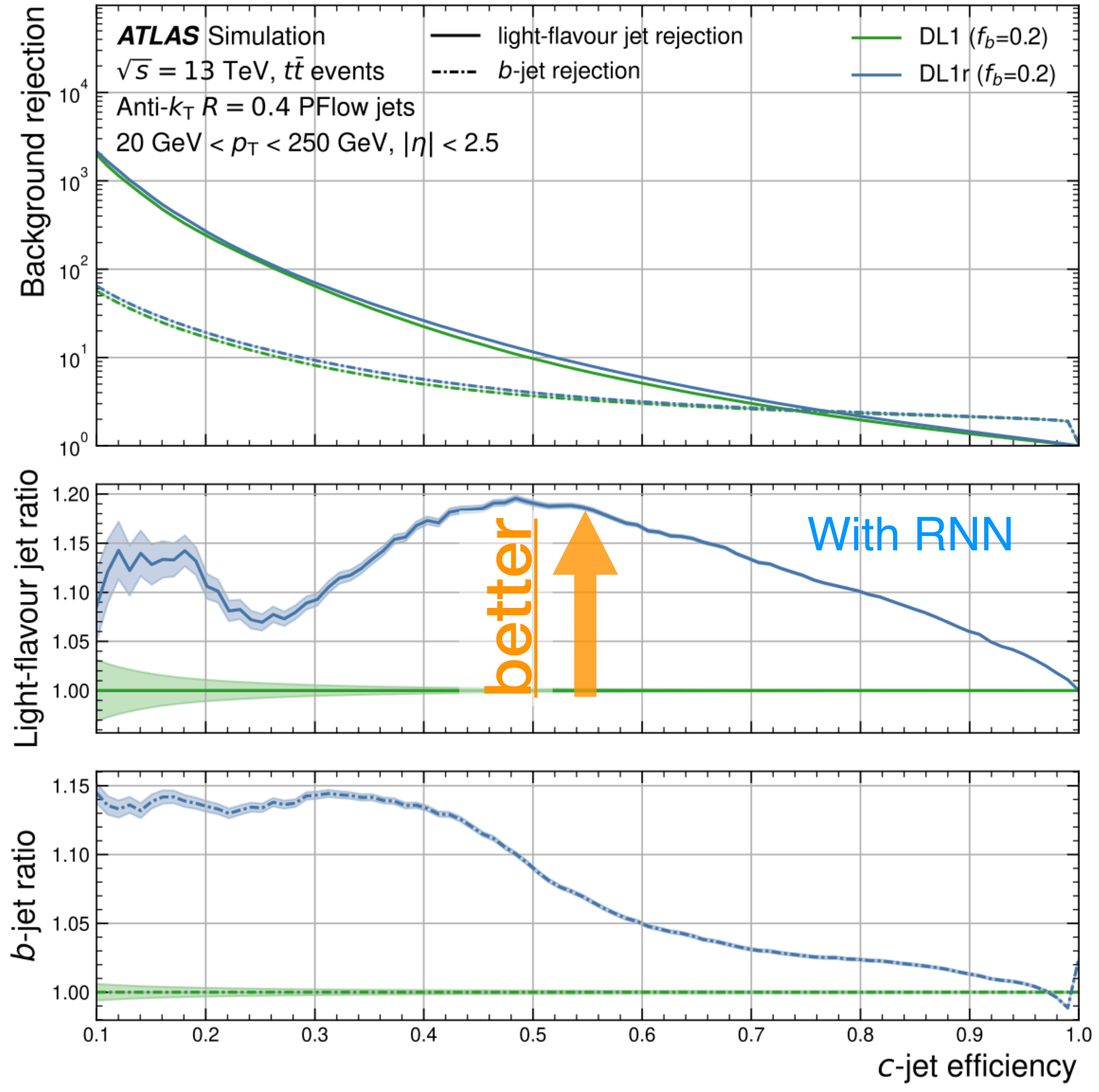
RNN applications

Tau identification





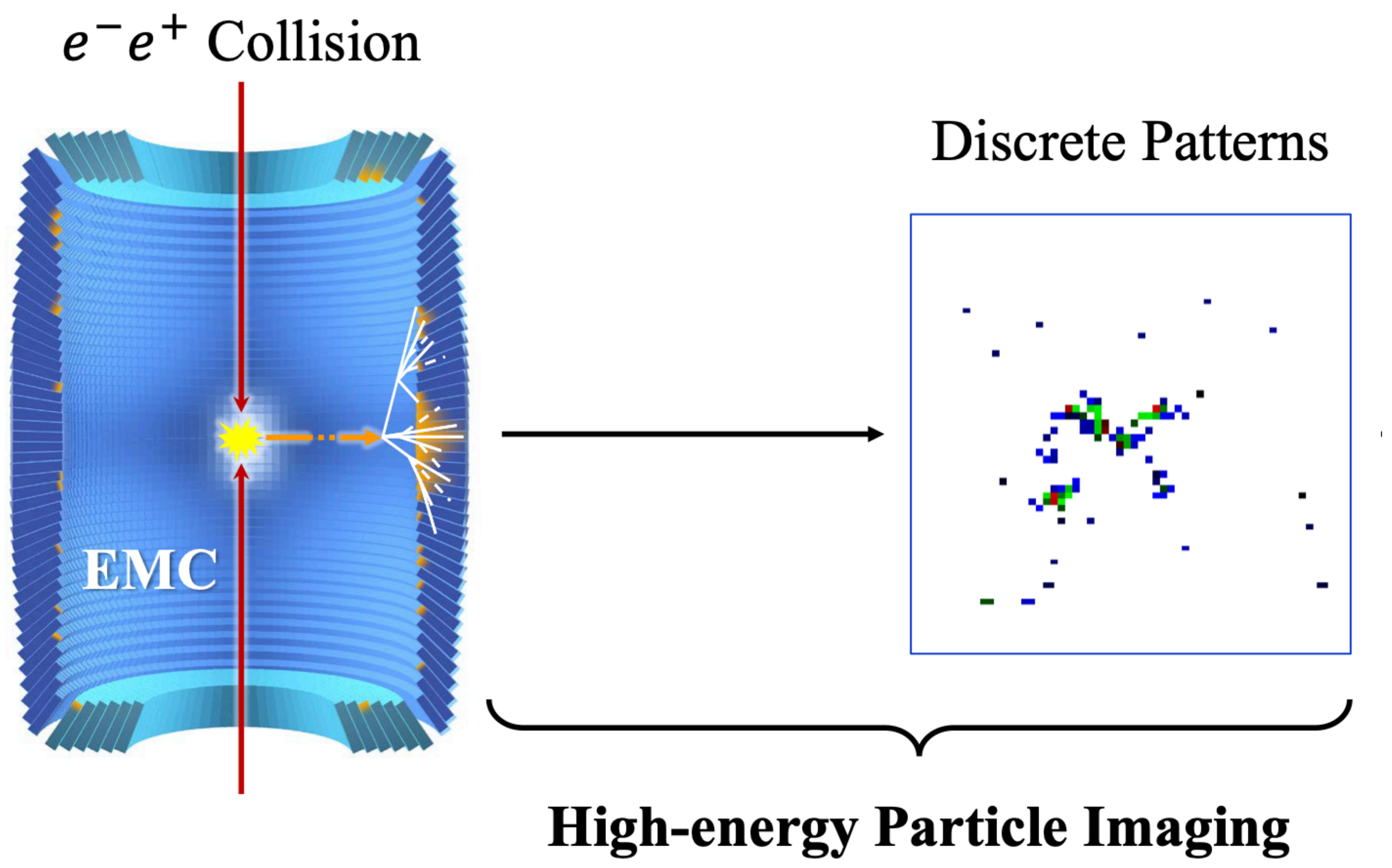
b-tagging



ATL-PHYS-PUB-2019-03

Eur. Phys. J. C 83 (2023) 681

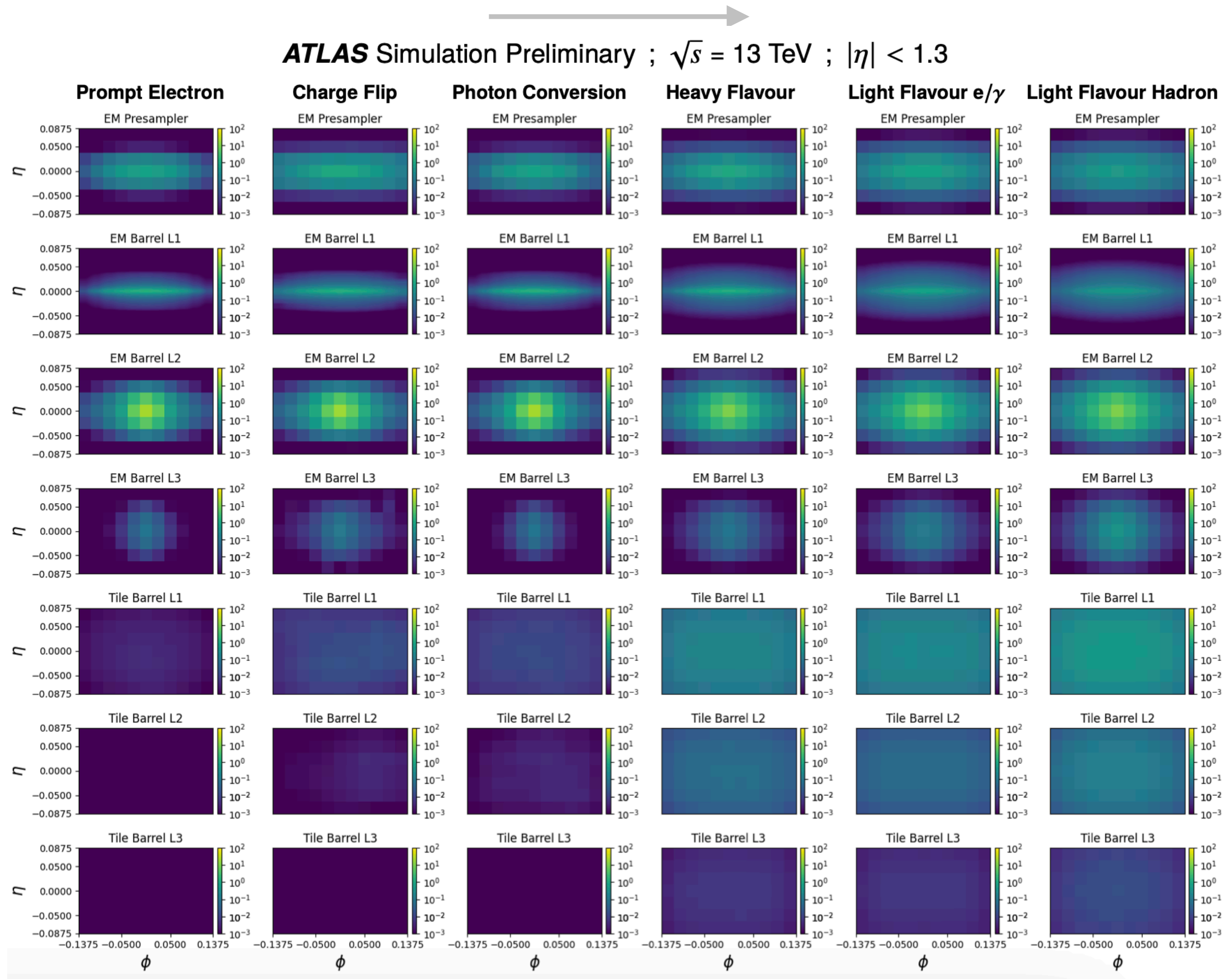
Images



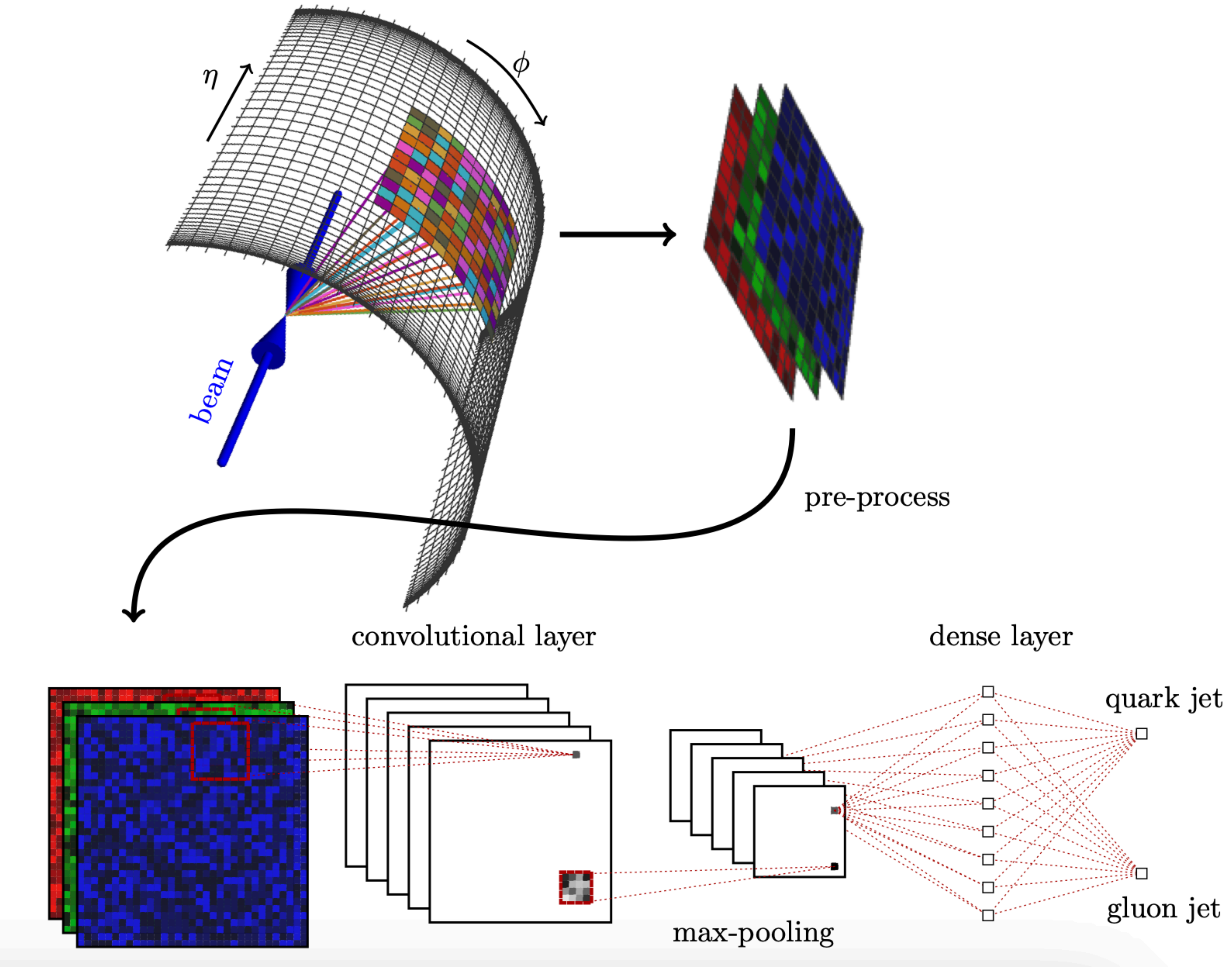
Hongtian Yu et al, Vision Calorimeter, arXiv:2408.10599

Electron classes

Calorimeter layers

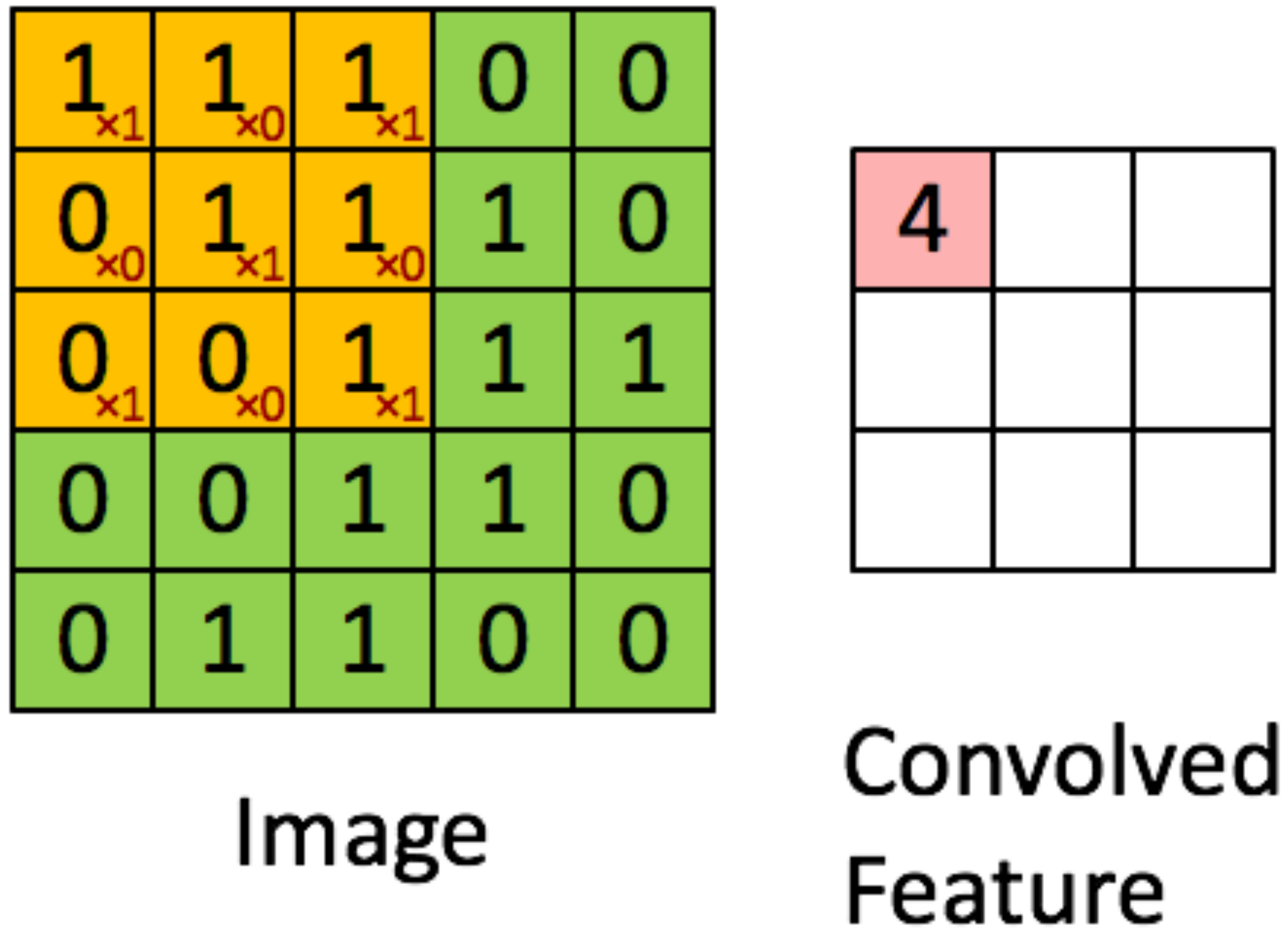


ATL-PHYS-PUB-2023-001

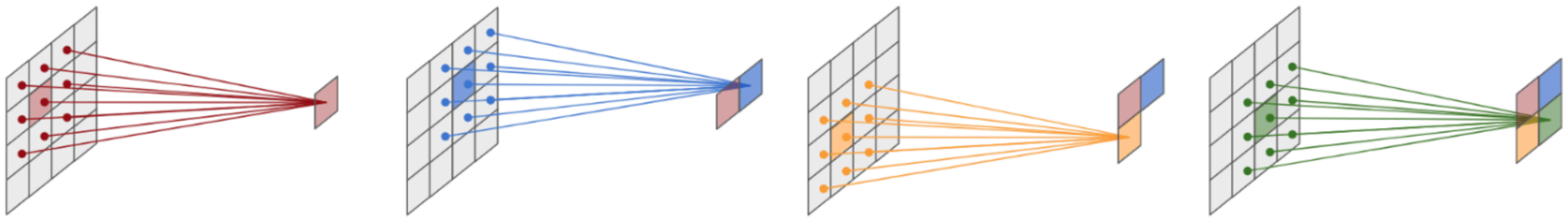


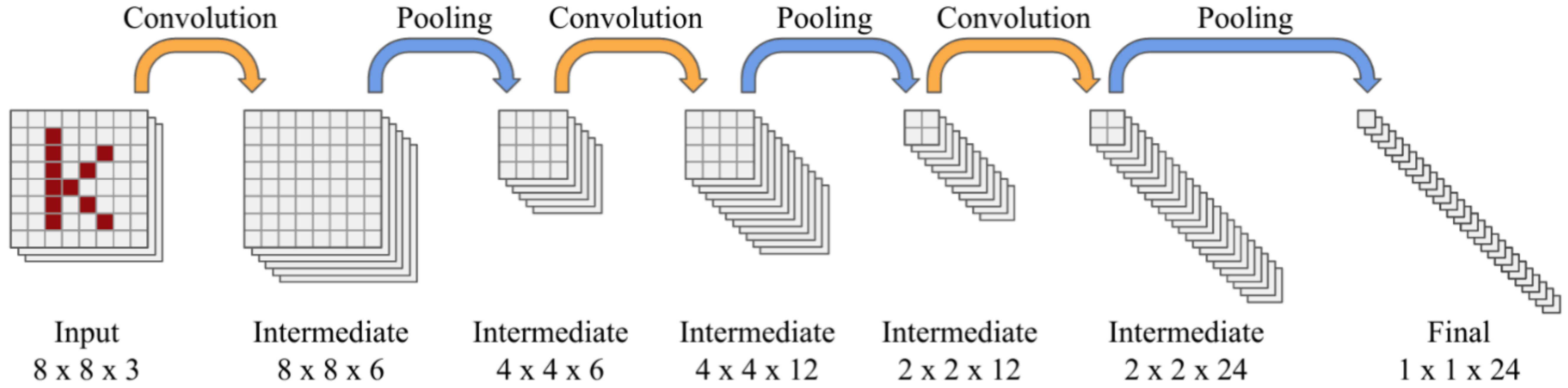
JHEP 01 (2017) 110

Convolutional neural networks (CNN)



Convolution operation:





Success in applying CNNs to a variety of problems:

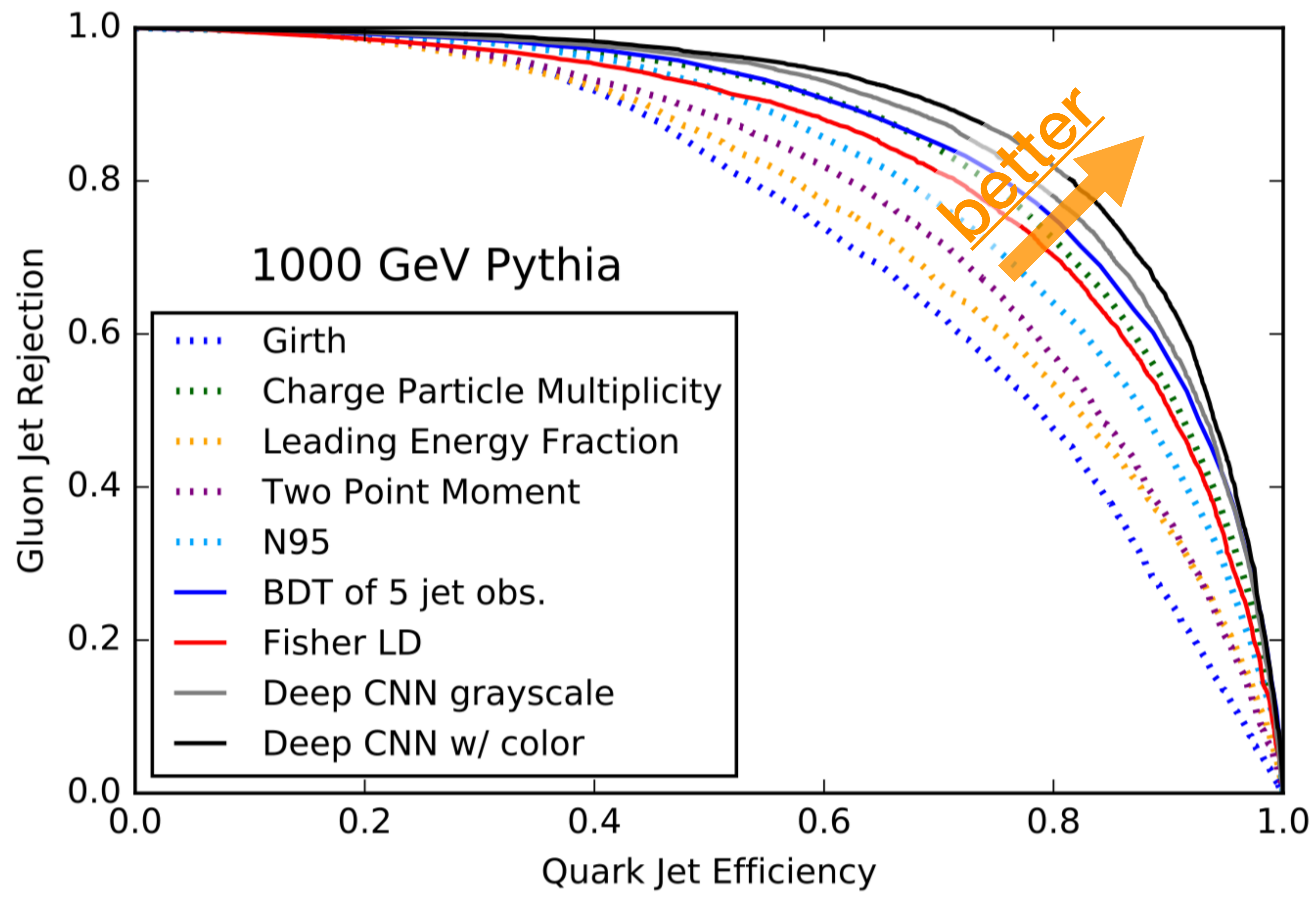
- Computer vision
- Face Recognition
- Medical Imaging

PDG Machine Learning

Goodfellow et al. Deep learning. MIT press, 2016.

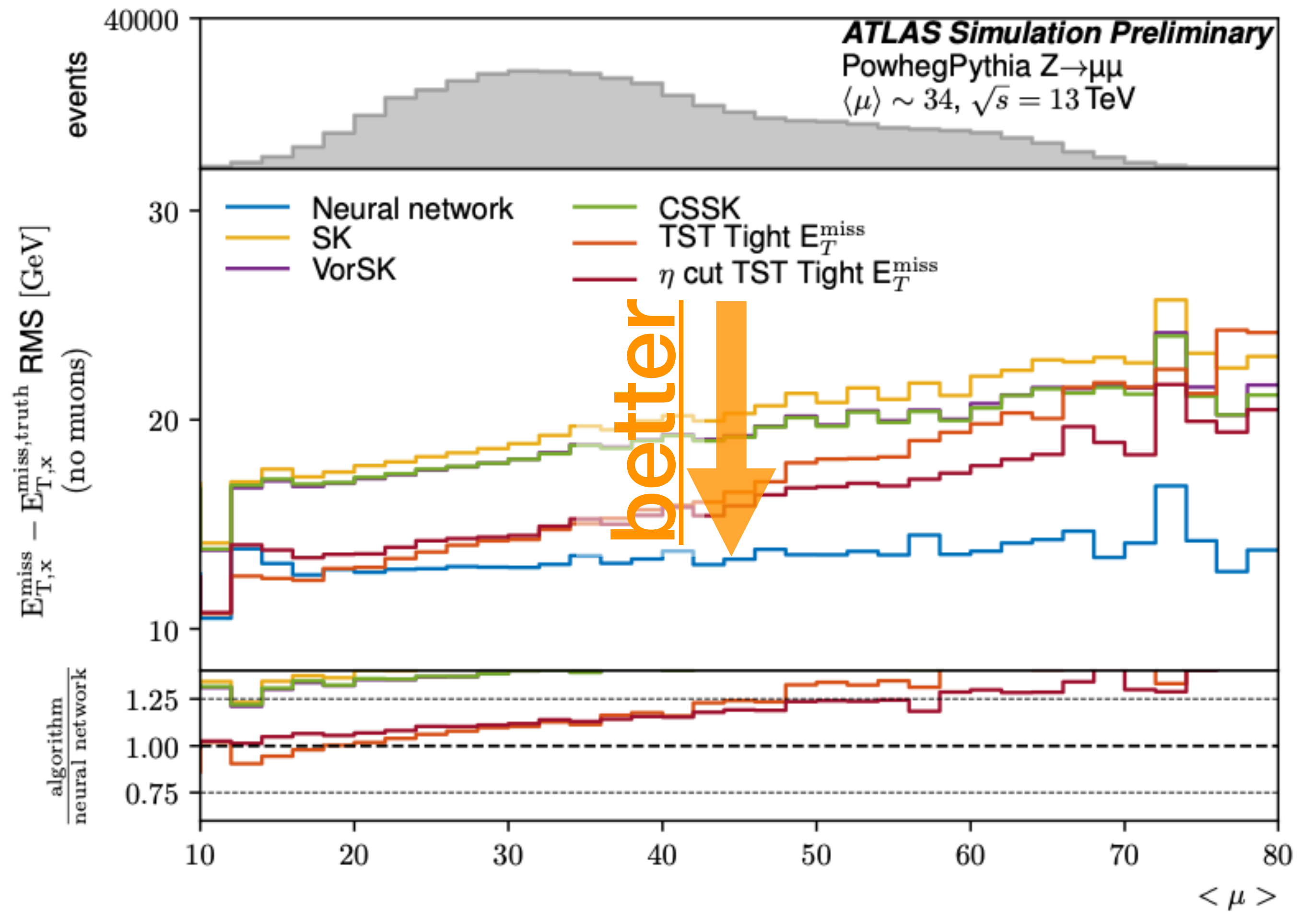
CNN applications

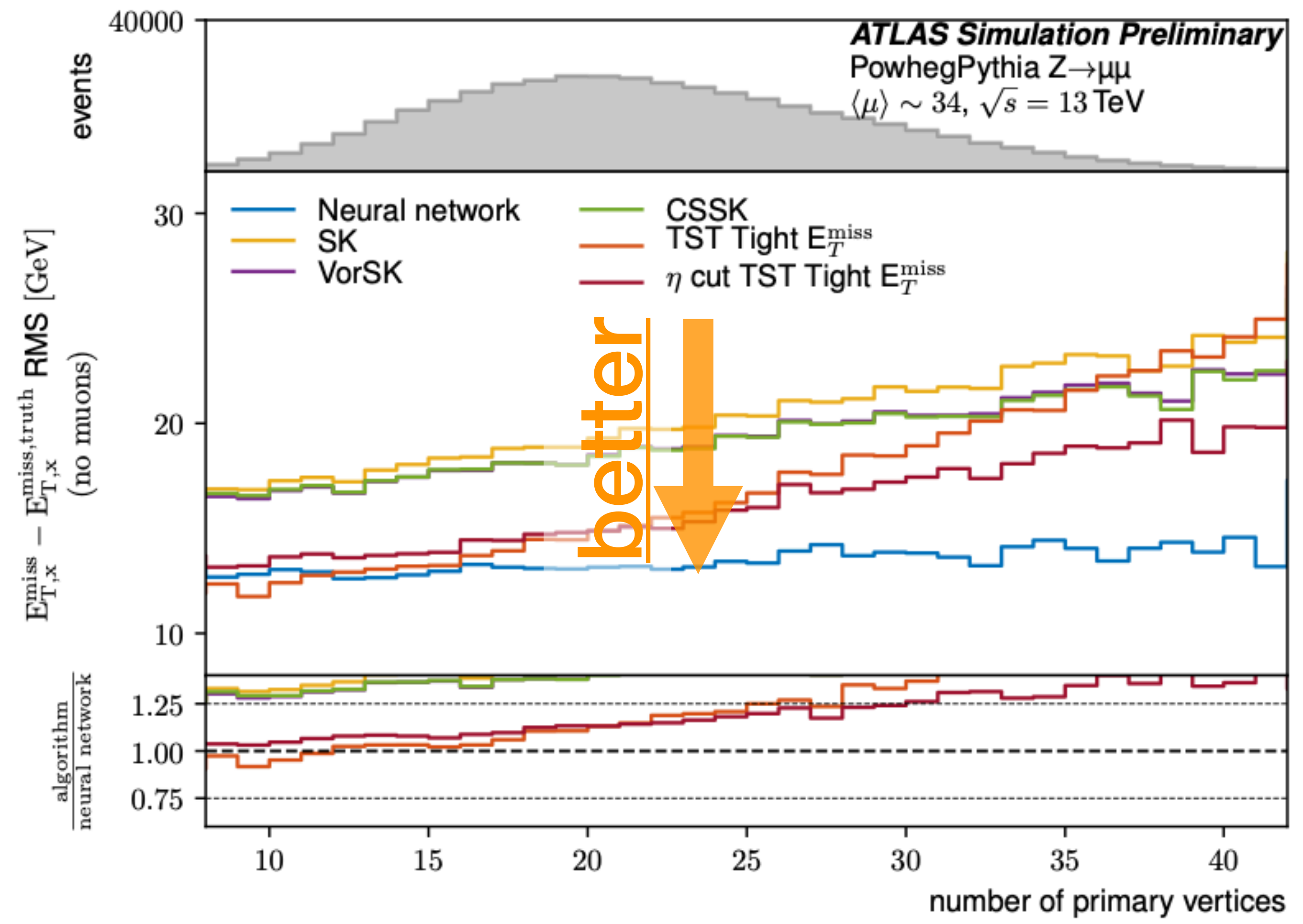
Quark-gluon jet discrimination



JHEP 01 (2017) 110

E_T^{miss} reconstruction



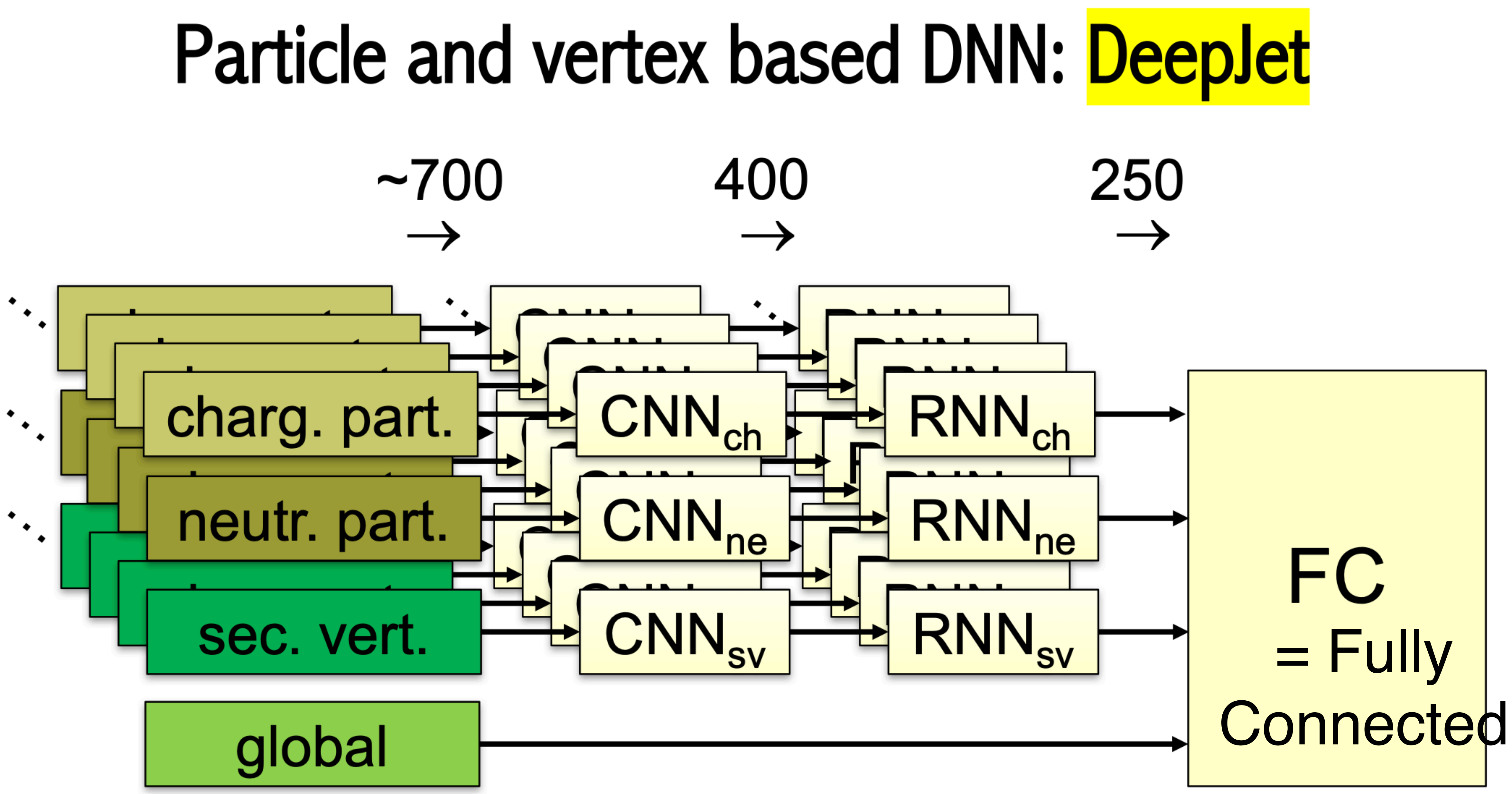


Flat in pileup and primary vertices

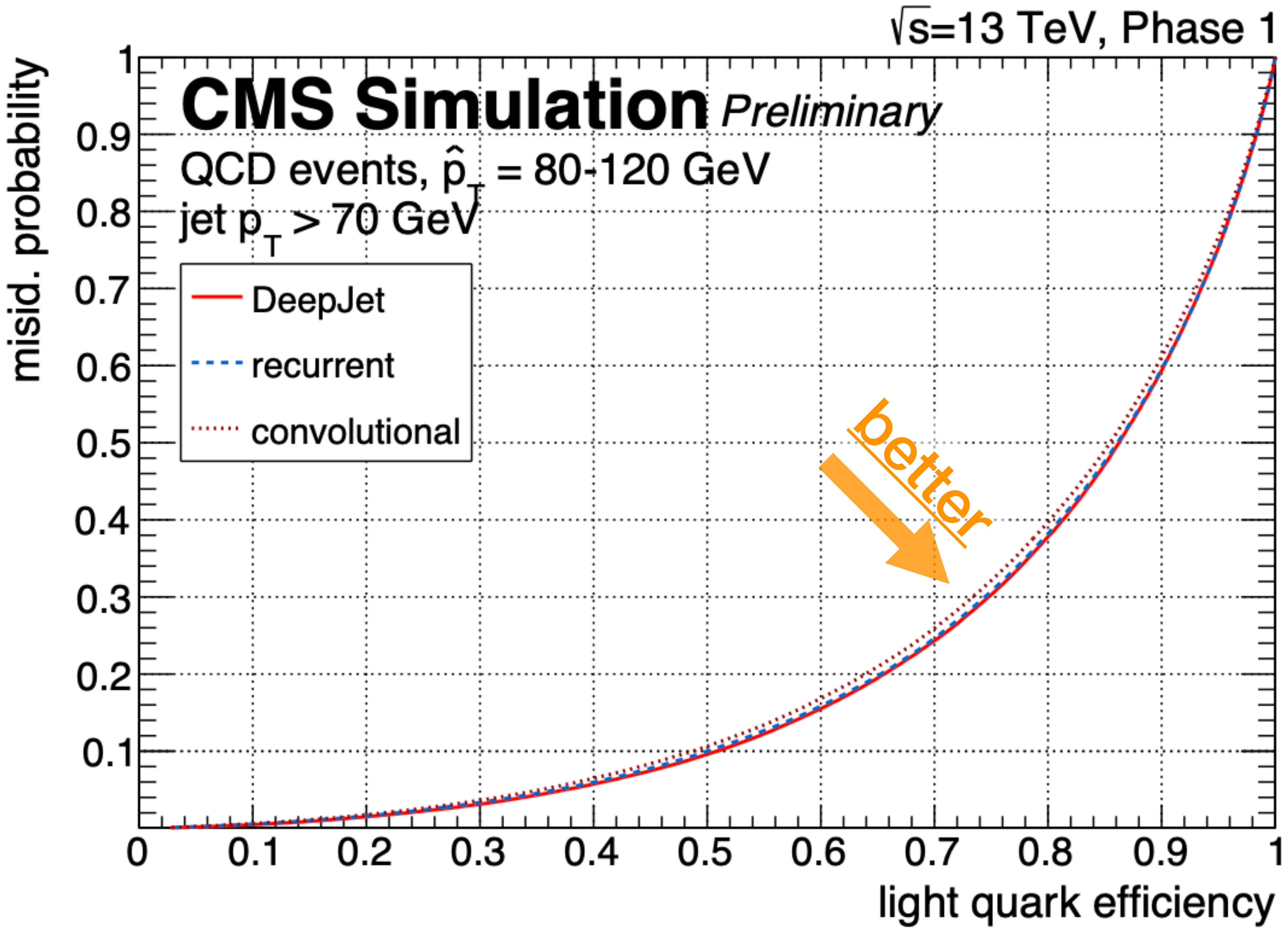
ATL-PHYS-PUB-2019-028

Hybrid: DNN + RNN + CNN application

CMS DeepJet algorithm used CNN, RNN and fully connected DNN at the same time



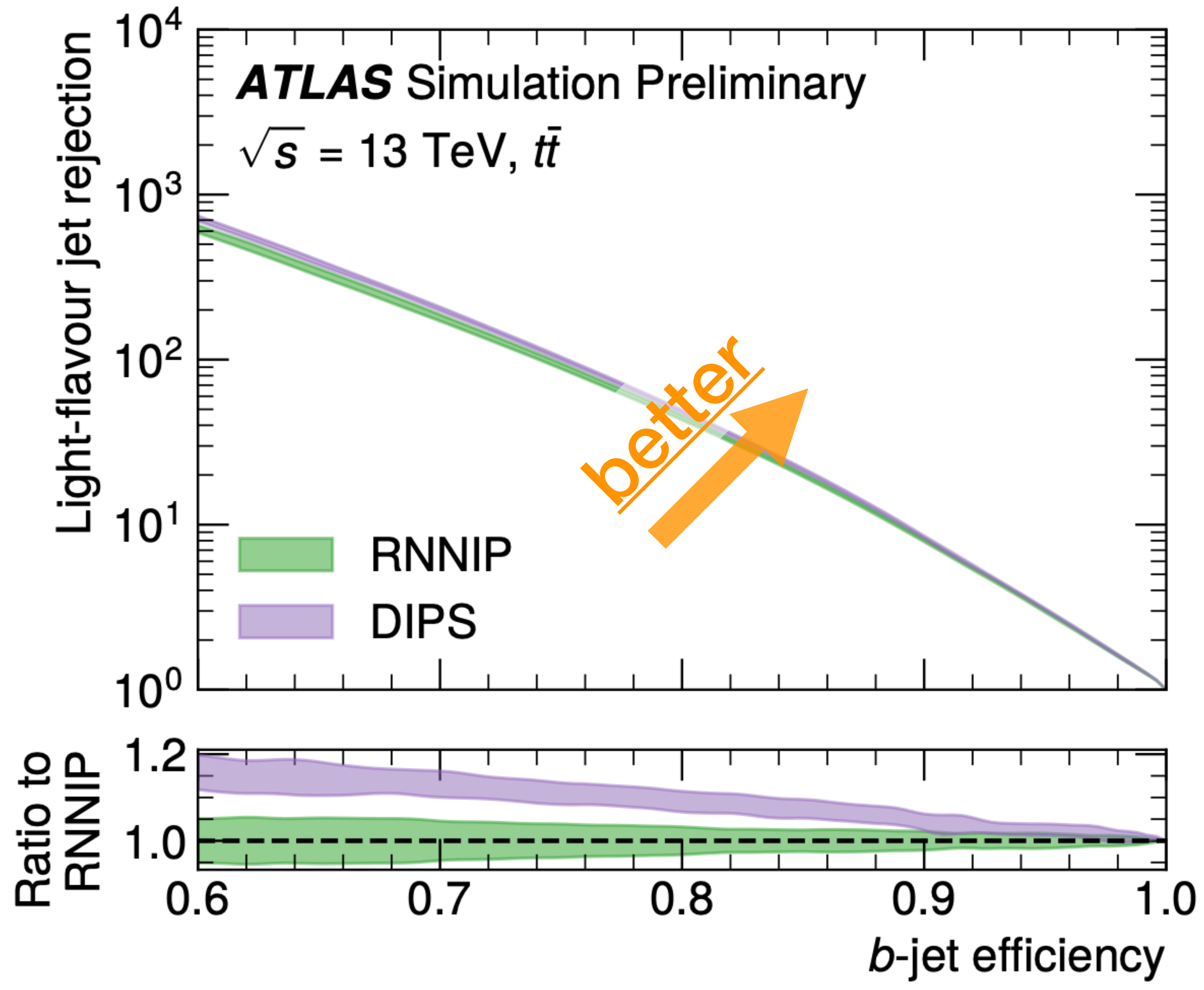
~ 700 inputs and 250.000 model parameters

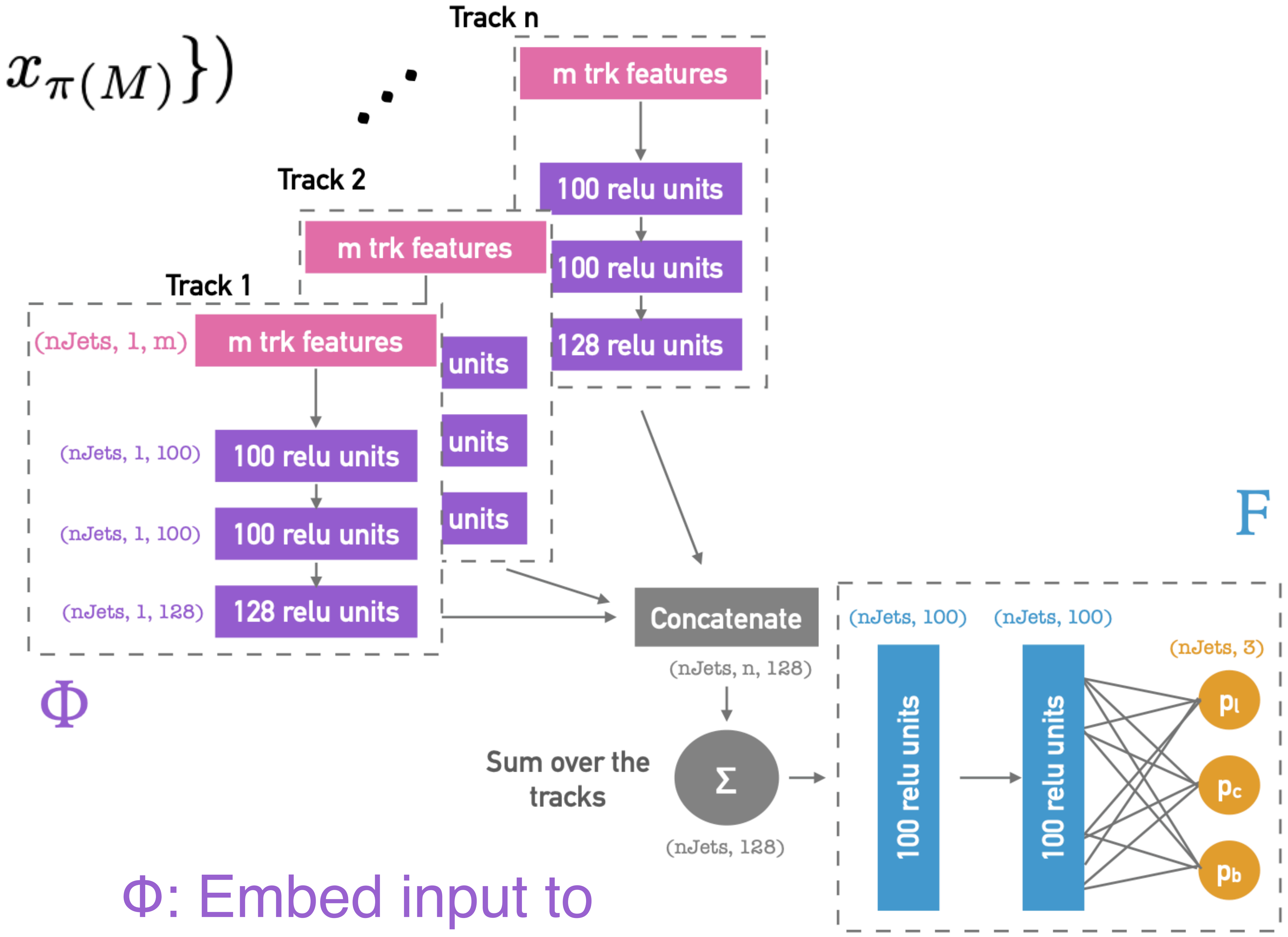


CERN-CMS-DP-2017-027

Sets

- Sequences and images imply a certain ordering
  - Lack of permutation invariance $f(x_1, x_2) \neq f(x_2, x_1)$
- Deepset [Manzil et al]
  - *for any permutation* $\pi : f(\{x_1, \ldots, x_M\}) = f(\{x_{\pi(1)}, \ldots, x_{\pi(M)}\})$
  - e.g. f = max, mean, etc

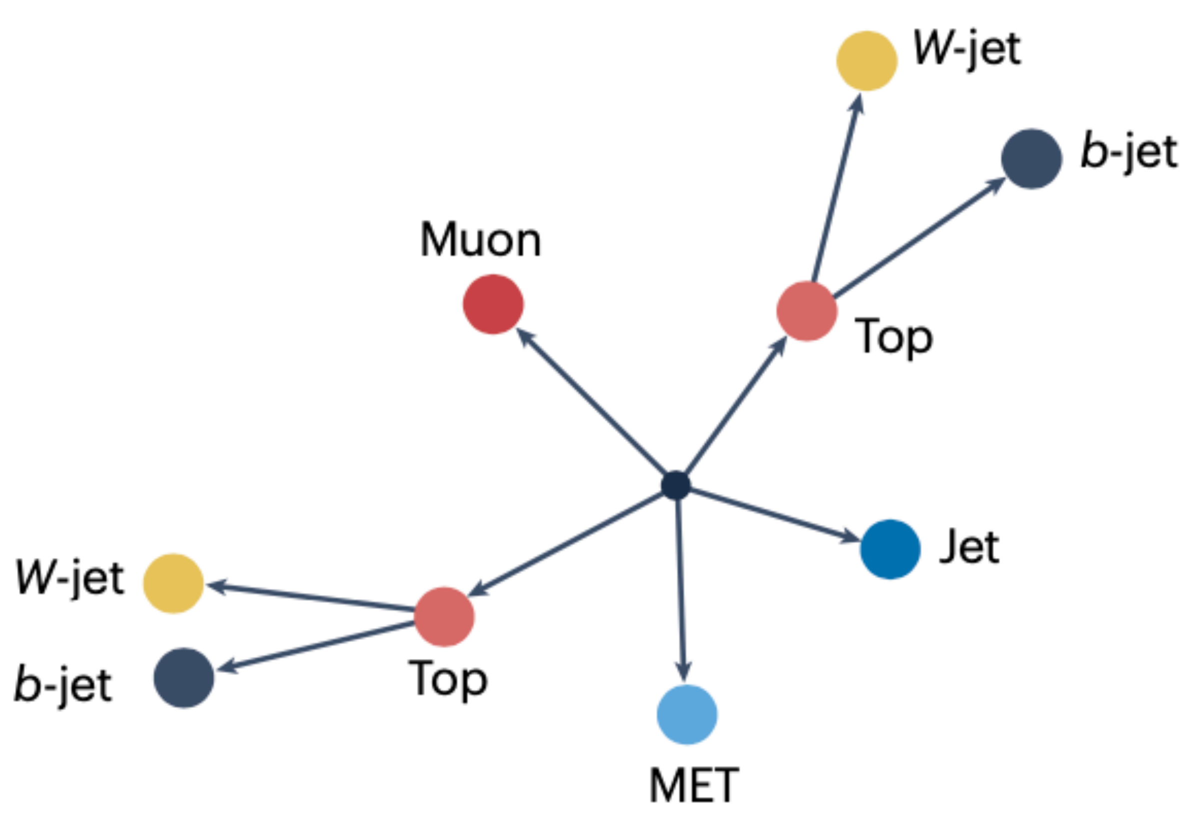


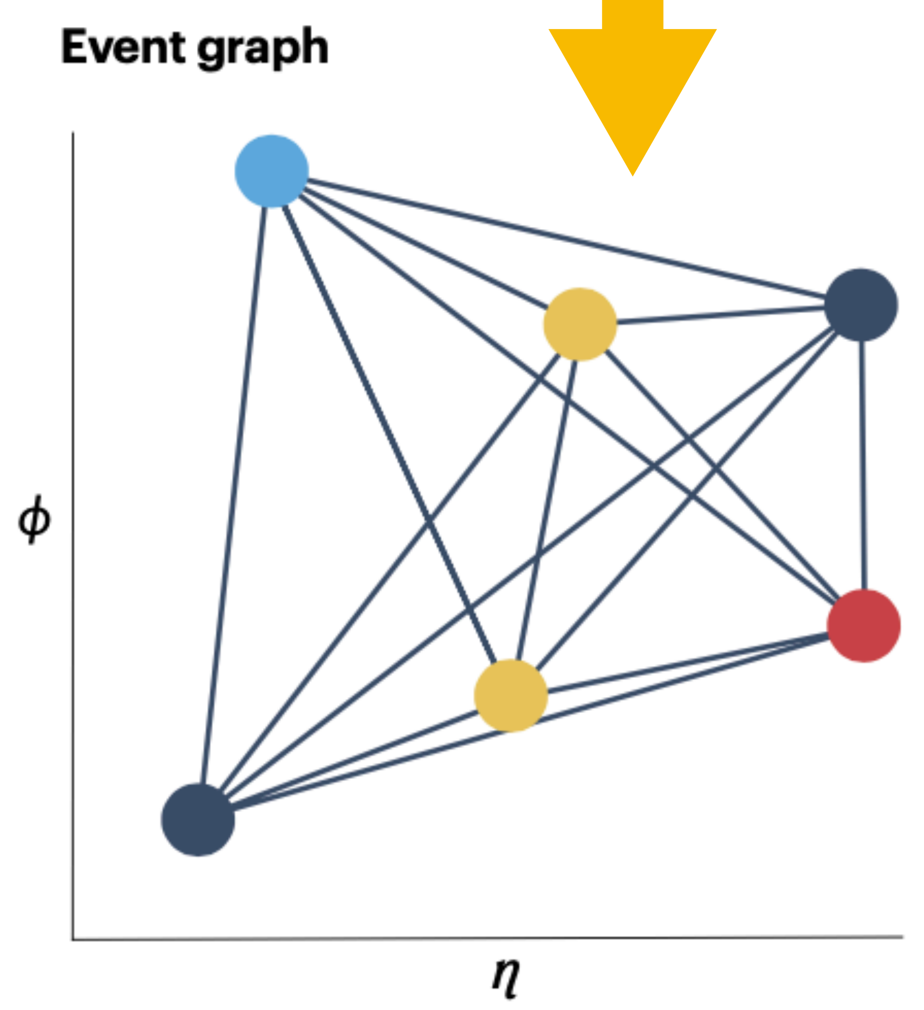


Φ: Embed input to high-dim space to preserve properties

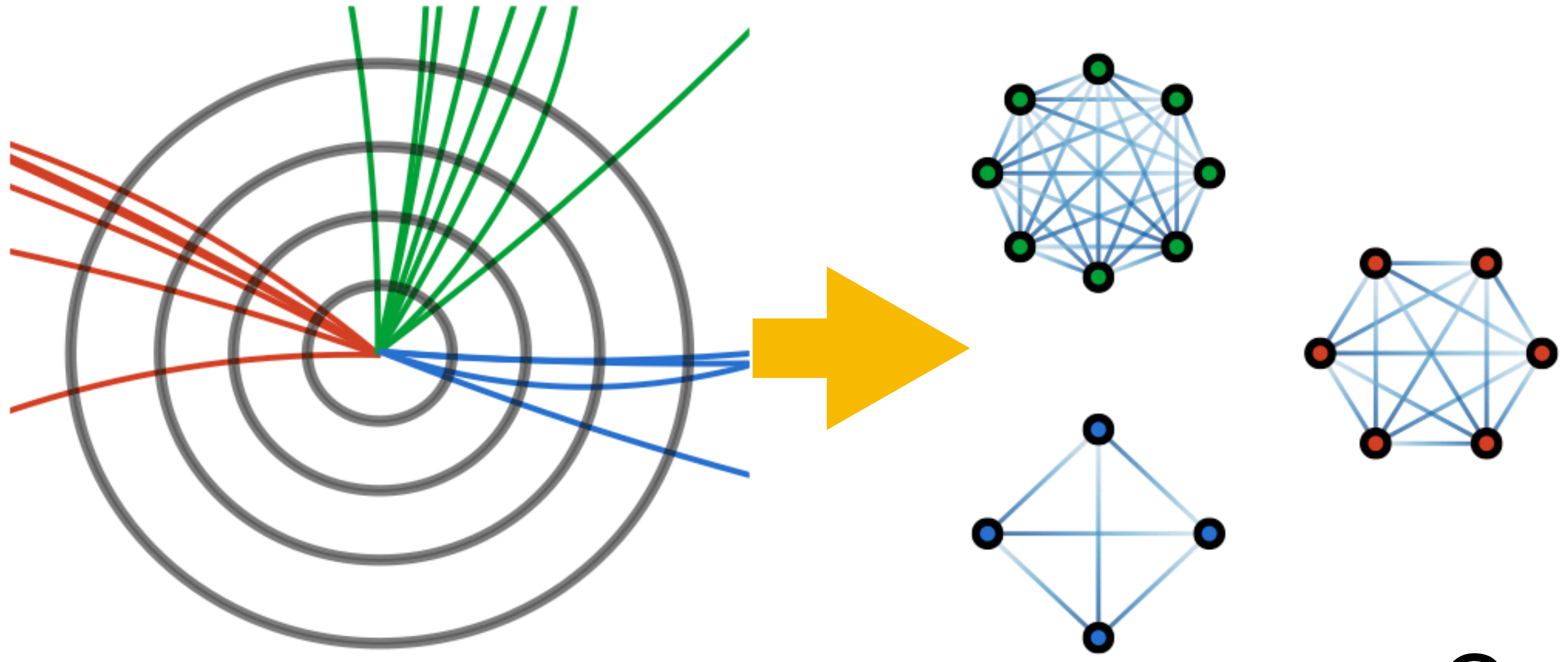
Graphs

A $t\bar{t}$ event

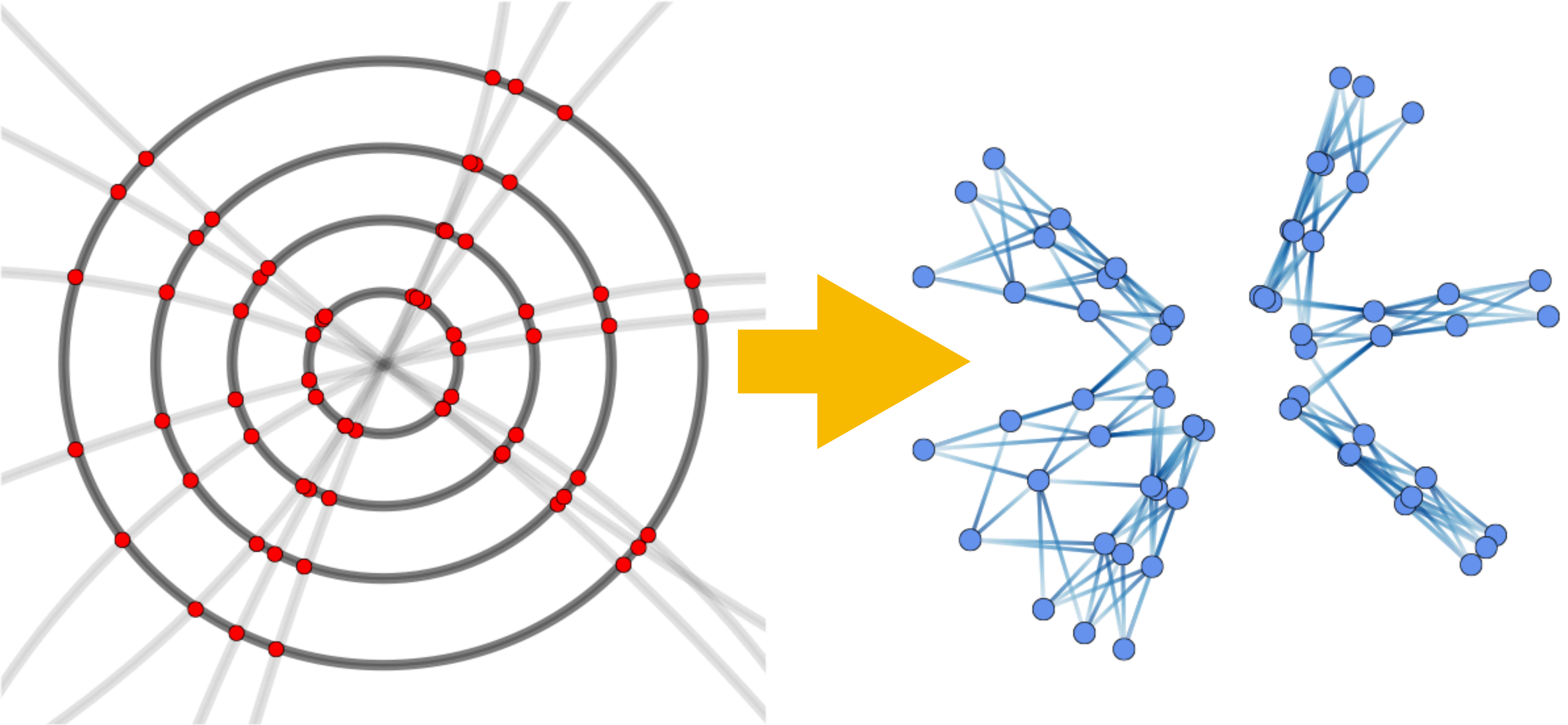




Jet is a graph of particles



Hits in tracker



Graph neural networks in particle physics

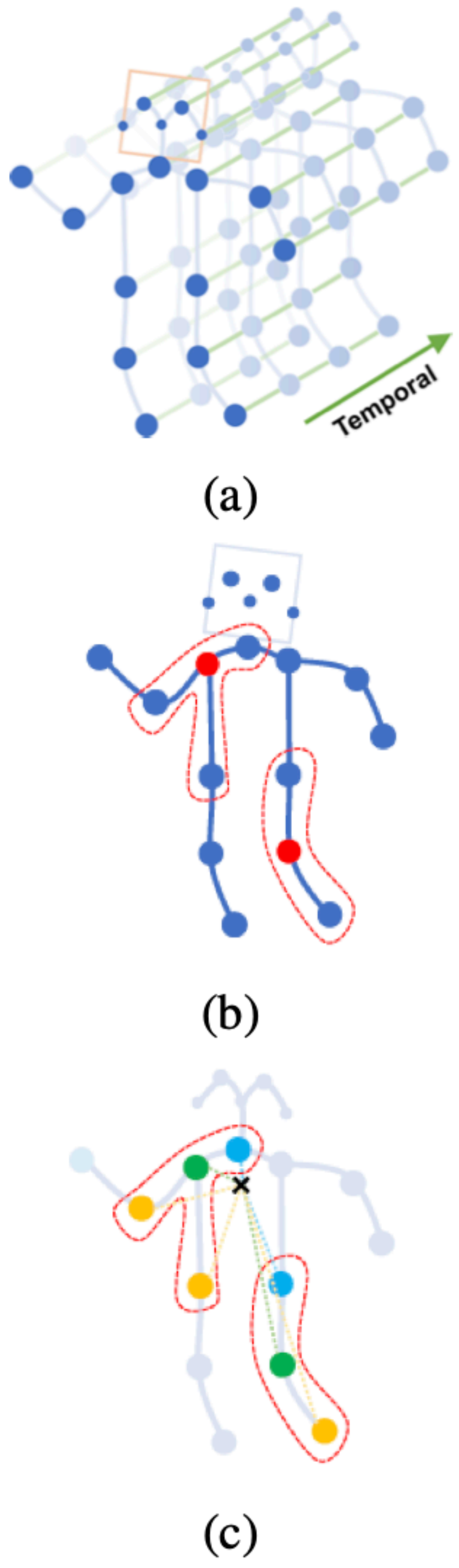
Graph neural networks at the Large Hadron Collider

Graph Neural Networks for Particle Tracking and Reconstruction

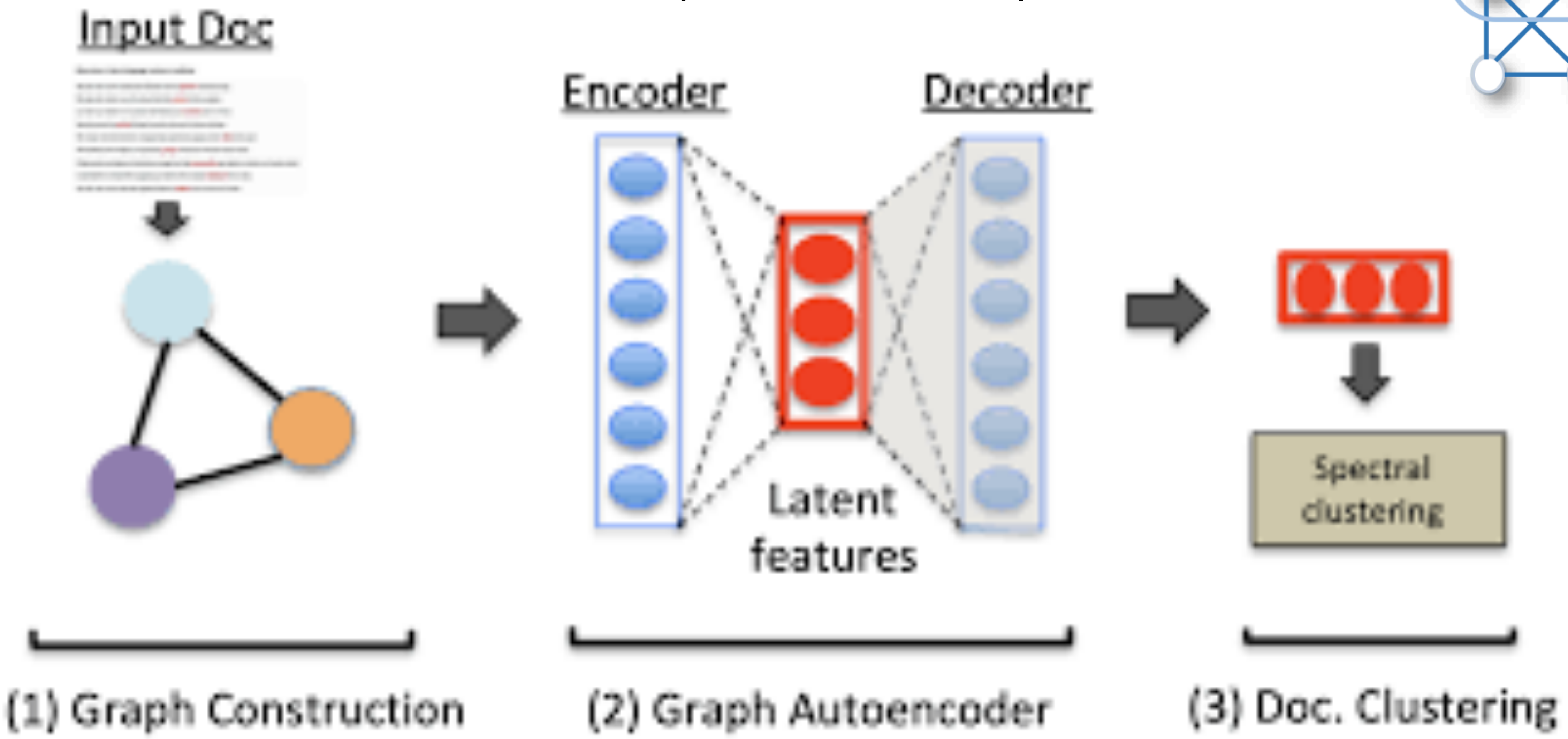
Graph Neural Networks in Particle Physics: Implementations, Innovations, and Challenges

Graphs neural networks

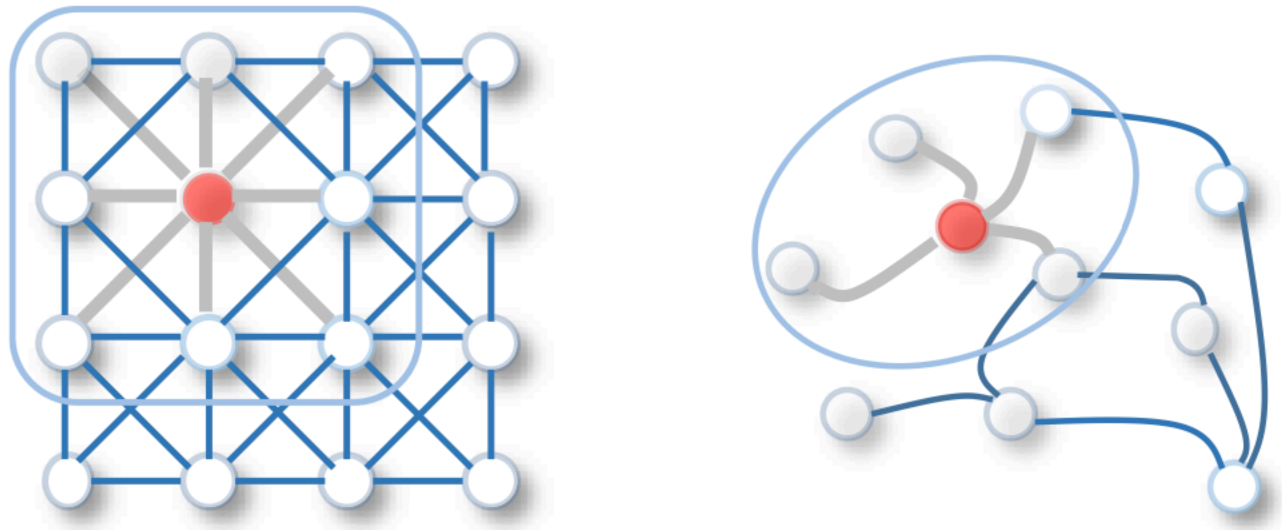
Spatial-temporal graph neural networks (STGNNs)



Convolutional graph neural networks (ConvGNNs)



Graph autoencoders (GAEs)

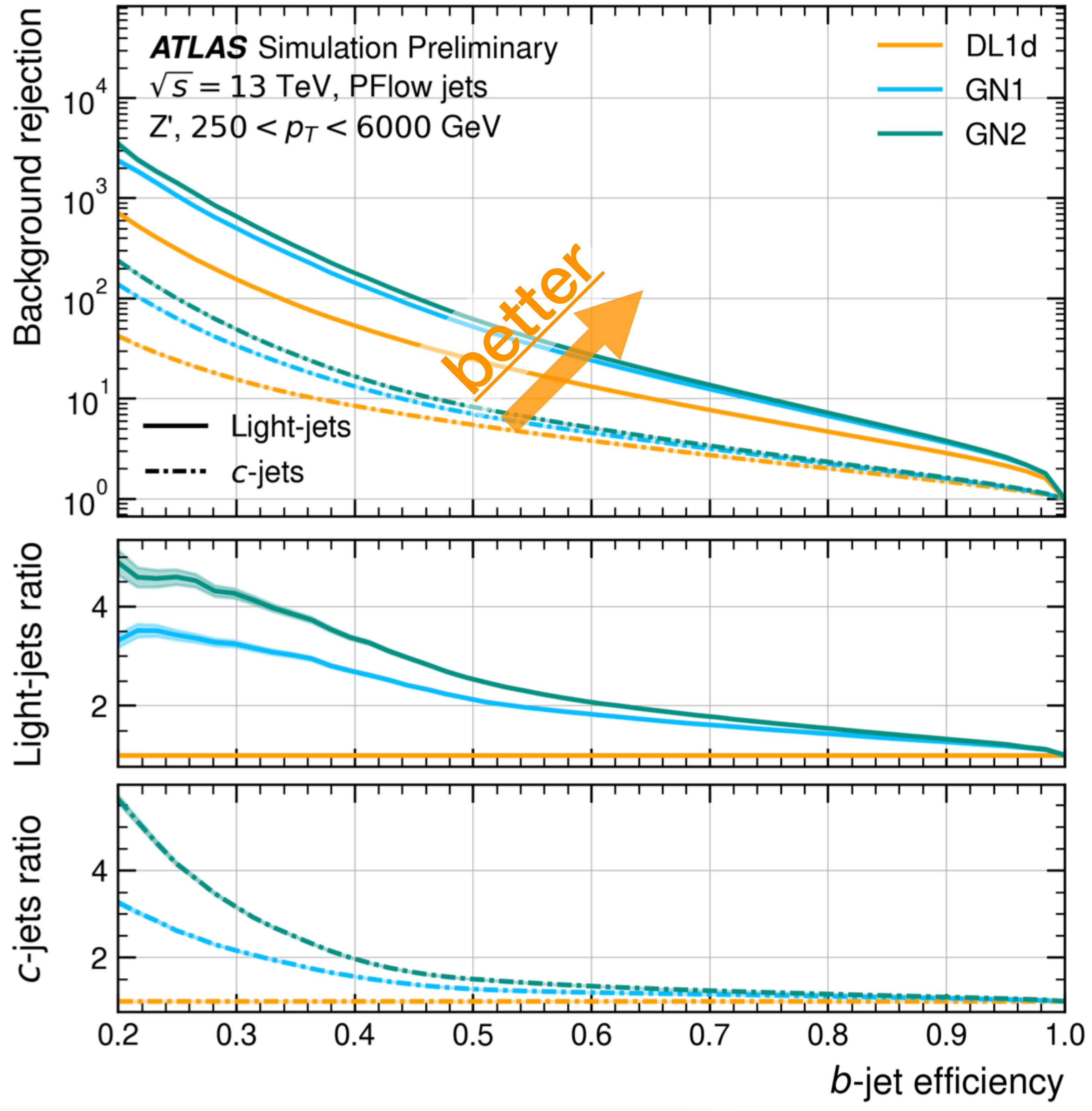


[A Comprehensive Survey on Graph Neural Networks]

[Spatial-Temporal Graph Convolutional Networks for Sign Language Recognition]

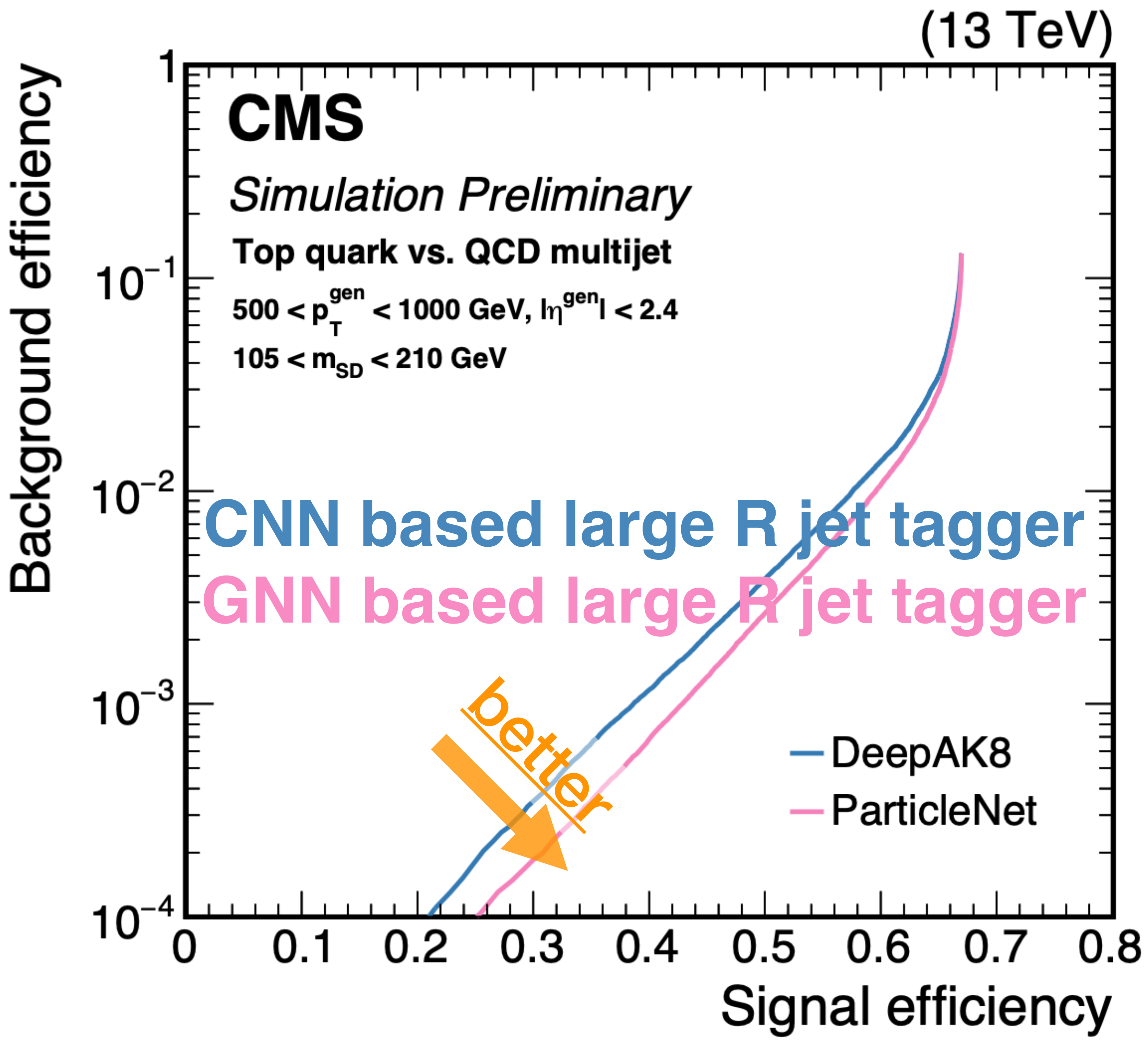
GNN applications

ATLAS b-tagging



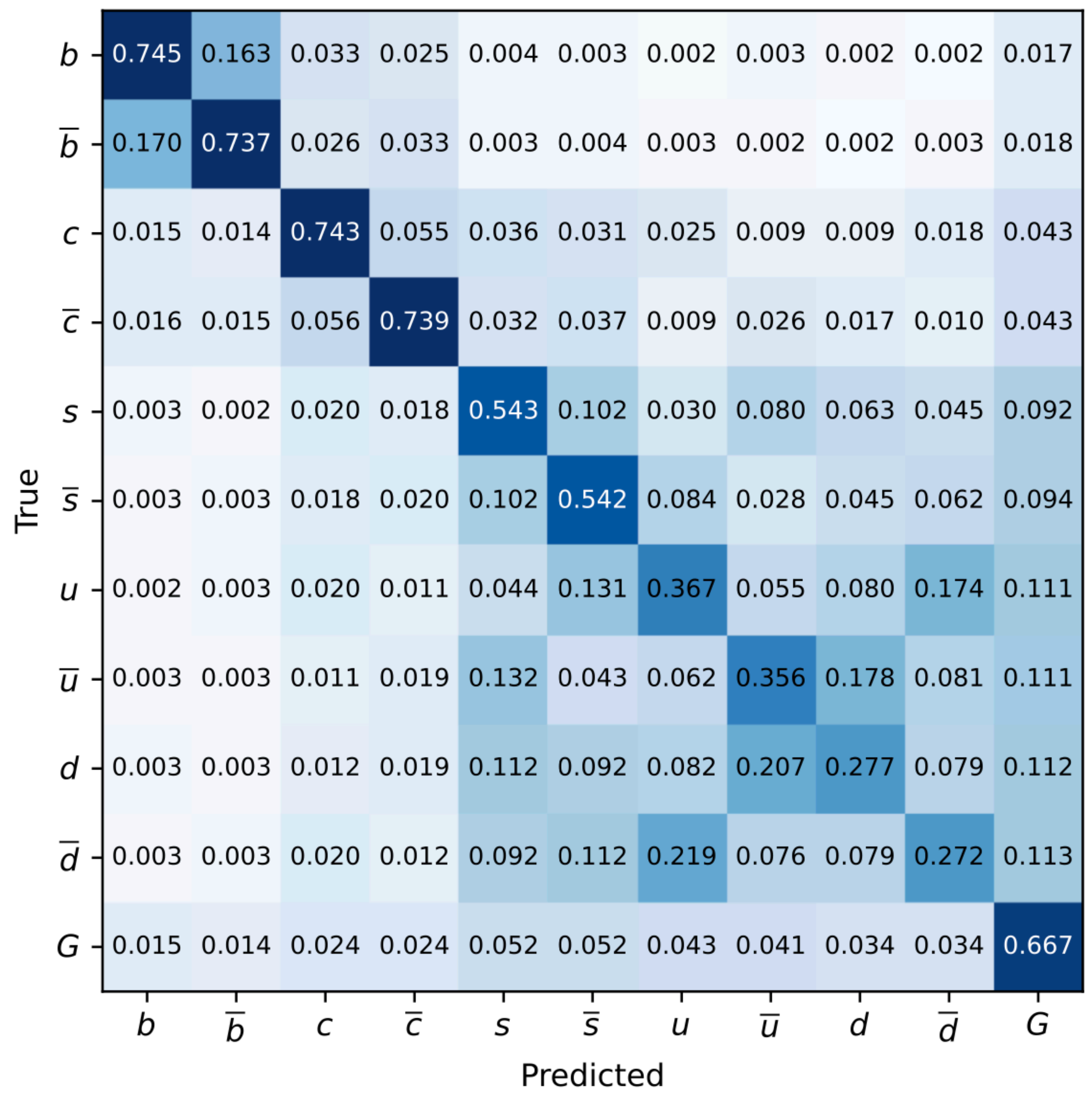
arXiv:1706.03762

CMS b-tagging



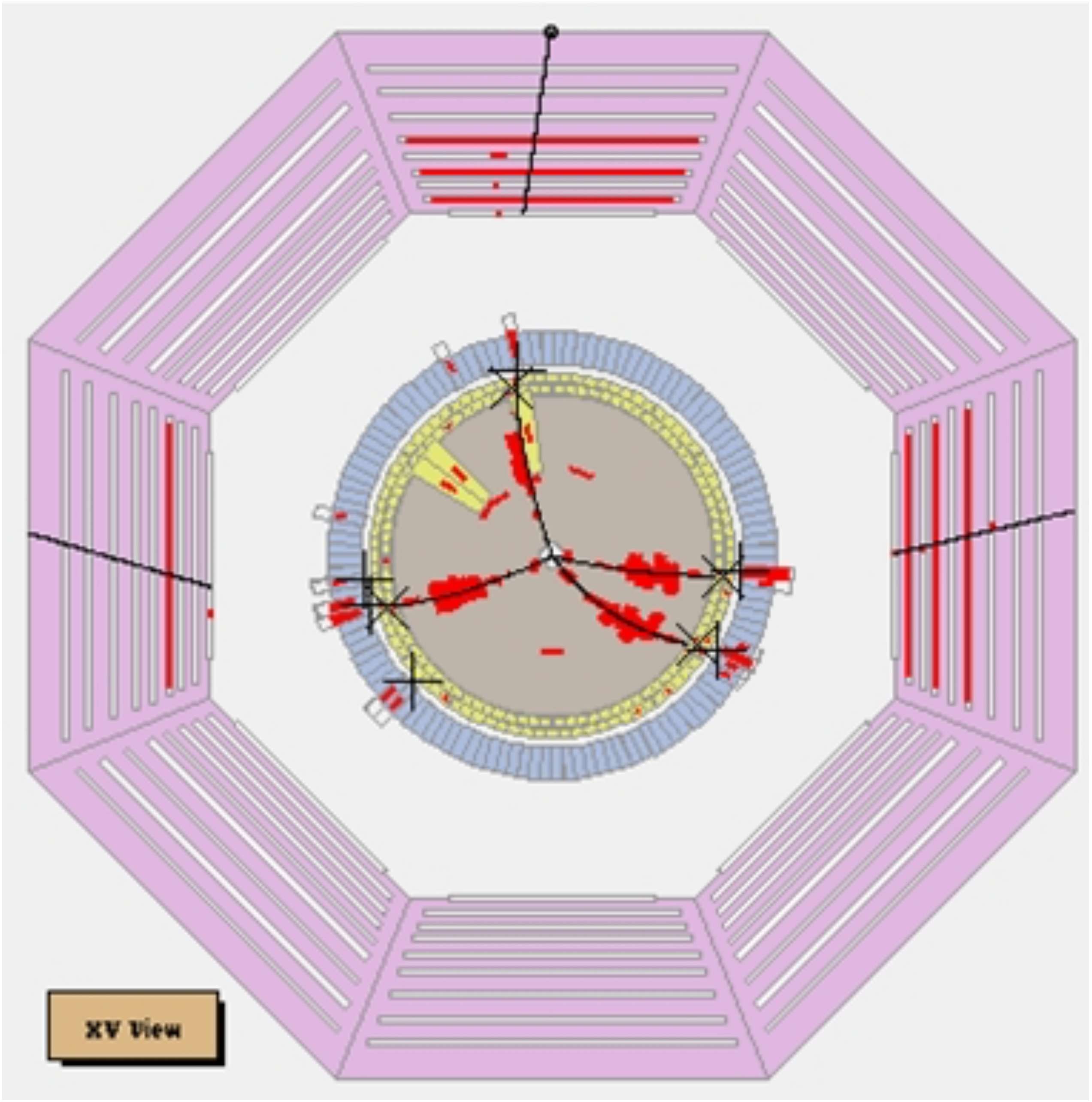
CMS-DP-2020-002

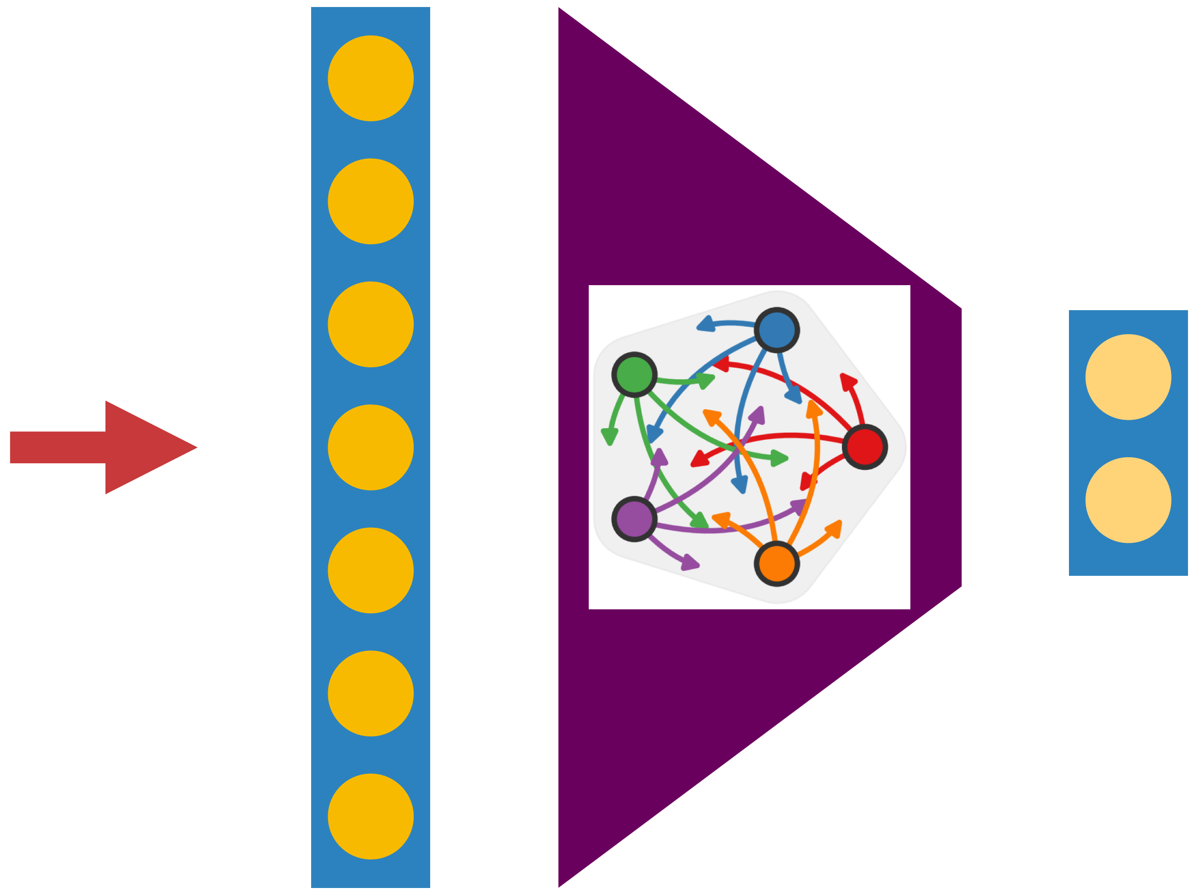
Jet origin identification at CEPC simulation



PhysRevLett.132.221802

Two paradigms





Un-supervised learning

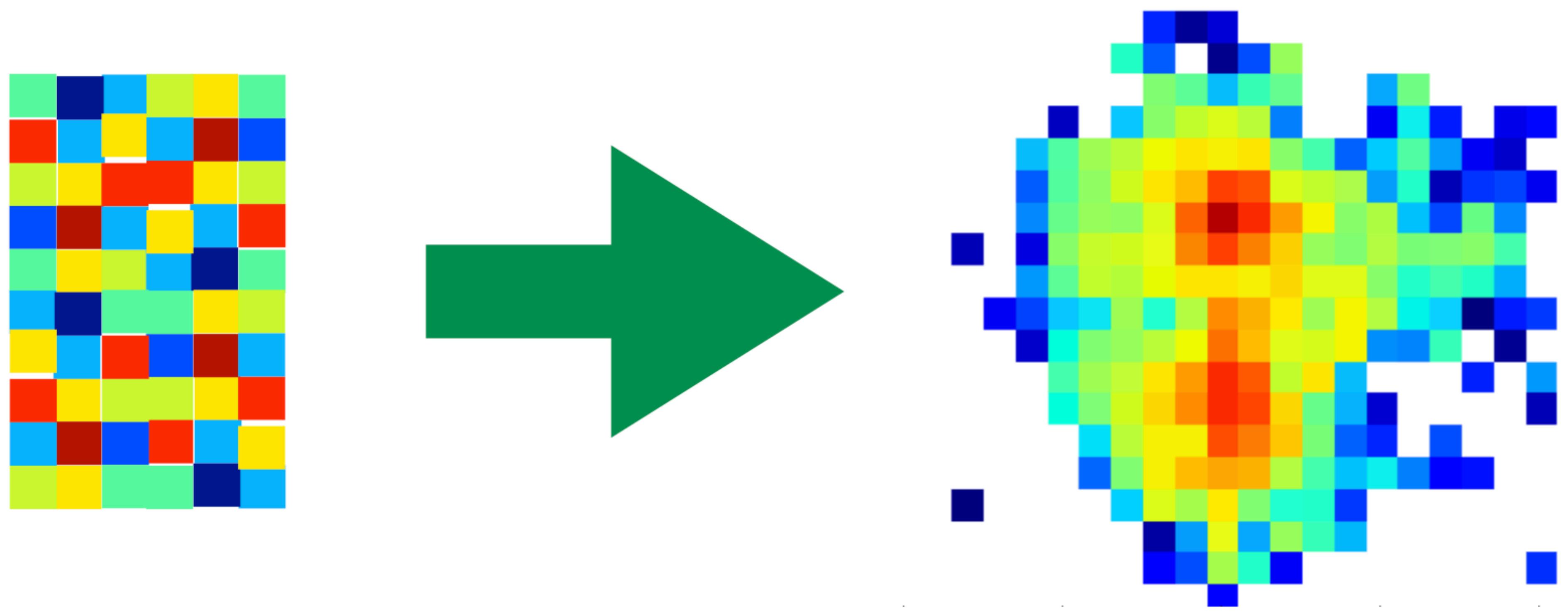
Supervised learning

High *Dimension* *Low*

Un-supervised learning

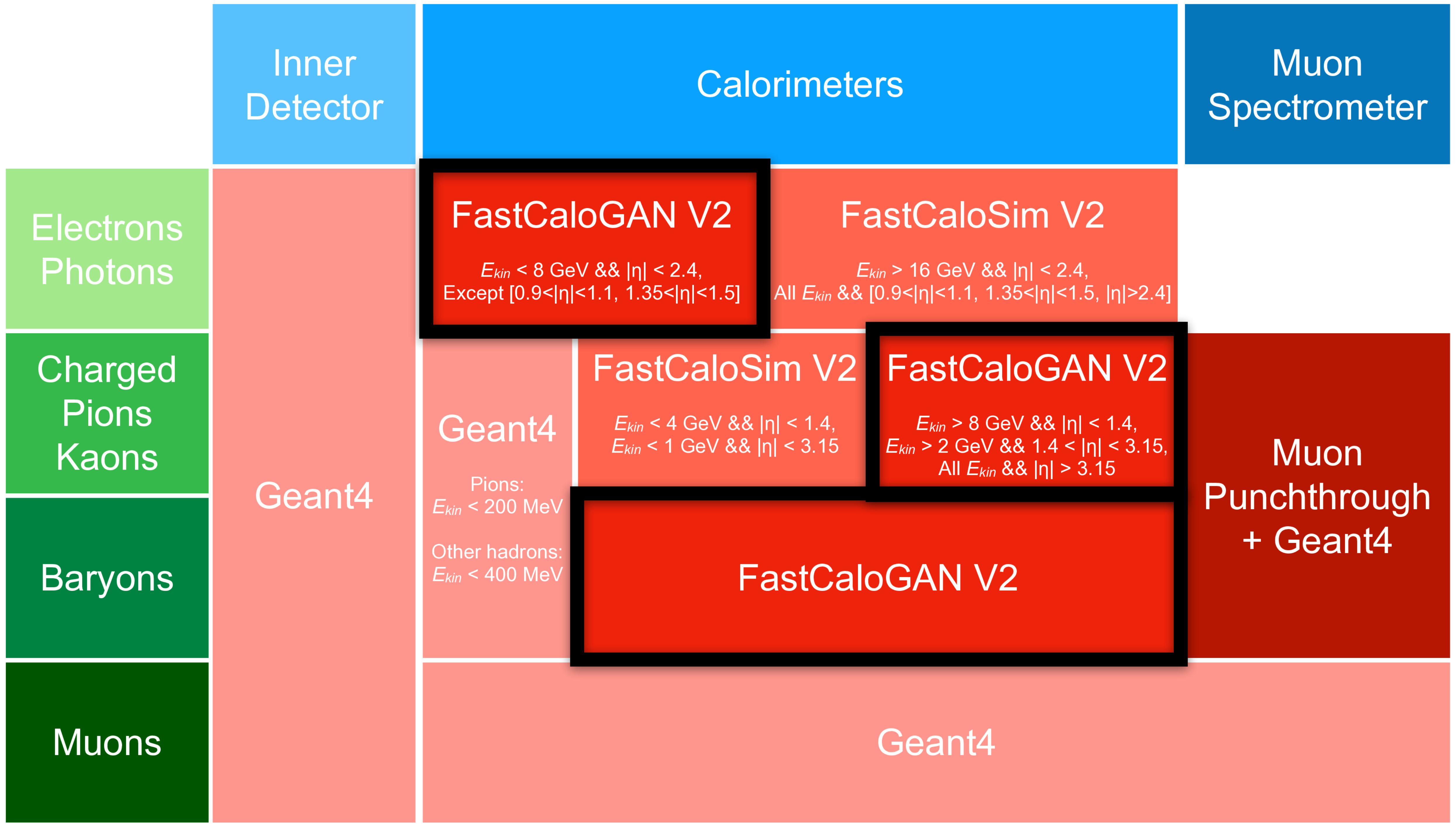
Simulation

A generator is a function that maps random numbers to structure.



Low dimension *High dimension*

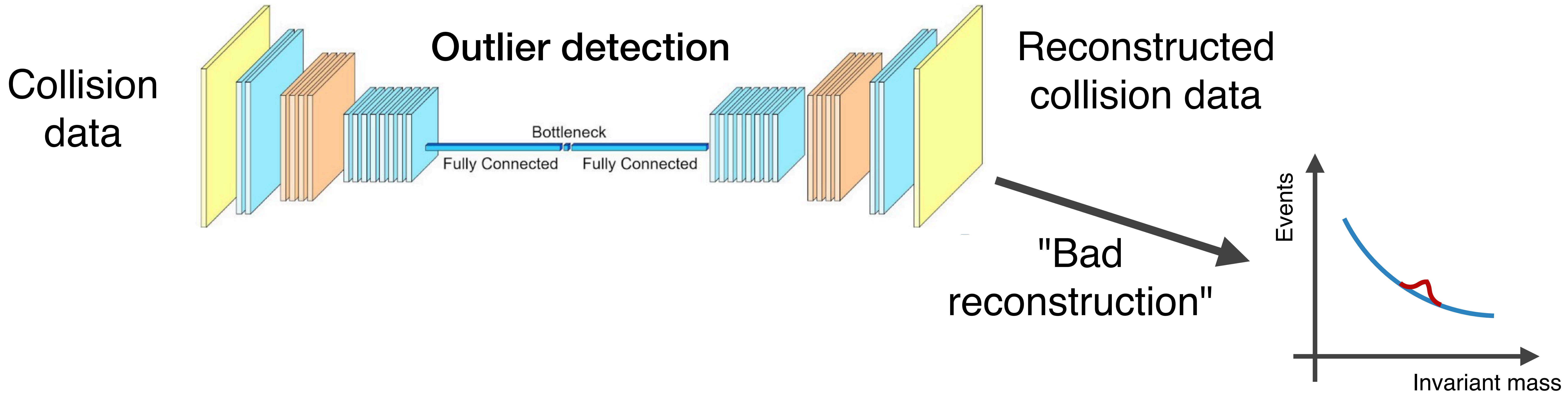
Generative Adversarial Net (GAN) has been used in the ATLAS Fast simulation

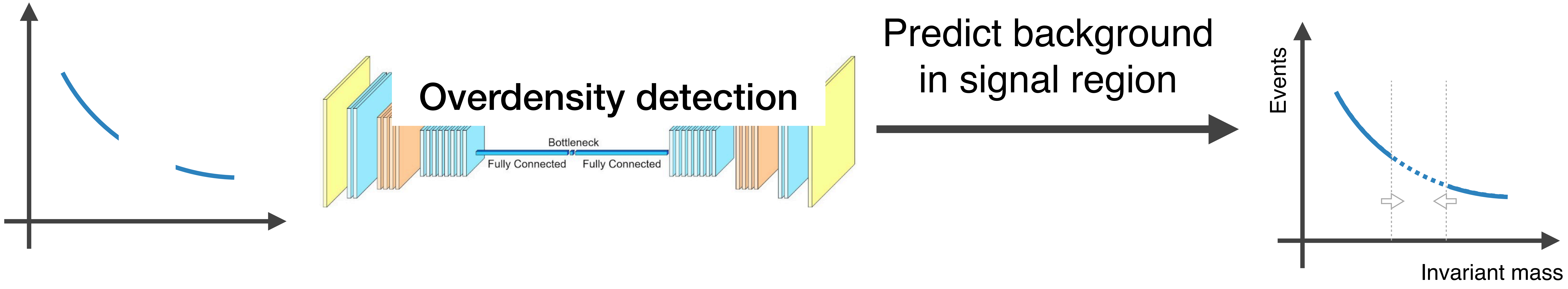


Eur. Phys. J. C 85 (2025) 234

Un-supervised learning

Anomaly detection for model-agnostic new physics searches



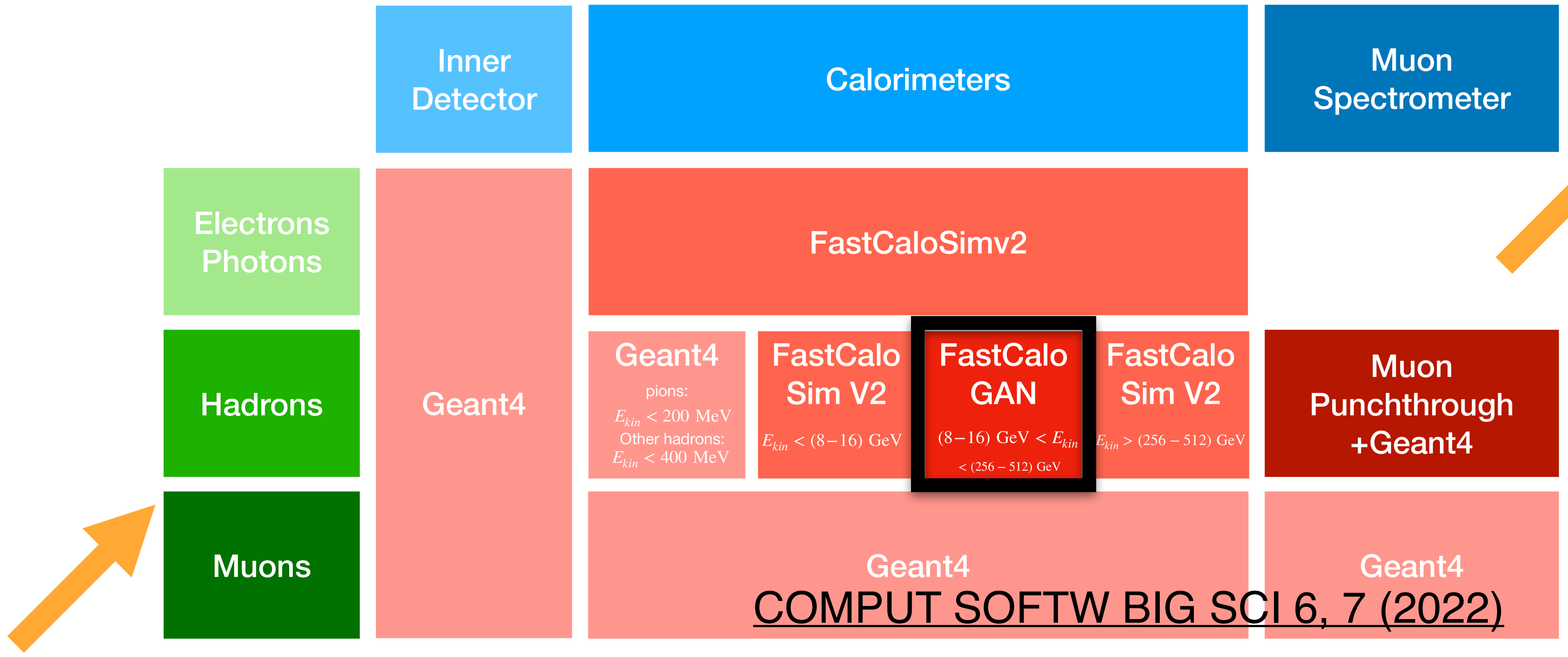


Contents

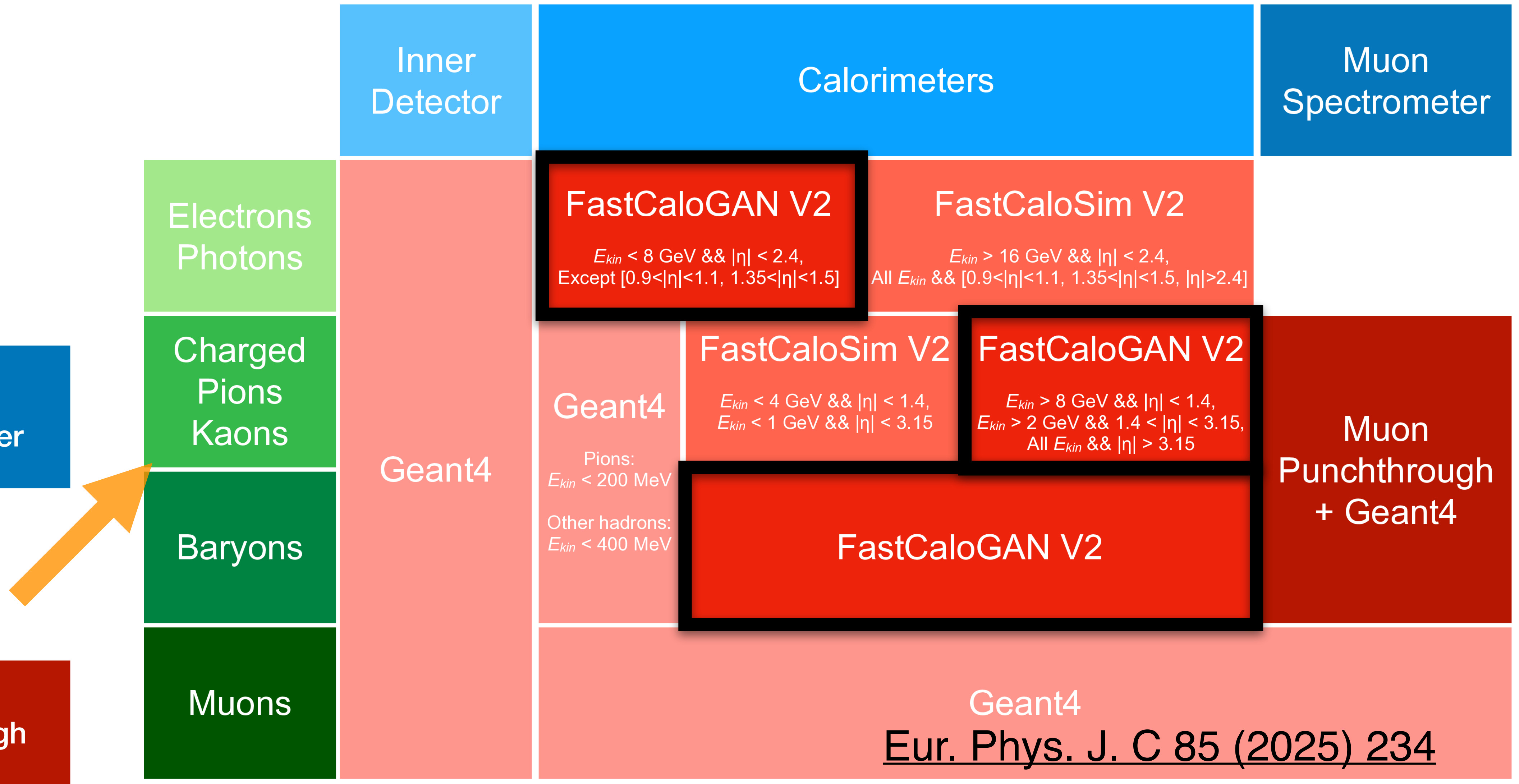
- Motivation to use machine learning (ML)
- How to map a HEP problem to a ML problem
- Examples of recent ML applications
- Summary

Fast calo shower simulation using GAN

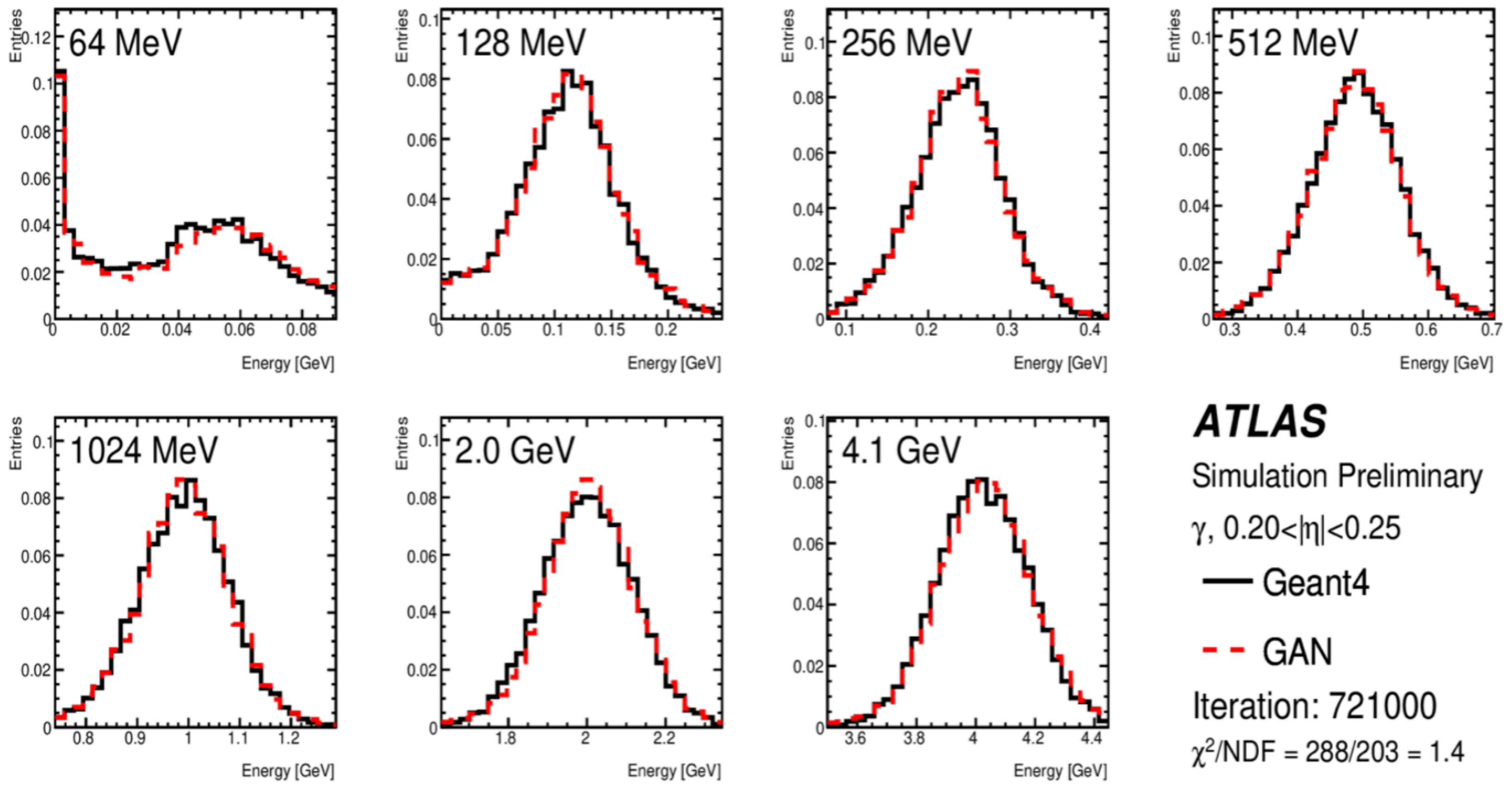
- Combination of classical histogram-based parametrisation and GAN based ML model



COMPUT SOFTW BIG SCI 6, 7 (2022)



Eur. Phys. J. C 85 (2025) 234

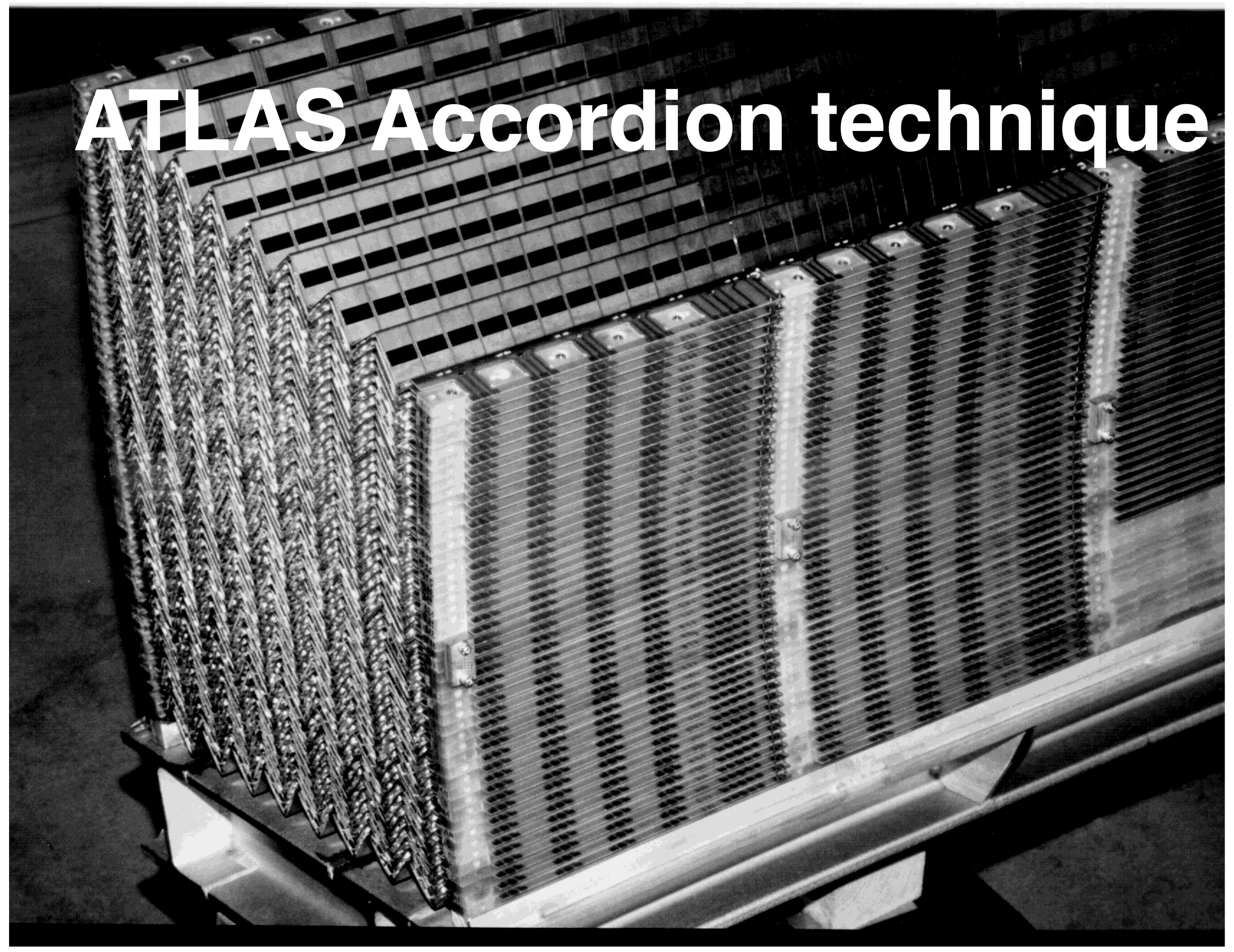


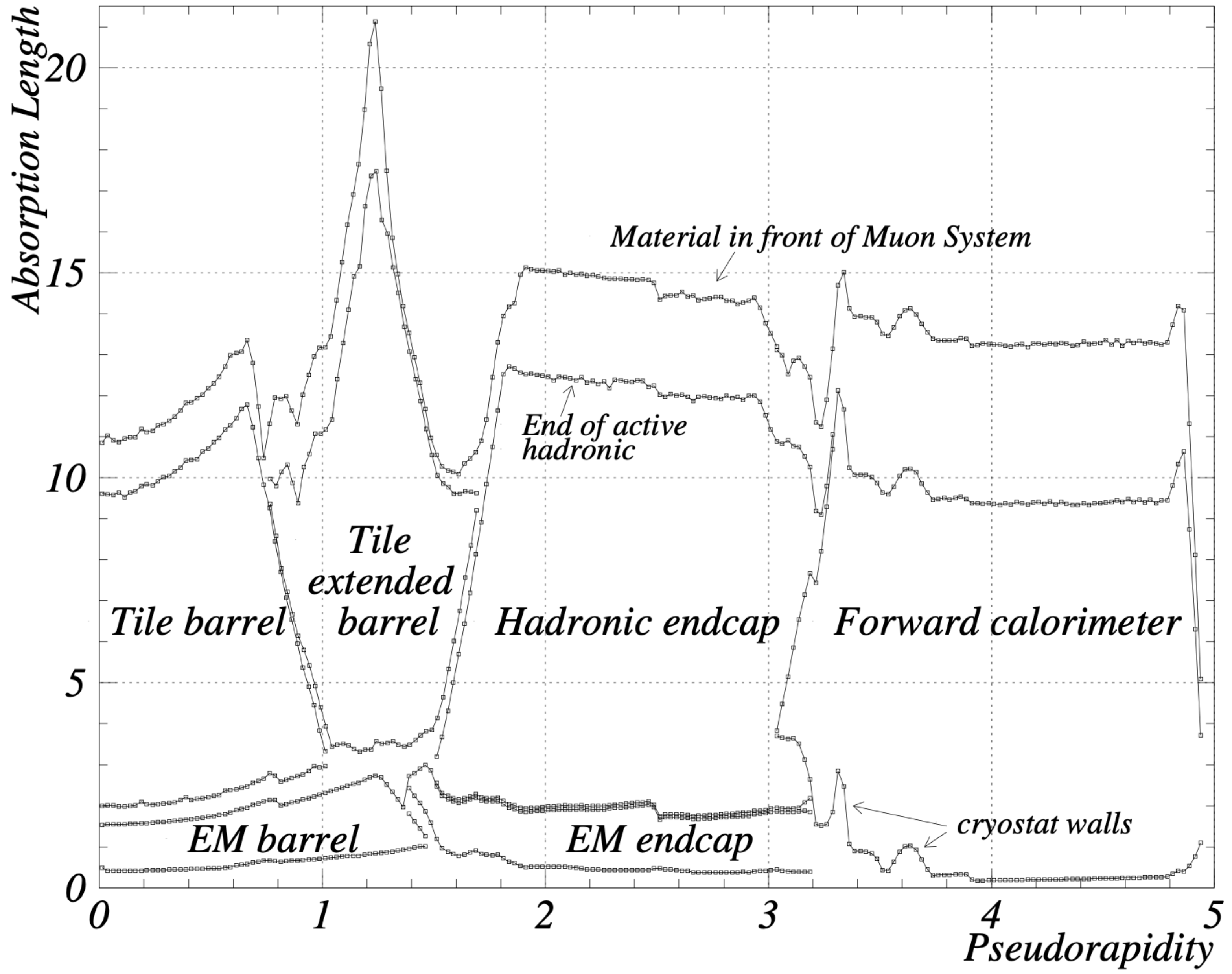
- Expanded coverage from Run 2 to Run 3

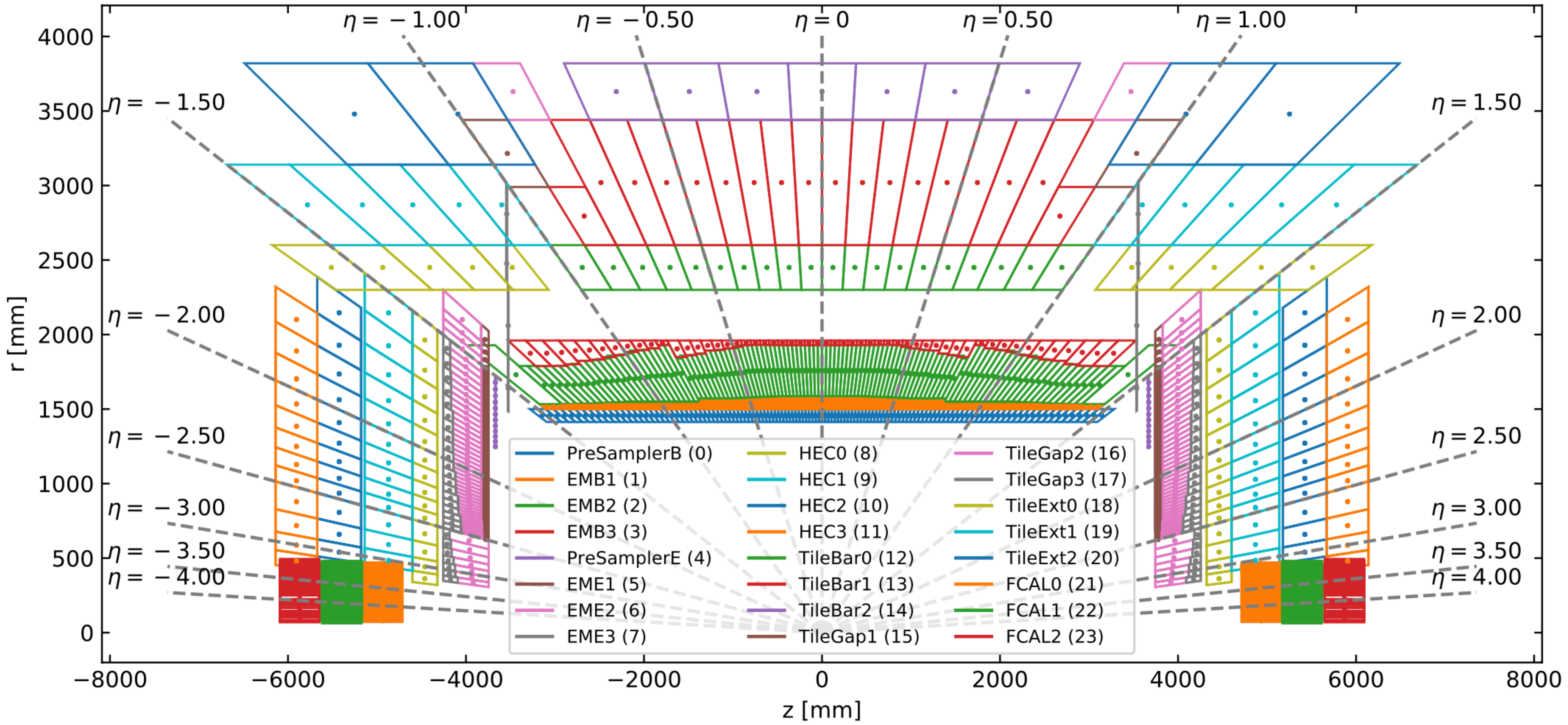
- From prototype to production

Challenges in fast simulation

- Complex geometry and non-uniform material distribution
- Many small effects to consider: energy correction, ϕ modulation correction, etc

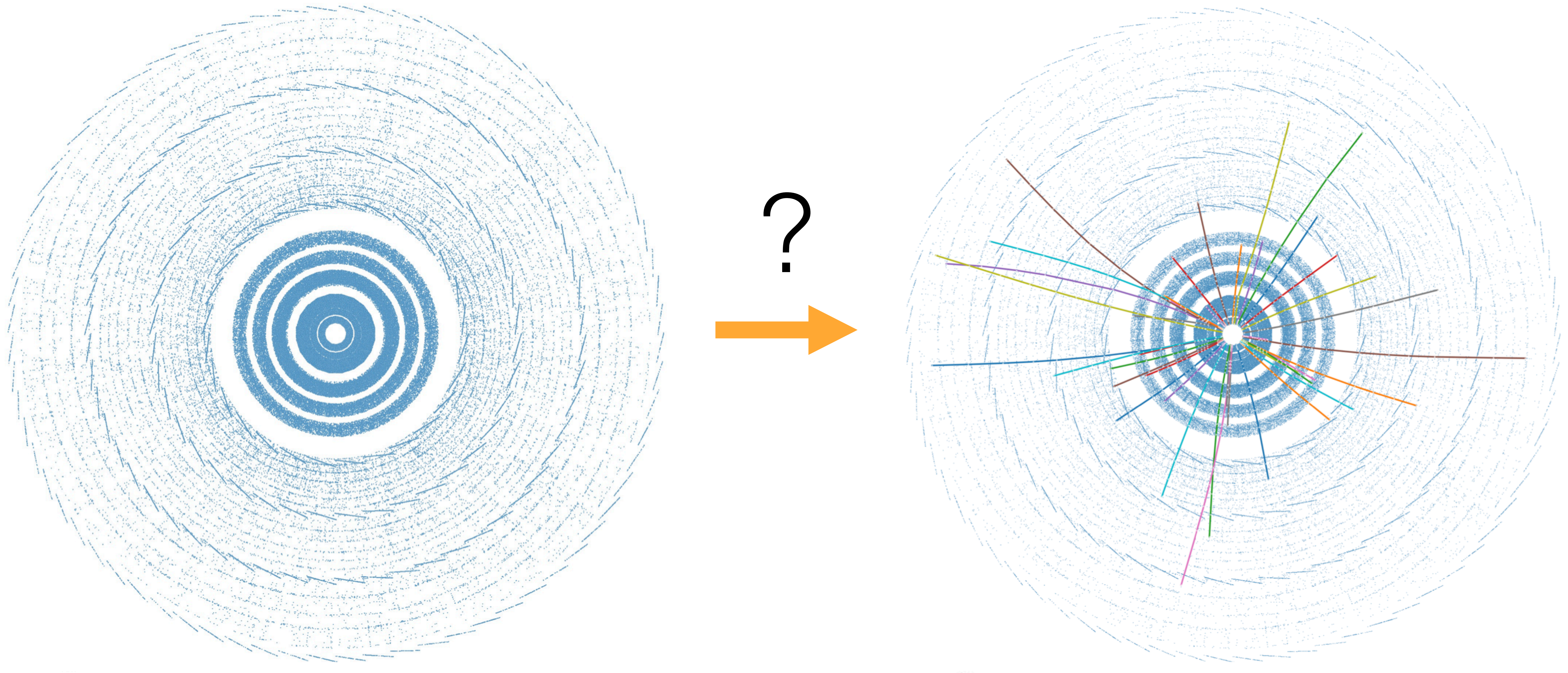




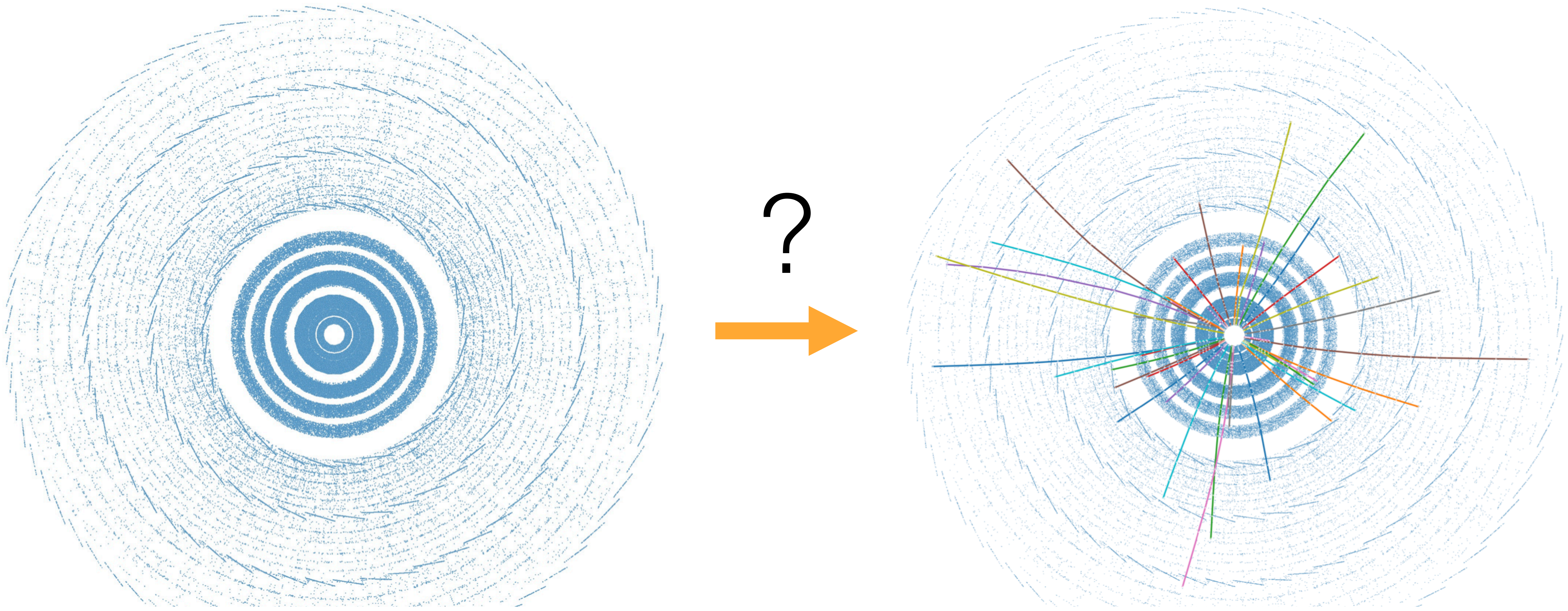


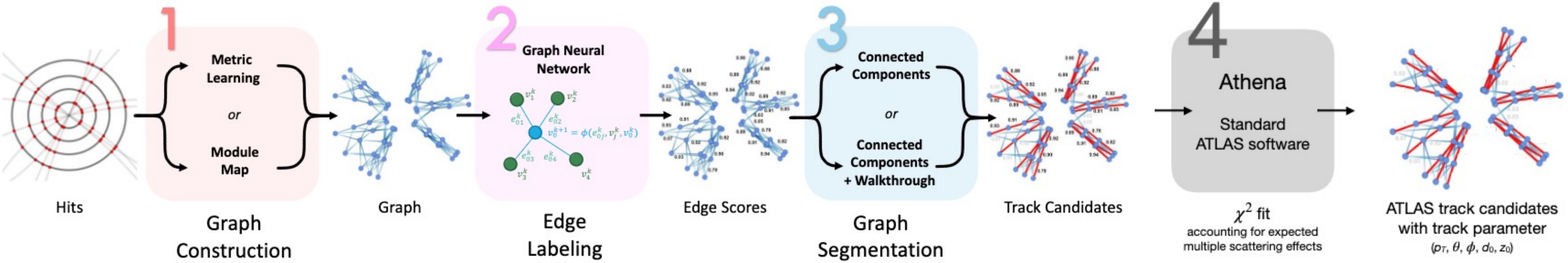
- Also exploring other types of model besides GAN

GNN for tracking



GNN for tracking





More details can be found at source

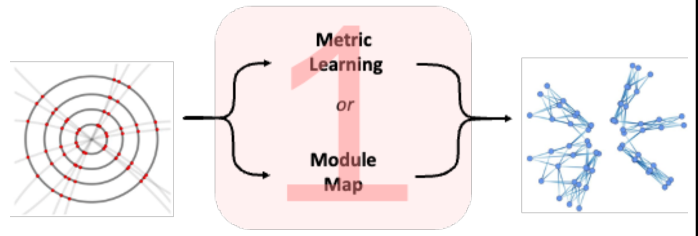
Contents

- Motivation to use machine learning (ML)
- How to map a HEP problem to a ML problem
- Examples of recent ML applications
- Summary

ML ❤️ HEP

- HEP is a data science
  - Large data volume, high dimensions
  - ML has been a longstanding companion in HEP in various stages of the data analysis pipeline
- ML technique advanced in recent years
  - Architectures become complex and mature, thanks to large training data and powerful computing ability
  - More new architectures on the way
- Challenges and opportunities
  - Data representation and architecture design should take into account underlying physics
  - Find ways to increase the size of training sample
  - Using low level features becomes crucial - advantage in precision electron-positron colliders

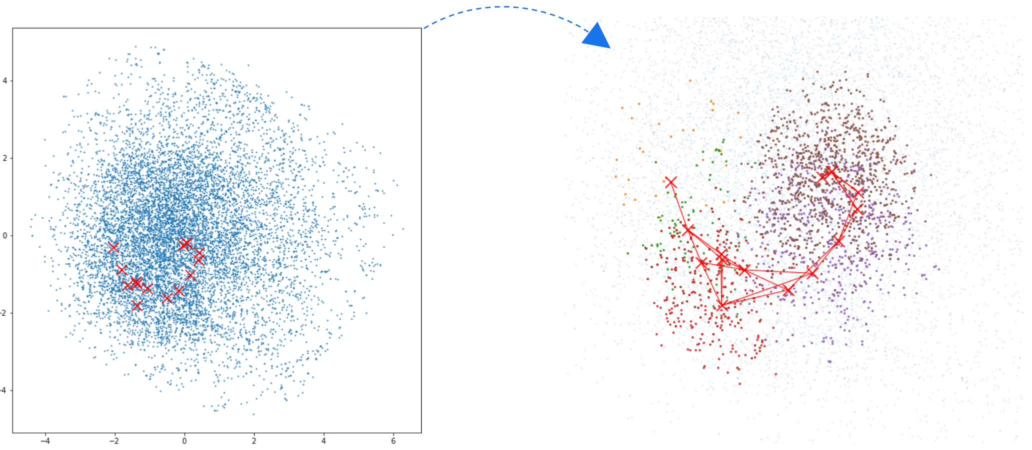
Backup



Graph construction

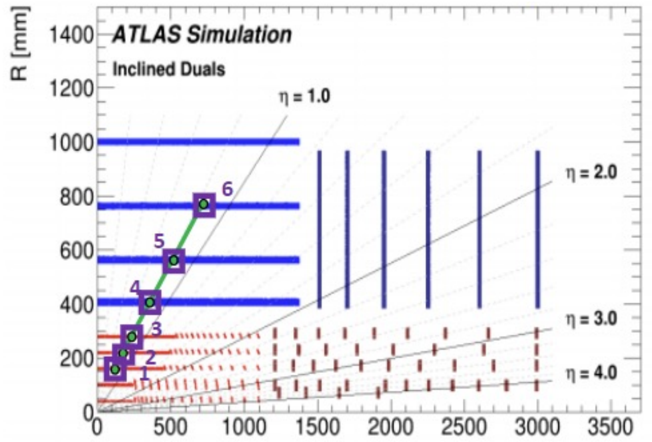
Machine learning approach: Metric Learning

- Train a DNN to project hits to an embedding space, such that Hits from the same particles are near each other by L_2-distance. Constructs graphs using kNN.
- Clean up easy fake edges by a DNN or a shallow GNN to reduce graph size and fit on GPU.

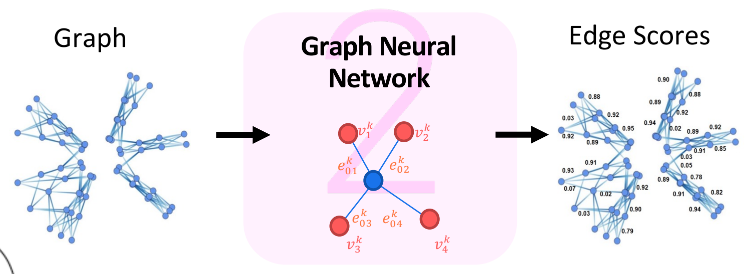


Data driven approach: Module Map

- Build a map of detector modules, where a triplet of hits ABC means at least 1 true track has passed sequentially through A, B, and C.
- Register a triplet ABC if all 3 modules get hit in the event.

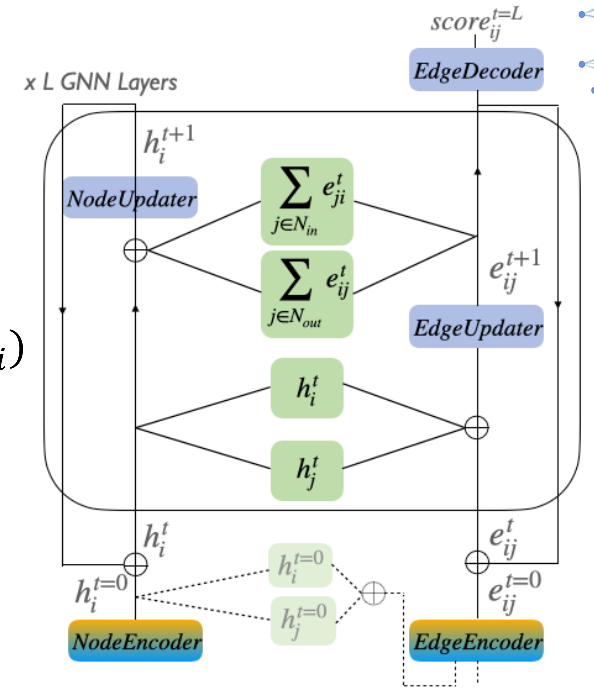


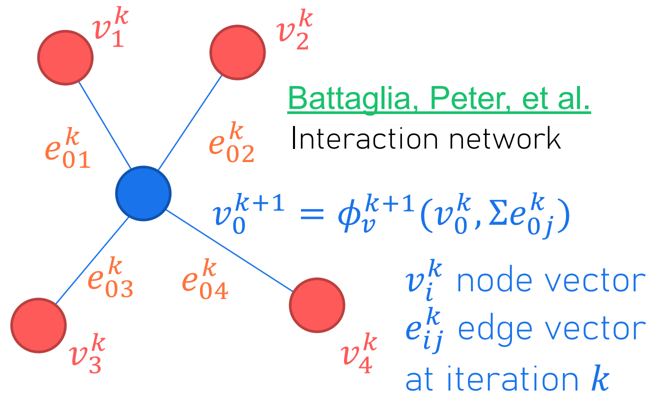
Connections added:
1 -> 2 -> 3
2 -> 3 -> 4
3 -> 4 -> 5
4 -> 5 -> 6

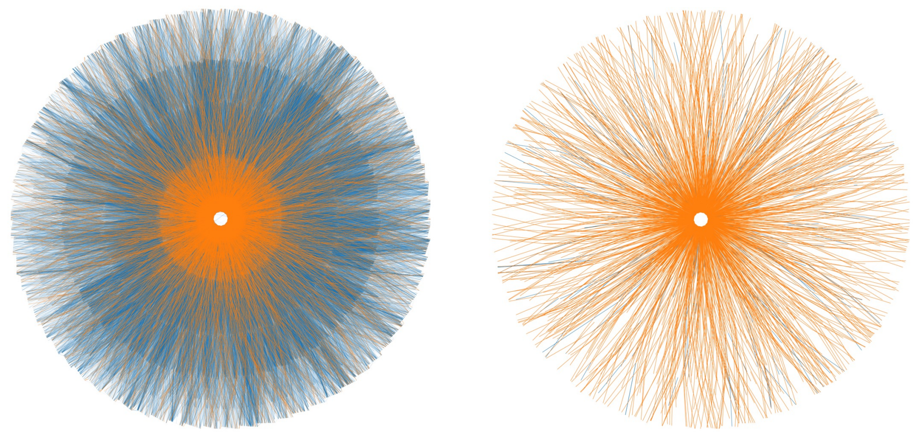


GNN edge classification

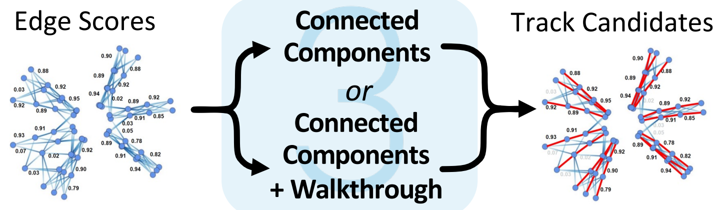
1. Encode nodes features (position, charge count, local measurements, etc.) to a latent node vector $v_i^0 = \phi_v(x_i)$
2. Concatenate node vectors of two hits connected by an edge and encode to edge vector, $e_{ij}^0 = \phi_e(v_i^0, v_j^0)$
3. Aggregate edge vectors, acting as messages between nodes, $m_i^0 = \sum_j e_{ij}^0$
4. Update node features using aggregated message, $v_i^1 = \psi_v^1(v_i^0, m_i^0)$. Update edge features using updated node features, $e_{ij}^1 = \psi_e^1(v_i^1, v_j^1, e_{ij}^0)$.
5. Repeat steps 3 and 4 $\boldsymbol{n} = \mathbf{8}$ times.
6. Compute an edge score representing the probability of being a true edge, $s_{ij} = \psi_d(e_{ij}^n)$





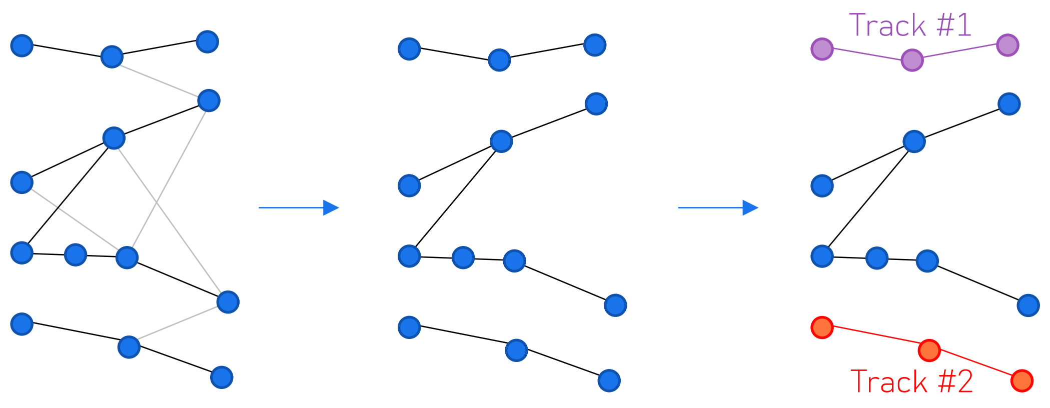


Input graph (left) and classified graph (right). Fake = blue. True = orange



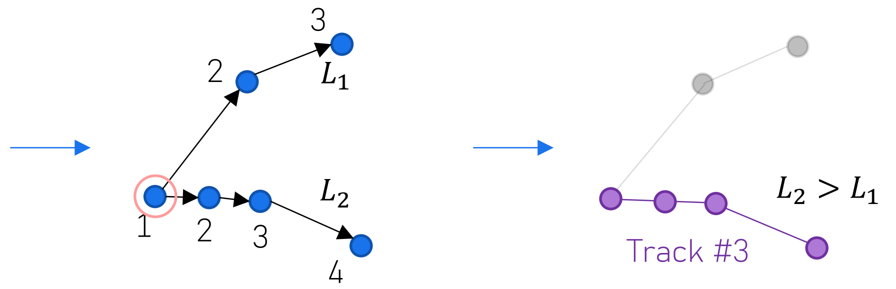
Track construction

1. Connected Components

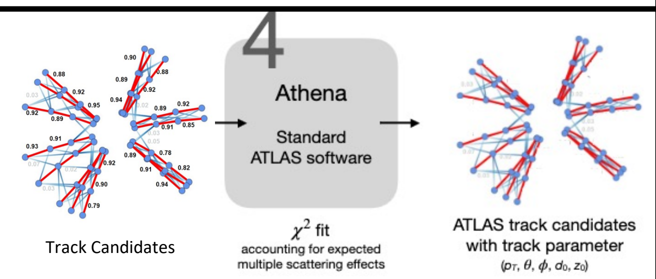


Classified edges — Loose score cut — Label simple candidates

2. Walkthrough, a.k.a "Wrangler"



Walk through paths from starting node, count length L — Assign longest path as candidate



Physics performance of the GNN4ITk pipeline

- Perform a global χ^2 fit on GNN track candidates. Evaluate the performance and compare to that of tracks found by the CKF.
- GNN tracks are selected using ATLAS requirements, with some selection cuts loosen.

Requirements	Pseudorapidity interval		
	$\lvert\eta\rvert < 2.0$	$2.0 < \lvert\eta\rvert < 2.6$	$2.6 < \lvert\eta\rvert < 4.0$
pixel + strip hits	≥ 9	≥ 8	≥ 7
pixel hits	≥ 1	≥ 1	≥ 1
holes	≤ 2	≤ 2	≤ 2
p_T [MeV]	> 1000		
$\lvert d_0\rvert$ [mm]	≤ 2.0	≤ 2.0	≤ 10.0
$\lvert z_0\rvert$ [cm]	≤ 20.0	≤ 20.0	≤ 20.0

For GNN4ITk 3 cuts are looser:
pixel + strip hits ≥ 8,
$|d_0| < 20$ mm
$|z_0| < 25$ cm

ATL-PHYS-PUB-2021-024

source