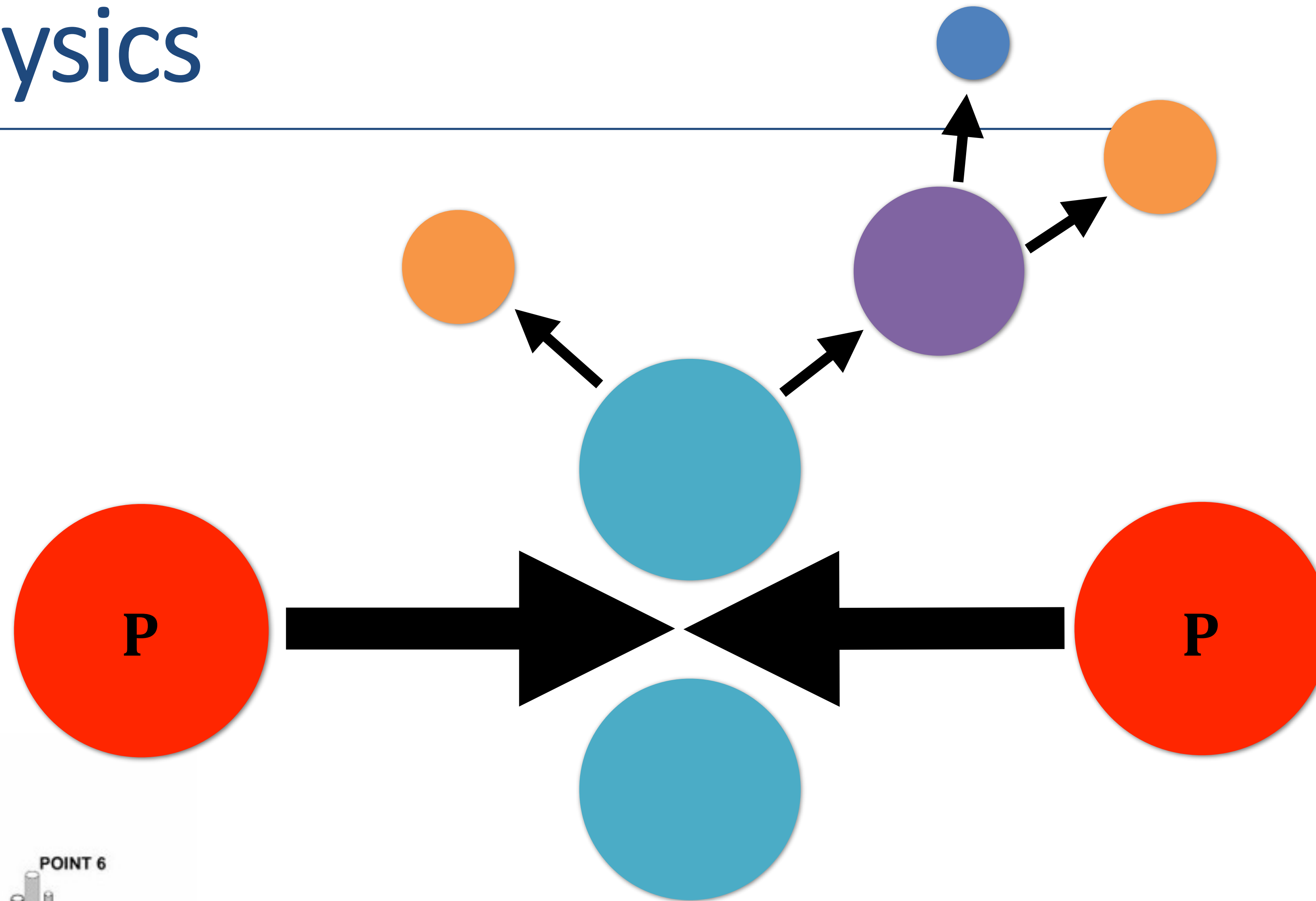
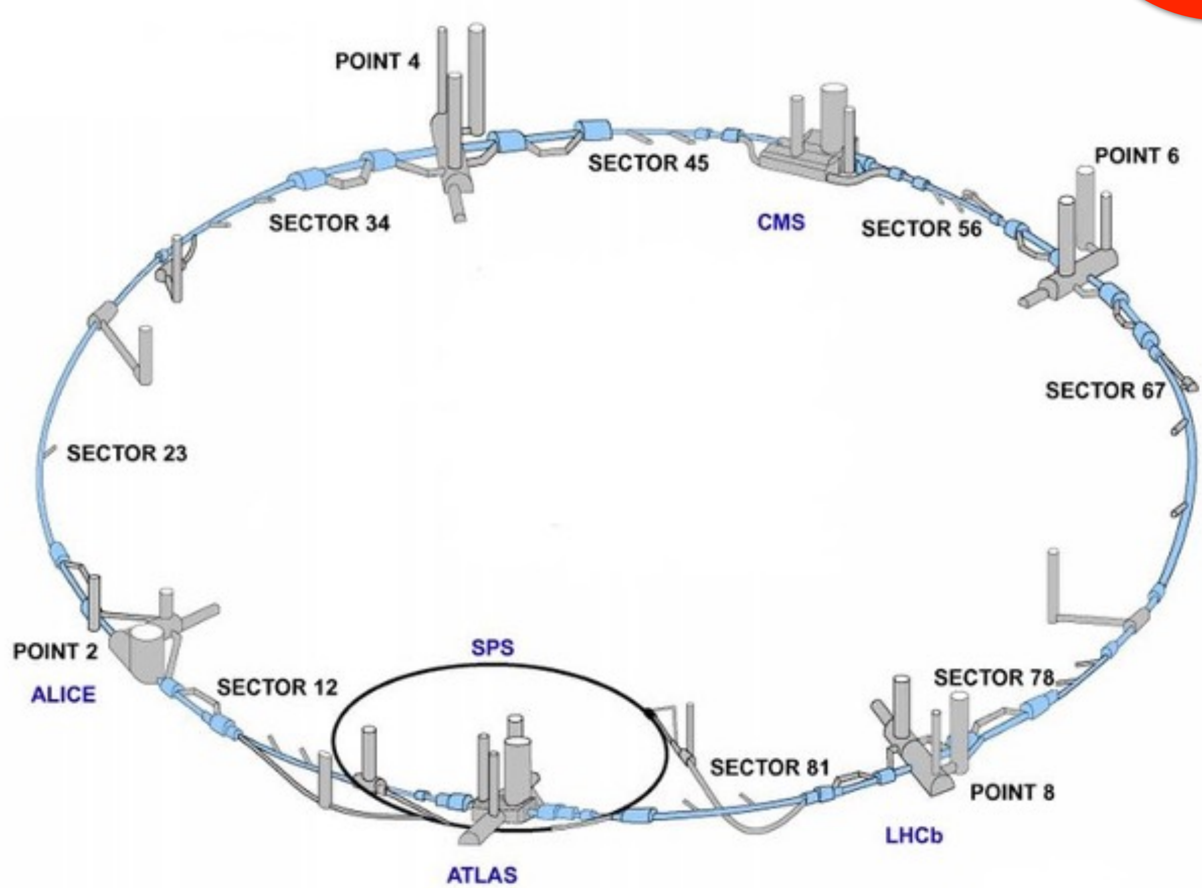
Deep Learning at the LHC: from Data to Analysis

Corentin Allaire

LHC: Collider physics

- Proton-Proton collision: produces **new particles**
- Most **decay** before reaching the detectors
- Need complex reconstruction algorithms to reconstruct the **original particles**

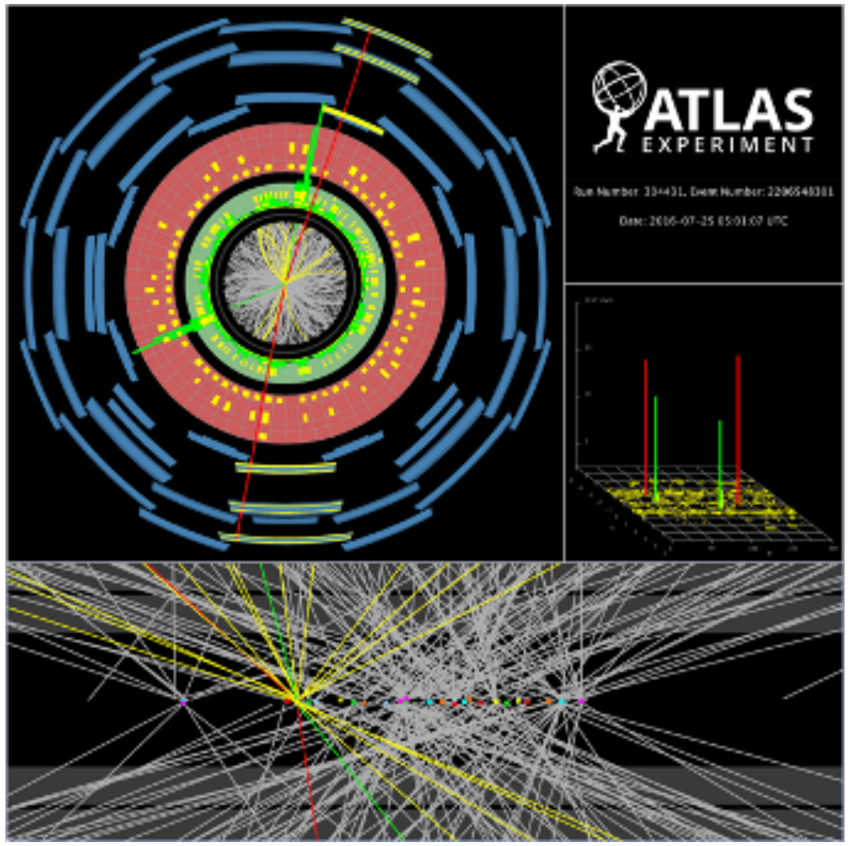


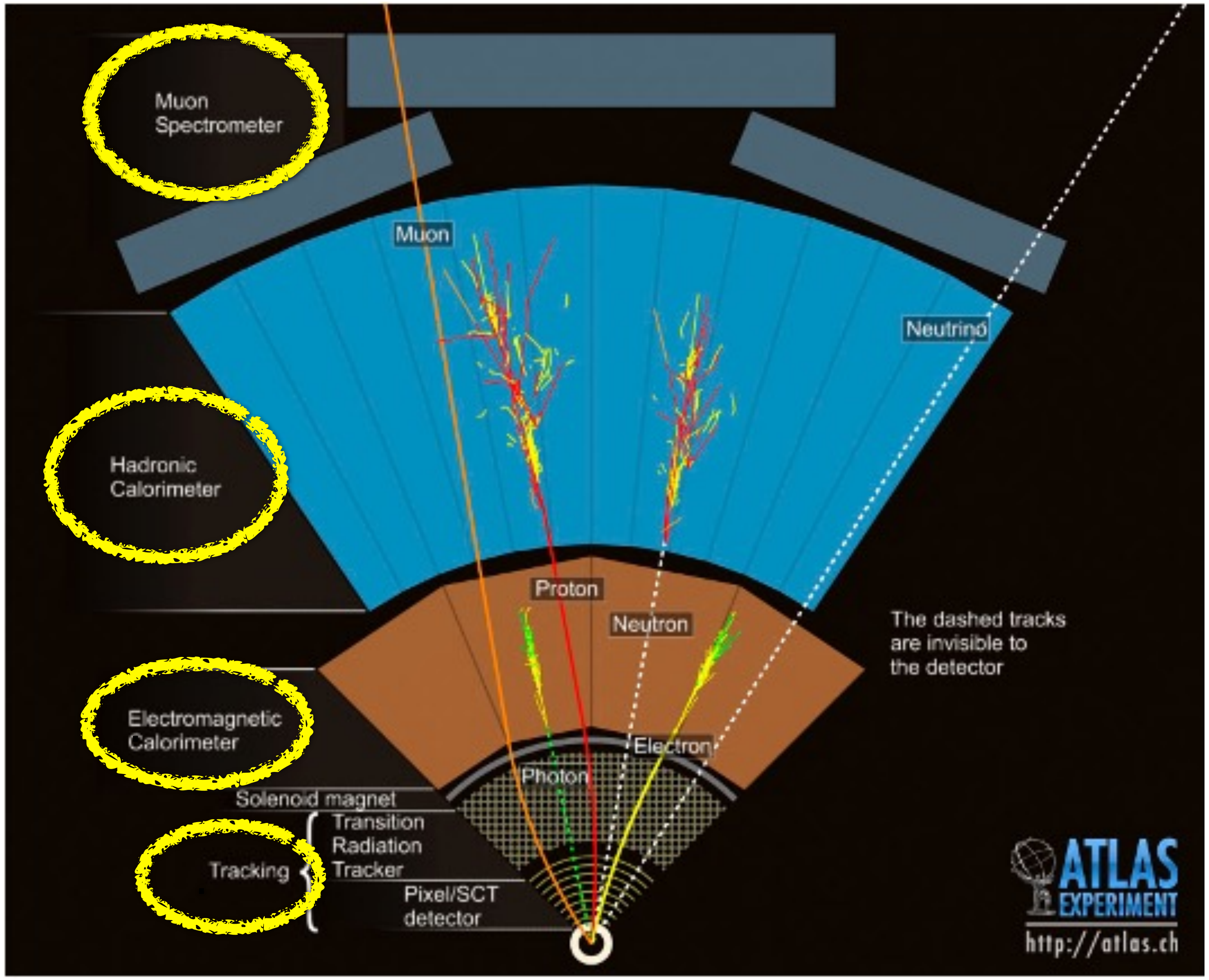


- **LHC:**
  - pp collision $\sqrt{s} = 14$ TeV
  - Collision every 25 ns (40 MHz)
  - **Multiple Petabytes** of data per experiment per year
- This presentation mostly focuses on ATLAS and CMS

Object reconstruction @ the LHC

- **Typical detector:**
  - ➡ Tracker:
    charged particle trajectories
  - ➡ Calorimeter (em & had):
    Energy of the particles (jets)
  - ➡ Muons spectrometer:
    Detect the muons (cross the entire detector)



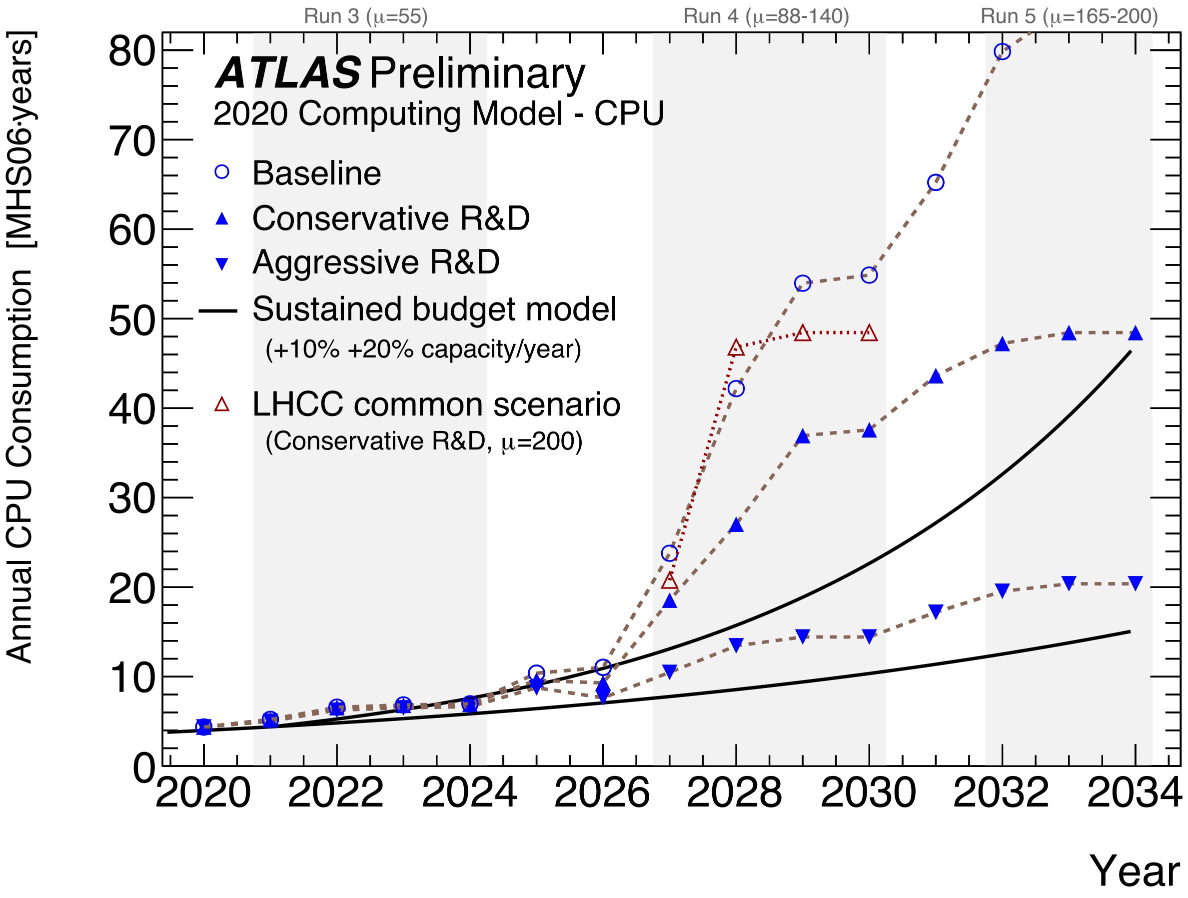


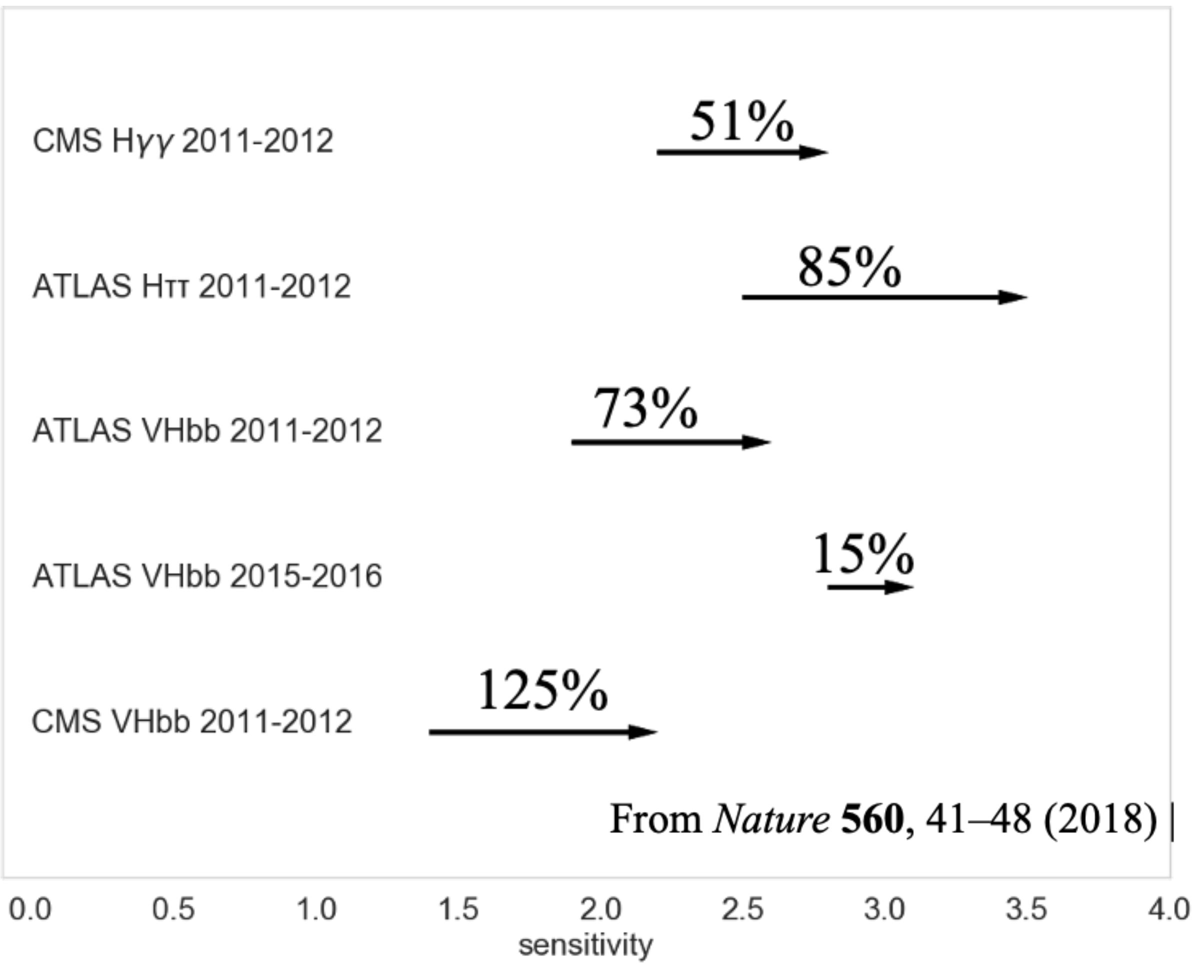
- **Pileup:**
  - ➡ Many interactions per crossing (~50 now; 200 in the future)
  - ➡ Complex algorithm needed for reconstruction
  - ➡ High CPU cost

Why use deep learning ?

Tremendous amount of data at the LHC:

- Huge amount of computing power needed to reconstruct the data
- Even more needed to simulate events for analysis





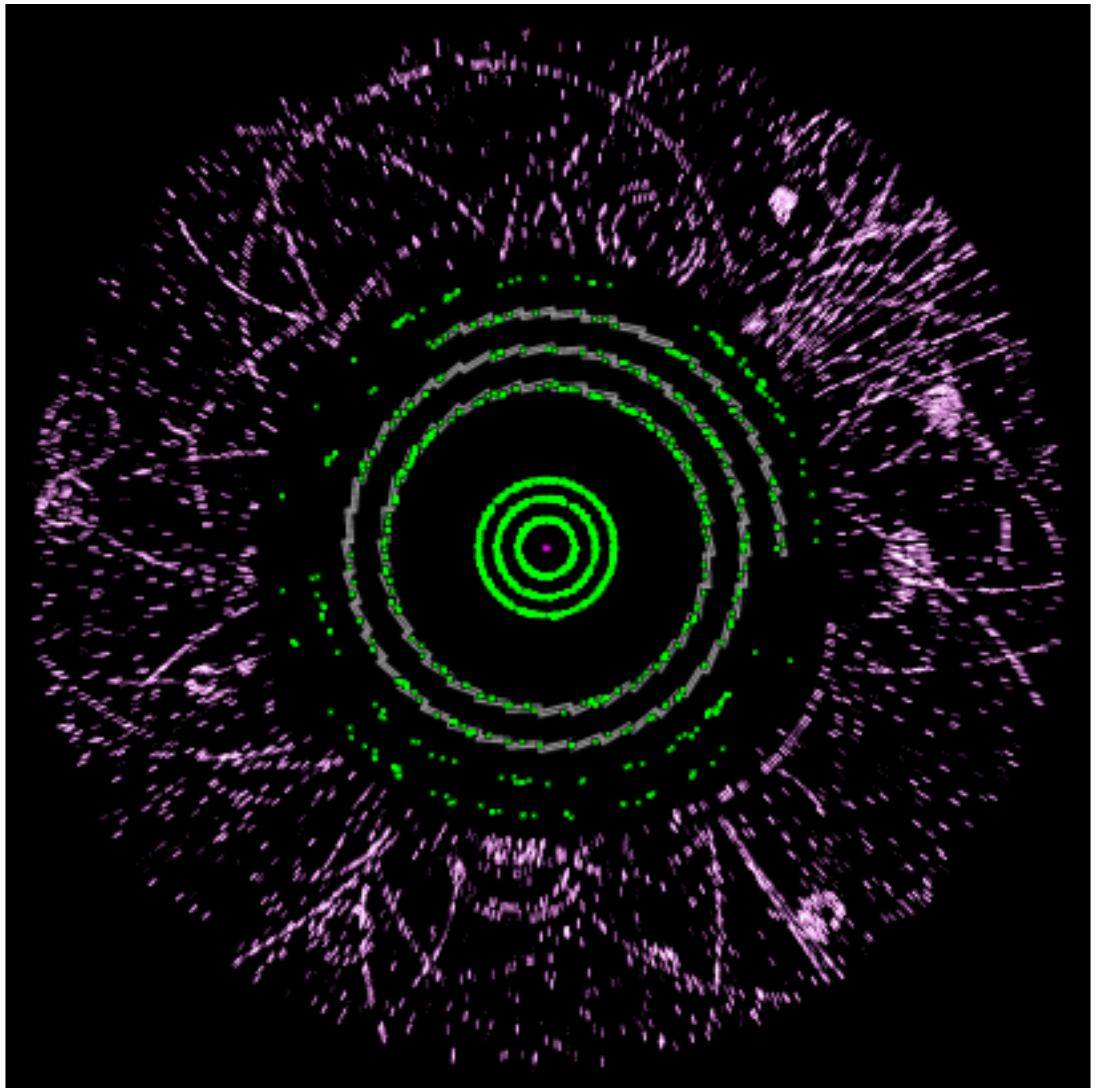
Impact on the analysis (Higgs boson) at the LHC:

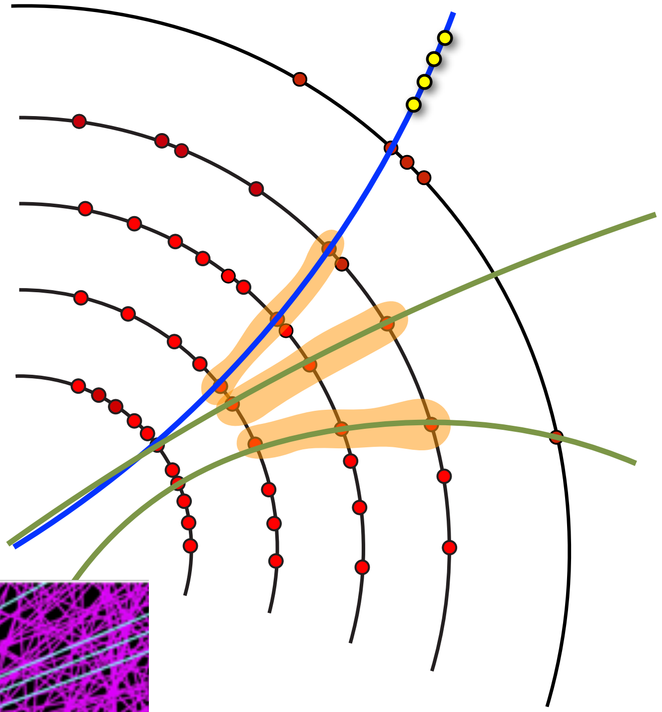
- Usually ~ 10 variables **BDT** (ML)
- Equivalent to collecting ~50% more data (~ +0.5 billion CHF per year)
- Maximise our use of the LHC

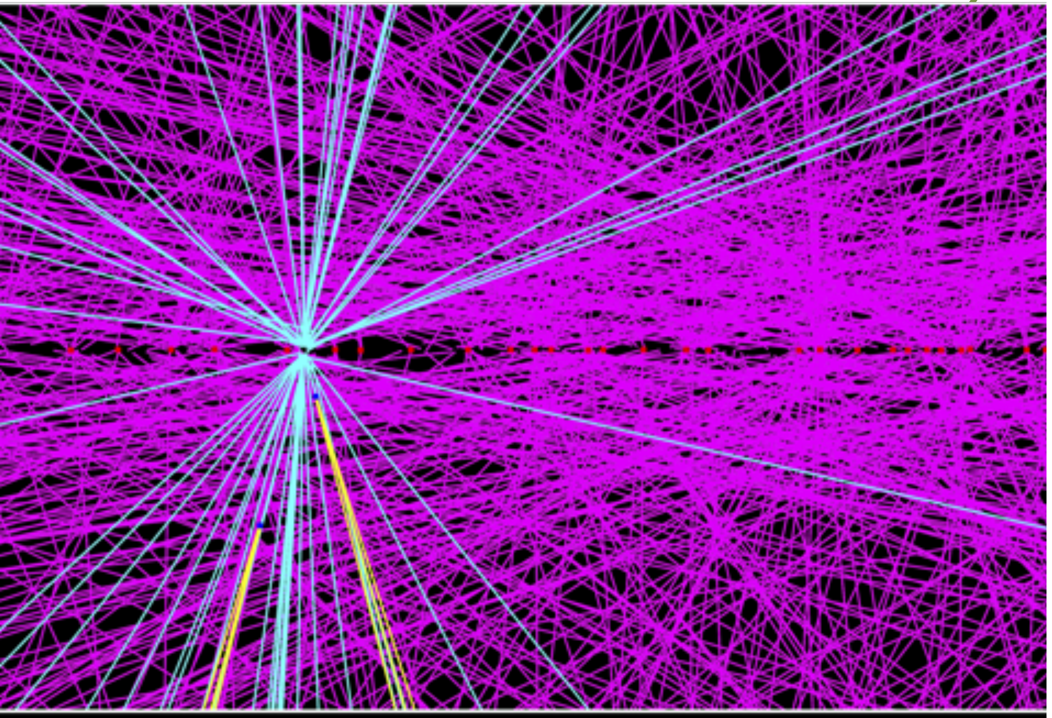
Particles Trajectory reconstruction

Charged particle tracking

- Connect together hits coming from the same particles
- Extremely high combinatorics
- Tracking involves complex algorithms: **Kalman Filtering**



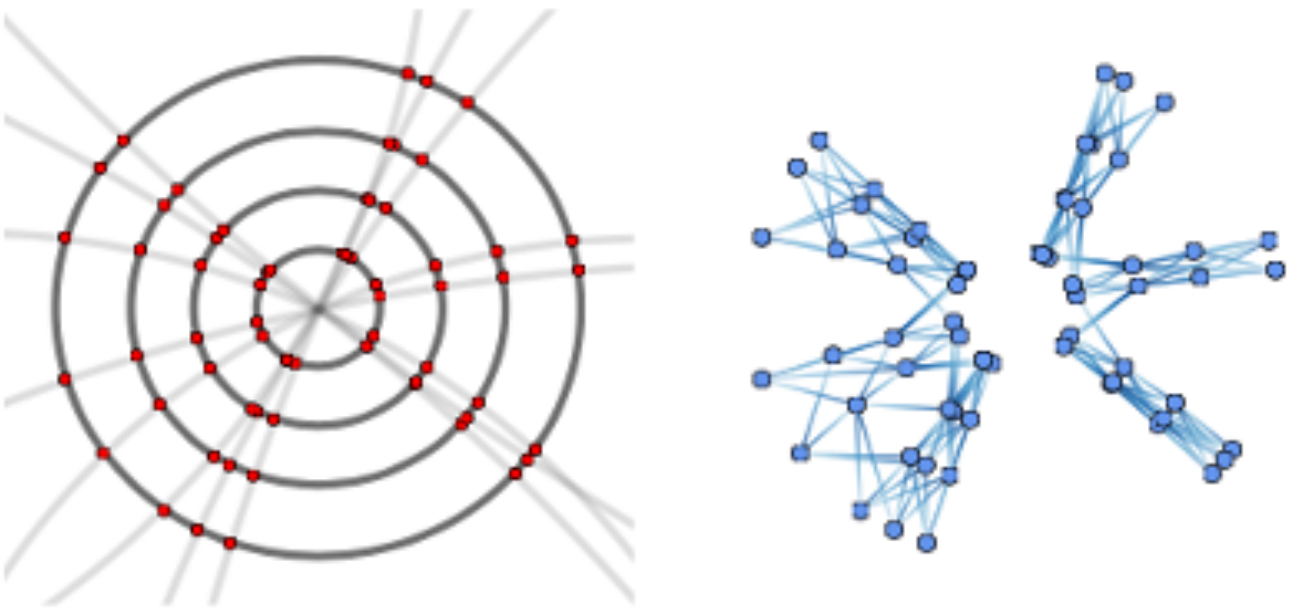


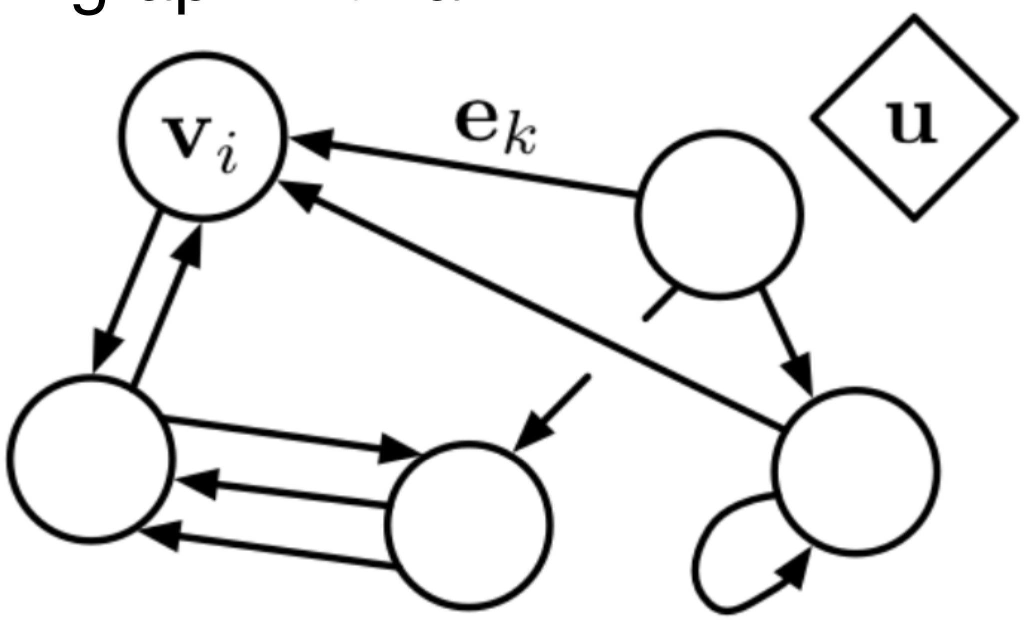


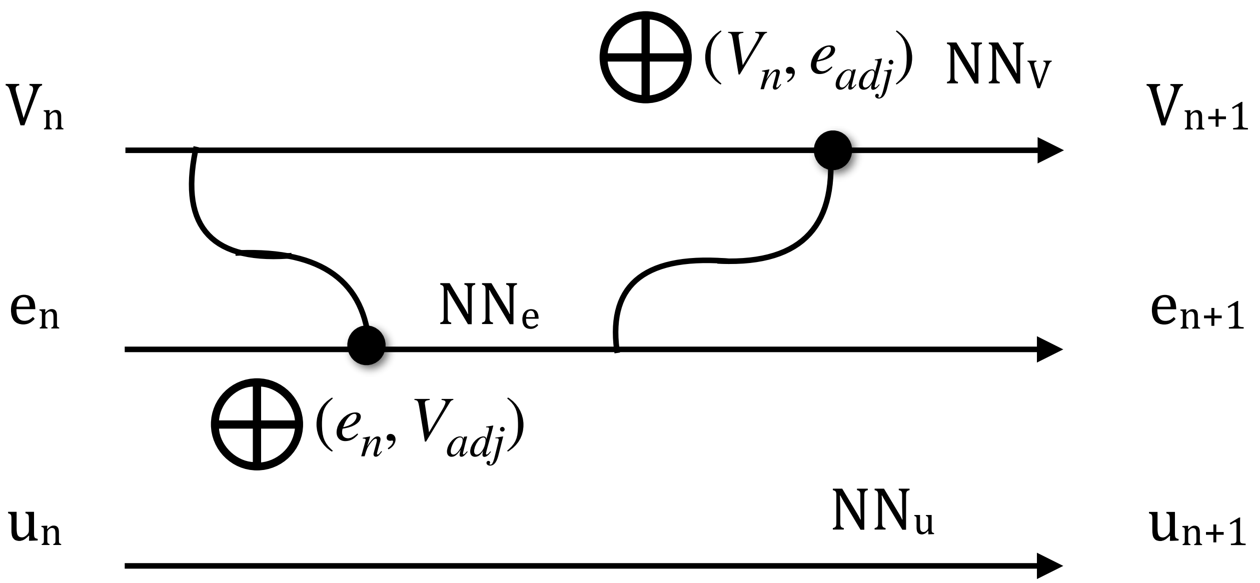
- Intensive in computing resources (dominate the reconstruction)
- Try to maintain good performances in future high combinatorics conditions
- Can Deep learning help us achieve our goals?

Sparse data : Graph Neural Network

- HEP Data ➔ Too sparse for image processing techniques
- Easy to represent as graphs
  ➔ Graph Neural Network
- Graph:
  - Nodes v_i
  - Connected via Edges e_k
  - With global variables u
- Propagate information through the graph with a NN

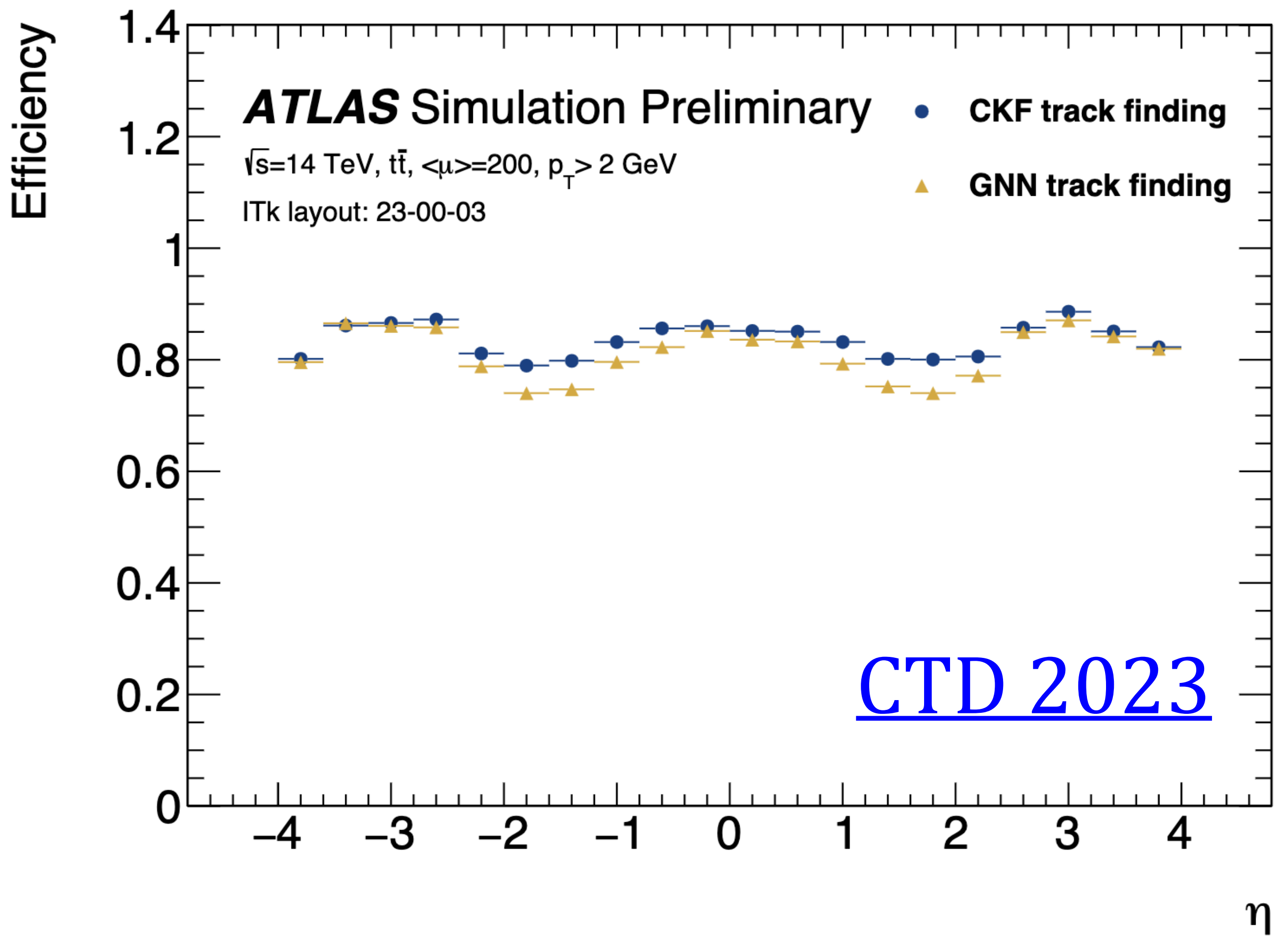


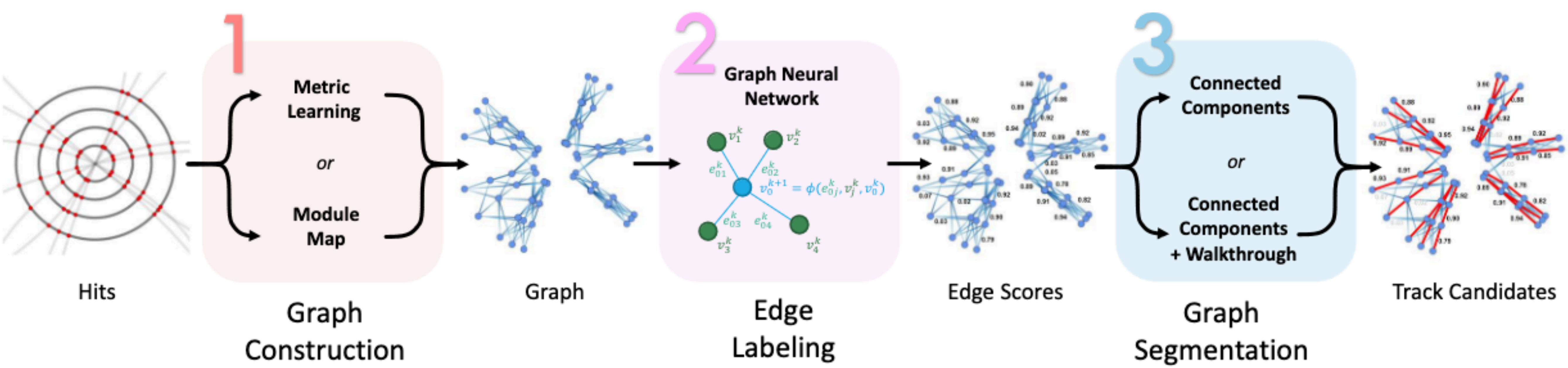




GNN Tracking : GNN4ITk

- Applied to charged particles tracking with the future ATLAS tracker (ITk)
- Treat all hits as nodes
- Try to classify the edges
  ➔ good edges = track path
- Competitive physics results
- Complexe graph construction step



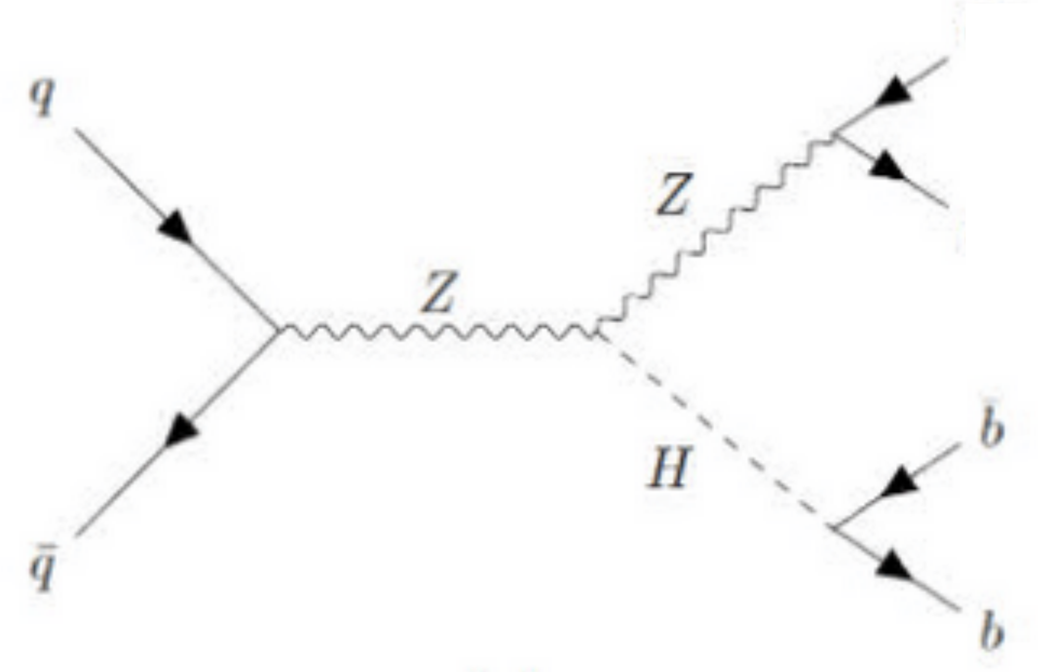


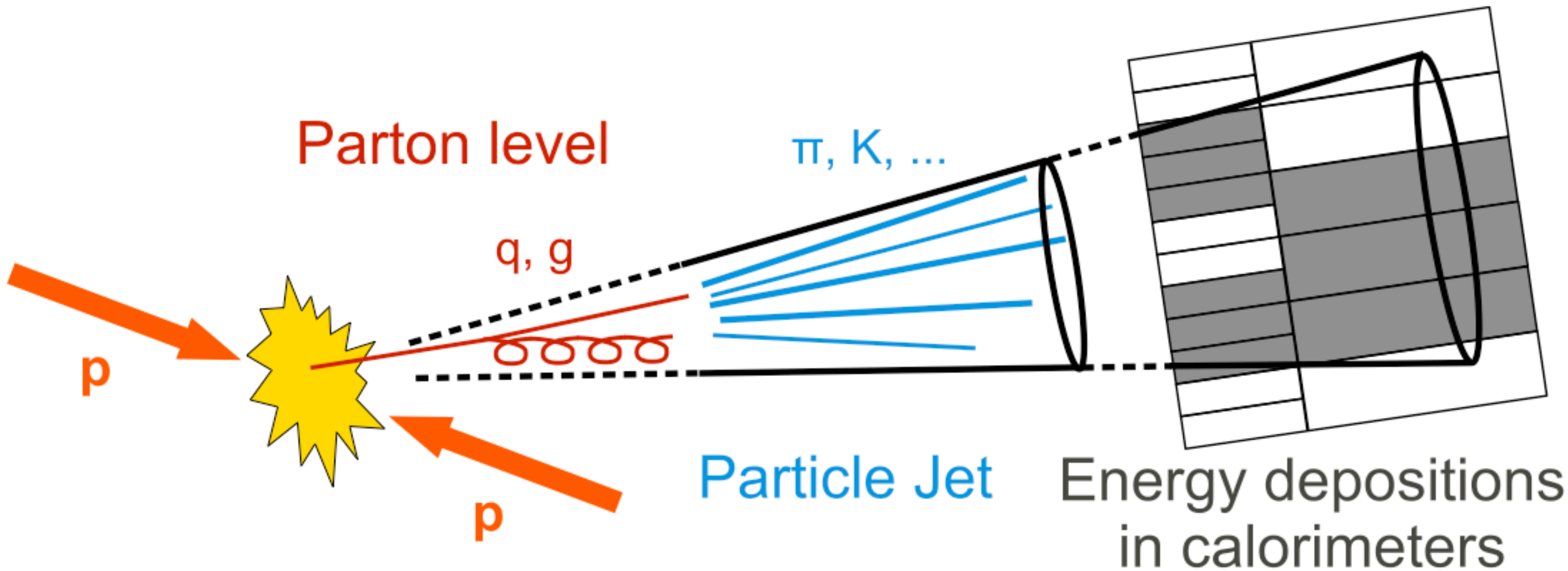
Physics Performance of the ATLAS GNN4ITk Track Reconstruction Chain

Particles Identification

Jet tagging

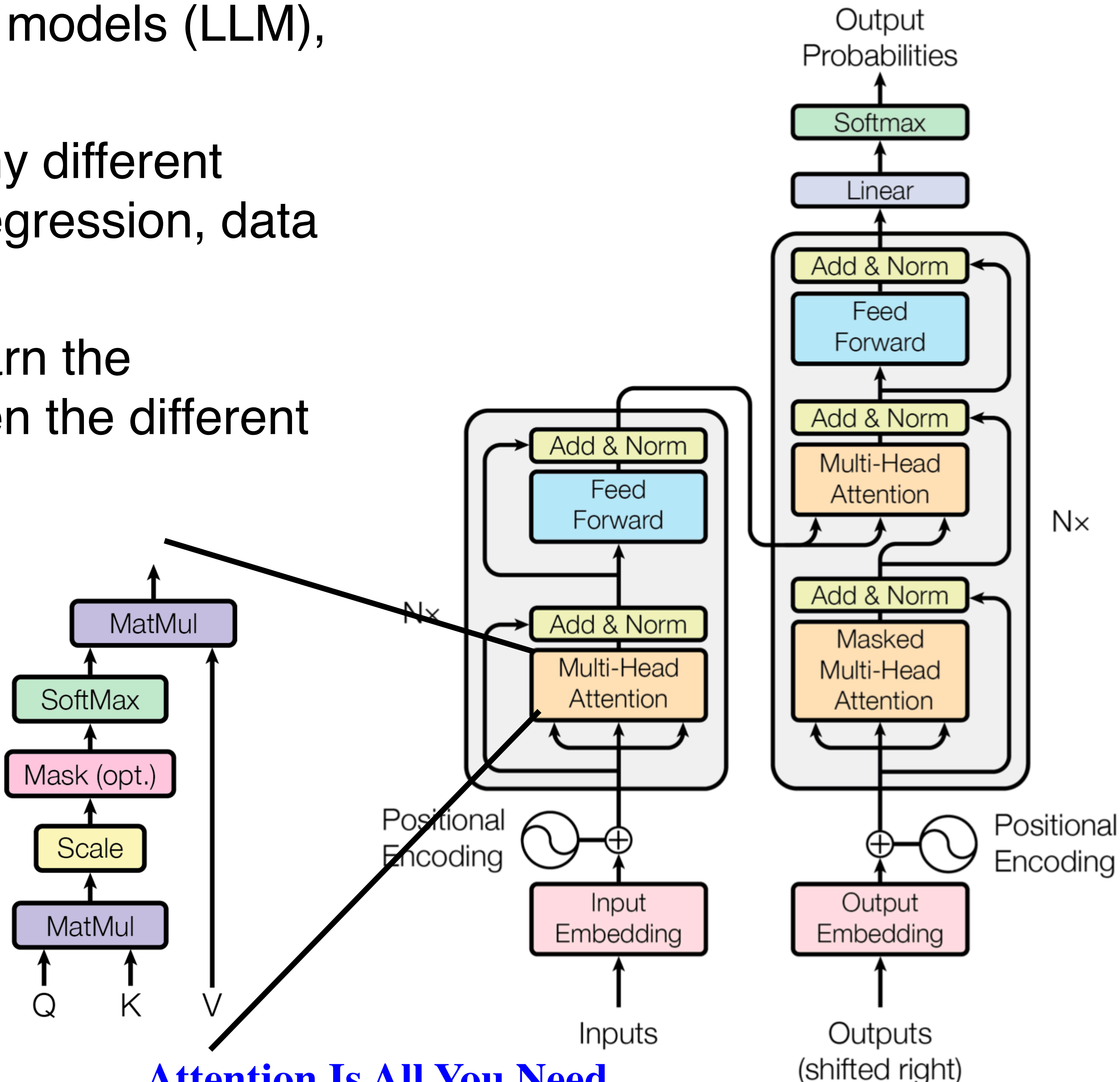
- Jets:
  - Collimated **spray** of particles
  - Originate from a single **quark** or **gluon**
  - Reconstructed via Calorimeter+Tracker
- Identifying if a jet comes from a **b** quark, **c** quark, **light** quark, or a **gluon** is extremely important for various analysis





Transformers

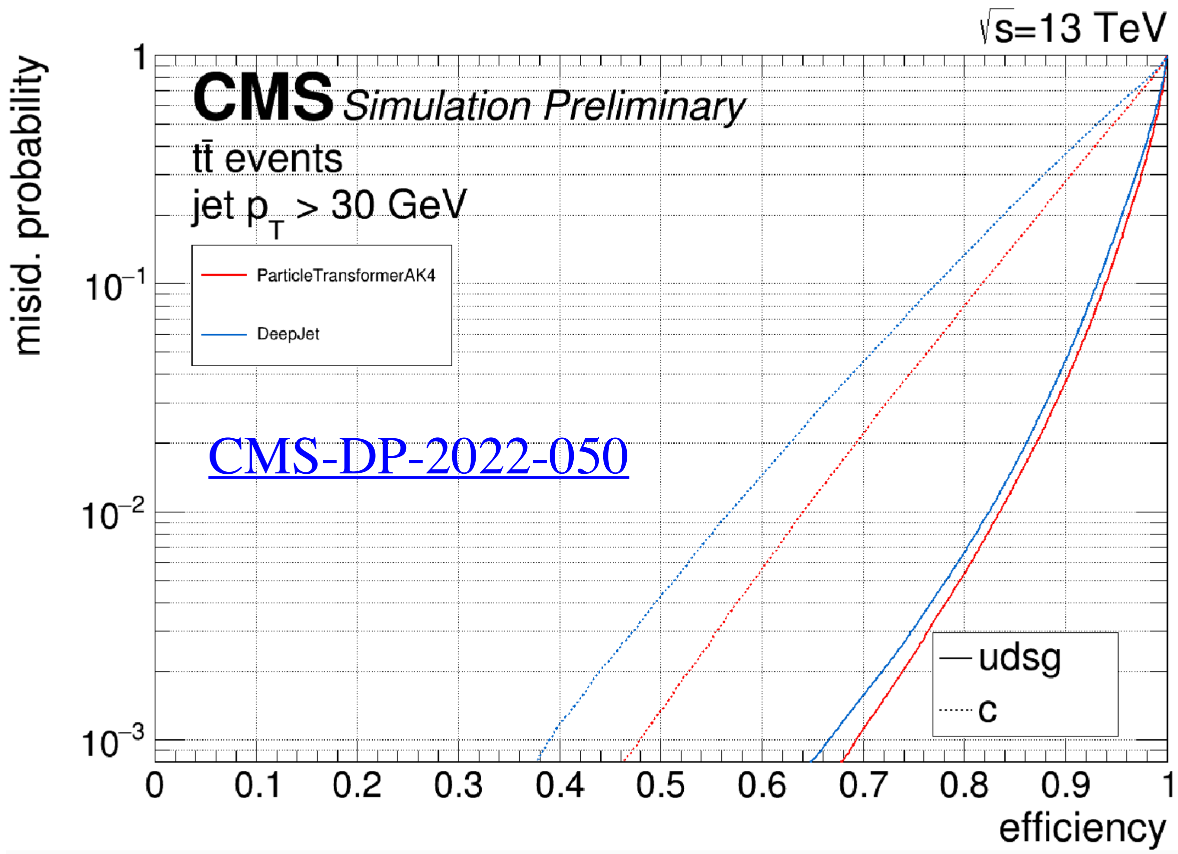
- Used in most large language models (LLM), i.e., chatGPT
- Great success: Used for many different applications: classification, regression, data generation
- **Attention mechanisms**: Learn the correlations that exist between the different inputs



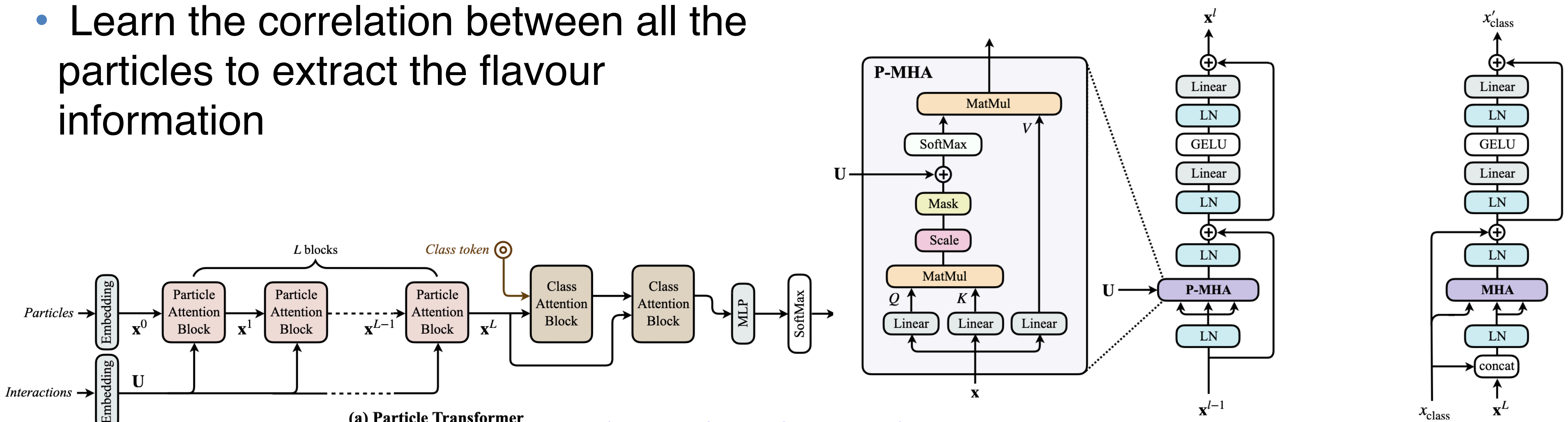
[Attention Is All You Need]

Flavour tagging with transformers

- Applied transformer-based technique to jet flavour reconstruction
- Tested in the CMS experiment
  ➔ better than previous DL approaches
- Inputs:
  - Information on the particles in the jets (up to 100)
  - « Interactions »: variable related to 2 particles
- Learn the correlation between all the particles to extract the flavour information



CMS-DP-2022-050



(a) Particle Transformer

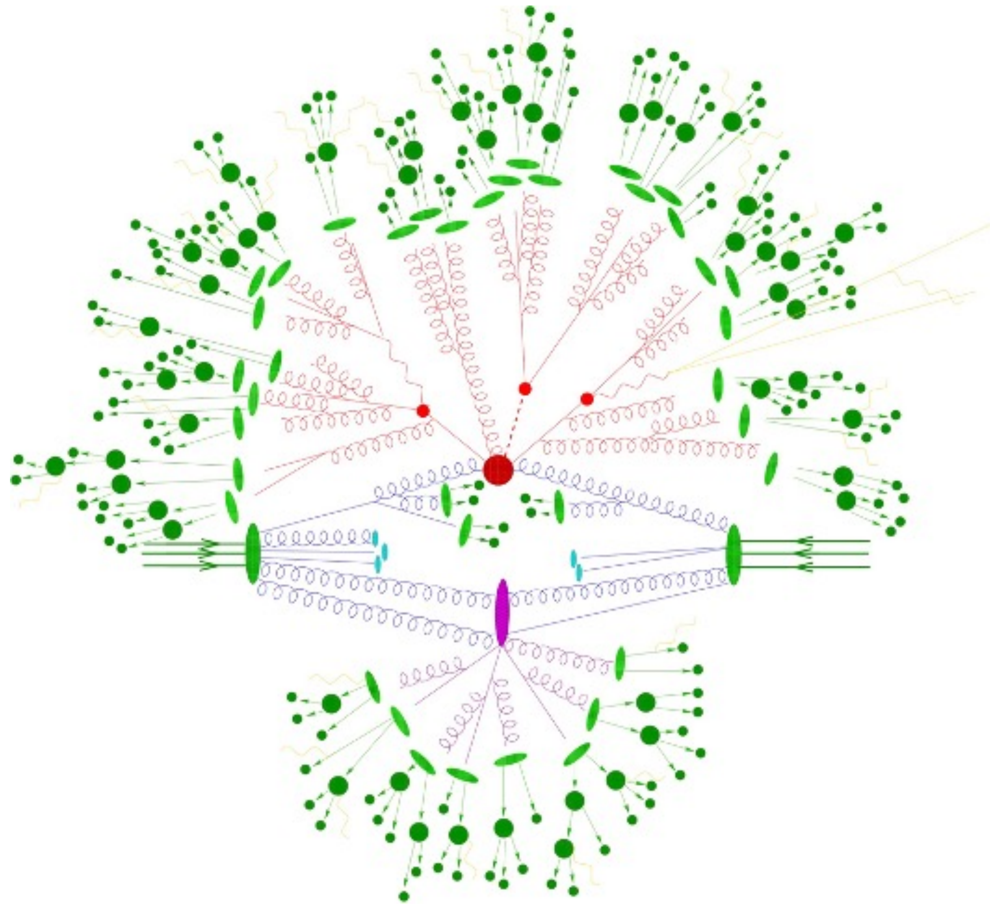
Particle Transformer for Jet Tagging

(b) Particle Attention Block

(c) Class Attention Block

Events/Detectors Simulations

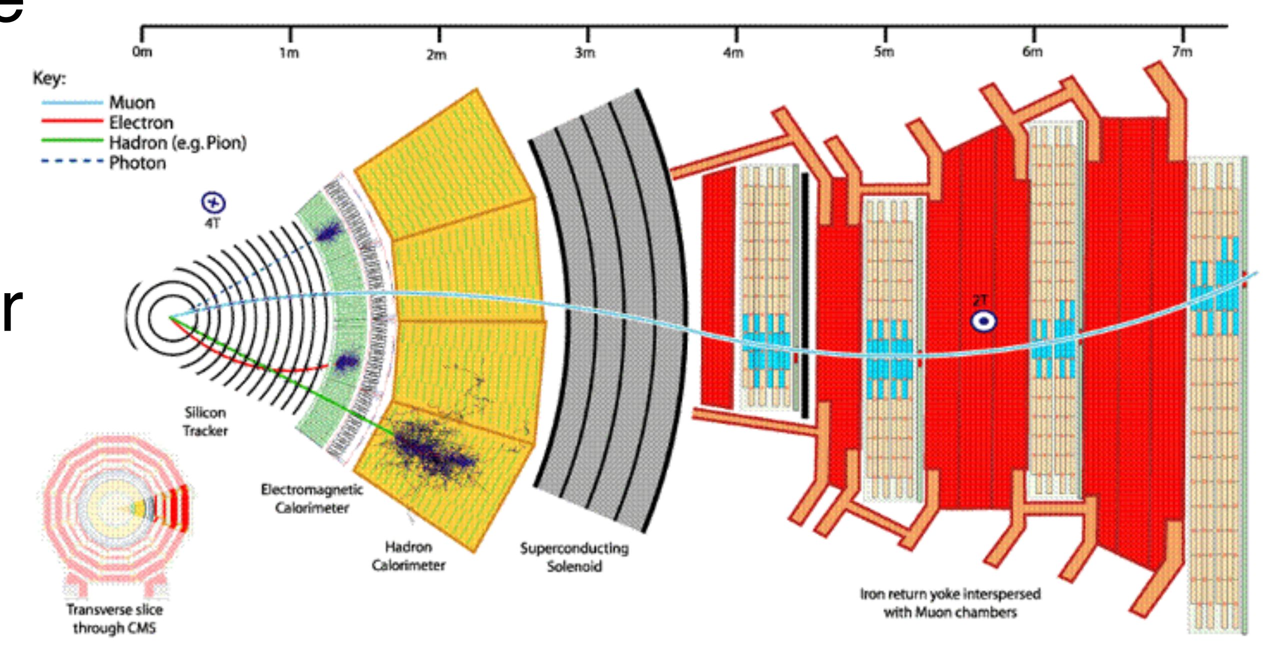
Event Simulation



Simulators:

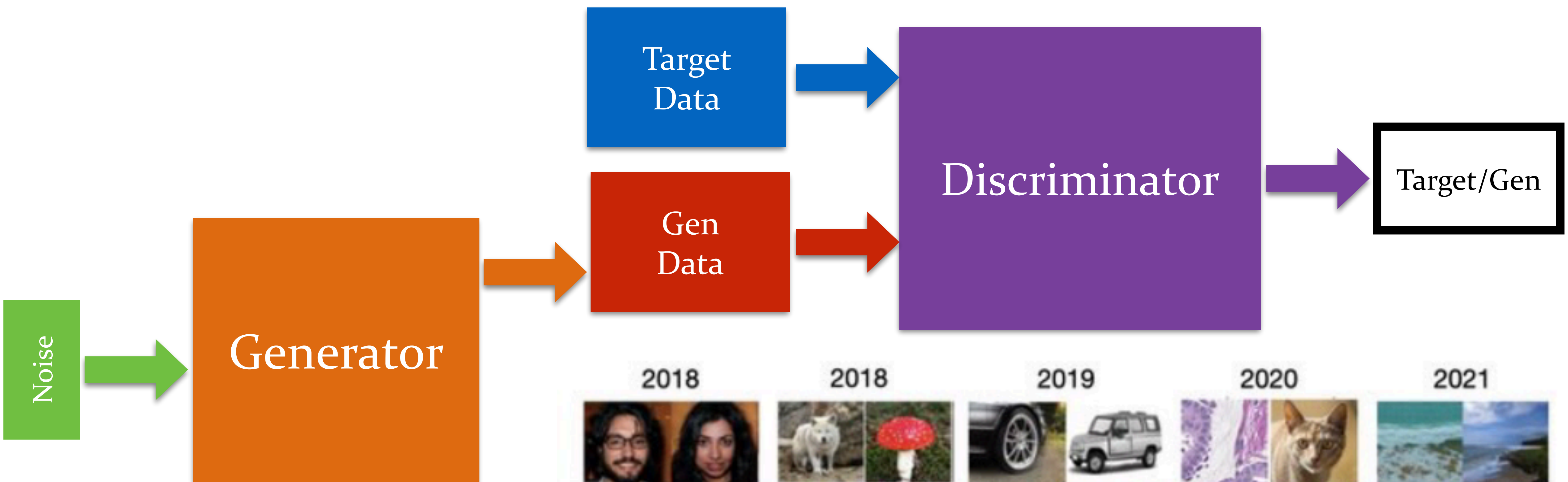
- Combine a precise simulation of the physics process with a proper accounting for the particle-matter interaction (Geant4)
- Result in extremely realistic detector signatures
- With available ground truth

- Standard in the HEP community since the seventies
- The basis for most physics analysis
- Requires a large amount of person-power
- Biggest CPU resources consumer for most LHC experiments
- Can we still use them in the future?

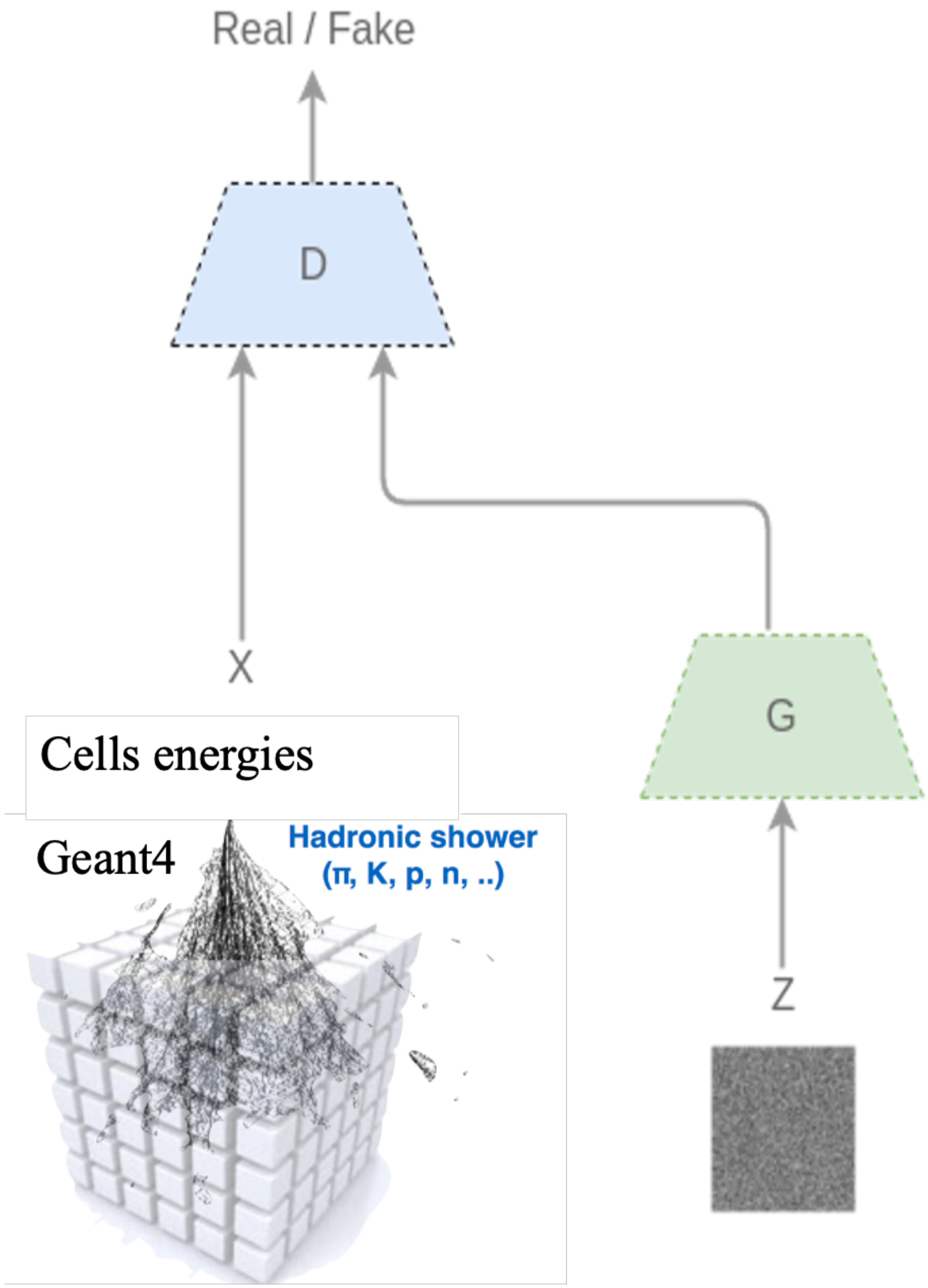


Generative Adversarial Network

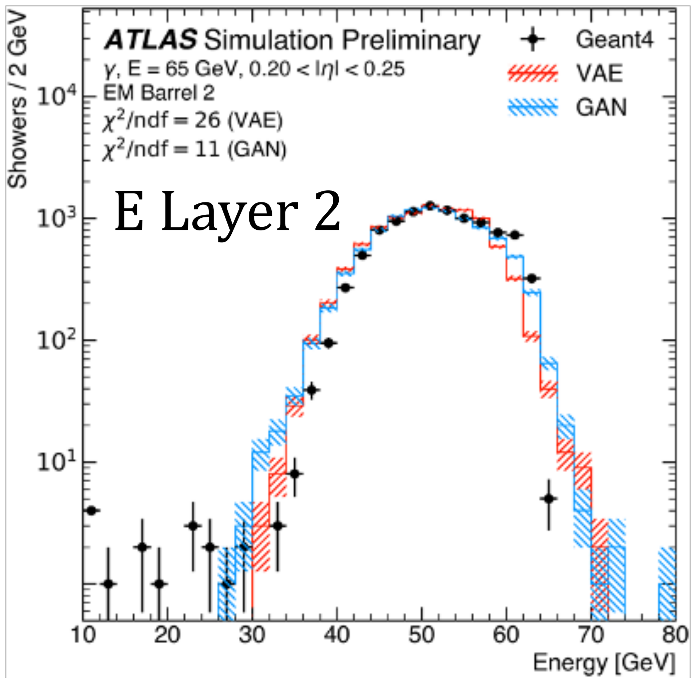
- Generative model: Create an object (picture) from random noise
- Uses two networks:
  - A **Generator**: Create data from noise
  - A **Discriminator**: Try to separate the generated data from the training data
- Unsupervised learning, where the Generator tries to trick the discriminator

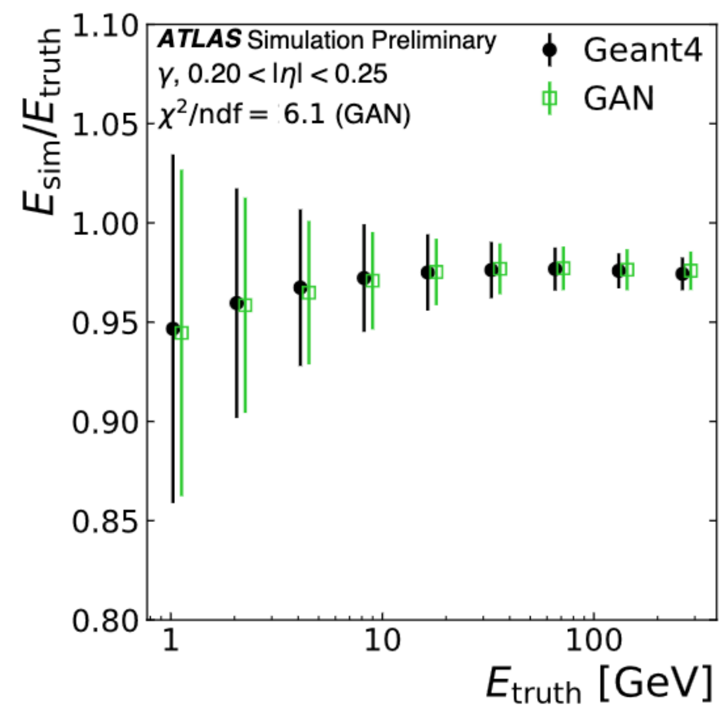


Calorimeter Simulation : GAN



- Tries to simulate jet energy deposition in a Calorimeter (ATLAS)
- Good agreement with G4 shower
- Generate realistic showers 100x faster
- Hard to train, other approaches being studied:
  - Variational auto-encoder
  - Diffusion Model

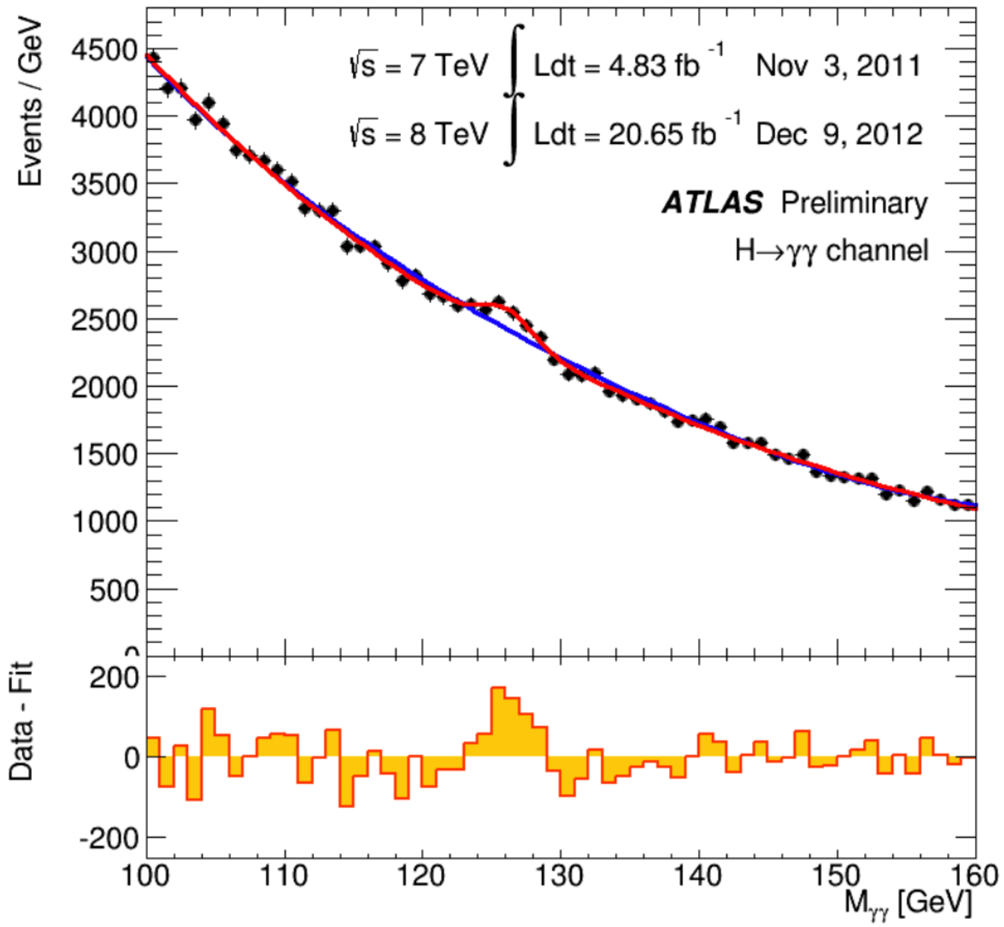




[Deep generative models for fast shower simulation in ATLAS]

Data Analysis

Simulation based inference



- « Historical »HEP analysis: binned histogram on a particles-level variable used to compute a likelihood ratio between two hypothesis
- When looking at more complex processes ➔ a single variable is not enough
- We would like to test multiple hypotheses

NN score for classifier :
Hypothesis θ_i / Null Hypothesis(ref)

$$s(x_i, \theta = \theta_1) = \frac{p(x_i \mid \theta_1)}{p(x_i \mid \theta_1) + p(x_i \mid ref)}$$

Data

Likelihood ratio for hypothesis θ_i

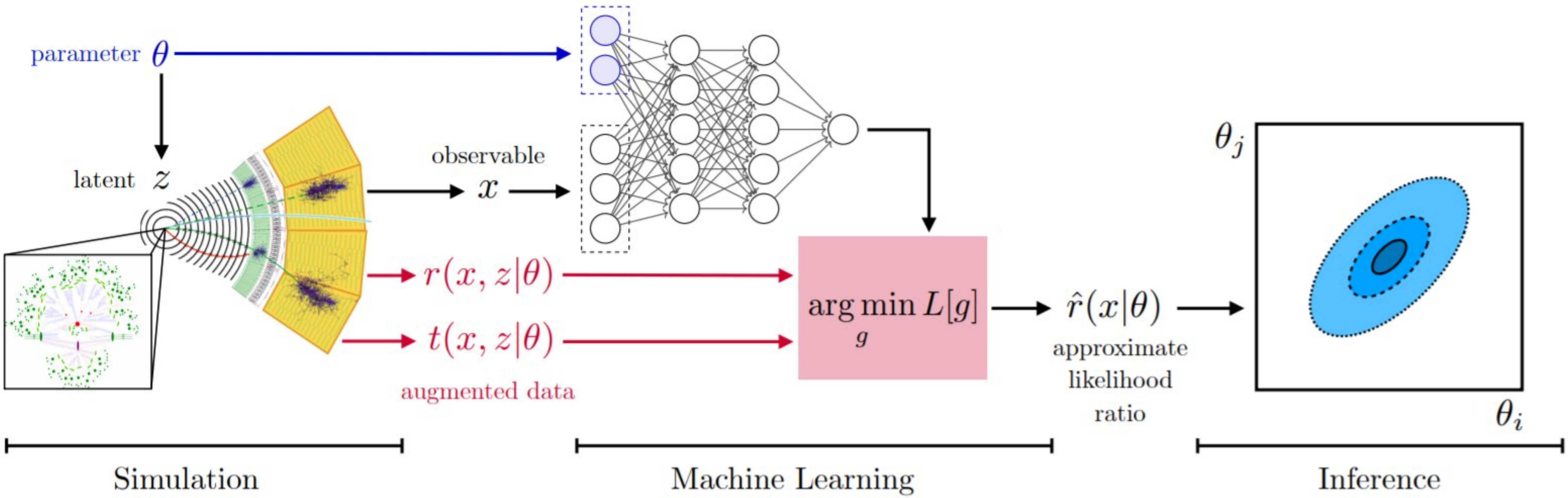
$$\frac{p(x_i \mid \theta_1)}{p(x_i \mid ref)} = \frac{s(x_i, \theta = \theta_1)}{1 - s(x_i, \theta = \theta_1)}$$

- SBI: use an NN binary classifier to estimate directly the likelihood ratio
- Can operate in high-dimension variable space
- Unbinned (can be applied event by event)

[Constraining Effective Field Theories with Machine Learning]

Simulation based inference

- Allows us to extract directly the likelihood ratio
- Large number of networks trained to account for NN uncertainty
- Analysis soon to be published demonstrating those methods



Conclusion

- Machine Learning is becoming a major tool for LHC experiments
- Long history of ML use: early adopters of the BDT techniques
- Used everywhere from Reconstruction to Simulation and Analysis
- Future developments are planned using the latest network architectures

Backup