ETH zürich

Machine Learning for semivisible jet searches

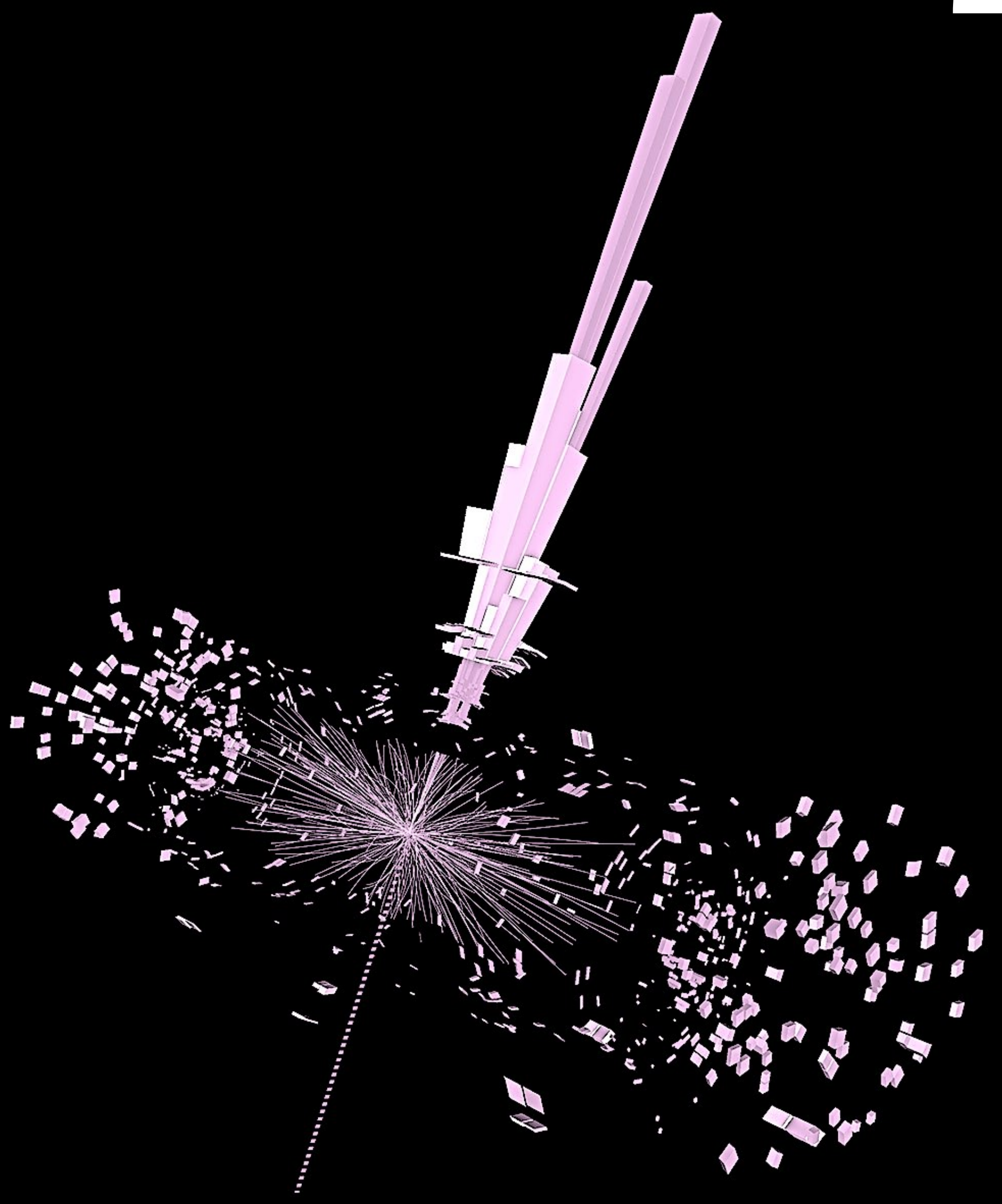
Thea Klæboe Årrestad
(ETH Zürich)

Semivisible Jets Workshop
ETH Zurich July 6th 2022


Jets with weird energy patterns
R_{inv} = 0.0-1.0

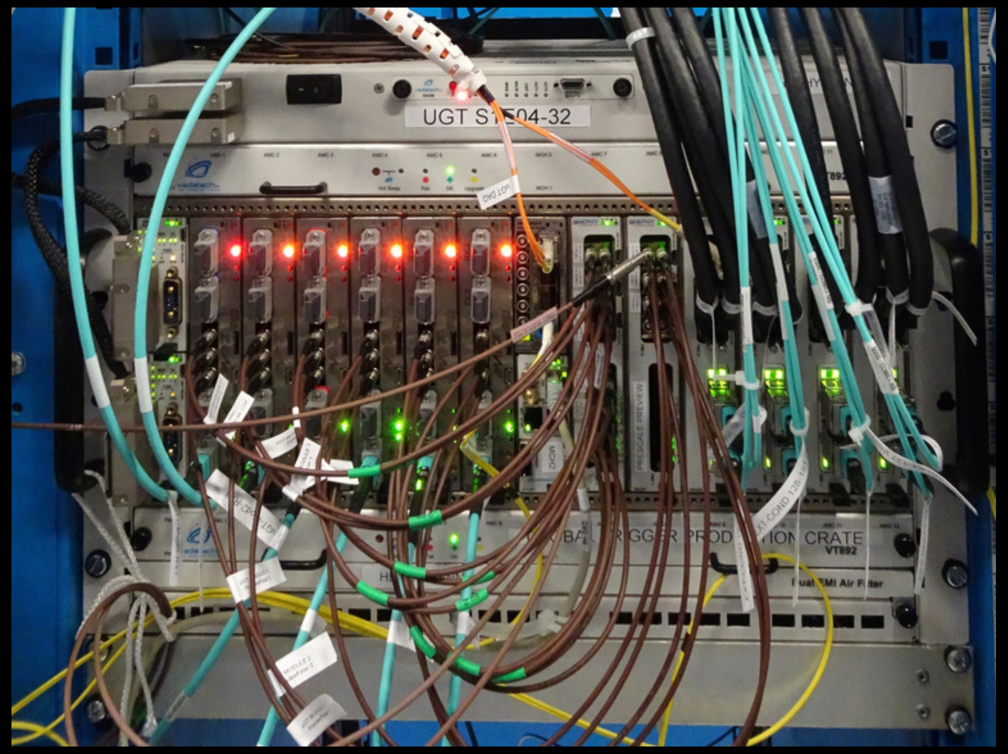
Where and how to best utilize ML in SVJ searches ?

Detector



40 MHz

Level-1 hardware trigger
- Improve signal acceptance?
- Latency O(1)μs



SVJ trigger?

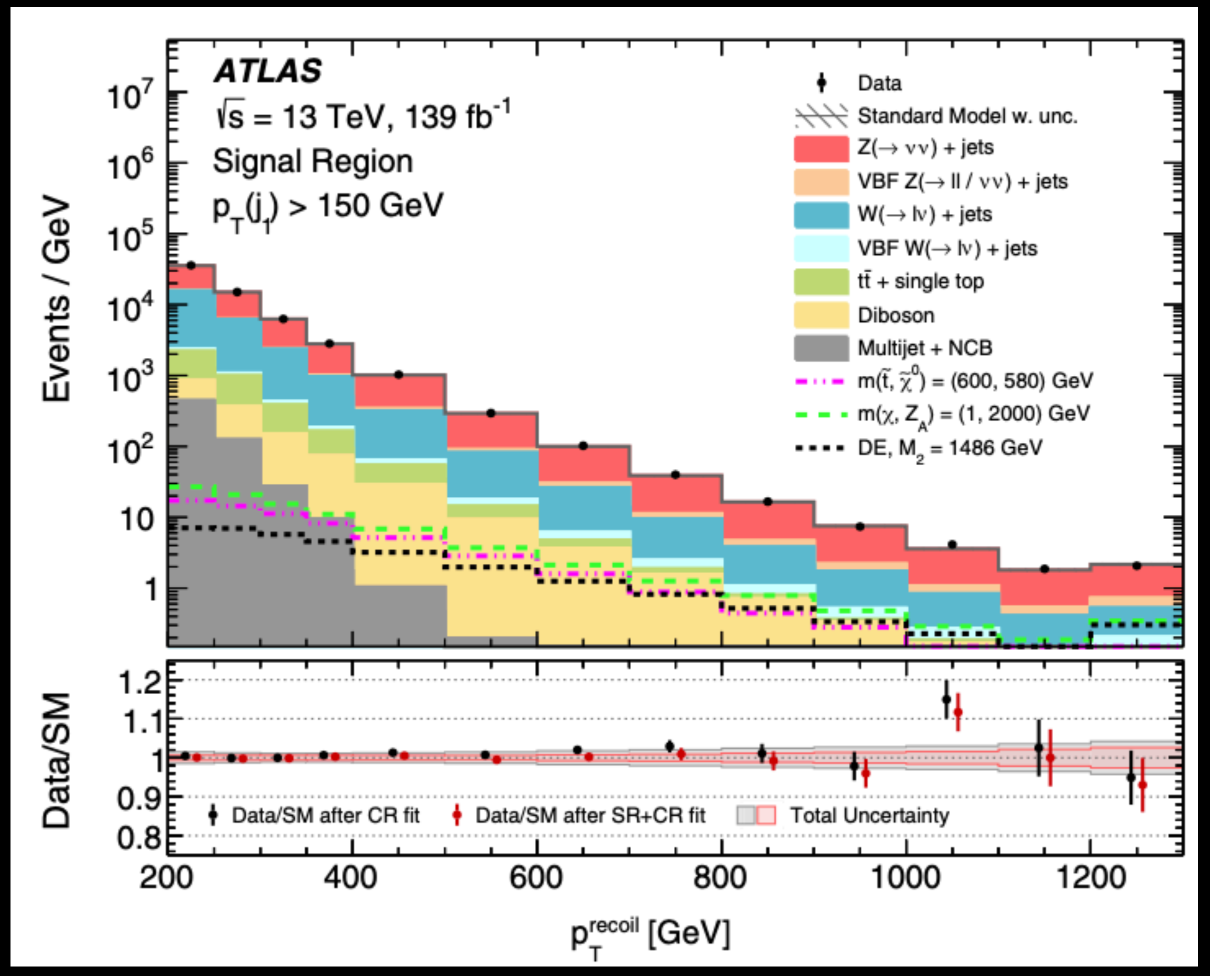
750 kHz

High Level Trigger
- Improve signal acceptance?
- Latency O(100) ms

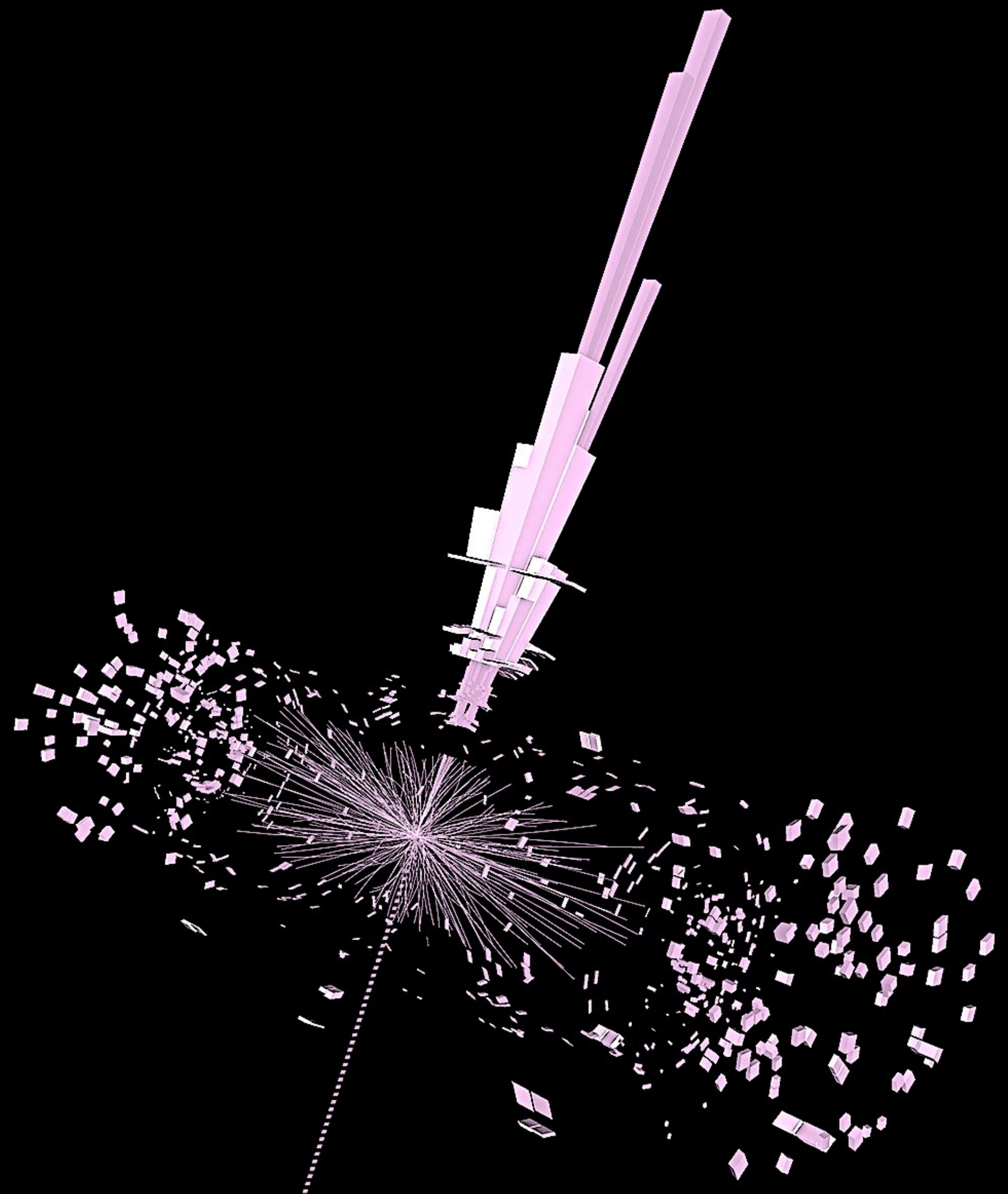
SVJ trigger?

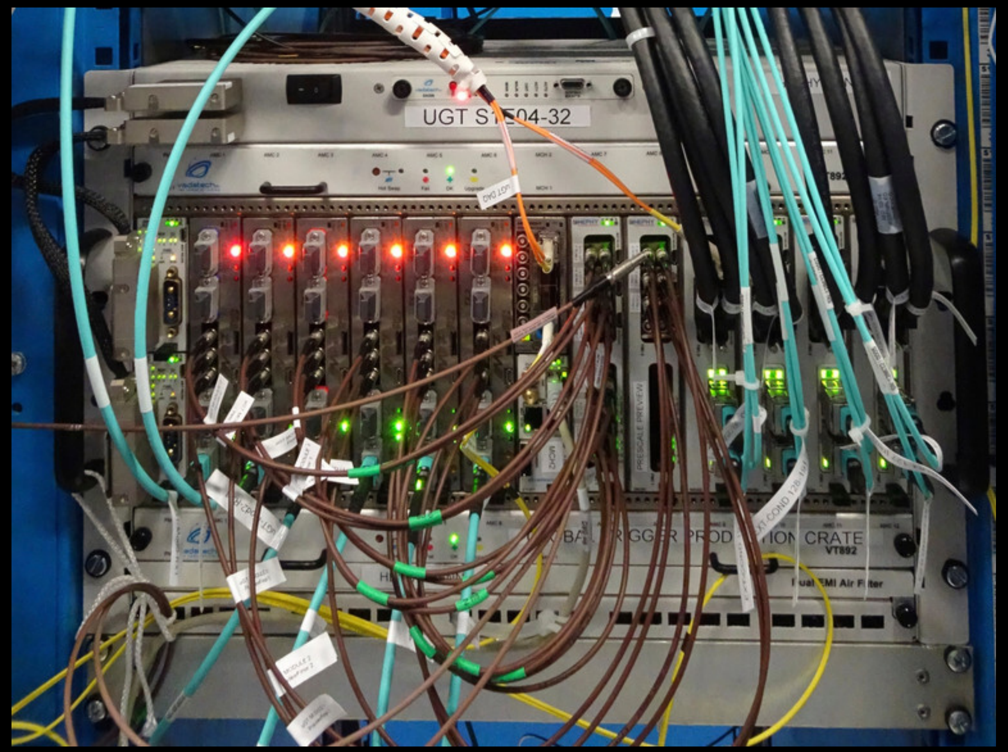
7.5 kHz

Offline reconstruction and analysis



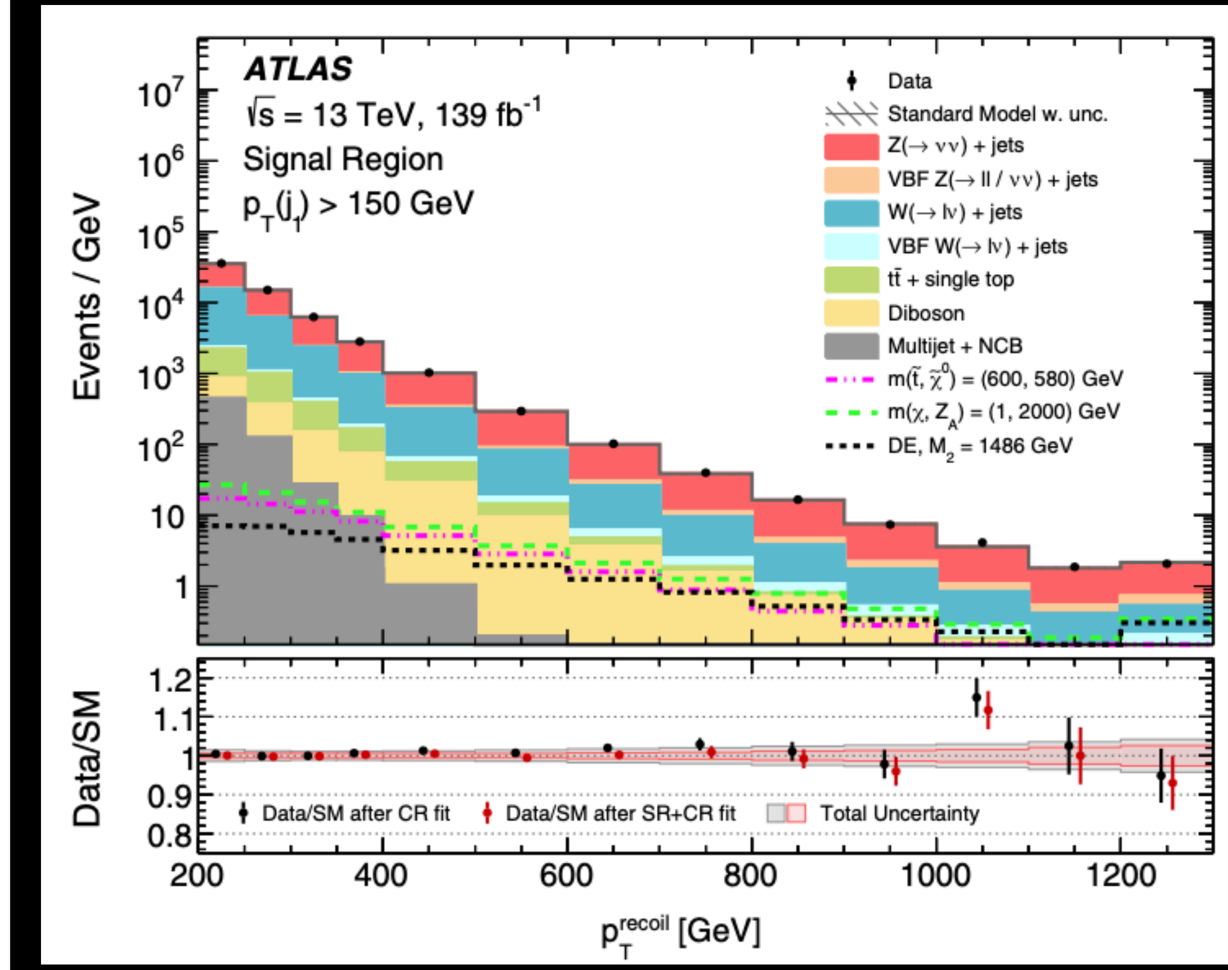
Model-independent searches?

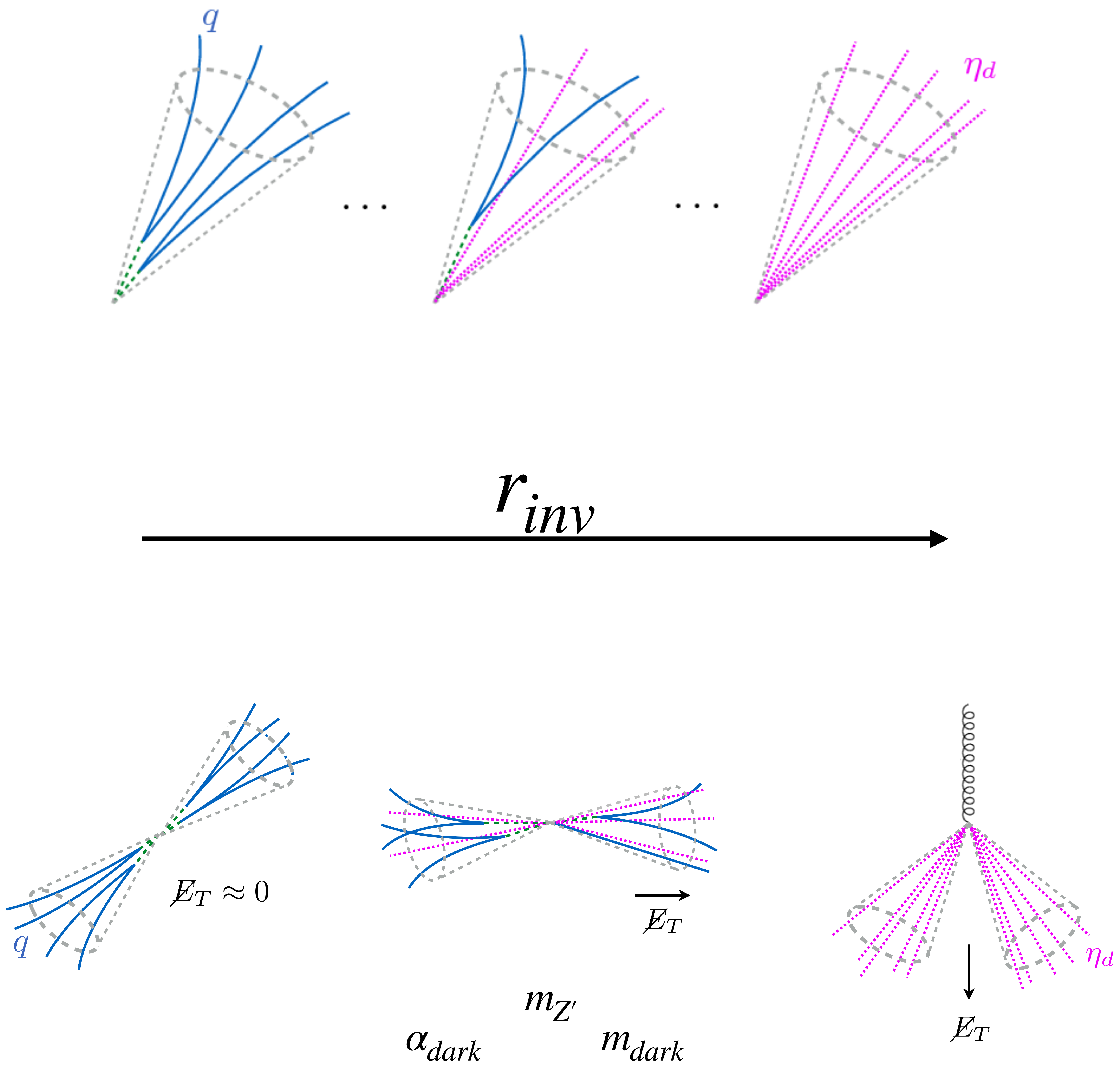


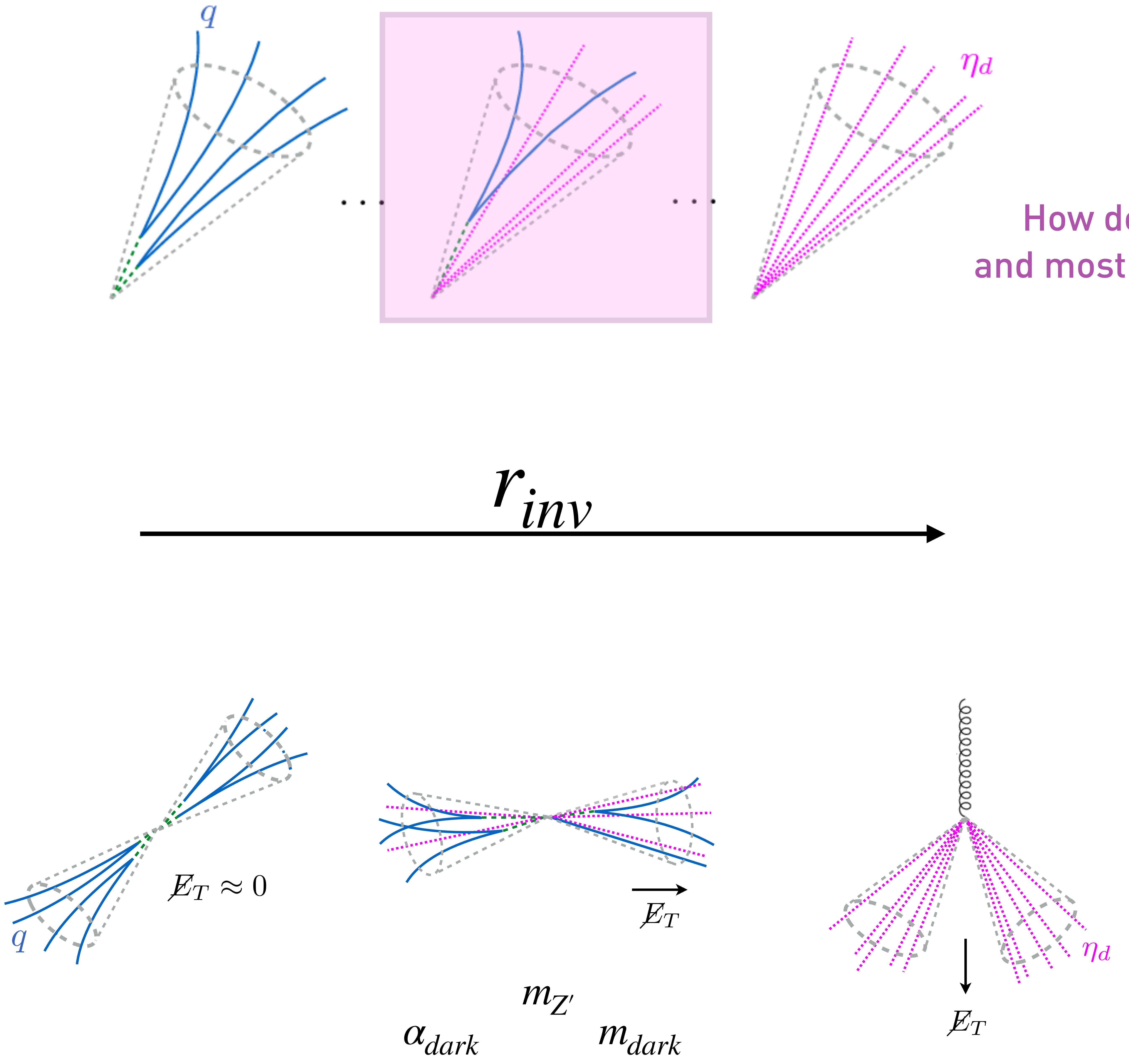


Offline reconstruction and analysis

- ML to improve sensitivity (and acceptance?)

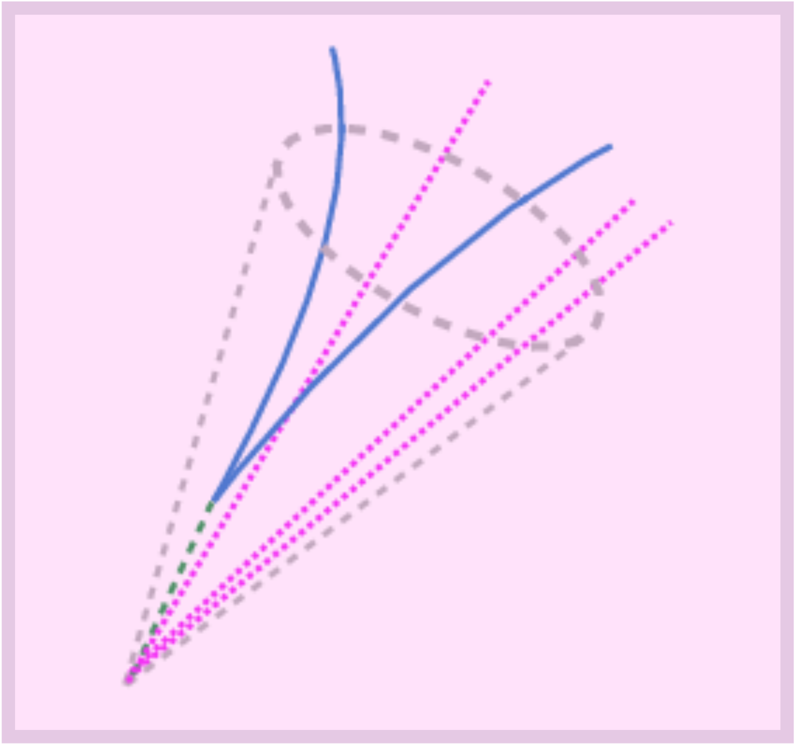



q
η_d
r_{inv}
$\not{E}_T \approx 0$
$\not{E}_T$
$m_{Z'}$
α_{dark}
m_{dark}
$\not{E}_T$
q
η_d



How do we build the strongest
and most generic classifier for SVJ?

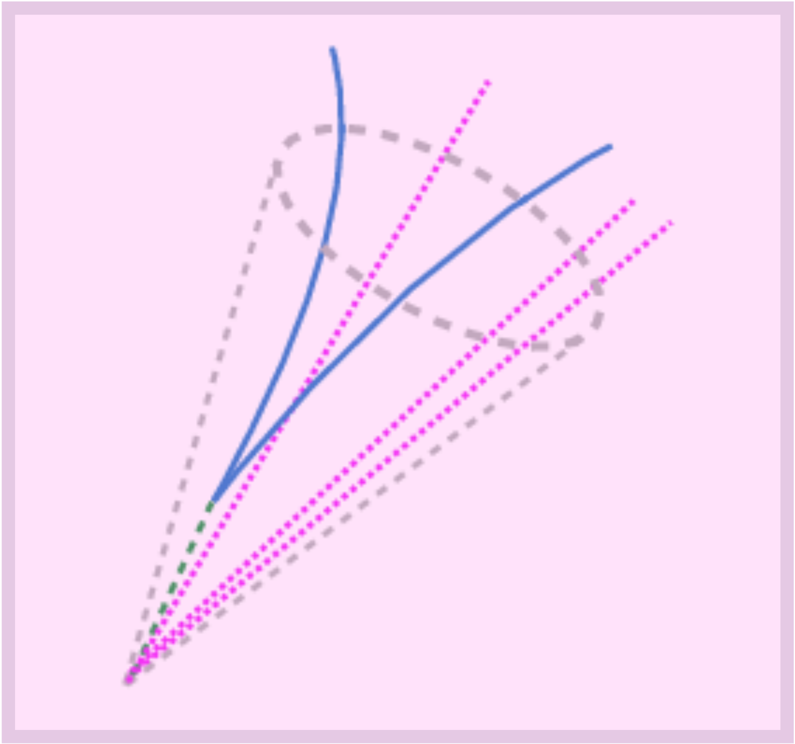
arxiv:1707.05326



Fully supervised

- Powerful classifier
- Learns features for S vs B
- Requires ~absolute knowledge of signal (MC)
- Model dependent

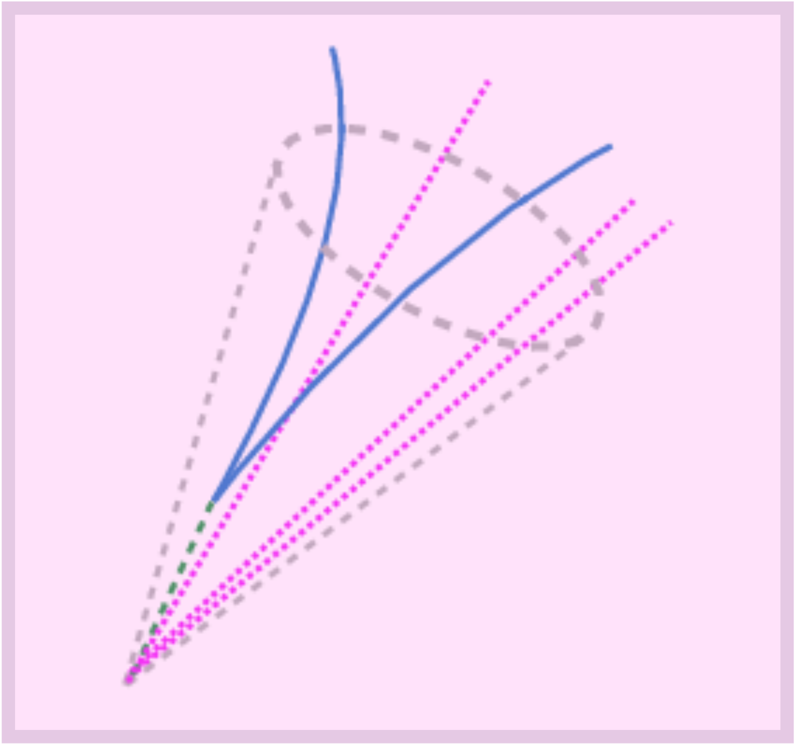
Degree of prior



Fully supervised
- Powerful classifier
- Learns features for S vs B
- Requires ~absolute knowledge of signal (MC)
- Model dependent

Weakly supervised
- Powerful classifier
- Learns distinct signal features
- Signal prior needed (no MC?)
- Some residual model dependence

Degree of prior



Fully supervised

- Powerful classifier
- Learns features for S vs B
- Requires ~absolute knowledge of signal (MC)
- Model dependent

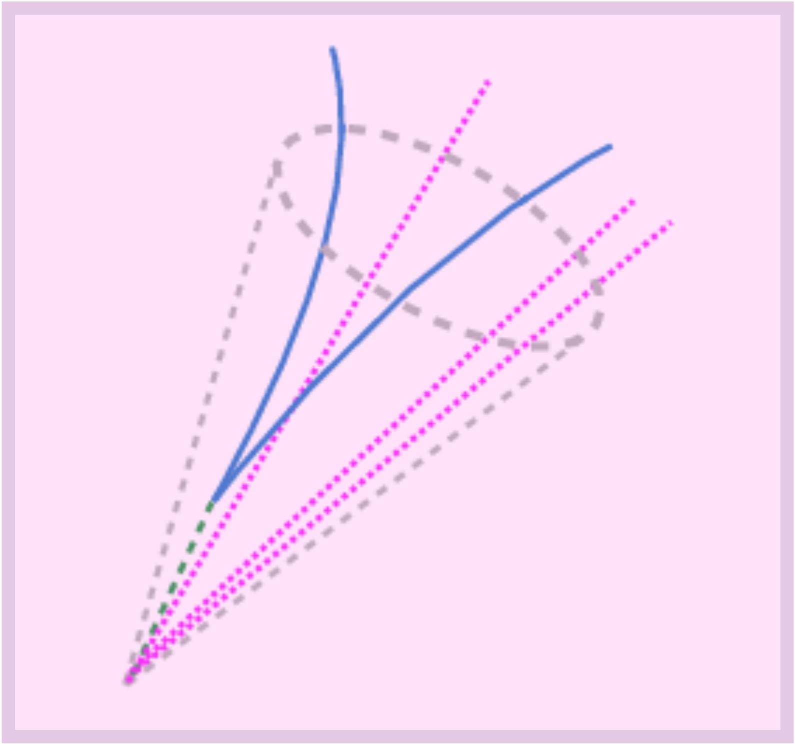
Weakly supervised

- Powerful classifier
- Learns distinct signal features
- Signal prior needed (no MC?)
- Some residual model dependence

Unsupervised

- Anomaly detection algorithm e.g (V)AE
- Learns background compression/density
- No signal prior (no MC)
- Model independent

Degree of prior



Fully supervised

- Powerful classifier
- Learns features for S vs B
- Requires ~absolute knowledge of signal (MC)
- Model dependent

Weakly supervised

- Powerful classifier
- Learns distinct signal features
- Signal prior needed (no MC?)
- Some residual model dependence

Unsupervised

- Anomaly detection algorithm e.g (V)AE
- Learns background compression/density
- No signal prior (no MC)
- Model independent

Degree of prior

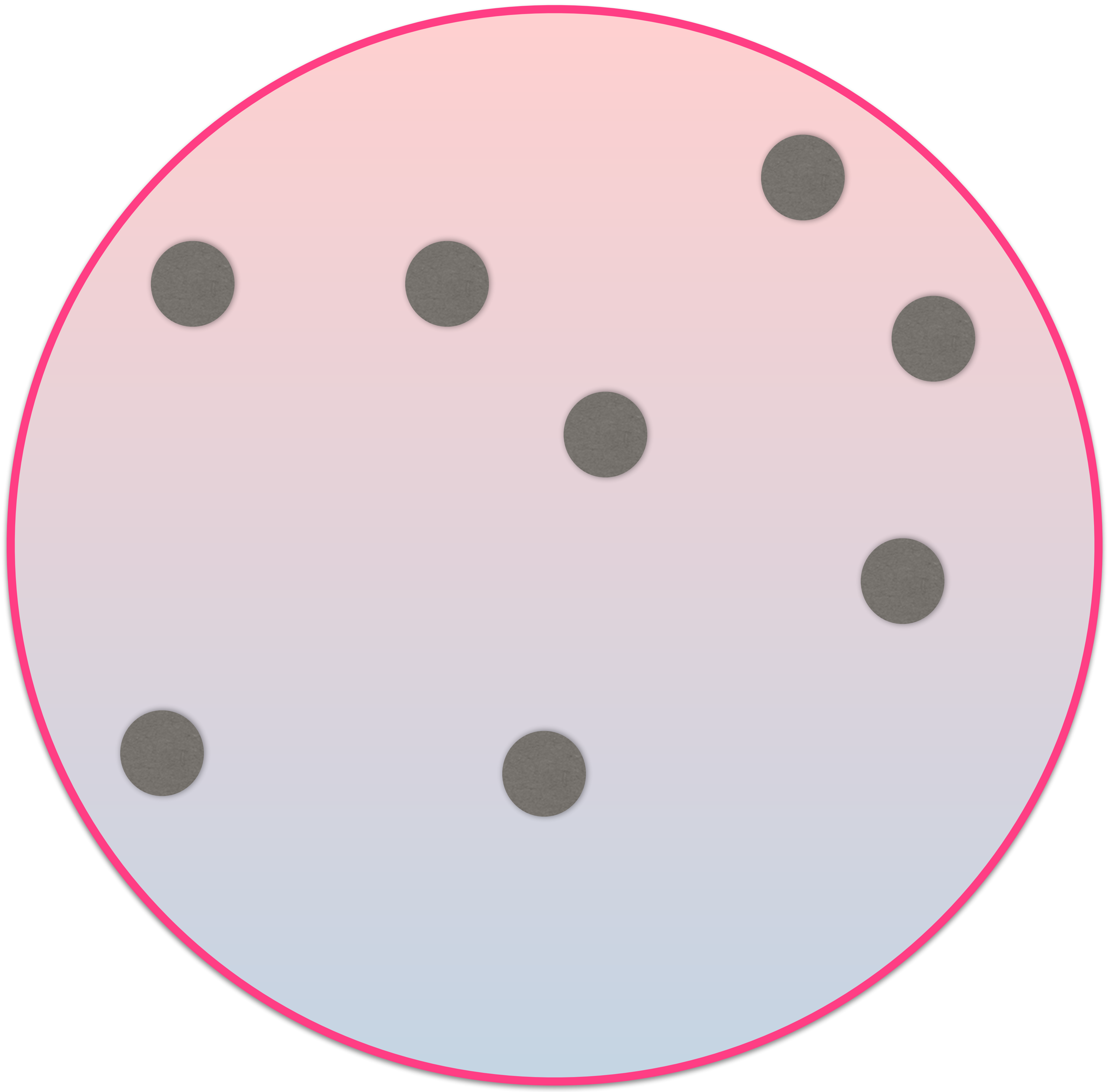
How to best represent a jet?

How to do classification without labels?

How to utilise in the way we select our data?

Fully supervised

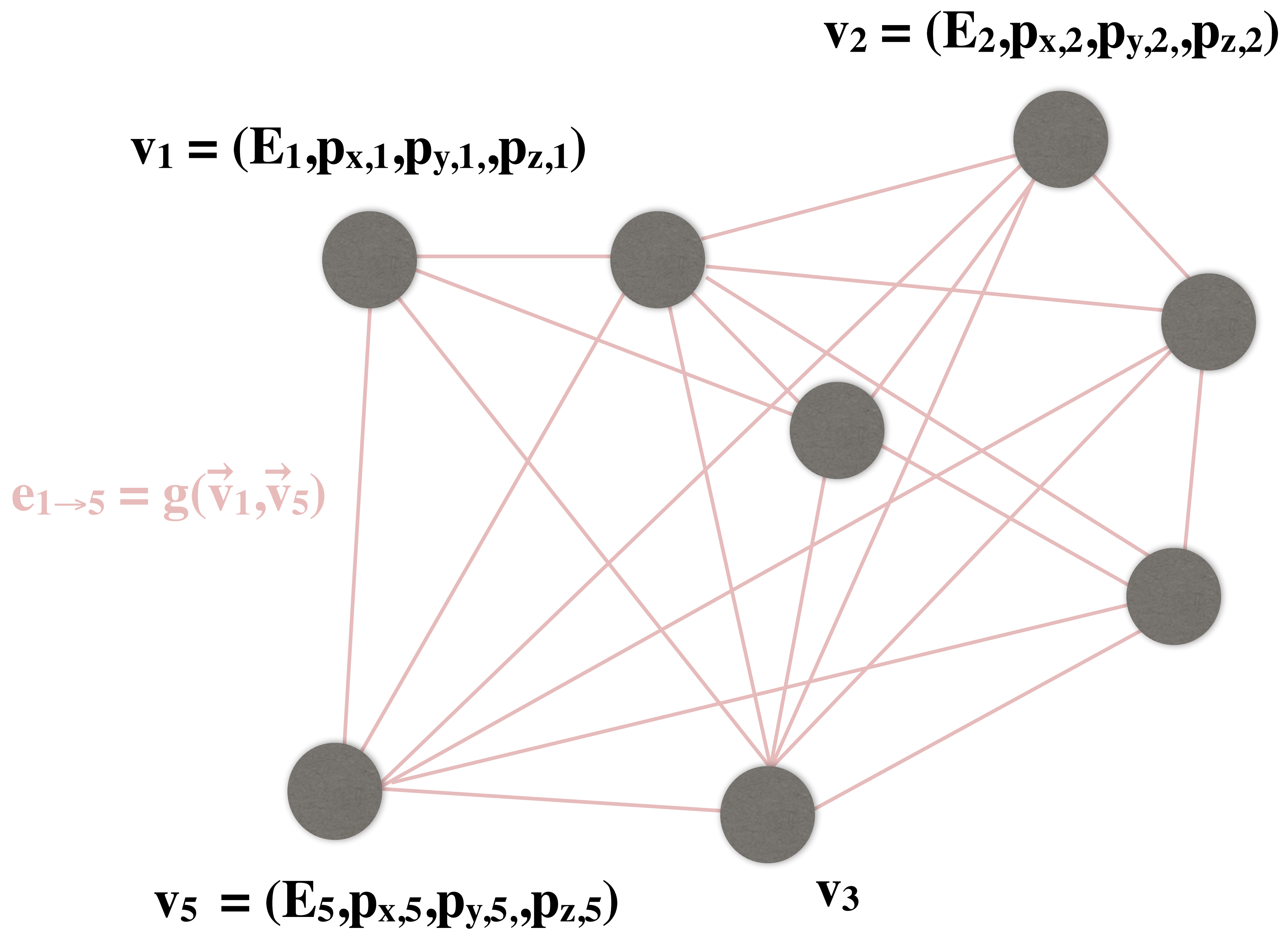
Best representation of jet (sparse, unordered, permutation inv.)?



Fully supervised

Best representation of jet (sparse, unordered, permutation inv.)?
SOTA: Graph Neural Networks acting on point cloud data

- ParticleNet (GNN on point cloud)
  LundNet (GNN,Lund plane)
  ABCNet (GNN, attention)
  Point Cloud Transformers (transformer, attention)
  ParticleNeXt (GNN, attention, Lund)
  ParT (transformer, attention)

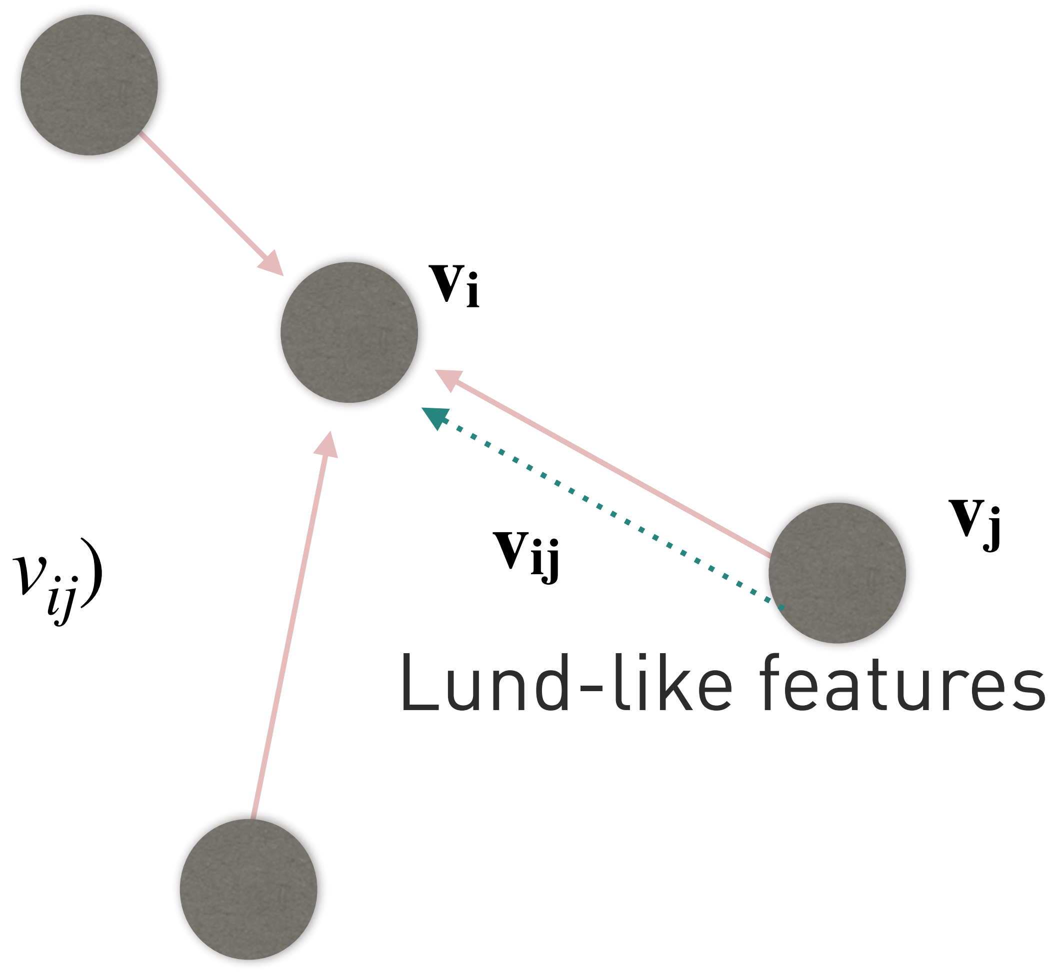


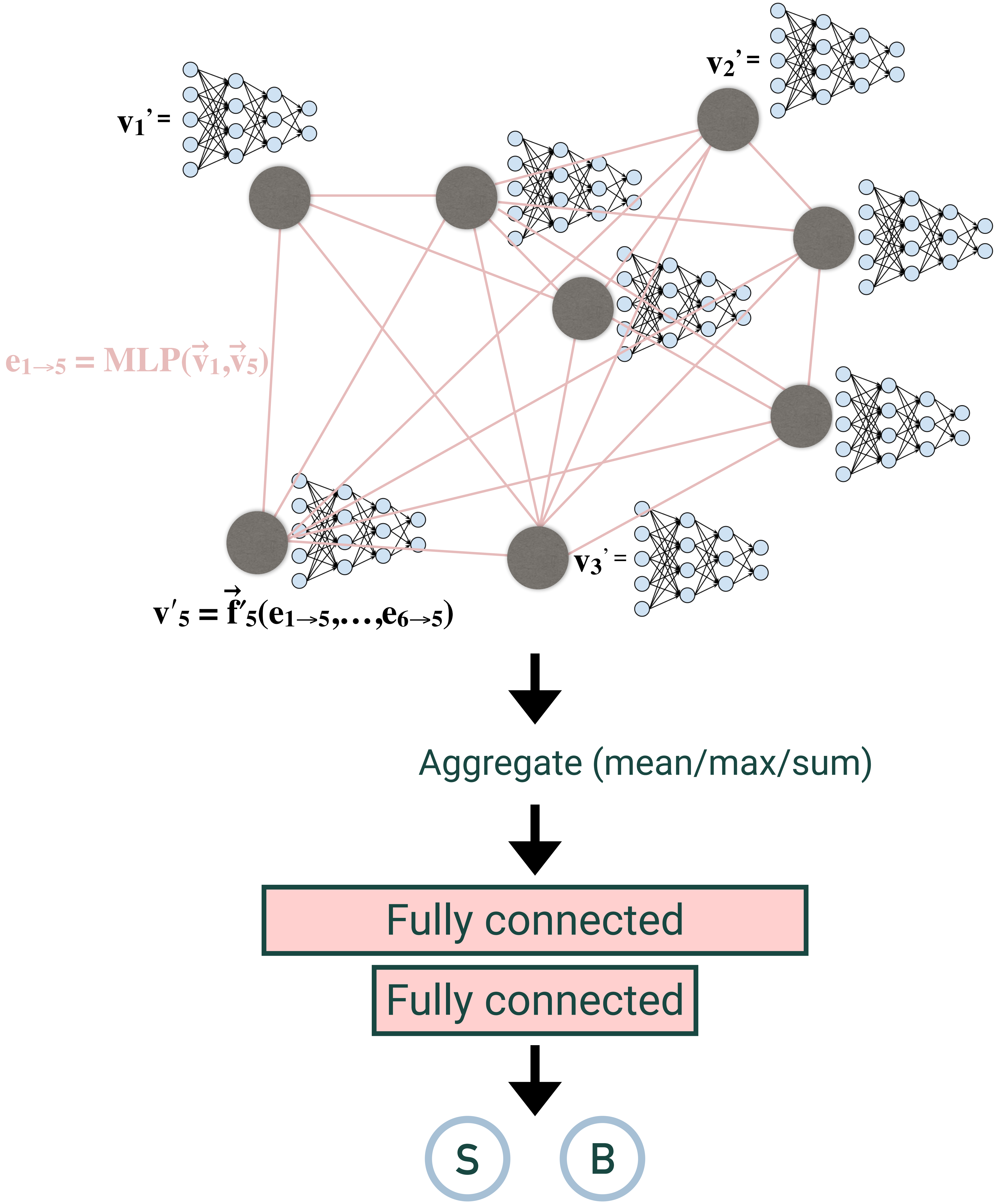
Fully supervised

Best representation of jet (sparse, unordered, permutation inv.)?
SOTA: Graph Neural Networks acting on point cloud data

- ParticleNet (GNN on point cloud)
  LundNet (GNN,Lund plane)
  ABCNet (GNN, attention)
  Point Cloud Transformers (transformer, attention)
  ParticleNeXt (GNN, attention, Lund)
  ParT (transformer, attention)

$$e_{ij} = MLP(v_i, v_j, v_{ij})$$





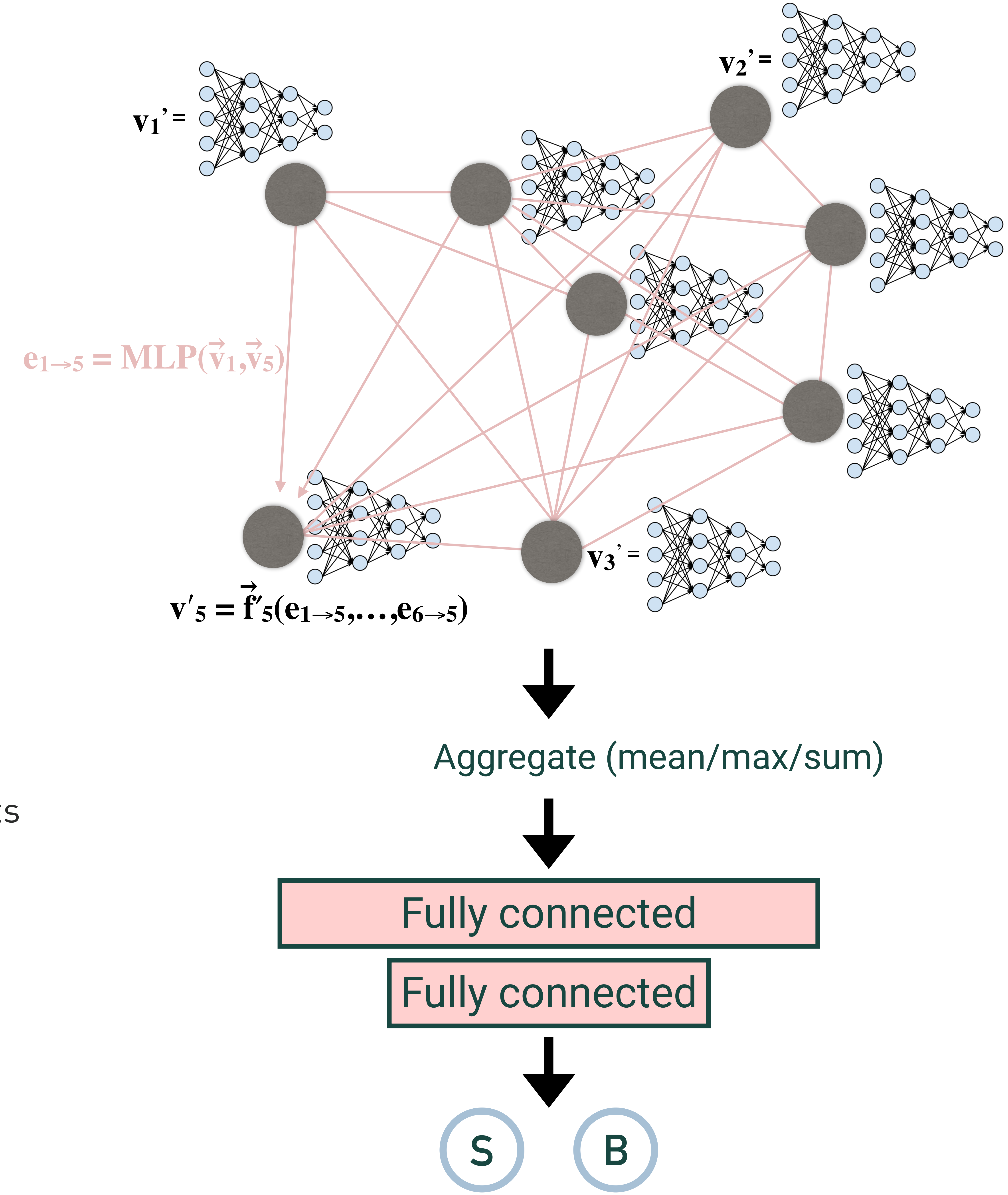
Fully supervised

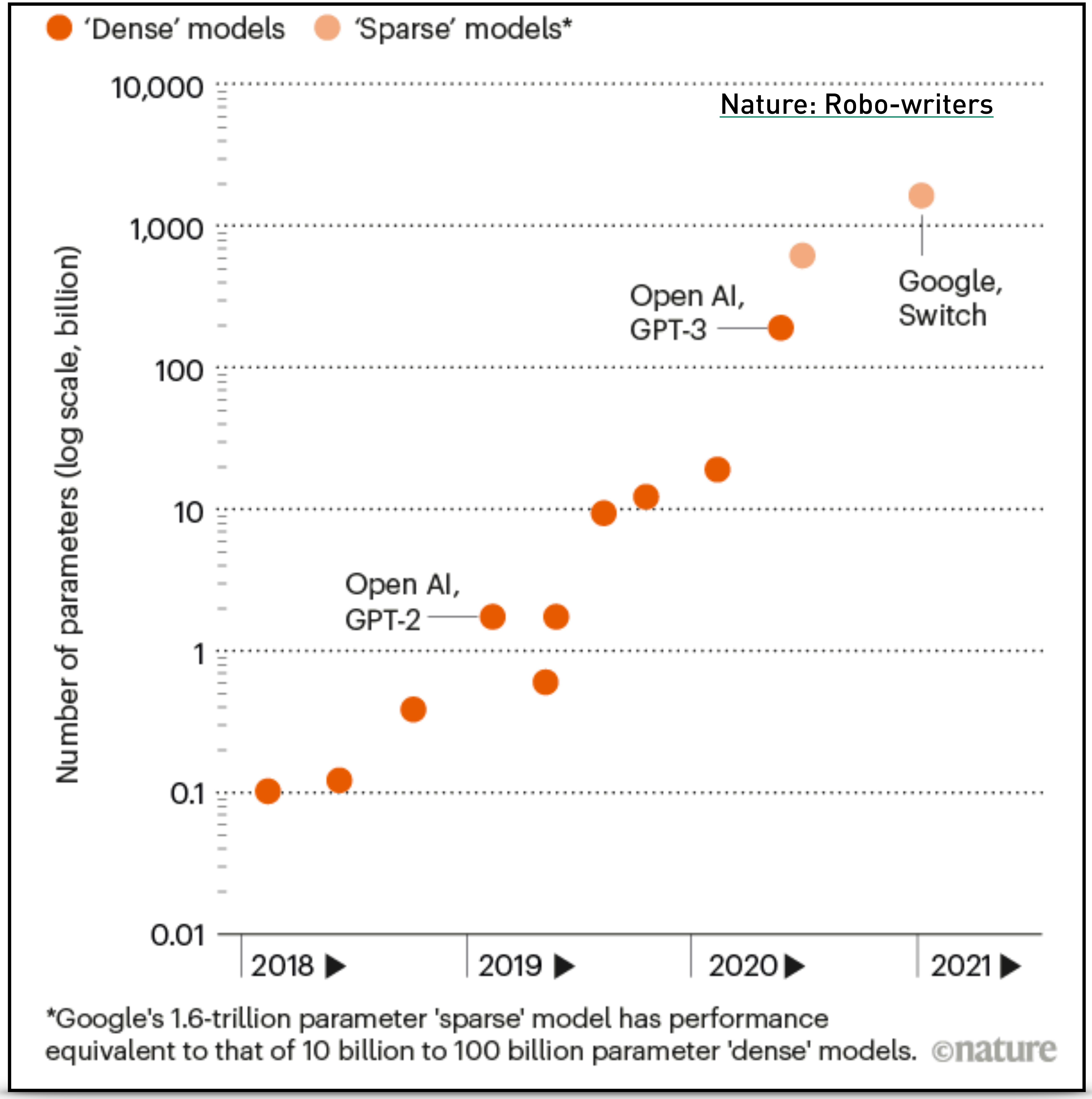
Best representation of jet (sparse, unordered, permutation inv.)?
SOTA: Graph Neural Networks acting on point cloud data

- ParticleNet (GNN on point cloud)
  LundNet (GNN,Lund plane)
  ABCNet (GNN, attention)
  Point Cloud Transformers (transformer, attention)
  ParticleNeXt (GNN, attention, Lund)
  ParT (transformer, attention)

What make these useful for SVJ:

- No high-level variables, these are learned from low-level inputs
- Attention and transformers:
  allow a network to learn unknown important jet features





GPT-3: 175 billion parameters (0.16% of the human brain)

Example prompt

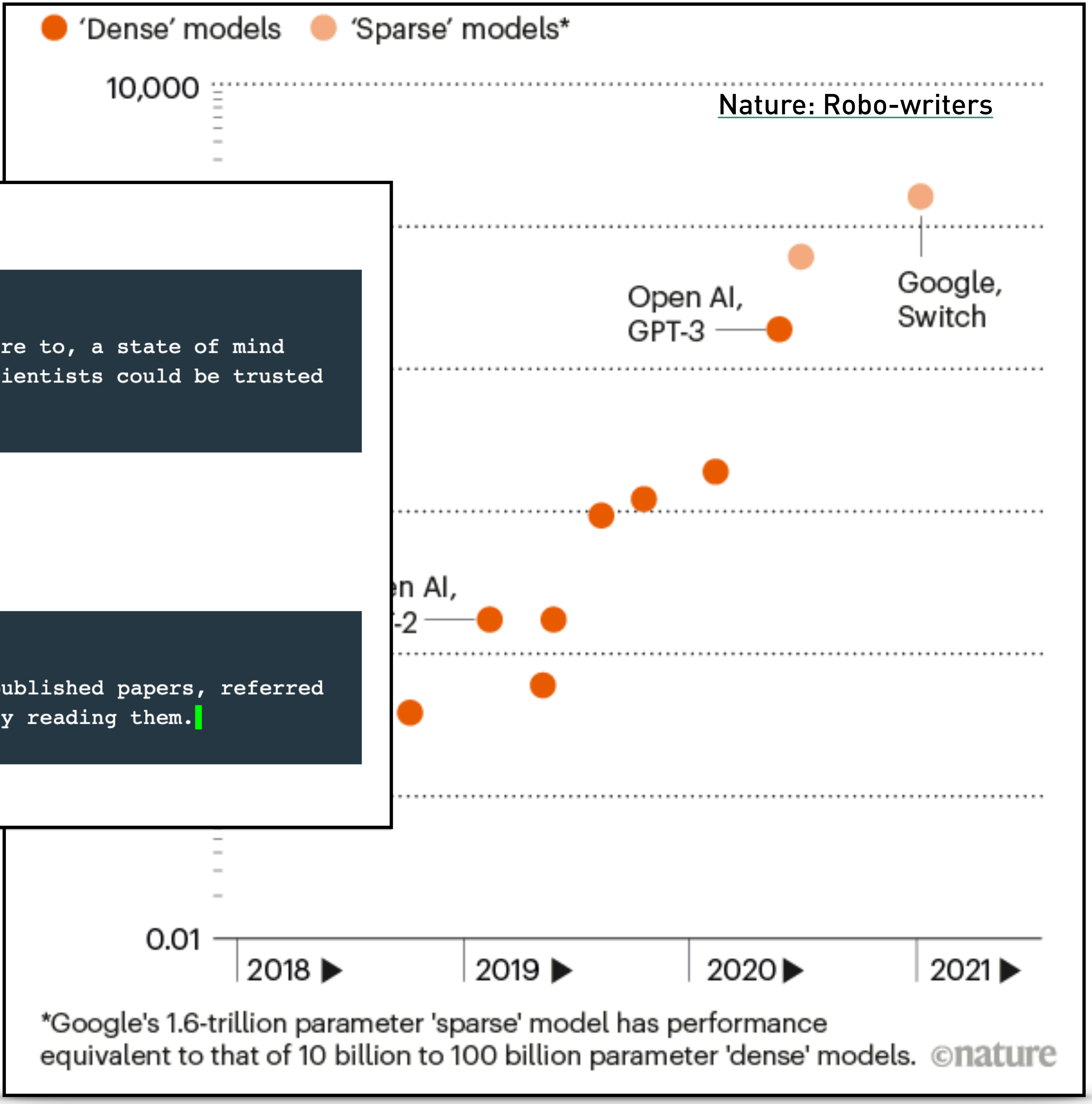
```
Rigor [adj.]
Something for scientists to aspire to, a state of mind
that would not be required if scientists could be trusted
to do their job.
```

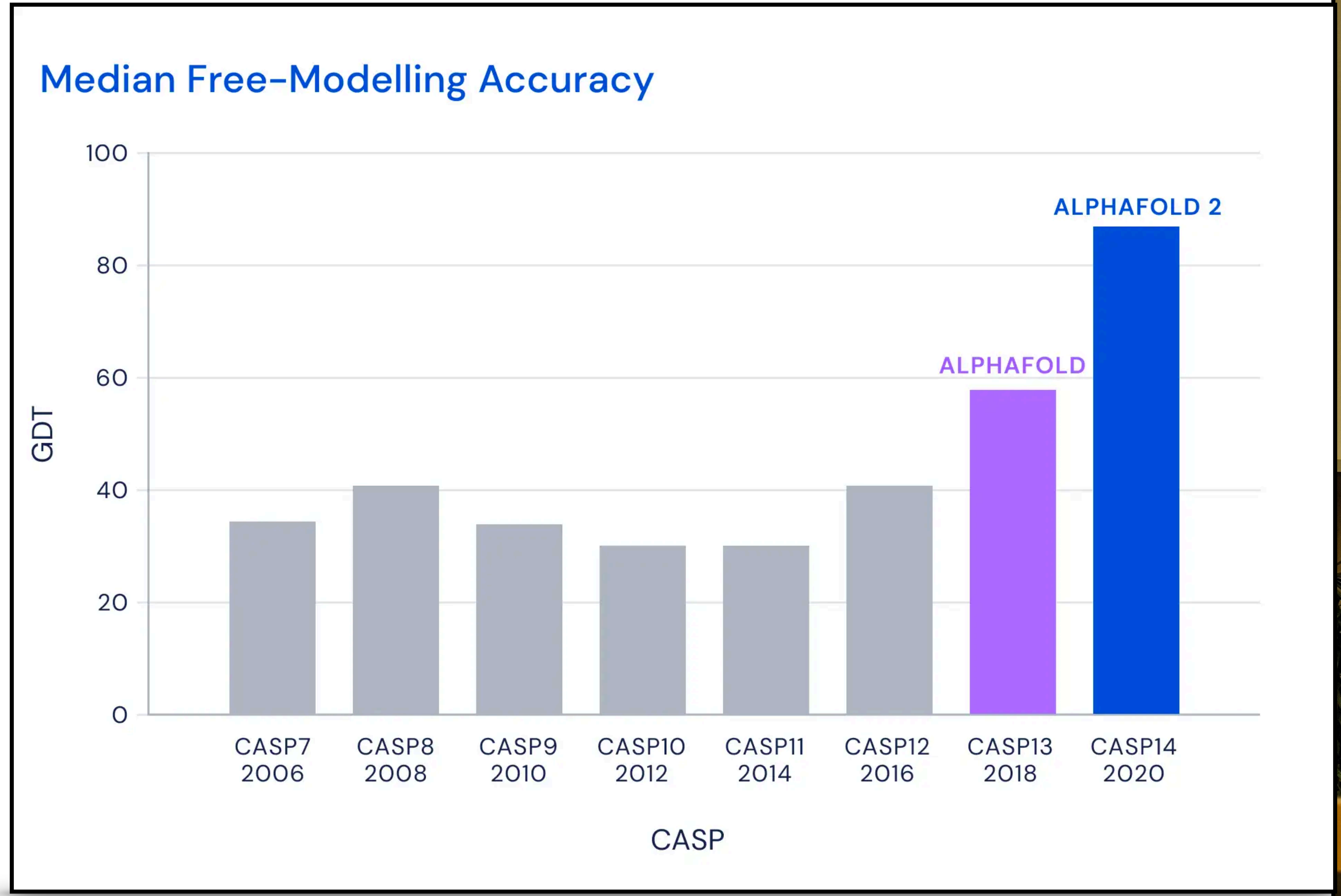
View next definition

GPT-3's output: 1 of 10

```
The Literature [noun]
A name given to other people's published papers, referred
to by scientists without actually reading them.
```

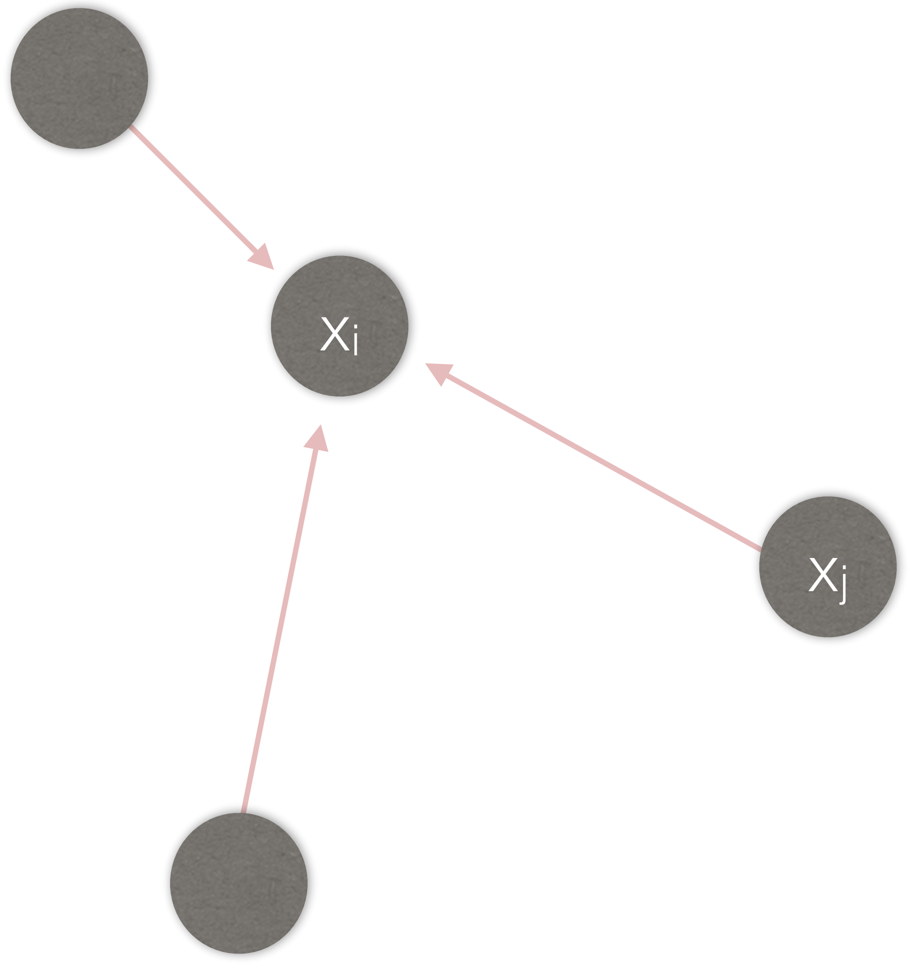
[Gwern.net](https://gwern.net)




Median Free-Modelling Accuracy
100
80
60
40
20
0
GDT
ALPHAFOLD 2
ALPHAFOLD
CASP7 2006
CASP8 2008
CASP9 2010
CASP10 2012
CASP11 2014
CASP12 2016
CASP13 2018
CASP14 2020
CASP


AlphaFold nature cover
The international journal of science / 30 July 2021
nature
PERFECT MATCH

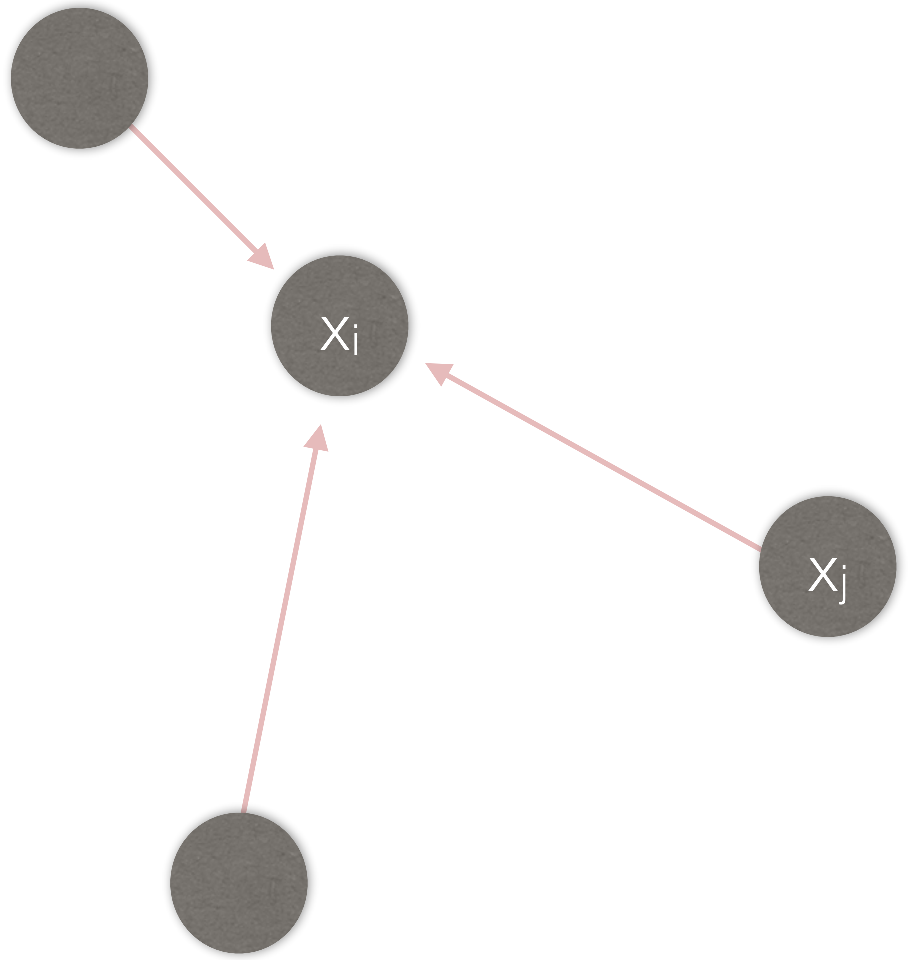
Attention & transformers



(Self-)Attention

- Let method learn relevant parts for task at hand
- Allows inputs to interact with each other (“self”) and find out who they should pay more attention to (“attention”).
- Outputs: aggregates of interactions and attention scores

Attention & transformers



(Self-)Attention

- Let method learn relevant parts for task at hand
- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores

Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

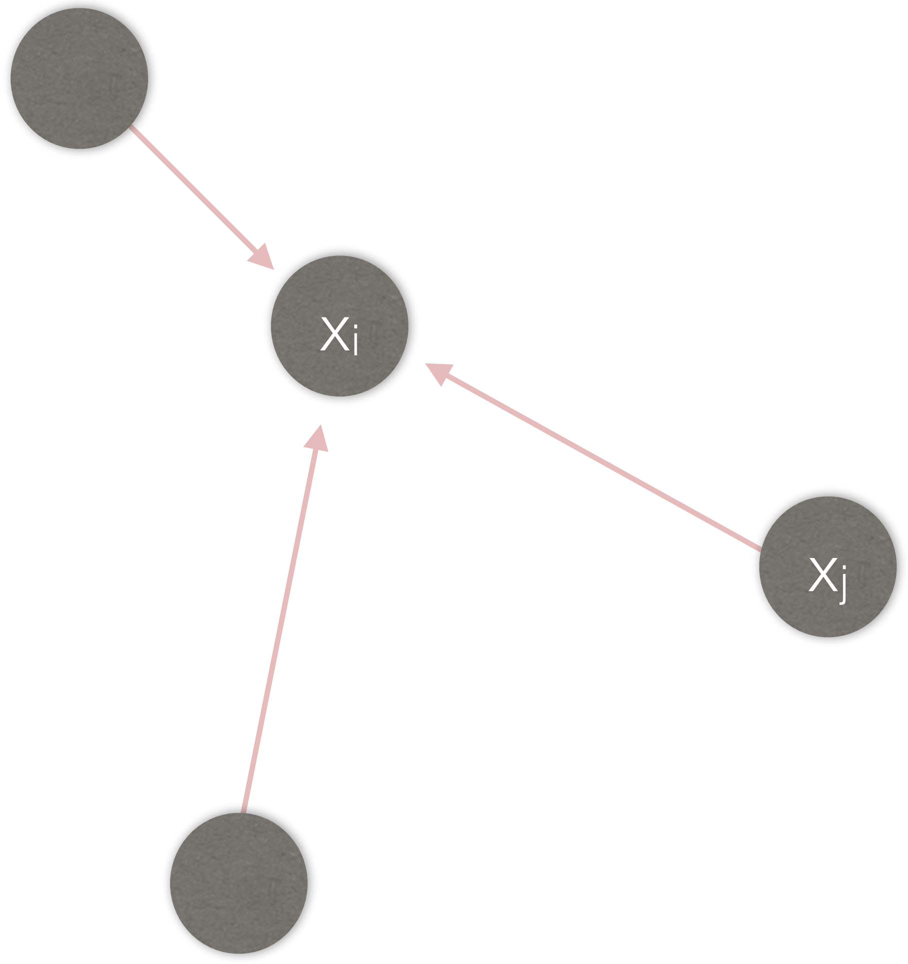
Weight (how related inputs are):

$$w'_{ij} = \boldsymbol{x}_i{}^{\mathsf{T}} \boldsymbol{x}_j$$

Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Attention & transformers



(Self-)Attention

- Let method learn relevant parts for task at hand
- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores

Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

Weight (how related inputs are):

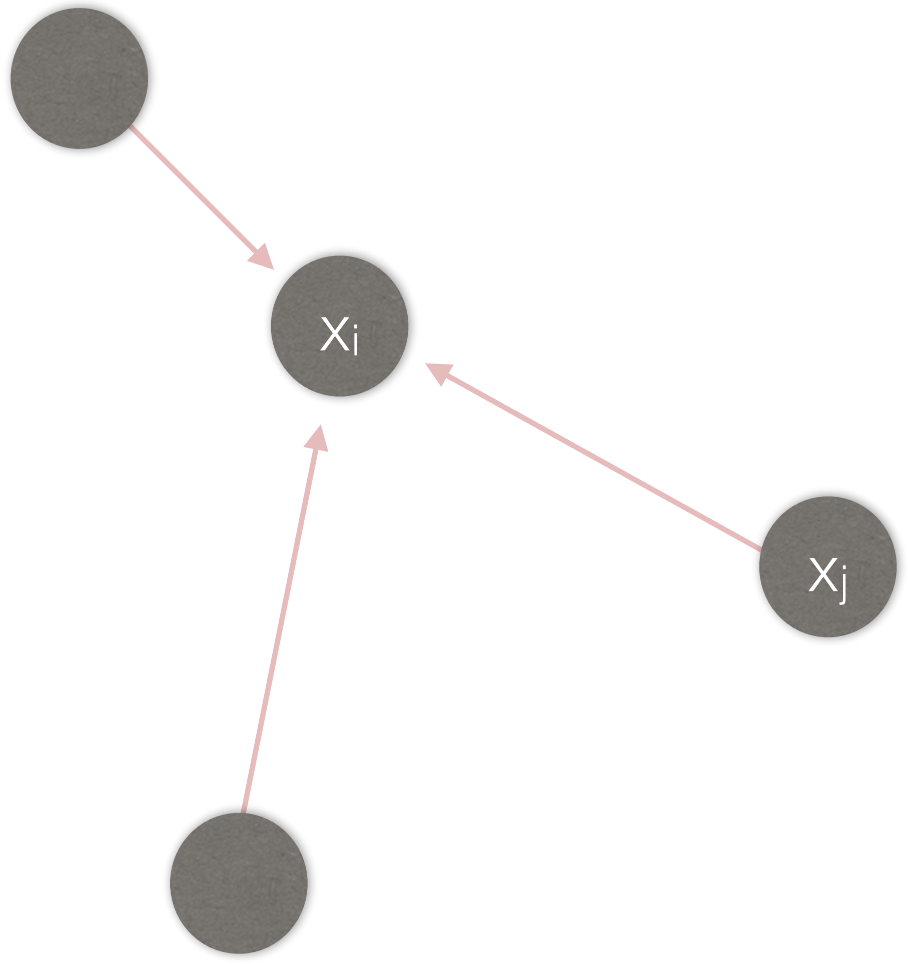
$x_j \rightarrow \mathrm{MLP}(x_J)$
$x_i \rightarrow \mathrm{MLP}(x_i)$

$$w'_{ij} = \boldsymbol{x}_i{}^{\mathsf{T}} \boldsymbol{x}_j$$

Map to [0,1]:

$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Attention & transformers



(Self-)Attention

- Let method learn relevant parts for task at hand
- Allows inputs to interact with each other ("self") and find out who they should pay more attention to ("attention").
- Outputs: aggregates of interactions and attention scores

Weighted sum over all input vectors:

$$\boldsymbol{y}_i = \sum_j w_{ij} \boldsymbol{x}_j$$

Weight (how related inputs are):

$x_j \rightarrow MLP(x_J)$
$x_i \rightarrow MLP(x_i)$

$$w'_{ij} = \boldsymbol{x}_i{}^{\mathsf{T}} \boldsymbol{x}_j$$

Attention weights: weighted importance between each pair of particles

- Determine relationship between all particles of point cloud
- Jet features become parameters of the model
- Several attention layers → different important features (multi-head attention)

Map to [0,1]:

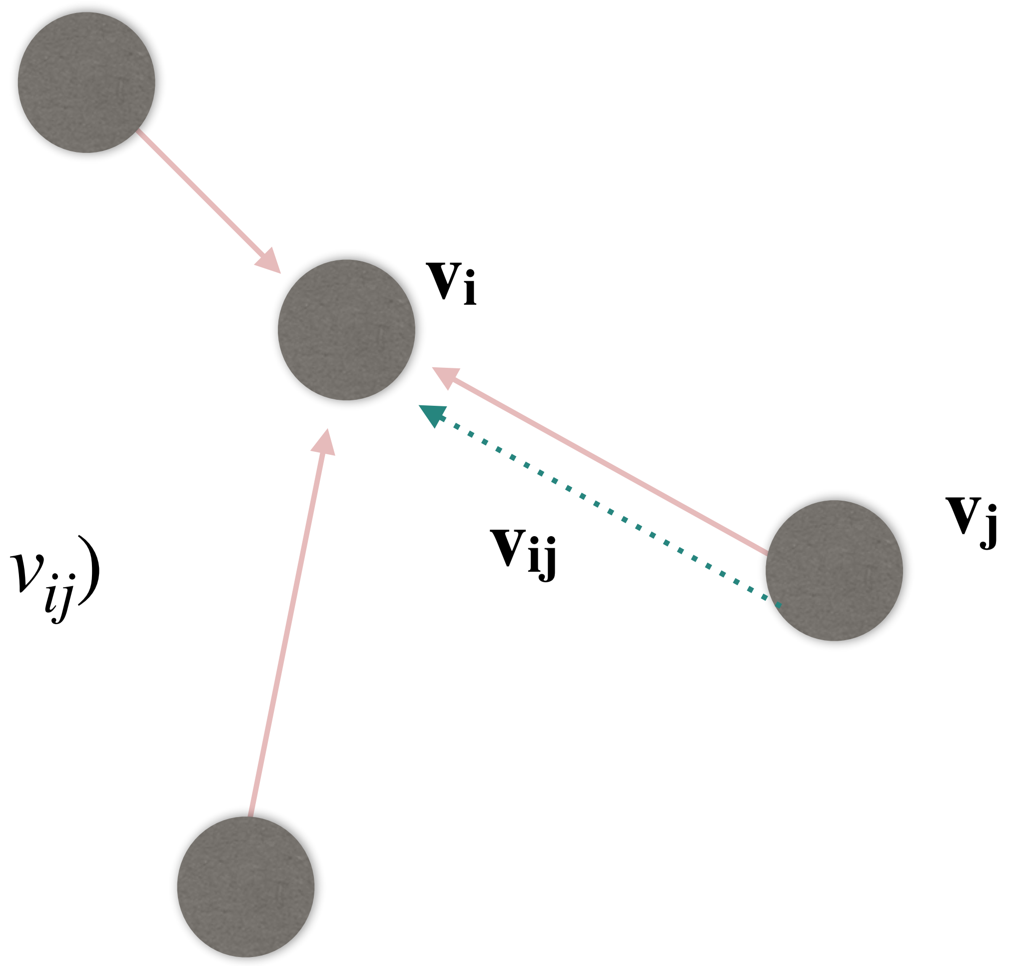
$$w_{ij} = \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$$

Attention & transformers

Transformer:

- Only set of interaction between units is self-attention!

$$e_{ij} = MLP(v_i, v_j, v_{ij})$$

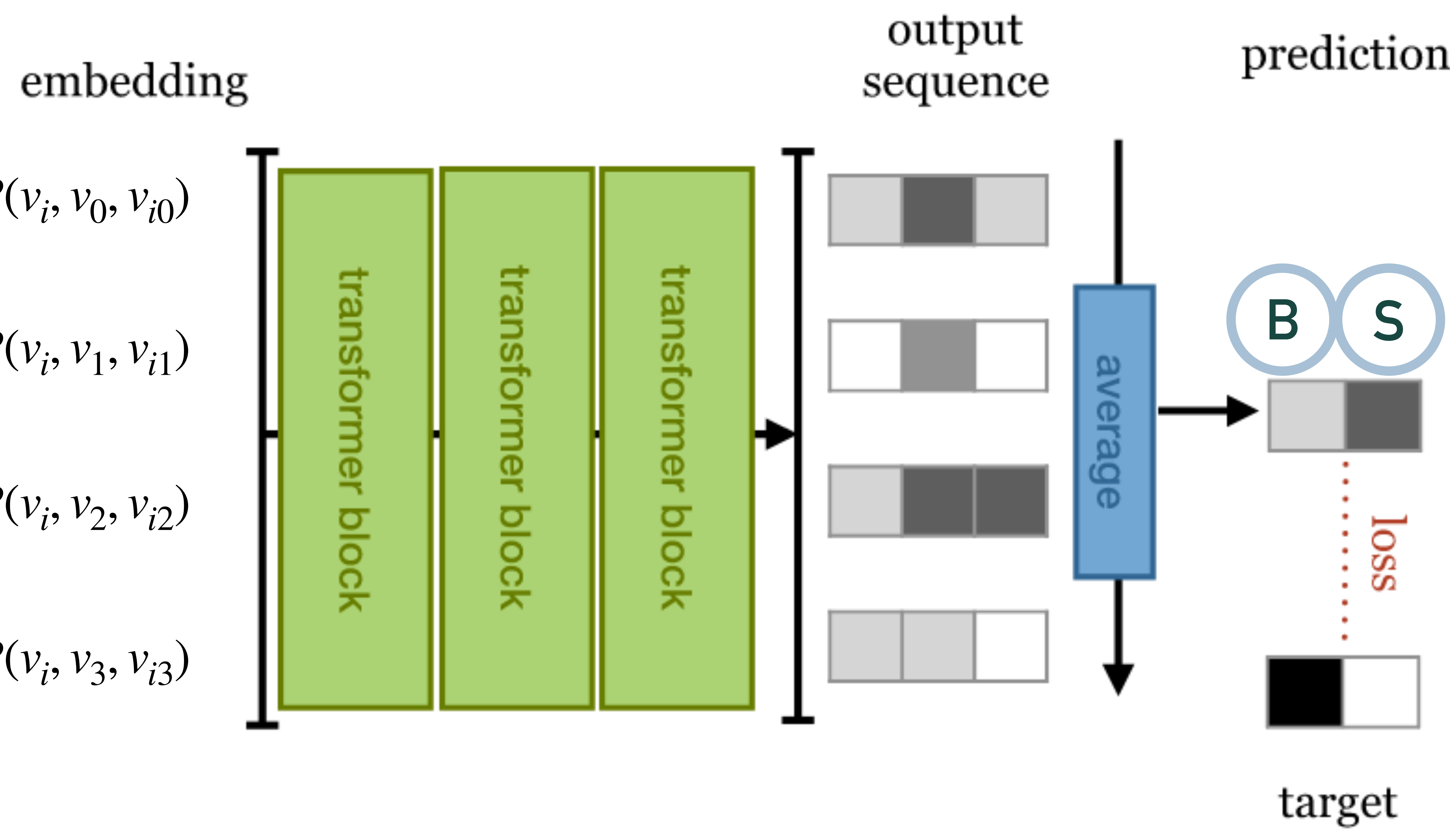


$$e_{i0} = MLP(v_i, v_0, v_{i0})$$

$$e_{i1} = MLP(v_i, v_1, v_{i1})$$

$$e_{i2} = MLP(v_i, v_2, v_{i2})$$

$$e_{i3} = MLP(v_i, v_3, v_{i3})$$



JetClass Dataset

Encoding a lot of information → a lot of parameters

- Critical to avoid overtraining and ensure generic embeddings

GPT-3: trained on ~200 billion words (estimated cost O(10) million dollars)

- Need huge statistics to train a jet transformer!
- ParT: Dedicated particle transformer, 2M parameters!

	Accuracy	# params	FLOPs
PFN	0.772	86.1 k	4.62 M
P-CNN	0.809	354 k	15.5 M
ParticleNet	0.844	370 k	540 M
ParT	**0.861**	2.14 M	340 M
ParT (plain)	0.849	2.13 M	260 M

JetClass Dataset

Encoding a lot of information → a lot of parameters

- Critical to avoid overtraining and ensure generic embeddings

GPT-3: trained on ~200 billion words (estimated cost O(10) million dollars)

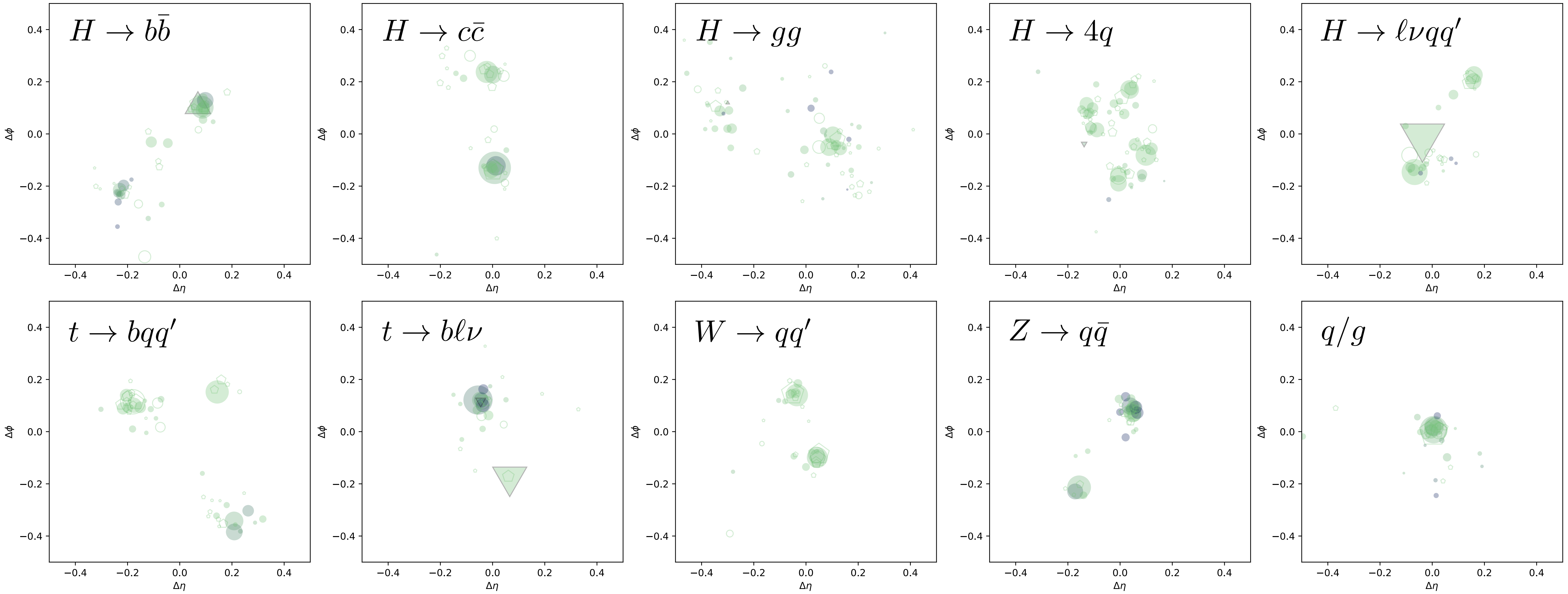
- Need huge statistics to train a jet transformer!
- ParT: Dedicated particle transformer, 2M parameters!

	Accuracy	# params	FLOPs
PFN	0.772	86.1 k	4.62 M
P-CNN	0.809	354 k	15.5 M
ParticleNet	0.844	370 k	540 M
ParT	**0.861**	2.14 M	340 M
ParT (plain)	0.849	2.13 M	260 M

New dedicated jet tagging dataset: Jetclass

- 10 types of jets
- 100M training
- 5M validation
- 20M test

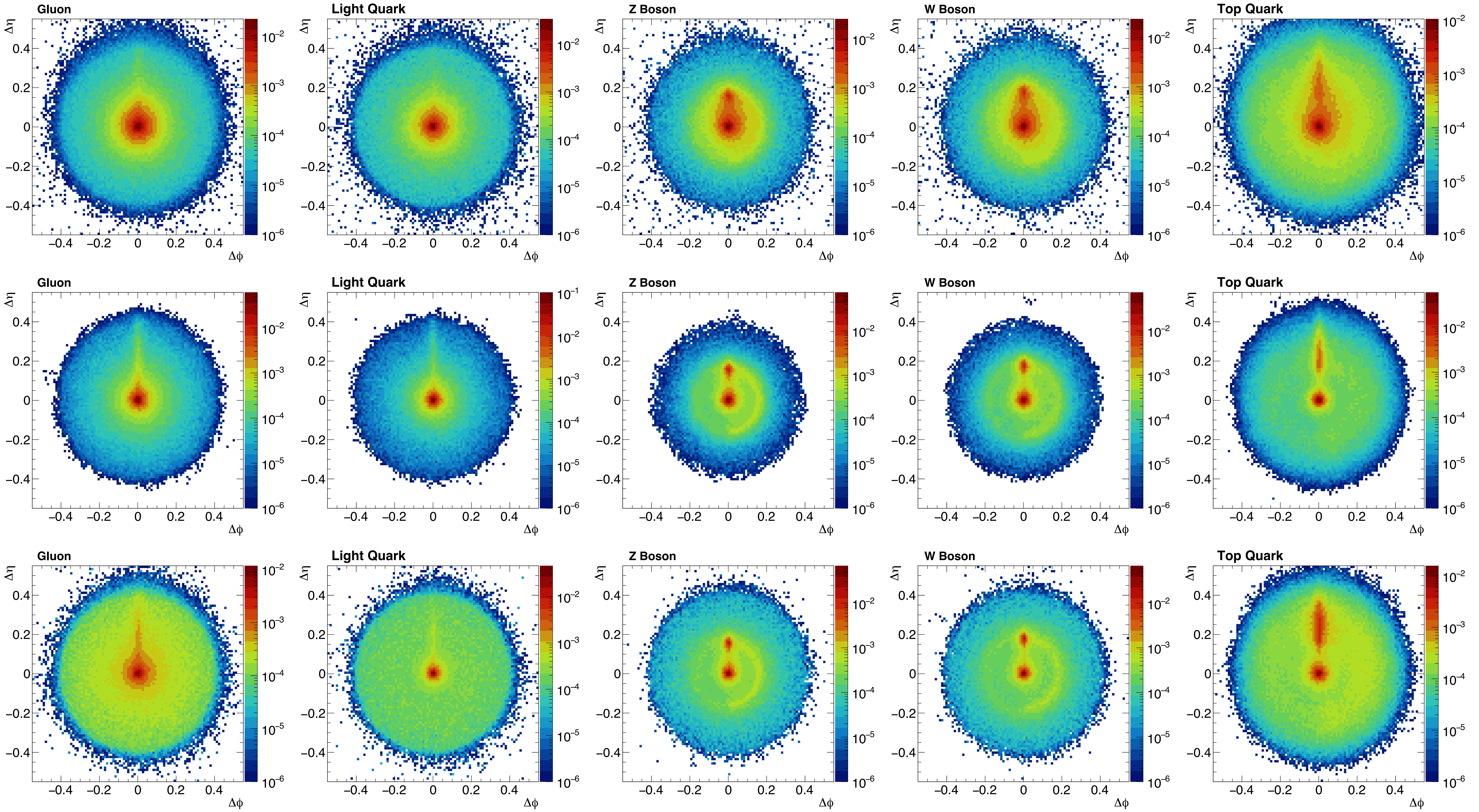
Extremely useful for benchmarking of new algorithms!



ABCNet:

Pixel intensity = particle importance w.r.t most energetic particle in jet, from attention weights

No substructure information given, learned through attention layers!



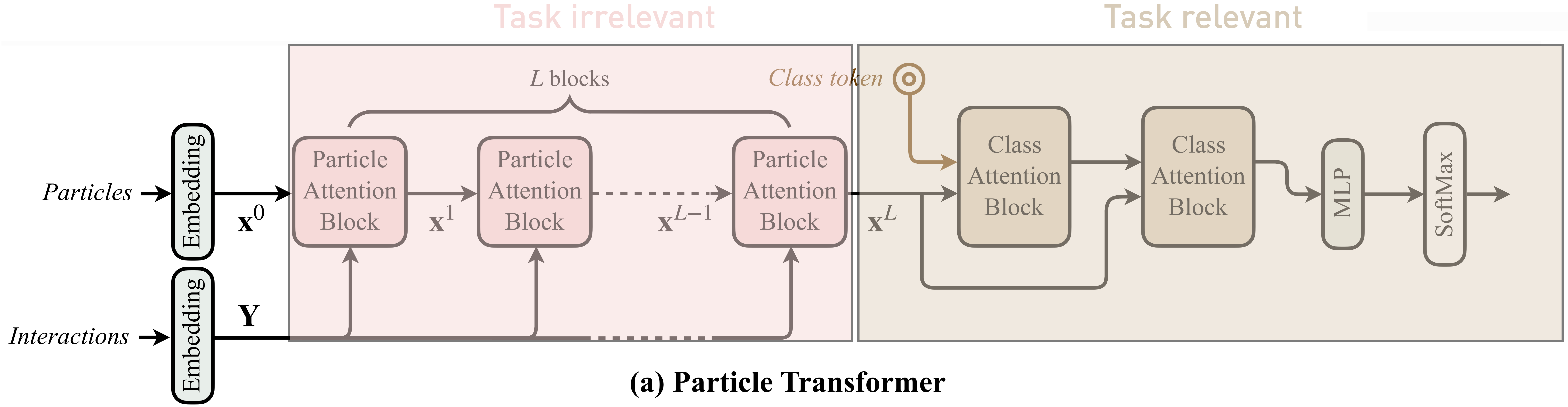
ParT

	Accuracy
P-CNN	0.930
PFN	—
ParticleNet	0.940
JEDI-net (w/ $\sum O$)	0.930
PCT	0.940
LGN	0.929
rPCN	—
ParT	0.940
ParT-f.t.	**0.944**

Particle Transformer (ParT): transformer designed for particle physics

Common in NLP: Use large pre-trained model, then fine-tune to specific task at hand!

- ParT: Transformer self-attention is task irrelevant embedding!



(a) Particle Transformer

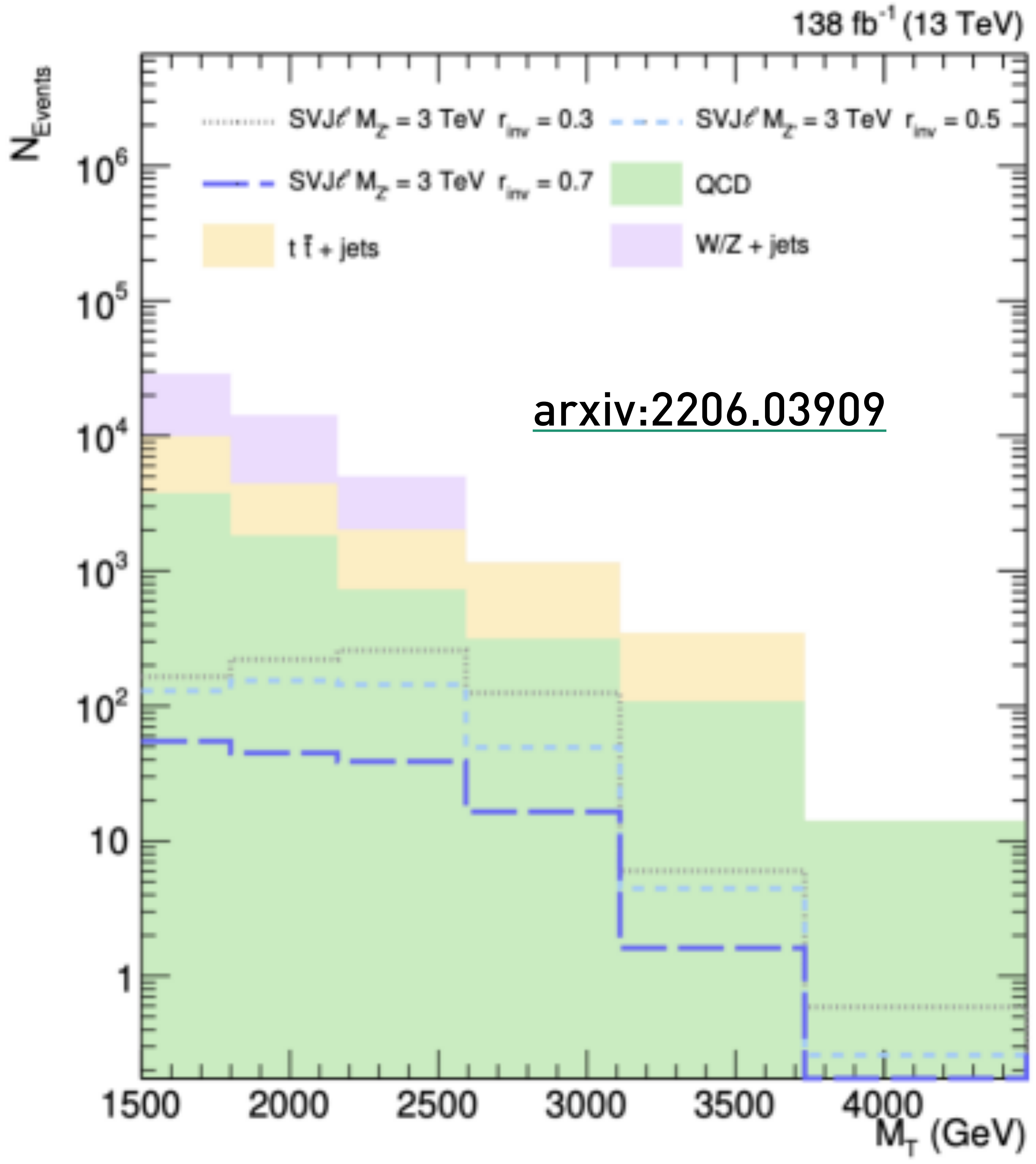
You can take ParT from here, fine-tune it and demonstrate for SVJ!

But we said we want model independence!

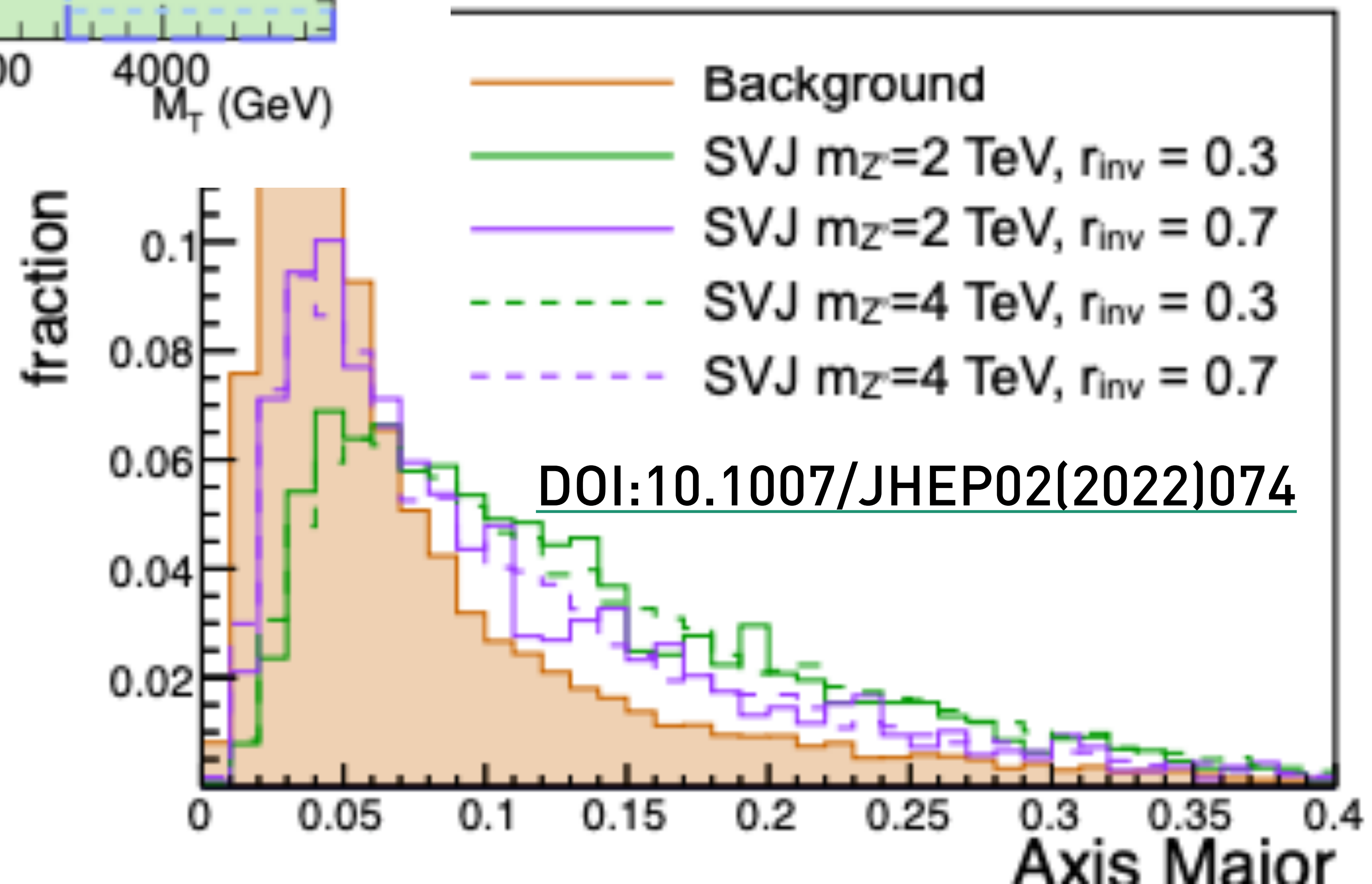
Weakly supervised

Yesterday, we said we want:

- Model independent taggers



arxiv:2206.03909

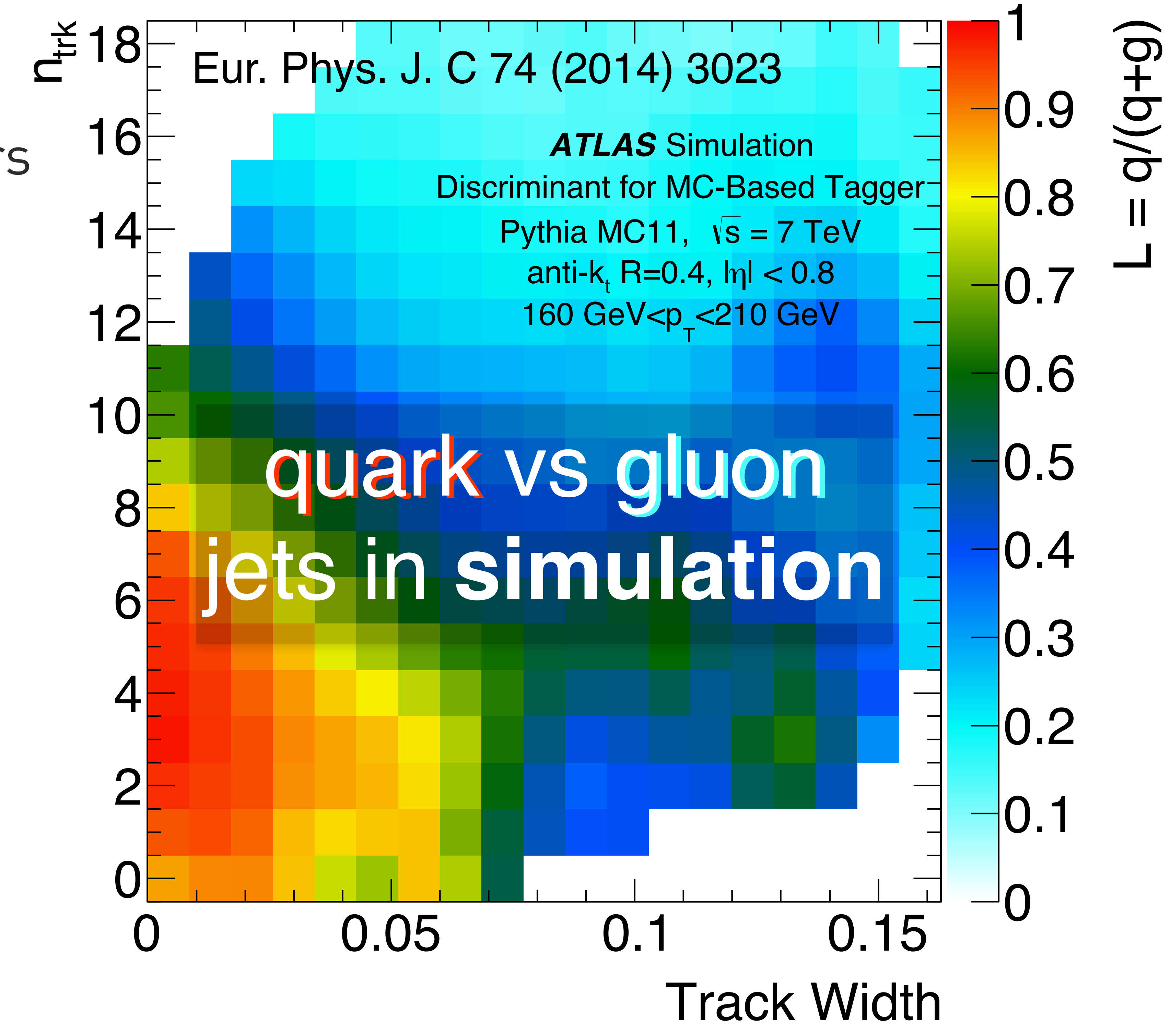


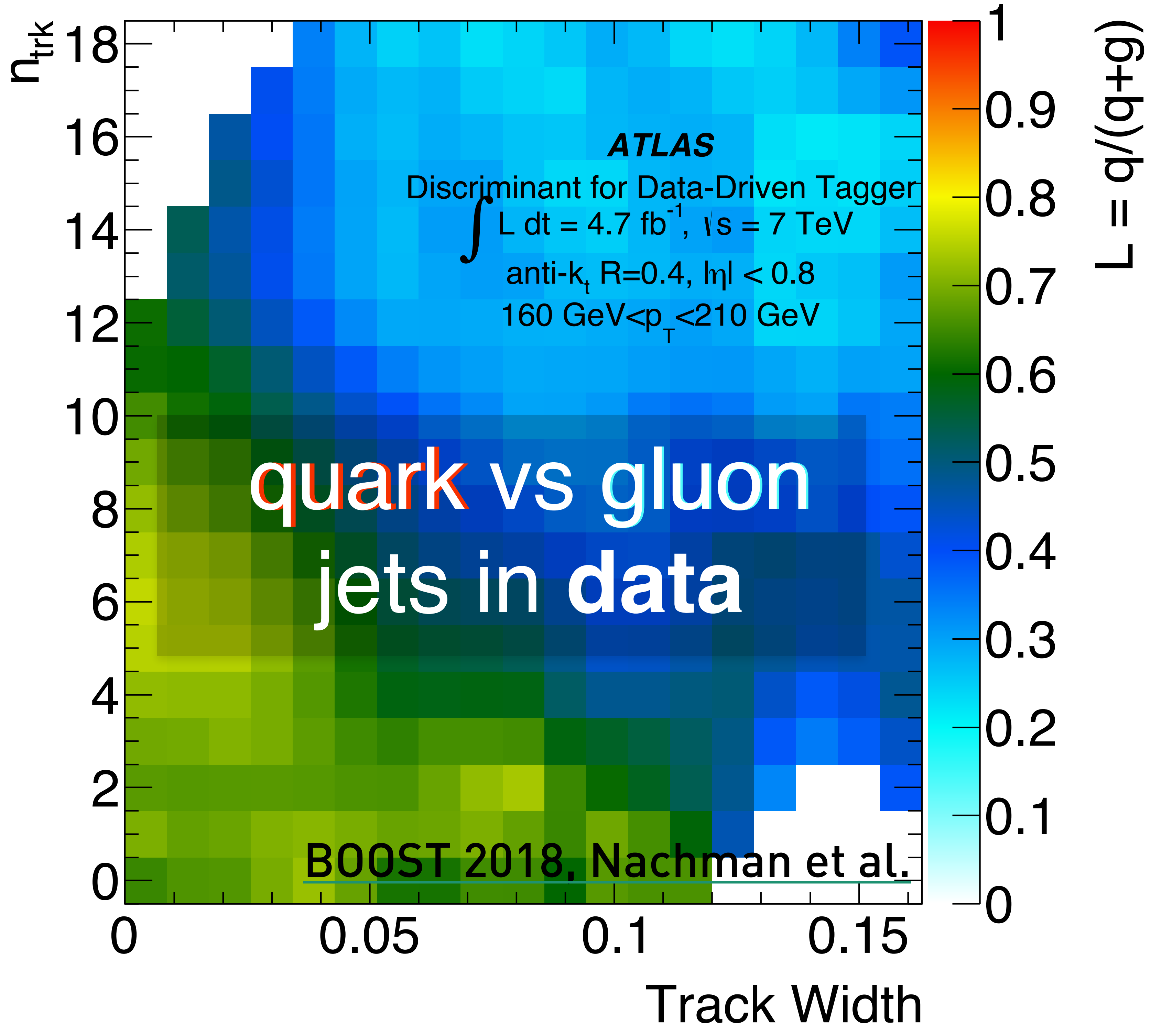
DOI:10.1007/JHEP02(2022)074

Weakly supervised

Yesterday, we said we want:

- Model independent taggers
- Simulation independent taggers



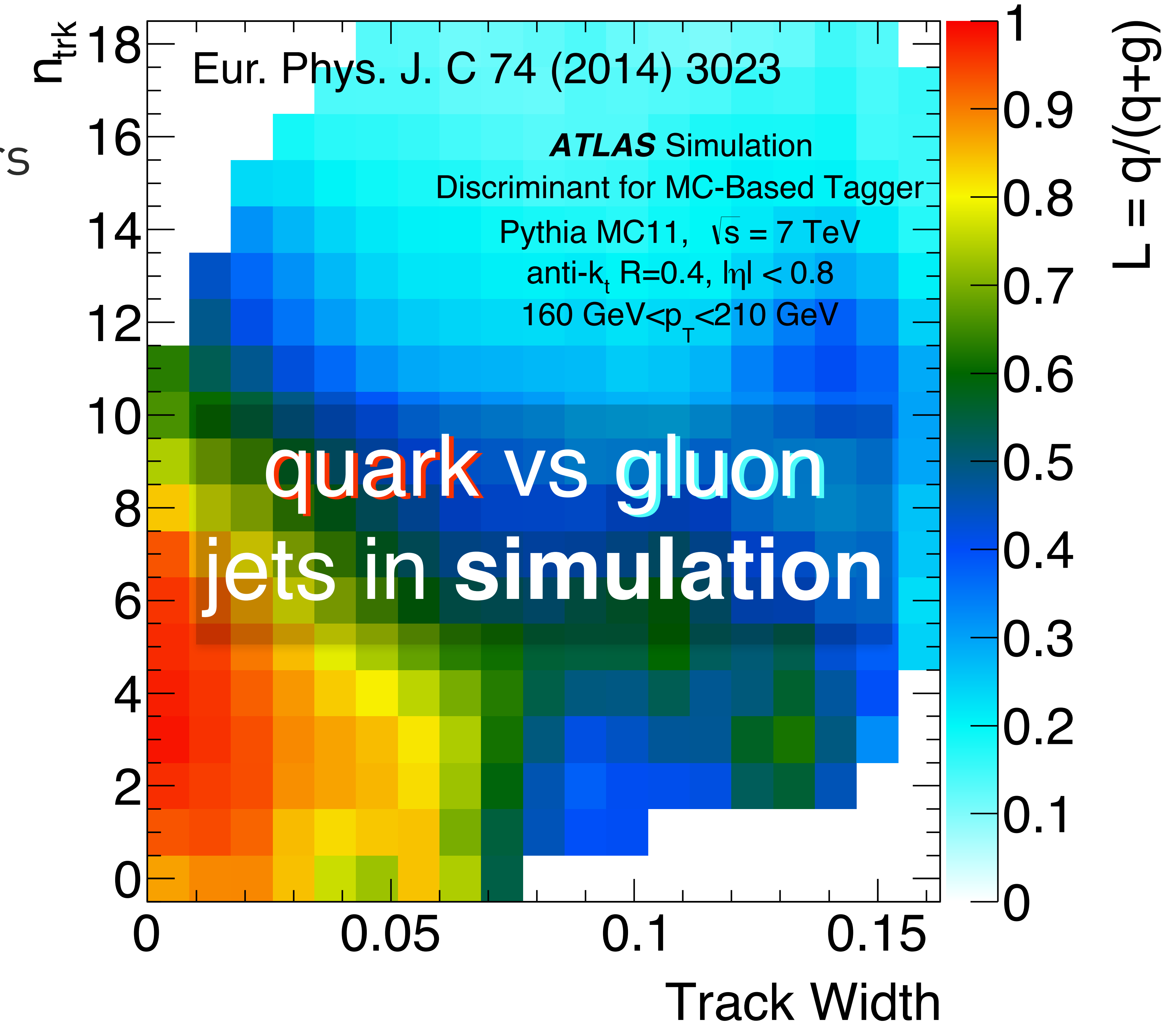


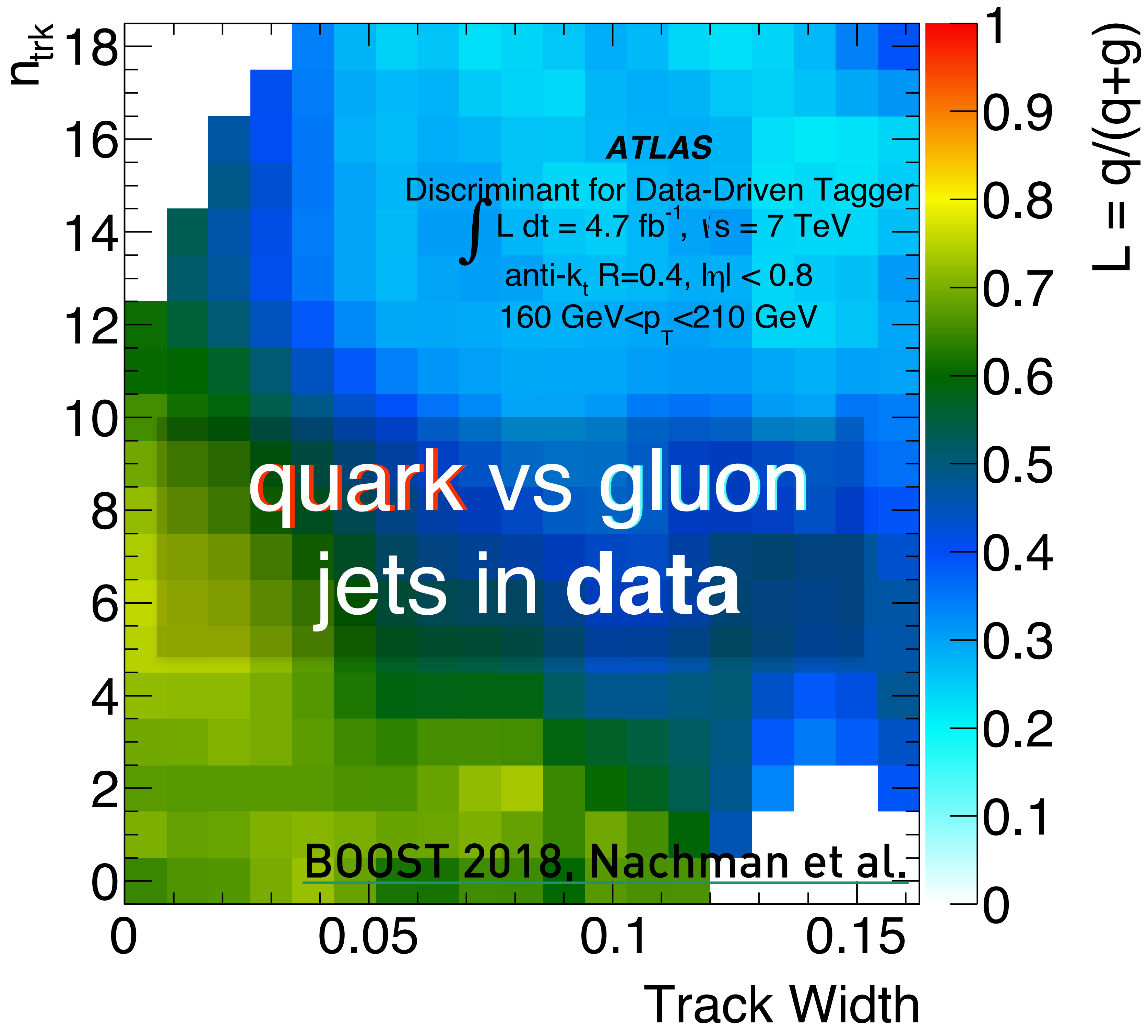
Weakly supervised

Yesterday, we said we want:

- Model independent taggers
- Simulation independent taggers
- Powerful taggers

Can we have it all?



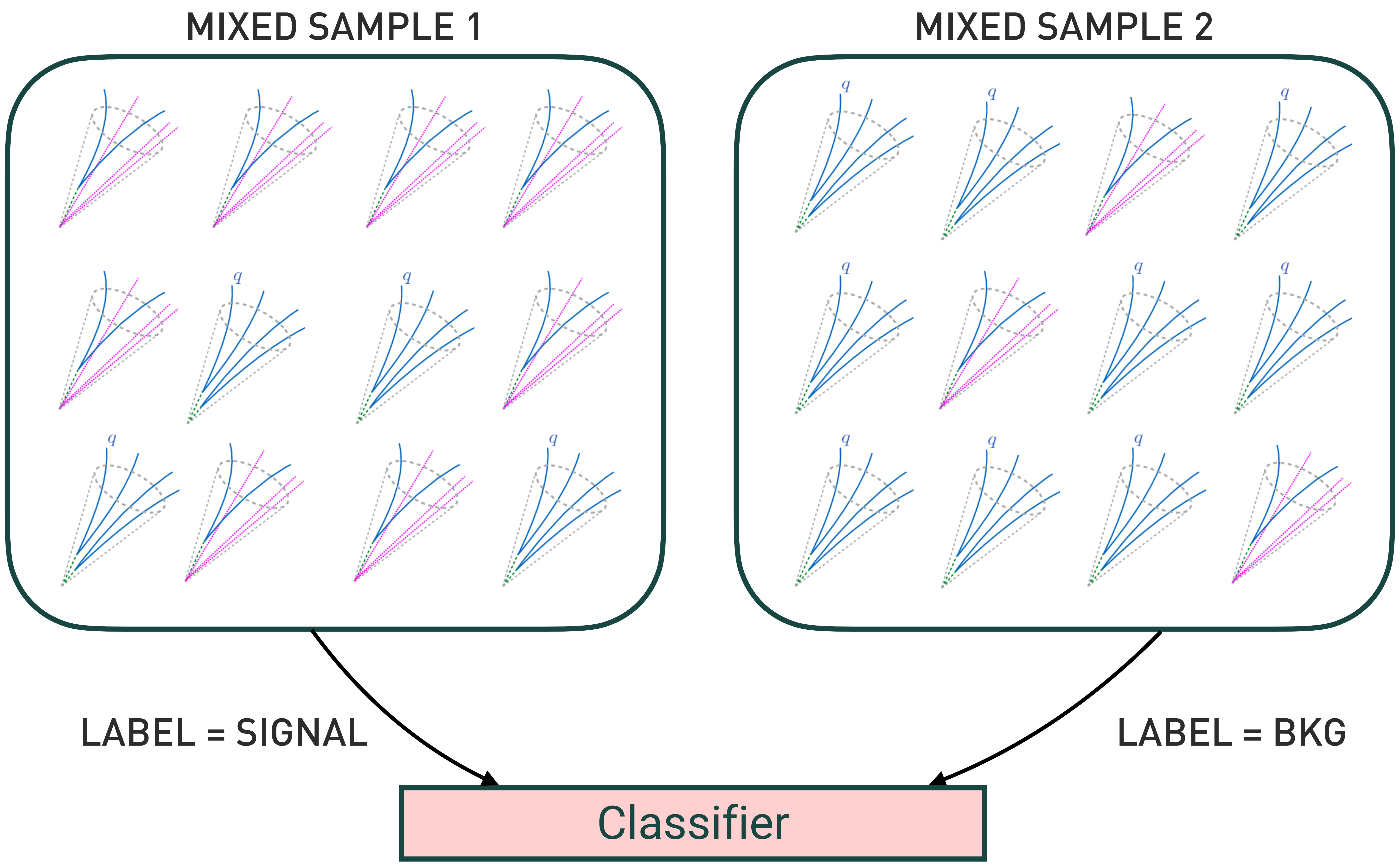


CWola

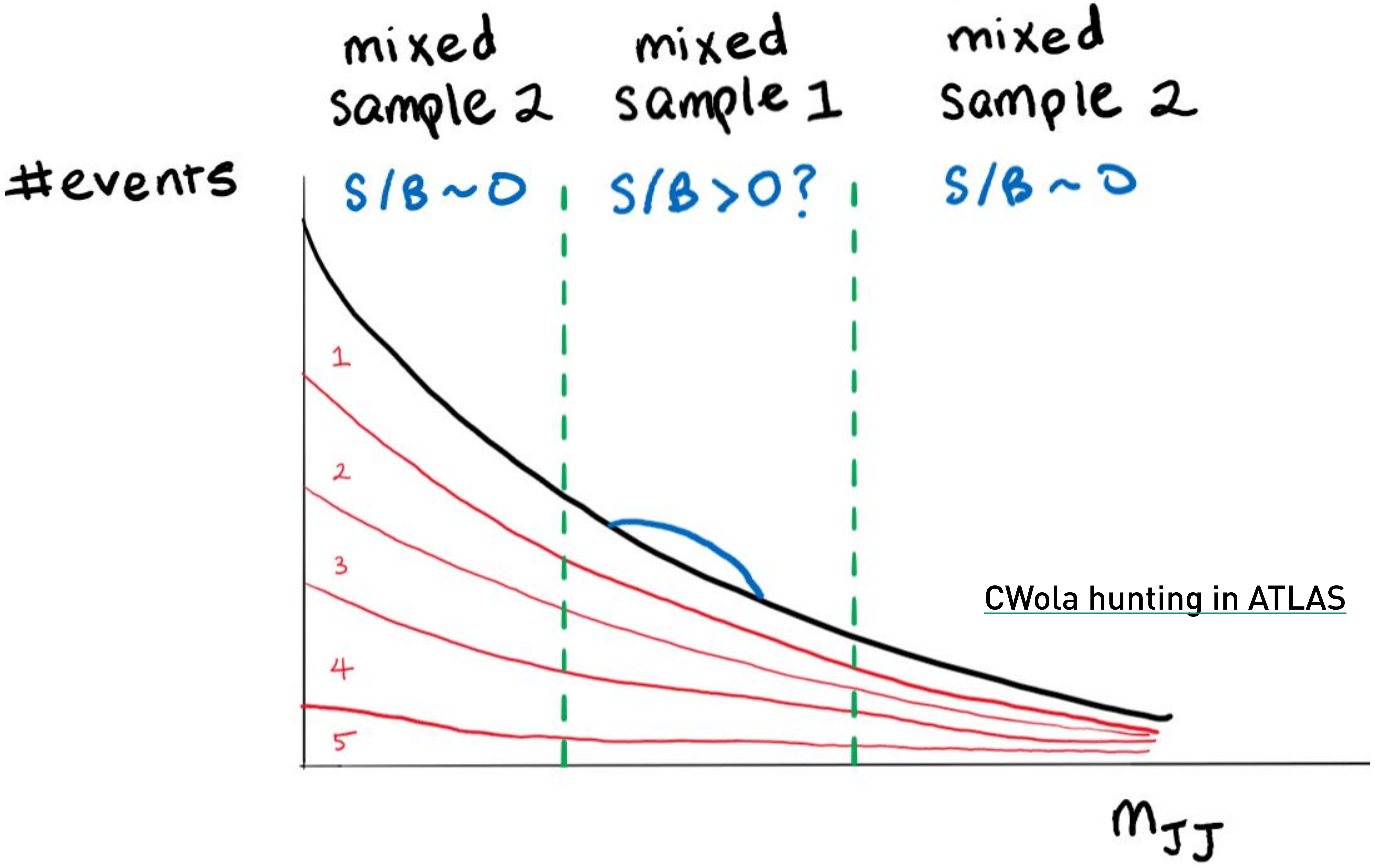
<u>Classification Without Labels</u>

- Design mixed samples in data s.t signal fraction $f_1 > f_2$
- Lemma: "Given mixed S+B samples SB and SR, optimal classifier trained to distinguish SB and SR is also optimal for distinguishing S from B"
- Higher signal fraction, better performance

How to design mixed samples for SVJ?

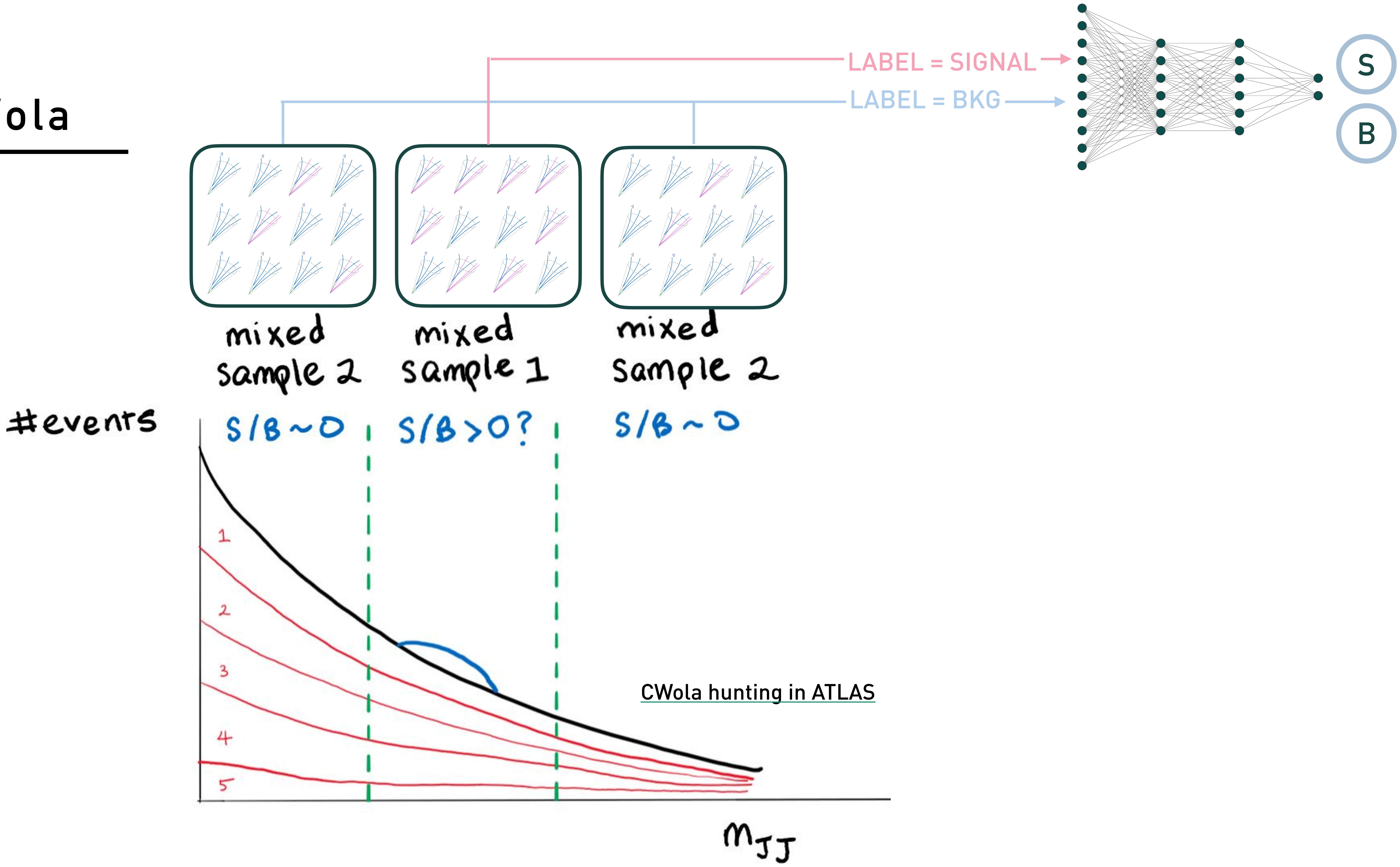


CWola



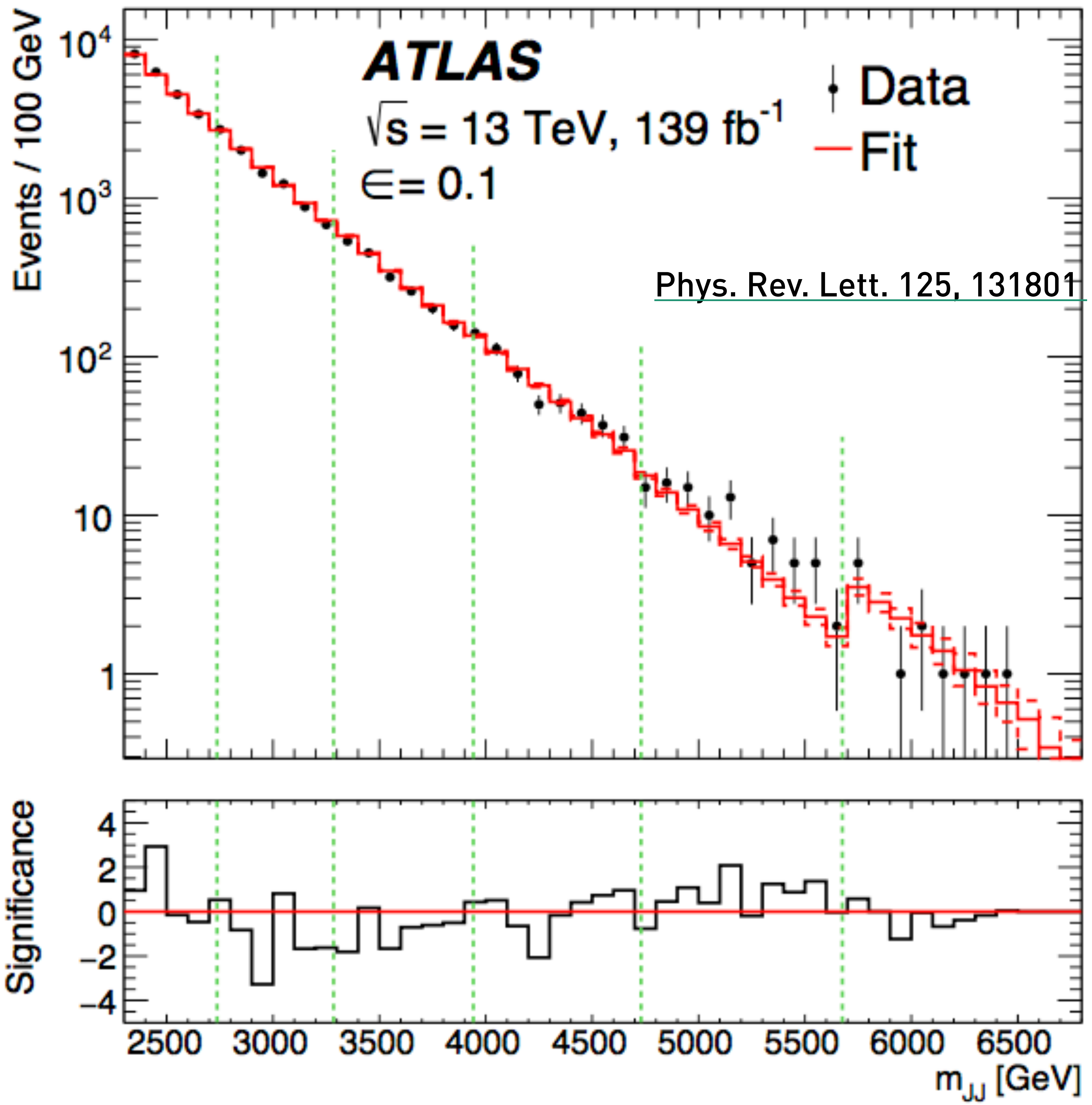
CWola hunting in ATLAS

CWola



CWola hunting in ATLAS

CWola



Phys. Rev. Lett. 125, 131801

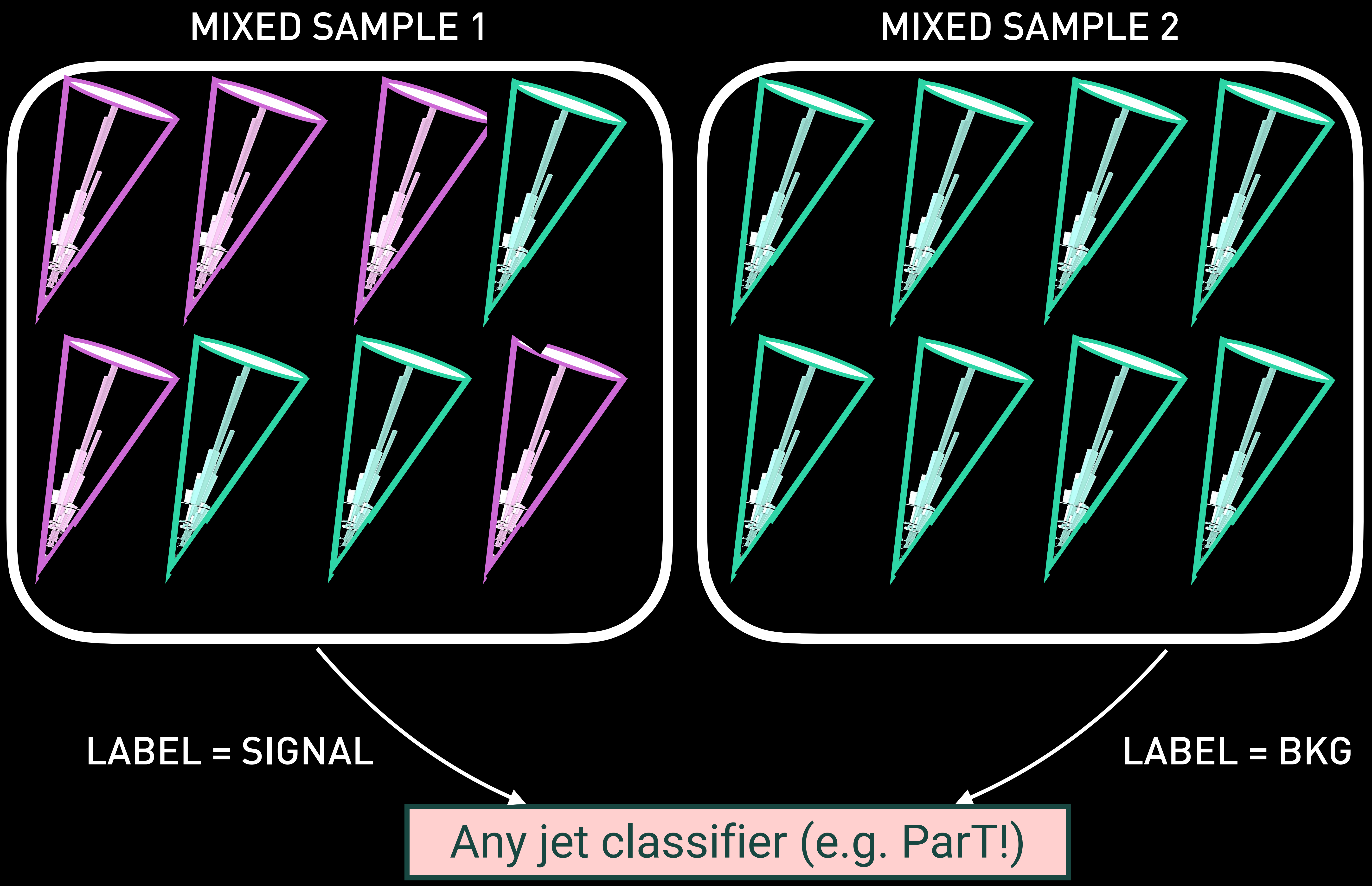

SIGNAL
$0<r_{inv}<1$
$r_{inv} = 1$
BACKGROUND
q/g jet
Z(vv)

0<r_inv<1
r_inv = 1
VS.
q/g jet
Z(νν)

"Boosting mono-jet searches with model-agnostic machine learning" Kraemer et al.

THESE JETS ARE FROM A
MET+JET TOPOLOGY
→ SVJ SIGNAL REGION

THESE JETS ARE FROM A
$\ell\ell$+JET TOPOLOGY
→ SVJ SIGNAL IS NOT EXPECTED HERE



f^{SR}	$n^{\mathrm{SR}}_{\mathrm{exp}}$	n^{SR}	$n^{\mathrm{SR}}_{\mathrm{A}}$	$n^{\mathrm{SR}}_{\mathrm{B}}$	$(n^{\mathrm{SR}} - n^{\mathrm{SR}}_{\mathrm{exp}})/\sqrt{2\,n^{\mathrm{SR}}_{\mathrm{exp}}}$
0%	1000	1048	0	1048	1.07
0.2%	1000	1065	47	1018	1.45
0.4%	1000	1107	100	1007	2.39
0.5%	1000	1175	184	991	3.91
0.6%	1000	1306	247	1059	6.84
0.7%	1000	1389	367	1022	8.70
0.8%	1000	1500	419	1081	11.18
1%	1000	1666	625	1041	14.89
2%	1000	2357	1392	965	30.34
4%	1000	4182	3269	913	71.15

Fraction of signal in SR

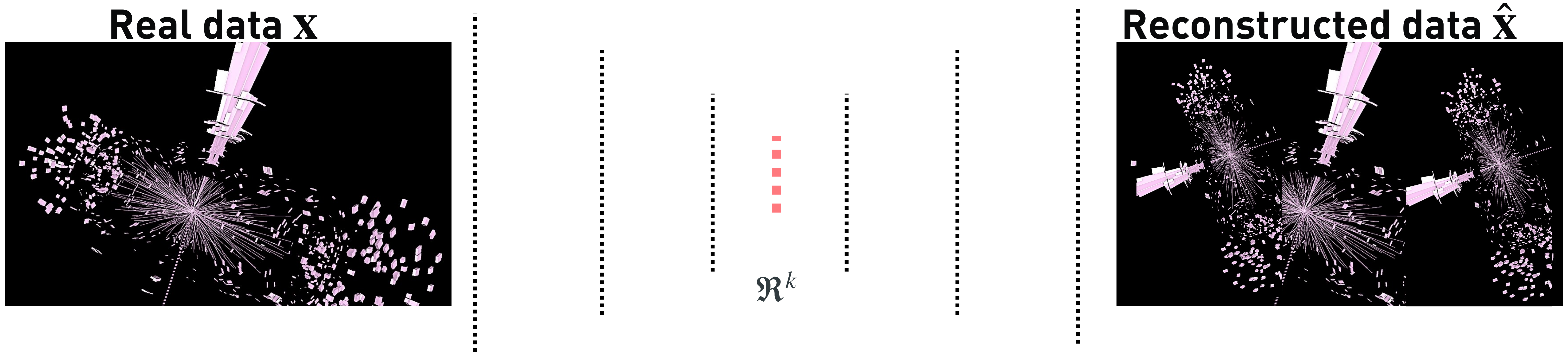
Statistical significance of possible discovery

Still consistent with constraints from ATLAS mono-jet search!

MY WISHLIST?
FIRST MODEL-INDEPENDENT SEARCH WITH PRE-TRAINED TRANSFORMER, FINE-TUNED ON CWOLA MIXED SAMPLES FOR SVJ SEARCHES!

Unsupervised

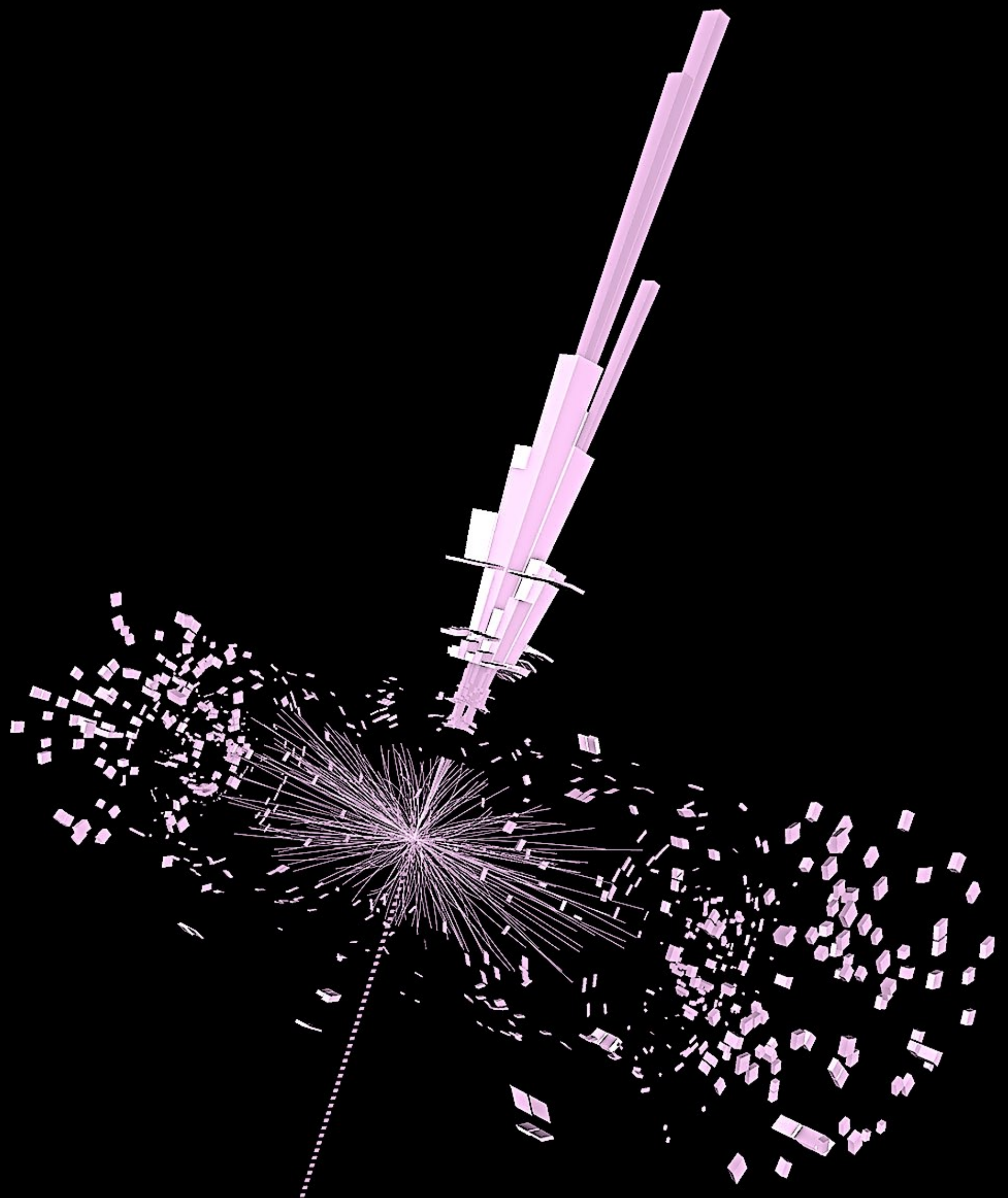
You have already heard about autoencoders and anomaly detection from Barry!

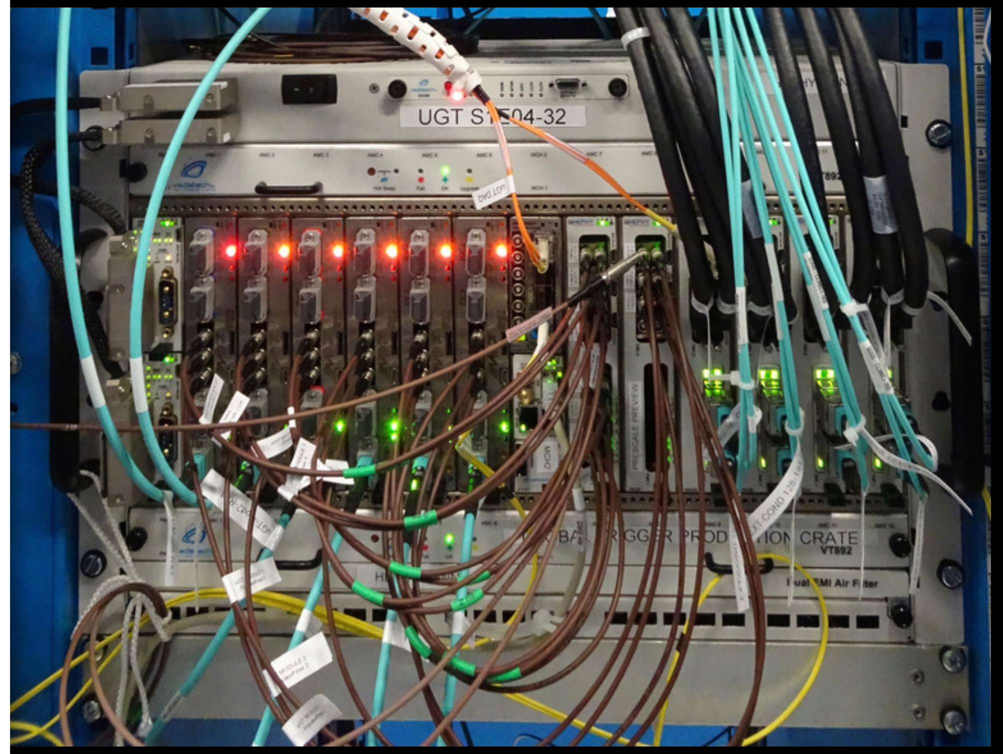


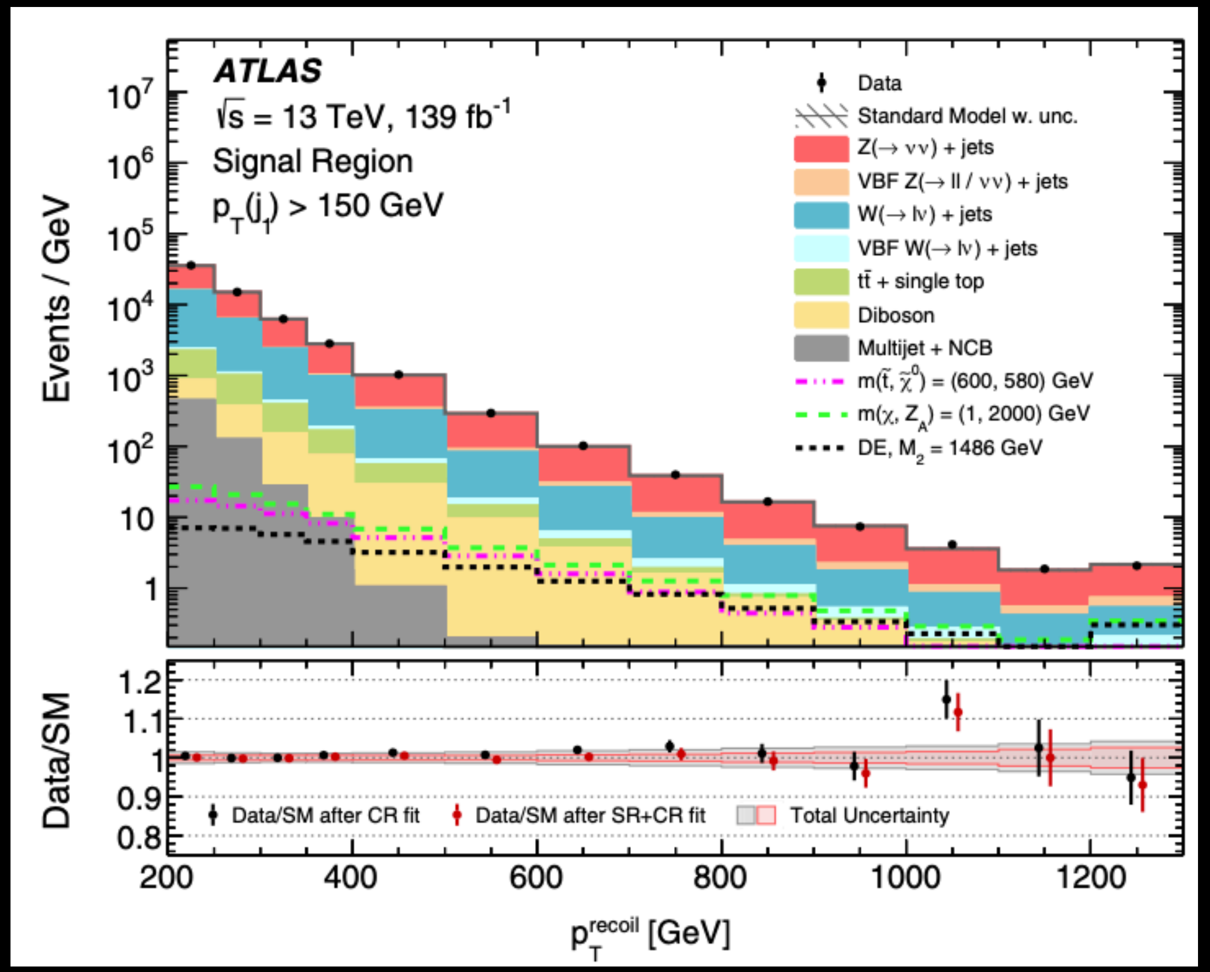
Where are we excited to try these out in experiment?

Event filtering systems

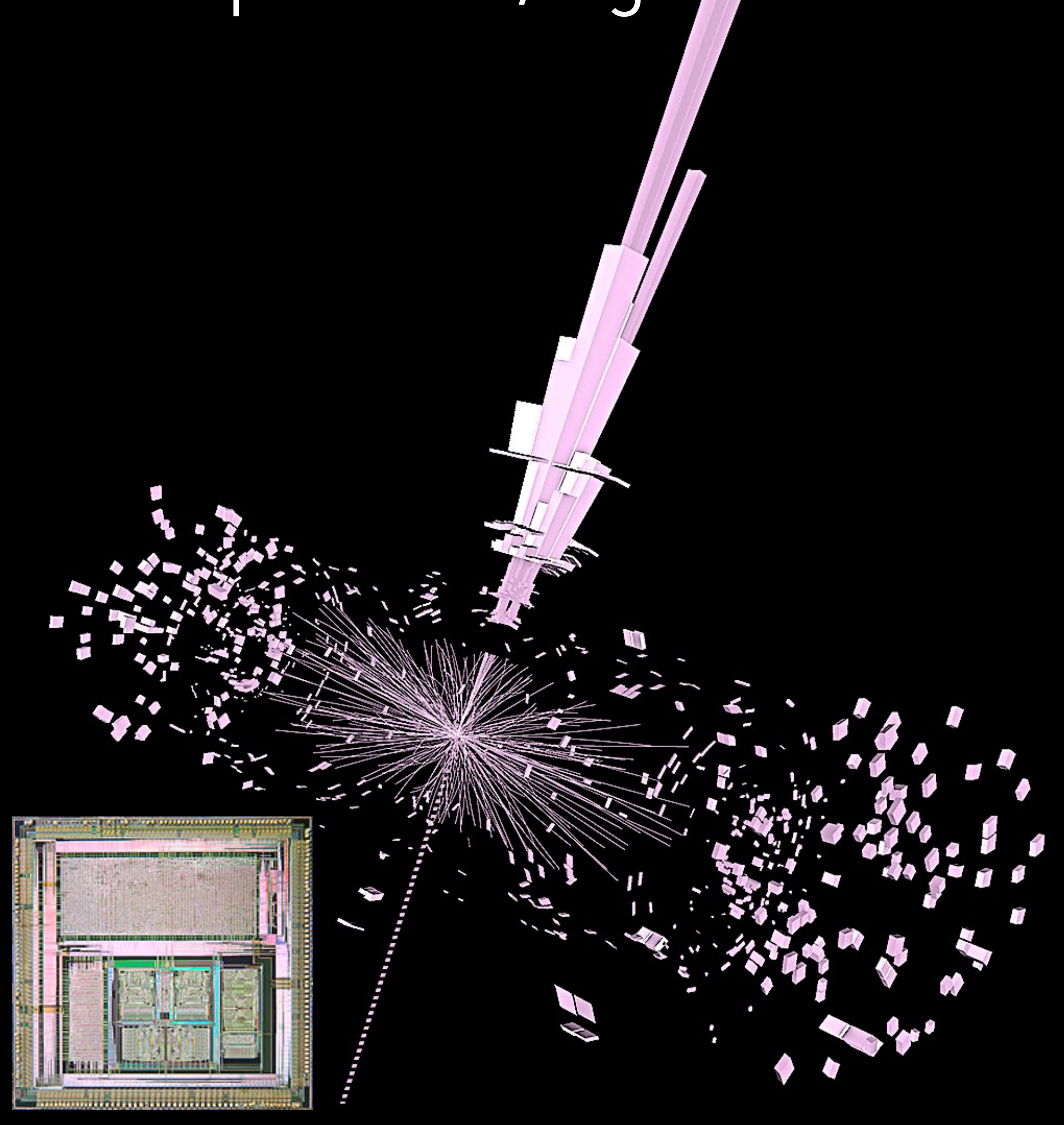
- Maximize SVJ signal acceptance through novel triggers!





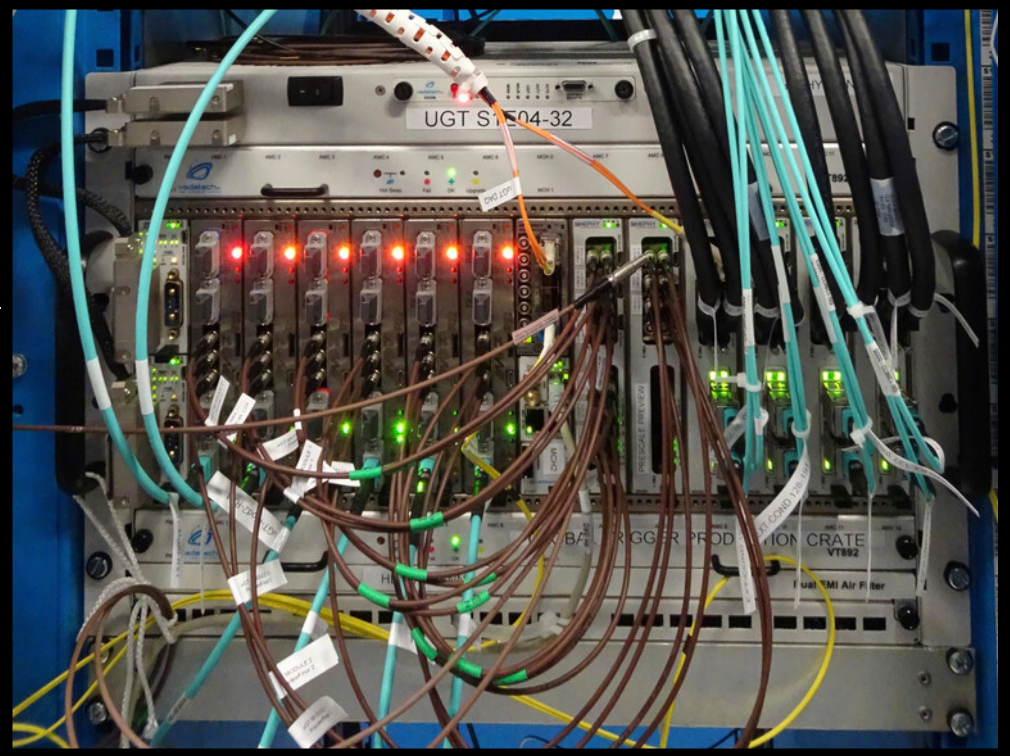


Detector
- Collisions every 25 ns
- Up to1 PB/s generated



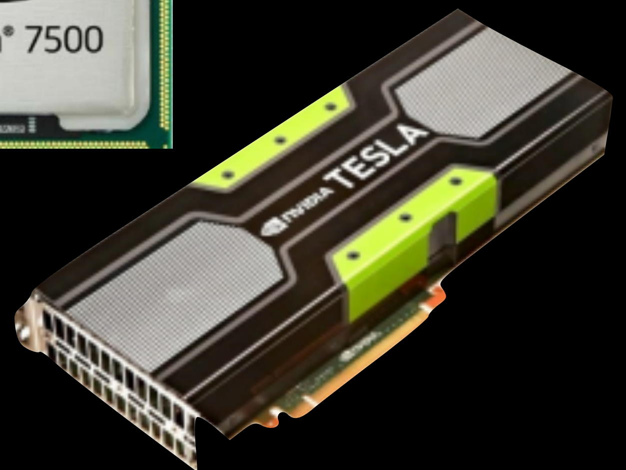
40 MHz
~PB/s

Level-1 hardware trigger
- Select __~2%__ of most interesting events
- O(1) μs latency



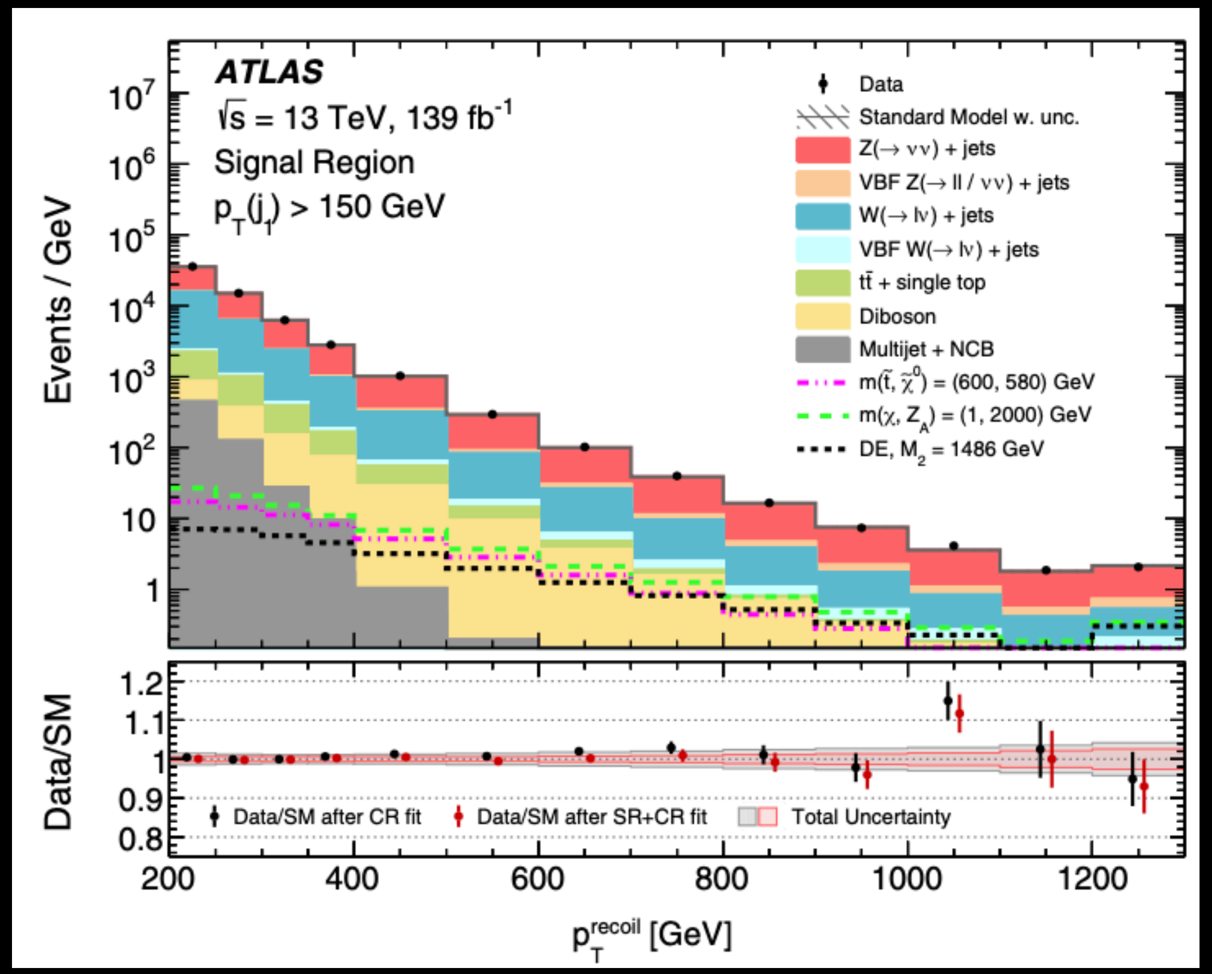
750 kHz
~TB/s

High Level Trigger CPU farm
- Select 1% of events from L1



7.5 kHz
~GB/s

Offline reconstruction and analysis

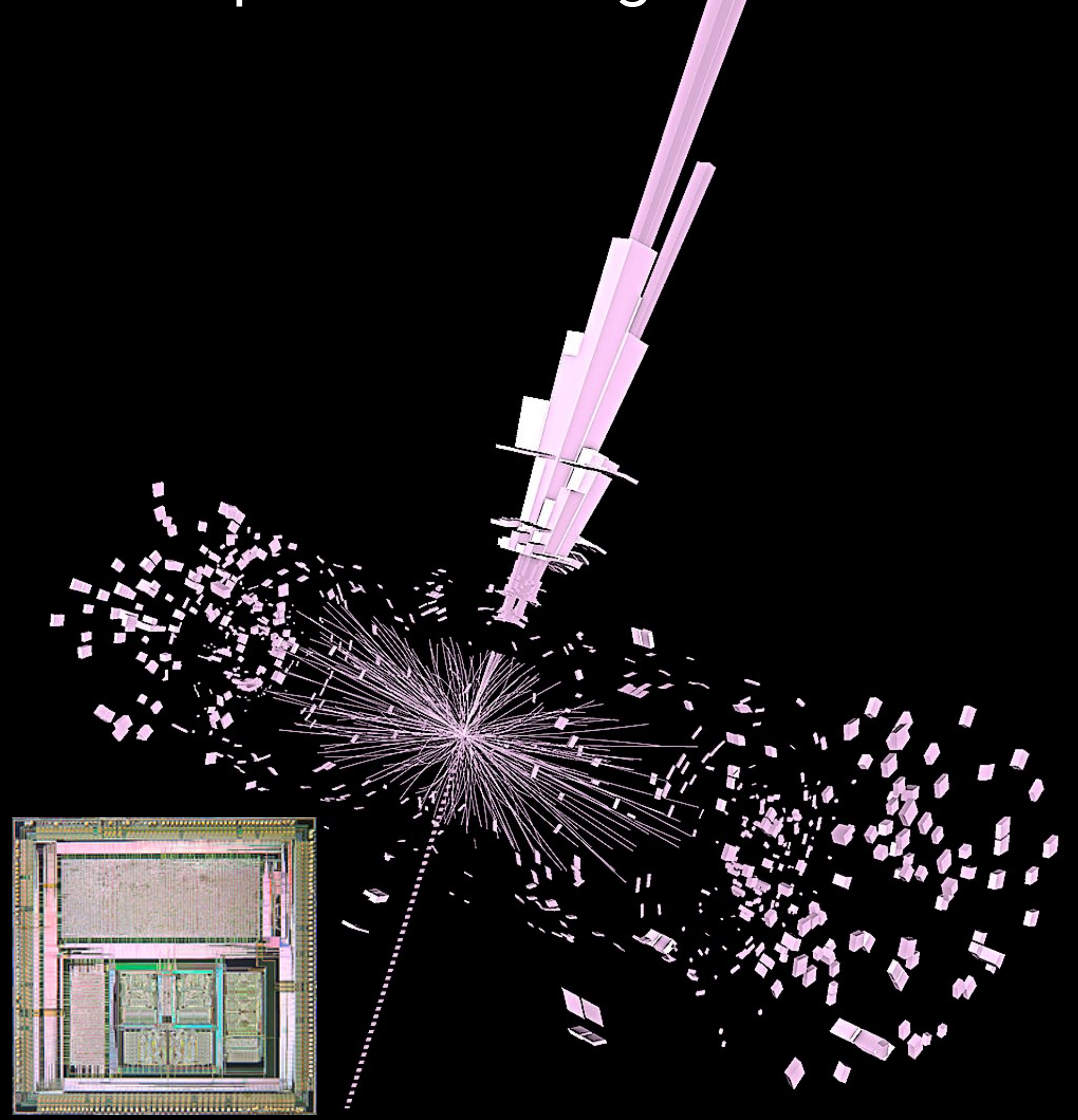


Do physics with 0.018% of collision events, the rest is discarded!

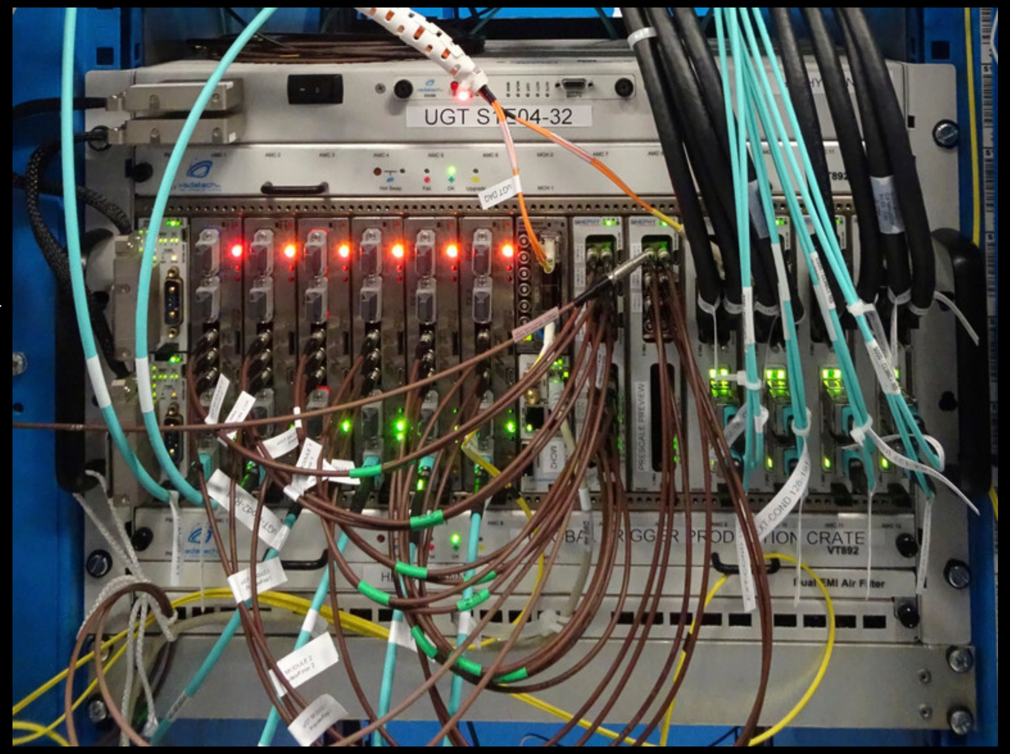
Level-1 hardware trigger
- Select __~2%__ of most interesting events
- O(1) μs latency

High Level Trigger CPU farm
- Select 1% of events from L1

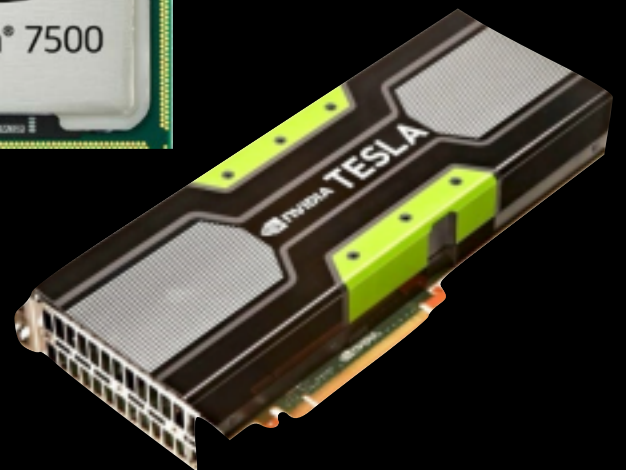
Detector
- Collisions every 25 ns
- Up to1 PB/s generated



40 MHz
~PB/s

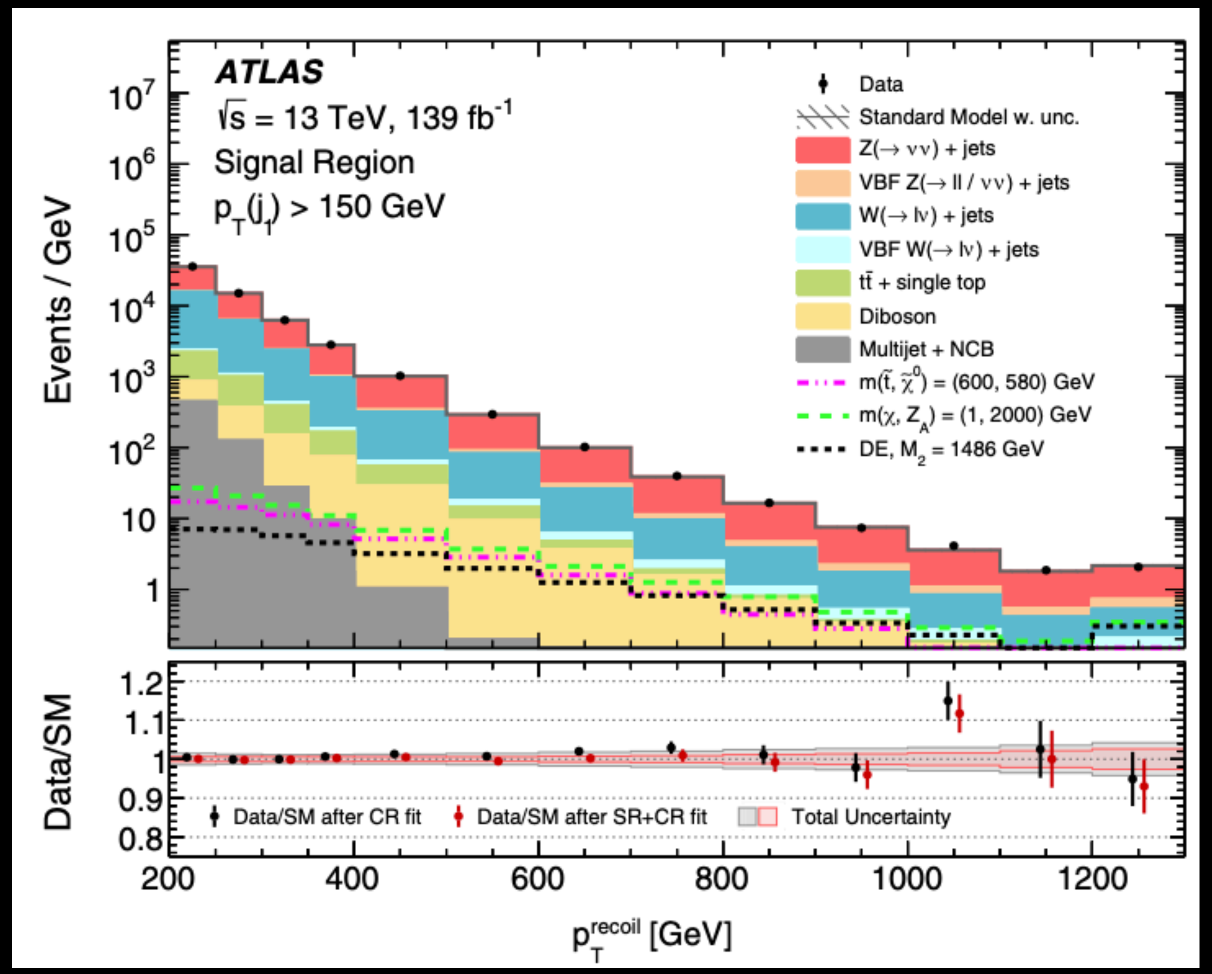


750 kHz
~TB/s



7.5 kHz
~GB/s

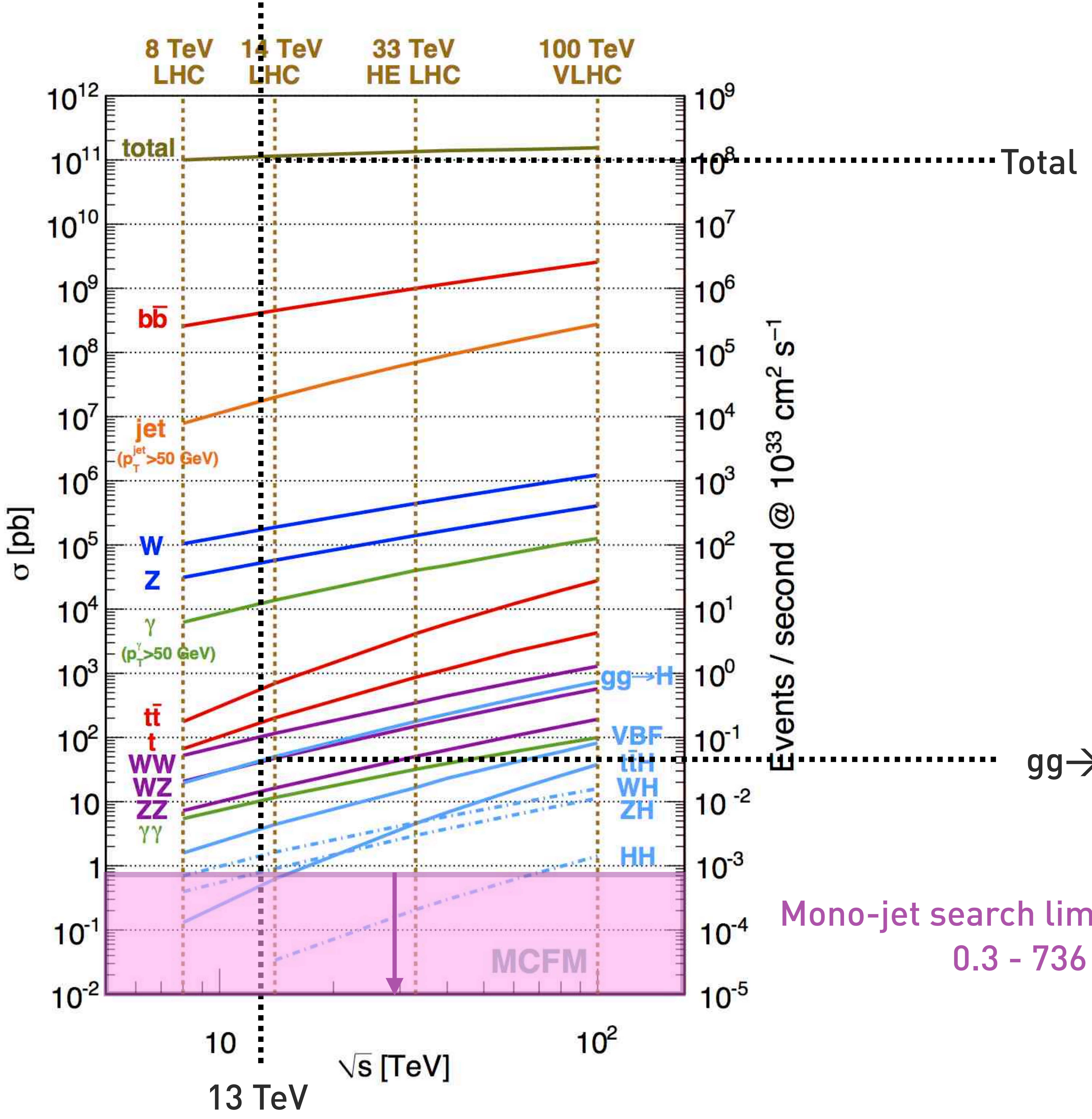
Offline reconstruction and analysis




CMS
CMS Experiment at the LHC, CERN
Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)
Run / Event: 151076 / 1405388
~1 billion collisions per second
~1 PB of data per second
X
Y
Z

New Physics is produced 1 in 10^{12}

Saving all collisions not useful
(even if we could)!



Mono-jet search limits at 95% CL
0.3 - 736 fb

Do physics with 0.018% of collision events, the rest is discarded!

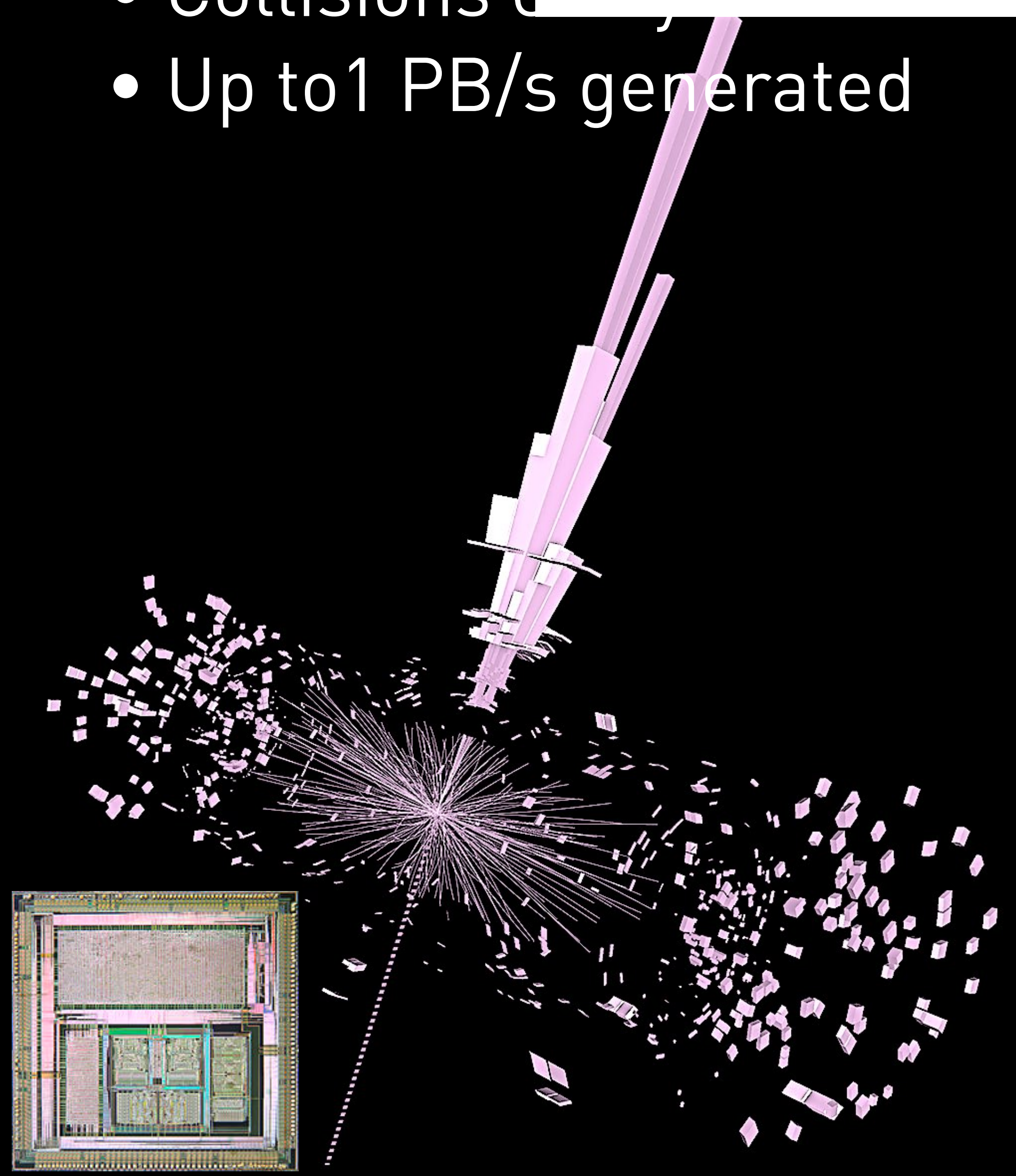
Level-1 hardware trigger
- Select **~2%** of most interesting events
- O(1) μs latency

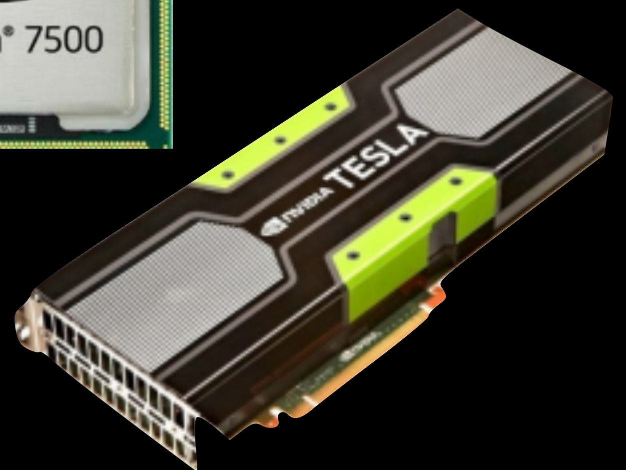
High Level Trigger CPU farm
- Select 1% of events from L1

40 MHz **750 kHz** **7.5 kHz**

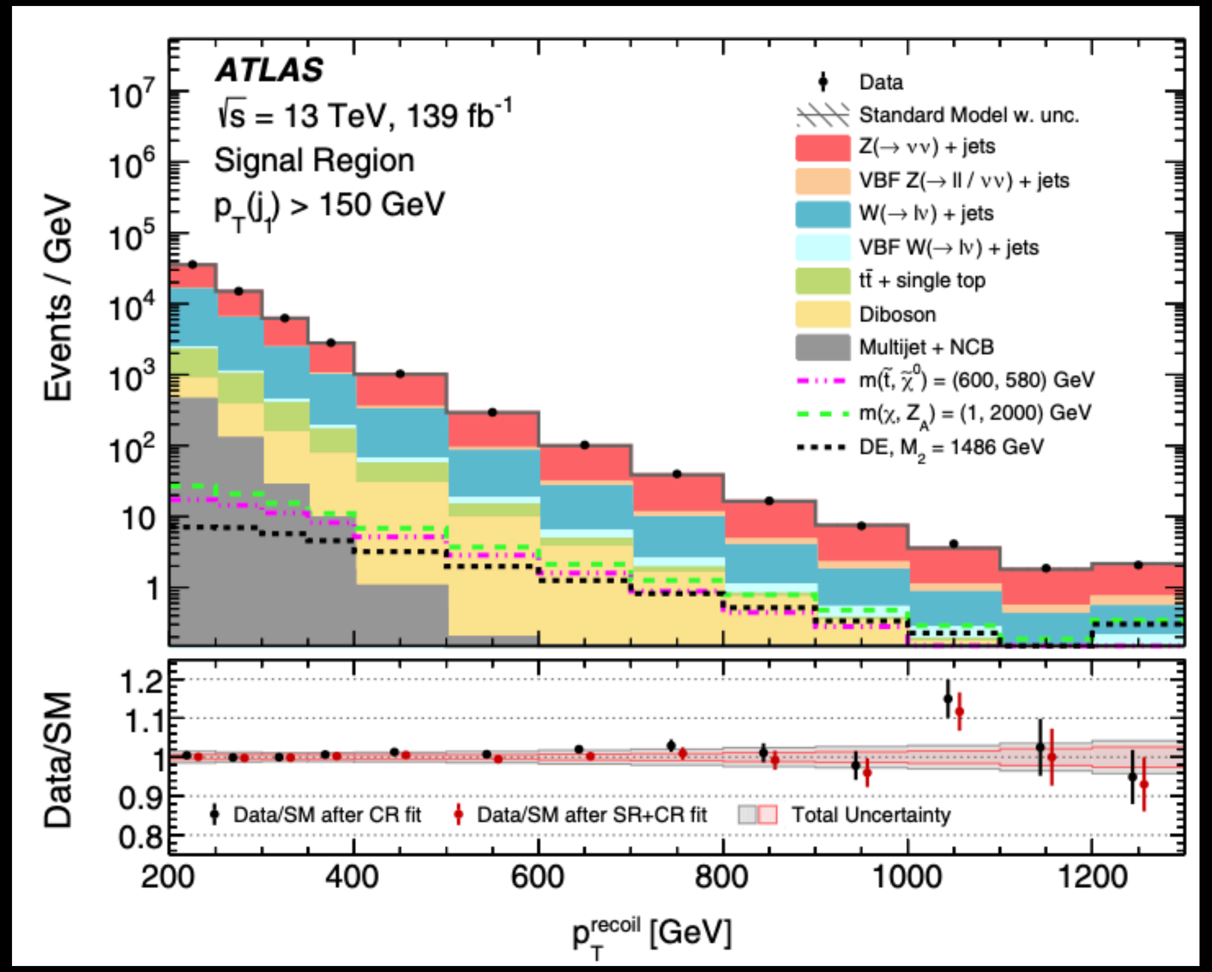
ARE WE LOOSING SVJ AT L1?

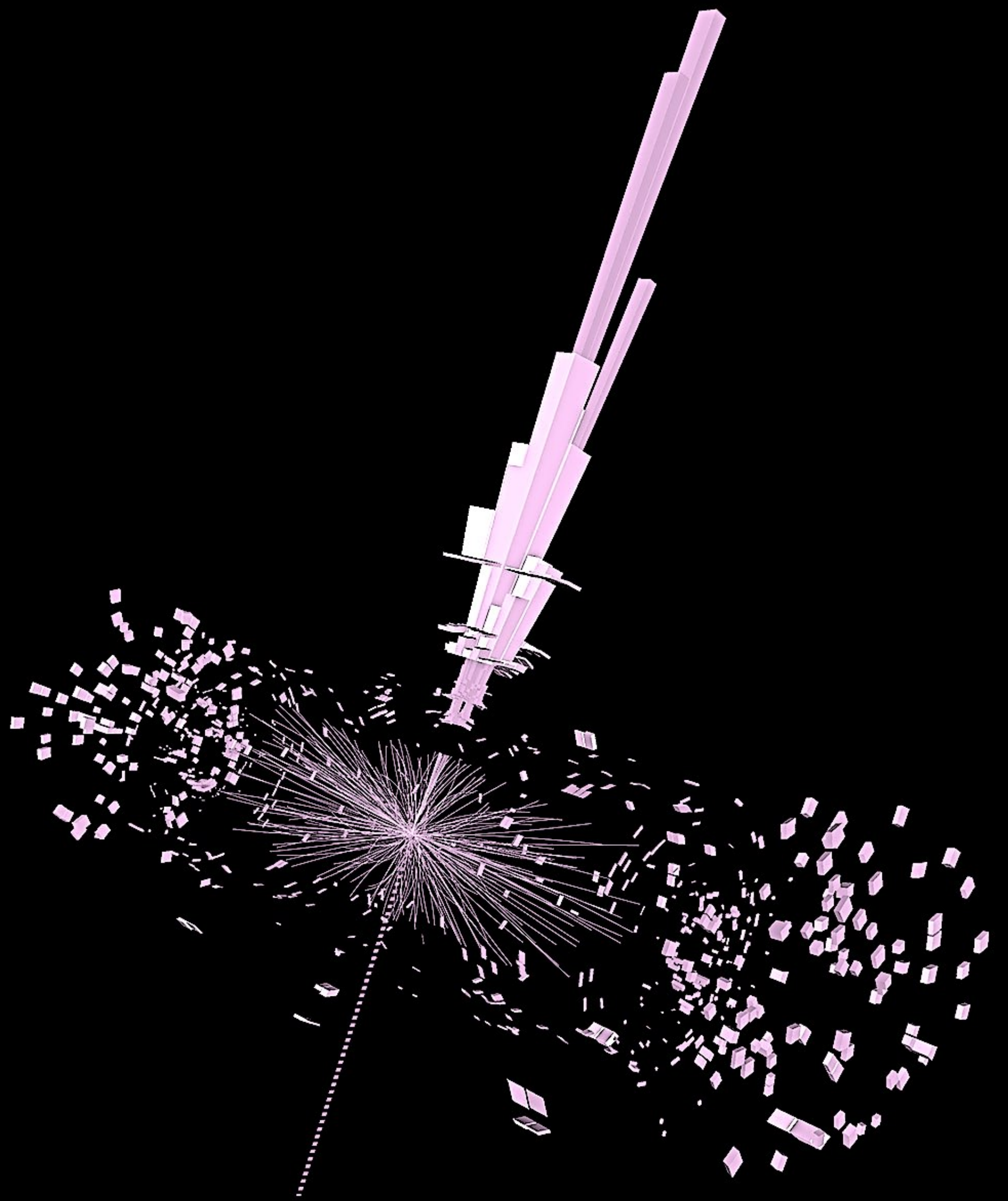
Detector
- Collisions e[...]y [...]
- Up to1 PB/s generated

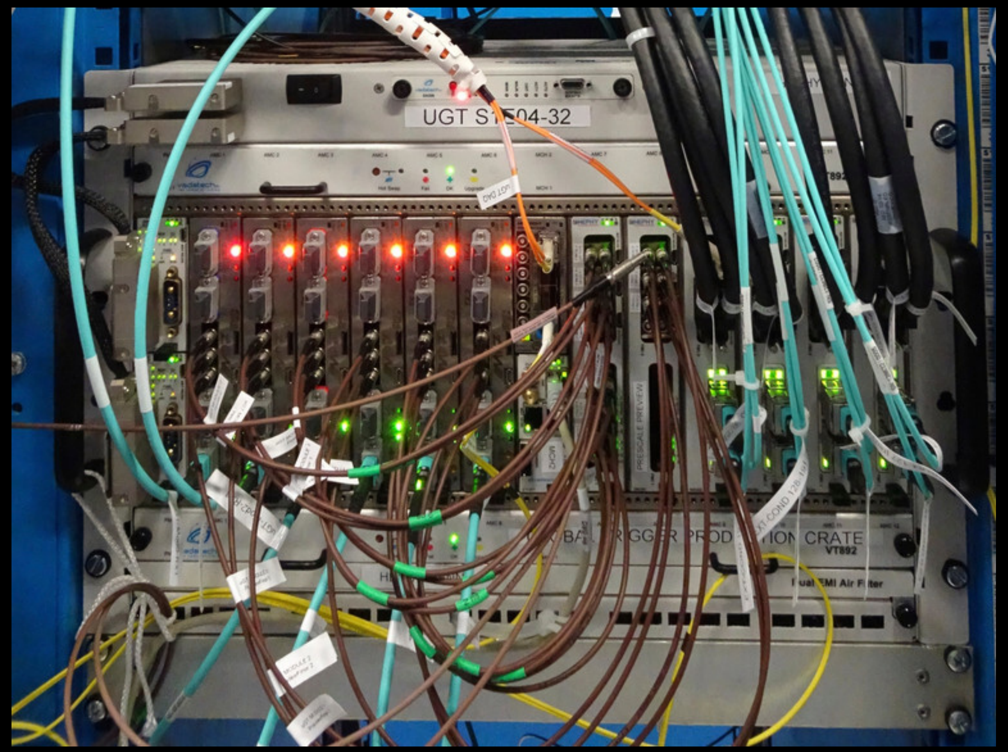




Offline reconstruction and storage



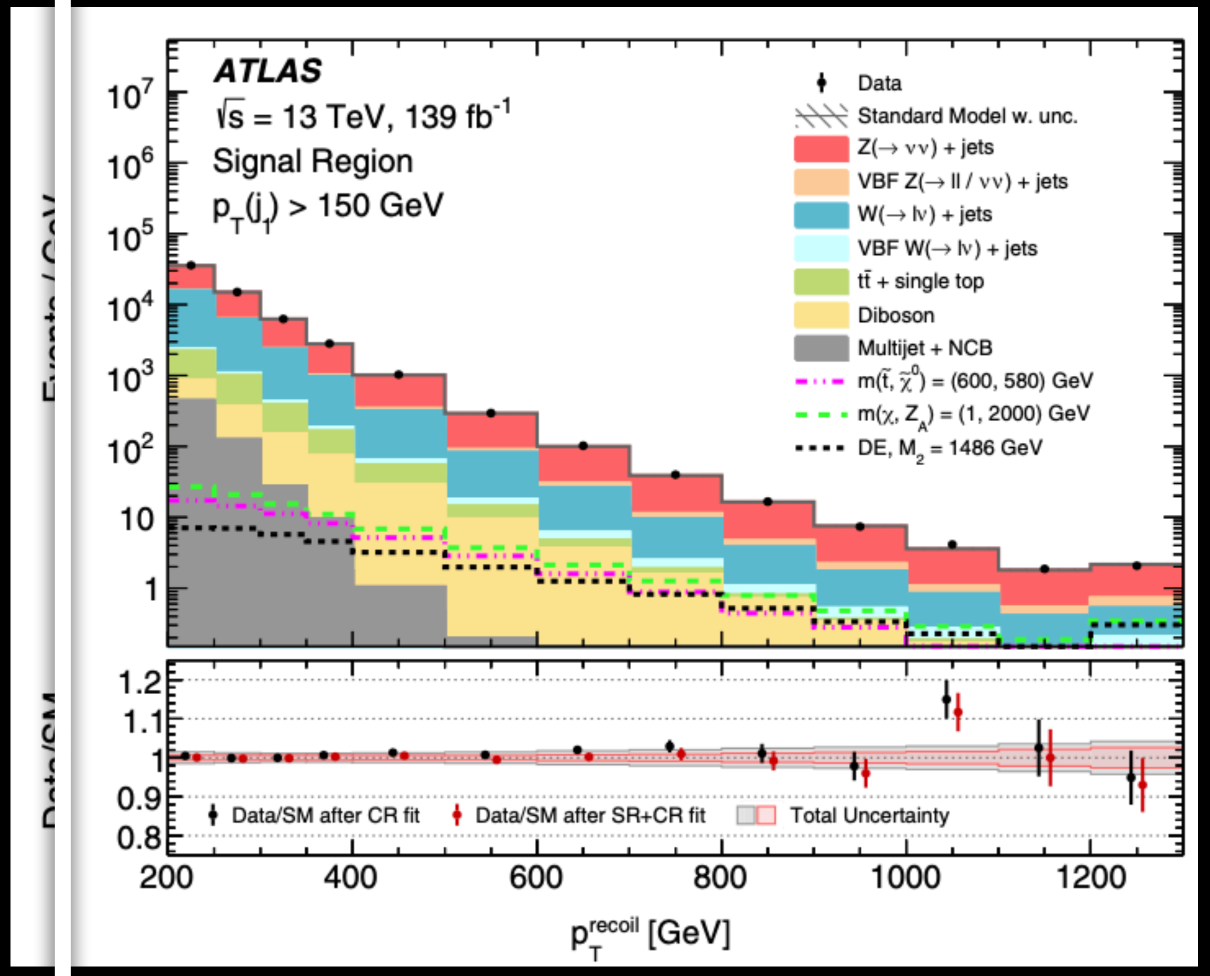


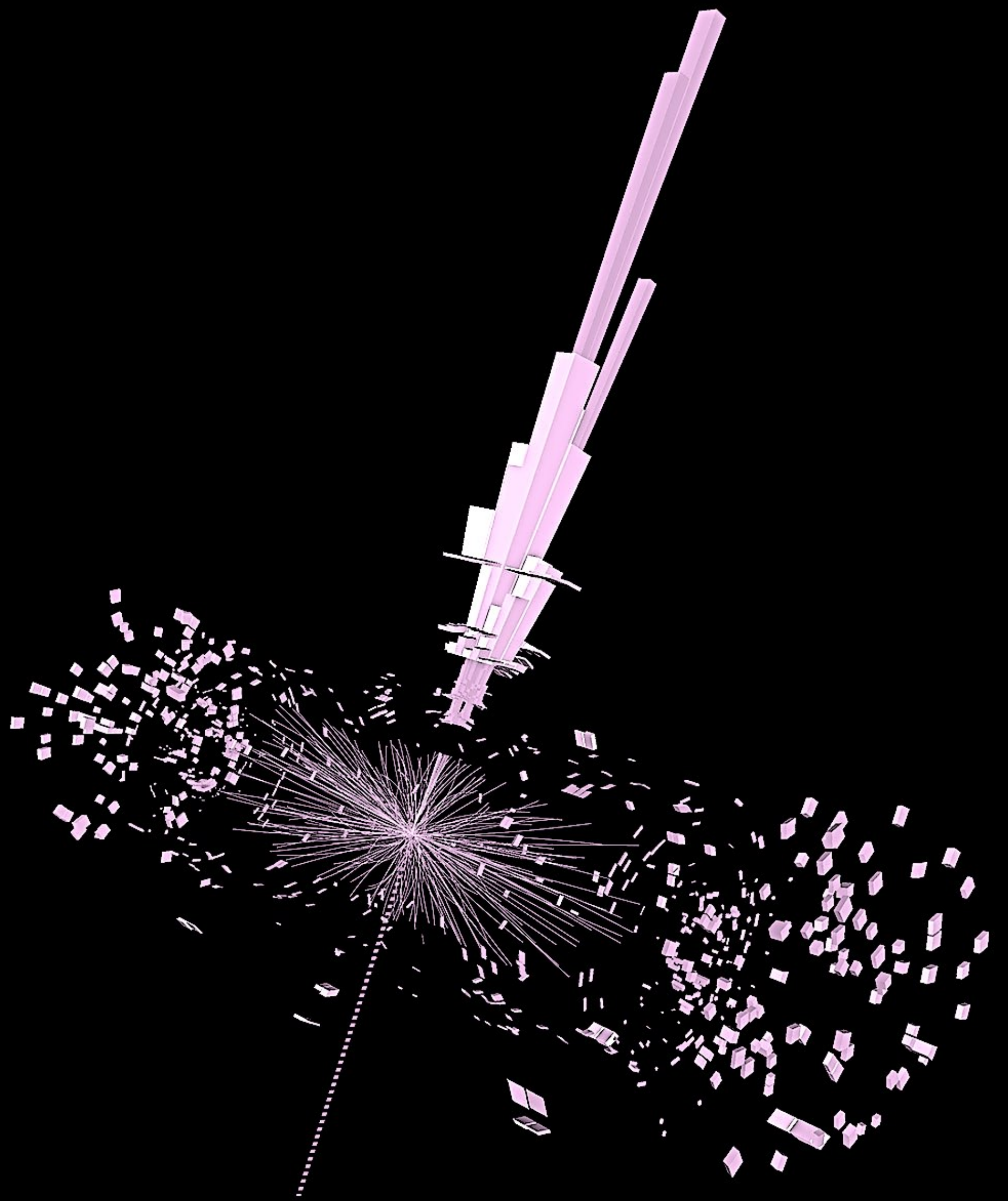


HLT

- Maximize SVJ signal acceptance through dedicated triggers

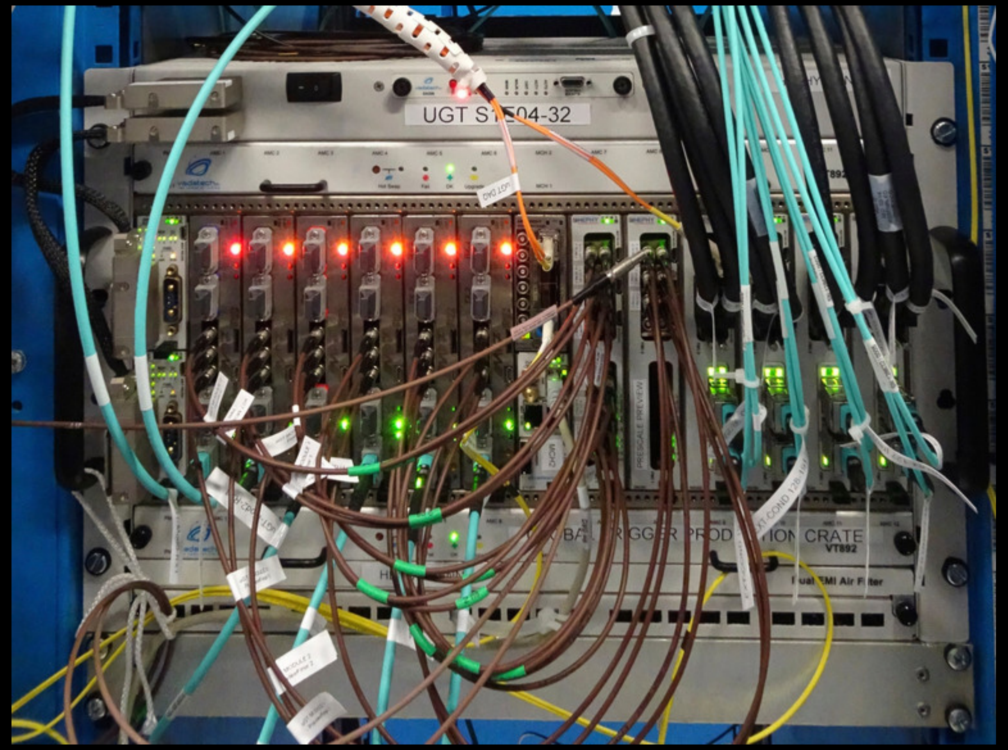
*see backup

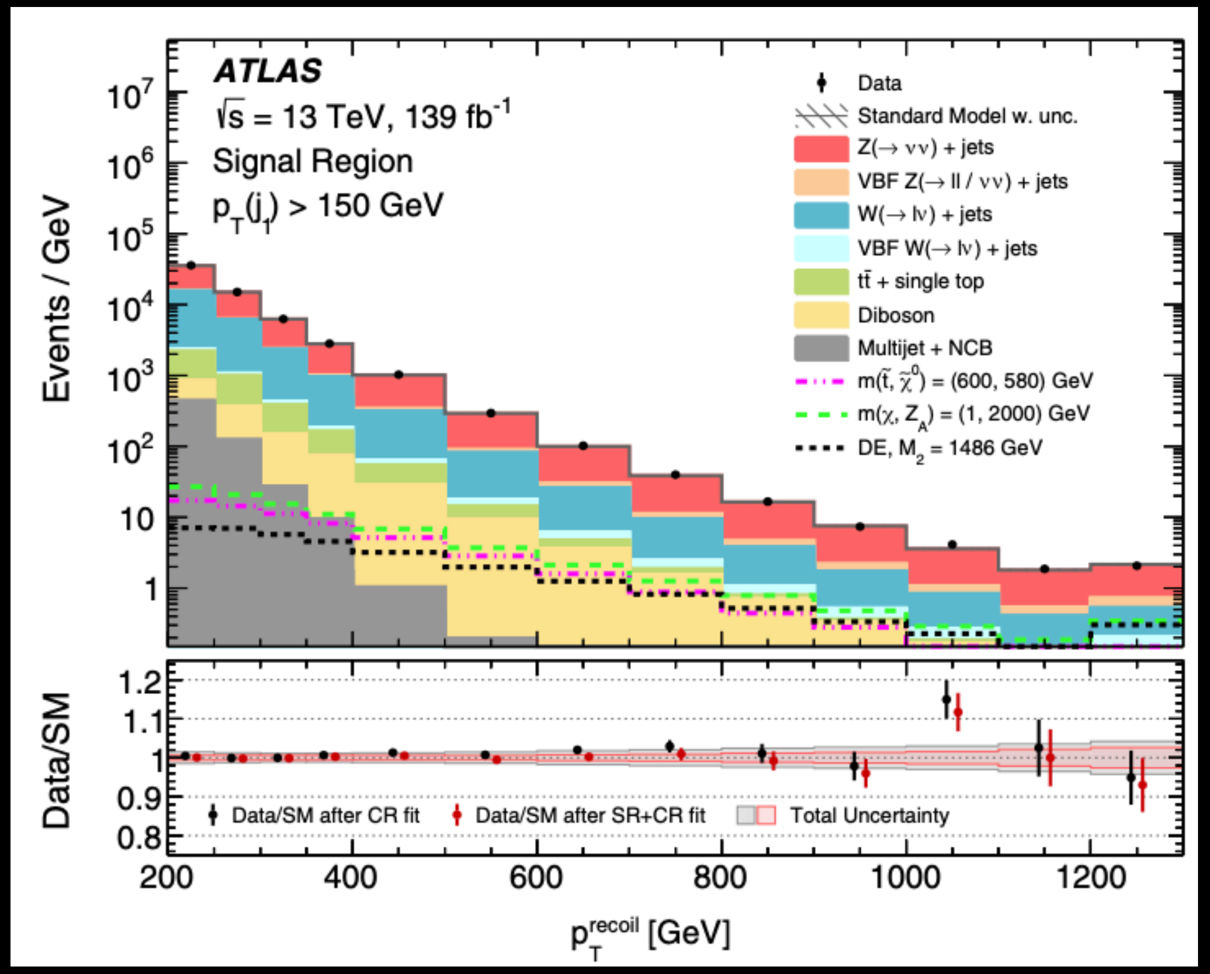




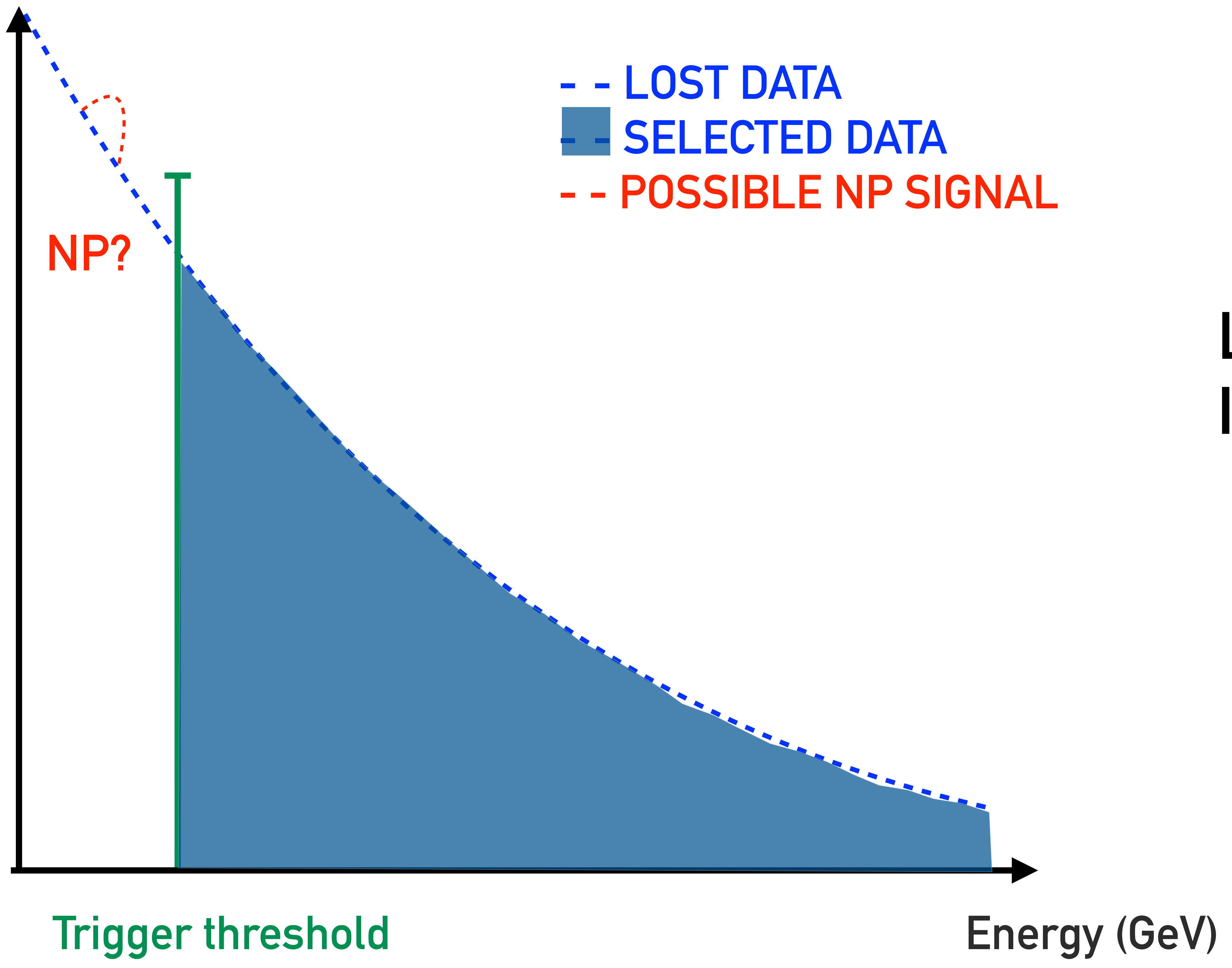
LEVEL-1

- >98% of events rejected here!
- Maximize signal acceptance through dedicated triggers



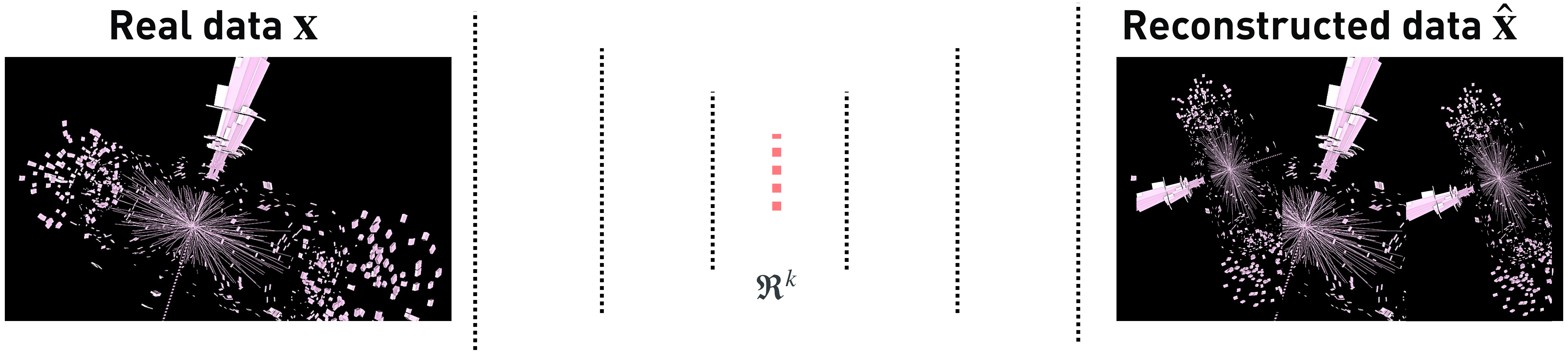


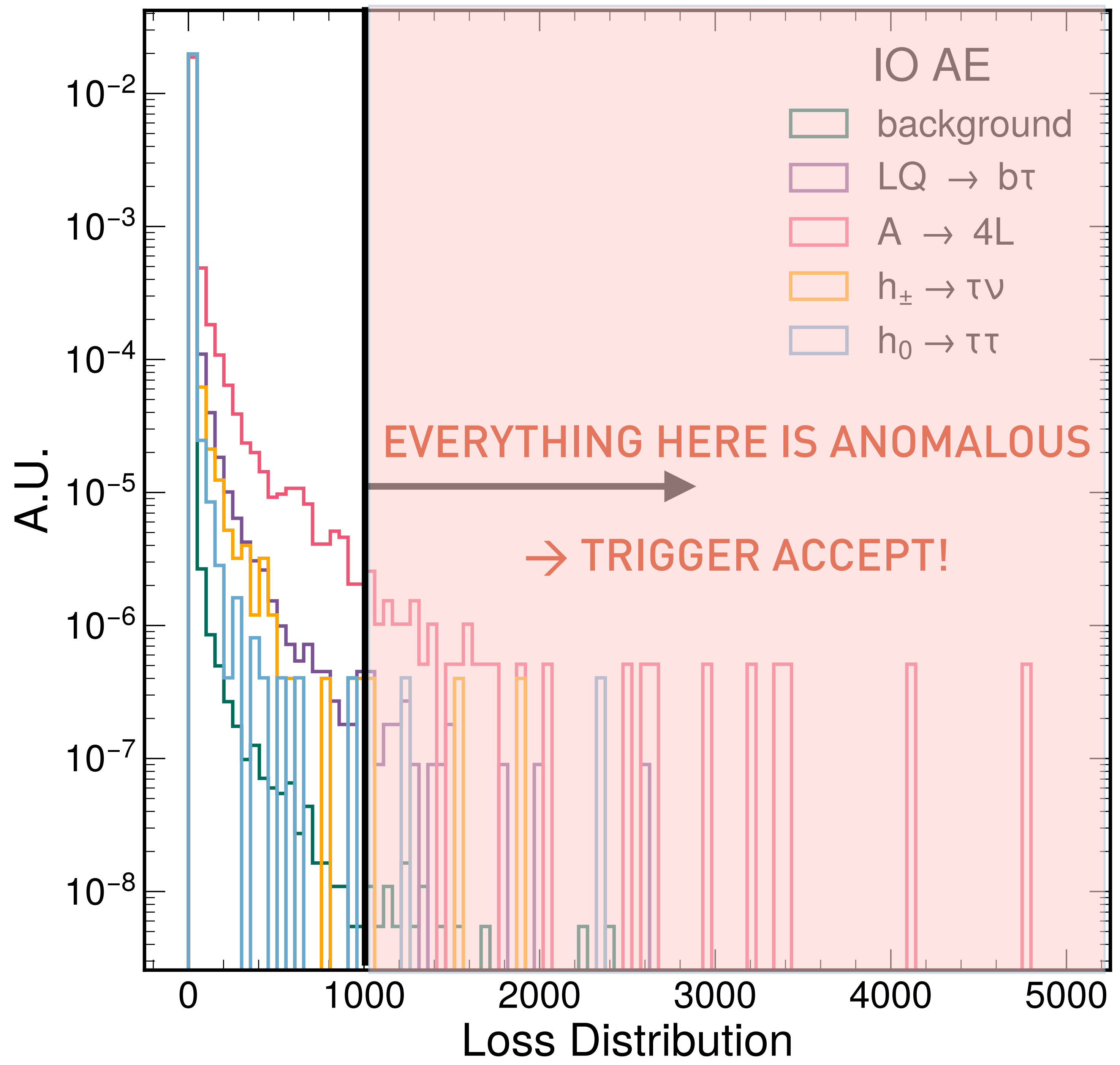
Limitations of current trigger



Level-1 rejects >99% of events!
Is there a more efficient way to select?

Anomaly detection at Level-1





Anomaly detection at Level-1

Extremely challenging task!

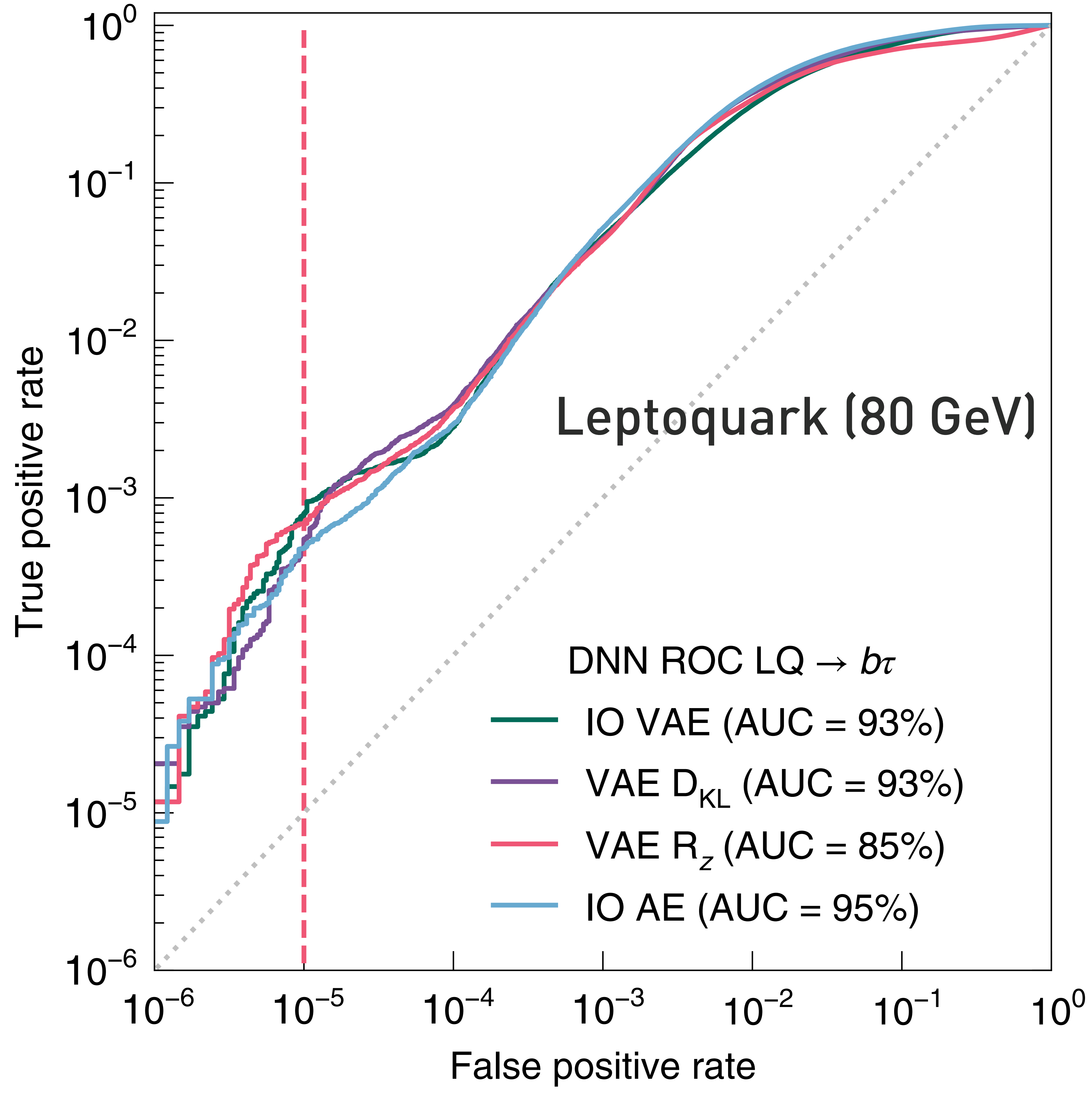
- Requires <100 ns inference in very busy FPGA devices

Preparing to deploy AD @ L1 during Run 3 in CM S!

- Select ~1000 anomalous events/month, can enhance New Physics signals by several orders of magnitude

Run 3: Use Global Trigger inputs (10 jets, 4 muons, 4 e/gamma)

- High-level objects, poor resolution
- Need to get “inside” the jets at L1 for SVJ!



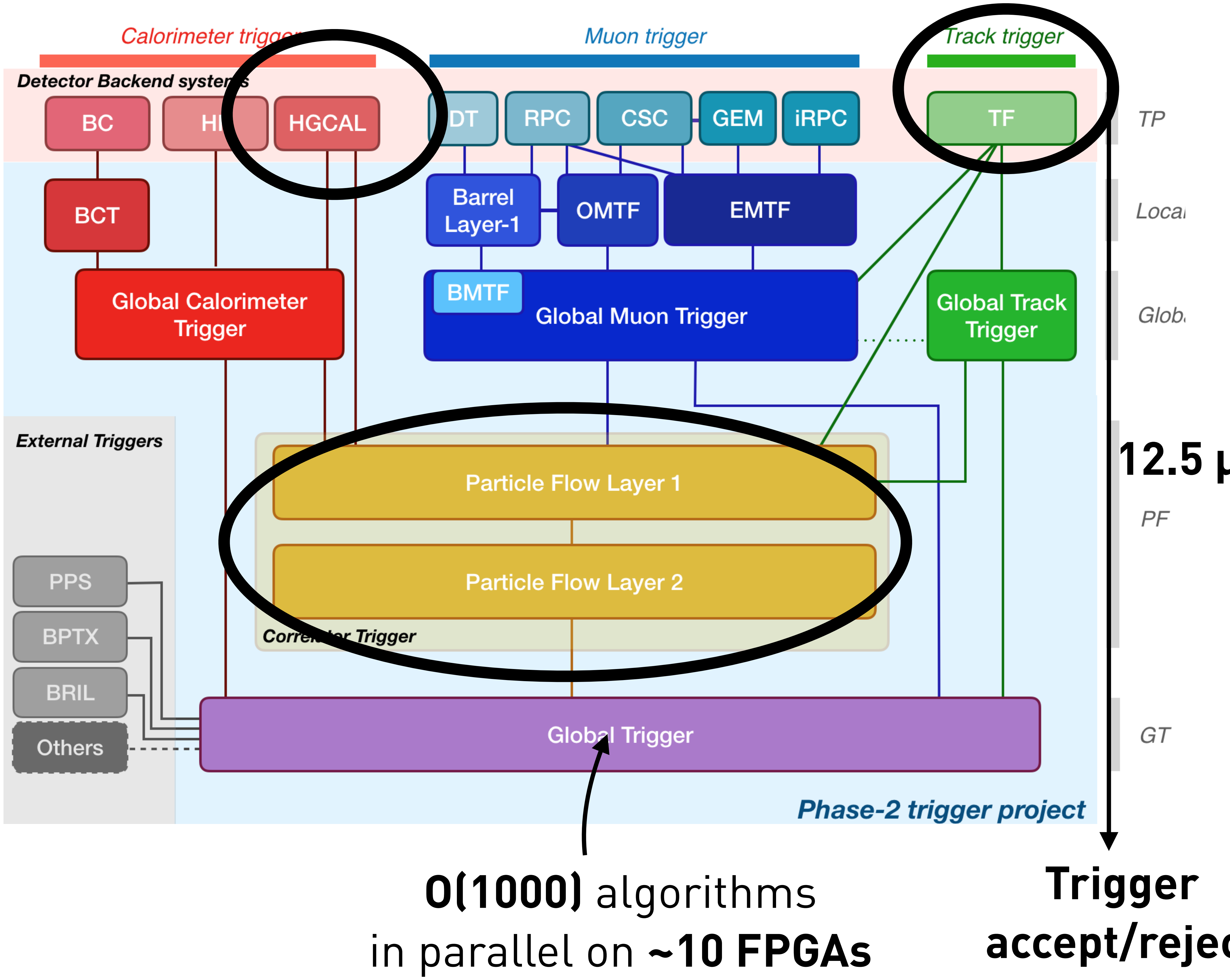
Model	Hardware	DSP [%]	LUT [%]	FF [%]	BRAM [%]	Latency [ns]	II [ns]
DNN VAE[a] PTQ 8 bits	Xilinx VU9P	1	3	0.5	0.3	80	5

Anomaly detection at Level-1

HL-LHC is where anomalous jet triggering at L1 might happen!

- Tracking + PF at L1 opens up many doors
- Must begin R&D now

- AI can provide highly efficient SVJ tags!
  Boils down to latency, resources and bandwidth



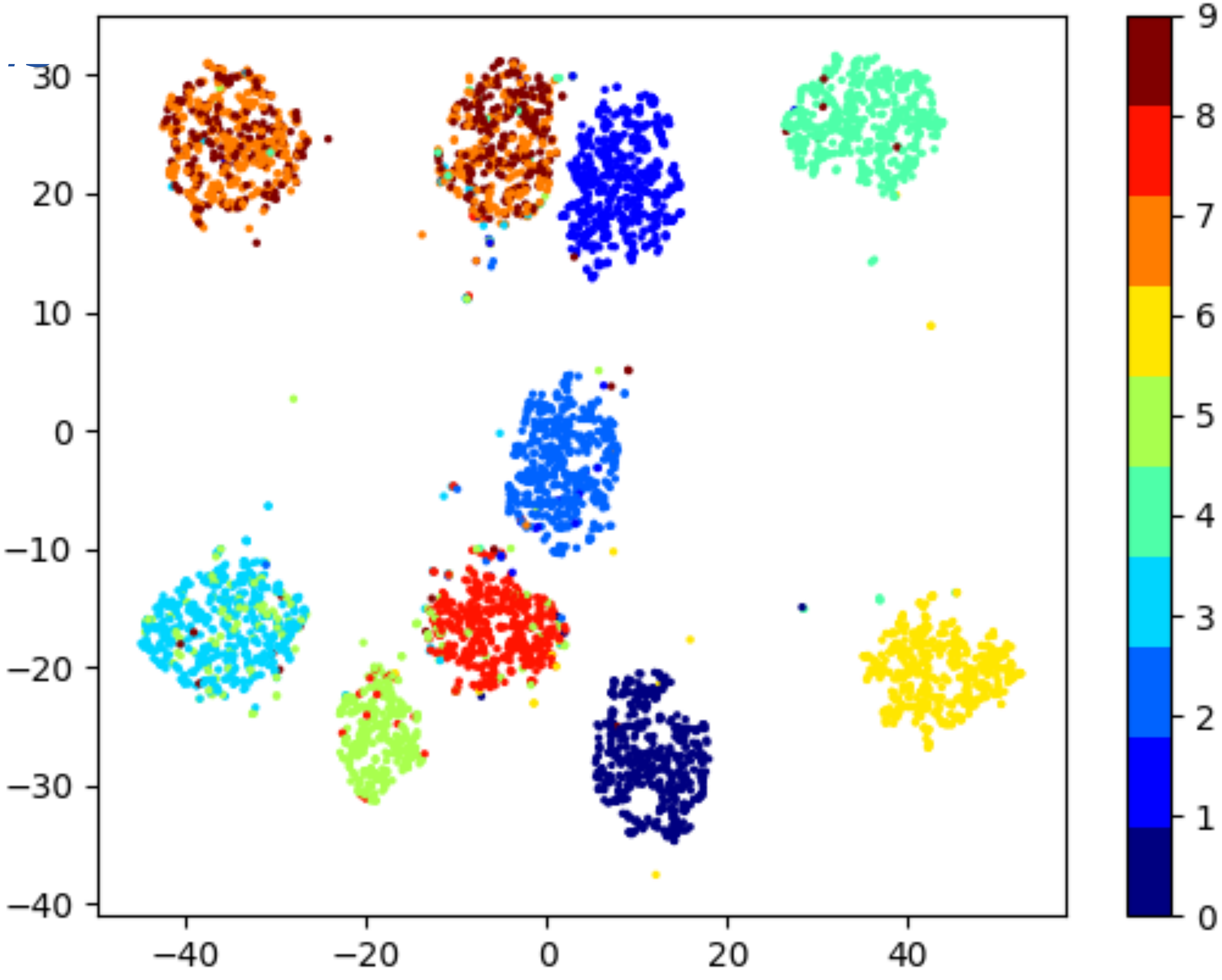
What will we do with these data?

?

Clustering algorithm?

?

Catalogue of anomalous events?



?

Statistical analysis?

?

Anomalous Open Data?

?

Conclusion

ADC 2021

Data challenge on real-time anomaly detection

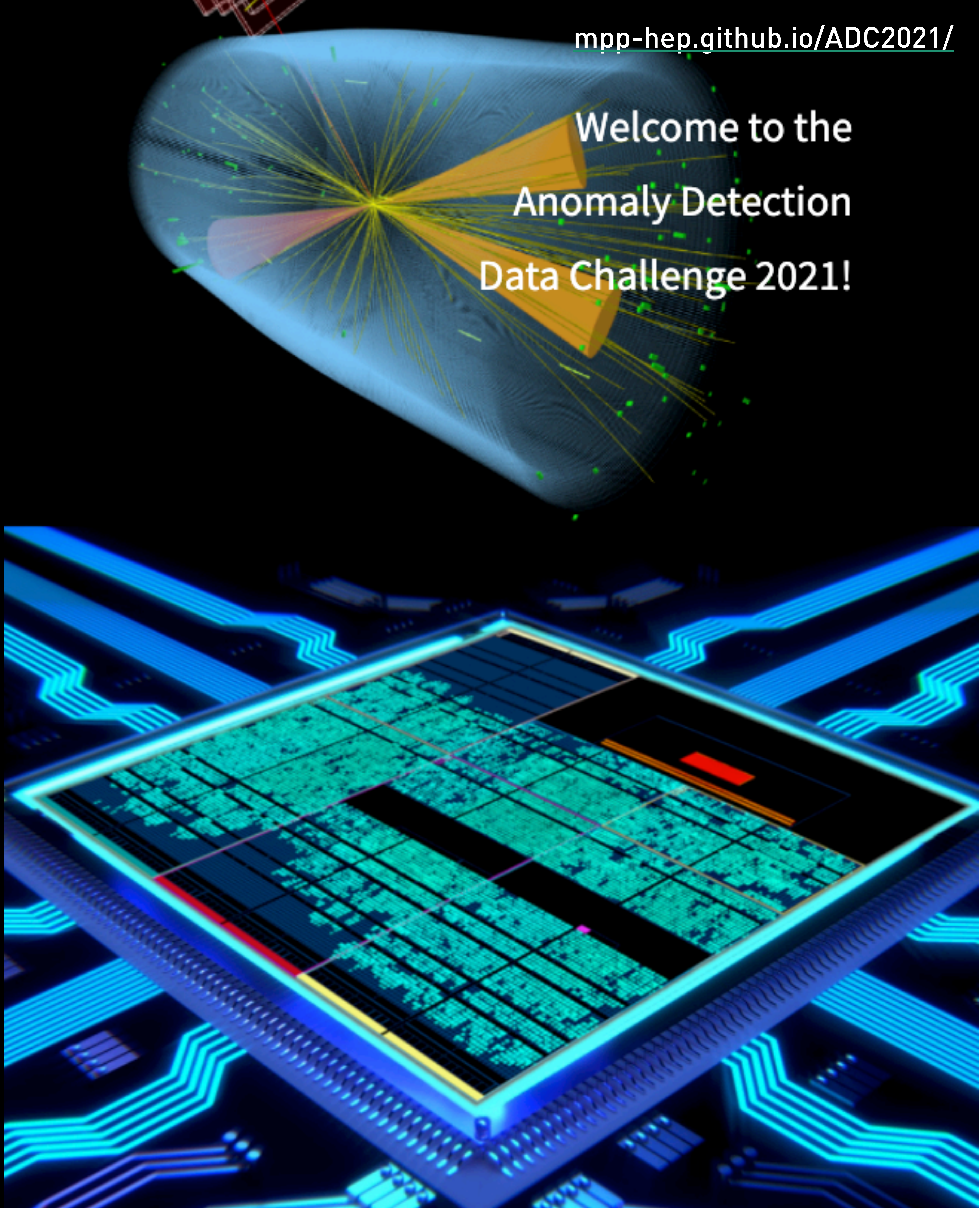
- Dataset: Nature Scientific Data (2022) 9:118
- Code: mpp-hep.github.io/ADC2021/

Tutorial: Anomaly detection on FPGA with hls4ml

- github:thaarres/quantumUniverse_pynqZ2

Join monthly Fast Machine Learning meetings

- Sign up to our Fast ML e-group hls-fml here

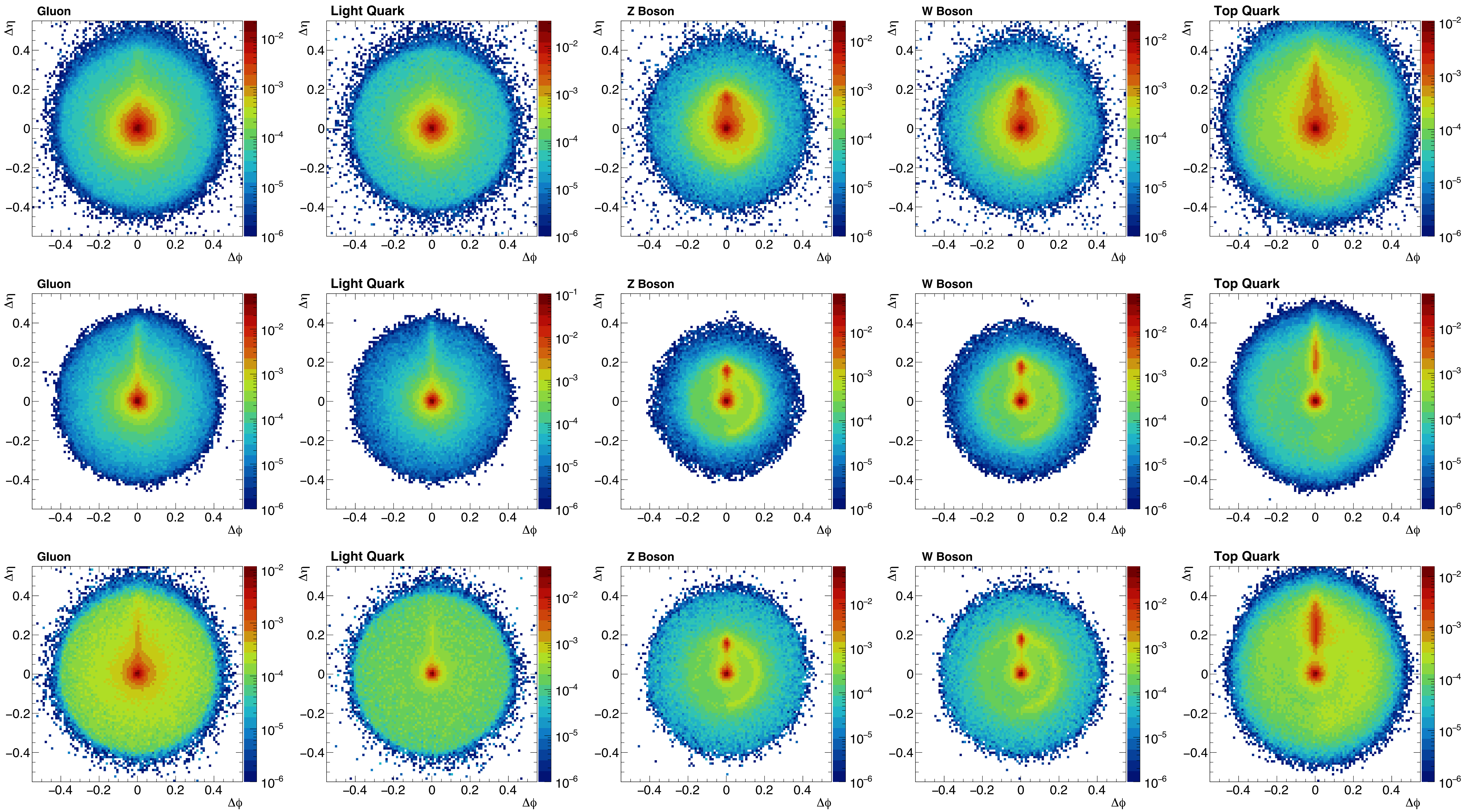


Backup

ABCNet:

Pixel intensity = particle importance w.r.t most energetic particle in jet, from attention weights

No substructure information given, learned through attention layers!



JetClass Dataset

Encoding a lot of information → a lot of parameters
- Critical to avoid overtraining and ensure generic embeddings

GPT-3: trained on ~200 billion words (estimated cost O(10) million dollars)
- Need huge statistics to train a jet transformer!
- ParT: Dedicated particle transformer, 2M parameters!

	Accuracy	# params	FLOPs
PFN	0.772	86.1 k	4.62 M
P-CNN	0.809	354 k	15.5 M
ParticleNet	0.844	370 k	540 M
ParT	**0.861**	2.14 M	340 M
ParT (plain)	0.849	2.13 M	260 M

JetClass Dataset

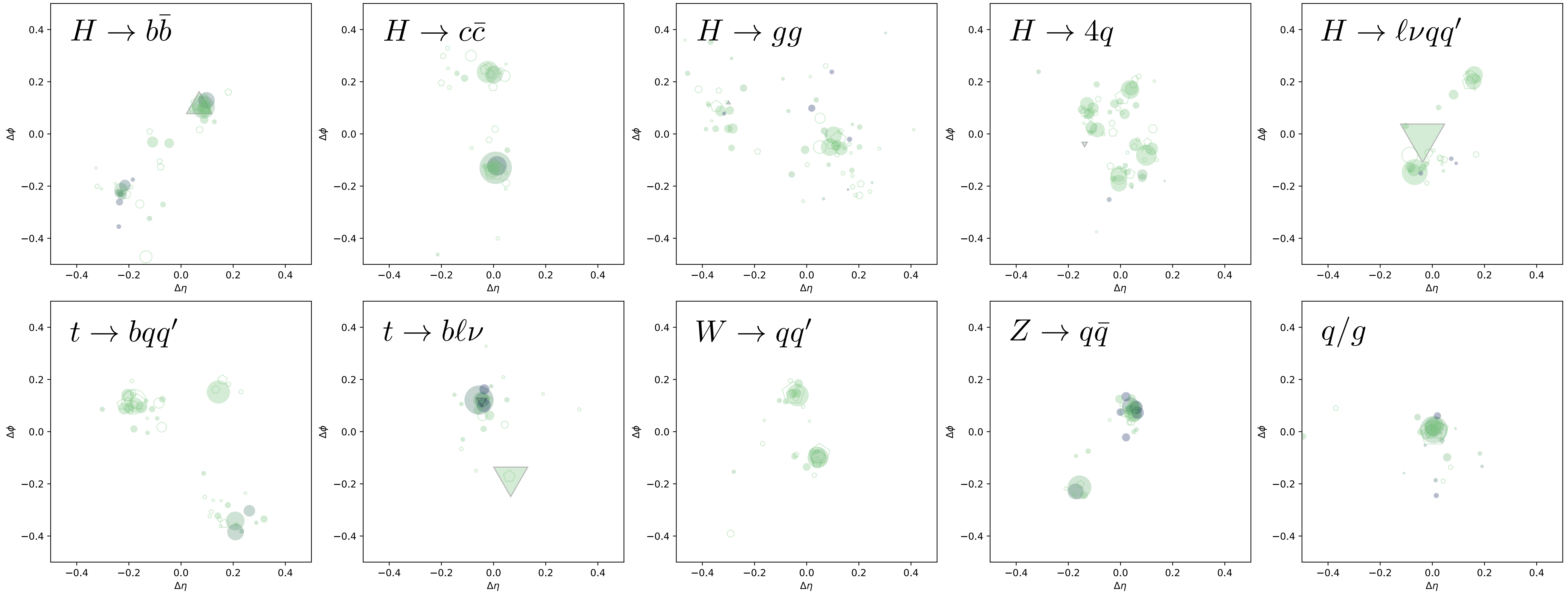
Encoding a lot of information → a lot of parameters
- Critical to avoid overtraining and ensure generic embeddings

GPT-3: trained on ~200 billion words (estimated cost O(10) million dollars)
- Need huge statistics to train a jet transformer!
- ParT: Dedicated particle transformer, 2M parameters!

New dedicated jet tagging dataset: Jetclass
- 10 types of jets
- 100M training
- 5M validation
- 20M test

Extremely useful for benchmarking of new algorithms!

	Accuracy	# params	FLOPs
PFN	0.772	86.1 k	4.62 M
P-CNN	0.809	354 k	15.5 M
ParticleNet	0.844	370 k	540 M
ParT	**0.861**	2.14 M	340 M
ParT (plain)	0.849	2.13 M	260 M

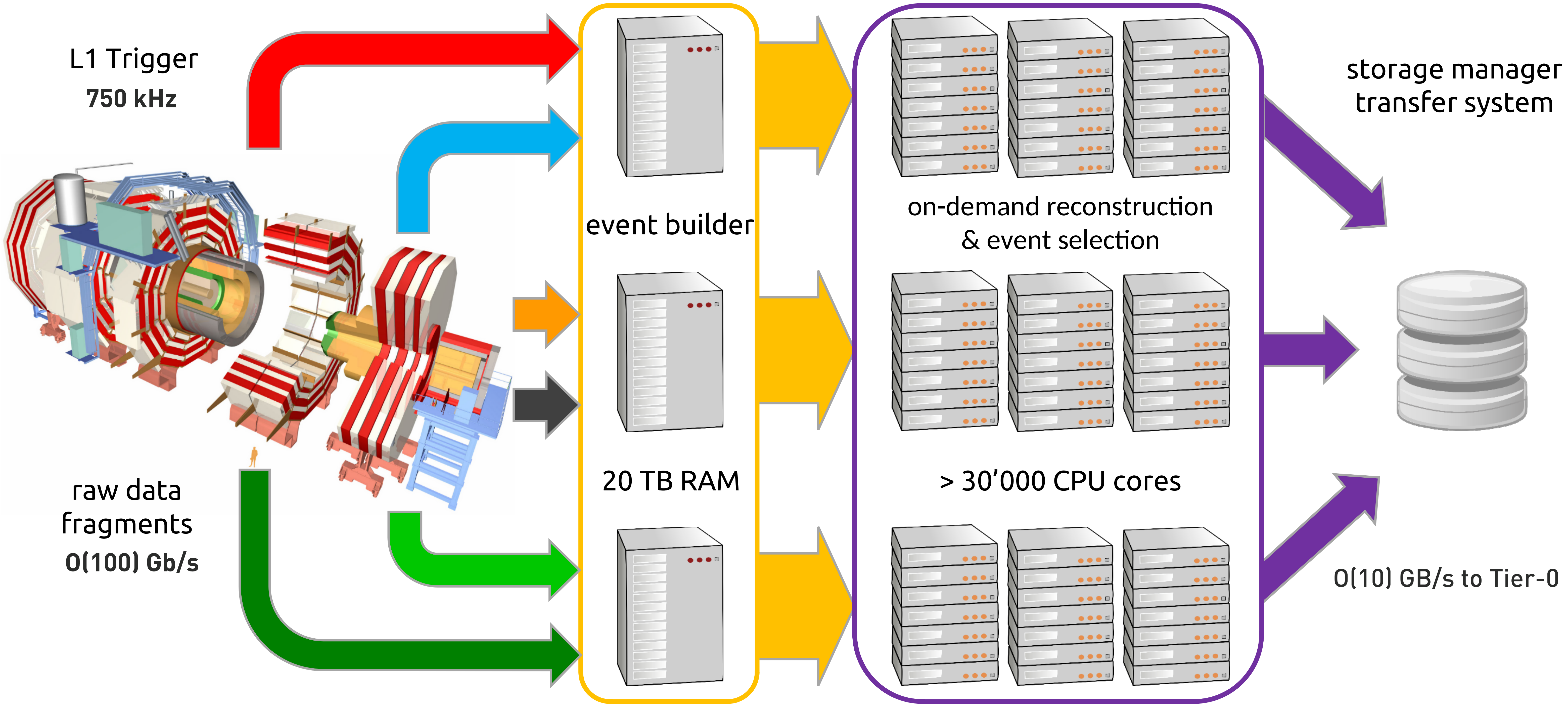


High Level Trigger

Amount of data we can store for use in analysis limited by bandwidth, O(10) GB/s to Tier-0

- 300 ms to decide keep/reject
- Running thousands of “modules” on many collision events in parallel

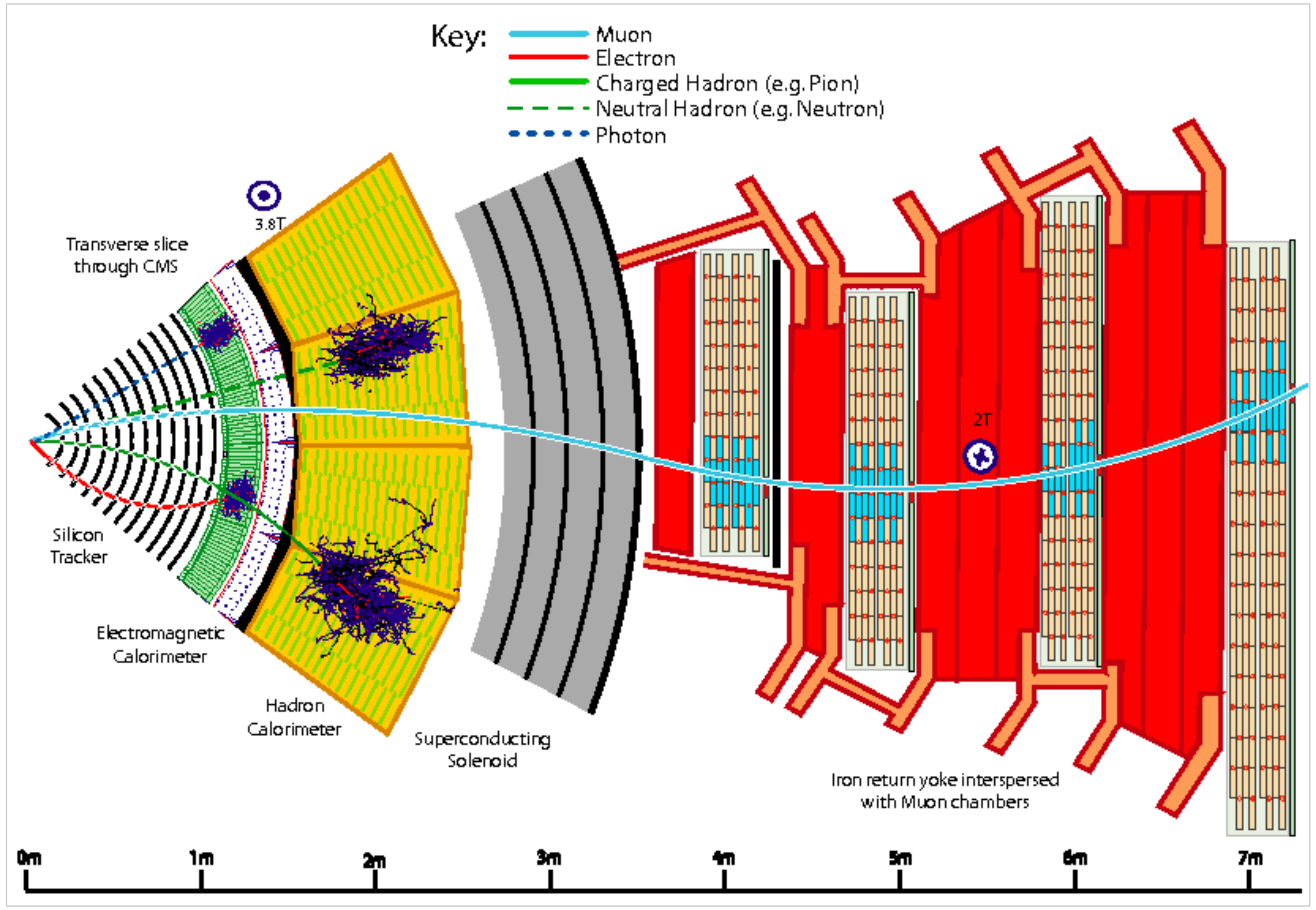
Bandwidth (kB/s) = Event rate (kHz) x Event size (kB)



HLT: More Particle Flow

Particle Flow is highest resolution reconstruction at HLT. Slow, can't run on all events! Currently only PF on 17% of total

- High resolution, but small rate



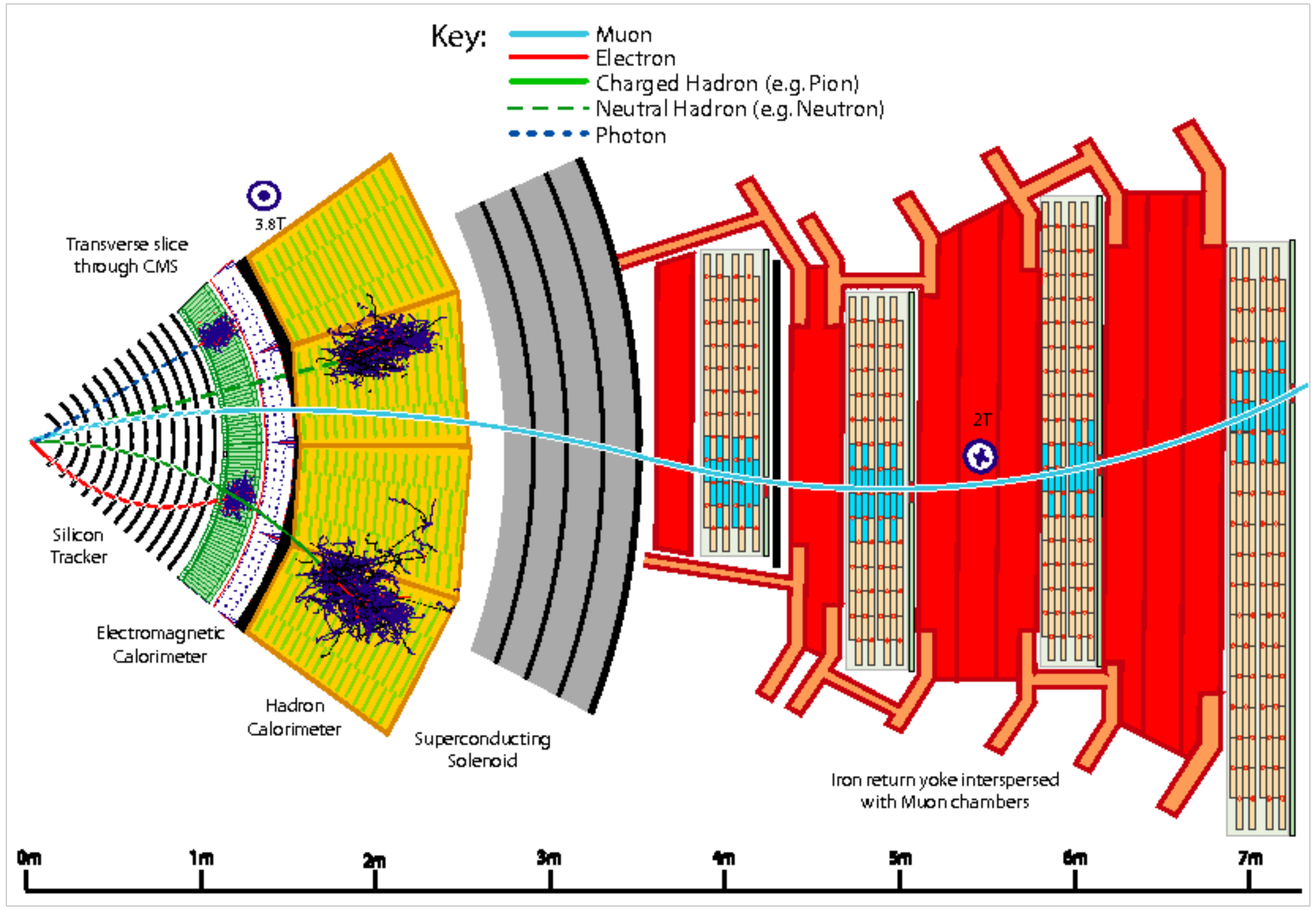
HLT: More Particle Flow

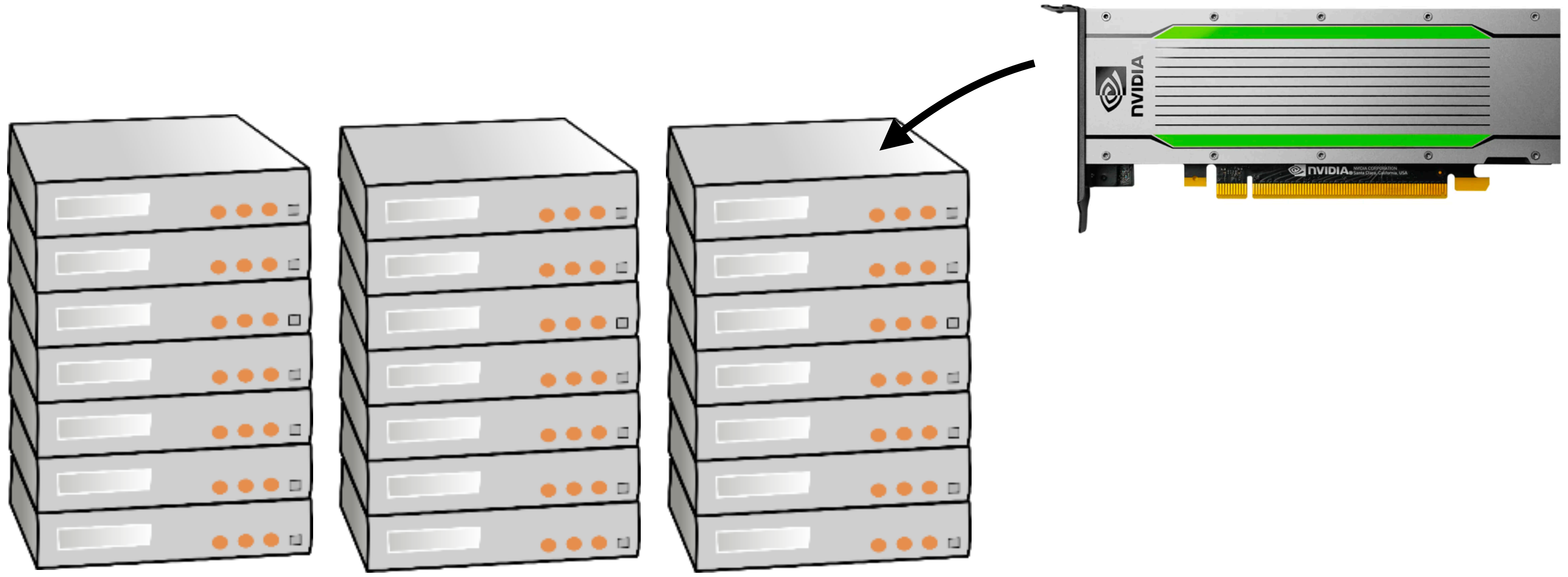
Particle Flow is highest resolution reconstruction at HLT. Slow, can't run on all events! Currently only PF on 17% of total

- High resolution, but small rate

To handle HL-LHC data rates

- Offload resource-intensive computations to GPU
- Can achieve speed-ups ~x3
- More compute to run PF!





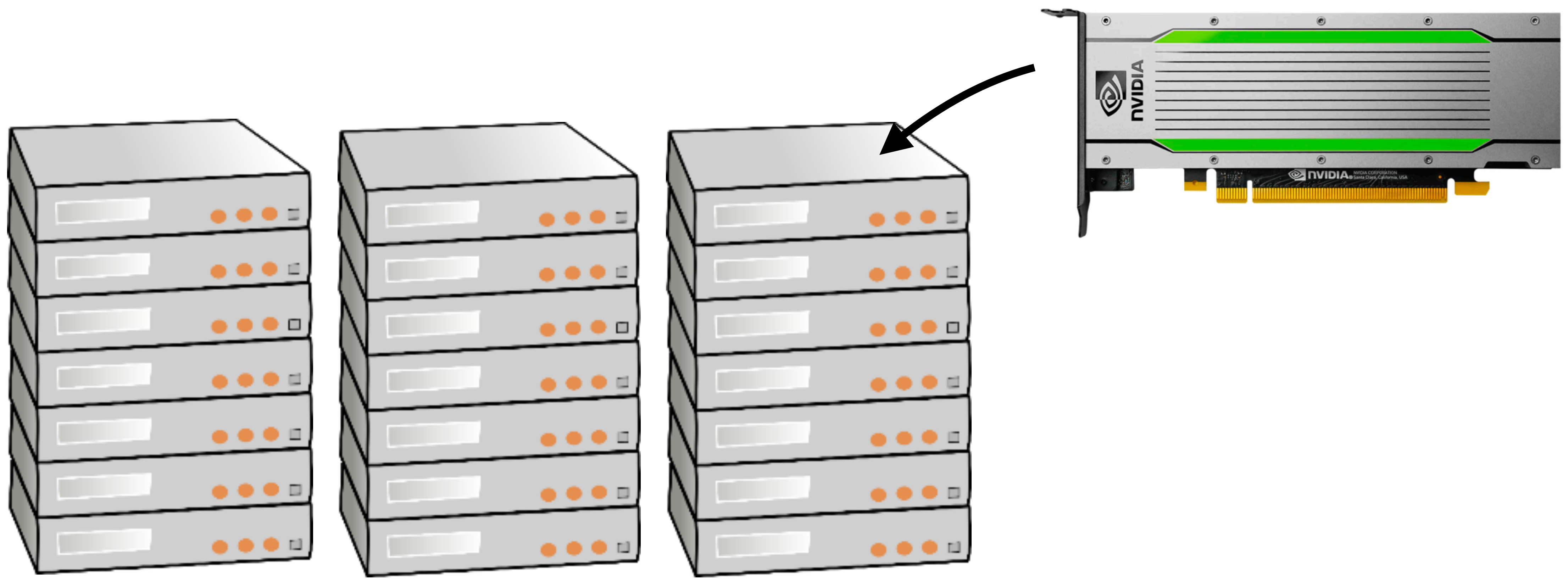
HLT: More Particle Flow

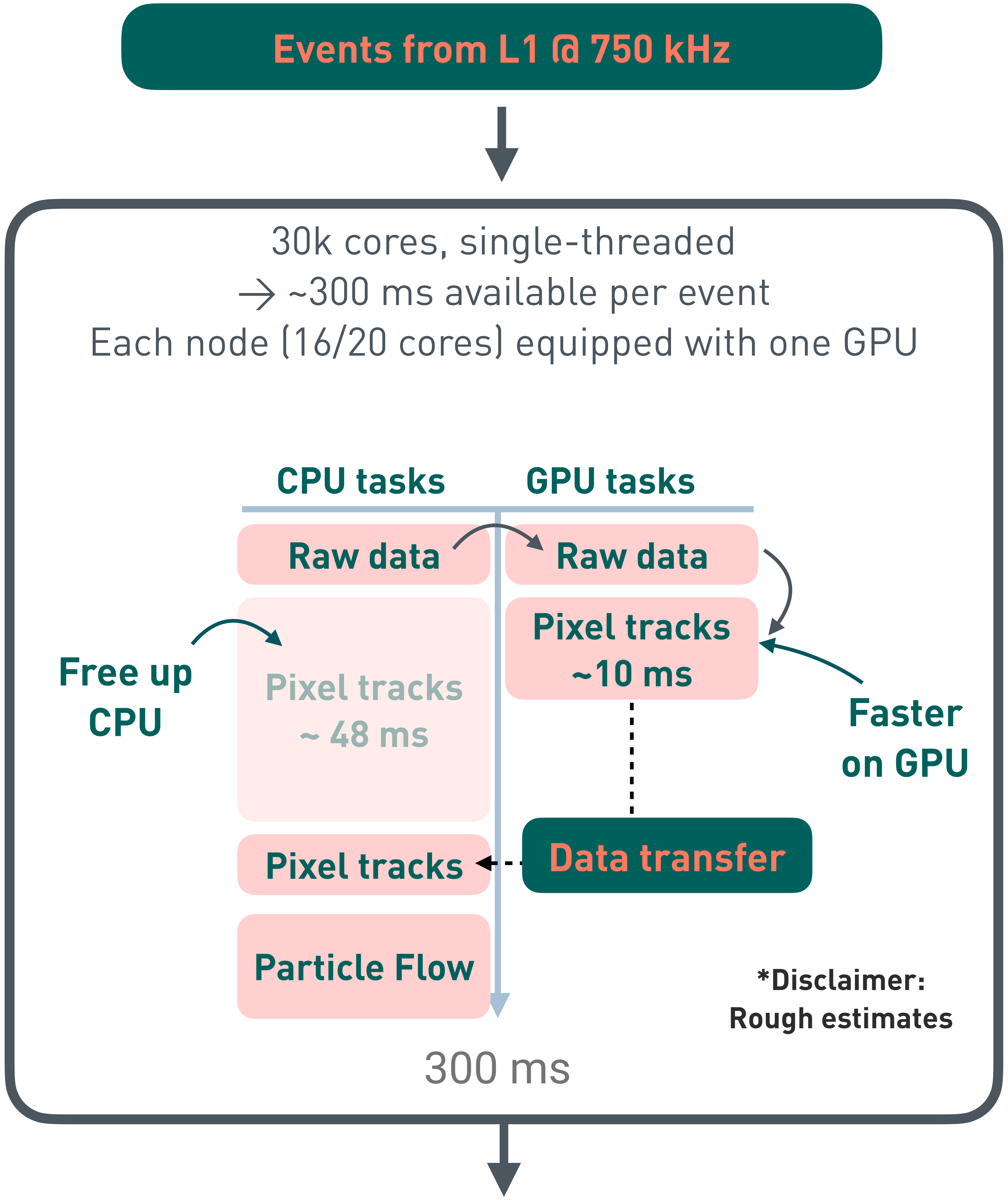
Particle Flow is highest resolution reconstruction at HLT. Slow, can't run on all events! Currently only PF on 17% of total

- High resolution, but small rate

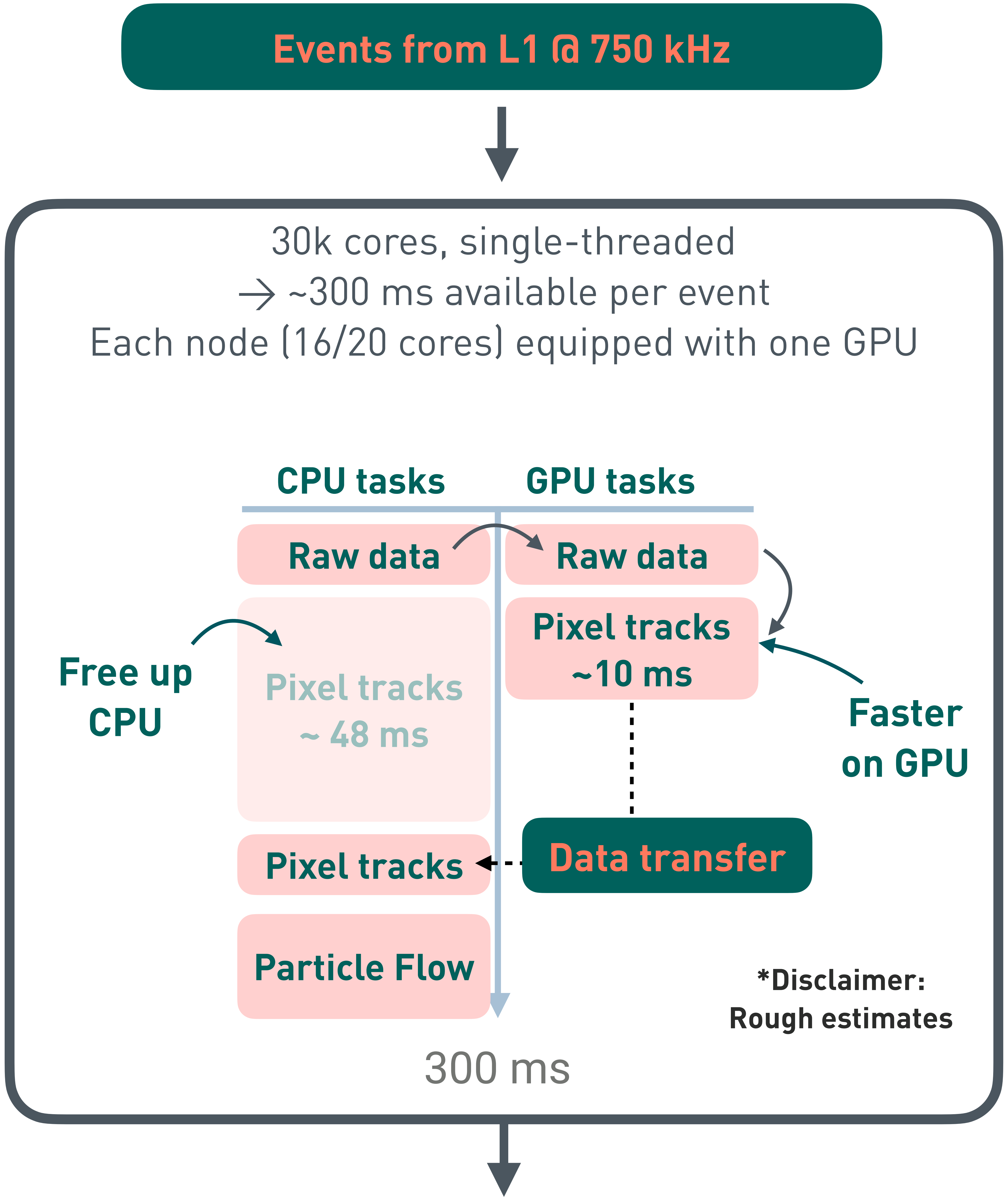
To handle HL-LHC data rates

- Offload resource-intensive computations to GPU
- Can achieve speed-ups ~x3
- More compute to run PF!





HLT: More Particle Flow

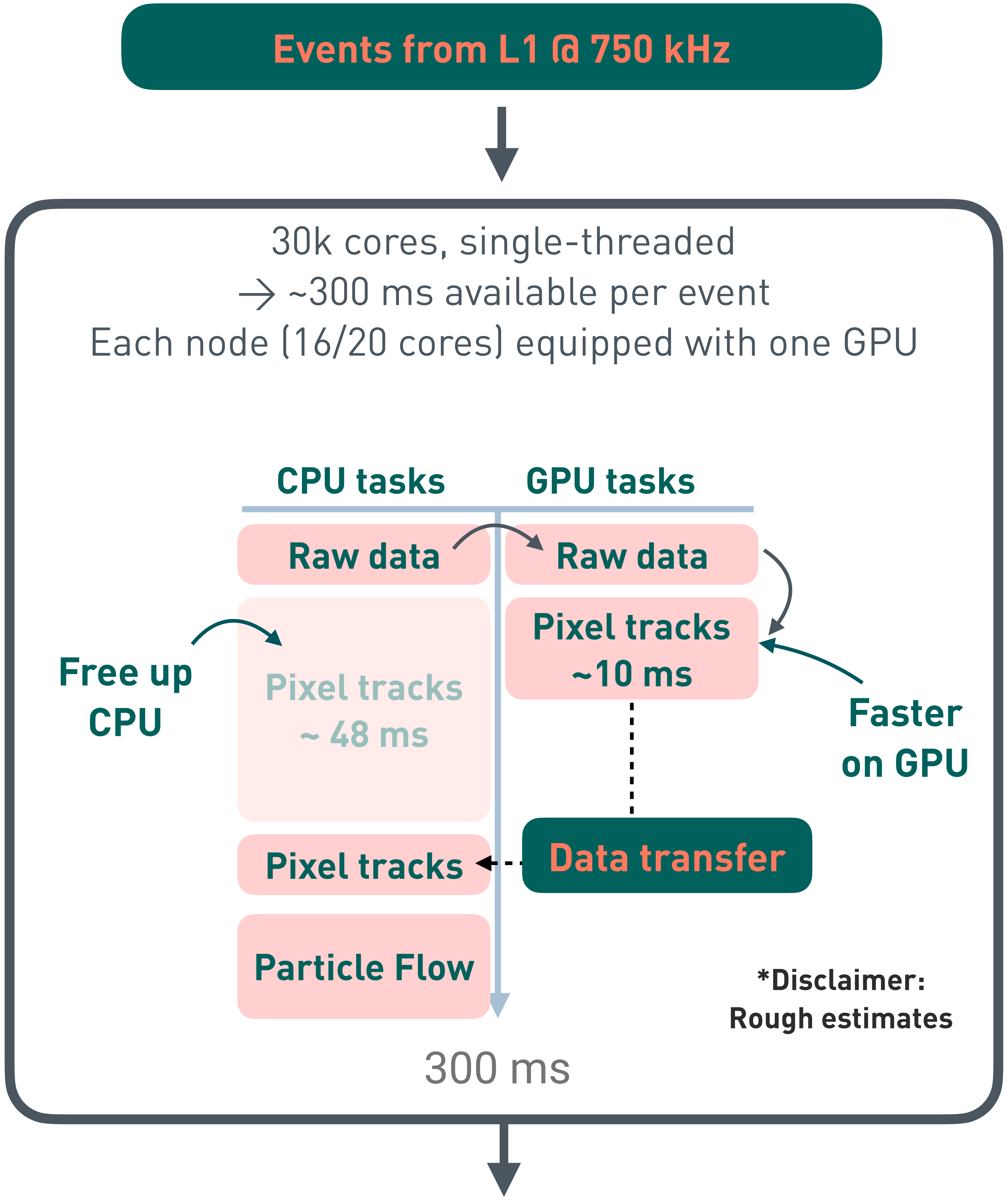


HLT: More Particle Flow

More PF means cleaner SVJ triggering!

- Jet substructure and anomaly detection at HLT important!

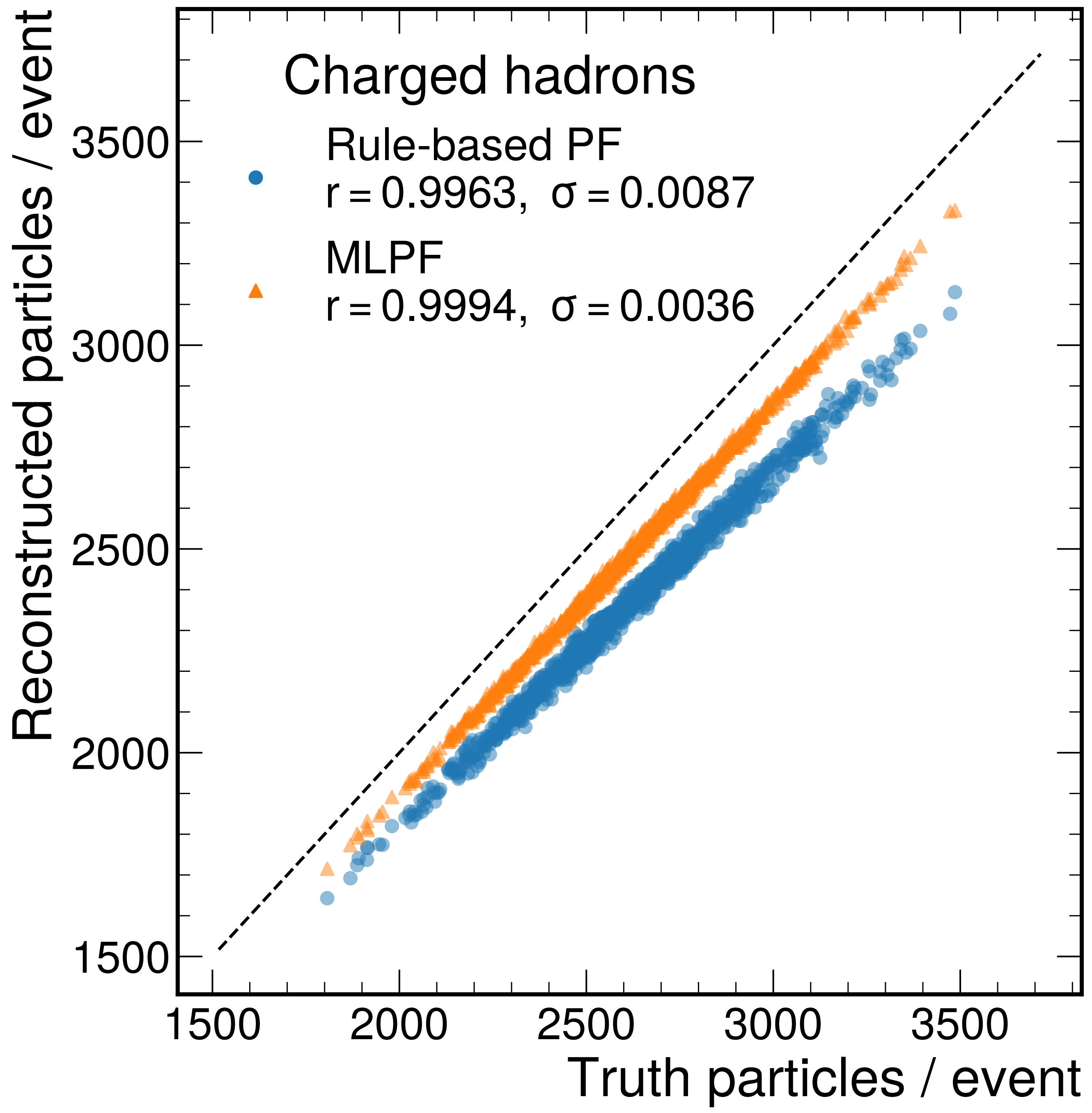
Transfer data GPU → CPU expensive, can we avoid it by doing PF on GPU?

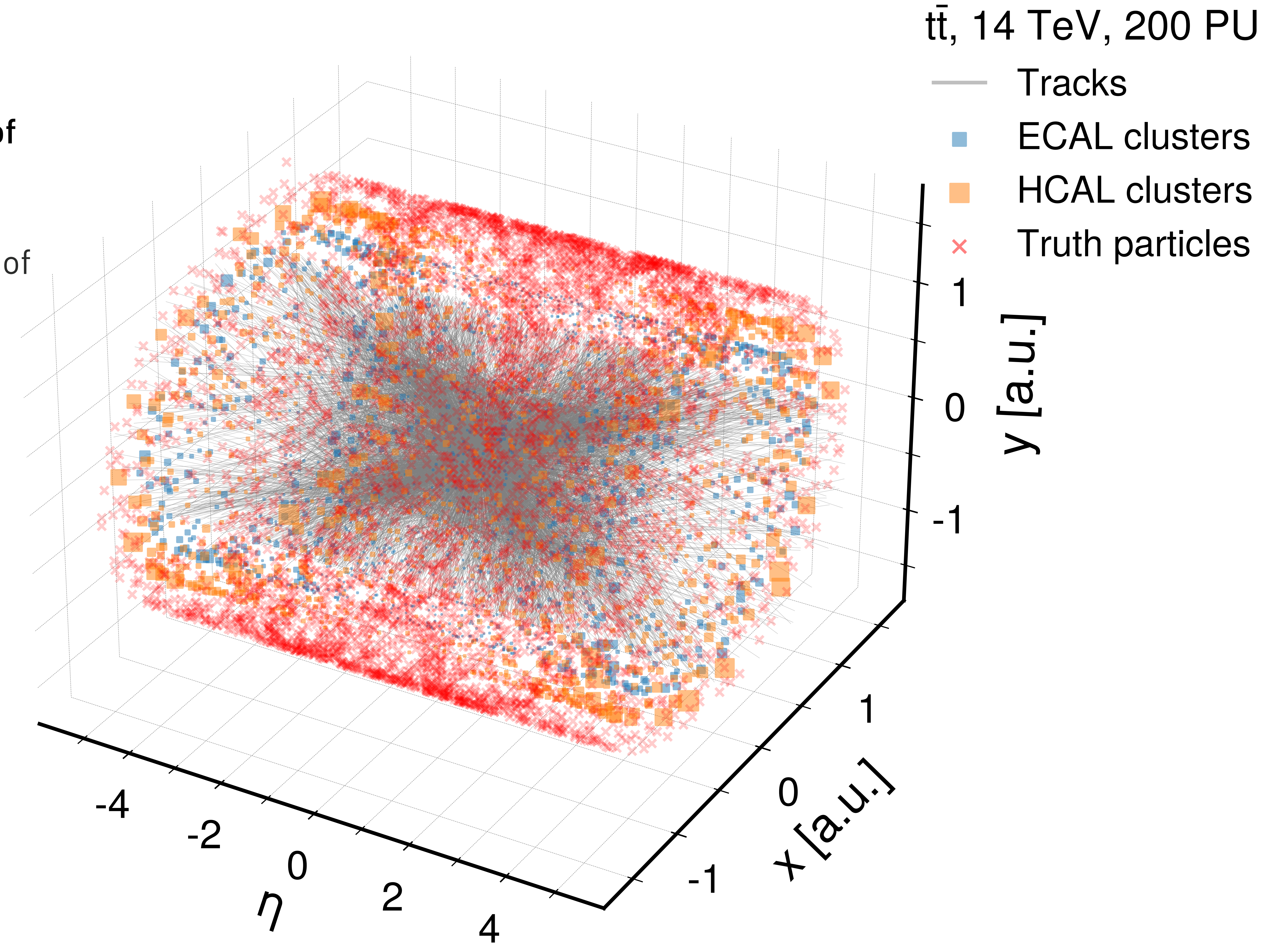


HLT: More Particle Flow

Deep Neural Networks as “fast” approximations of classical ParticleFlow

- Inherently parallelizeable, can take advantage of GPU acceleration
- High accuracy in high PU environment





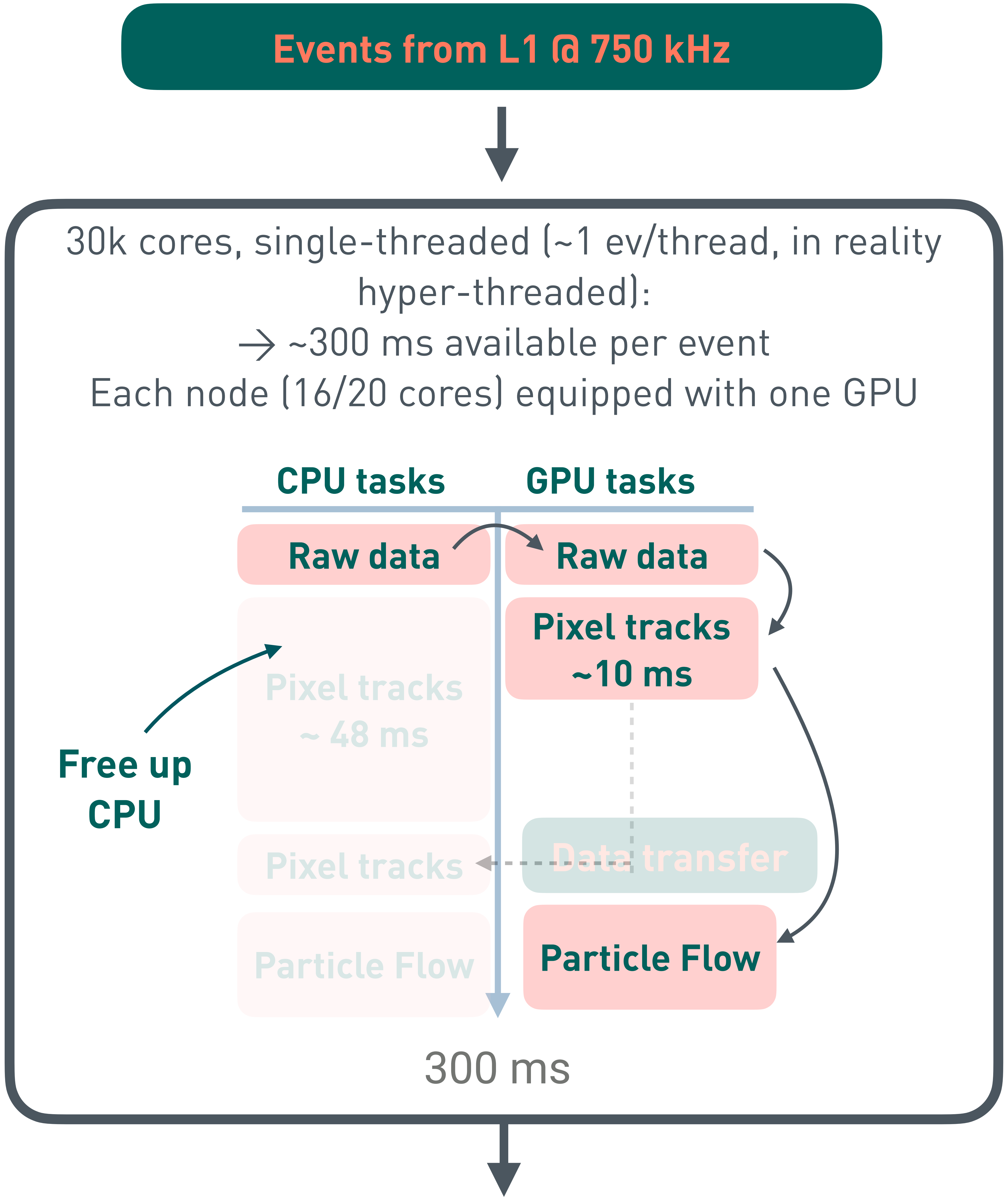
arXiv:2101.08578

HLT: More Particle Flow

We will (in general) store more of better data

Dedicated SVJ PF-based triggers

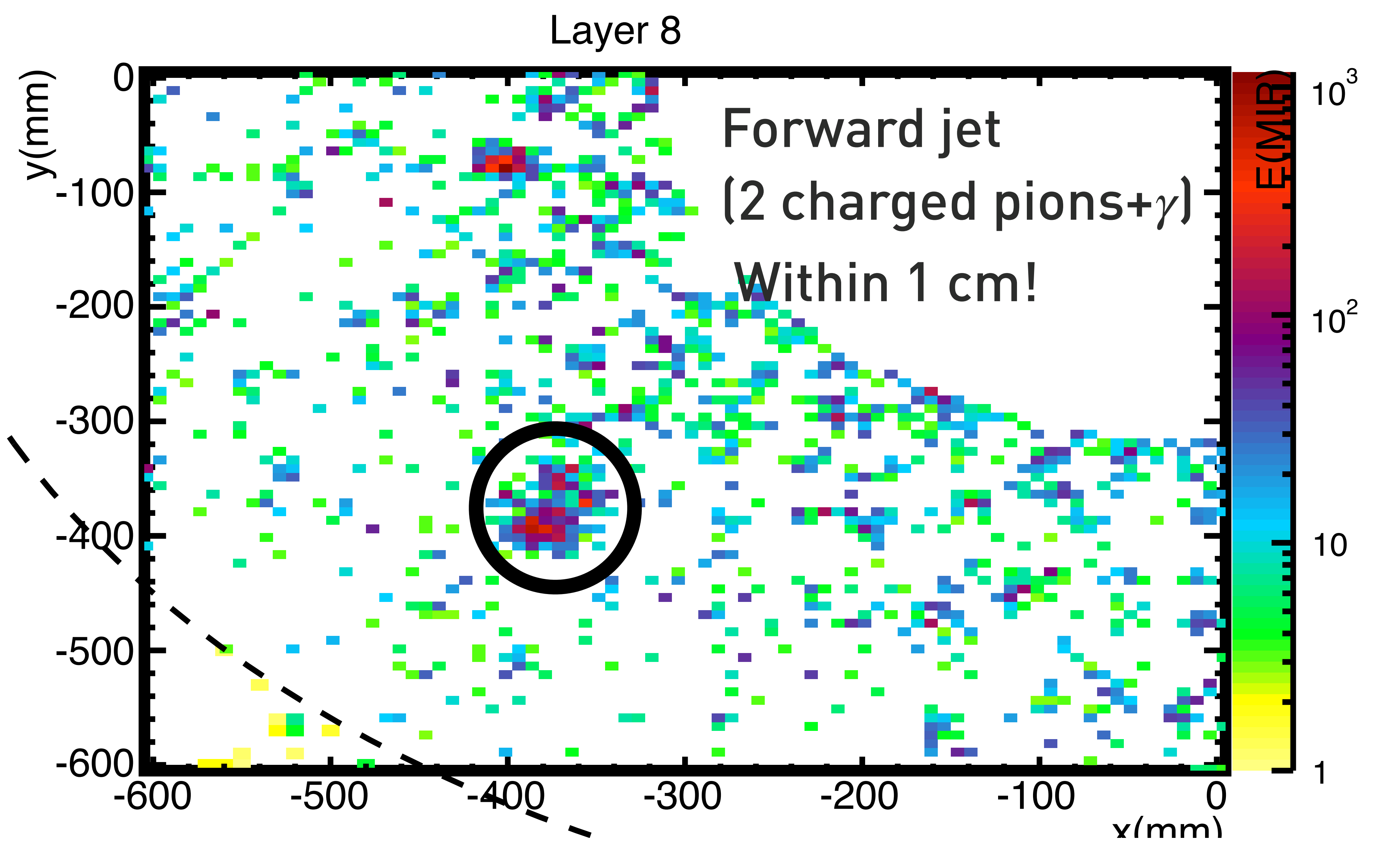
- Autoencoders for anomalous jets

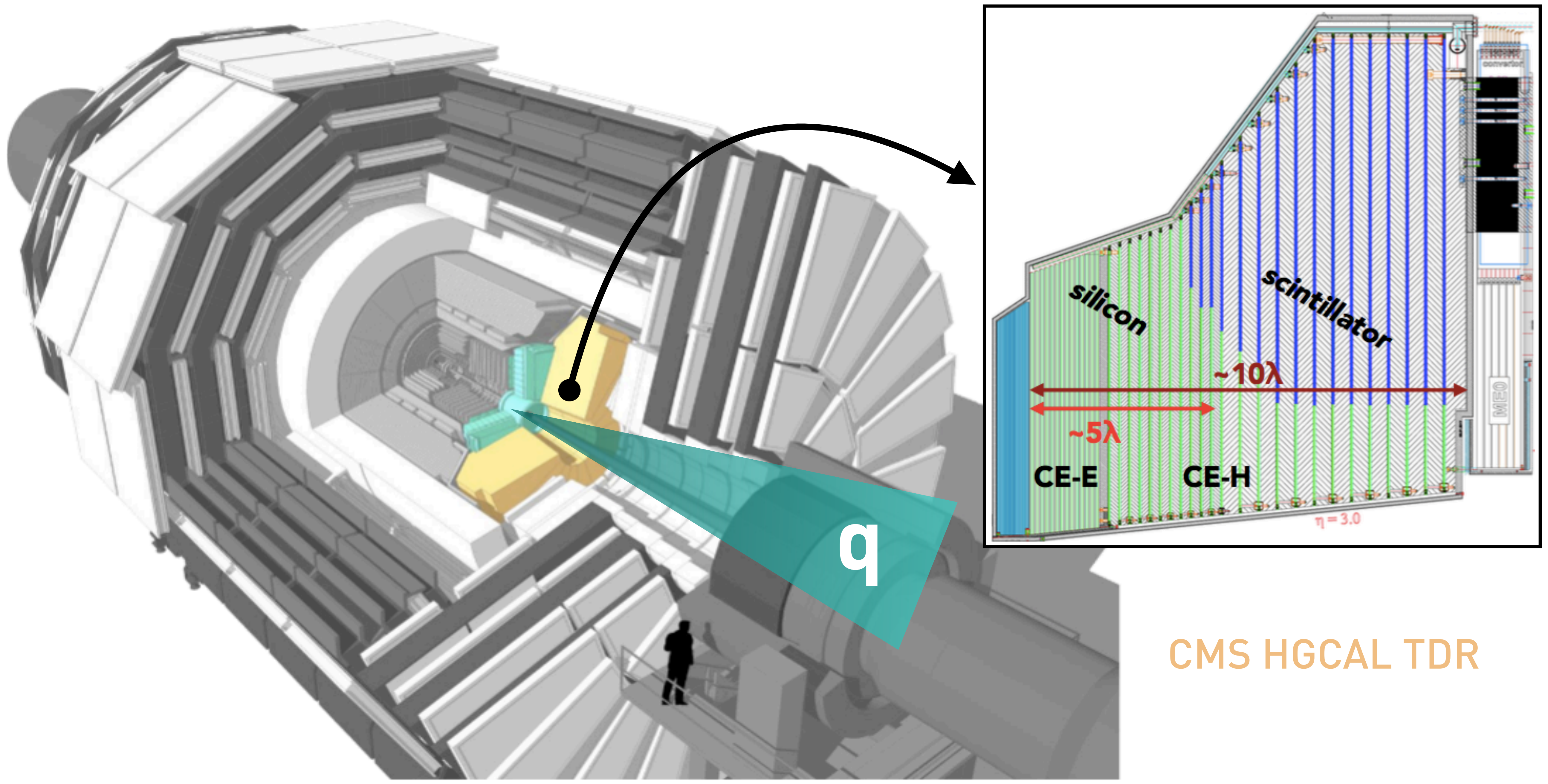


On-detector: HGCAL

CMS Endcap High-Granularity Calorimeter (1.5<η<3)

- Unprecedented transverse/longitudinal segmentation
- Pile-up suppression and forward jet resolution





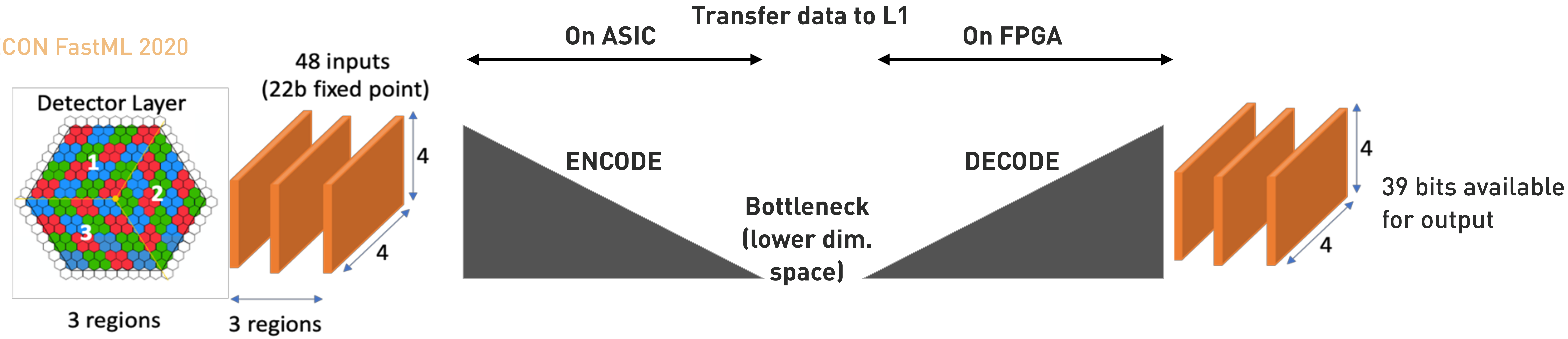
CMS HGCAL TDR

- 52 layers, 6 million silicon channels, limited output bandwidth
- Operate at -30°C → need low-power on-ASIC preprocessing

On-detector: HGCAL

Optimise information output using ML! Maximise resolution on extremely low power.

ECON FastML 2020



ABCNet:

Pixel intensity = particle importance w.r.t most energetic particle in jet, from attention weights

No substructure information given, learned through attention layers!