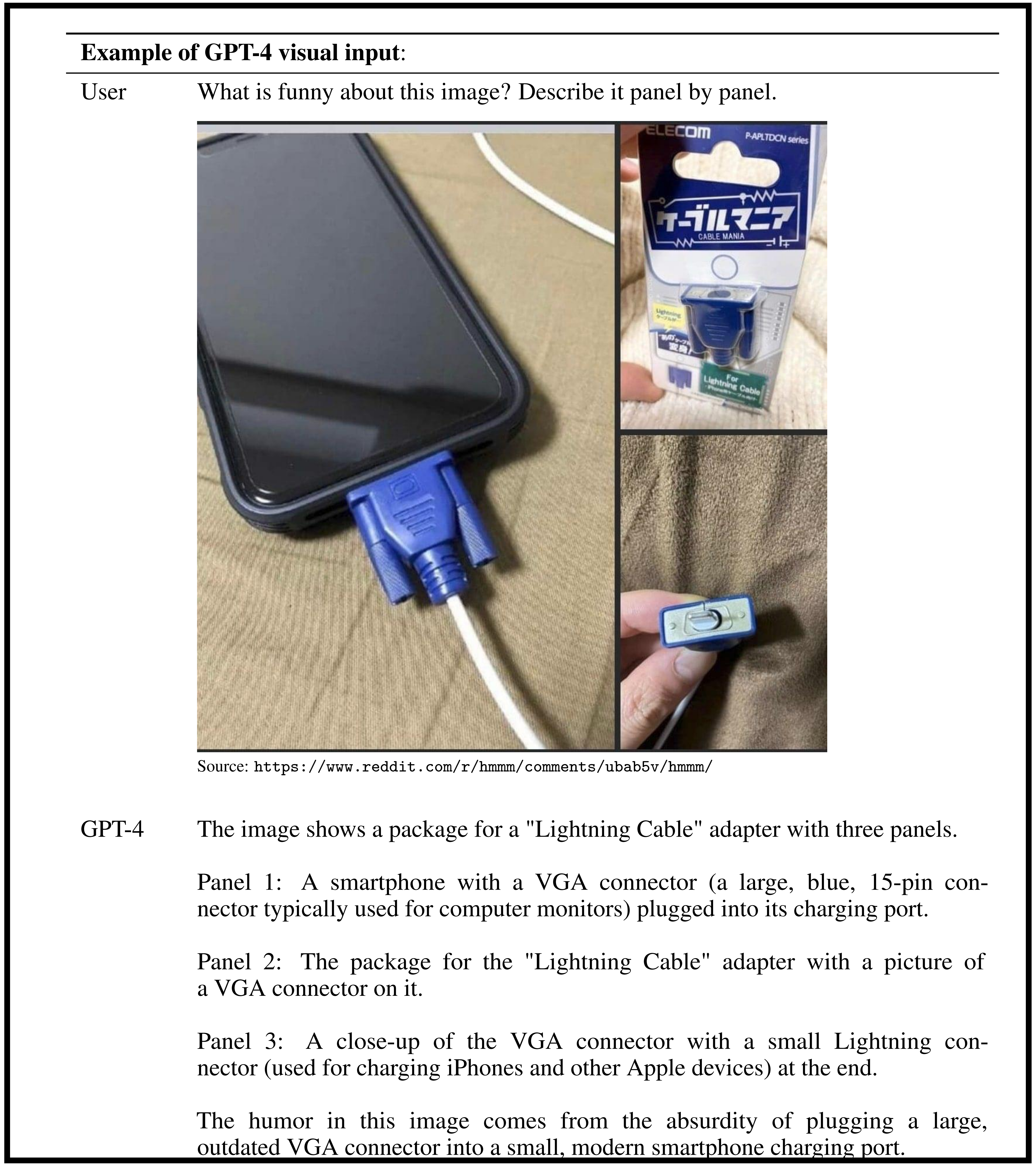
Real-time Machine Learning for ~~triggering~~ at the LHC

making very fast decisions

Thea Klæboe Årrestad (ETH Zürich)
thea.aarrestad@cern.ch

IPA-ML 21.03.2023

Open AI (2023)



Example of GPT-4 visual input:

User	What is funny about this image? Describe it panel by panel.

Source: `https://www.reddit.com/r/hmmm/comments/ubab5v/hmmm/`

GPT-4 The image shows a package for a "Lightning Cable" adapter with three panels.

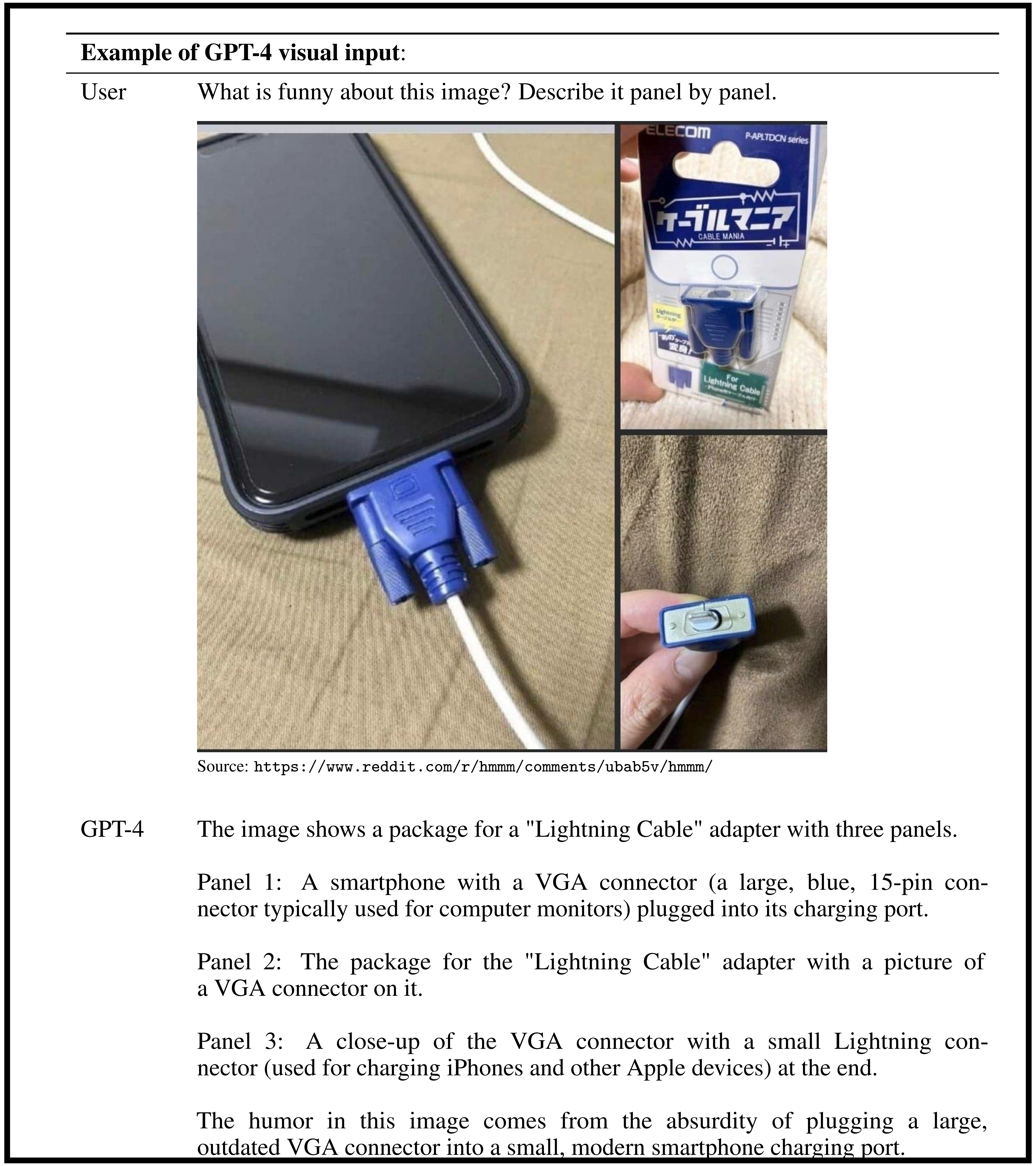
Panel 1: A smartphone with a VGA connector (a large, blue, 15-pin connector typically used for computer monitors) plugged into its charging port.

Panel 2: The package for the "Lightning Cable" adapter with a picture of a VGA connector on it.

Panel 3: A close-up of the VGA connector with a small Lightning connector (used for charging iPhones and other Apple devices) at the end.

The humor in this image comes from the absurdity of plugging a large, outdated VGA connector into a small, modern smartphone charging port.

Open AI (2023)



Example of GPT-4 visual input:

User What is funny about this image? Describe it panel by panel.

Source: `https://www.reddit.com/r/hmmm/comments/ubab5v/hmmm/`

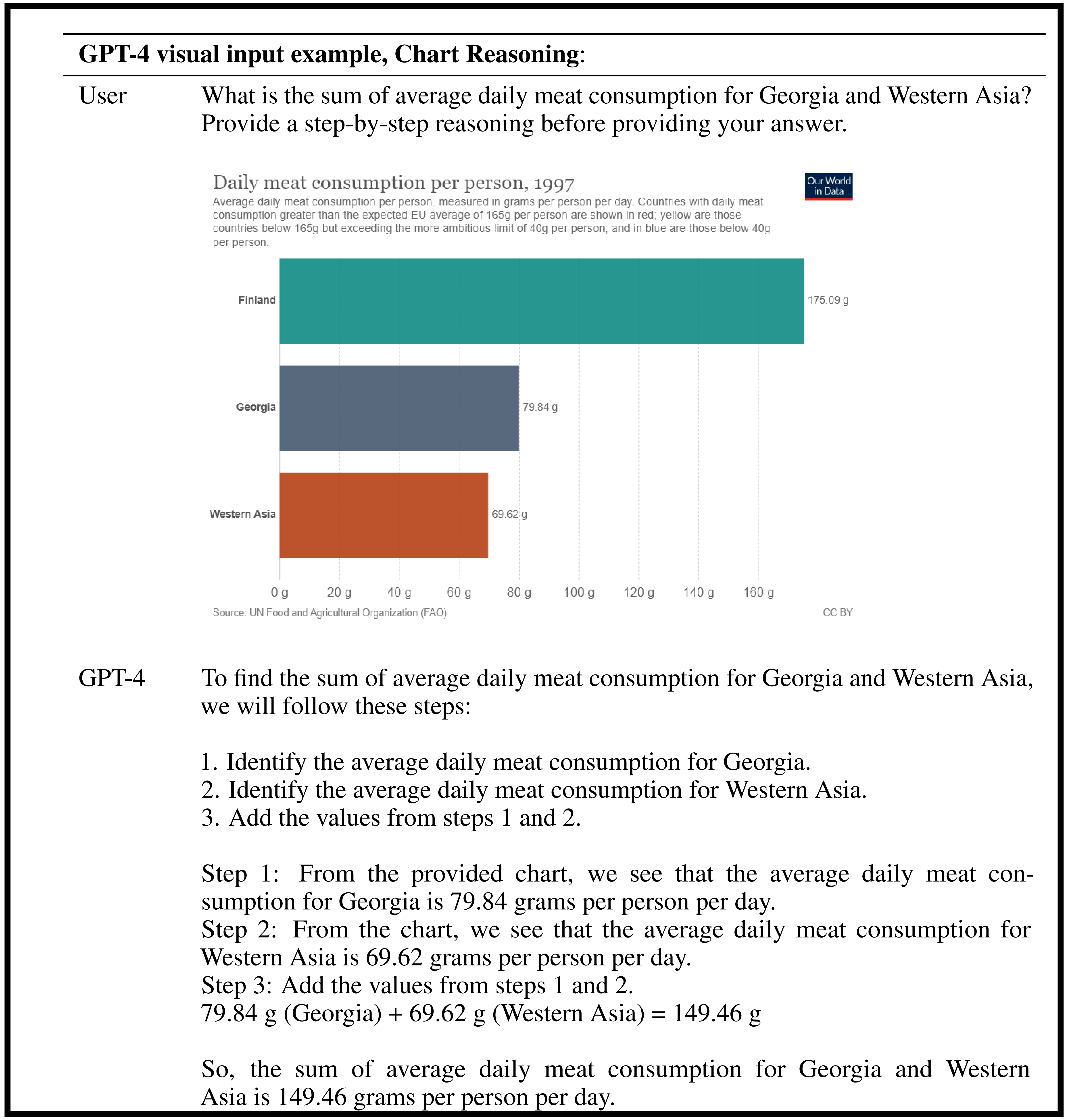
GPT-4 The image shows a package for a "Lightning Cable" adapter with three panels.

Panel 1: A smartphone with a VGA connector (a large, blue, 15-pin connector typically used for computer monitors) plugged into its charging port.

Panel 2: The package for the "Lightning Cable" adapter with a picture of a VGA connector on it.

Panel 3: A close-up of the VGA connector with a small Lightning connector (used for charging iPhones and other Apple devices) at the end.

The humor in this image comes from the absurdity of plugging a large, outdated VGA connector into a small, modern smartphone charging port.



GPT-4 visual input example, Chart Reasoning:

User What is the sum of average daily meat consumption for Georgia and Western Asia? Provide a step-by-step reasoning before providing your answer.

Daily meat consumption per person, 1997
Average daily meat consumption per person, measured in grams per person per day. Countries with daily meat consumption greater than the expected EU average of 165g per person are shown in red; yellow are those countries below 165g but exceeding the more ambitious limit of 40g per person; and in blue are those below 40g per person.
Our World in Data
Finland
175.09 g
Georgia
79.84 g
Western Asia
69.62 g
0 g
20 g
40 g
60 g
80 g
100 g
120 g
140 g
160 g
Source: UN Food and Agricultural Organization (FAO)
CC BY

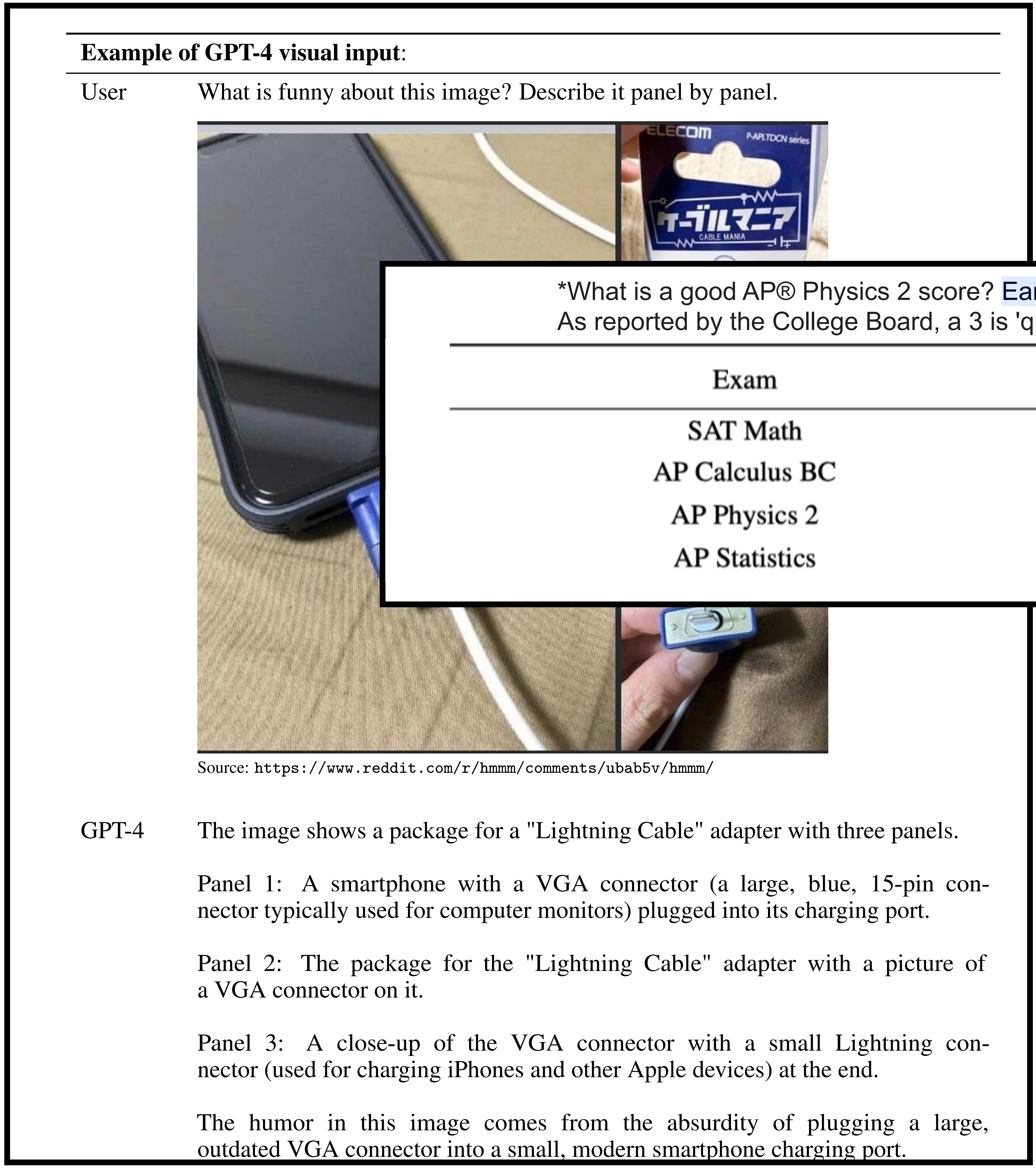
GPT-4 To find the sum of average daily meat consumption for Georgia and Western Asia, we will follow these steps:

1. Identify the average daily meat consumption for Georgia.
2. Identify the average daily meat consumption for Western Asia.
3. Add the values from steps 1 and 2.

Step 1: From the provided chart, we see that the average daily meat consumption for Georgia is 79.84 grams per person per day.
Step 2: From the chart, we see that the average daily meat consumption for Western Asia is 69.62 grams per person per day.
Step 3: Add the values from steps 1 and 2.
79.84 g (Georgia) + 69.62 g (Western Asia) = 149.46 g

So, the sum of average daily meat consumption for Georgia and Western Asia is 149.46 grams per person per day.

Open AI (2023)



Example of GPT-4 visual input:

User What is funny about this image? Describe it panel by panel.

ELECOM
P-APLTDCN series
ケーブルマニア
CABLE MANIA

Source: `https://www.reddit.com/r/hmmm/comments/ubab5v/hmmm/`

GPT-4 The image shows a package for a "Lightning Cable" adapter with three panels.

Panel 1: A smartphone with a VGA connector (a large, blue, 15-pin connector typically used for computer monitors) plugged into its charging port.

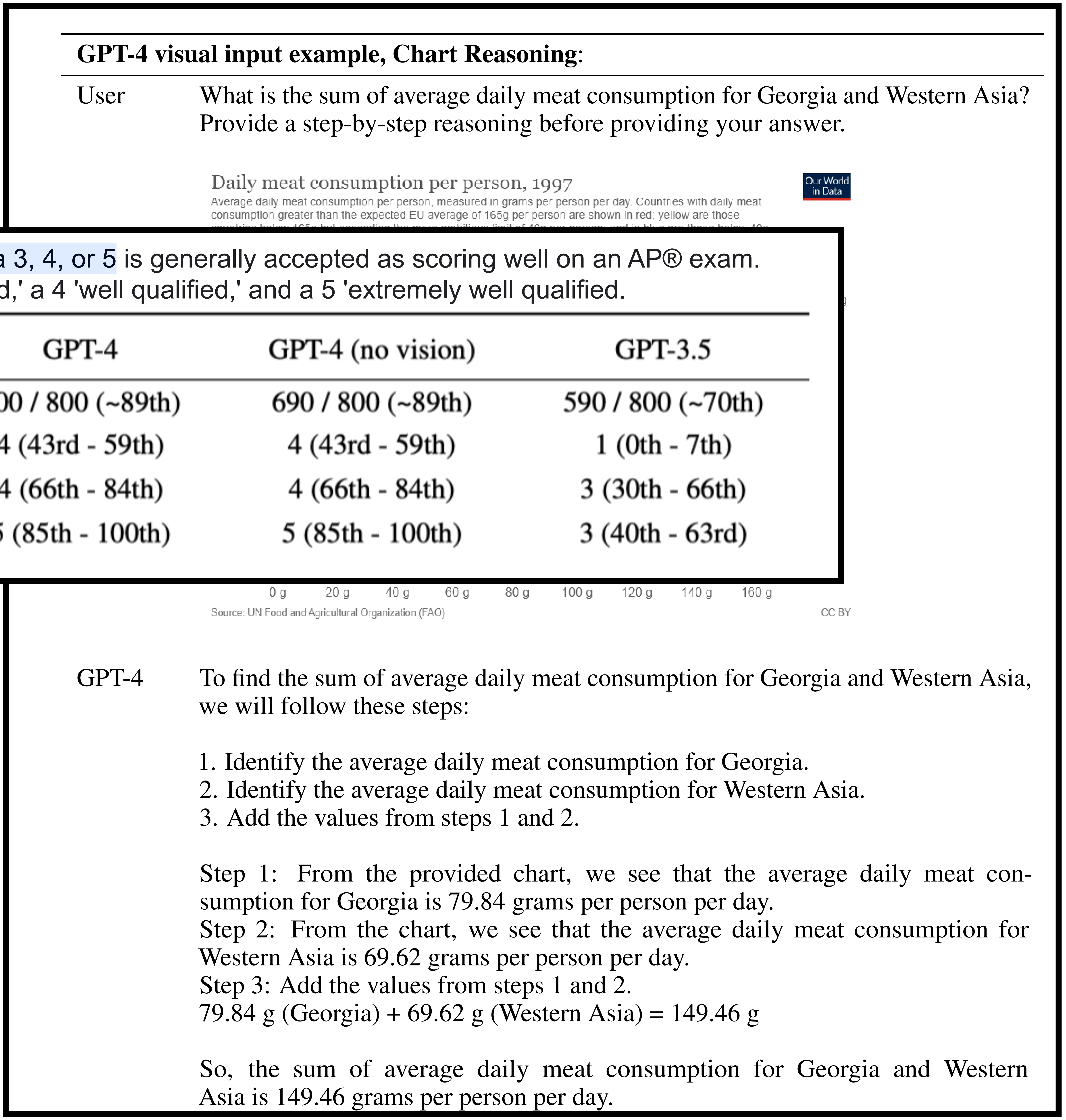
Panel 2: The package for the "Lightning Cable" adapter with a picture of a VGA connector on it.

Panel 3: A close-up of the VGA connector with a small Lightning connector (used for charging iPhones and other Apple devices) at the end.

The humor in this image comes from the absurdity of plugging a large, outdated VGA connector into a small, modern smartphone charging port.

*What is a good AP® Physics 2 score? Earning a 3, 4, or 5 is generally accepted as scoring well on an AP® exam. As reported by the College Board, a 3 is 'qualified,' a 4 'well qualified,' and a 5 'extremely well qualified.'

Exam	GPT-4	GPT-4 (no vision)	GPT-3.5
SAT Math	700 / 800 (~89th)	690 / 800 (~89th)	590 / 800 (~70th)
AP Calculus BC	4 (43rd - 59th)	4 (43rd - 59th)	1 (0th - 7th)
AP Physics 2	4 (66th - 84th)	4 (66th - 84th)	3 (30th - 66th)
AP Statistics	5 (85th - 100th)	5 (85th - 100th)	3 (40th - 63rd)



GPT-4 visual input example, Chart Reasoning:

User What is the sum of average daily meat consumption for Georgia and Western Asia? Provide a step-by-step reasoning before providing your answer.

Daily meat consumption per person, 1997

Average daily meat consumption per person, measured in grams per person per day. Countries with daily meat consumption greater than the expected EU average of 165g per person are shown in red; yellow are those

Source: UN Food and Agricultural Organization (FAO)

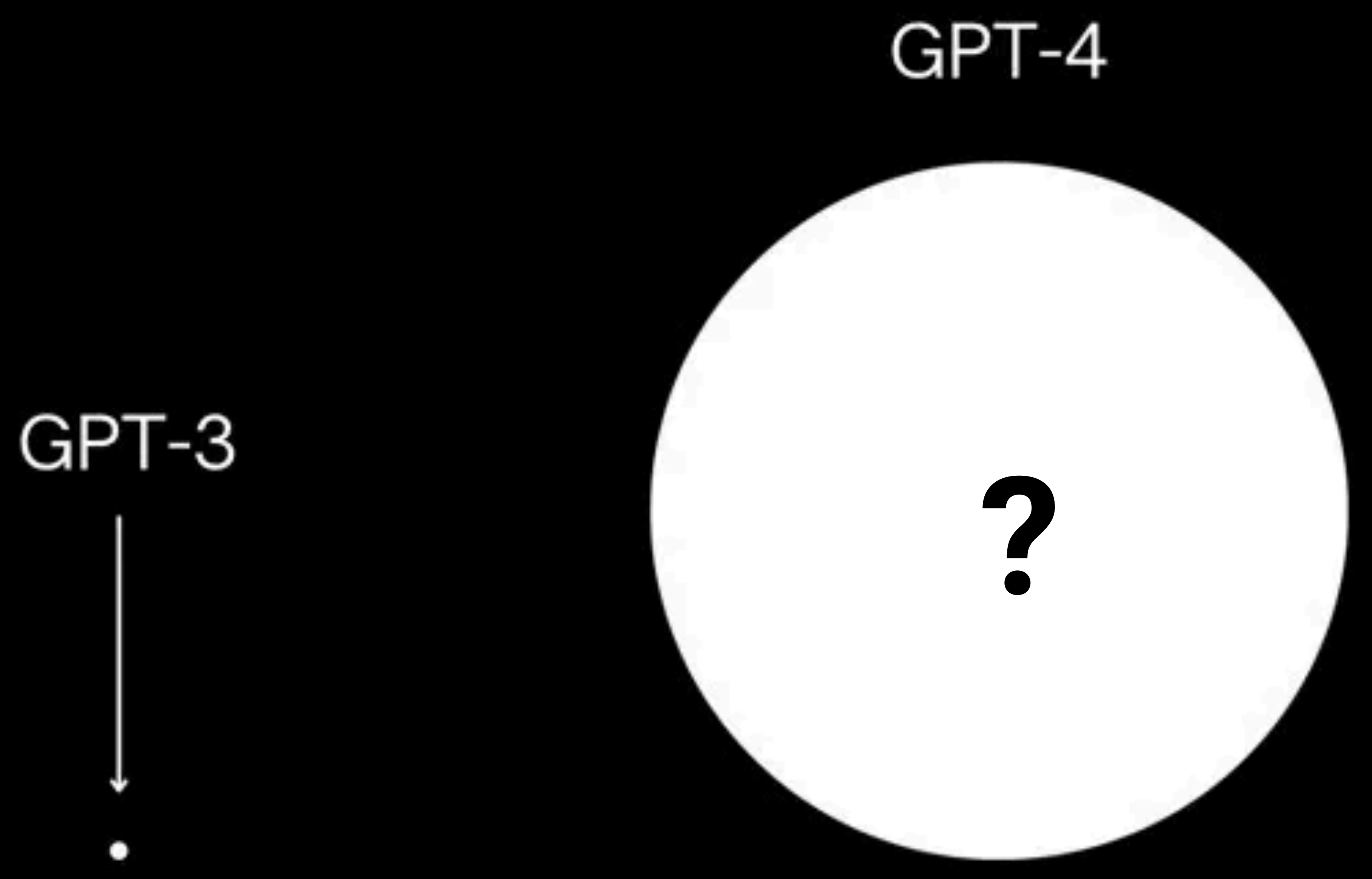
CC BY

GPT-4 To find the sum of average daily meat consumption for Georgia and Western Asia, we will follow these steps:

1. Identify the average daily meat consumption for Georgia.
2. Identify the average daily meat consumption for Western Asia.
3. Add the values from steps 1 and 2.

Step 1: From the provided chart, we see that the average daily meat consumption for Georgia is 79.84 grams per person per day.
Step 2: From the chart, we see that the average daily meat consumption for Western Asia is 69.62 grams per person per day.
Step 3: Add the values from steps 1 and 2.
79.84 g (Georgia) + 69.62 g (Western Asia) = 149.46 g

So, the sum of average daily meat consumption for Georgia and Western Asia is 149.46 grams per person per day.



GPT-3

175,000,000,000

(0.16% of neurons in your brain)

GPT-4

100,000,000,000,000

(All the neurons in your brain?)

NVIDIA Press Release 2020

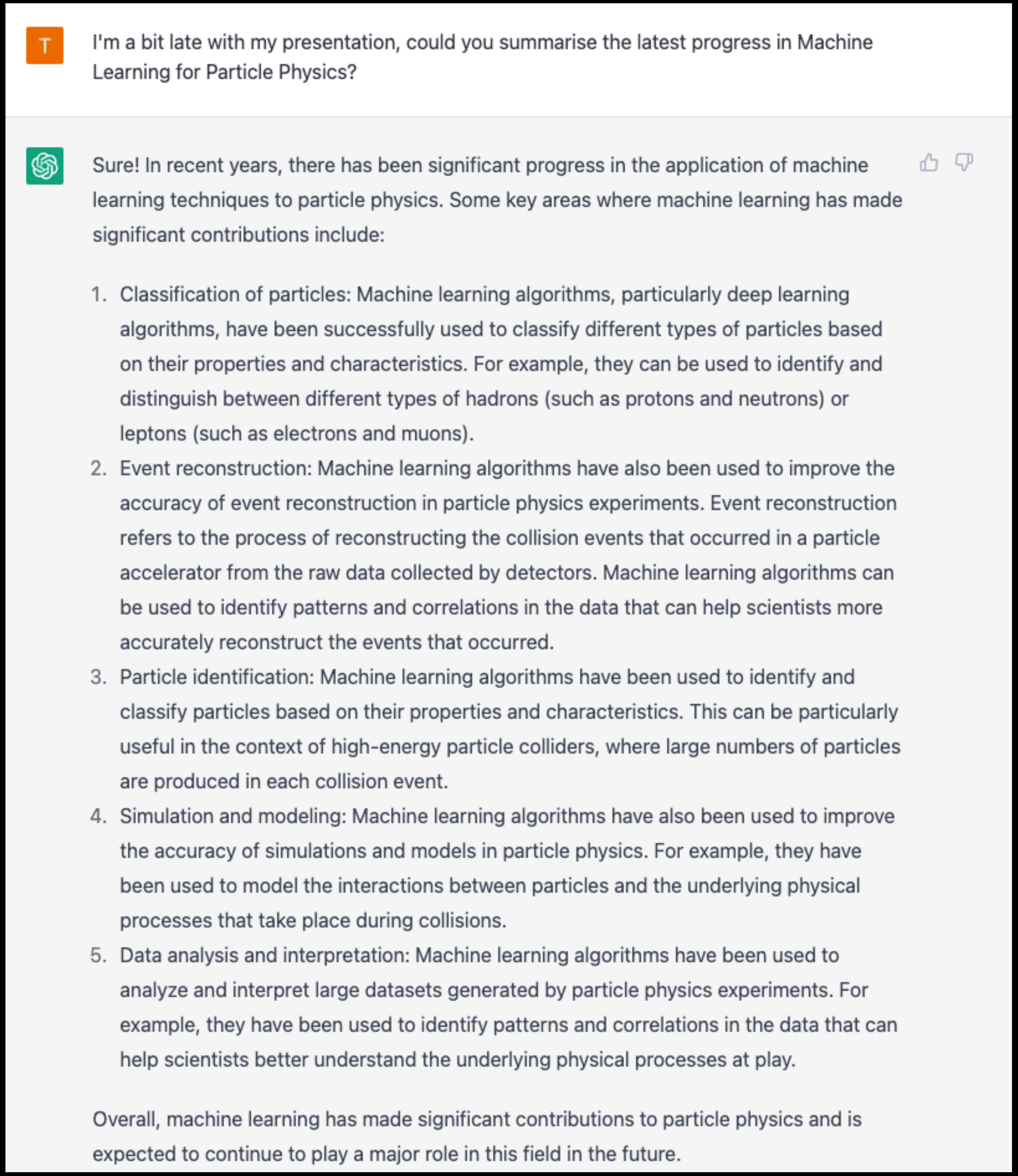
Train (GPT-3):

- 285,000 CPU cores
- 10,000 GPUs
- 400 Gb/s network
- Several weeks
- Trained on ~25 km high book of text

NVIDIA Press Release 2020

Train (GPT-3):

- 285,000 CPU cores
- 10,000 GPUs
- 400 Gb/s network
- Several weeks
- Trained on ~25 km high book of text



I'm a bit late with my presentation, could you summarise the latest progress in Machine Learning for Particle Physics?

Sure! In recent years, there has been significant progress in the application of machine learning techniques to particle physics. Some key areas where machine learning has made significant contributions include:

1. Classification of particles: Machine learning algorithms, particularly deep learning algorithms, have been successfully used to classify different types of particles based on their properties and characteristics. For example, they can be used to identify and distinguish between different types of hadrons (such as protons and neutrons) or leptons (such as electrons and muons).
2. Event reconstruction: Machine learning algorithms have also been used to improve the accuracy of event reconstruction in particle physics experiments. Event reconstruction refers to the process of reconstructing the collision events that occurred in a particle accelerator from the raw data collected by detectors. Machine learning algorithms can be used to identify patterns and correlations in the data that can help scientists more accurately reconstruct the events that occurred.
3. Particle identification: Machine learning algorithms have been used to identify and classify particles based on their properties and characteristics. This can be particularly useful in the context of high-energy particle colliders, where large numbers of particles are produced in each collision event.
4. Simulation and modeling: Machine learning algorithms have also been used to improve the accuracy of simulations and models in particle physics. For example, they have been used to model the interactions between particles and the underlying physical processes that take place during collisions.
5. Data analysis and interpretation: Machine learning algorithms have been used to analyze and interpret large datasets generated by particle physics experiments. For example, they have been used to identify patterns and correlations in the data that can help scientists better understand the underlying physical processes at play.

Overall, machine learning has made significant contributions to particle physics and is expected to continue to play a major role in this field in the future.

Inference (GPT-3):

- ?

AI and Memory Wall

Transformer Size: 240x / 2 yrs

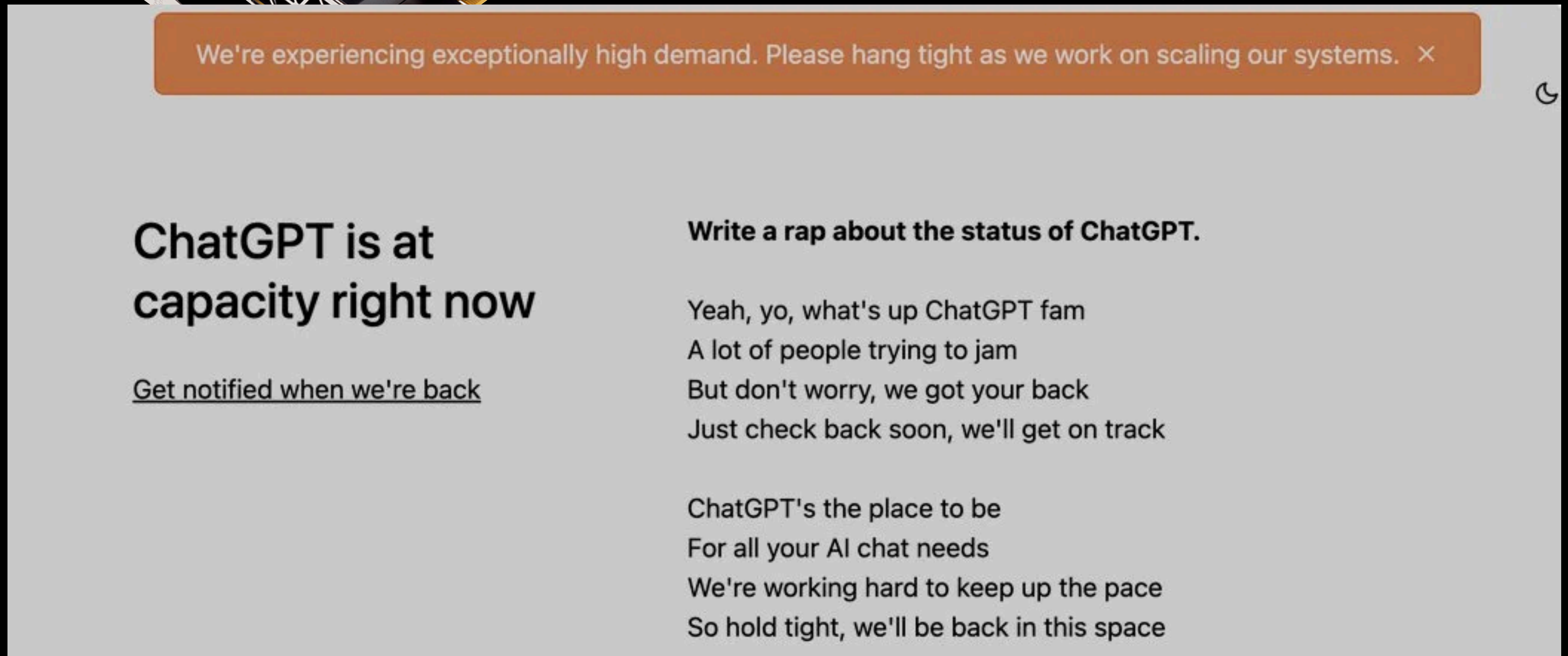
AI HW Memory: 2x / 2 yrs

x11 of these to fit one GPT-3 at inference time!
Price each: $10,000

Parameter Count (Billion)
10000
1000
100
10
1
0.1
0.01
YEAR
2016
2017
2018
2019
2020
2021
10TB Baidu RecSys
2TB Baidu RecSys
GShard
GPT-3
Microsoft T-NLG
A100-80 (80GB)
Megatron LM
A100 (40GB)
V100 (32GB)
TPUv3 (32GB)
TPUv2 (16GB)
P100 (12GB)
GPT-2
BERT
ALBERT
ELECTRA
GPT-1
Transformer
Inception V4
ResNext101
ResNet50
DenseNet

Resources: *11 interconnected GPUs*

Latency : *10^1 seconds*



We're experiencing exceptionally high demand. Please hang tight as we work on scaling our systems.

ChatGPT is at capacity right now

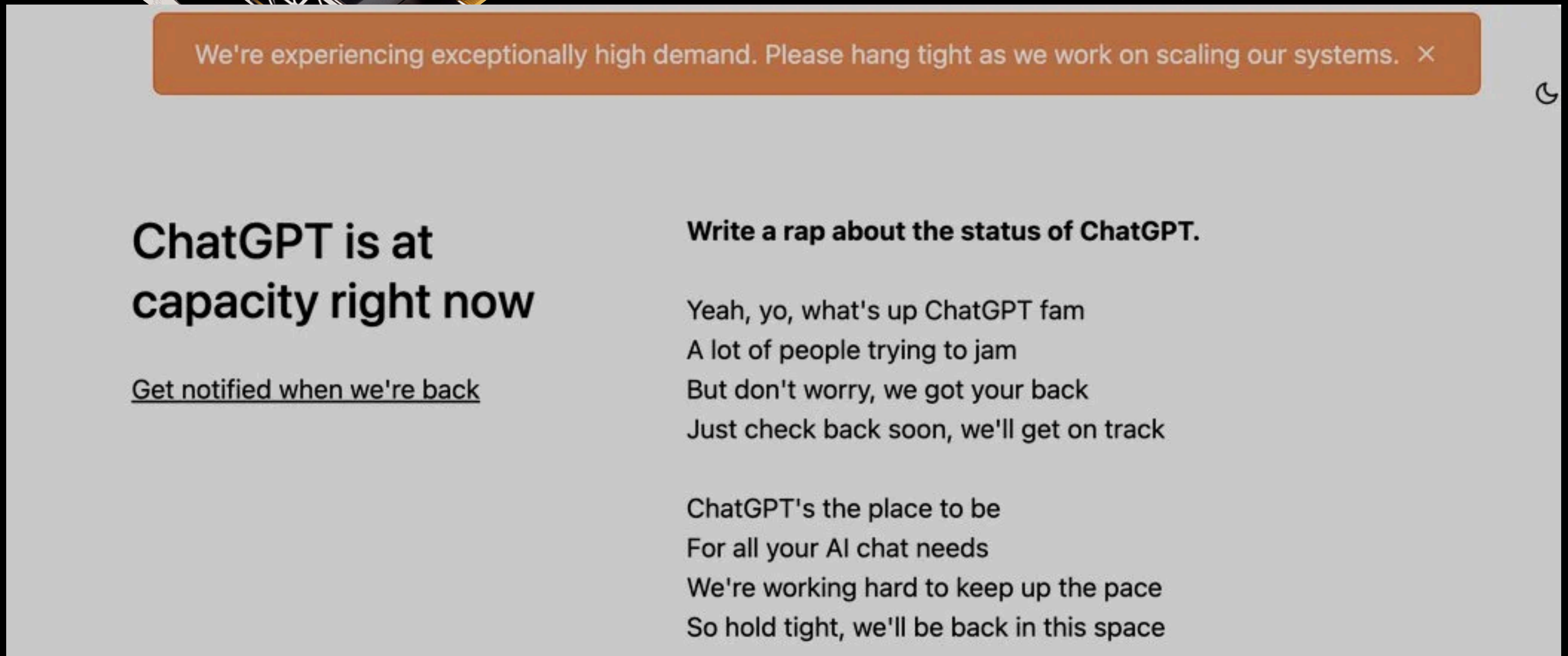
Get notified when we're back

Write a rap about the status of ChatGPT.

Yeah, yo, what's up ChatGPT fam
A lot of people trying to jam
But don't worry, we got your back
Just check back soon, we'll get on track

ChatGPT's the place to be
For all your AI chat needs
We're working hard to keep up the pace
So hold tight, we'll be back in this space

Resources: ***11 interconnected GPUs***
Latency : ***10^1 seconds***



We're experiencing exceptionally high demand. Please hang tight as we work on scaling our systems. ×

ChatGPT is at capacity right now

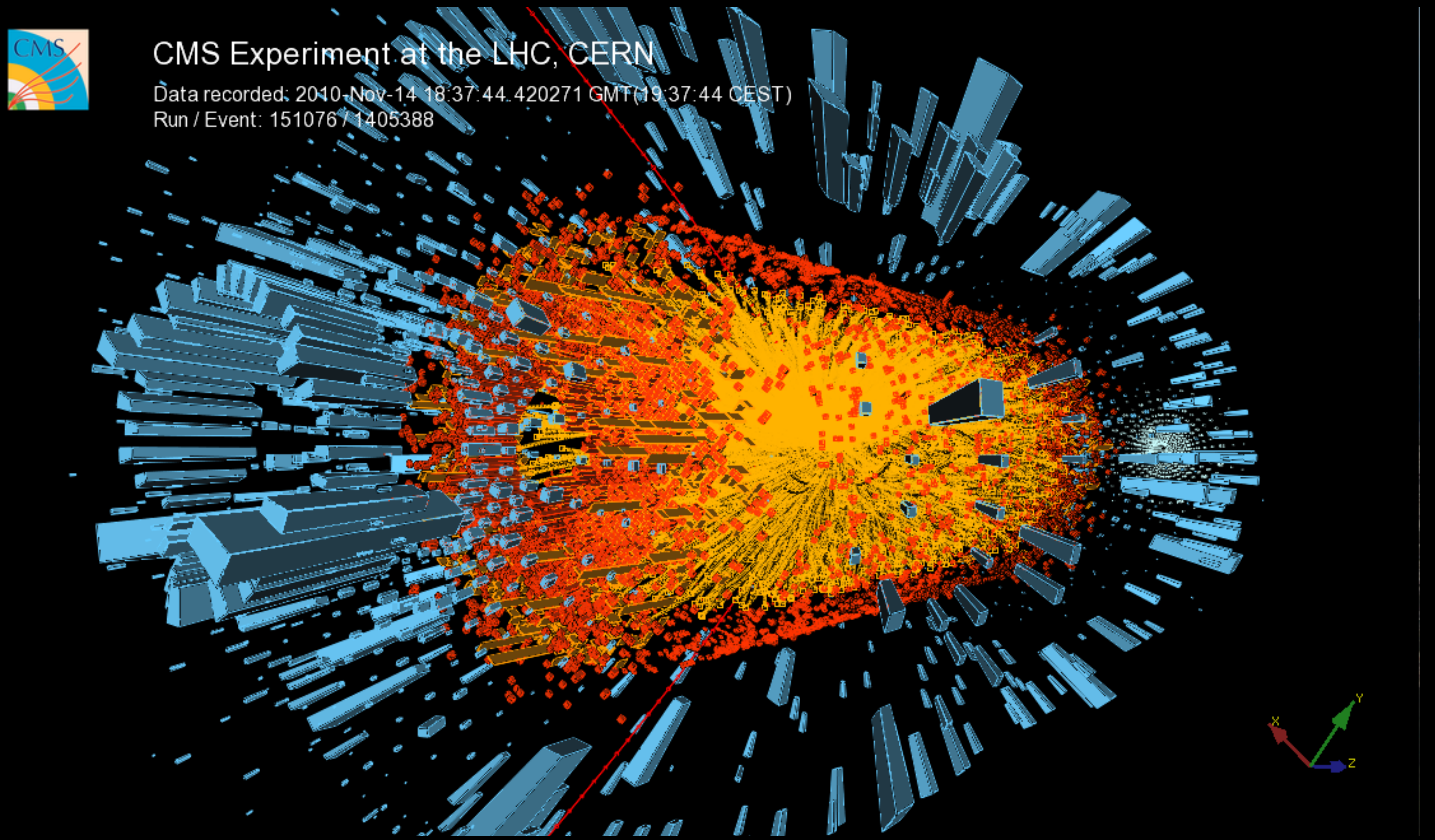
Get notified when we're back

Write a rap about the status of ChatGPT.

Yeah, yo, what's up ChatGPT fam
A lot of people trying to jam
But don't worry, we got your back
Just check back soon, we'll get on track

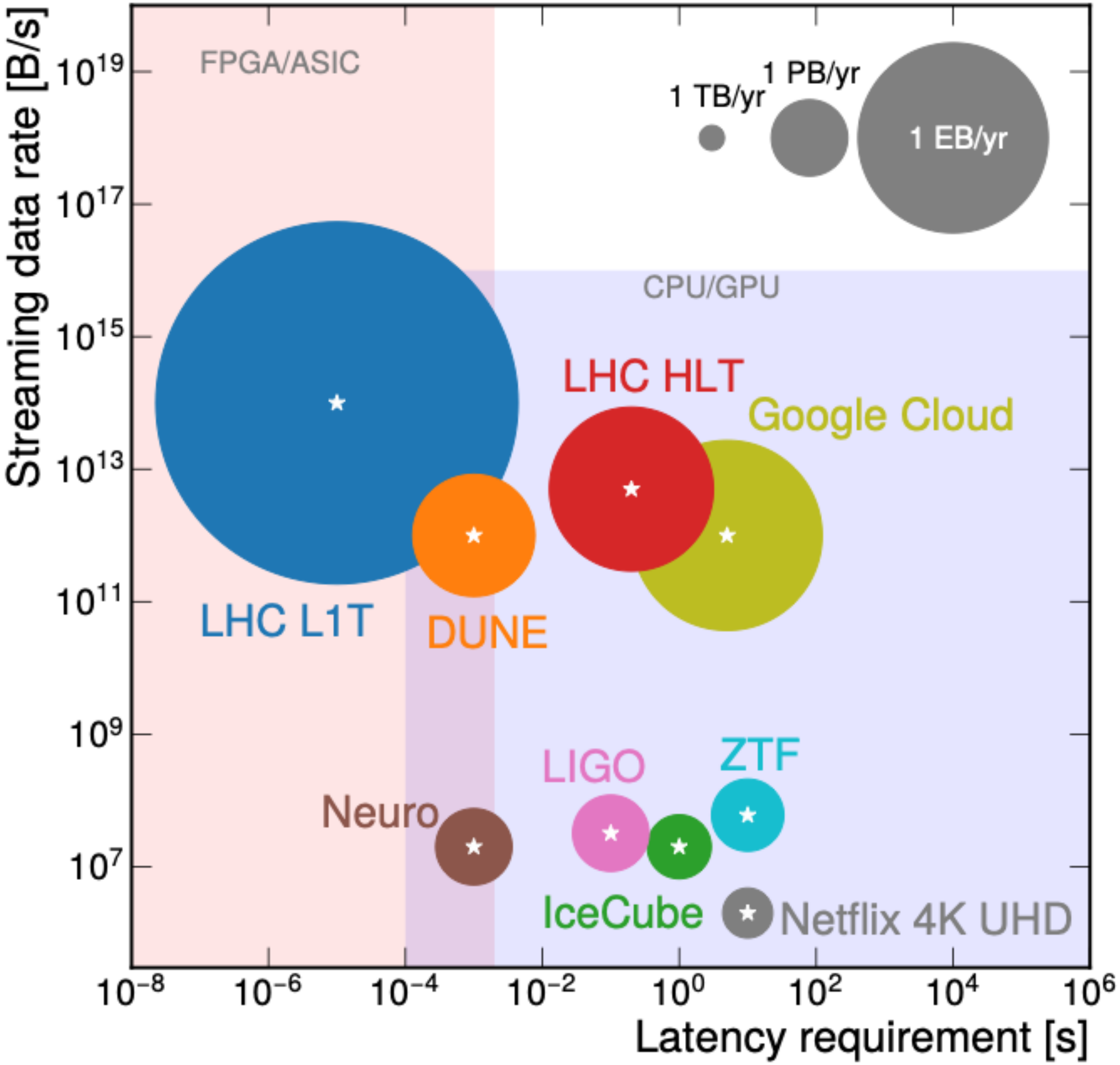
ChatGPT's the place to be
For all your AI chat needs
We're working hard to keep up the pace
So hold tight, we'll be back in this space

Resources: ***One single chip***
Latency: ***10^{-9} seconds***


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
X
Y
Z

A3D3 Institute

https://a3d3.ai/



LHCb
ATLAS
CERN Meyrin
PS 628 m
CERN Prévessin
SPS 7 km
SUISSE
FRANCE
ALICE
CMS
LHC 27 km

2.2·10¹¹ protons
25 ns
7.5 m

HL-LHC:
Up to 200 pp collisions per crossing!

CMS Experiment at the LHC, CERN

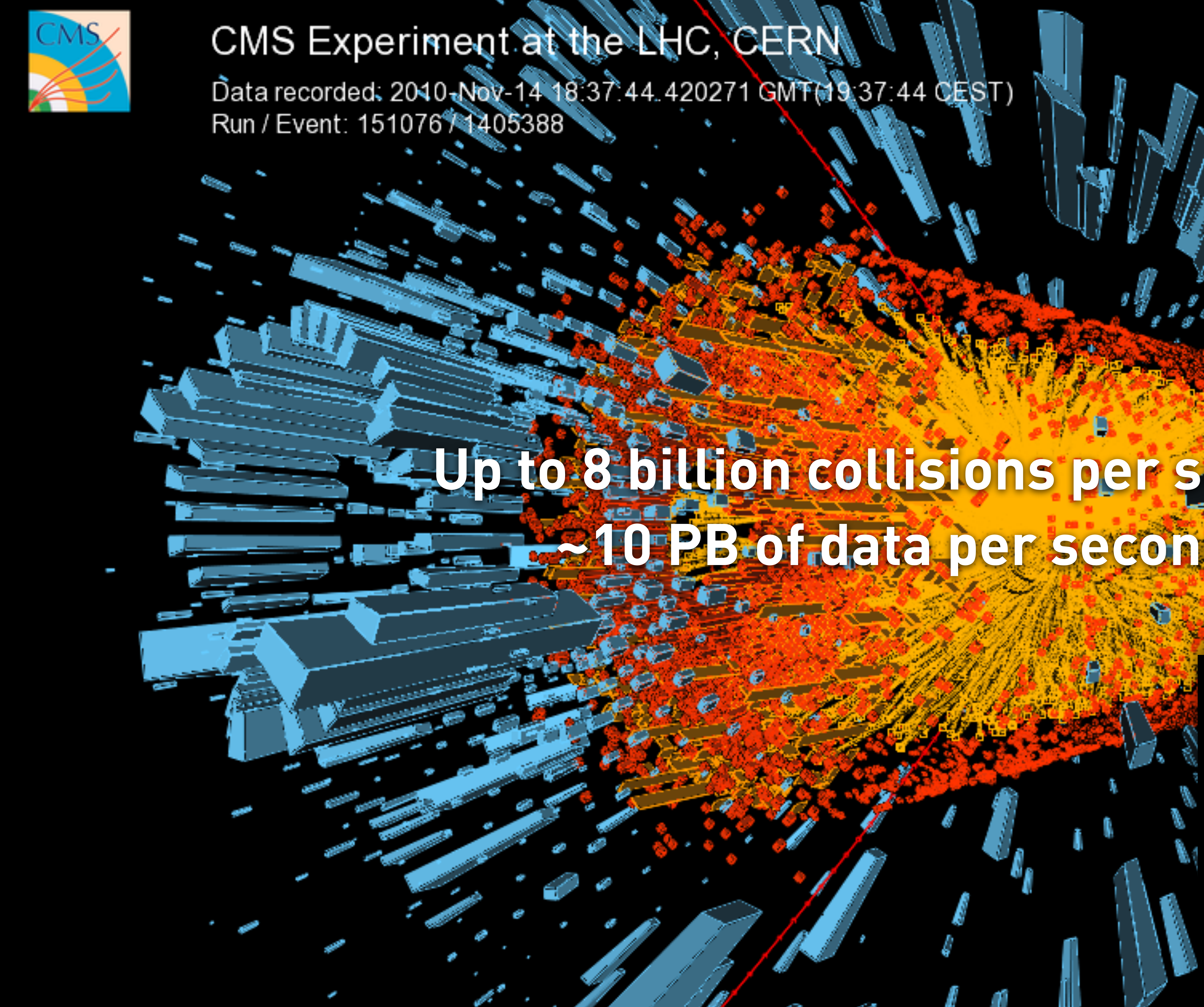
Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)

Run / Event: 151076 / 1405388

CMS Experiment at the LHC, CERN
Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)
Run / Event: 151076 / 1405388
X
Y
Z

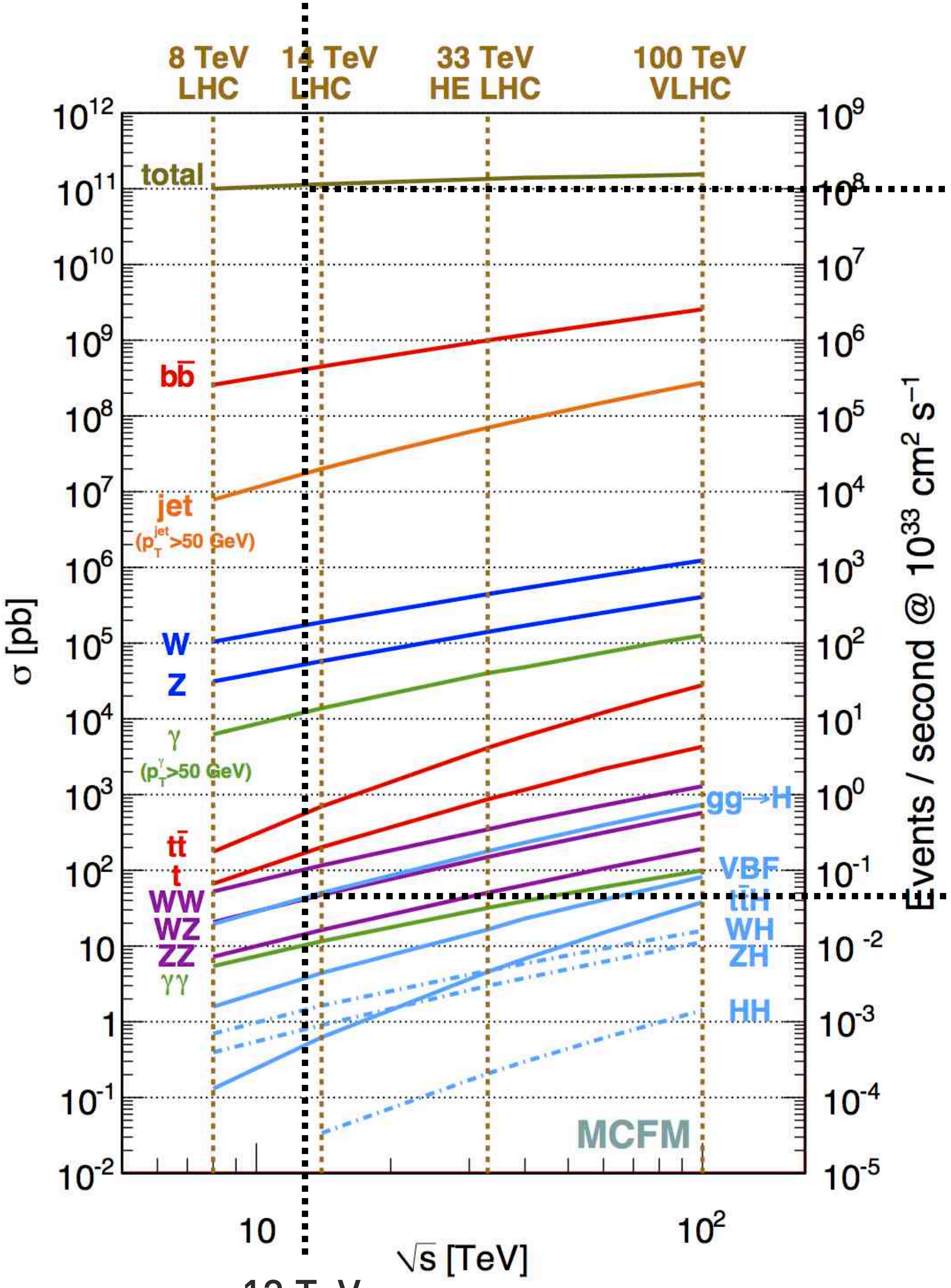
Up to 8 billion collisions per second

~10 PB of data per second


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

Up to 8 billion collisions per second

~10 PB of data per second



Total

13 TeV

Higgs produced
~1 in a billion collisions

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

gg→H

Geneva
Geneva Lake
LHCb
ATLAS
ALICE
CMS

Billions of collisions/s
Petabytes of data/s

Geneva
Geneva Lake
LHCb
AT
CMS
LHC

Data temporarily stored INSIDE detector for ~3-12 μs

63 Tb/s to L1

Level-1 trigger: Get rid of >98% of collisions!

Geneva
Geneva
LHCb
ATLAS
ALICE
CMS
LHC

TIER 0: ∞

High Level Trigger:
Latency O(100) ms

~2% of events remain
750 kHz
Tb/s

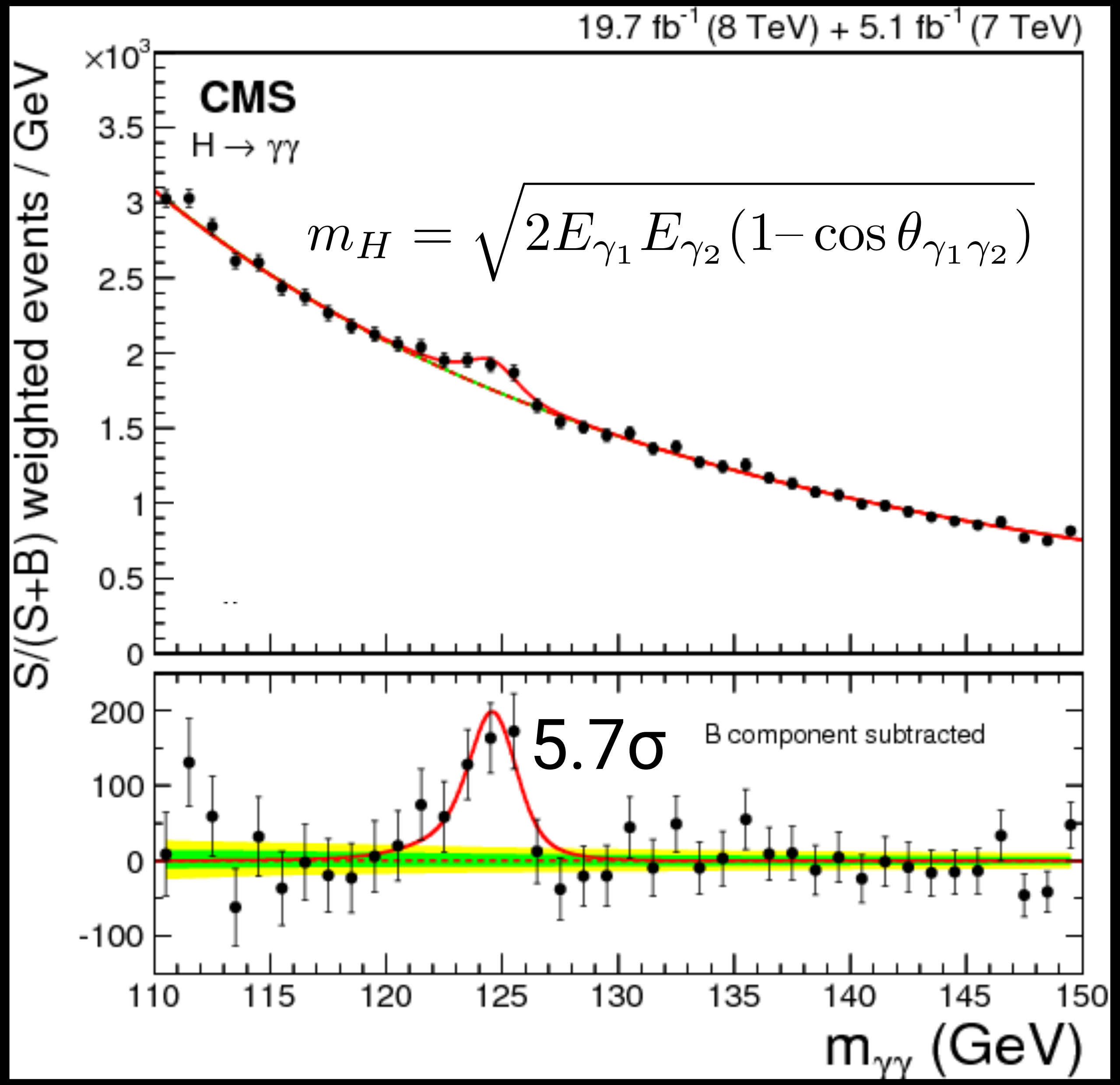
100% of events remain
40 MHz
Pb/s

Geneva
Geneva Lake
LHCb
ATLAS
ALICE
CMS
LHC

TIER 0: ∞
Processing and permanent storage

0.02% of events remain :(
7.5 kHz
Gb/s

ALL OF OUR PHYSICS RESULTS USE ONLY ~0.02% OF COLLISION DATA!



To make sure we select “the right” 0.02%, algorithms must be

- Fast (get more data through)
- Accurate (select the right data)

CMS Experiment at the LHC, CERN

Data recorded: 2010-Nov-14 18:37:44.420271 GMT(19:37:44 CEST)

Run / Event: 151076 / 1405388

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
X
Y
Z

→ Utilise modern Machine Learning to become
faster
better
and do more

Geneva
LHCb
ATLAS
ALICE
CMS
LHC
TESLA
intel
Xeon® 7500
XILINX
VIRTEX-5

High Level Trigger:
Software
(Latency ~ms)

Level-1 trigger:
Firmware
FPGAs (Latency ~μs)

Inside detector:
Hardware
ASIC (Latency ~ns)

Geneva
LHCb
ATLAS
ALICE
CMS
LHC
TESLA
intel
Xeon® 7500
VIRTEX

High Level Trigger:
Software
(Latency ~ms)

Nanosecond ML inference
on specialised hardware for triggering!

Level-1 trigger:
Firmware
FPGAs (Latency ~μs)

Inside detector:
Hardware
ASIC (Latency ~ns)

LHCb
ATLAS
ALICE
CMS
LHC
TESLA
intel
Xeon® 7500
VIRTEX-5

Not covered in this talk. Great work here by S. Hasan (IPA)

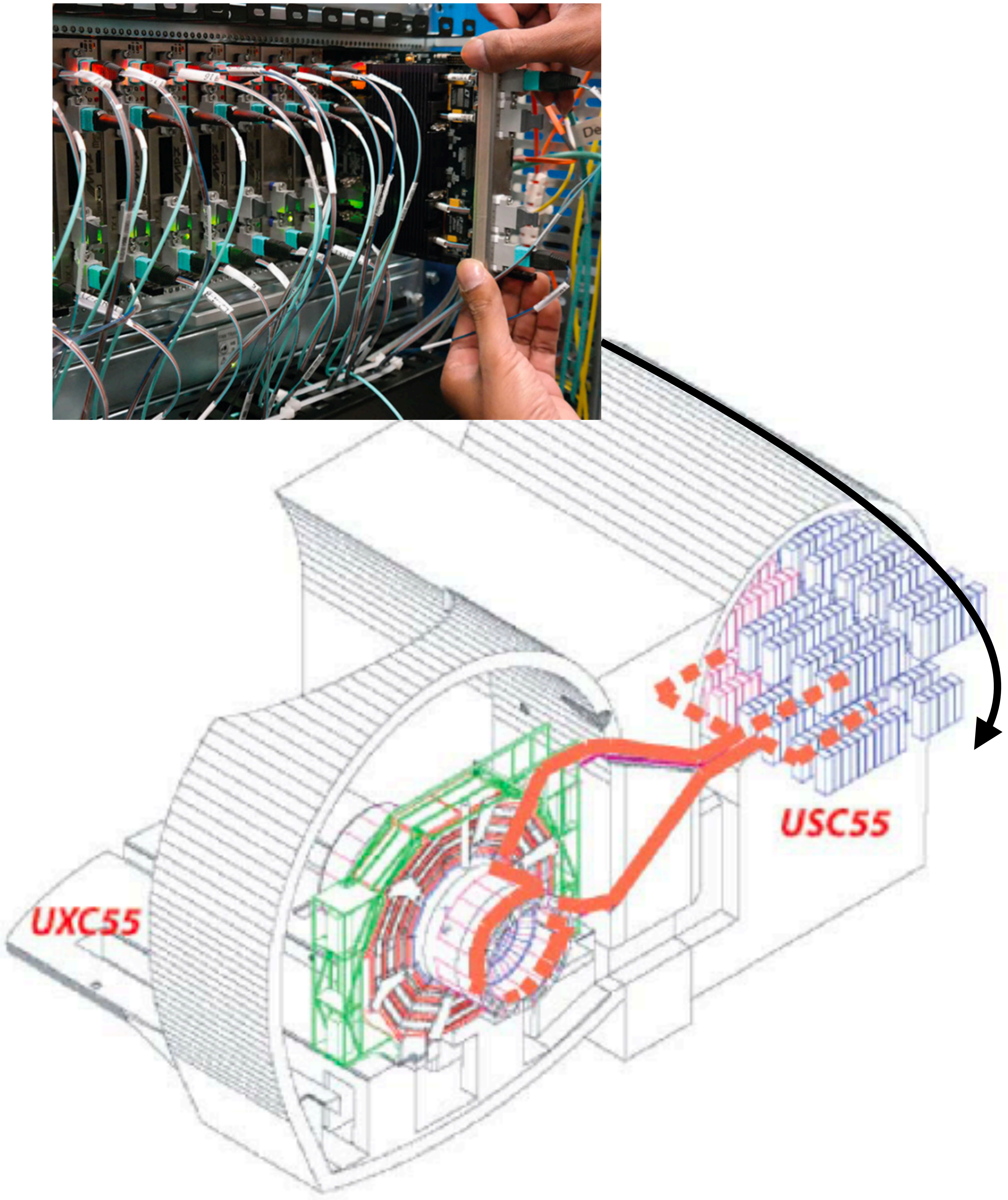
High Level Trigger:
Software
(Latency ~ms)

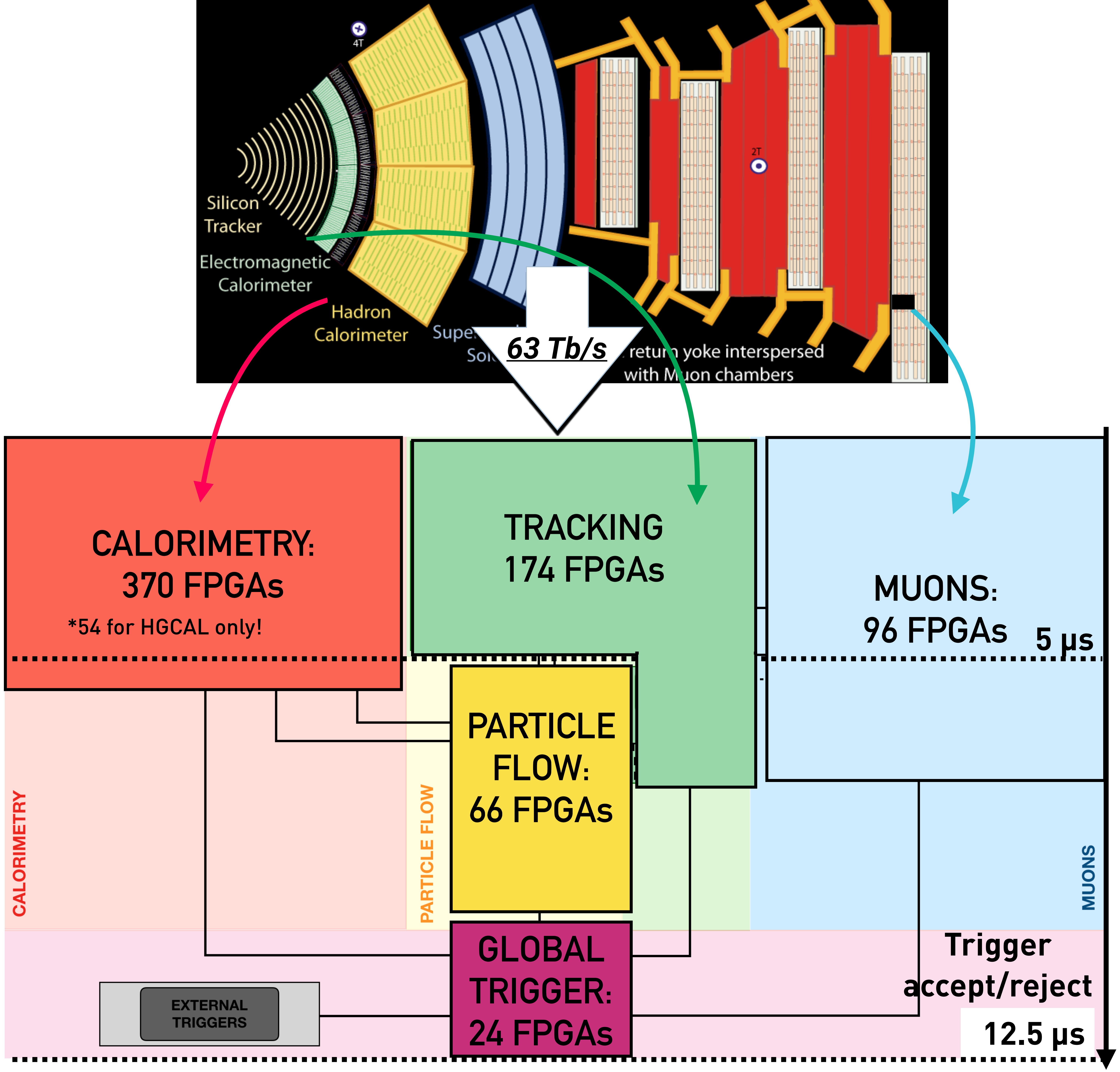
Nanosecond ML inference
on specialised hardware for triggering!

Level-1 trigger:
Firmware
FPGAs (Latency ~μs)

Inside detector:
Hardware
ASIC (Latency ~ns)

The Level-1 trigger





63 Tb/s

CALORIMETRY:
370 FPGAs
*54 for HGCAL only!

TRACKING
174 FPGAs

MUONS:
96 FPGAs

5 µs

PARTICLE FLOW:
66 FPGAs

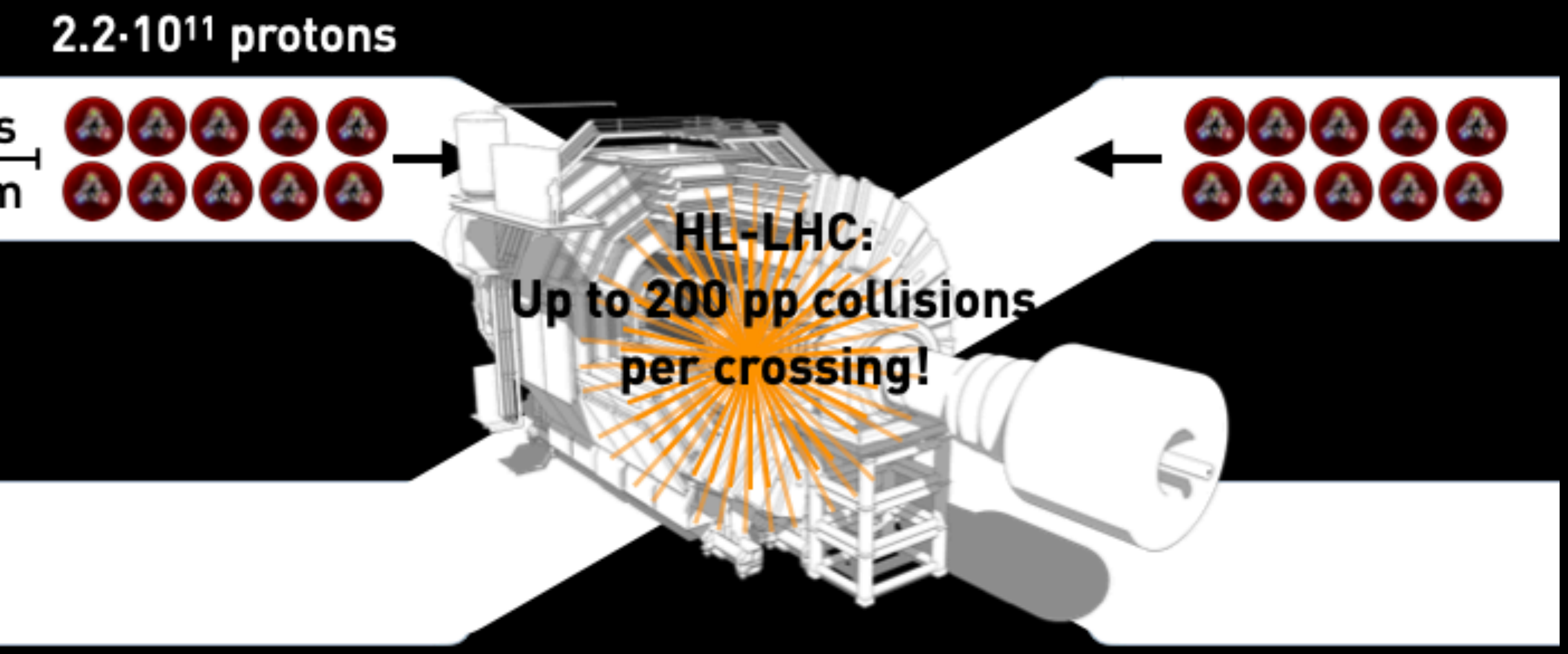
GLOBAL TRIGGER:
24 FPGAs

Trigger accept/reject

12.5 µs

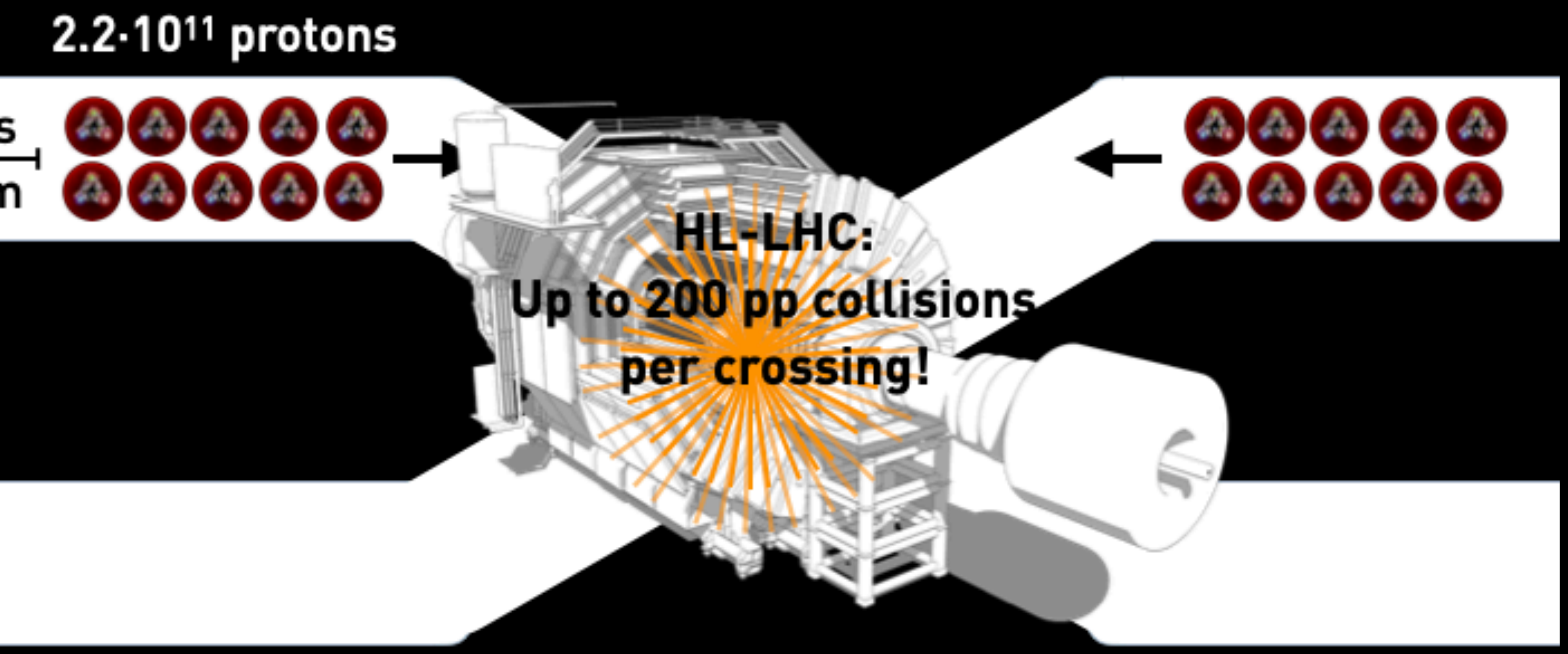
Low latency

- Strictly limited by collisions occurring every 25 ns


$2.2\cdot10^{11}$ protons
HL-LHC:
Up to 200 pp collisions per crossing!

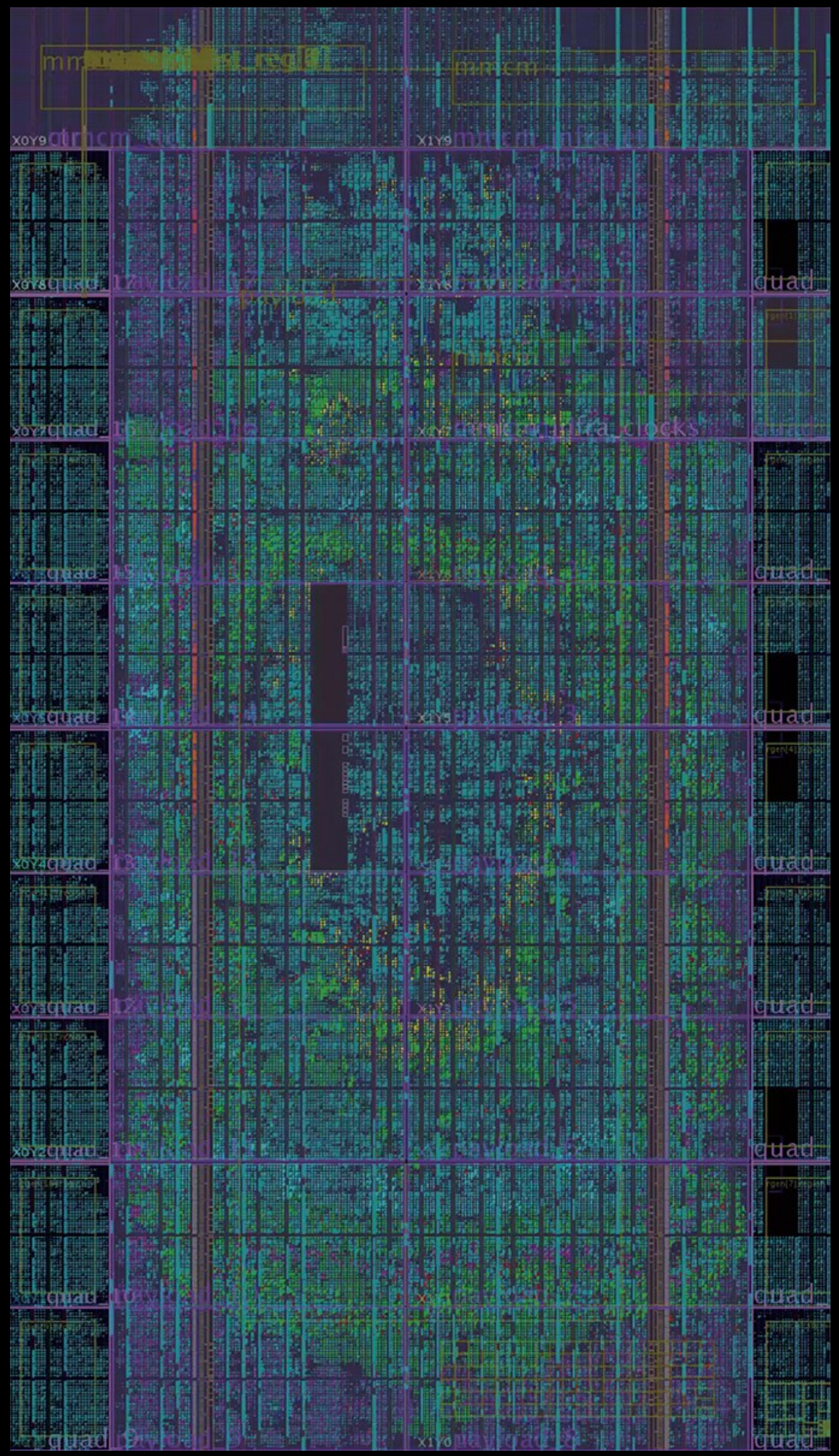
Low latency

- Strictly limited by collisions occurring every 25 ns


$2.2 \cdot 10^{11}$ protons
HL-LHC:
Up to 200 pp collisions per crossing!

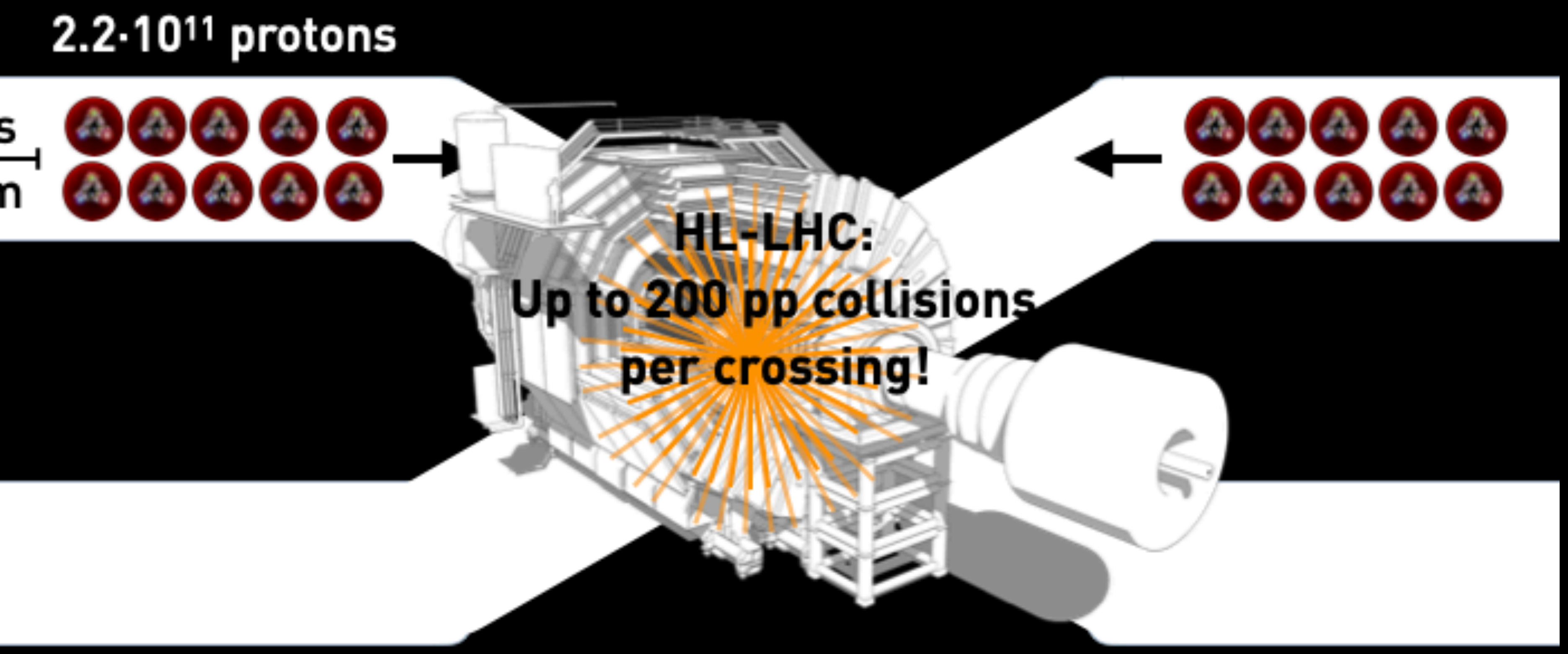
Low resource usage

- Several algorithms in parallel on single device



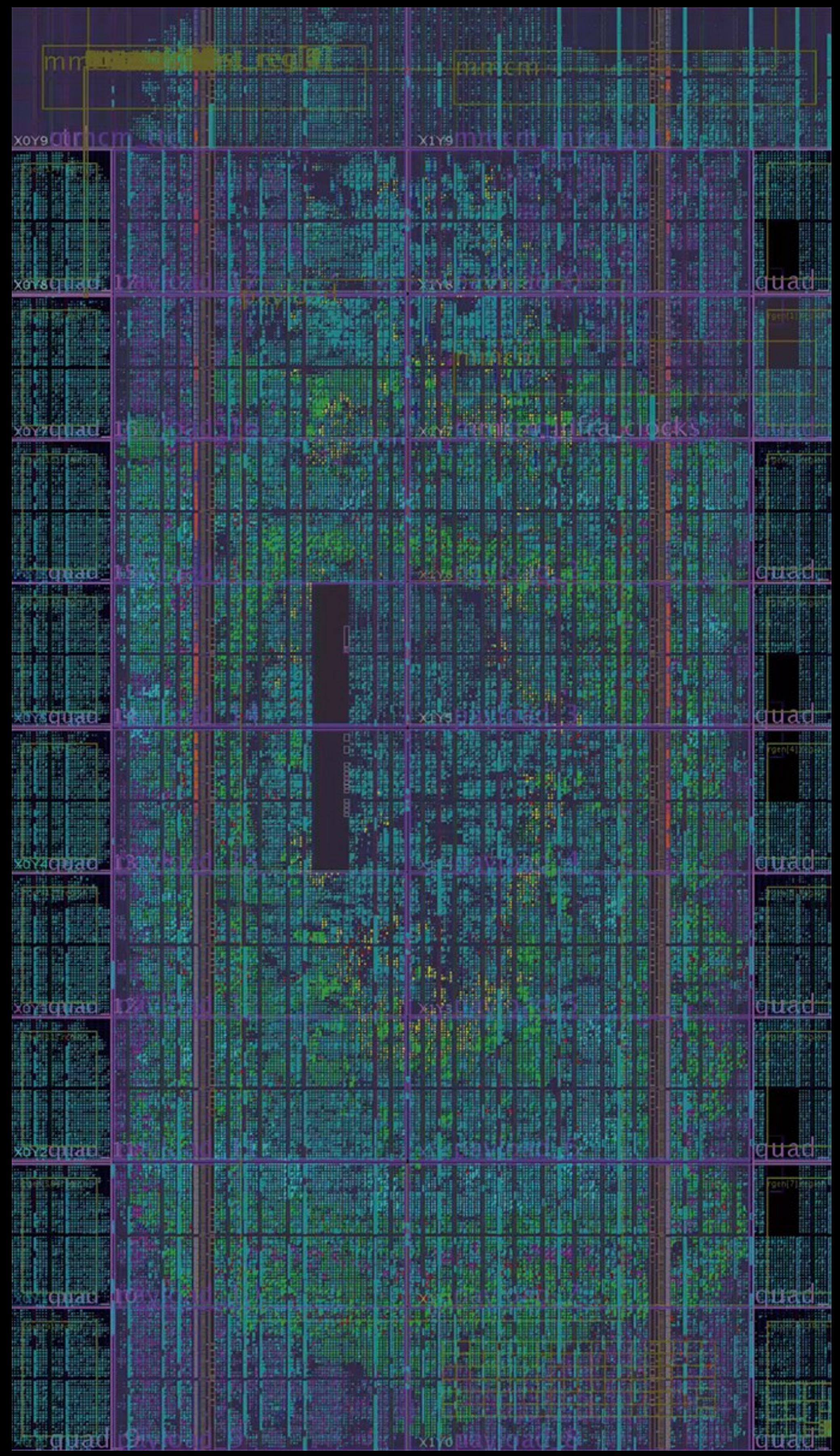
Low latency

- Strictly limited by collisions occurring every 25 ns


$2.2 \cdot 10^{11}$ protons
HL-LHC:
Up to 200 pp collisions per crossing!

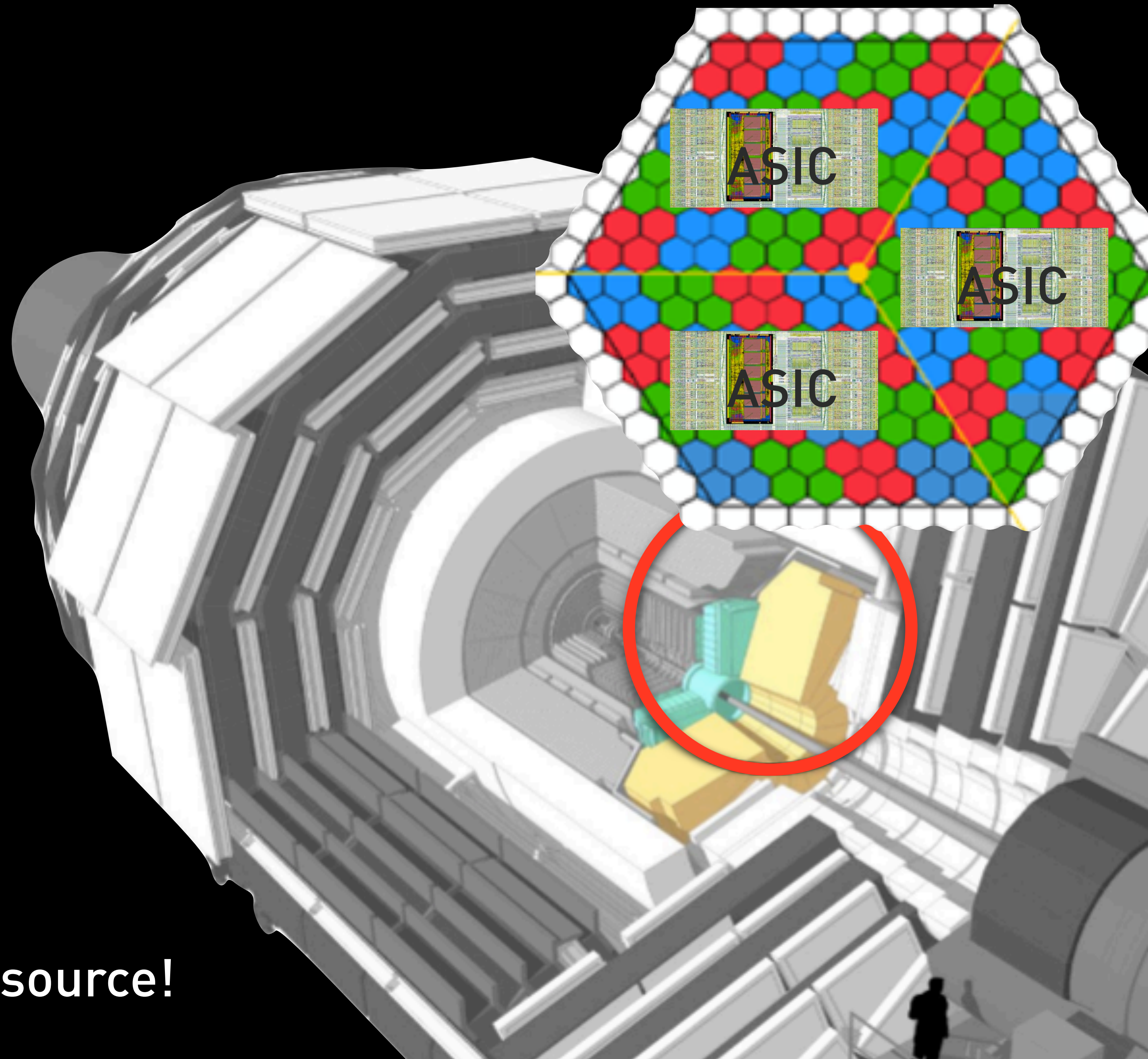
Low resource usage

- Several algorithms in parallel on single device



Power efficient

- On detector: Detector is cooled! Limited to mW



ML algorithms here must be low power, low latency, low resource!

Why FPGAs at LHC?

Why FPGAs at LHC?

High parallelism = **Low latency**

- Can work on different data simultaneously (pipelining)! **High bandwidth**

Why FPGAs at LHC?

High parallelism = **Low latency**

- Can work on different data simultaneously (pipelining)! **High bandwidth**

Power efficient

- FPGAS ~x10 more power efficient than GPUs
  (even our FPGAs dissipate heat of ~7W/cm^2 while processing 5% of total internet traffic!)

Why FPGAs at LHC?

High parallelism = **Low latency**

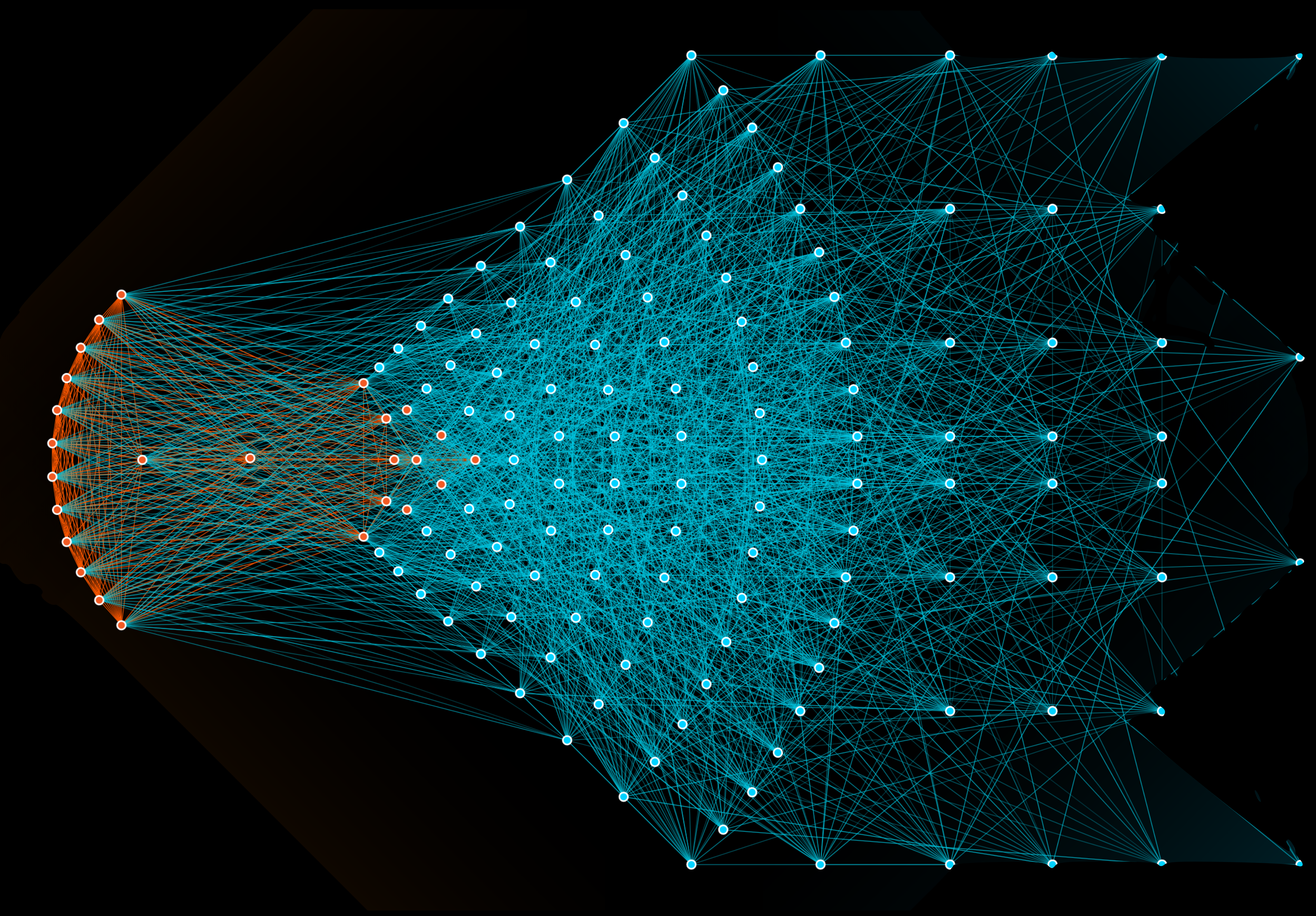
- Can work on different data simultaneously (pipelining)! **High bandwidth**

Power efficient

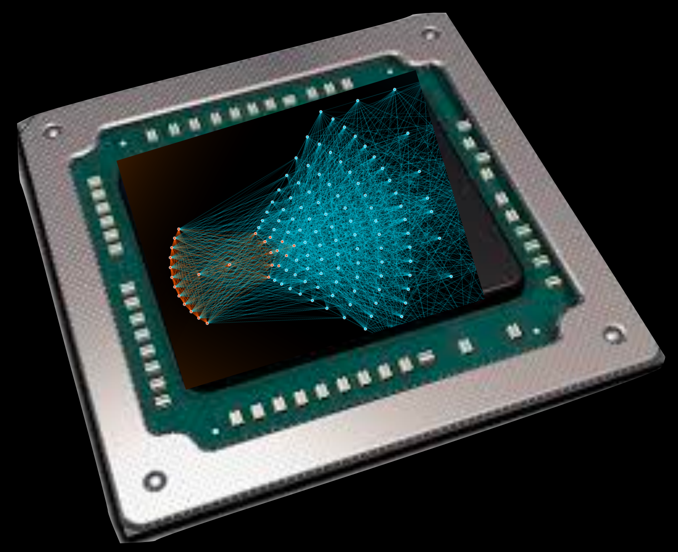
- FPGAS ~x10 more power efficient than GPUs
  (even our FPGAs dissipate heat of ~7W/cm^2 while processing 5% of total internet traffic!)

Latency deterministic

- CPU/GPU has processing randomness, FPGAs **repeatable and predictable latency**



Ideally



Reality

Edge inference

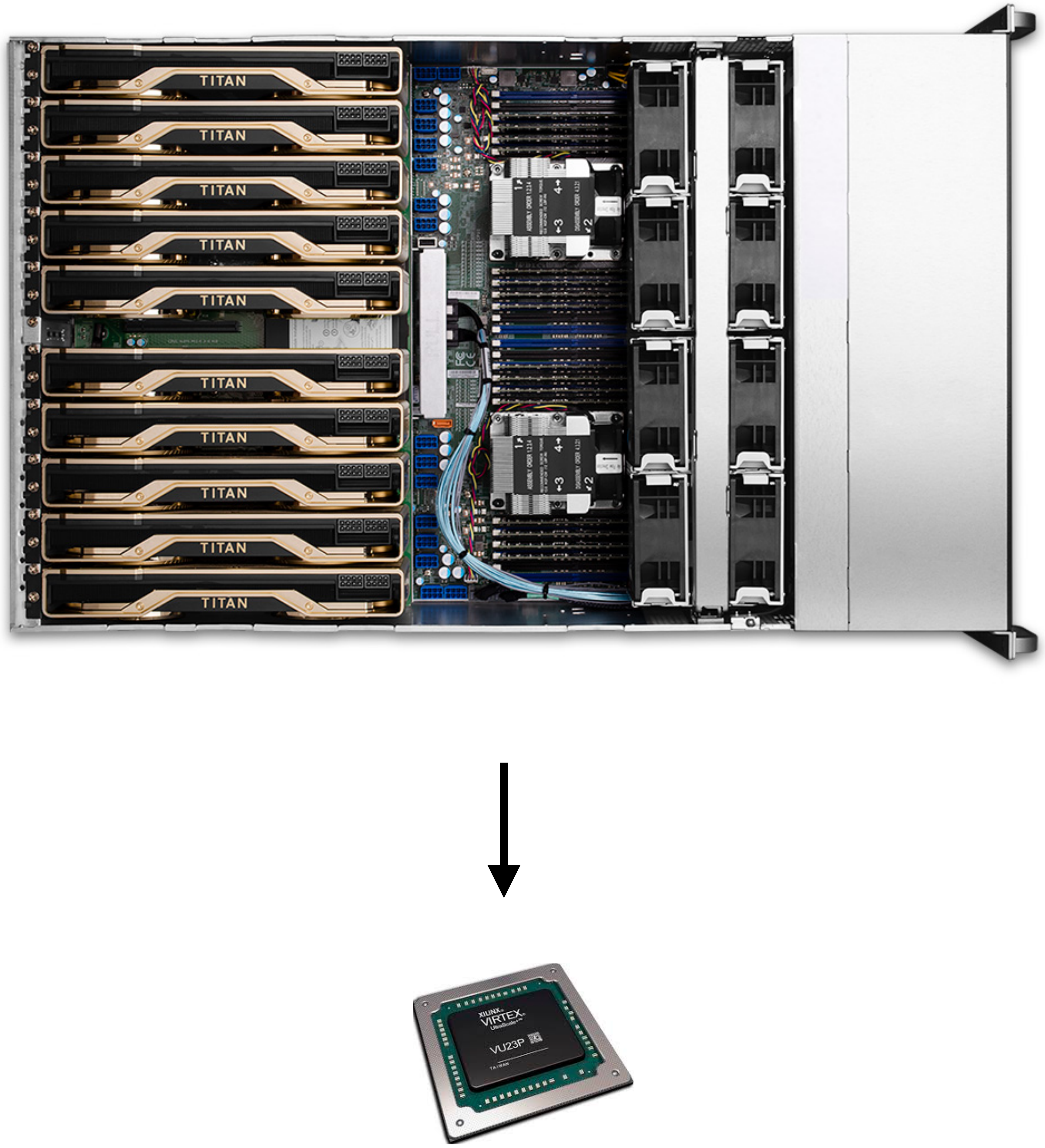
Before deploying any DNN on the edge, must make it efficient!

During training

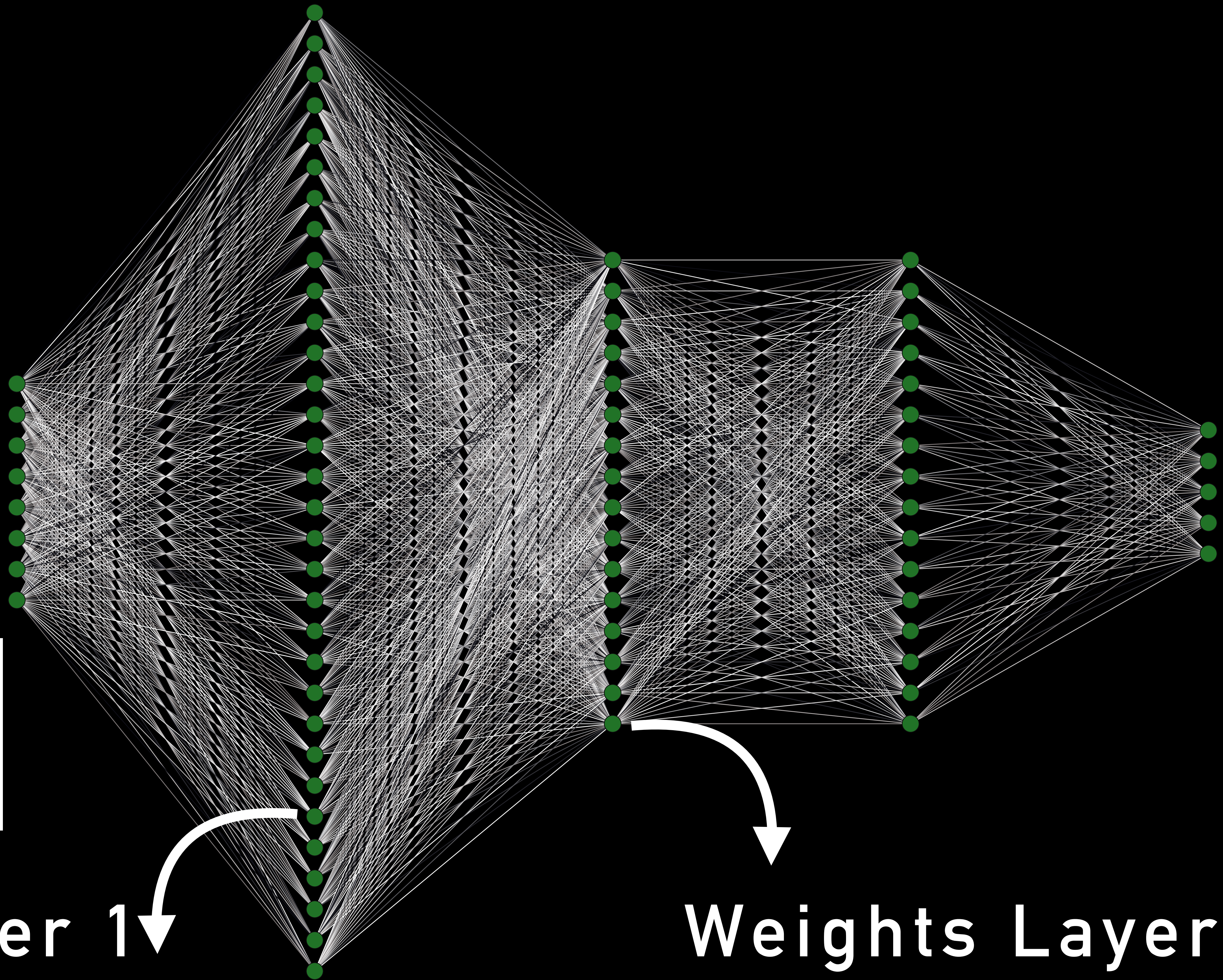
- **Quantization:** do you really need 32-bit FP precision?
- **Pruning:** removal insignificant synapses
- **Knowledge distillation** (see Patrick Odagiu talk tomorrow!)

Post-training

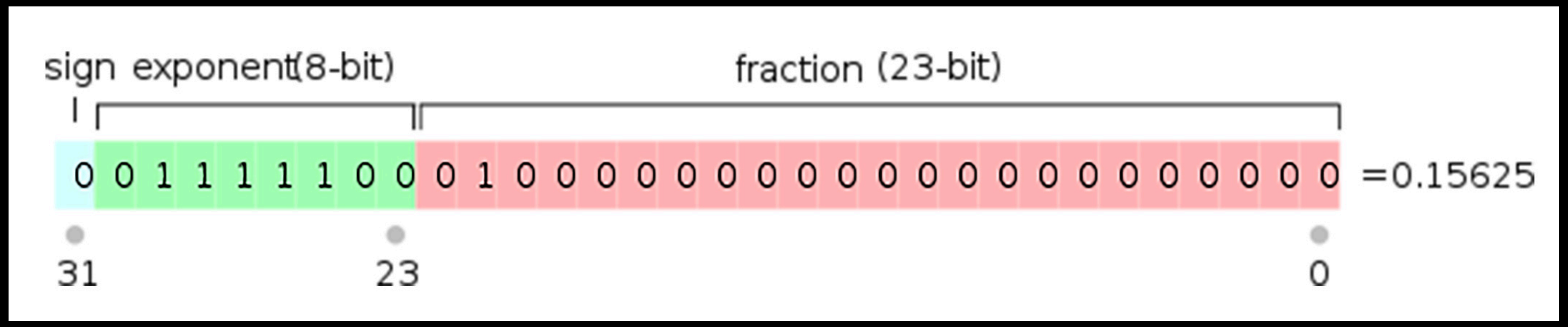
- **Parallelise**:
  all computation that can be done in parallel, do in parallel!



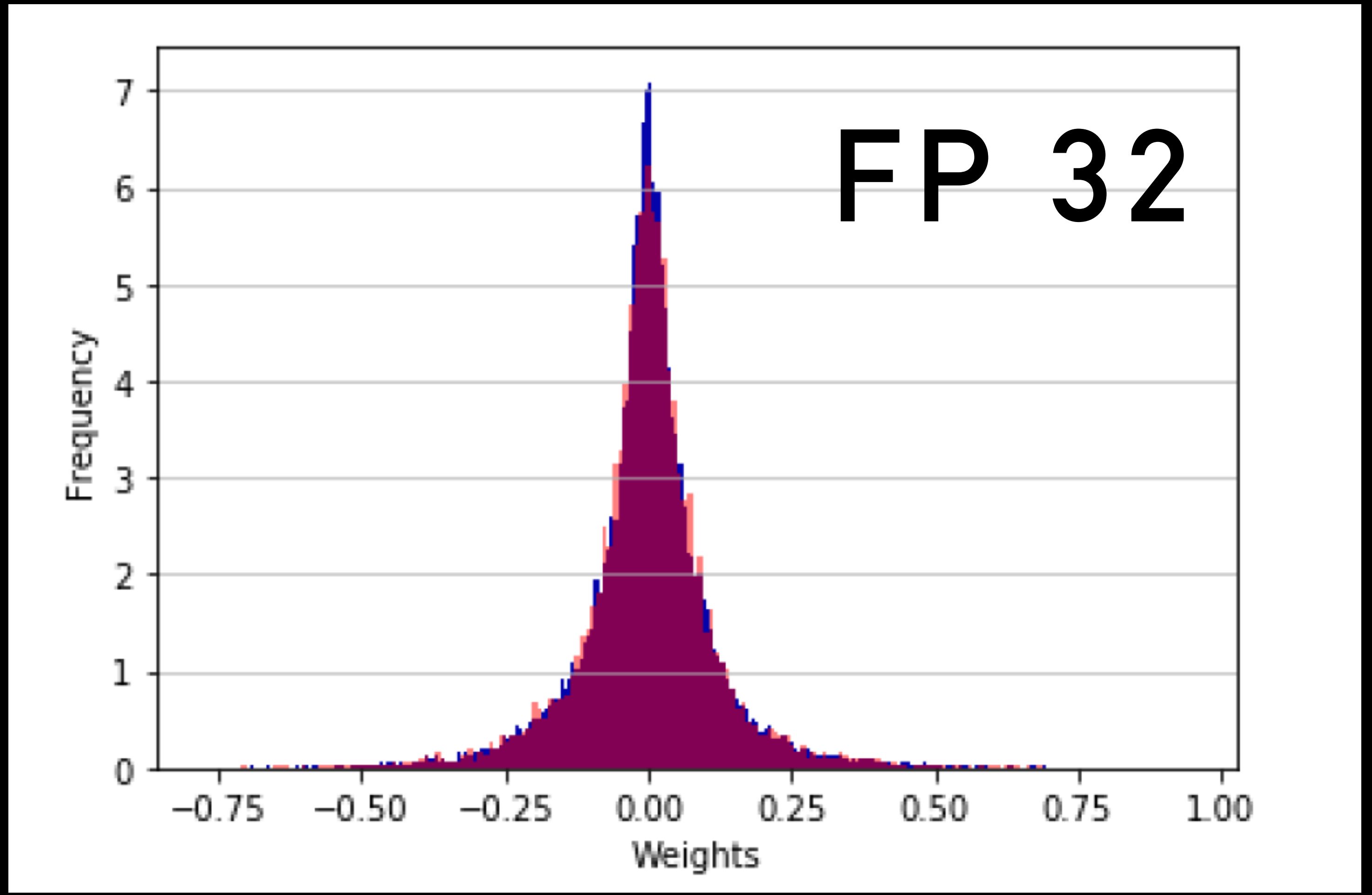
Quantization



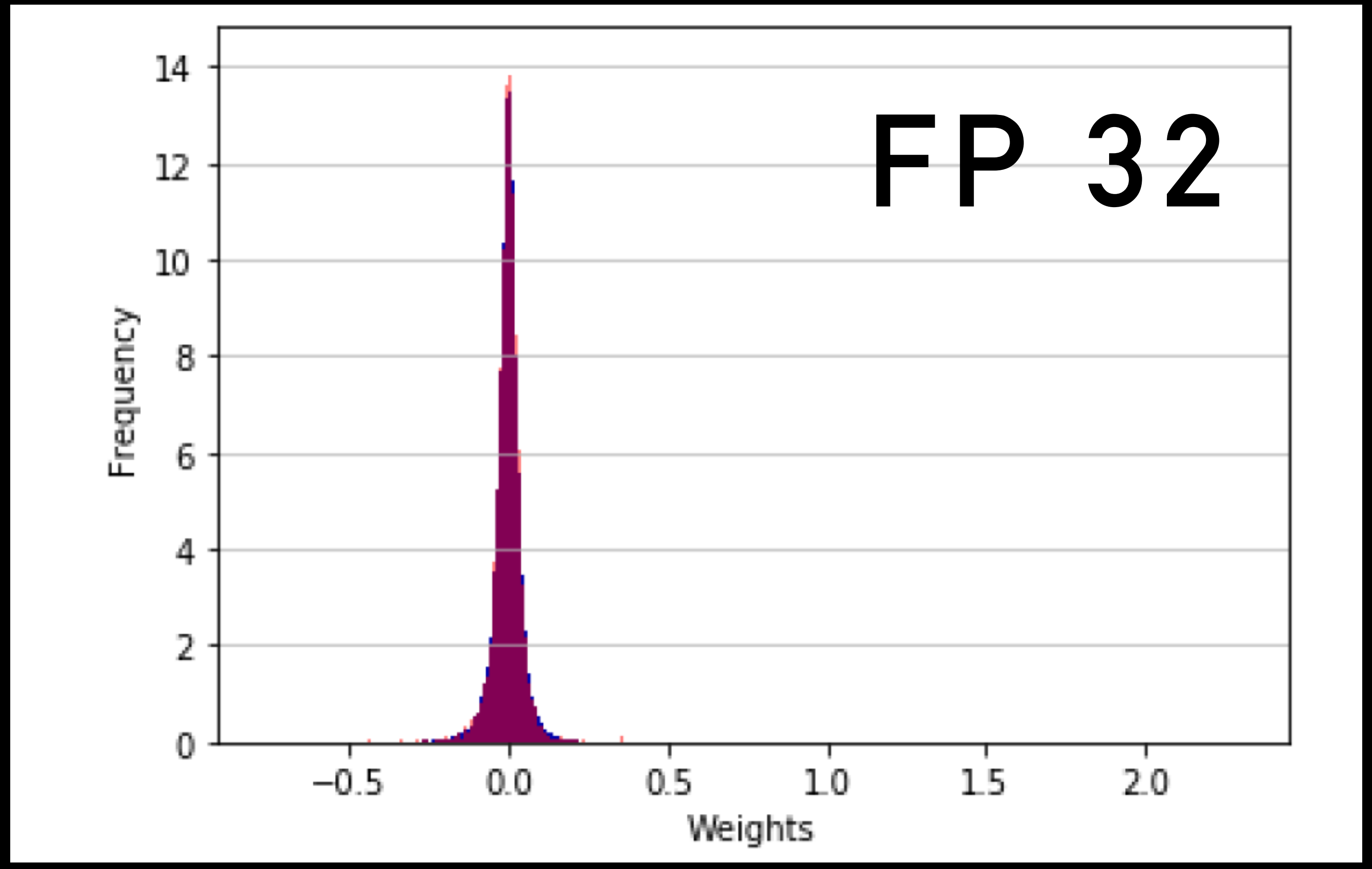
Floating point 32



Weights Layer 1

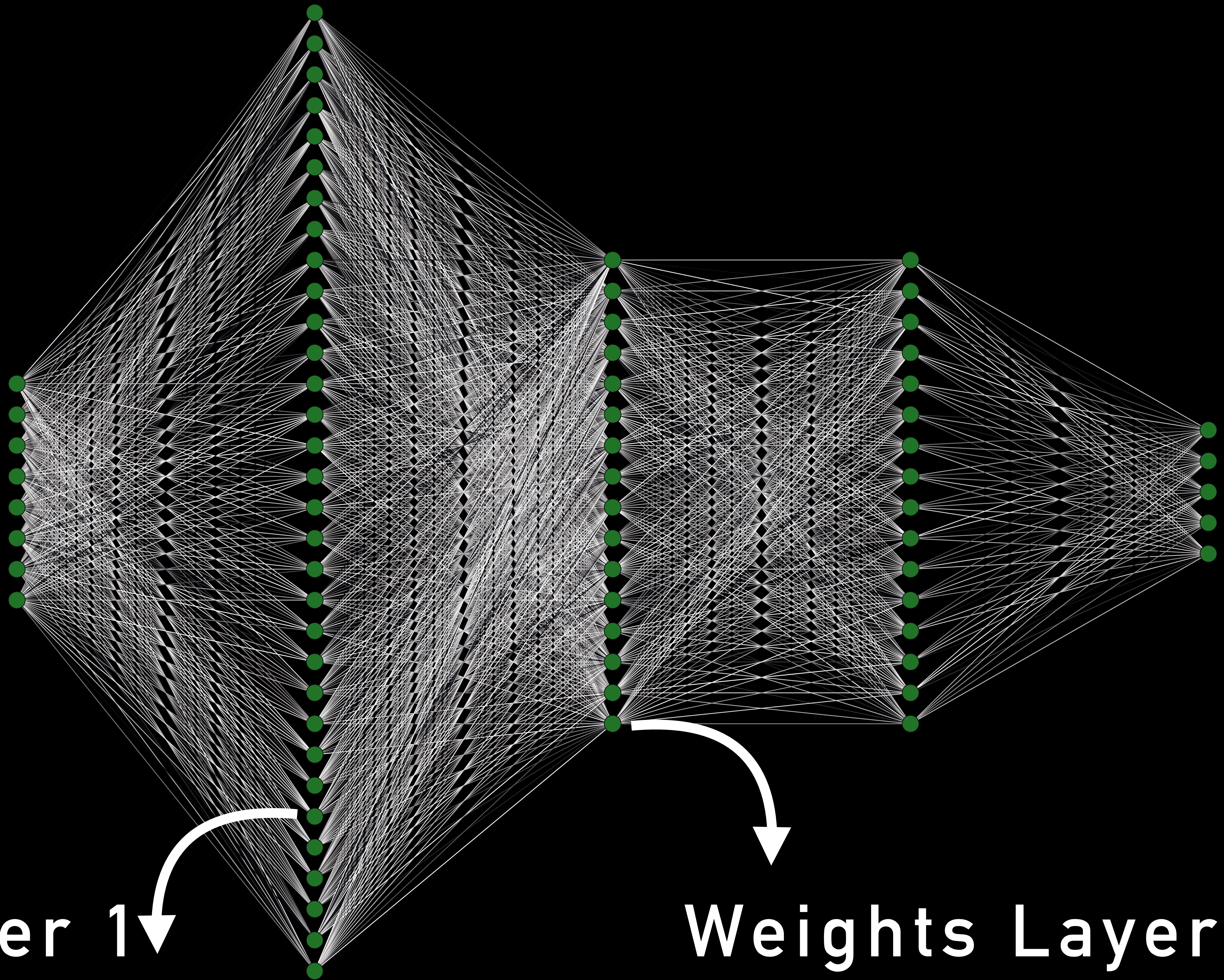


Weights Layer 2

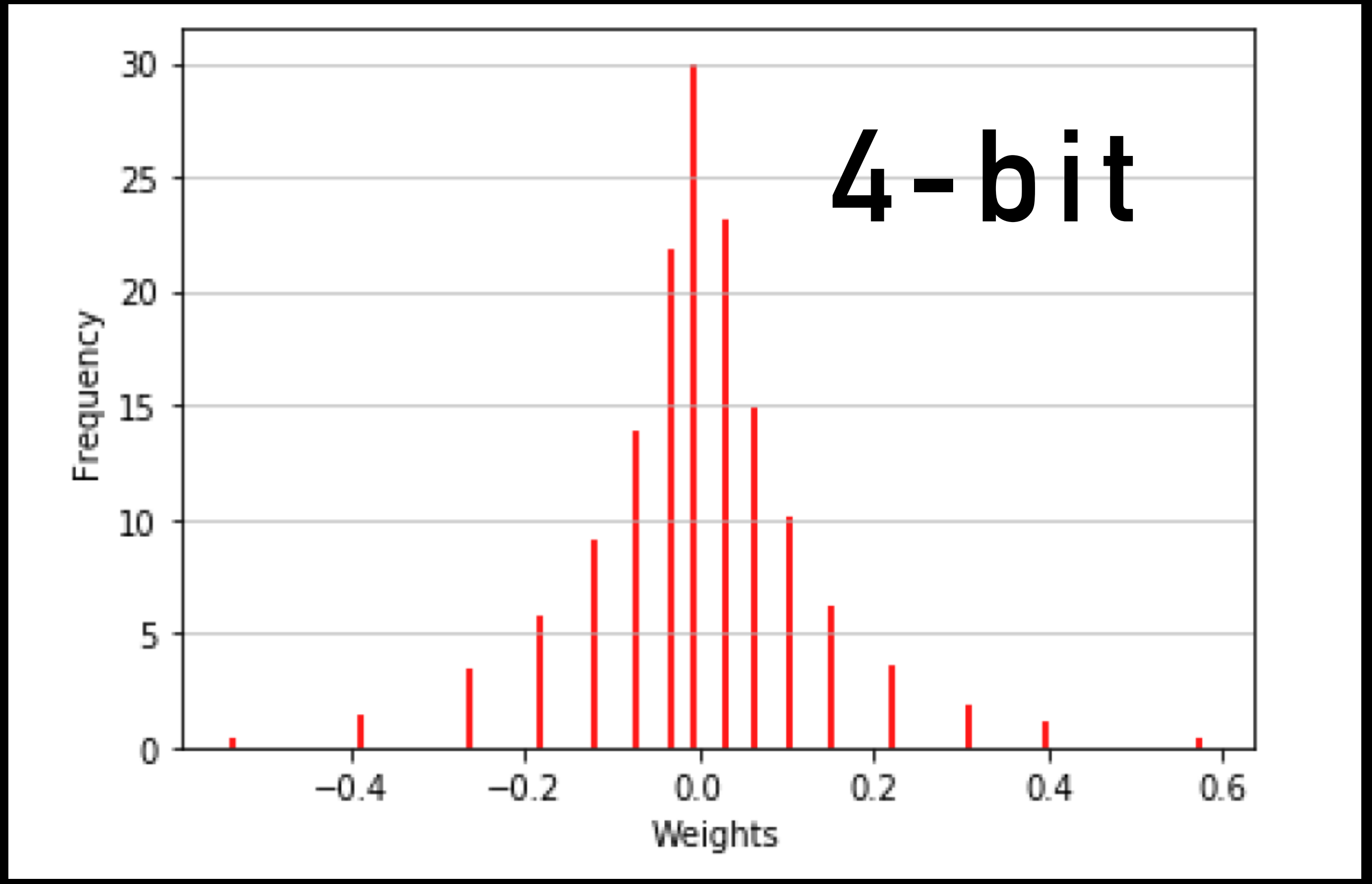


Quantization

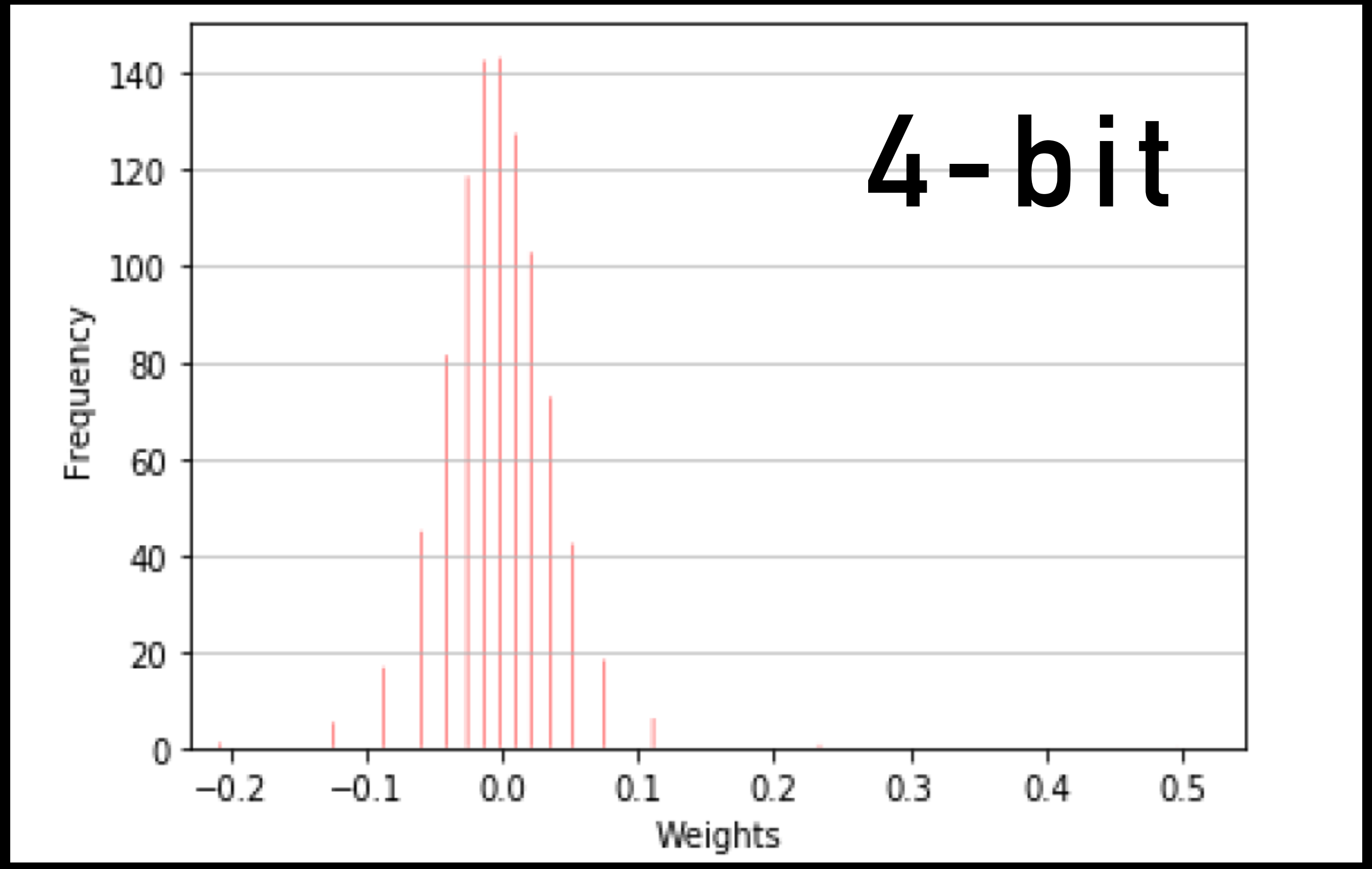
Fixed point



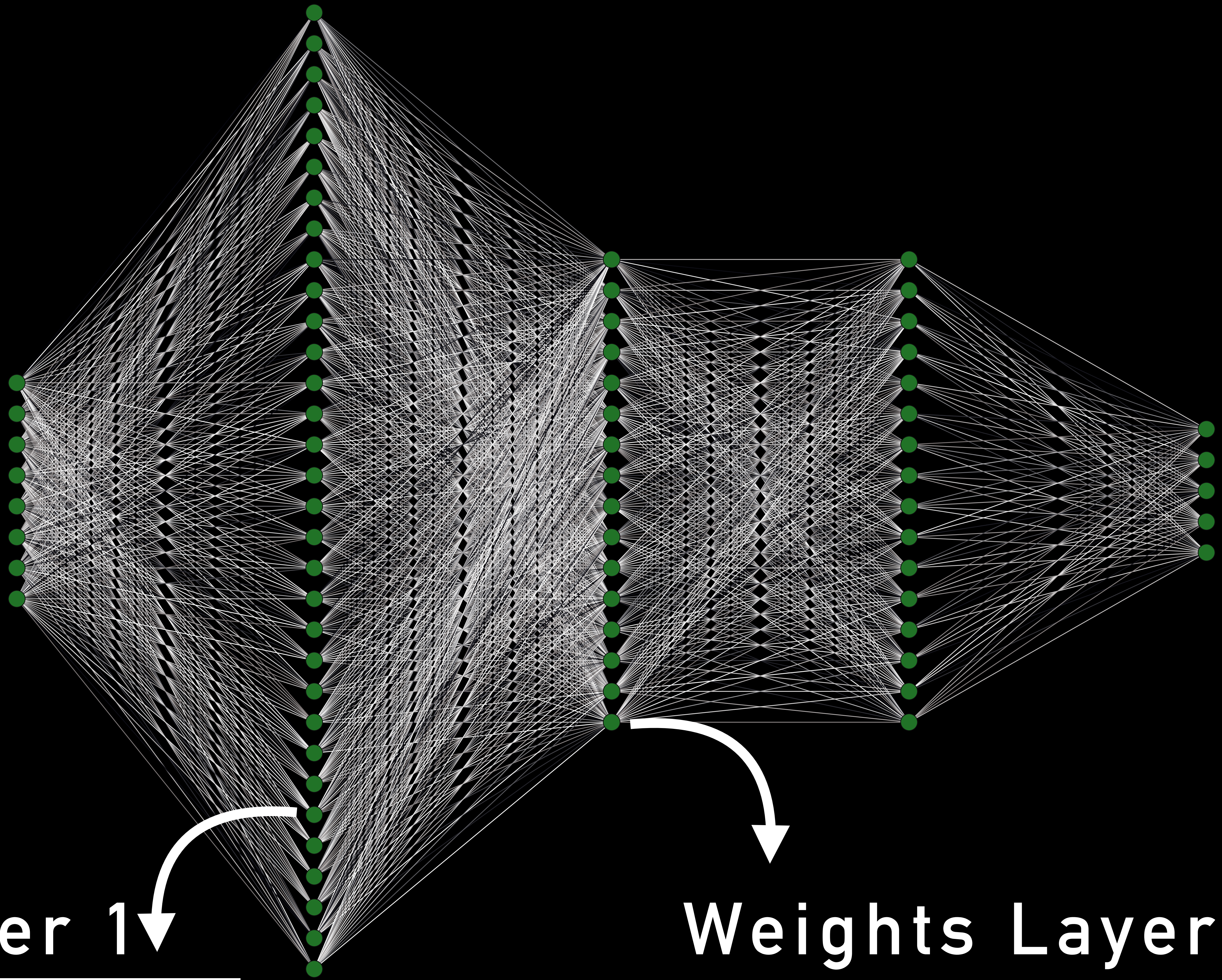
Weights Layer 1



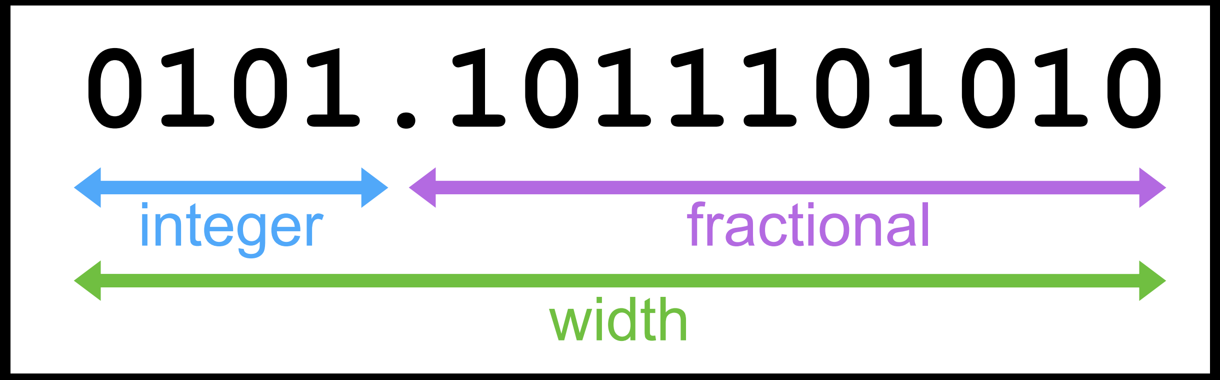
Weights Layer 2



Quantization



Fixed point



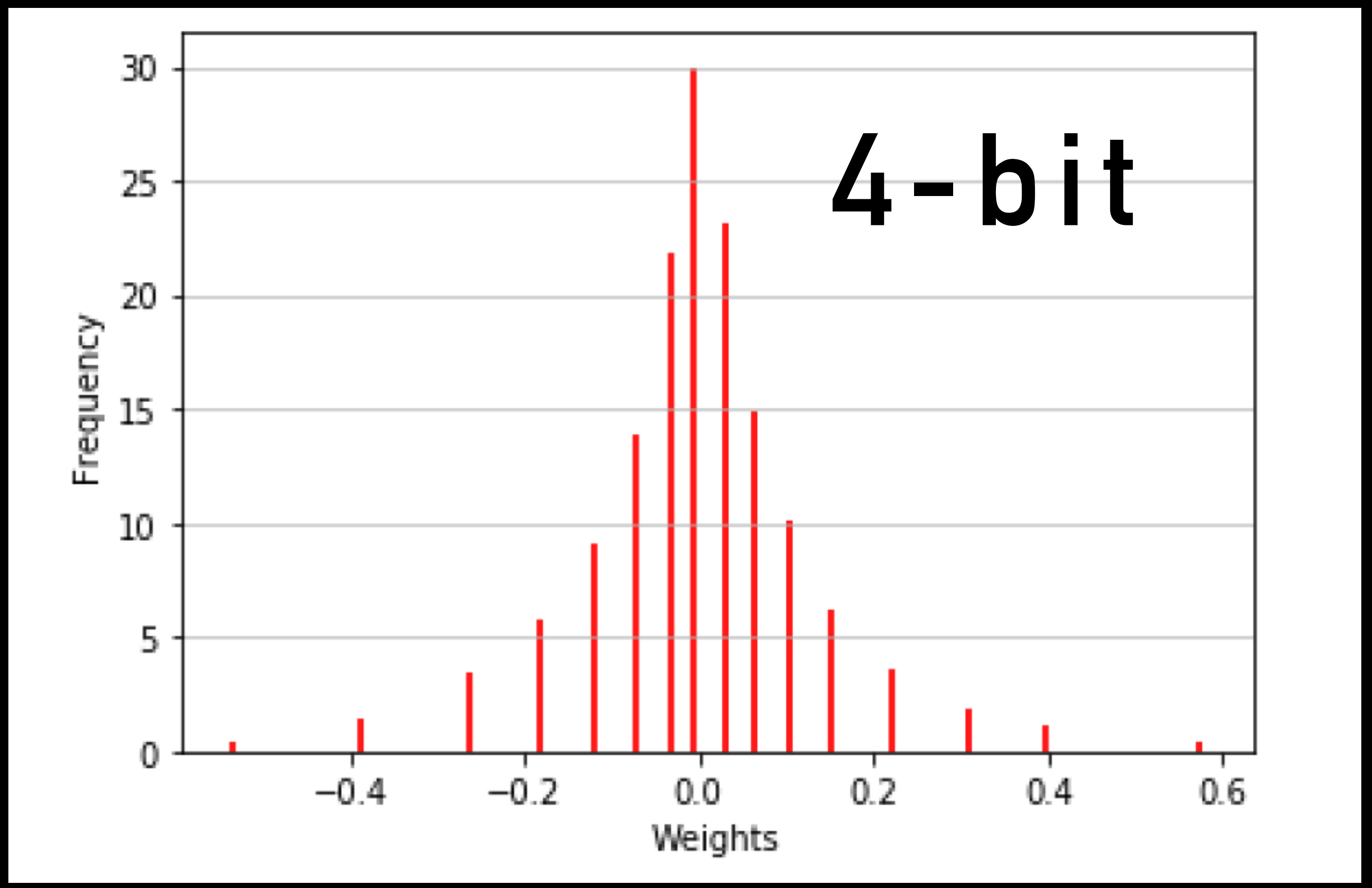
0101.1011101010

integer

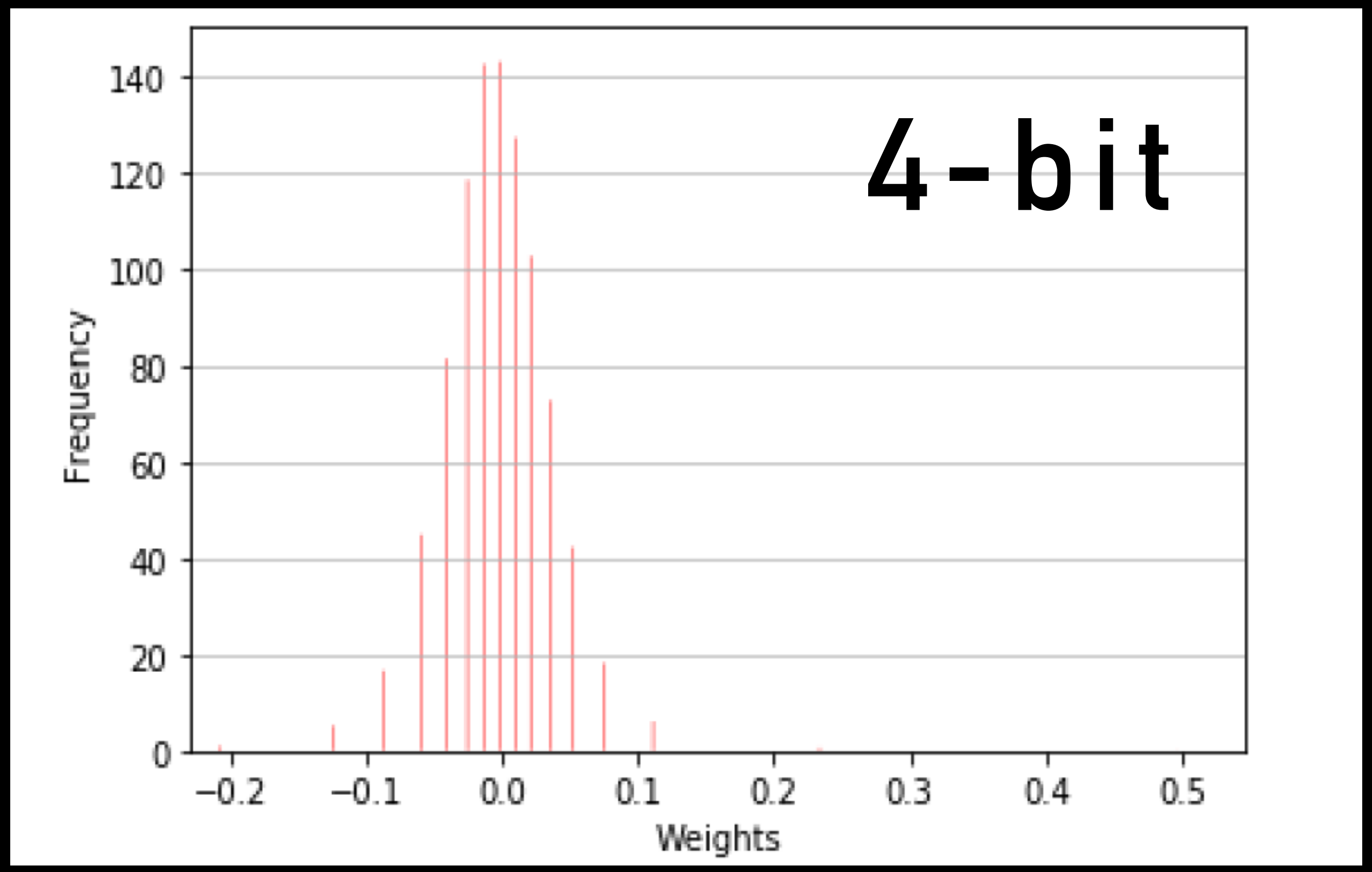
fractional

width

Weights Layer 1

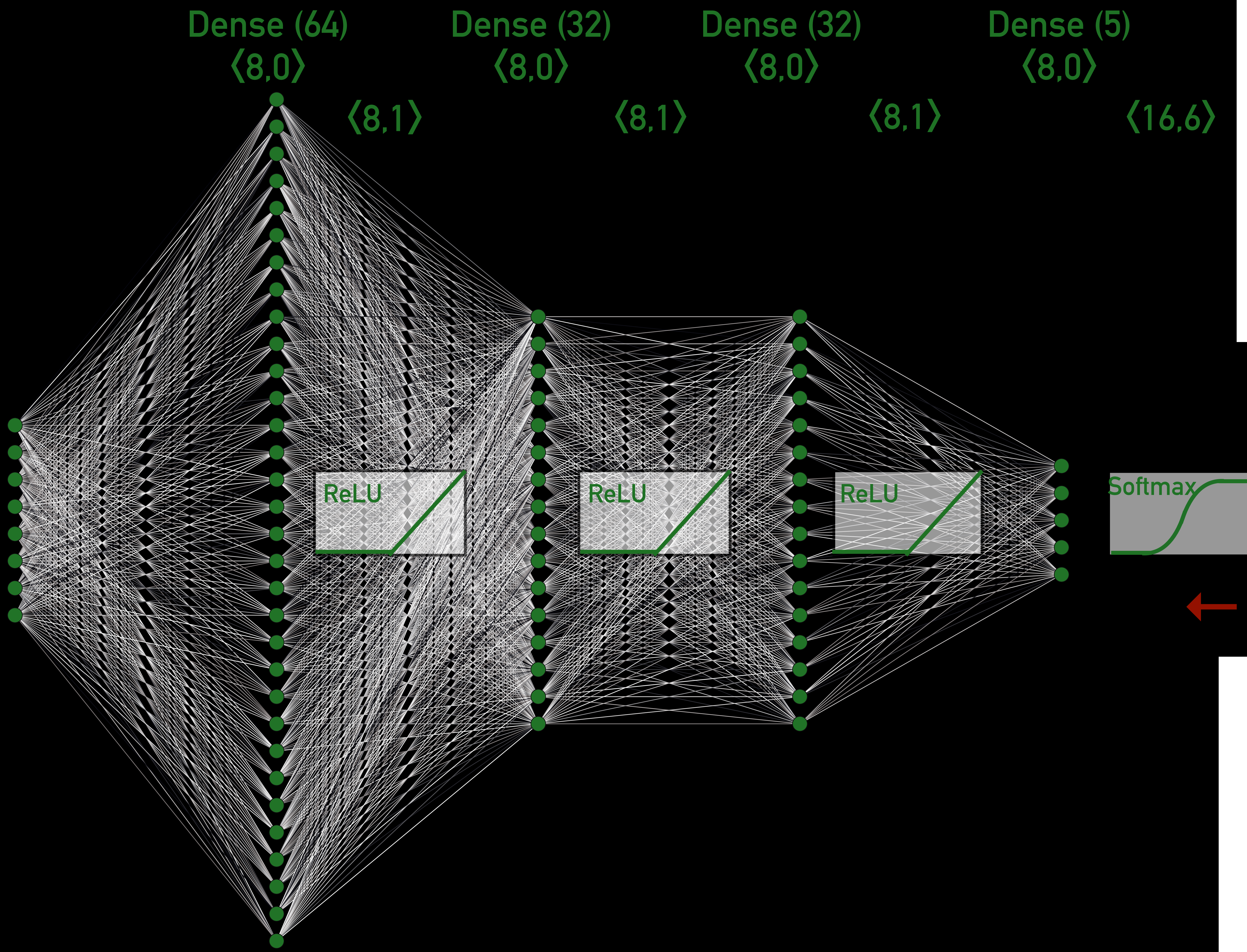


Weights Layer 2



Quantization-aware training

Forward pass →

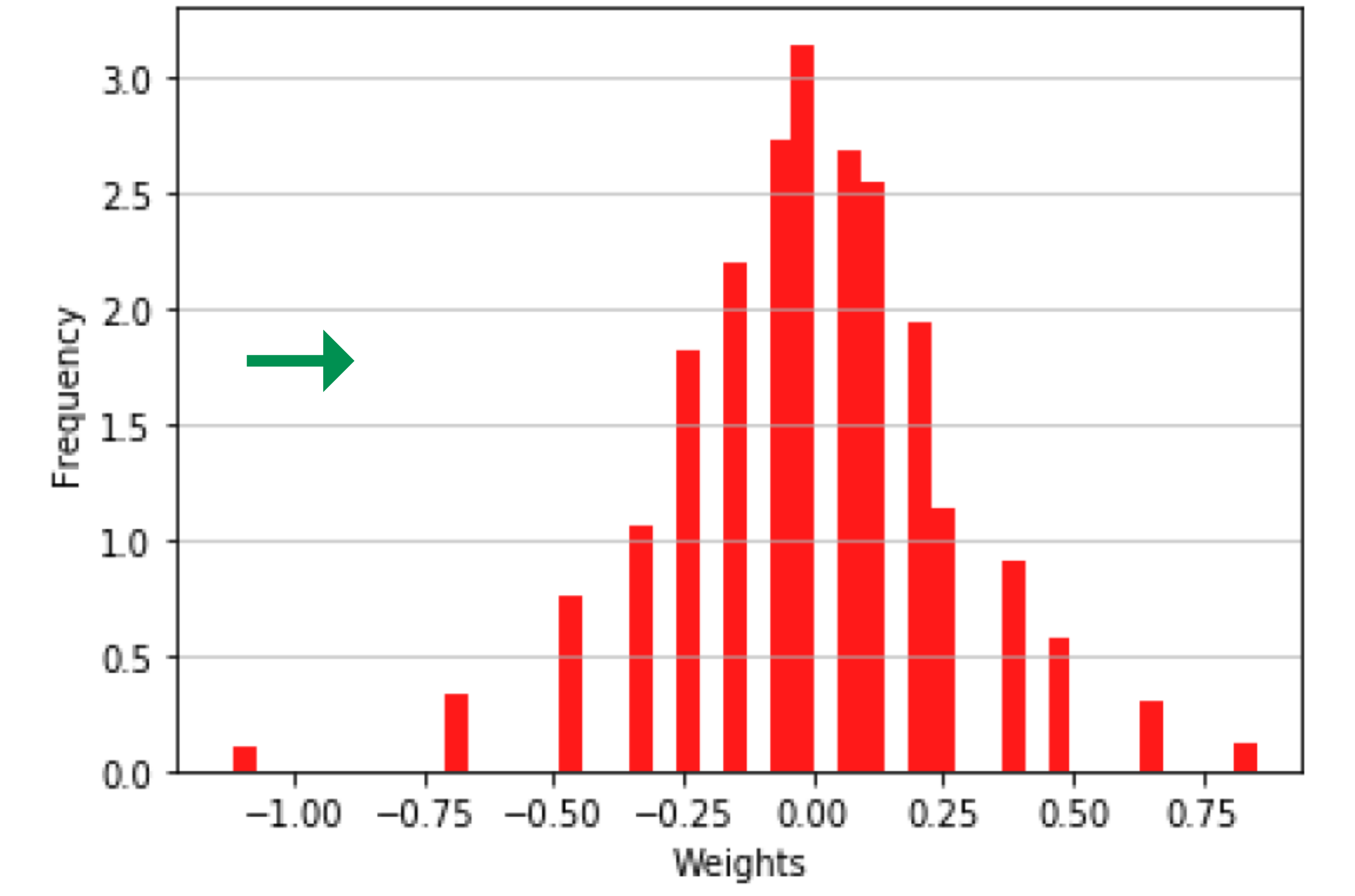


Dense (64) ⟨8,0⟩
Dense (32) ⟨8,0⟩
Dense (32) ⟨8,0⟩
Dense (5) ⟨8,0⟩

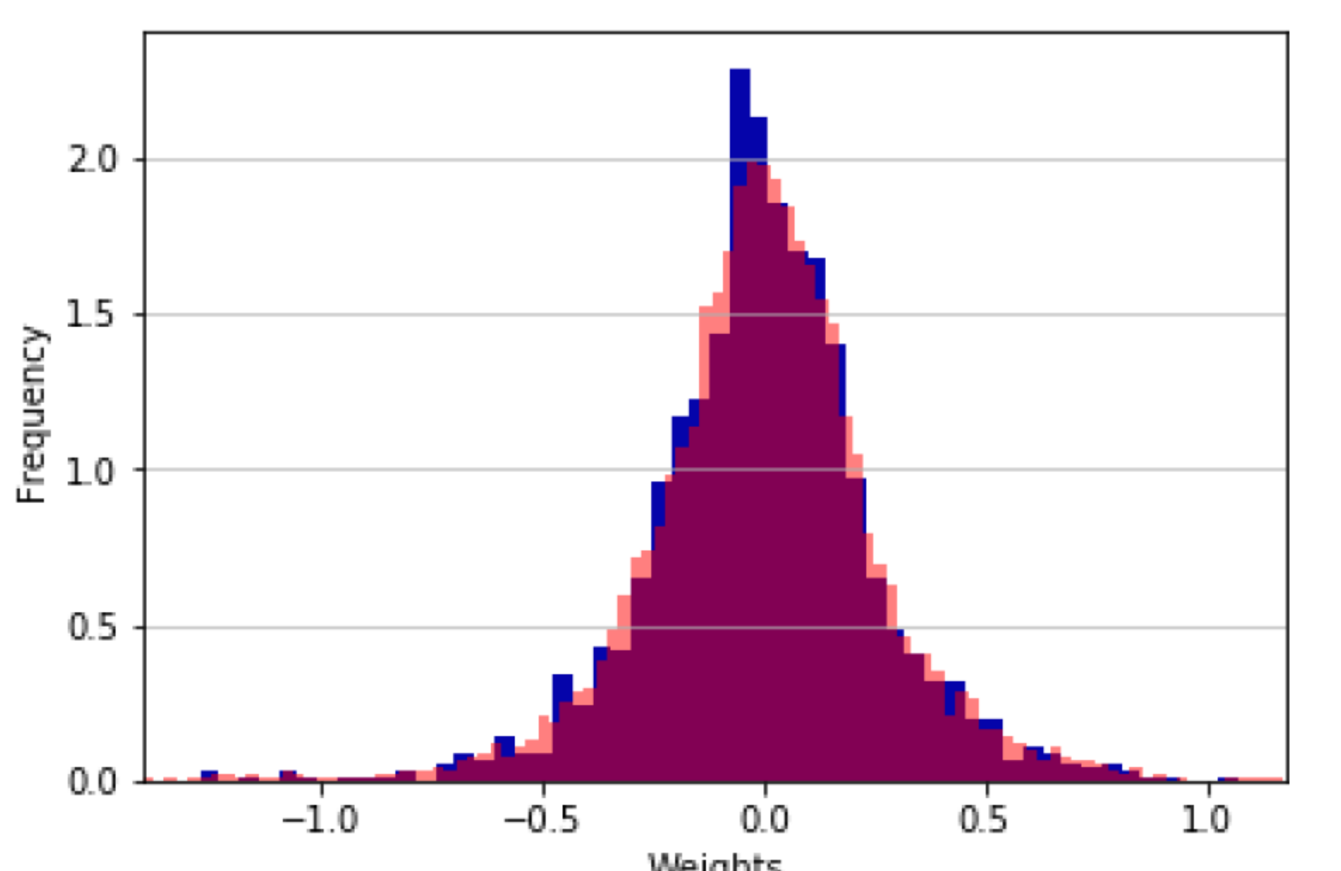
⟨8,1⟩ ⟨8,1⟩ ⟨8,1⟩ ⟨16,6⟩

FP 32 FP 32 FP 32 FP 32

FP 32 FP 32 FP 32 FP 32



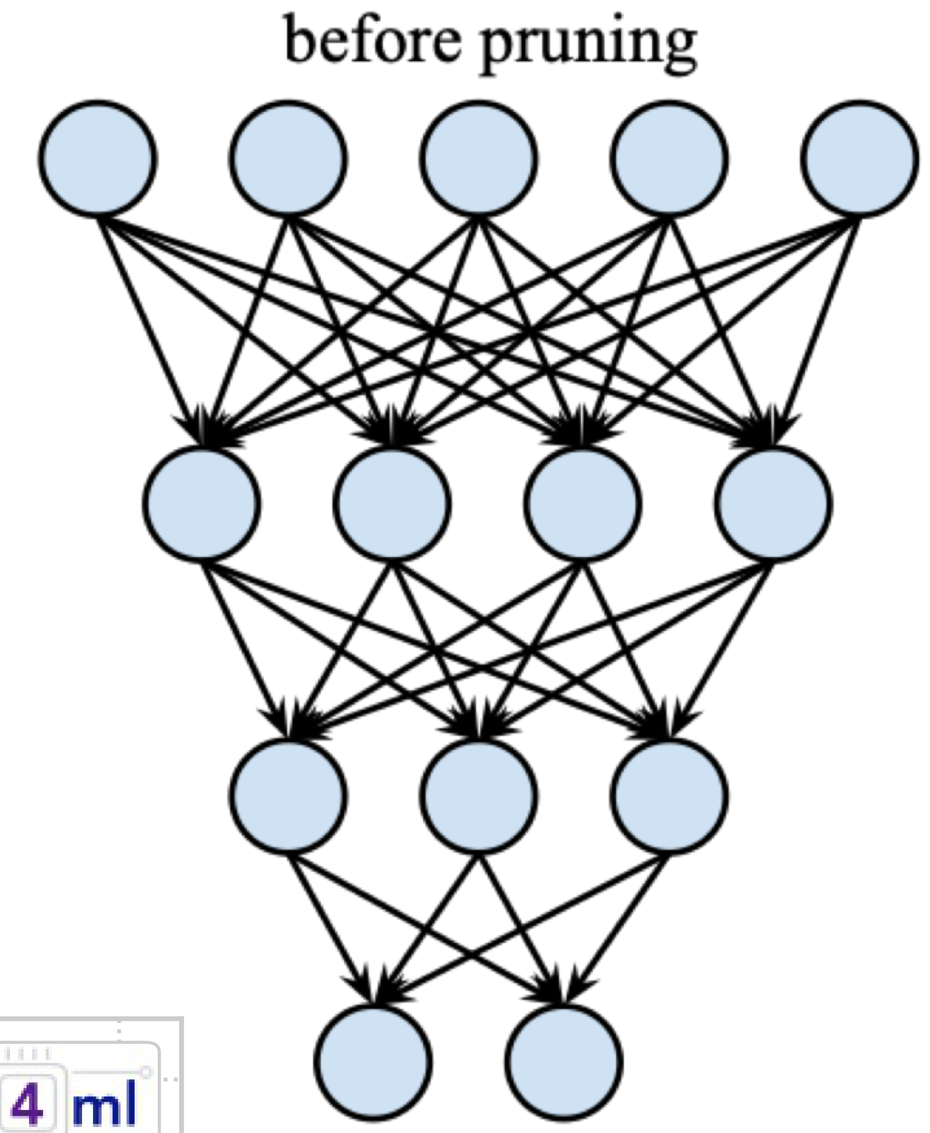
← Back propagation

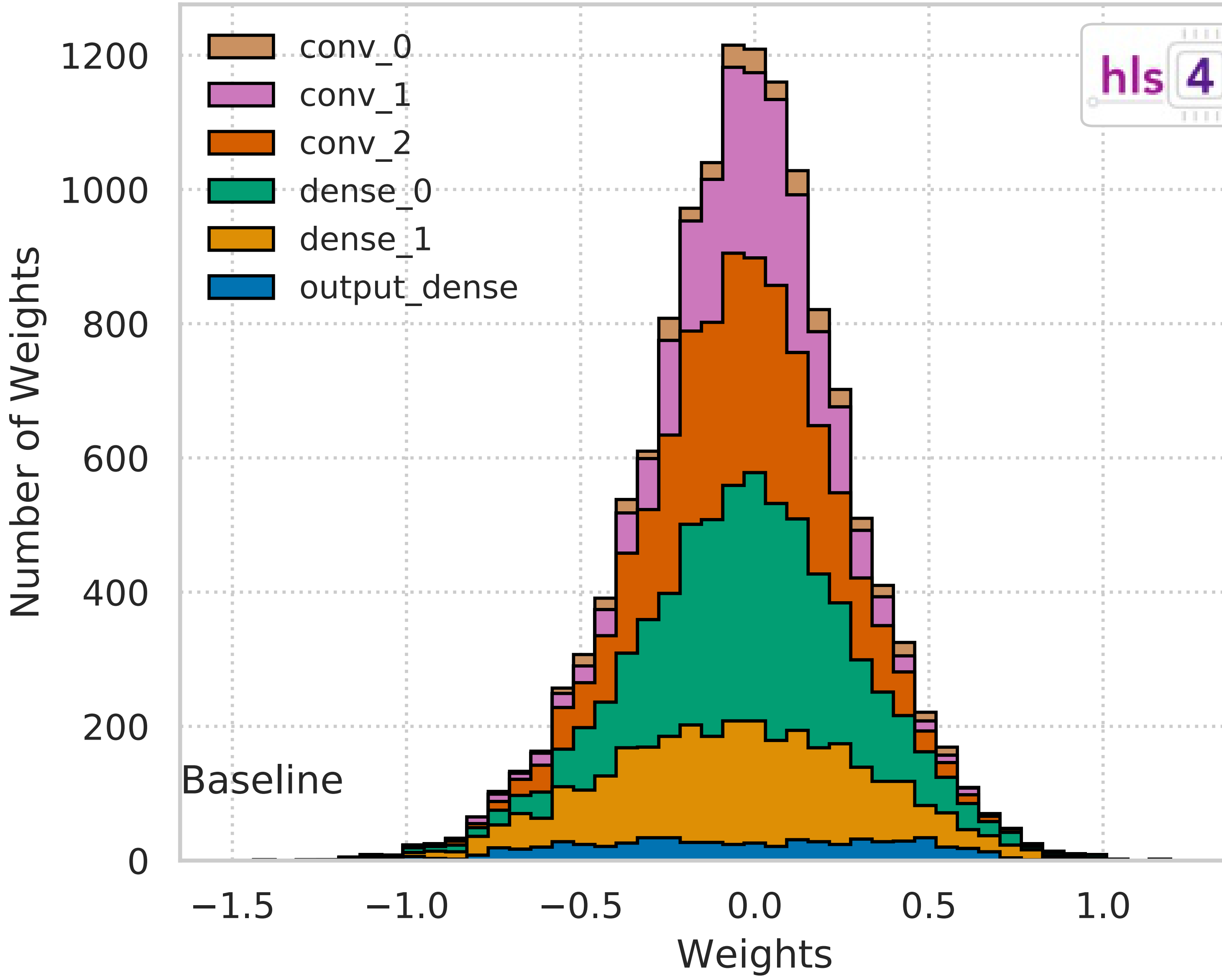


Nature Machine Intelligence 3 (2021)

https://blog.tensorflow.org/2019/05/tf-model-optimization-toolkit-pruning-API.html

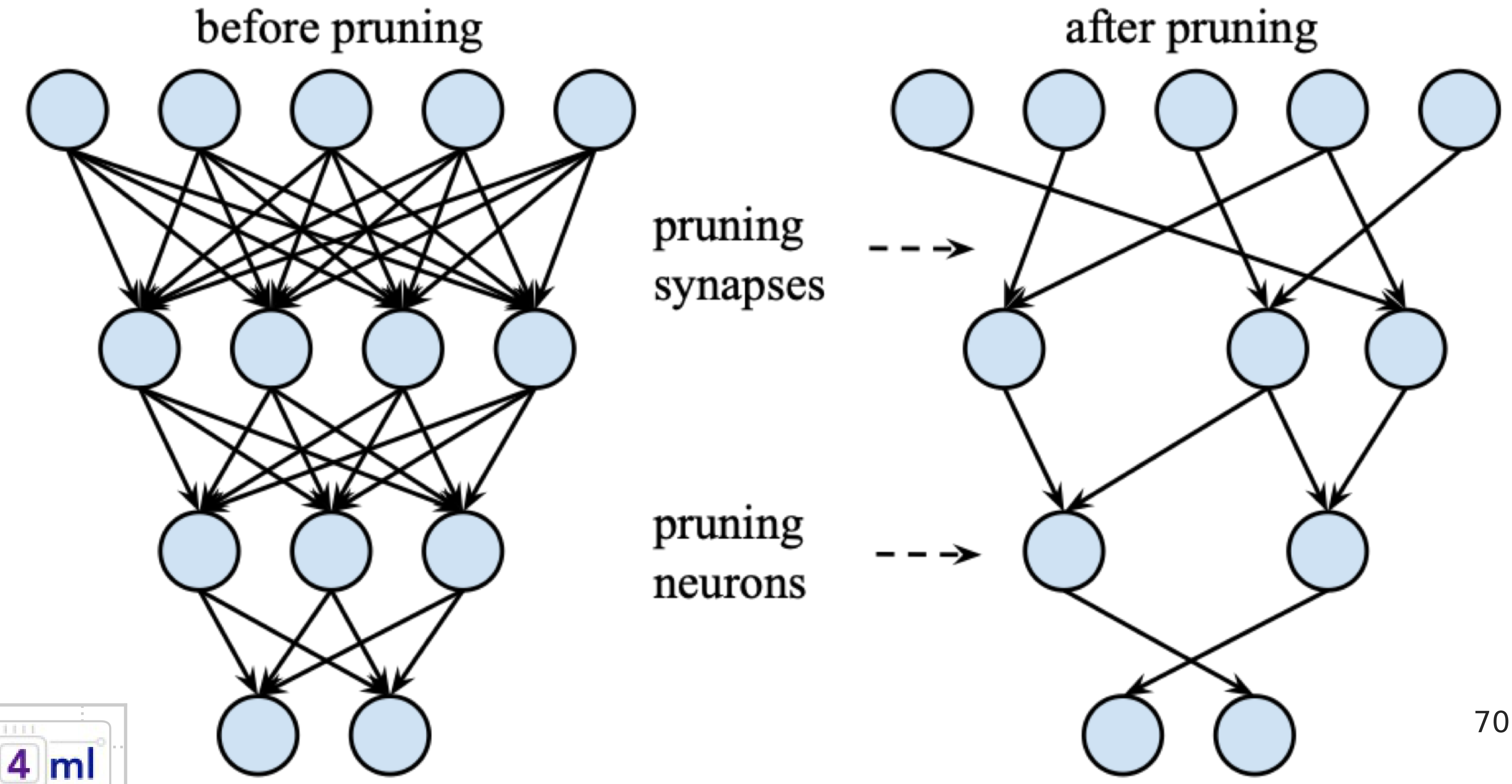
Pruning

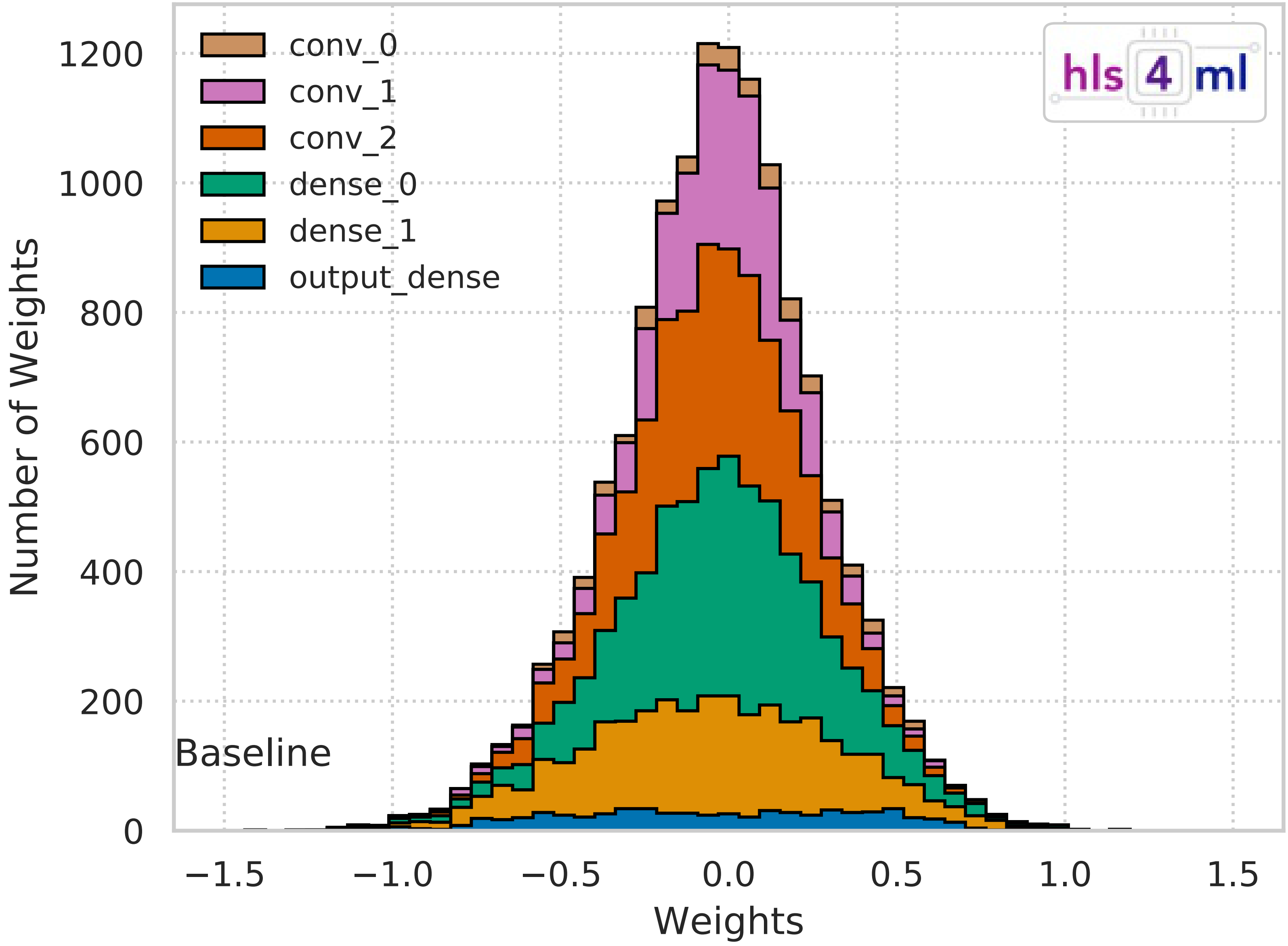


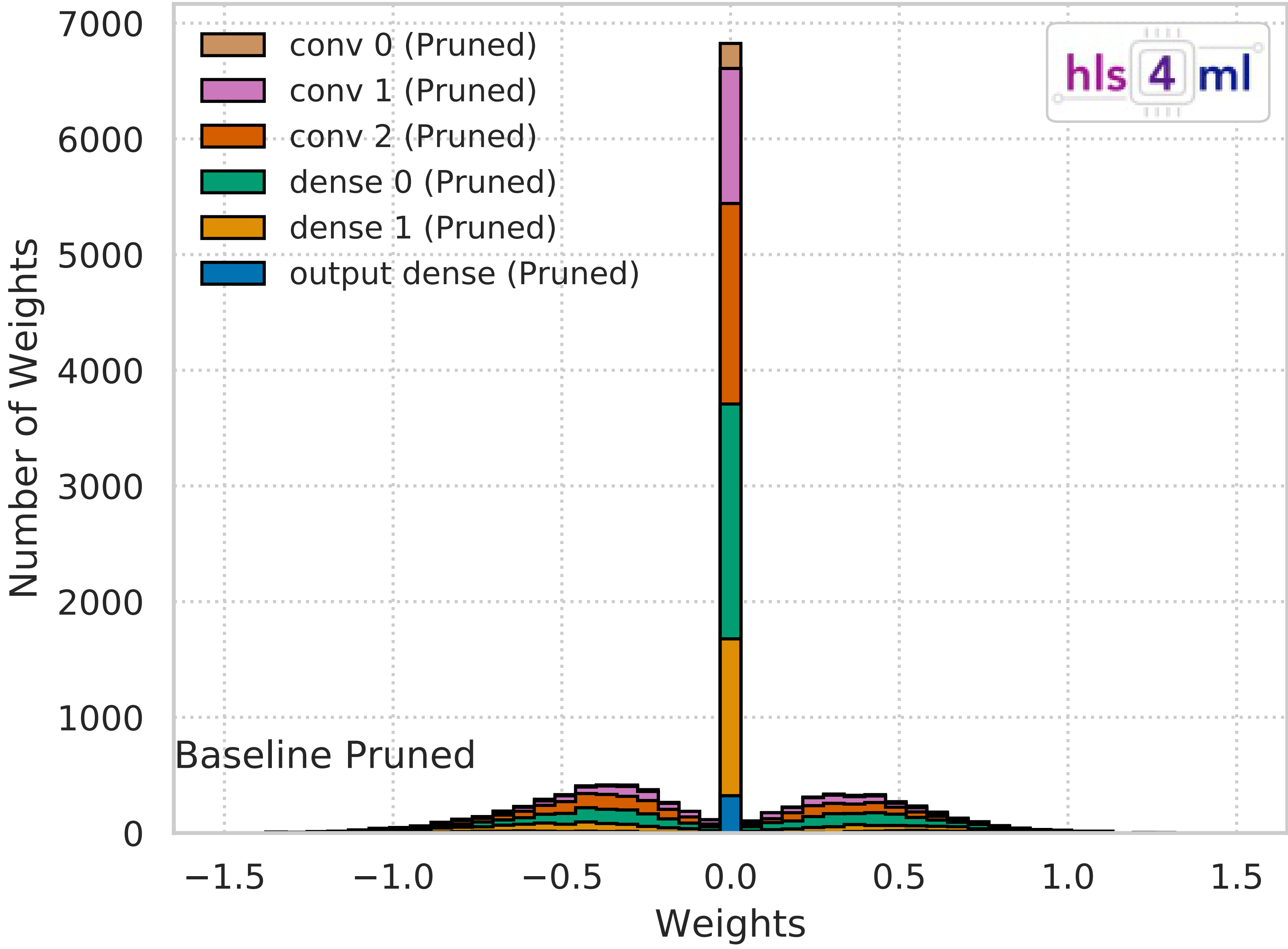


https://blog.tensorflow.org/2019/05/tf-model-optimization-toolkit-pruning-API.html

Pruning

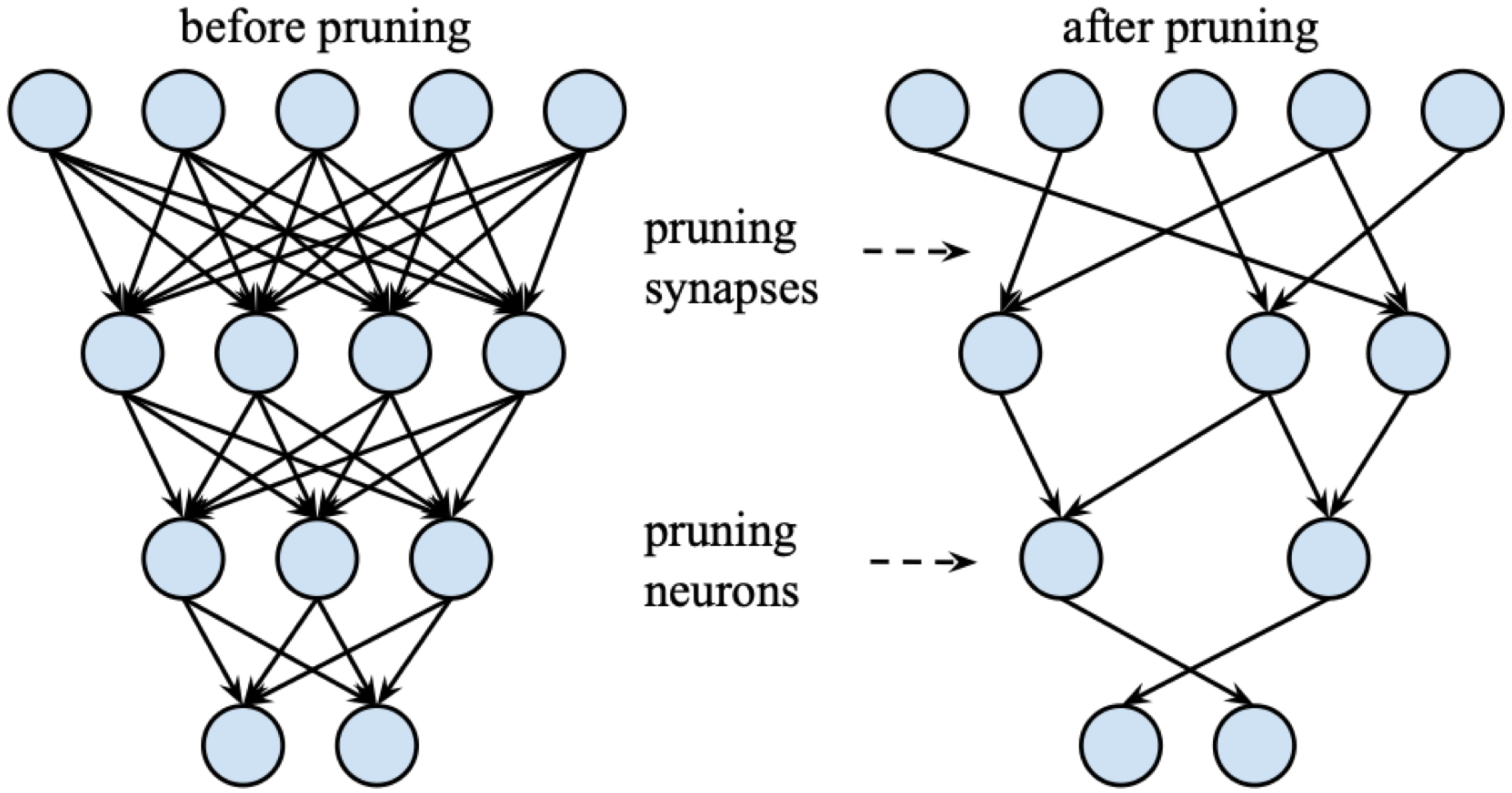


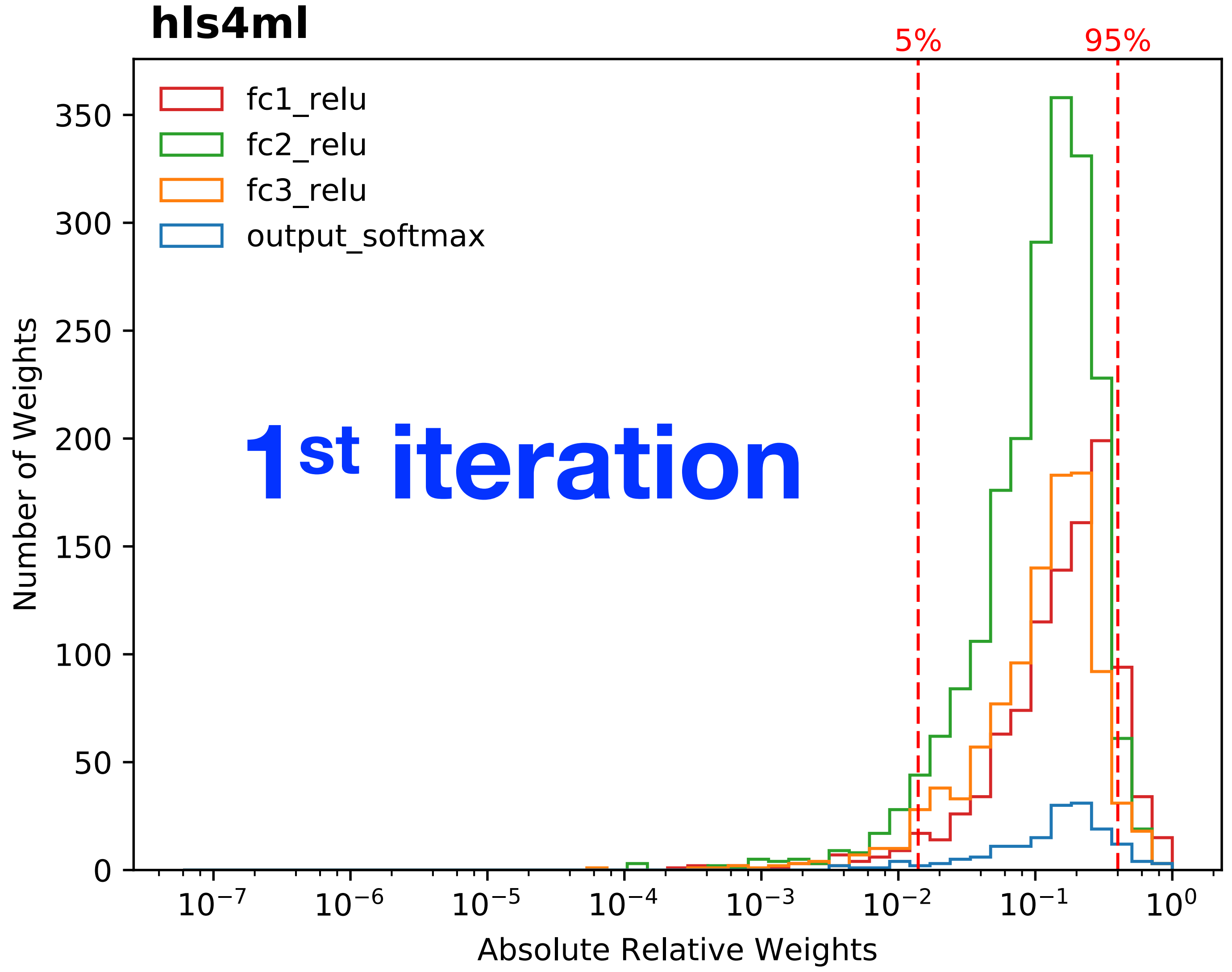




https://blog.tensorflow.org/2019/05/tf-model-optimization-toolkit-pruning-API.html

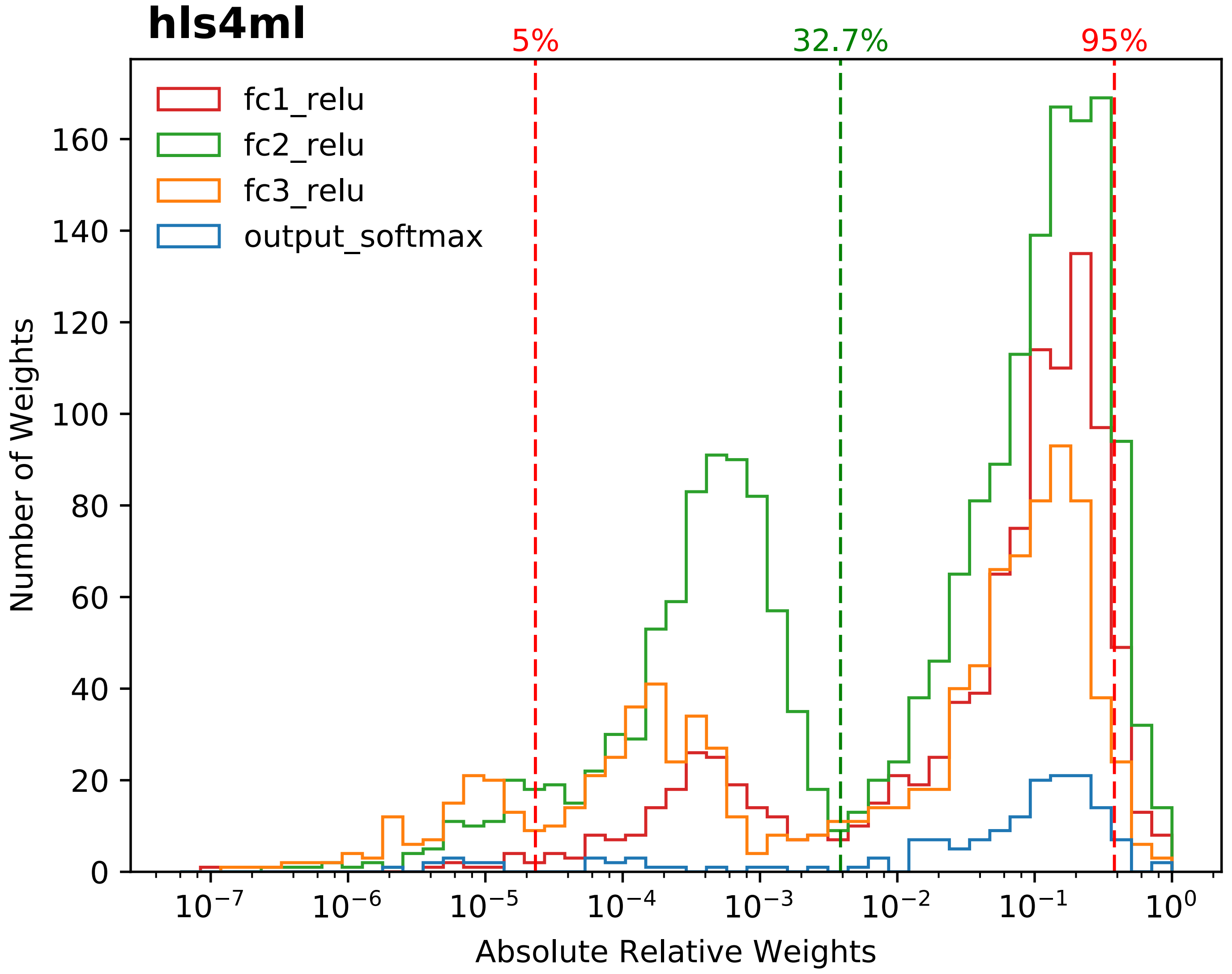
Pruning



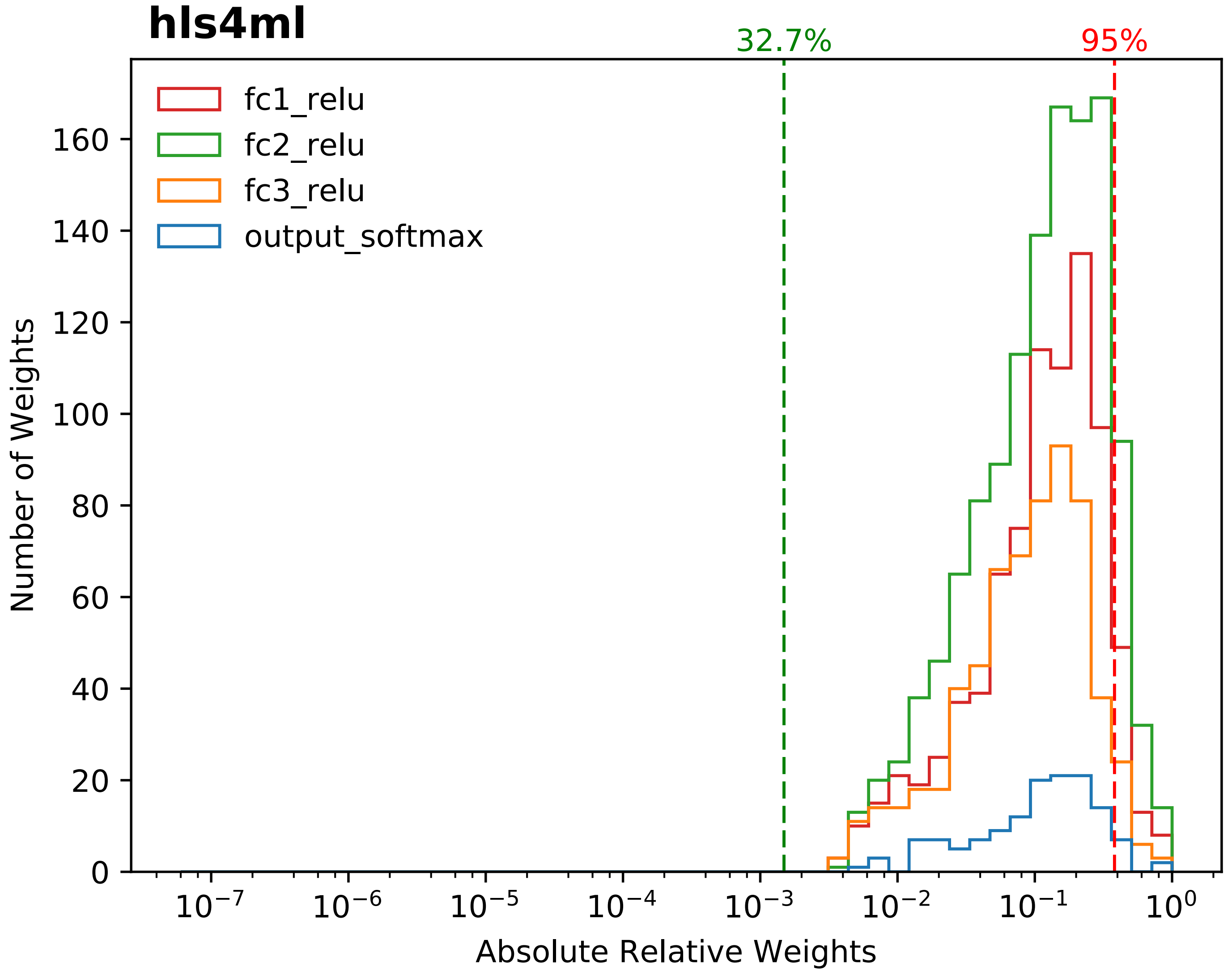


1st iteration

Train with L_1

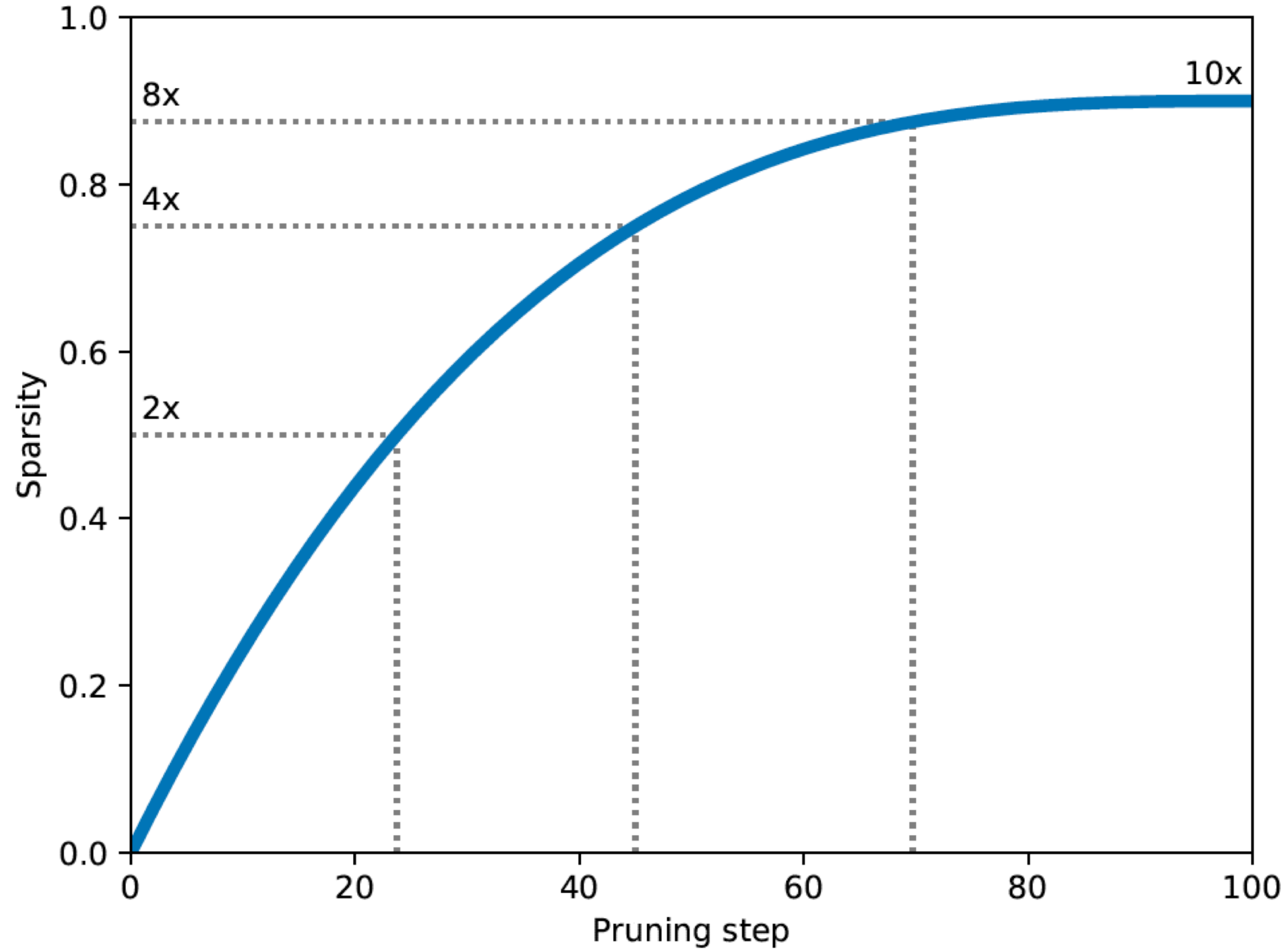


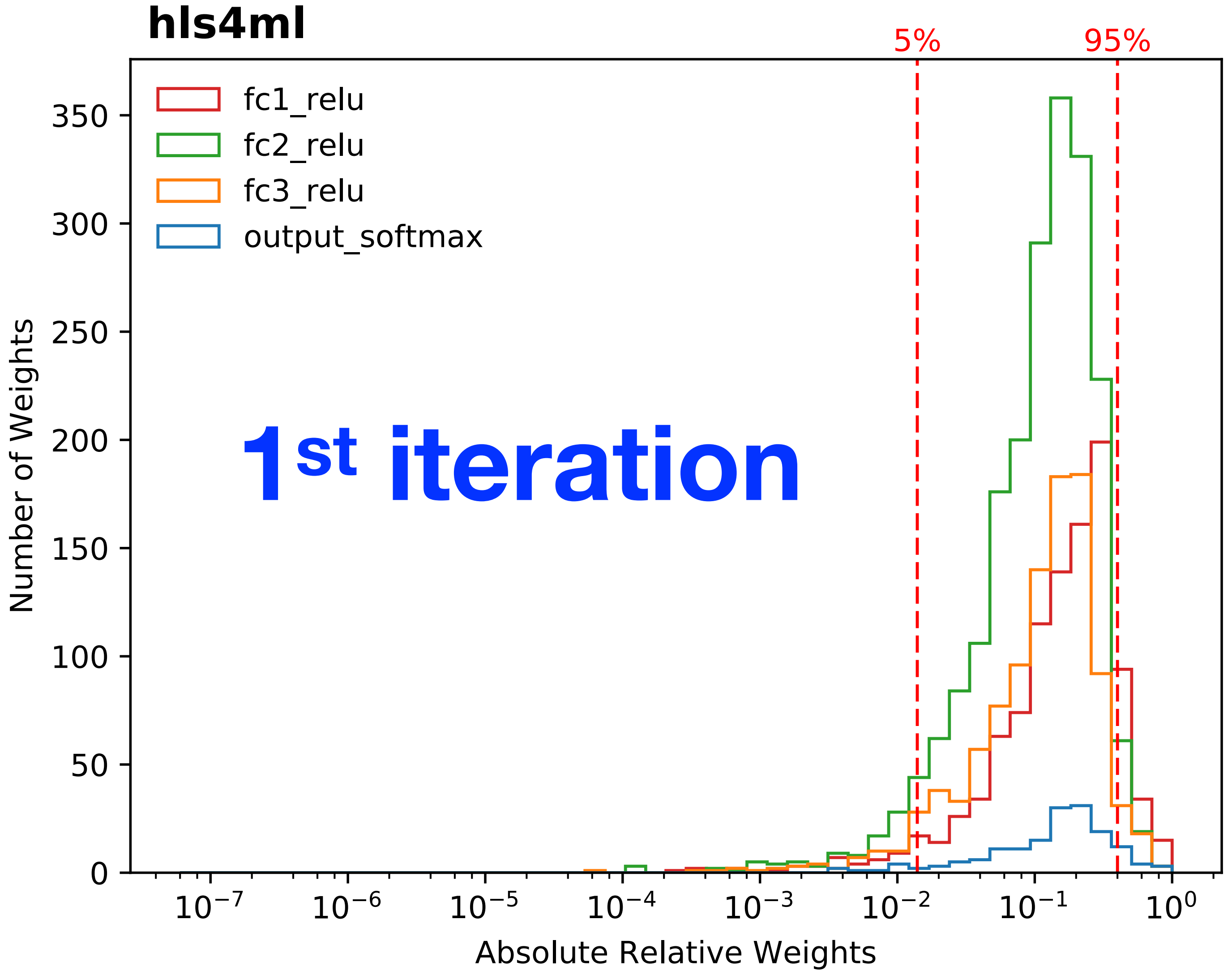
Prune



Pruning

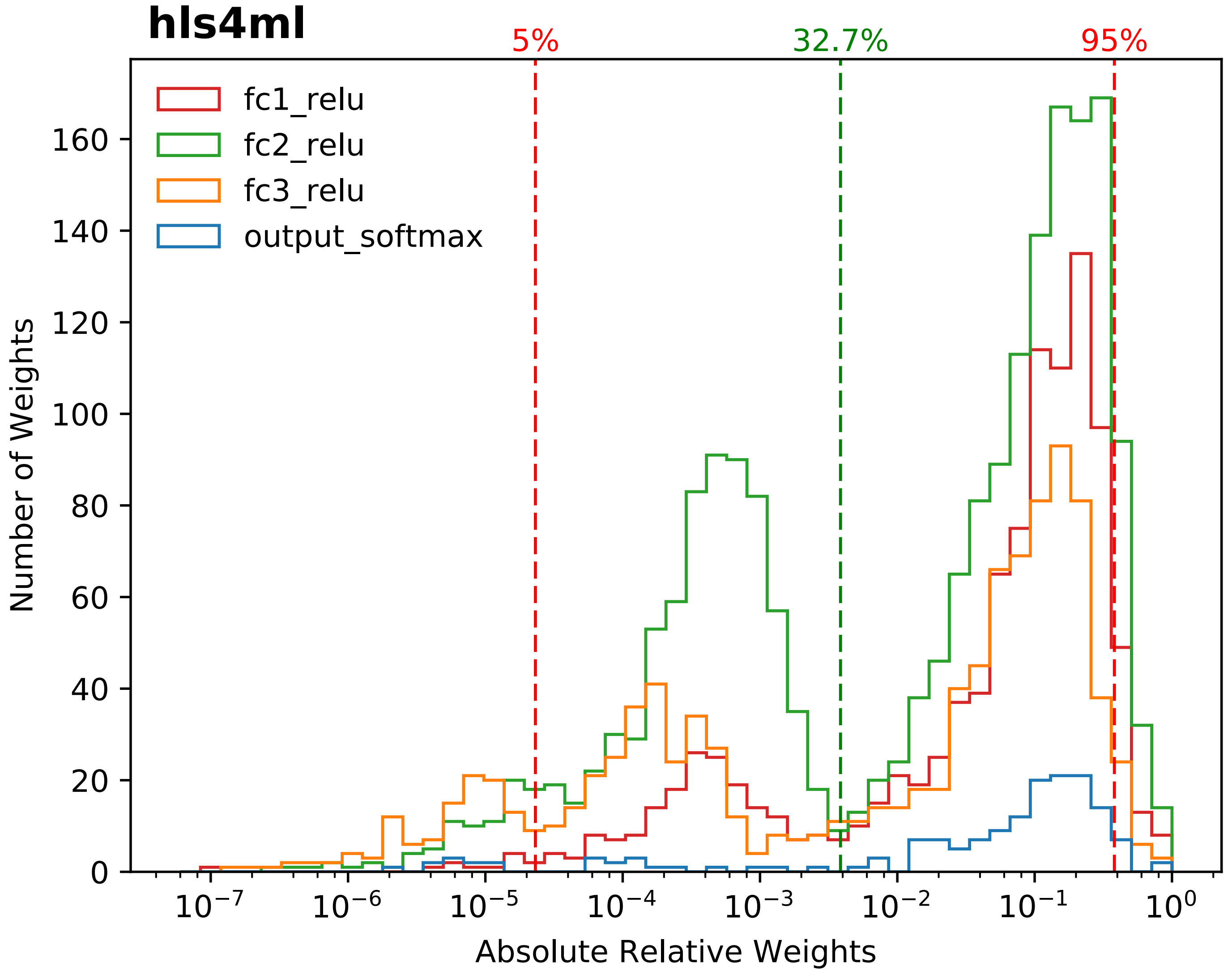
https://blog.tensorflow.org/2019/05/tf-model-optimization-toolkit-pruning-API.html



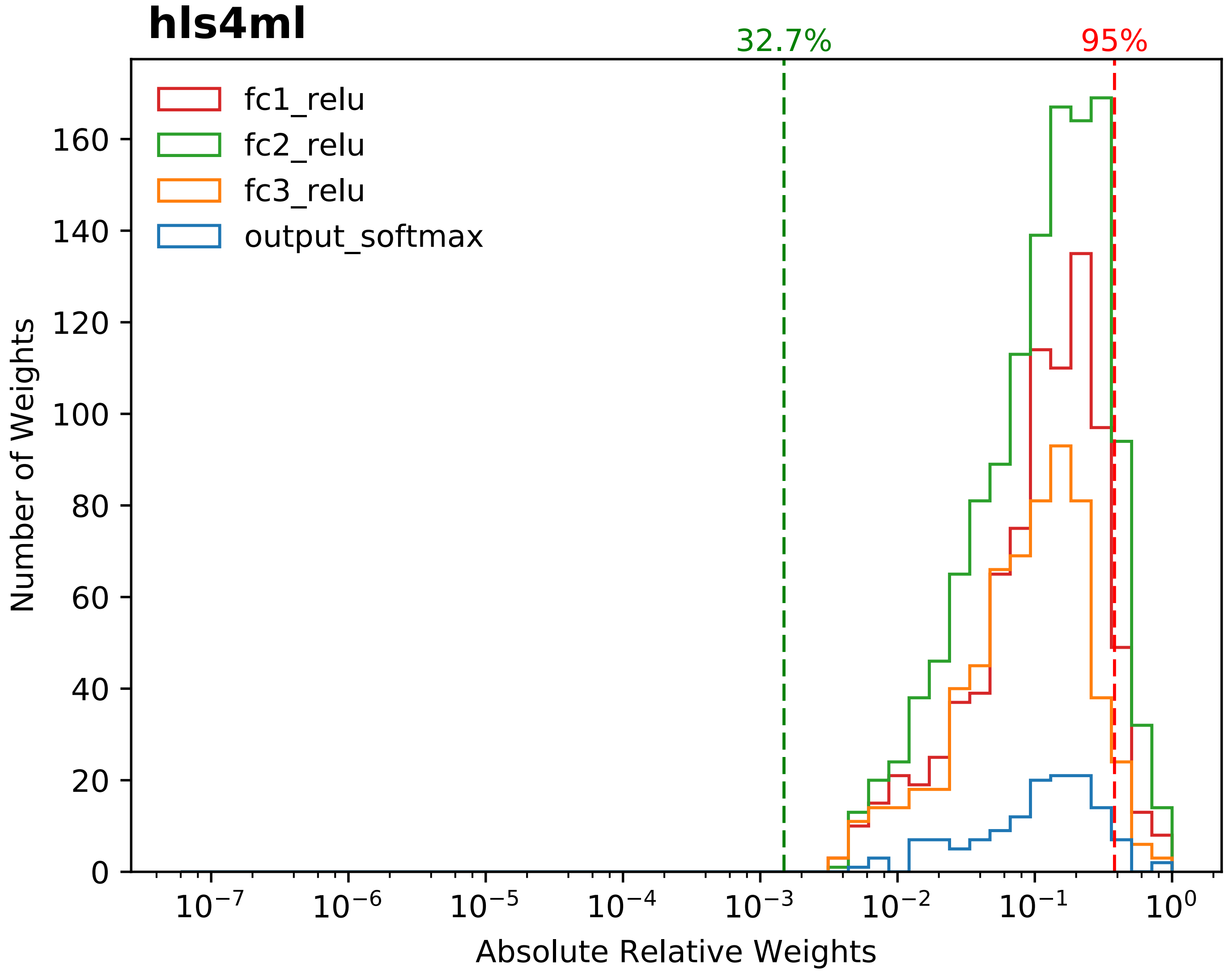


1st iteration

Train with L_1

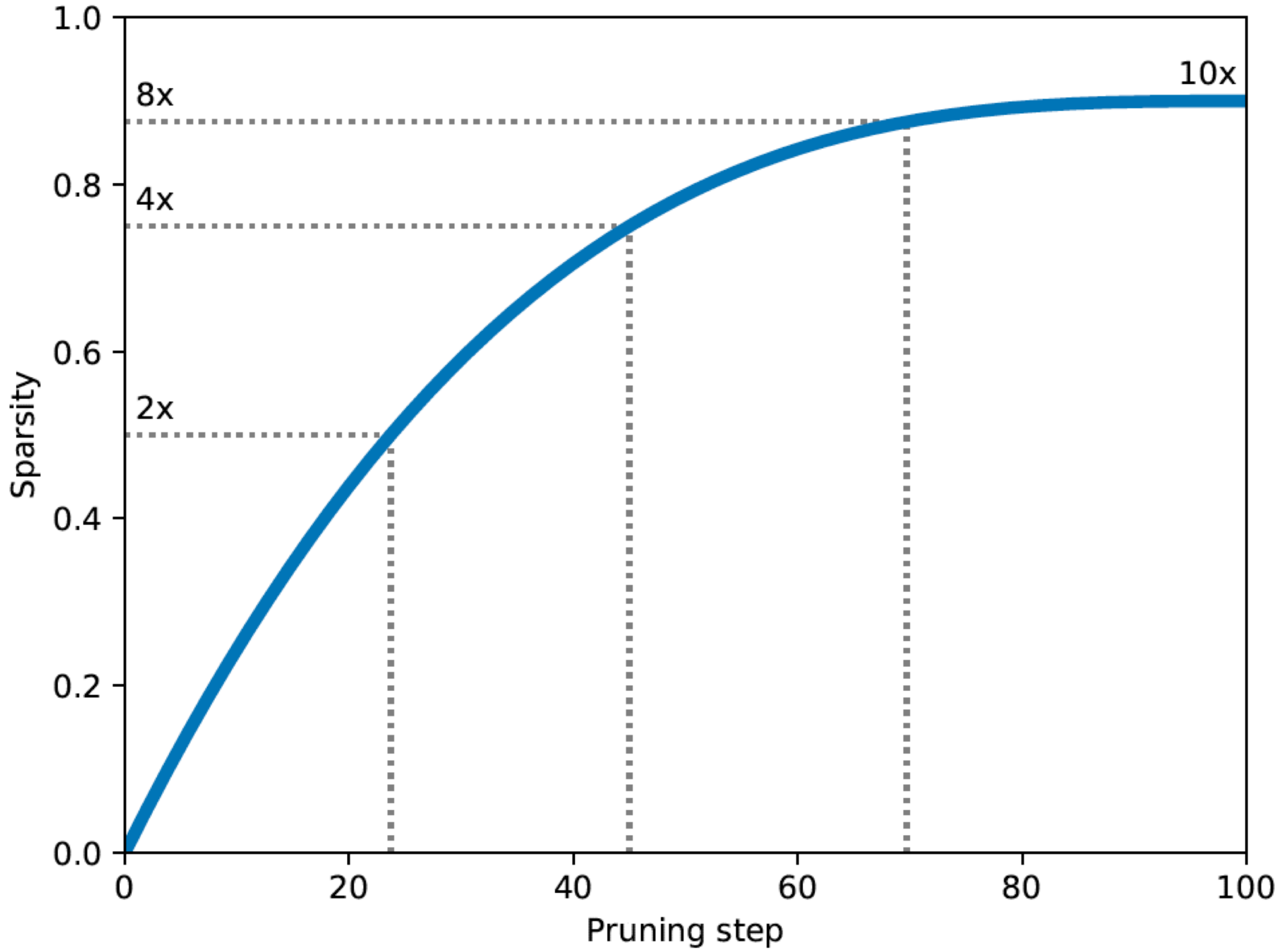


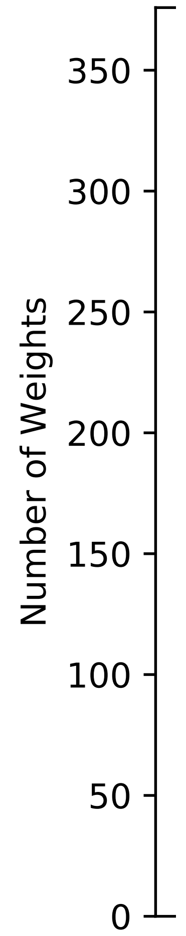
Prune



https://blog.tensorflow.org/2019/05/tf-model-optimization-toolkit-pruning-API.html

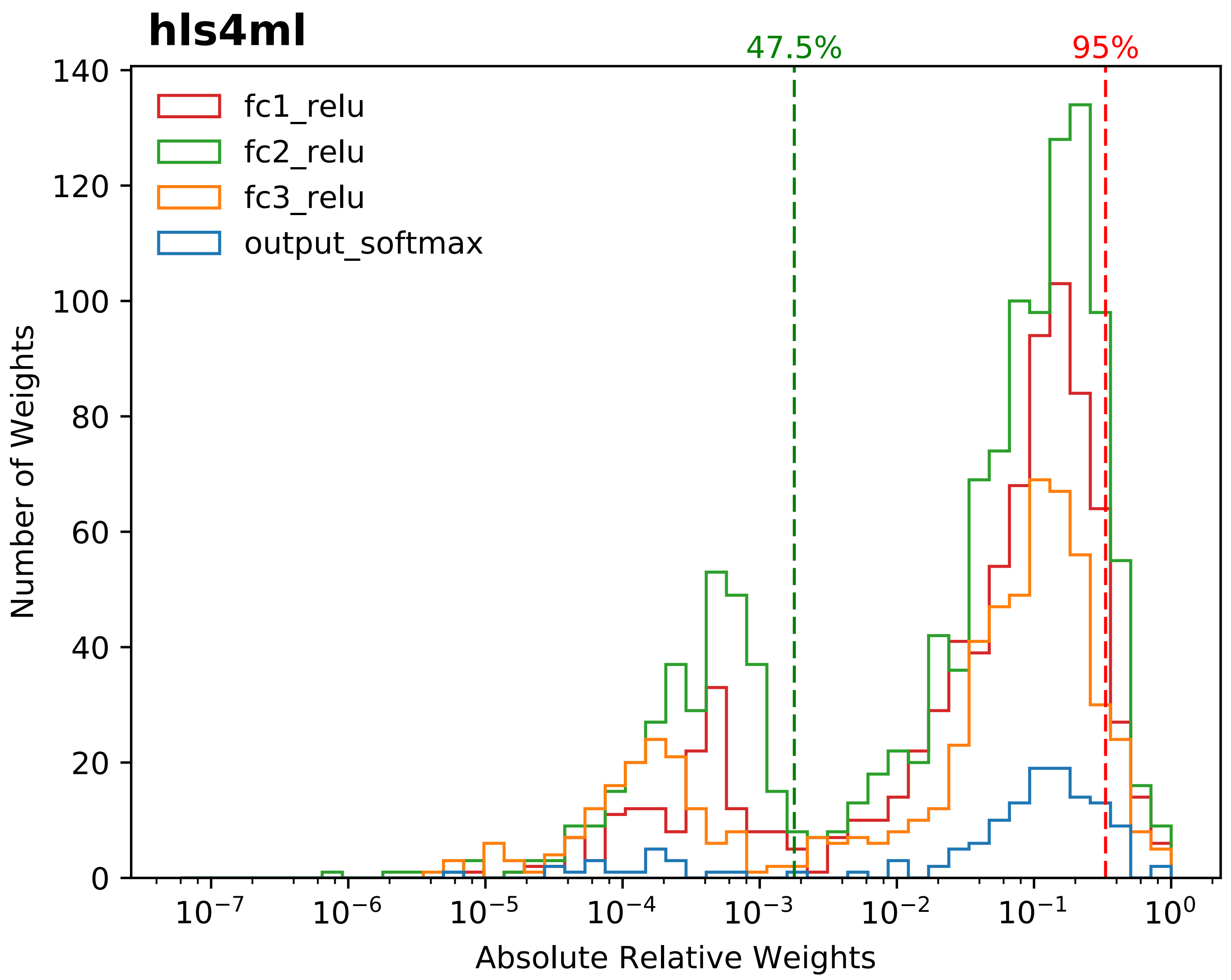
Pruning



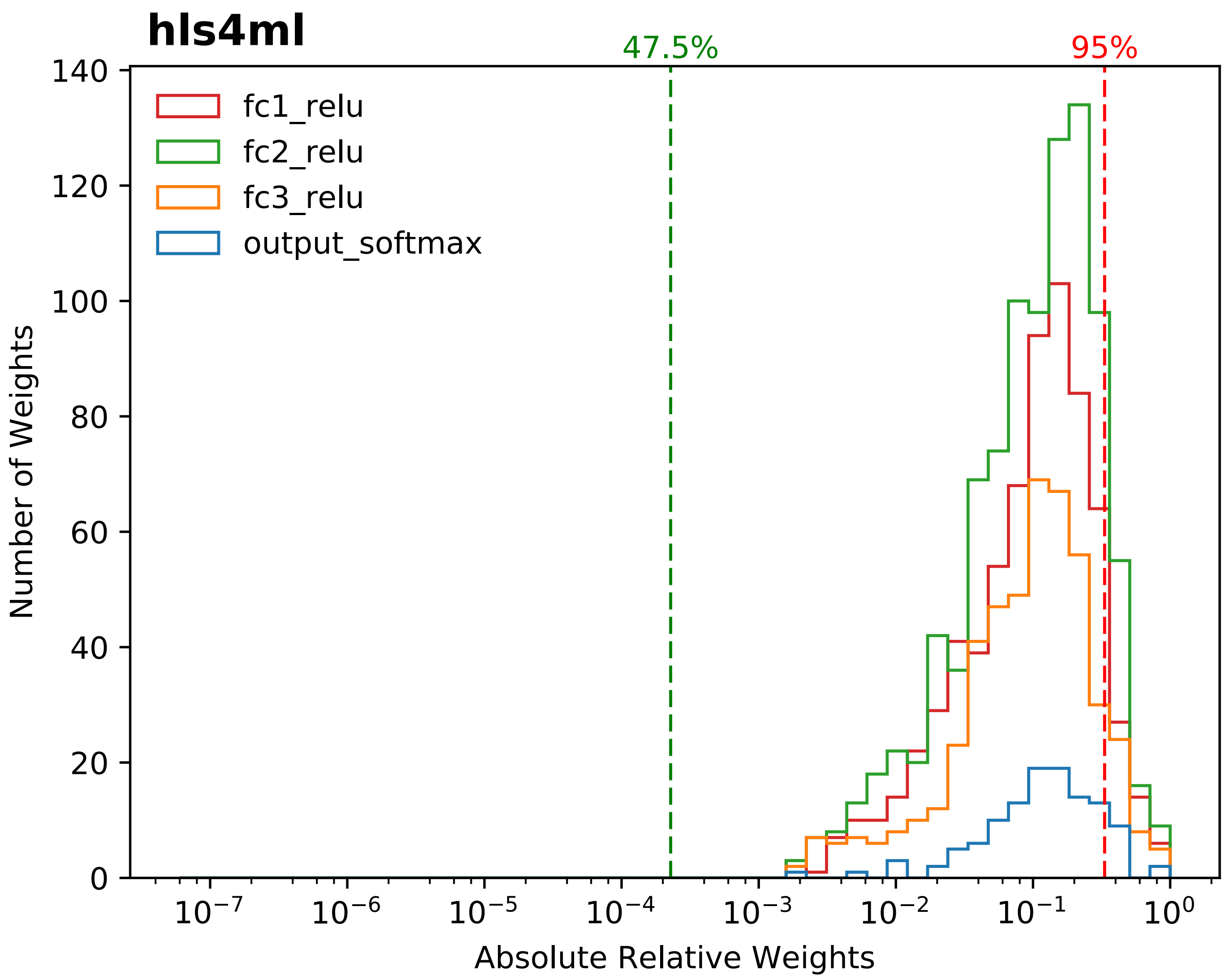


2nd iteration

Retrain with L_1

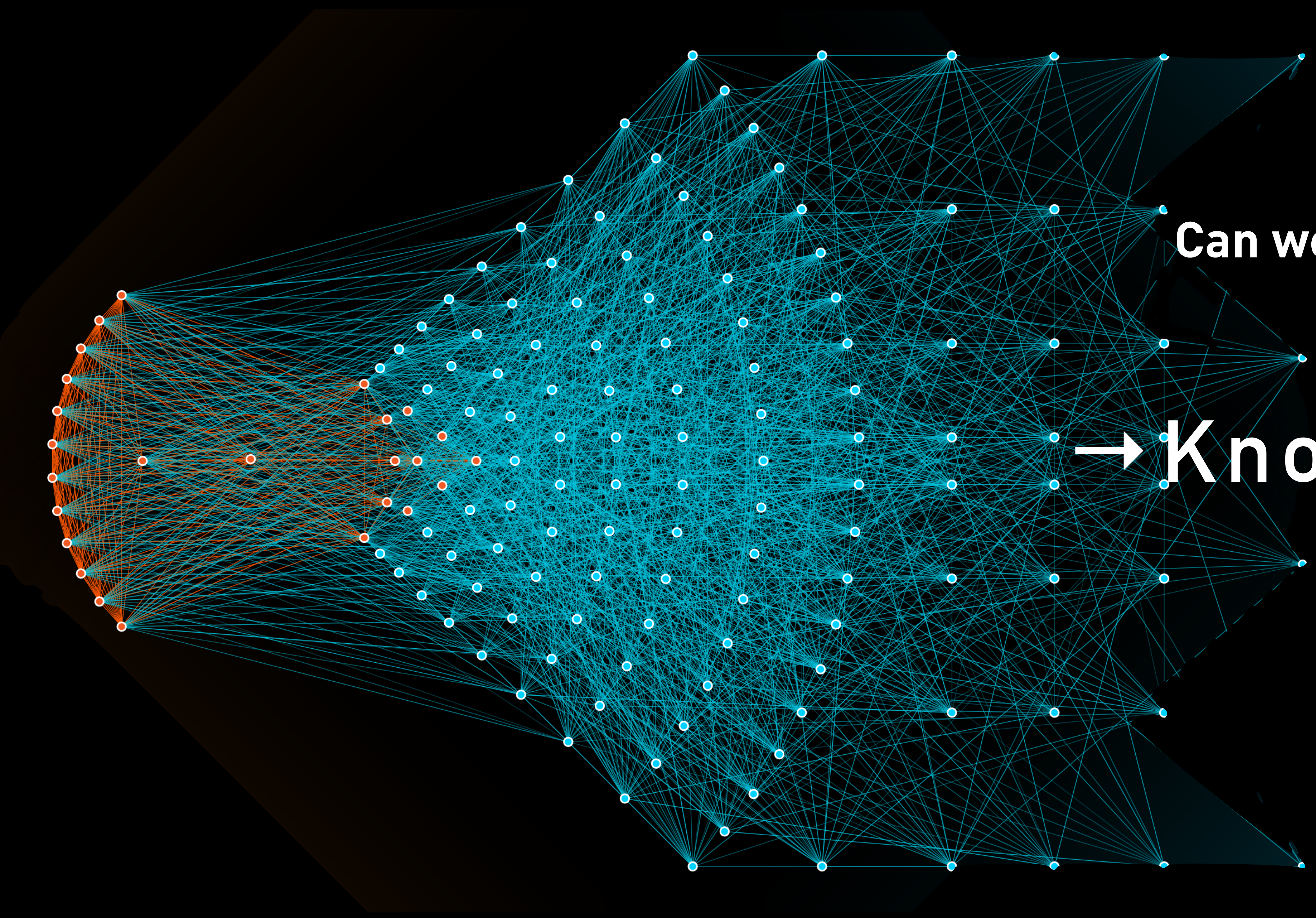


Prune

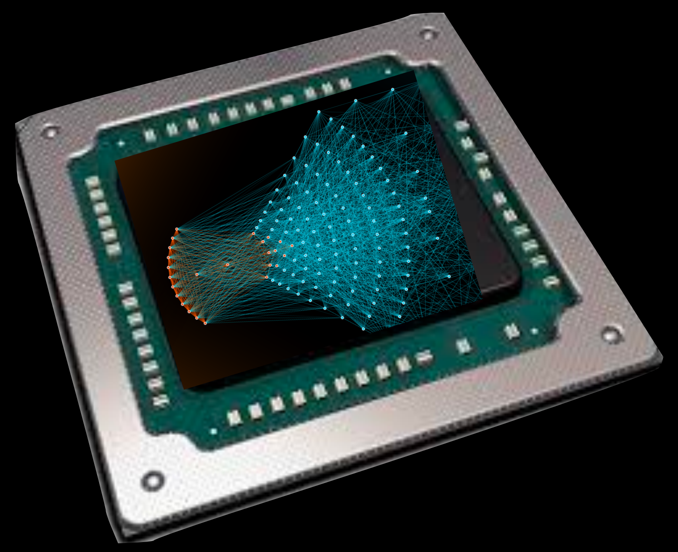


Can we have the best of both worlds?

→Knowledge Distillation



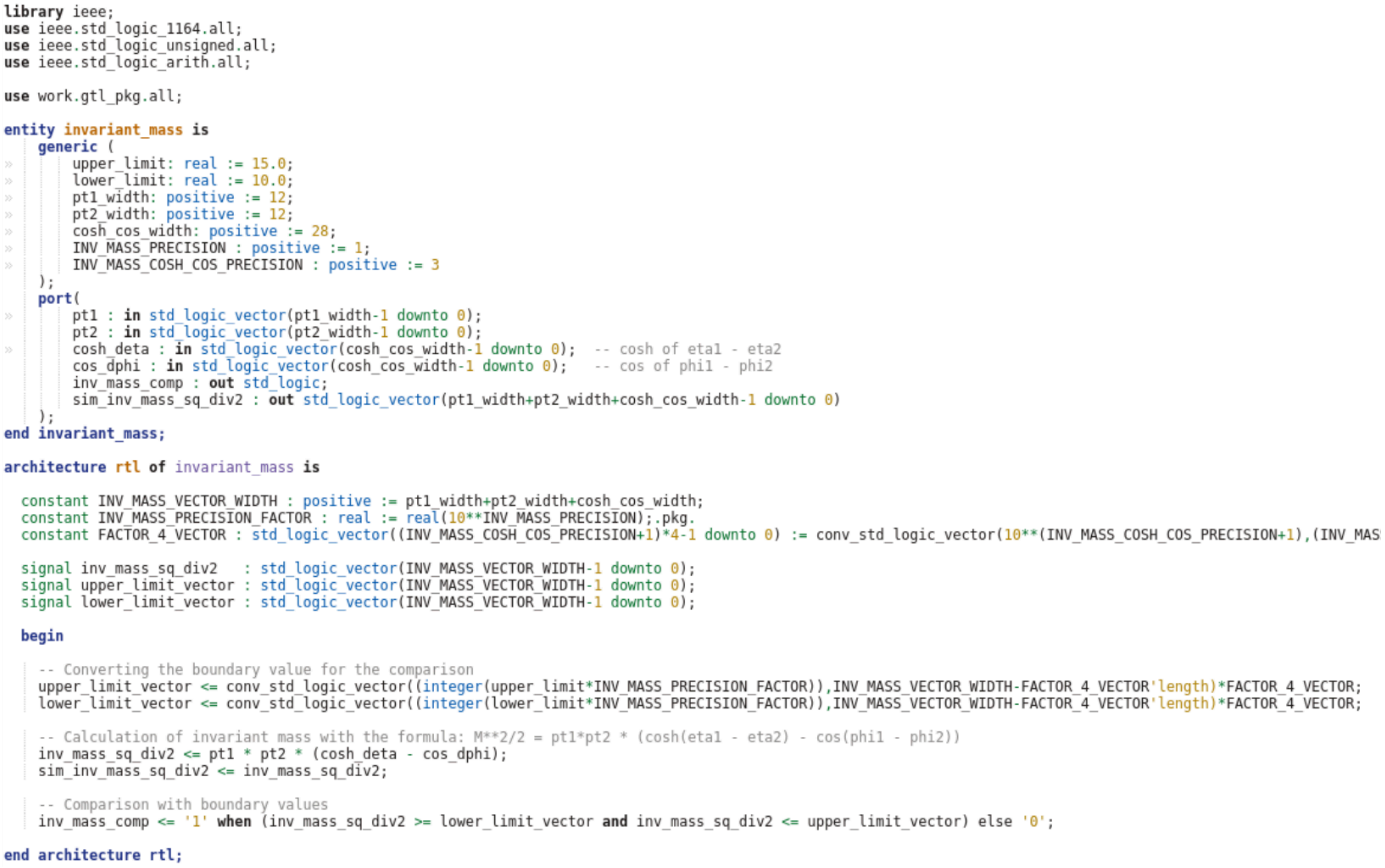
Train



Inference

See talk by Patrick Odagiu (IPA) tomorrow!

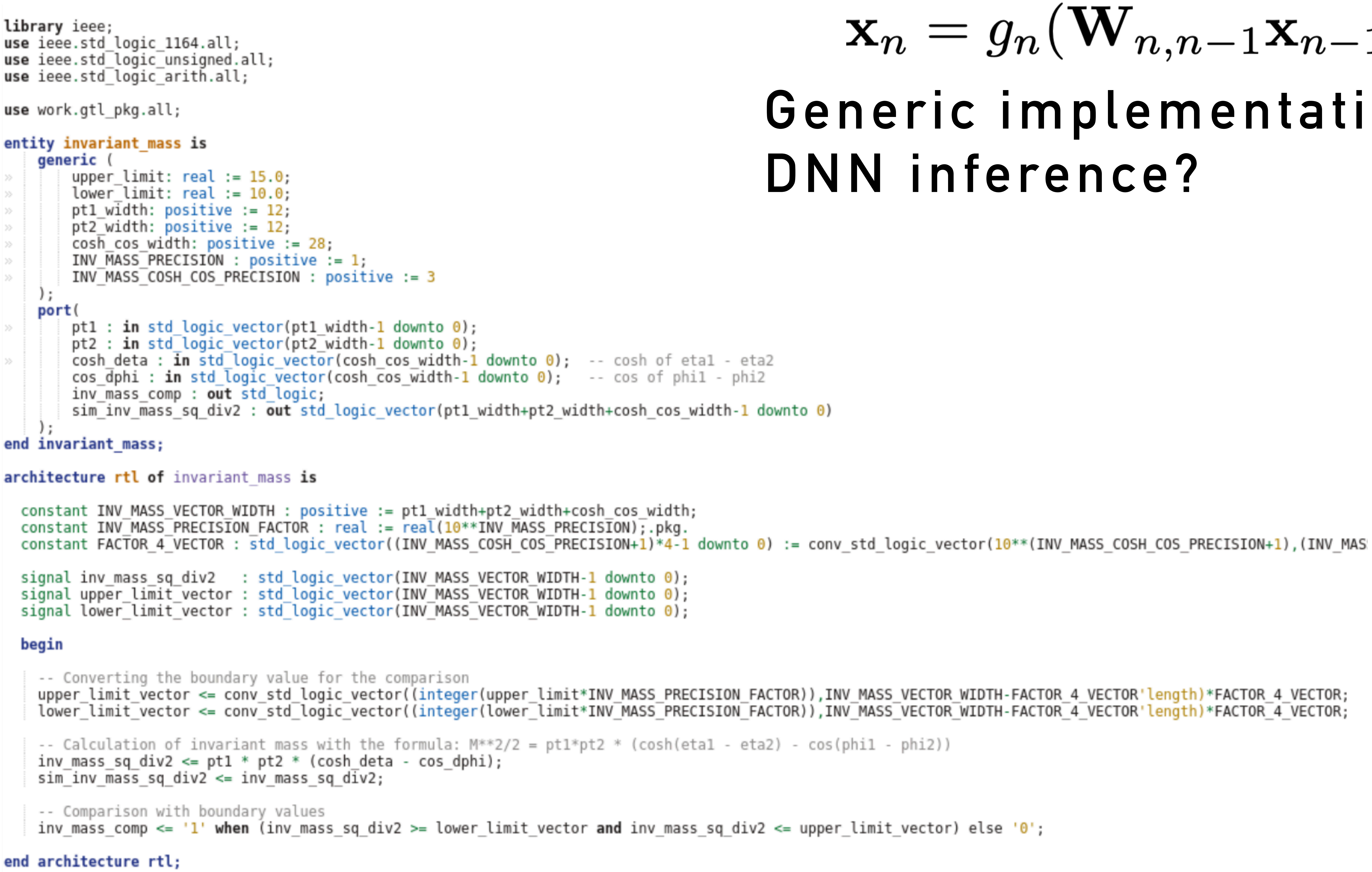
FPGA trigger code



```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;

use work.gtl_pkg.all;

entity invariant_mass is
    generic (
        upper_limit: real := 15.0;
        lower_limit: real := 10.0;
        pt1_width: positive := 12;
        pt2_width: positive := 12;
        cosh_cos_width: positive := 28;
        INV_MASS_PRECISION : positive := 1;
        INV_MASS_COSH_COS_PRECISION : positive := 3
    );
    port(
        pt1 : in std_logic_vector(pt1_width-1 downto 0);
        pt2 : in std_logic_vector(pt2_width-1 downto 0);
        cosh_deta : in std_logic_vector(cosh_cos_width-1 downto 0);  -- cosh of eta1 - eta2
        cos_dphi : in std_logic_vector(cosh_cos_width-1 downto 0);   -- cos of phi1 - phi2
        inv_mass_comp : out std_logic;
        sim_inv_mass_sq_div2 : out std_logic_vector(pt1_width+pt2_width+cosh_cos_width-1 downto 0)
    );
end invariant_mass;

architecture rtl of invariant_mass is

  constant INV_MASS_VECTOR_WIDTH : positive := pt1_width+pt2_width+cosh_cos_width;
  constant INV_MASS_PRECISION_FACTOR : real := real(10**INV_MASS_PRECISION);.pkg.
  constant FACTOR_4_VECTOR : std_logic_vector((INV_MASS_COSH_COS_PRECISION+1)*4-1 downto 0) := conv_std_logic_vector(10**(INV_MASS_COSH_COS_PRECISION+1),(INV_MAS

  signal inv_mass_sq_div2   : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);
  signal upper_limit_vector : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);
  signal lower_limit_vector : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);

  begin

    -- Converting the boundary value for the comparison
    upper_limit_vector <= conv_std_logic_vector((integer(upper_limit*INV_MASS_PRECISION_FACTOR)),INV_MASS_VECTOR_WIDTH-FACTOR_4_VECTOR'length)*FACTOR_4_VECTOR;
    lower_limit_vector <= conv_std_logic_vector((integer(lower_limit*INV_MASS_PRECISION_FACTOR)),INV_MASS_VECTOR_WIDTH-FACTOR_4_VECTOR'length)*FACTOR_4_VECTOR;

    -- Calculation of invariant mass with the formula: M**2/2 = pt1*pt2 * (cosh(eta1 - eta2) - cos(phi1 - phi2))
    inv_mass_sq_div2 <= pt1 * pt2 * (cosh_deta - cos_dphi);
    sim_inv_mass_sq_div2 <= inv_mass_sq_div2;

    -- Comparison with boundary values
    inv_mass_comp <= '1' when (inv_mass_sq_div2 >= lower_limit_vector and inv_mass_sq_div2 <= upper_limit_vector) else '0';

end architecture rtl;
```

M. Jeitler

FPGA trigger code



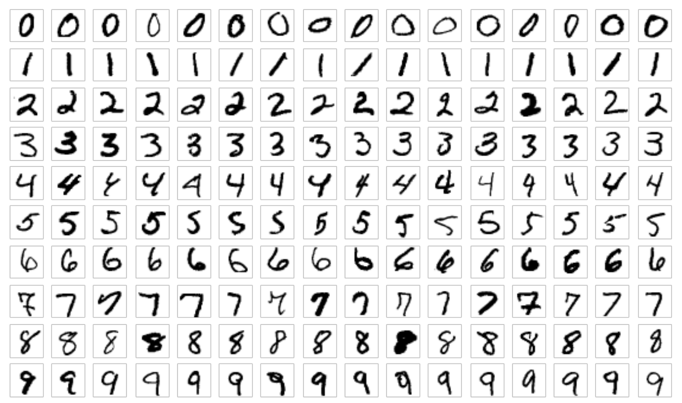
```vhdl
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;

use work.gtl_pkg.all;

entity invariant_mass is
    generic (
        upper_limit: real := 15.0;
        lower_limit: real := 10.0;
        pt1_width: positive := 12;
        pt2_width: positive := 12;
        cosh_cos_width: positive := 28;
        INV_MASS_PRECISION : positive := 1;
        INV_MASS_COSH_COS_PRECISION : positive := 3
    );
    port(
        pt1 : in std_logic_vector(pt1_width-1 downto 0);
        pt2 : in std_logic_vector(pt2_width-1 downto 0);
        cosh_deta : in std_logic_vector(cosh_cos_width-1 downto 0);  -- cosh of eta1 - eta2
        cos_dphi : in std_logic_vector(cosh_cos_width-1 downto 0);   -- cos of phi1 - phi2
        inv_mass_comp : out std_logic;
        sim_inv_mass_sq_div2 : out std_logic_vector(pt1_width+pt2_width+cosh_cos_width-1 downto 0)
    );
end invariant_mass;

architecture rtl of invariant_mass is

  constant INV_MASS_VECTOR_WIDTH : positive := pt1_width+pt2_width+cosh_cos_width;
  constant INV_MASS_PRECISION_FACTOR : real := real(10**INV_MASS_PRECISION);.pkg.
  constant FACTOR_4_VECTOR : std_logic_vector((INV_MASS_COSH_COS_PRECISION+1)*4-1 downto 0) := conv_std_logic_vector(10**(INV_MASS_COSH_COS_PRECISION+1),(INV_MAS

  signal inv_mass_sq_div2   : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);
  signal upper_limit_vector : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);
  signal lower_limit_vector : std_logic_vector(INV_MASS_VECTOR_WIDTH-1 downto 0);

  begin

    -- Converting the boundary value for the comparison
    upper_limit_vector <= conv_std_logic_vector((integer(upper_limit*INV_MASS_PRECISION_FACTOR)),INV_MASS_VECTOR_WIDTH-FACTOR_4_VECTOR'length)*FACTOR_4_VECTOR;
    lower_limit_vector <= conv_std_logic_vector((integer(lower_limit*INV_MASS_PRECISION_FACTOR)),INV_MASS_VECTOR_WIDTH-FACTOR_4_VECTOR'length)*FACTOR_4_VECTOR;

    -- Calculation of invariant mass with the formula: M**2/2 = pt1*pt2 * (cosh(eta1 - eta2) - cos(phi1 - phi2))
    inv_mass_sq_div2 <= pt1 * pt2 * (cosh_deta - cos_dphi);
    sim_inv_mass_sq_div2 <= inv_mass_sq_div2;

    -- Comparison with boundary values
    inv_mass_comp <= '1' when (inv_mass_sq_div2 >= lower_limit_vector and inv_mass_sq_div2 <= upper_limit_vector) else '0';

end architecture rtl;
```

$$\mathbf{x}_n = g_n(\mathbf{W}_{n,n-1}\mathbf{x}_{n-1} + \mathbf{b}_n)$$

Generic implementations for DNN inference?

M. Jeitler

From Python code…….. to hardware blocks that can do inference in nanoseconds!

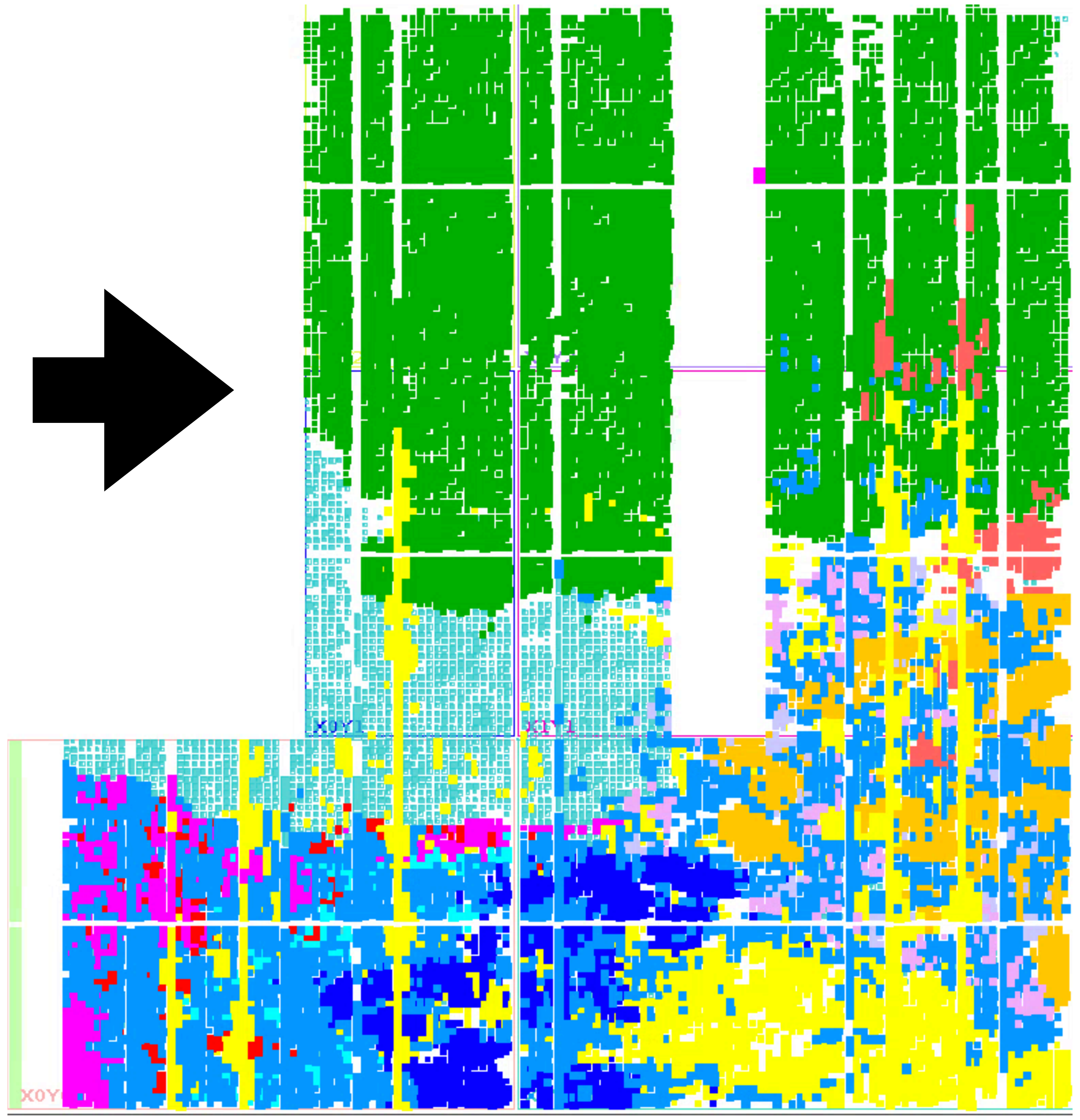


Conv2D
ReLU
MaxPool2D
Conv2D
ReLU
MaxPool2D
Flatten
Dense
Softmax

Prediction

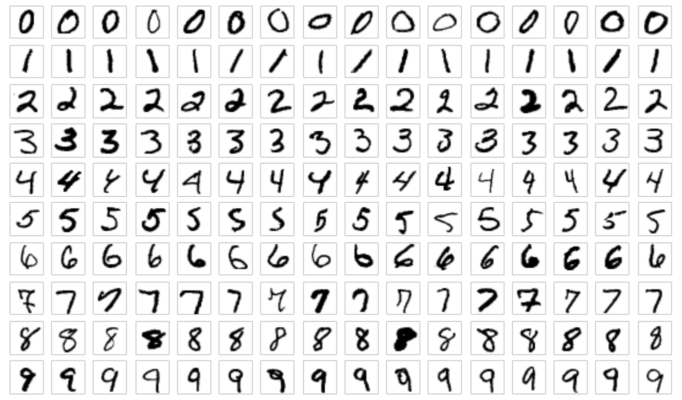
hls4ml

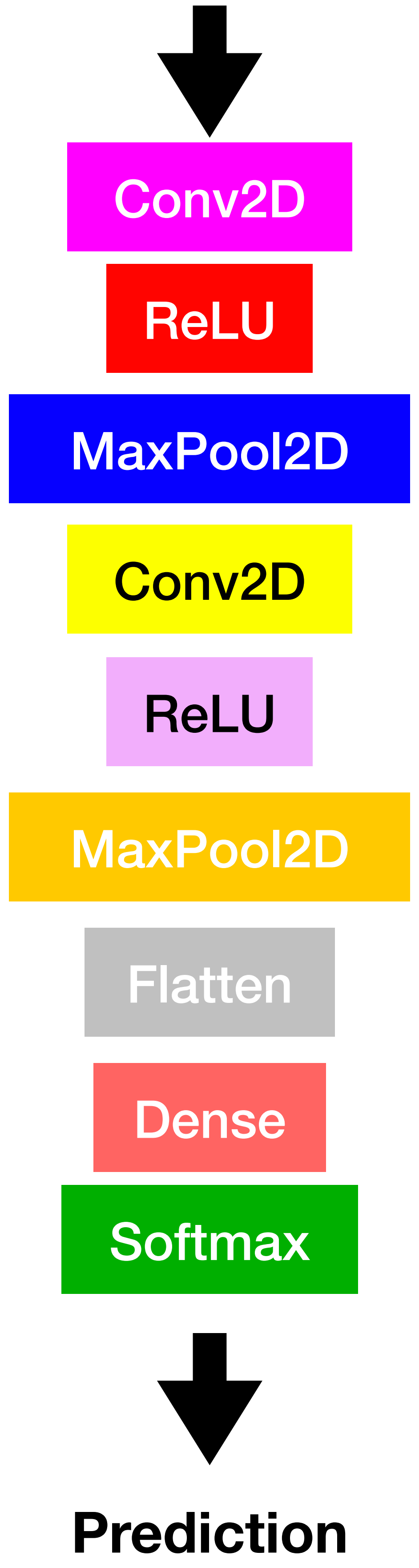
pynq-z2 floorplan



Our work here at IPA!
P. Odagiu, C. Sun, A. Oliver, T. Årrestad

hls4ml





Conv2D

ReLU

MaxPool2D

Conv2D

ReLU

MaxPool2D

Flatten

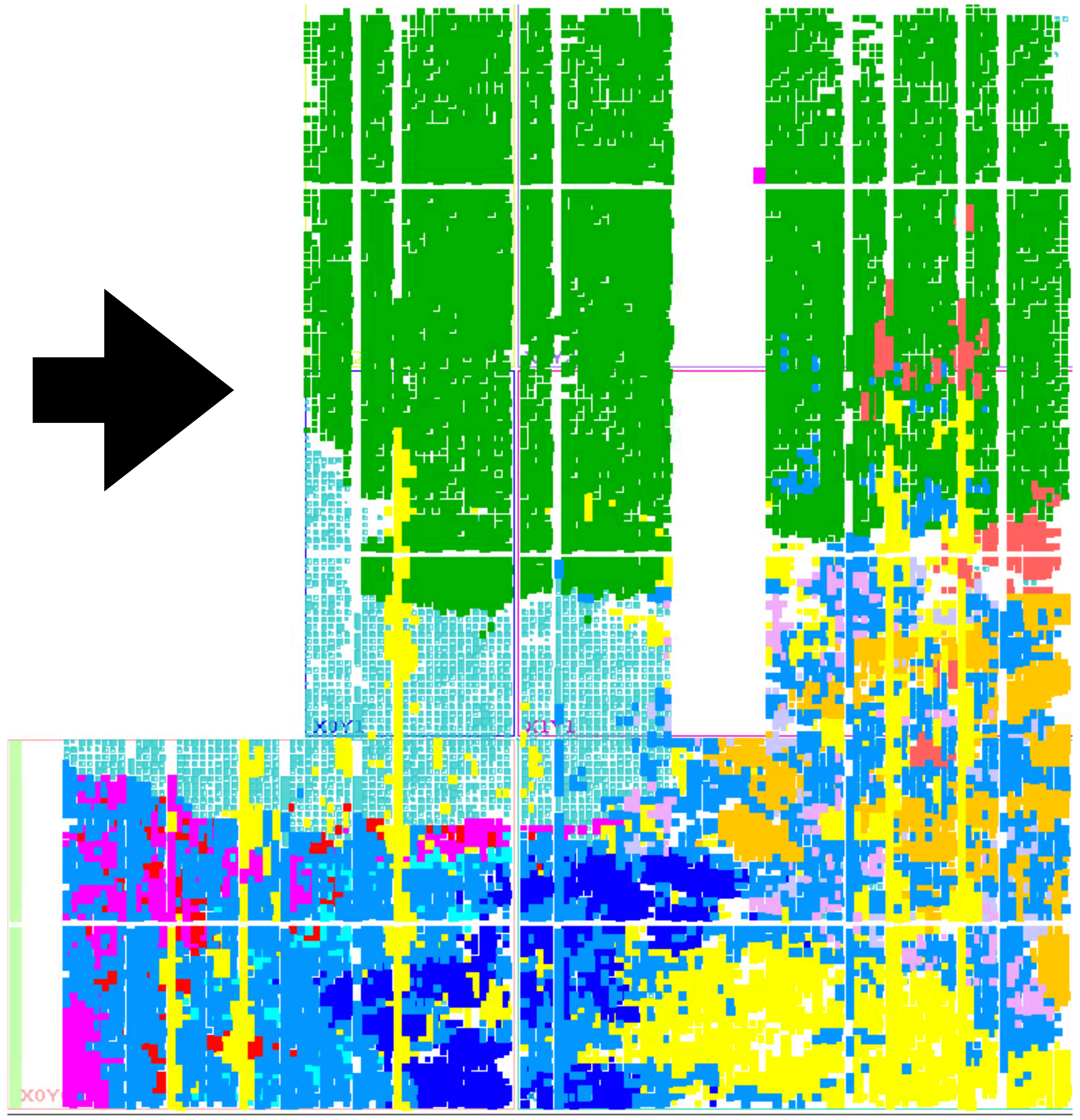
Dense

Softmax

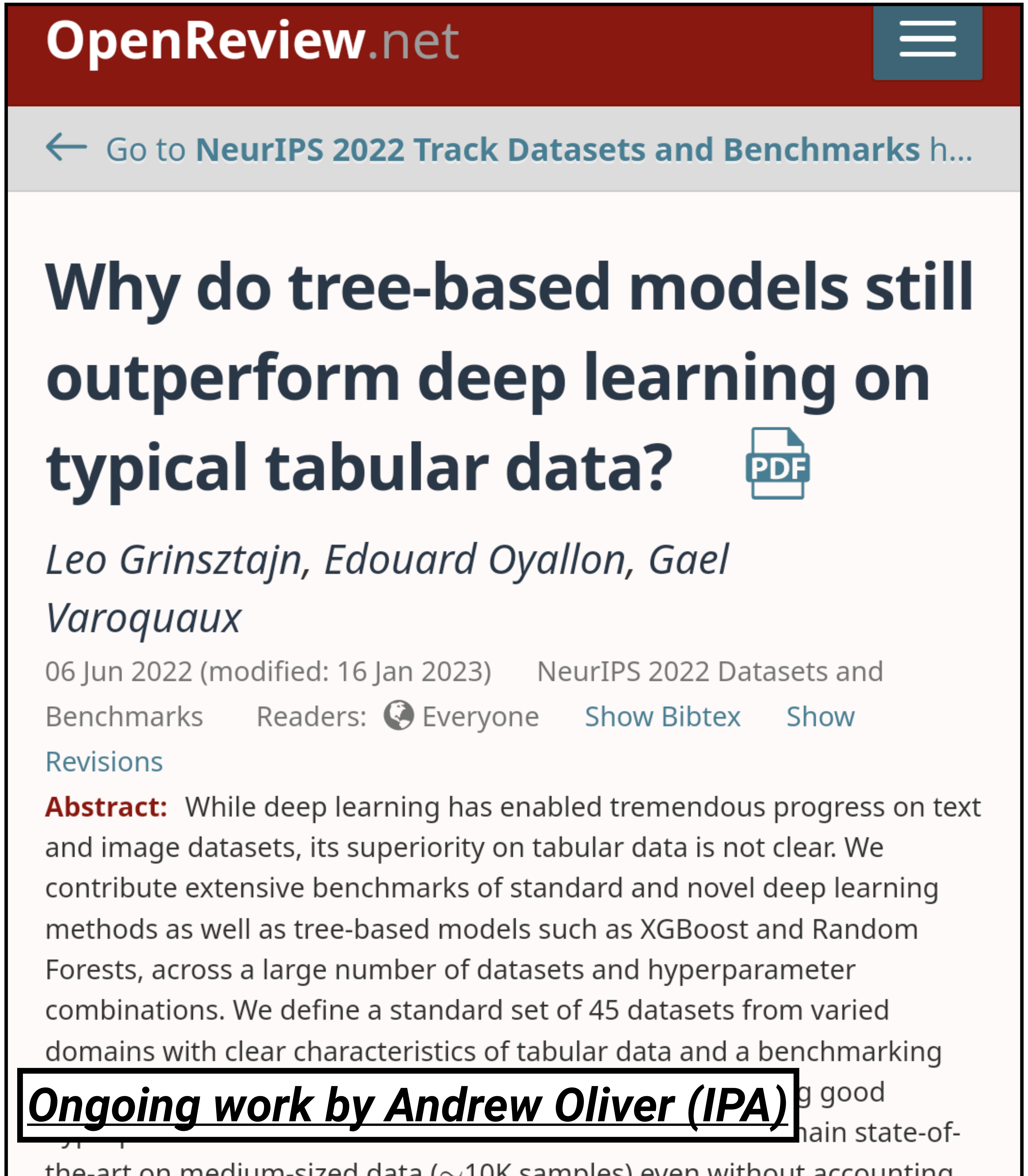
Prediction

```
from hls4ml import …
import tensorflow as tf

# train or load a model
model = … # e.g. tf.keras.models.load_model(…)

# make a config template
cfg = config_from_keras_model(model,
granularity=‘name’)

# tune the config
cfg[‘LayerName’][‘layer2’][‘ReuseFactor’] = 4

# do the conversion
hmodel = convert_from_keras_model(model, cfg)

# write and compile the HLS
hmodel.compile()

# run bit accurate emulation
y_tf = model.predict(x)
y_hls = hmodel.predict(x)

# do some validation
np.testing.assert_allclose(y_tf, y_hls)

# run HLS synthesis
hmodel.build()
```

pynq-z2 floorplan



Our work here at IPA!
P. Odagiu, C. Sun, A. Oliver, T. Årrestad



← Go to **NeurIPS 2022 Track Datasets and Benchmarks** h...

Why do tree-based models still outperform deep learning on typical tabular data?

Leo Grinsztajn, Edouard Oyallon, Gael Varoquaux

06 Jun 2022 (modified: 16 Jan 2023) NeurIPS 2022 Datasets and Benchmarks Readers: Everyone Show Bibtex Show Revisions

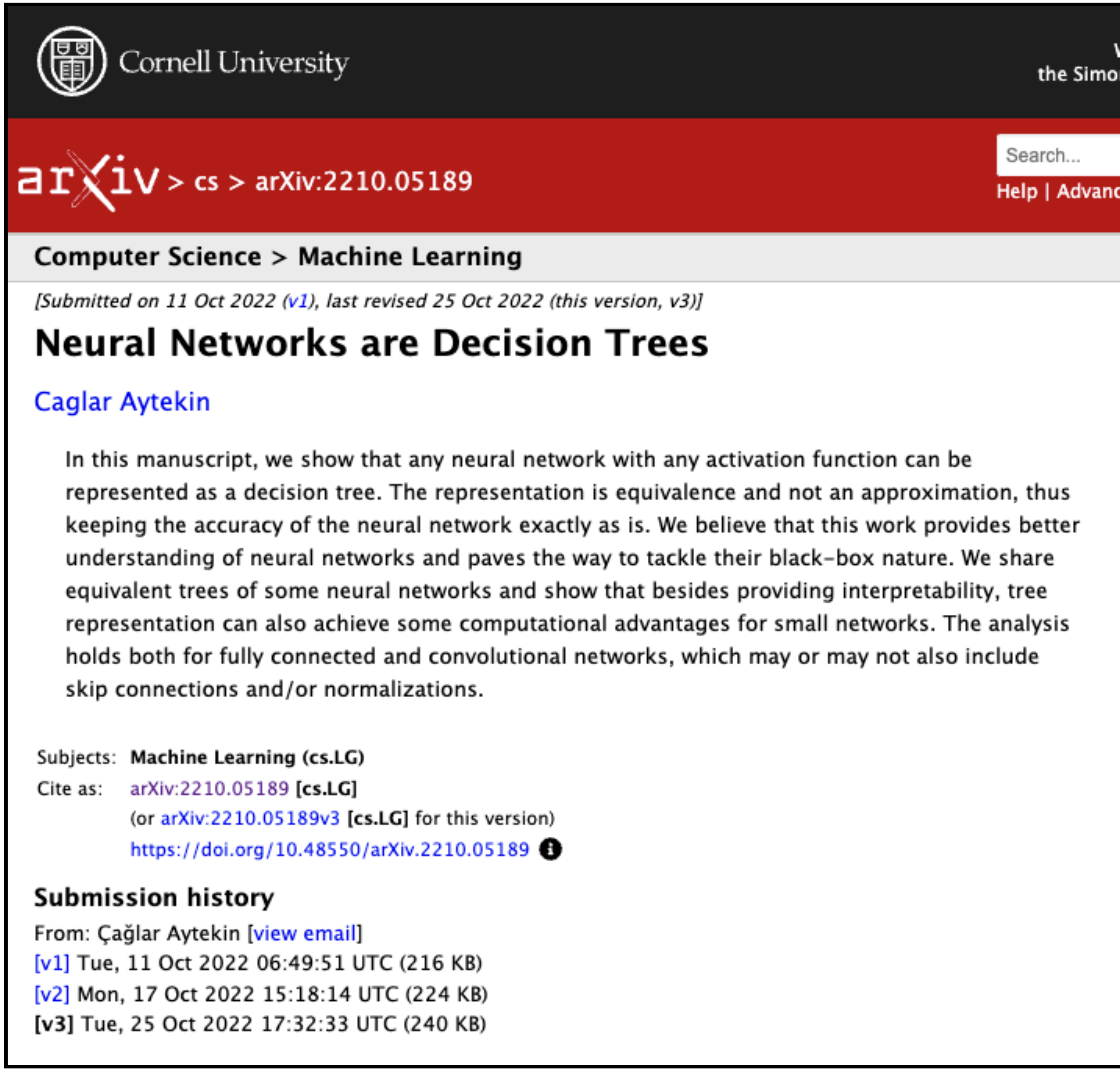
Abstract: While deep learning has enabled tremendous progress on text and image datasets, its superiority on tabular data is not clear. We contribute extensive benchmarks of standard and novel deep learning methods as well as tree-based models such as XGBoost and Random Forests, across a large number of datasets and hyperparameter combinations. We define a standard set of 45 datasets from varied domains with clear characteristics of tabular data and a benchmarking

Ongoing work by Andrew Oliver (IPA)

g good

hain state-of-

the-art on medium-sized data (~10K samples) even without accounting



Computer Science > Machine Learning

[Submitted on 11 Oct 2022 (v1), last revised 25 Oct 2022 (this version, v3)]

Neural Networks are Decision Trees

Caglar Aytekin

In this manuscript, we show that any neural network with any activation function can be represented as a decision tree. The representation is equivalence and not an approximation, thus keeping the accuracy of the neural network exactly as is. We believe that this work provides better understanding of neural networks and paves the way to tackle their black-box nature. We share equivalent trees of some neural networks and show that besides providing interpretability, tree representation can also achieve some computational advantages for small networks. The analysis holds both for fully connected and convolutional networks, which may or may not also include skip connections and/or normalizations.

Subjects: **Machine Learning (cs.LG)**
Cite as: arXiv:2210.05189 **[cs.LG]**
(or arXiv:2210.05189v3 **[cs.LG]** for this version)
https://doi.org/10.48550/arXiv.2210.05189

Submission history

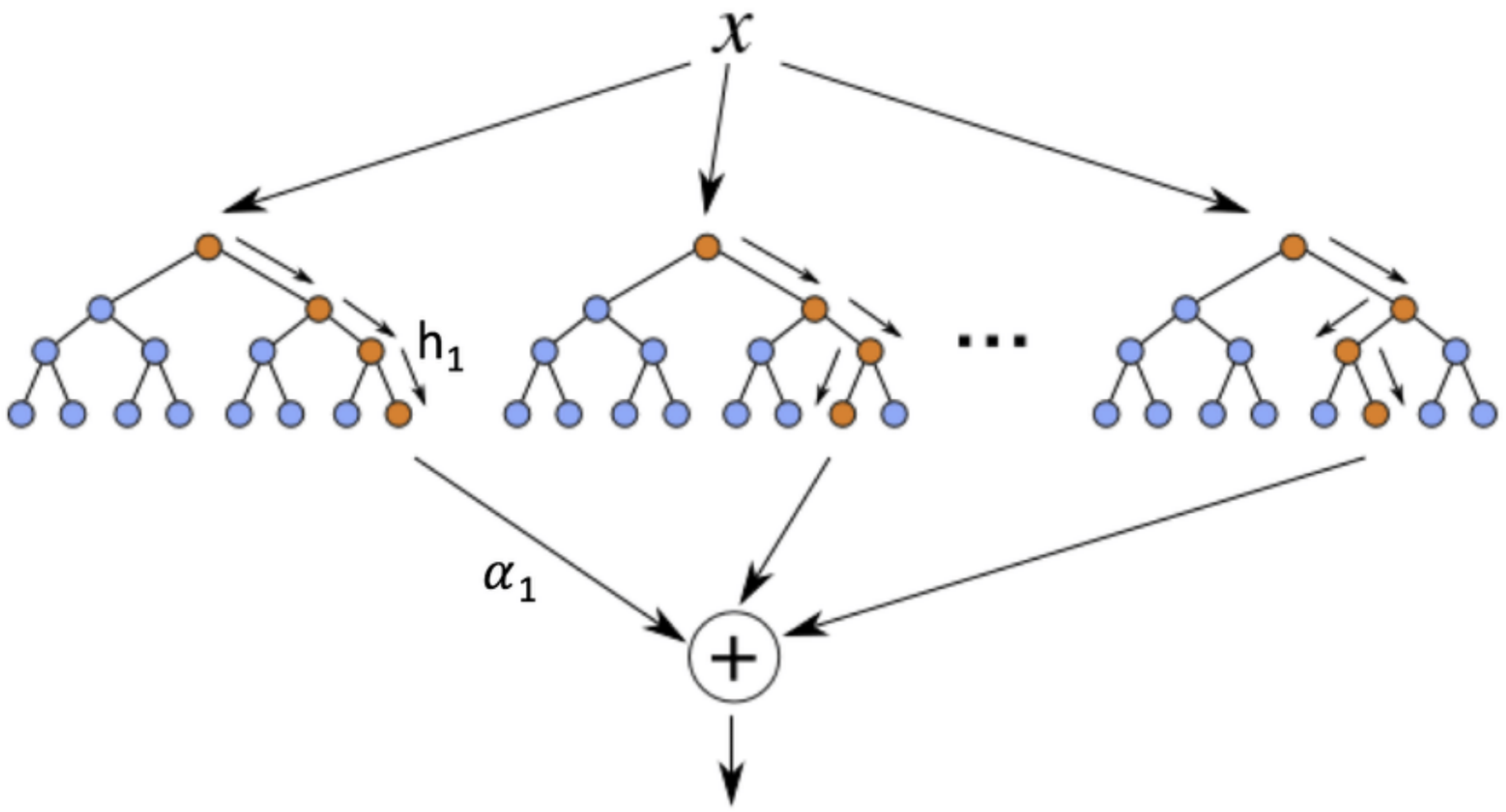
From: Çağlar Aytekin [view email]
[v1] Tue, 11 Oct 2022 06:49:51 UTC (216 KB)
[v2] Mon, 17 Oct 2022 15:18:14 UTC (224 KB)
[v3] Tue, 25 Oct 2022 17:32:33 UTC (240 KB)

https://arxiv.org/abs/2210.05189

%VU9P	Accuracy	Latency	DSP	LUT
qDNN	75.6%	40 ns	22 (~0%)	1%
BDT	74.9%	5 ns	-	0.5%

Google AI

TensorFlow
Decision Forests



IPA

CERN

Conifer

Ongoing work by Andrew Oliver (IPA)

Where and how are we using this at LHC?

Geneva
Geneva Lake
LHCb
ATLAS
ALICE
CMS
LHC

Geneva
Geneva Lake
LHCb
AT
CMS
LHC
XILINX
VIRTEX 5
XC5VLX30

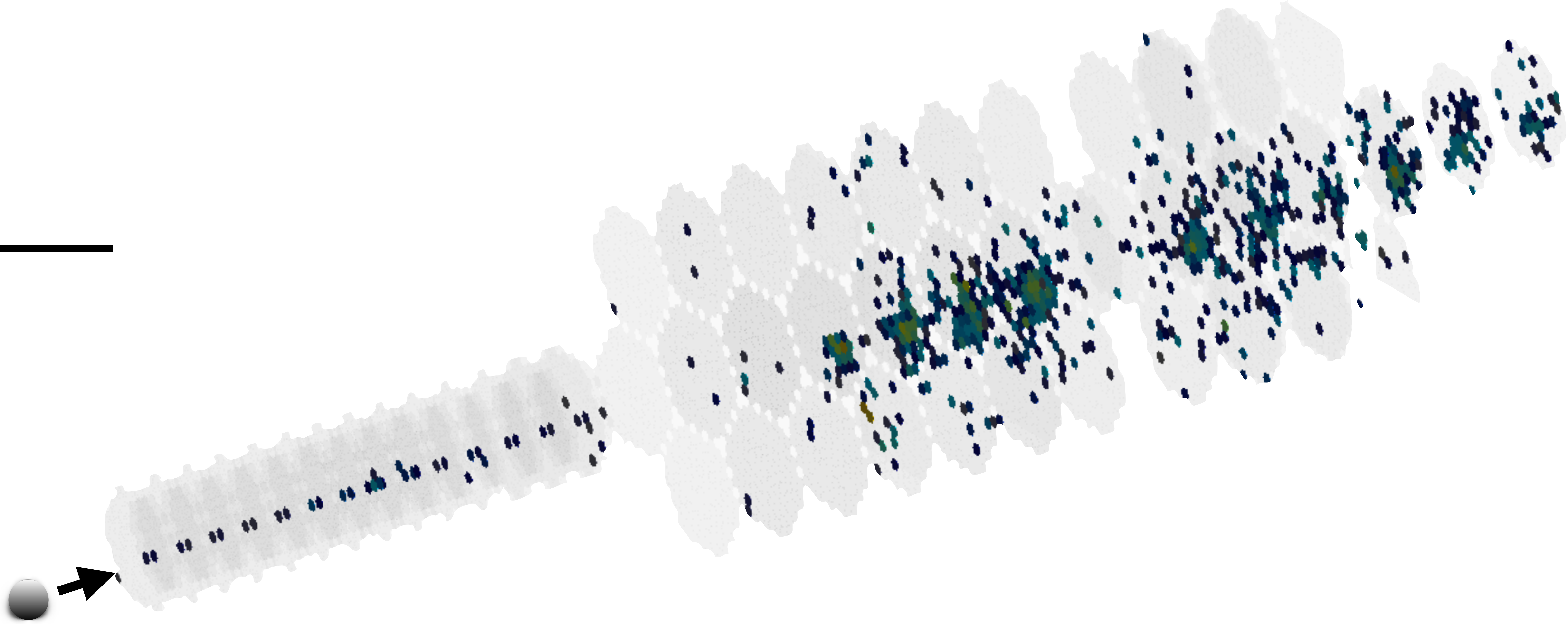
Petabytes of data generated

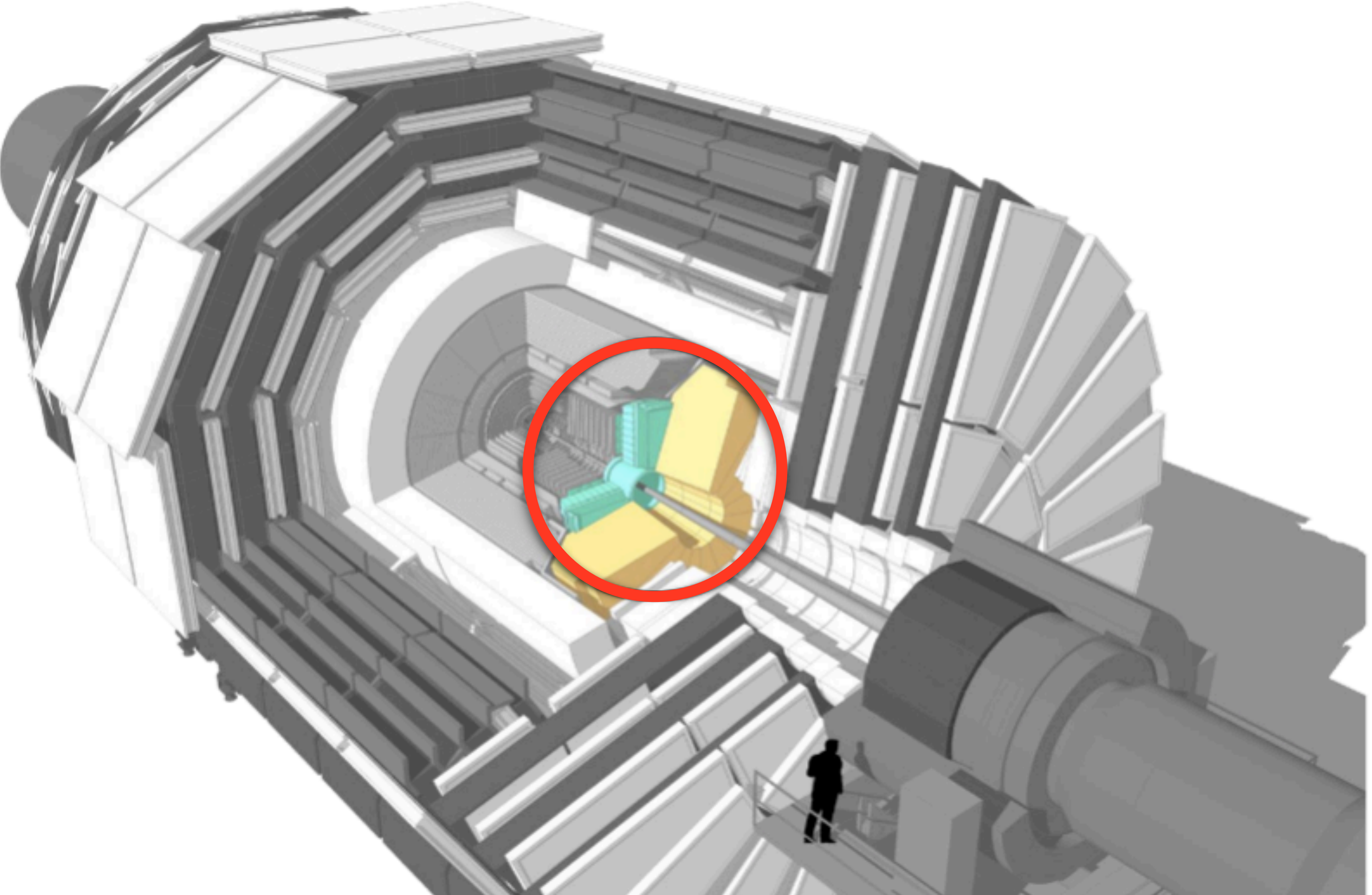
Terabytes of data to L1!

ML for compression

CMS High Granularity calorimeter

- 6.5 million readout channels, 50 layers

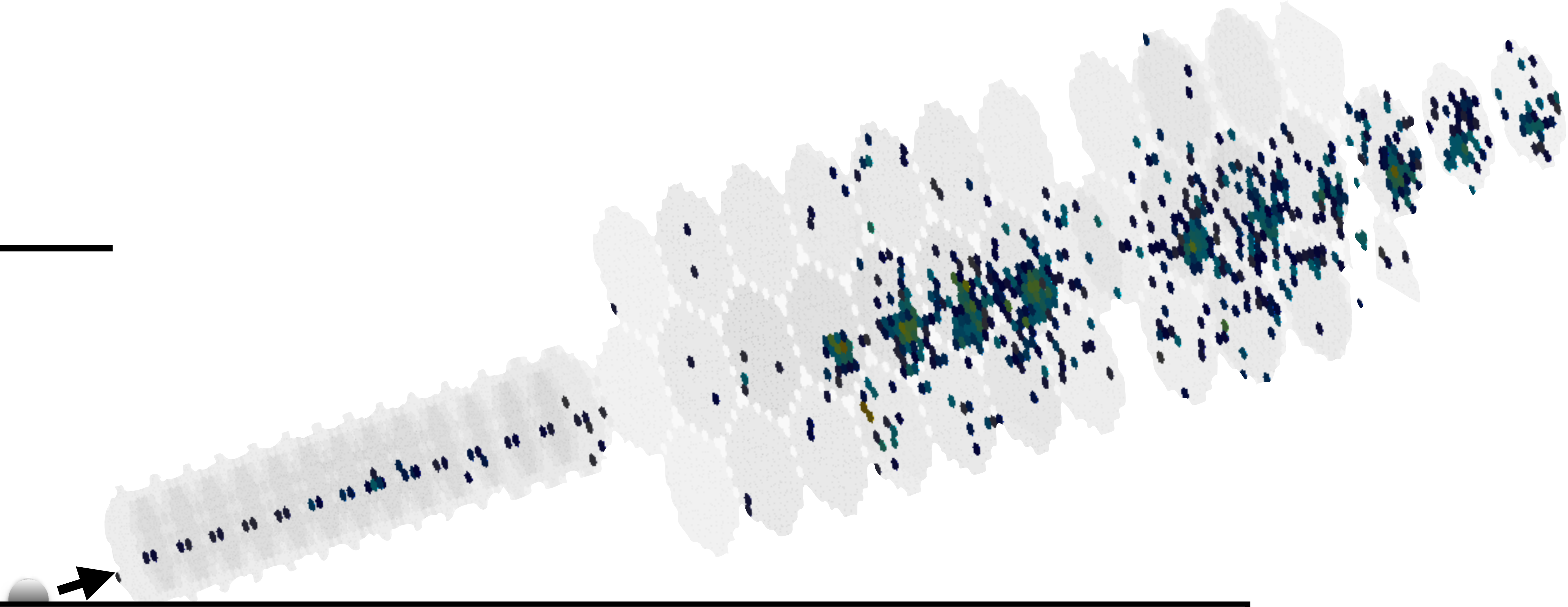




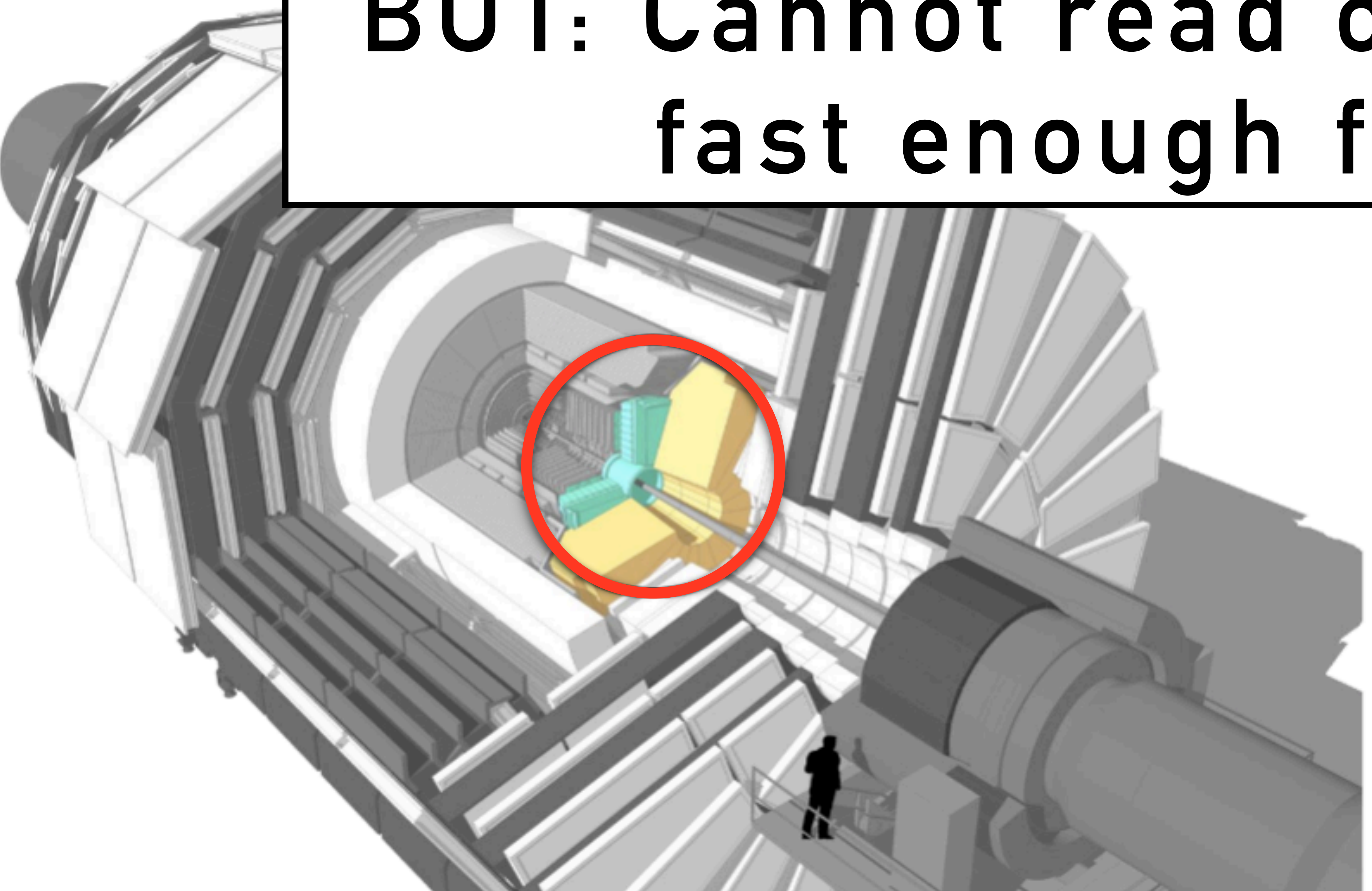
ML for compression

CMS High Granularity calorimeter

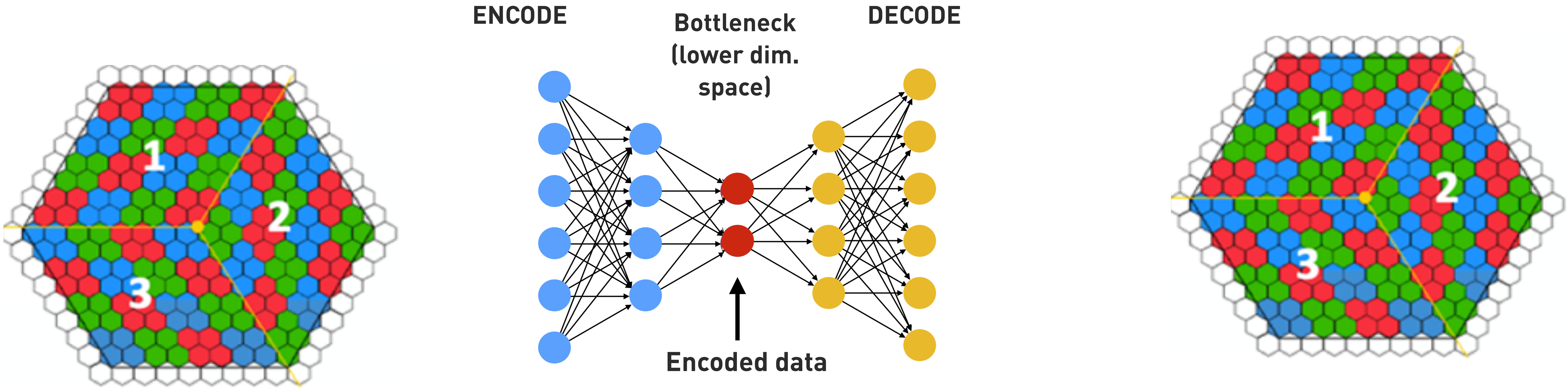
- 6.5 million readout channels, 50 layers



BUT: Cannot read out all these channels fast enough for L1 to trigger!

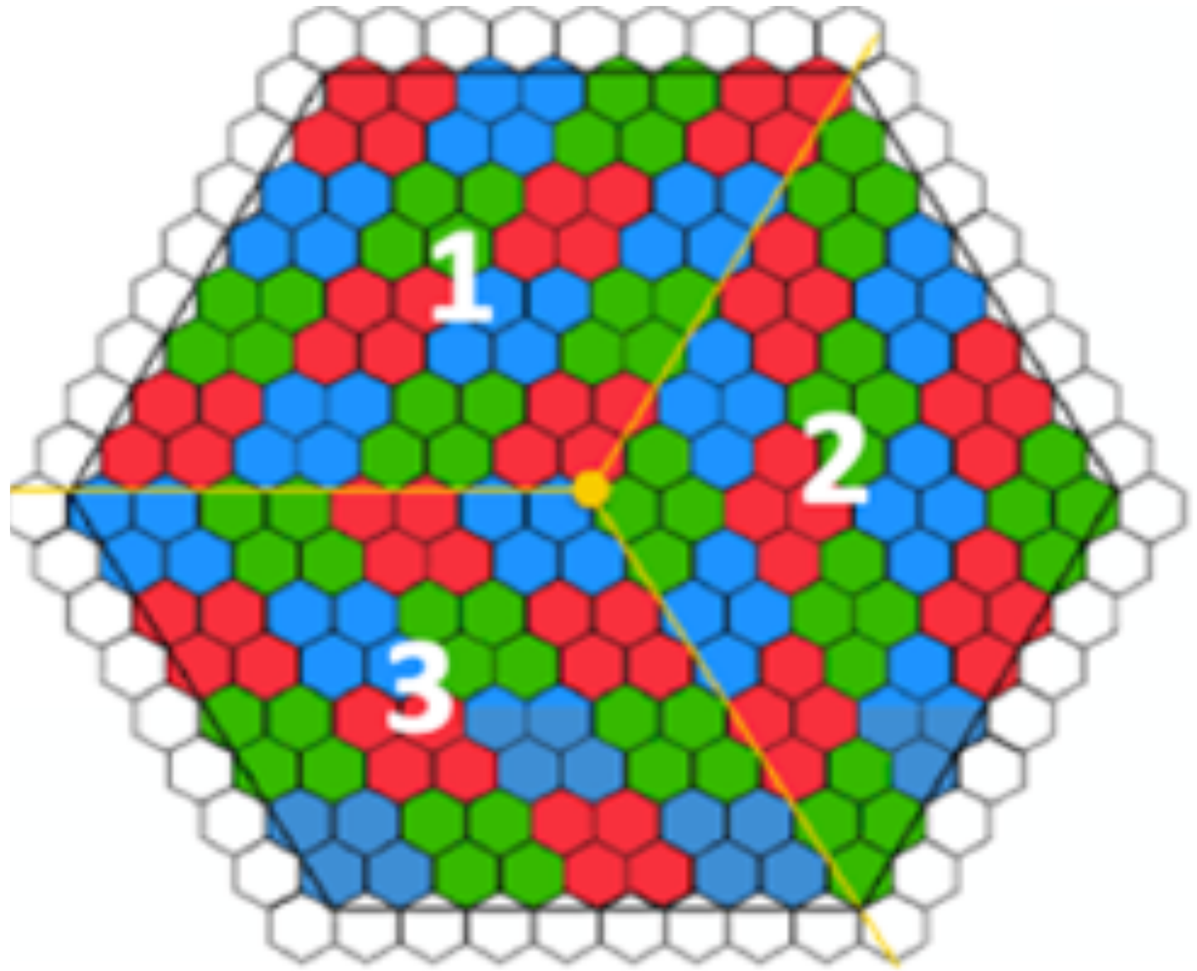


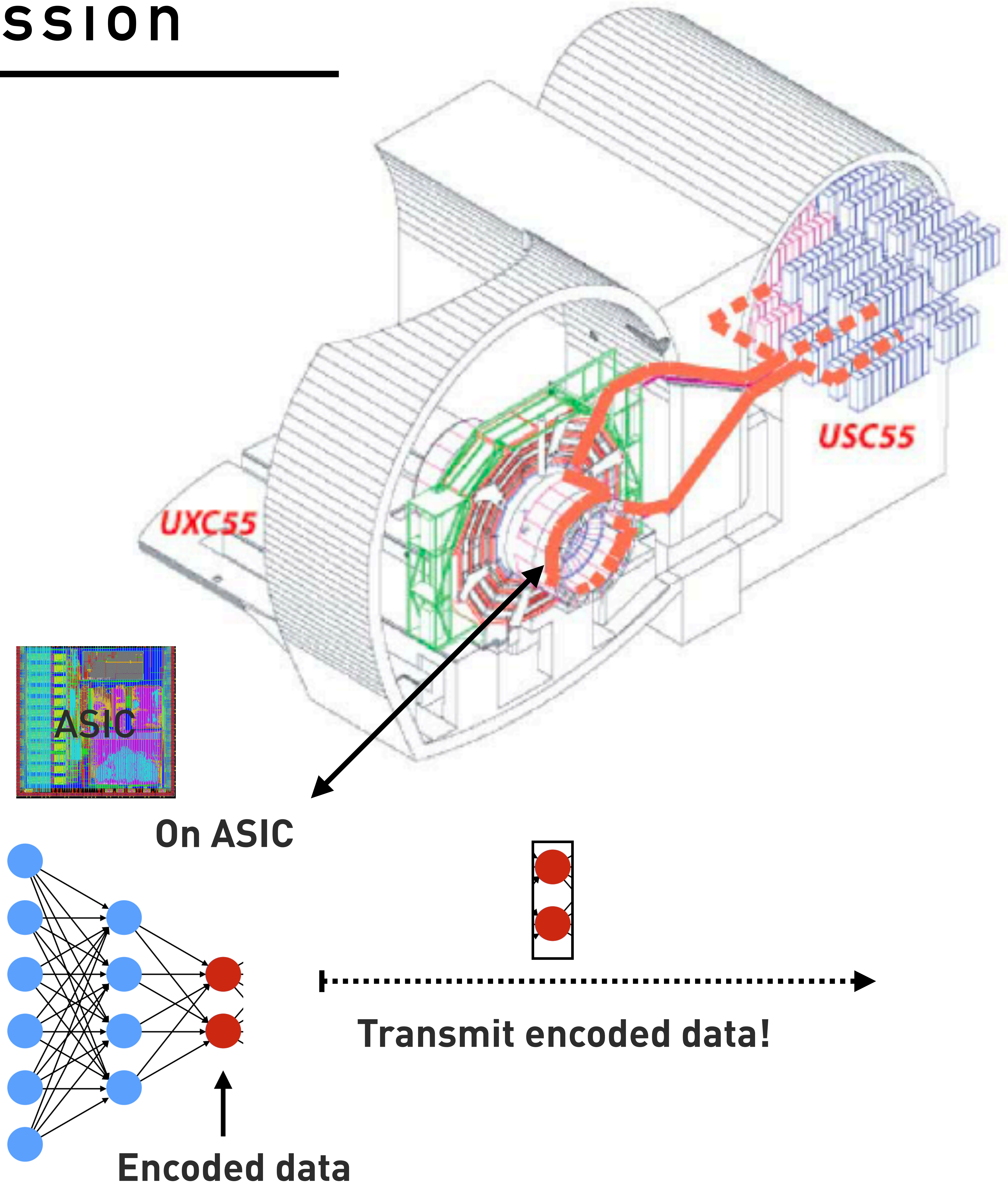
ML for compression



Variational Autoencoder

ML for compression





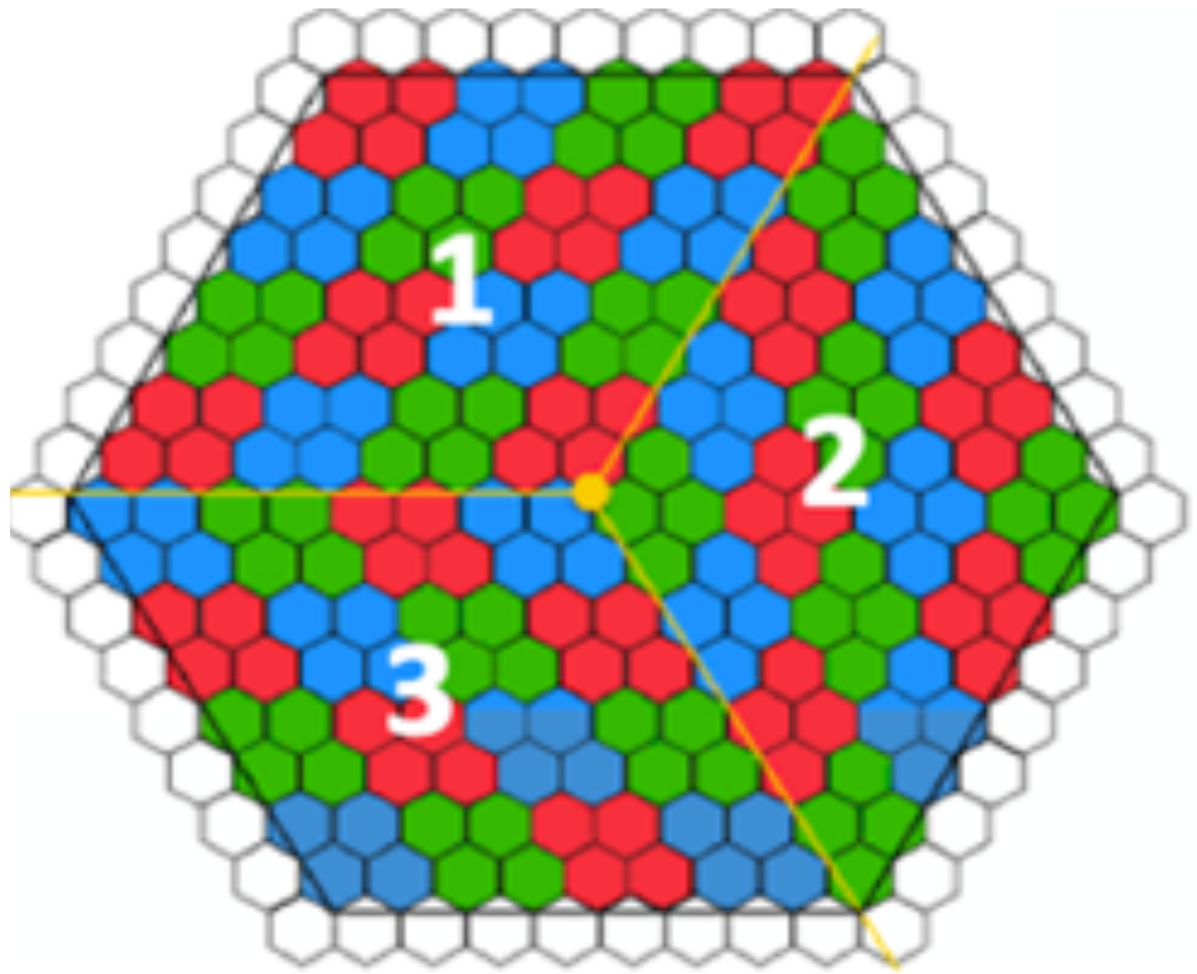
On ASIC

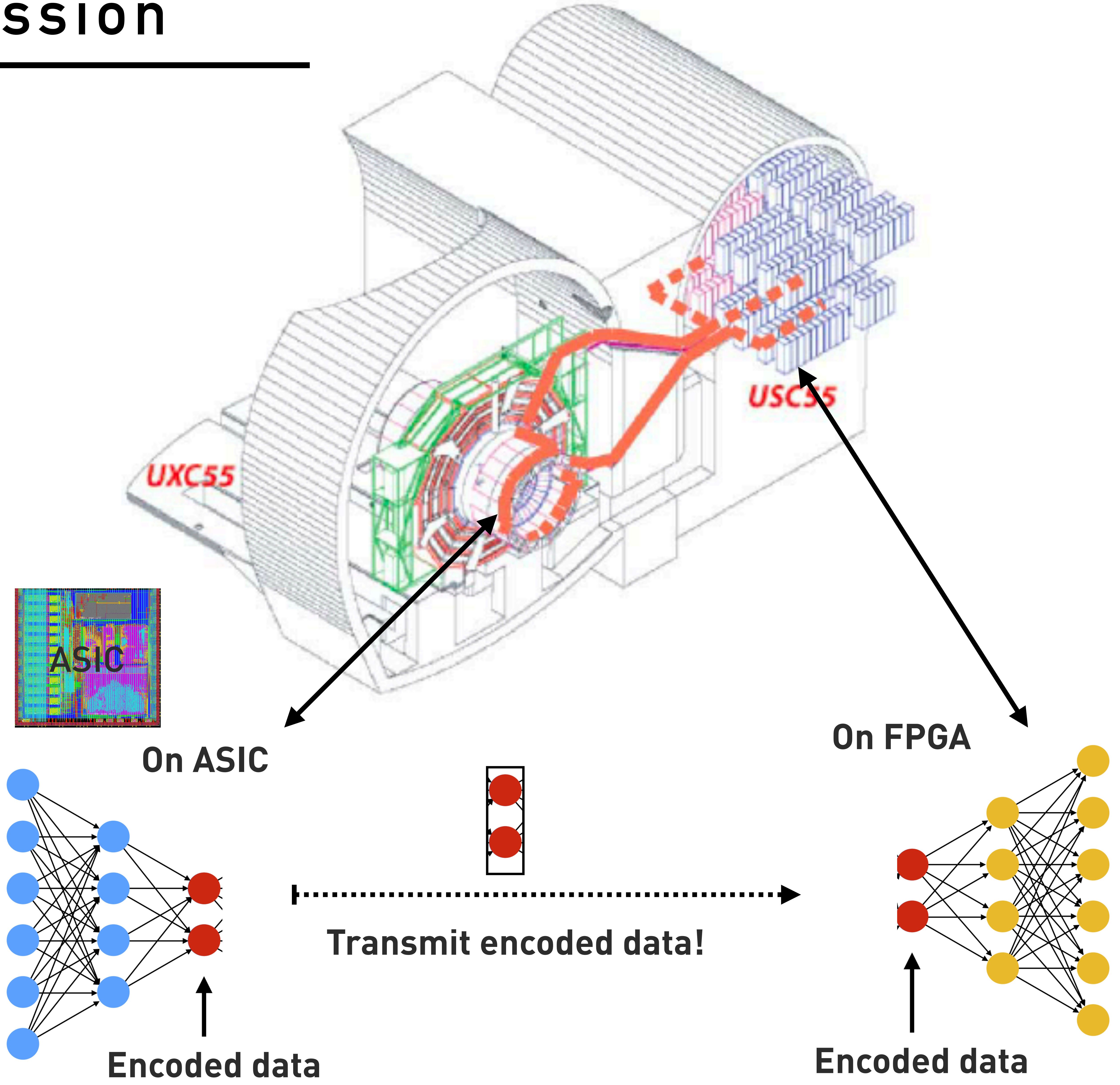
Encoded data

Transmit encoded data!

ECON-T, D. Noonan

ML for compression





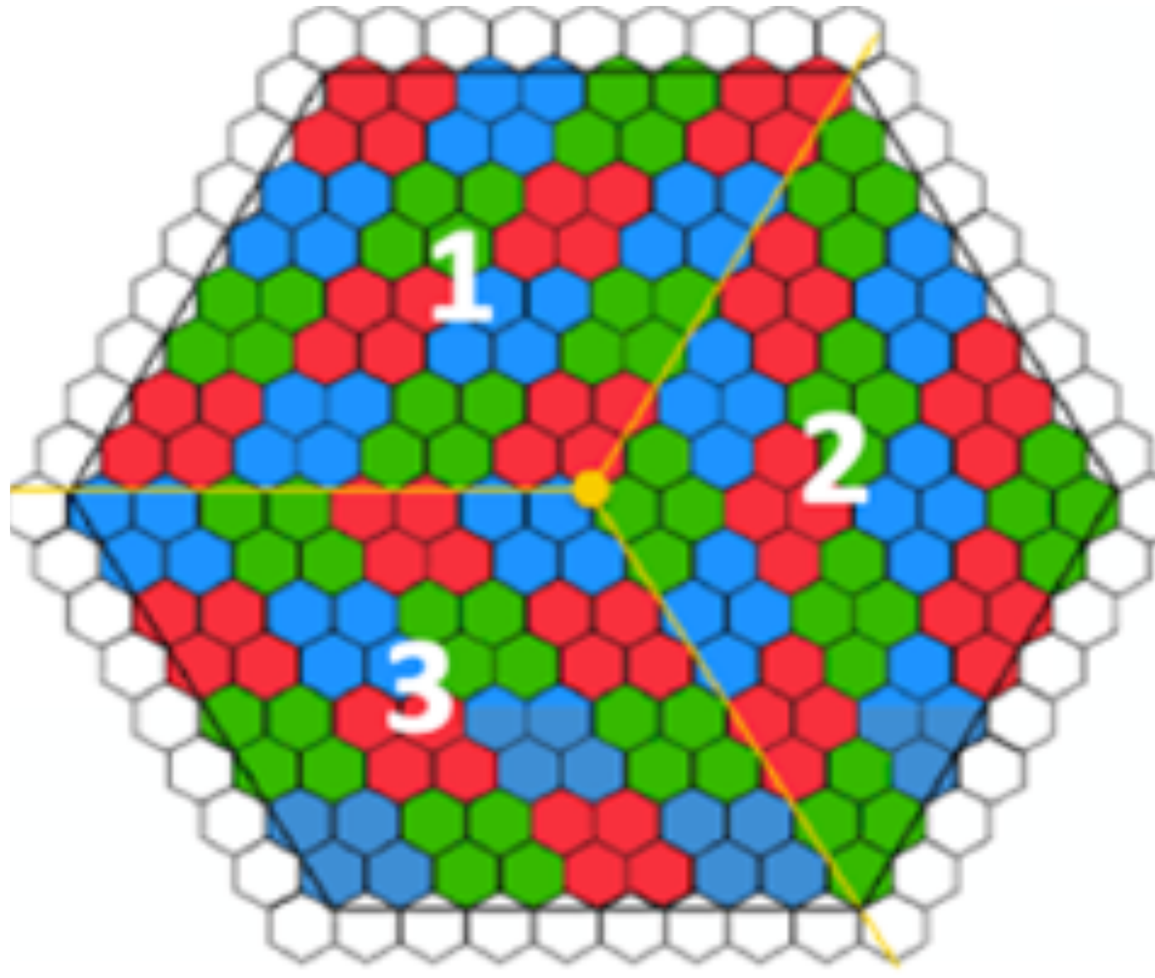
On ASIC

Encoded data

Transmit encoded data!

On FPGA

Encoded data

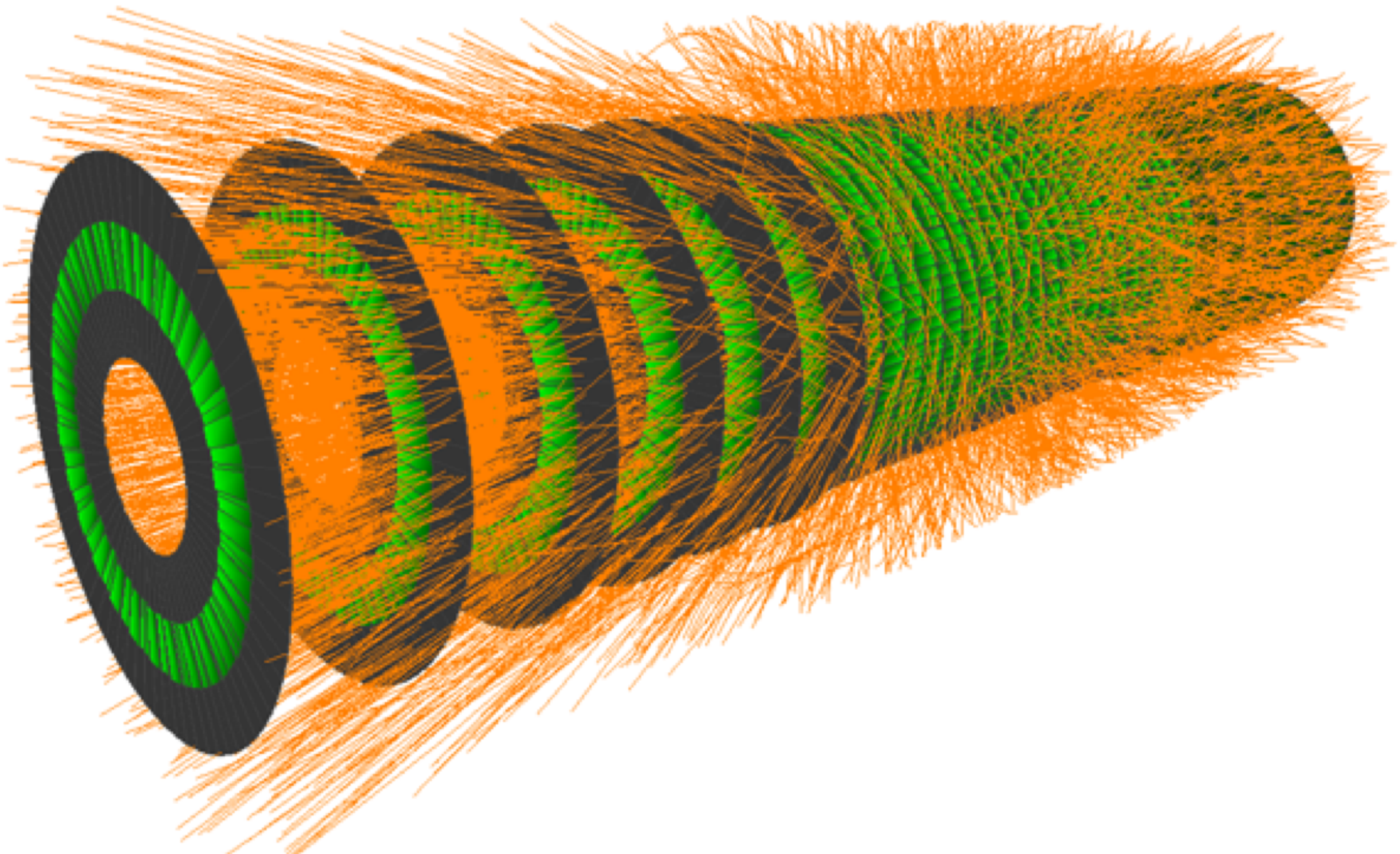


Geneva
Geneva Lake
LHCb
AT
CMS
LHC
XILINX
VIRTEX-5
XC5VLX30

12 μs to make accurate decisions!

DOI:10.3389/fdata.2022.828666

ML for track reconstruction



- O(1000) hits
- O(100) tracks
- 5 μs latency!

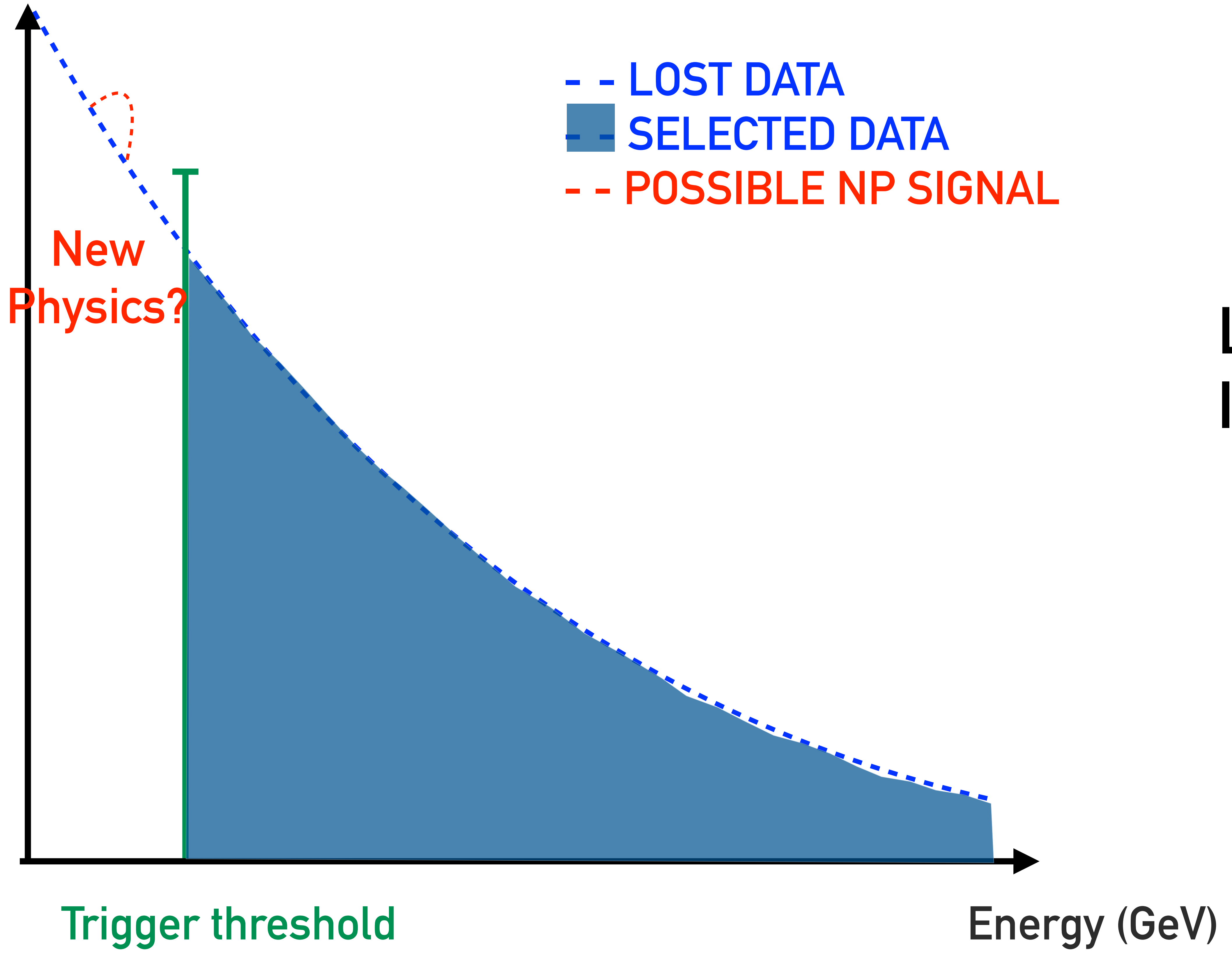
Graph Neural Networks for fast reconstruction!

Design	(n_{nodes}, n_{edges})	RF	Precision	Latency [cycles]	II [cycles]	DSP [%]	LUT [%]	FF [%]	BRAM [%]
Throughput-opt.	(28, 56)	1	`ap_fixed<14,7>`	59	1	99.9	66.0	11.7	0.7

3 μs, but all the FPGA resources are used!

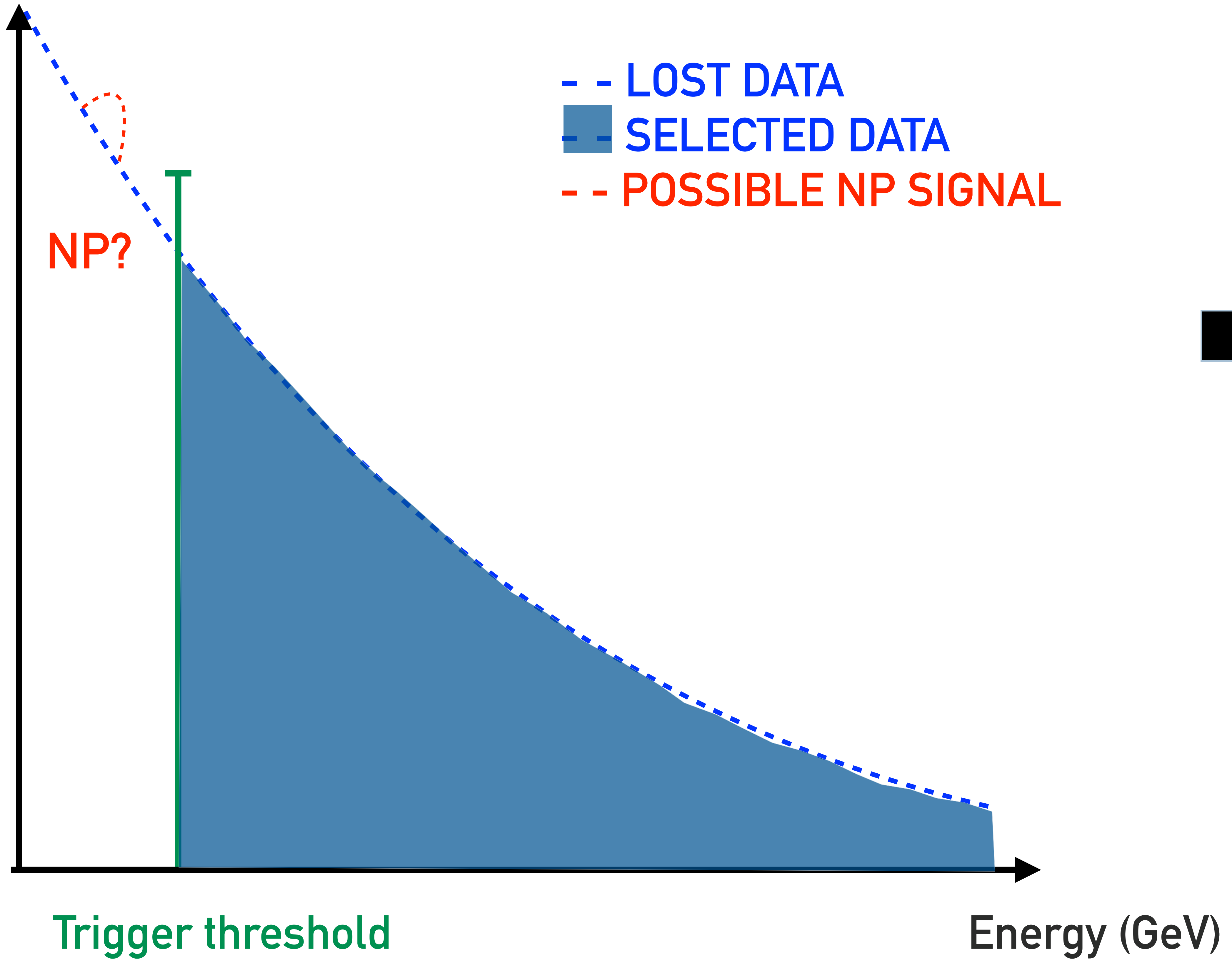
How to get fast and small graph-like NNs on FPGAs? Ongoing work by Patrick Odagiu (IPA)

Limitations of current trigger



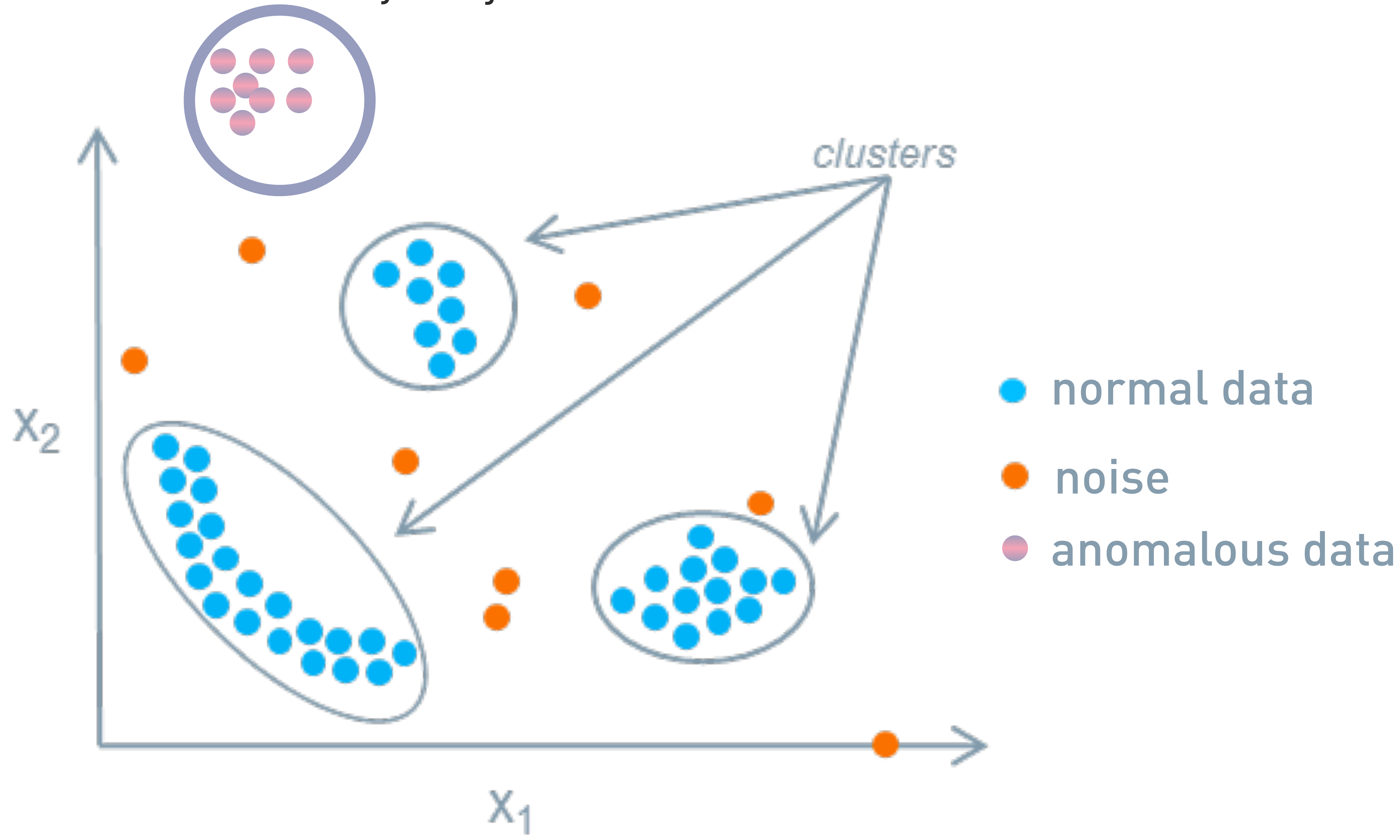
Level-1 rejects >98% of events!
Is there a smarter way to select?

Learning from data



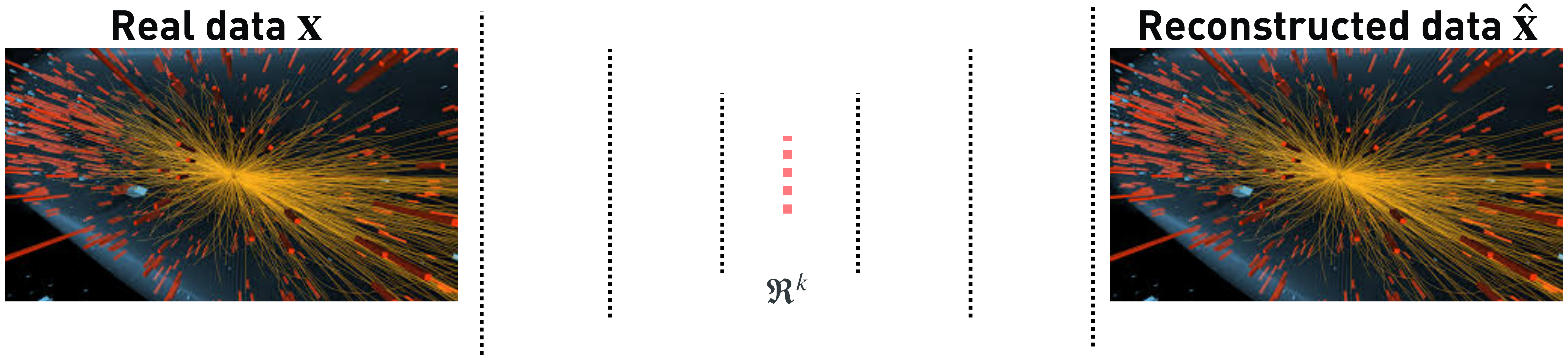
Look at **data** rather than defining signal hypothesis a priori

- Can we “classify” objects/events?

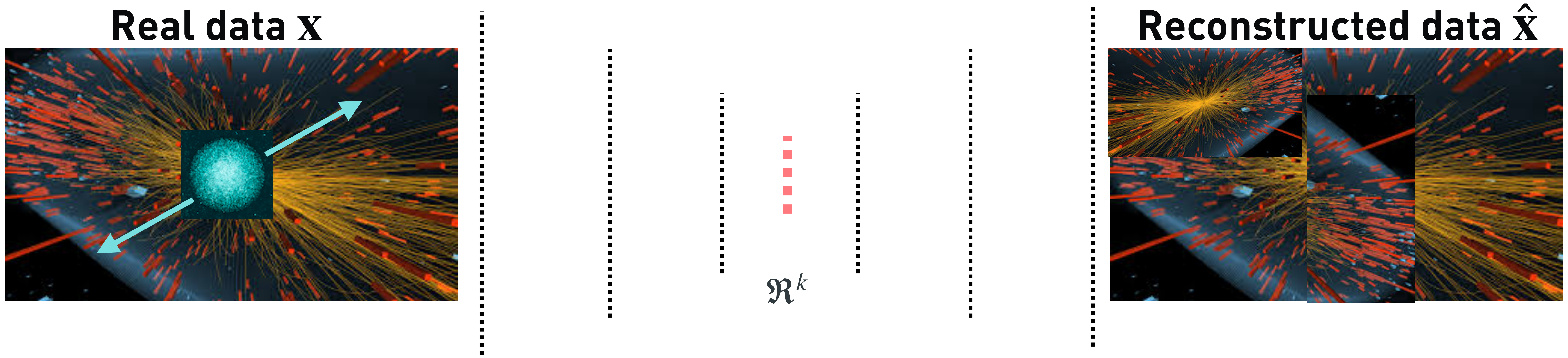


ML for anomaly detection

VAE: Learn from data

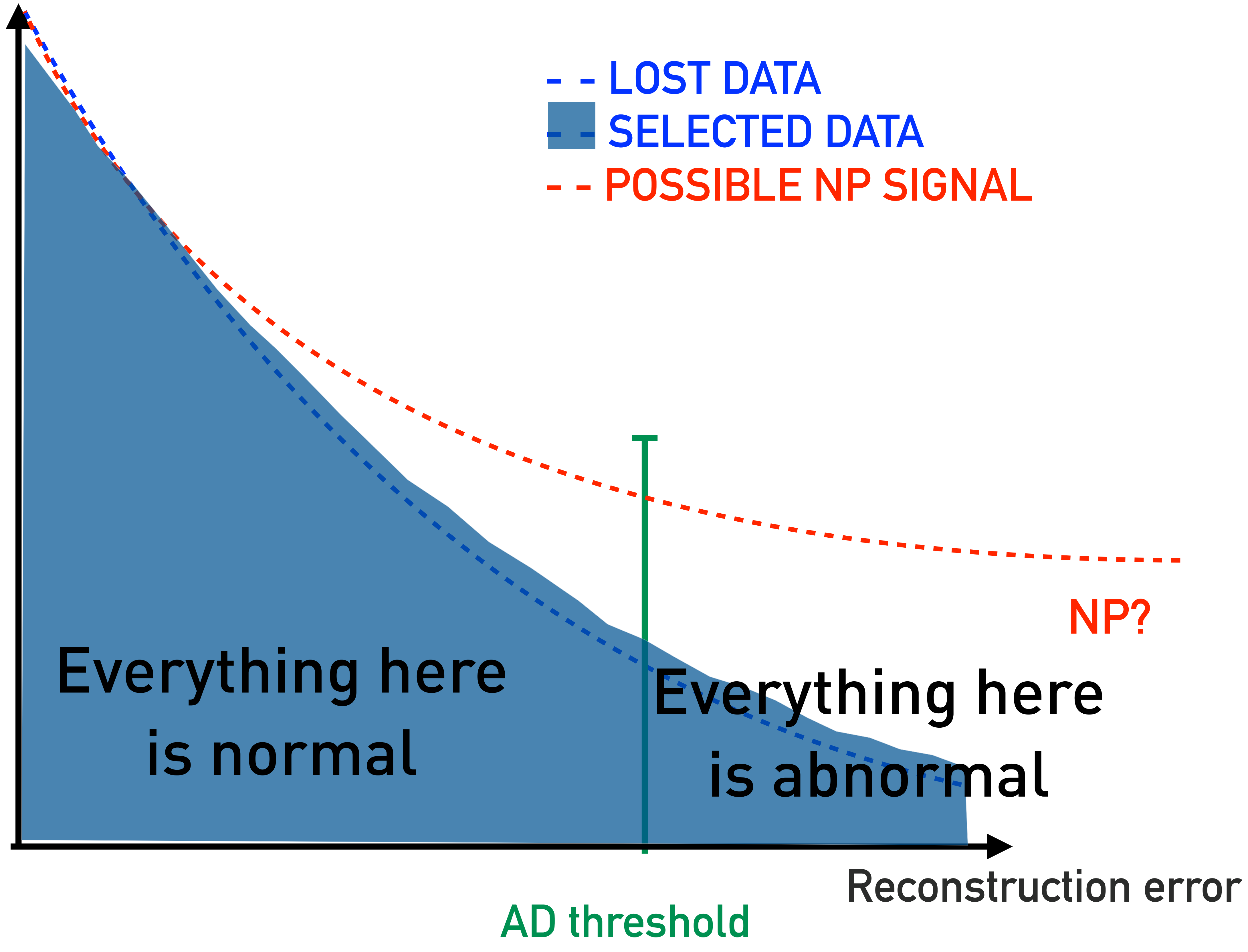


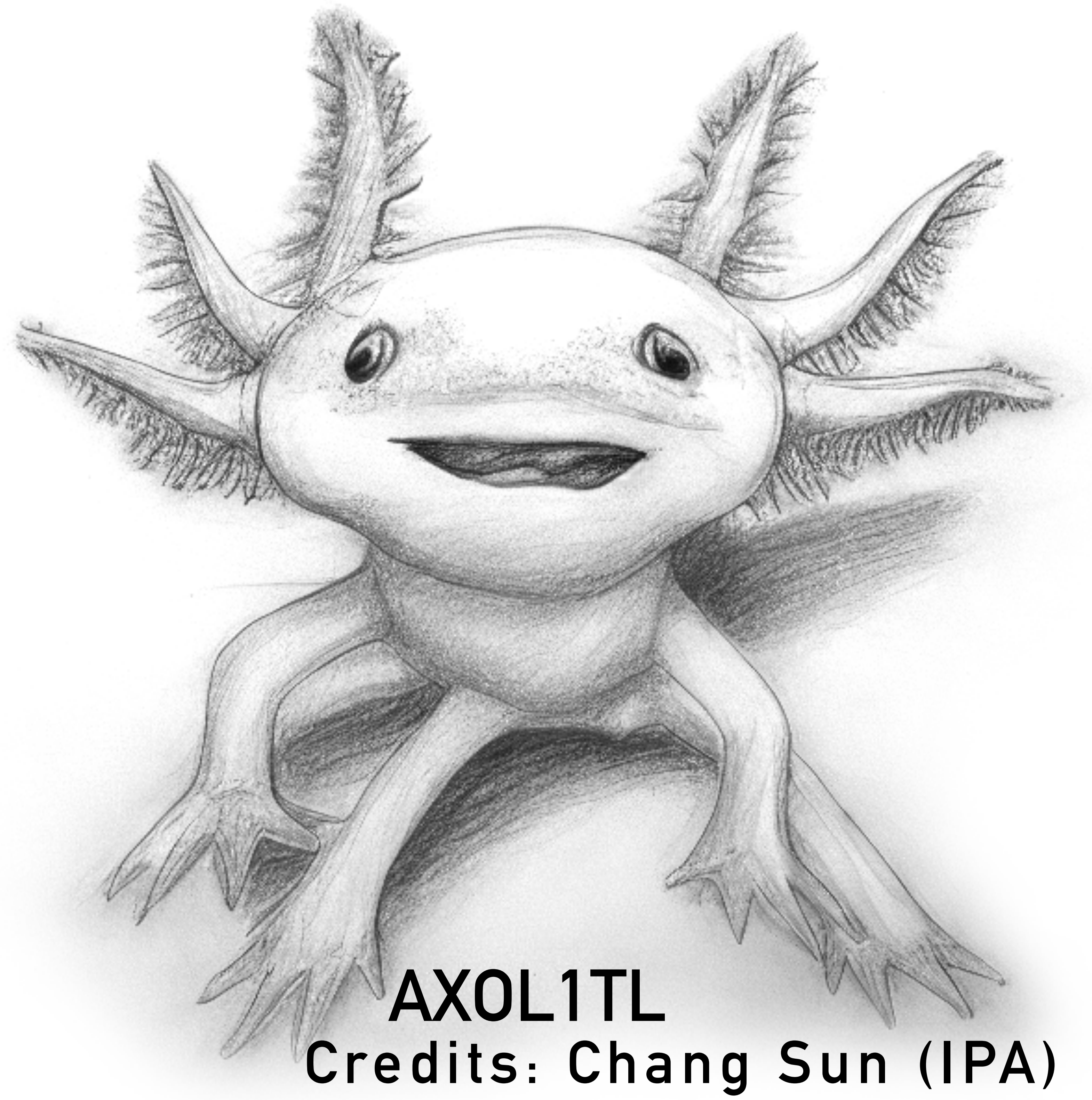
ML for anomaly detection



- Difference $\mathbf{x}$-$\hat{\mathbf{x}}$ defines "degree of abnormality"

ML for anomaly detection





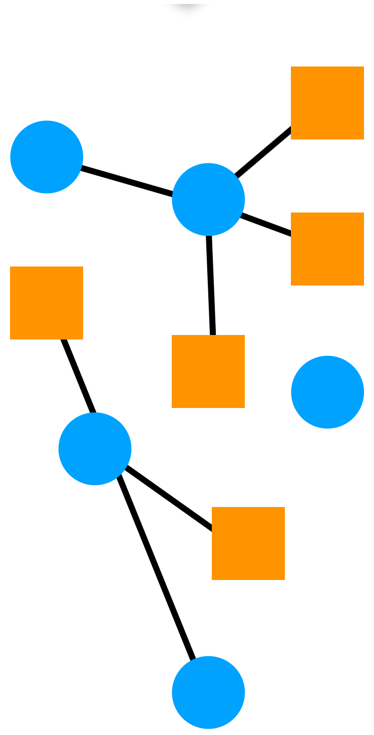
Credits: Chang Sun (IPA)

Select based on degree of abnormality!

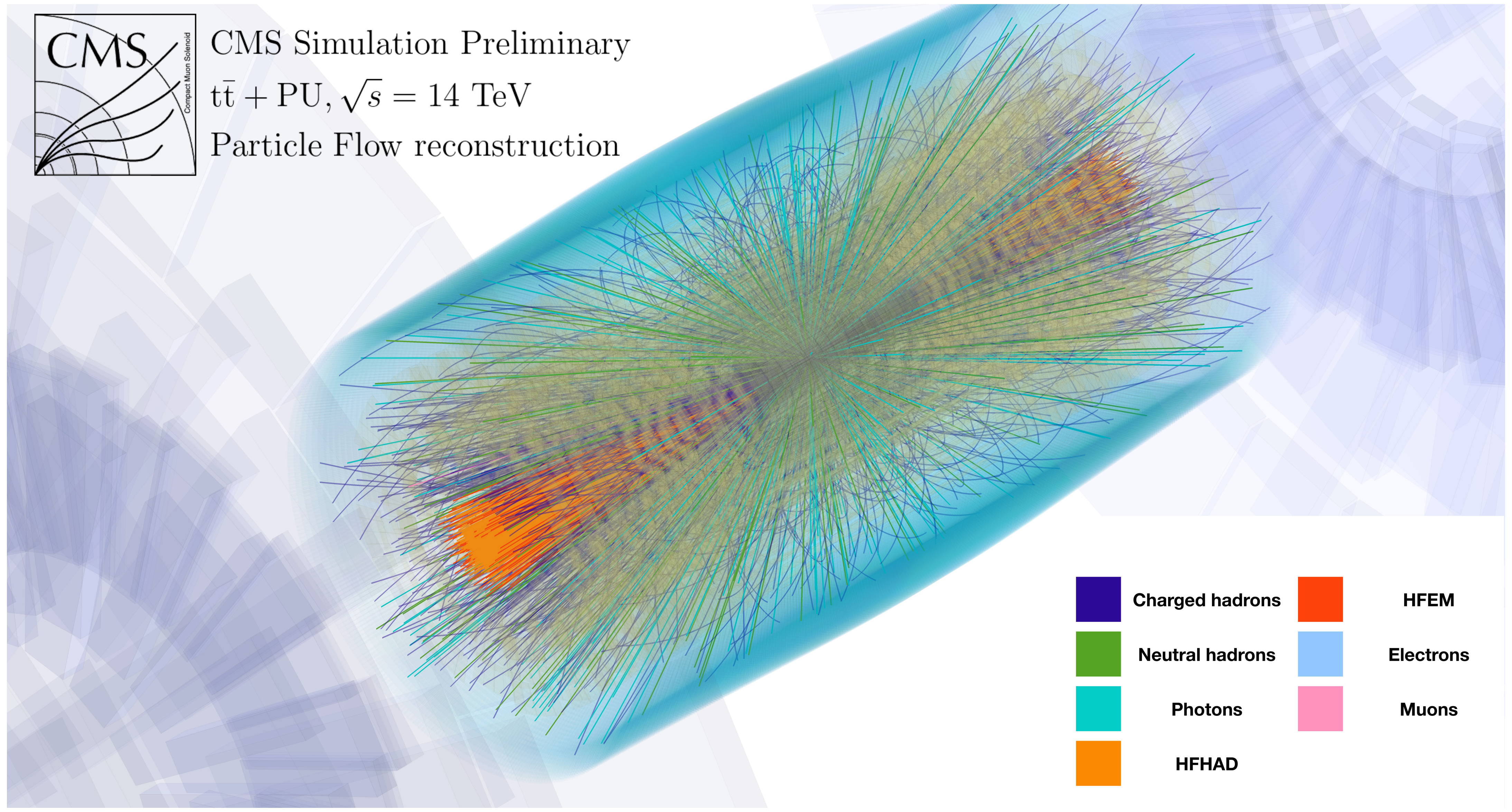
Ongoing work by Chang Sun, P. Odagiu (IPA)

The dream: ML for everything?

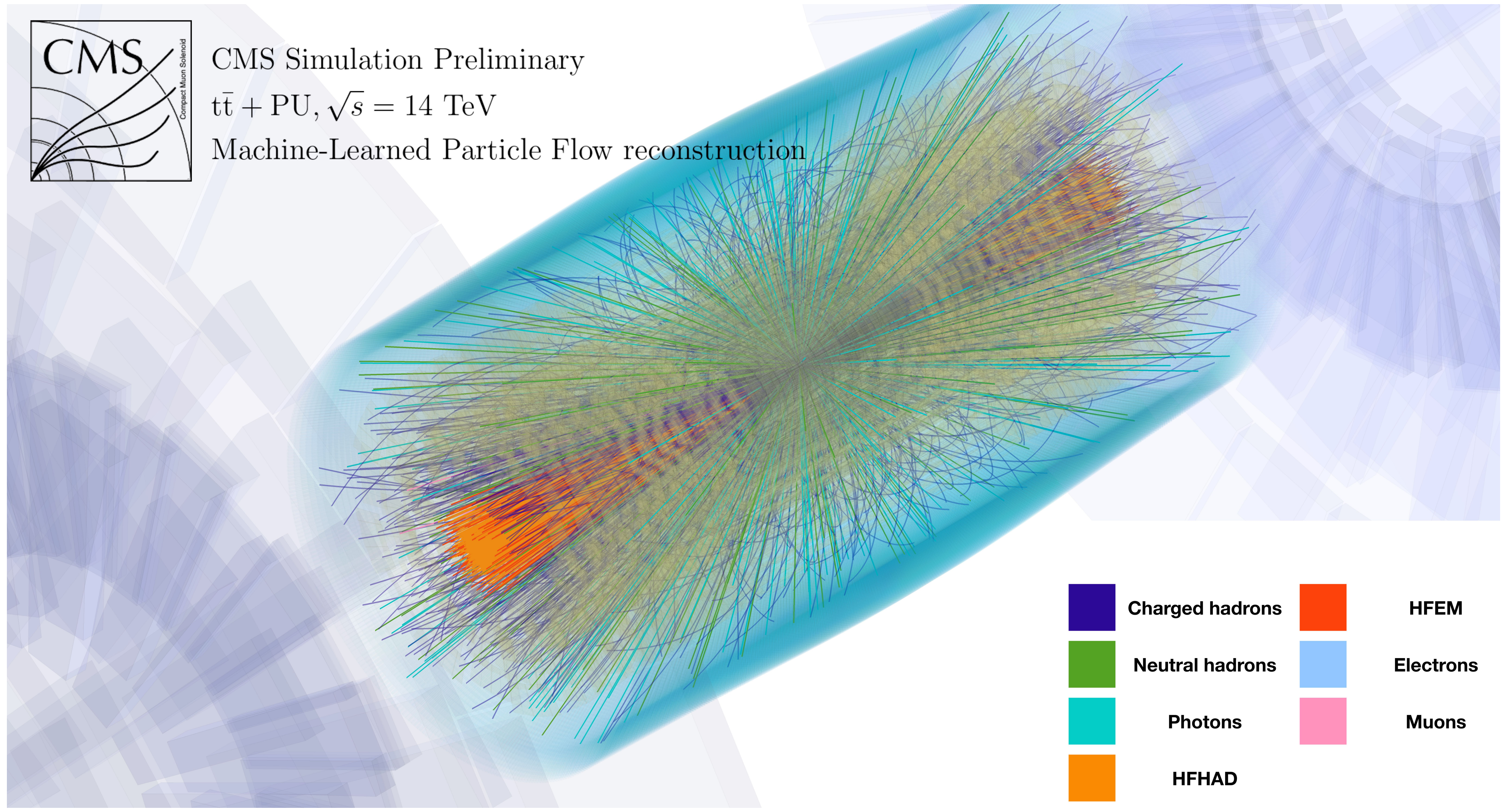
Graph Deep Neural Networks:
End-to-end ML for reconstructing the full collision?
Faster and better than classical reconstruction algorithms



Classical Particle Flow


CMS
CMS Simulation Preliminary
$t\bar{t}$ + PU, $\sqrt{s} = 14$ TeV
Particle Flow reconstruction
Charged hadrons
Neutral hadrons
Photons
HFHAD
HFEM
Electrons
Muons

Graph Neural Network


CMS
CMS Simulation Preliminary
$t\bar{t}$ + PU, $\sqrt{s} = 14$ TeV
Machine-Learned Particle Flow reconstruction
Charged hadrons
Neutral hadrons
Photons
HFHAD
HFEM
Electrons
Muons

arXiv:2101.08578

F. Capel et al.

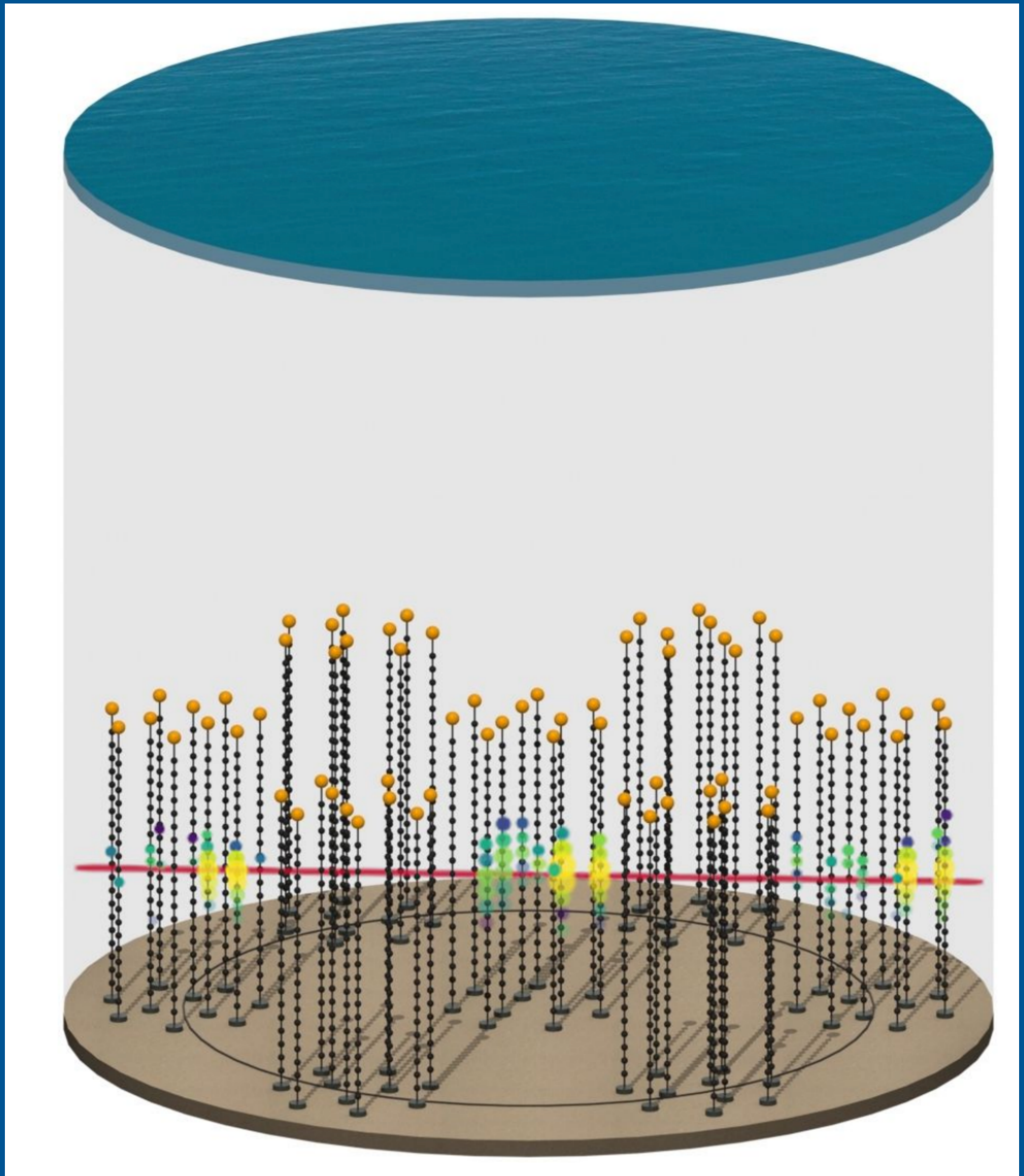
Triggering in other experiments

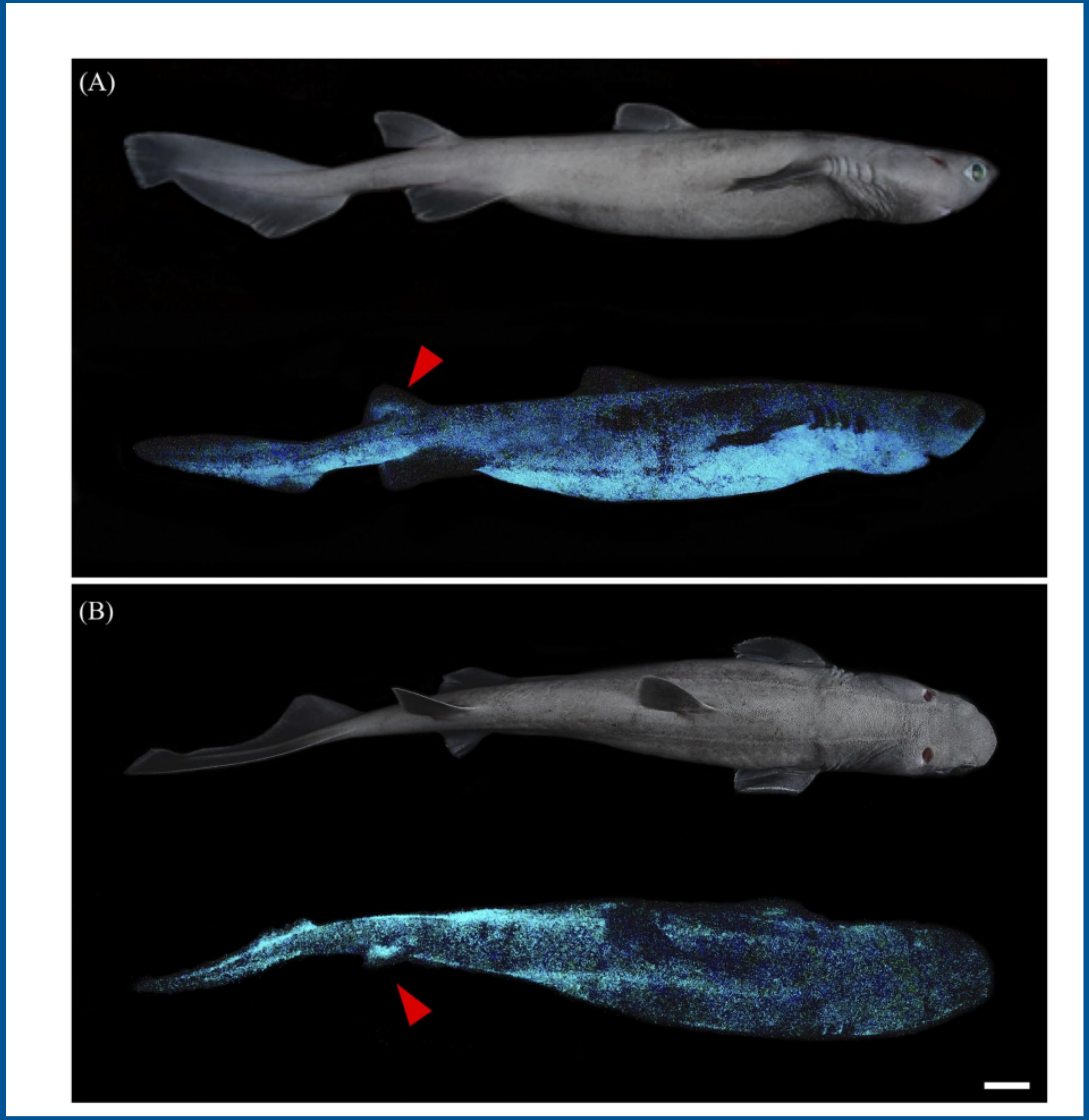
CERNCOURIER

November/December 2022 cerncourier.com Reporting on international high-energy physics

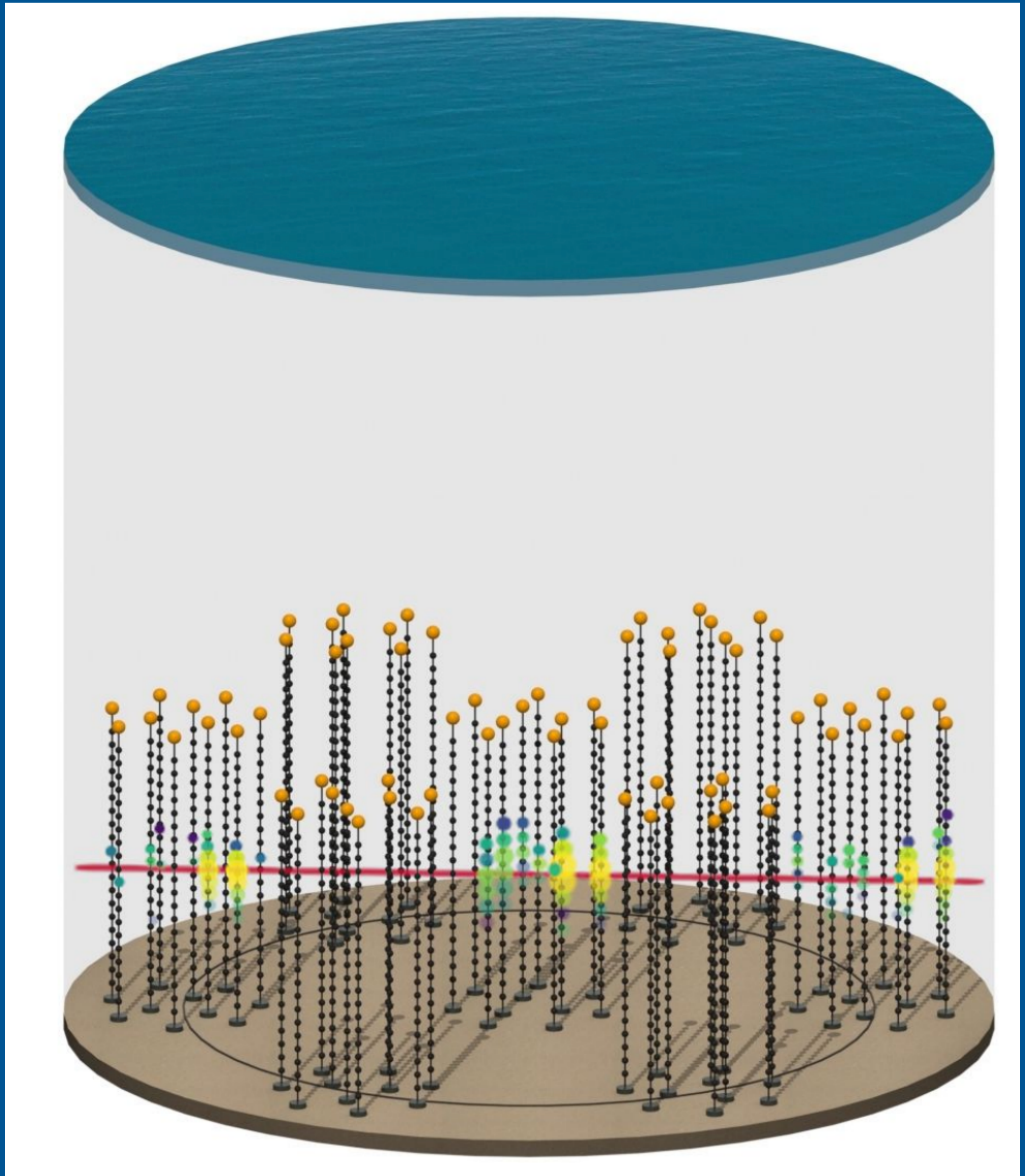
NEUTRINOS
OUT OF
THE BLUE

HL-LHC civil engineering complete
Snowmass: the full report
Taking plasma accelerators to market

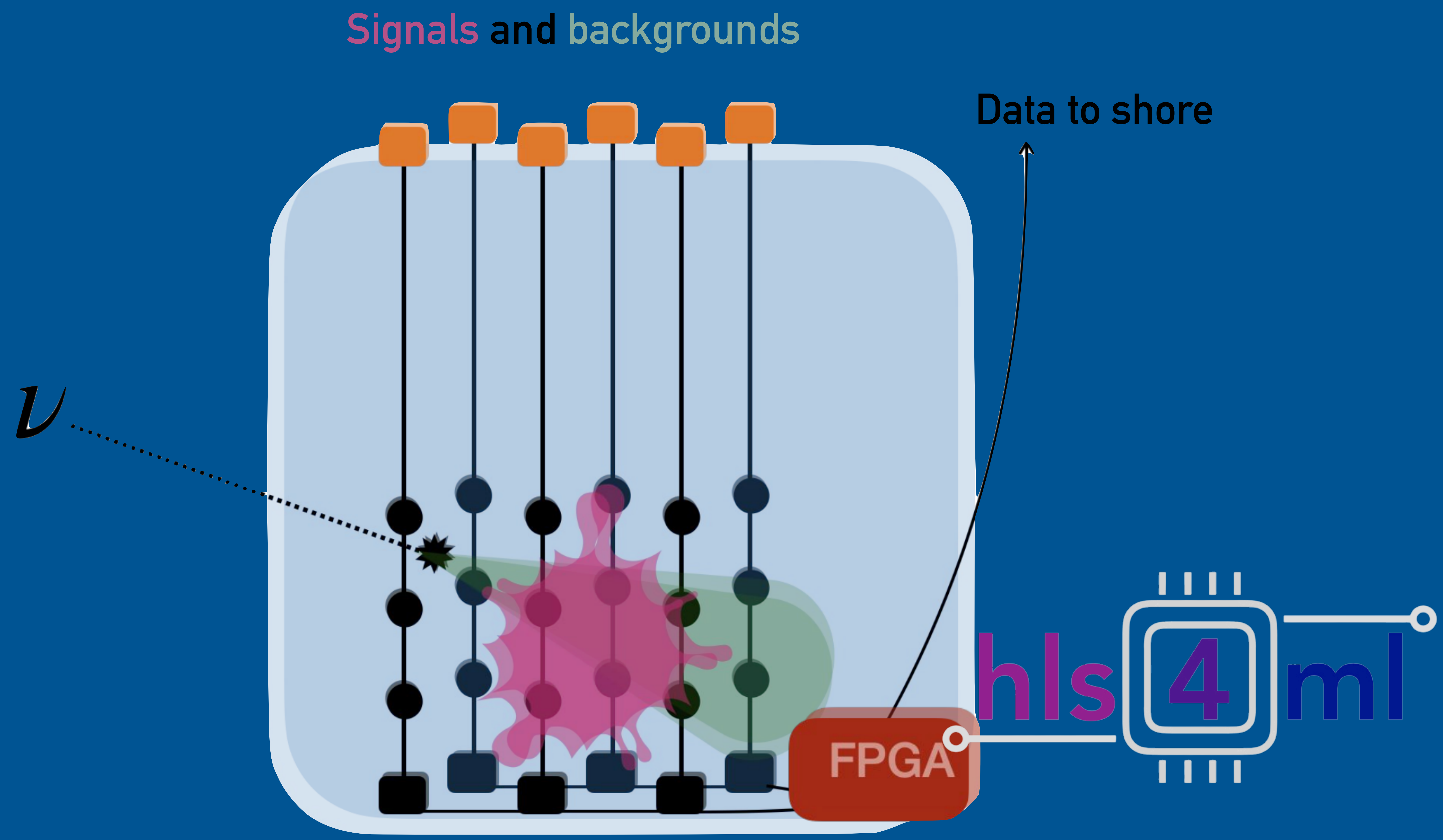




Bioluminescence bursts up to few MHz!



F. Capel et al.



Signals and backgrounds

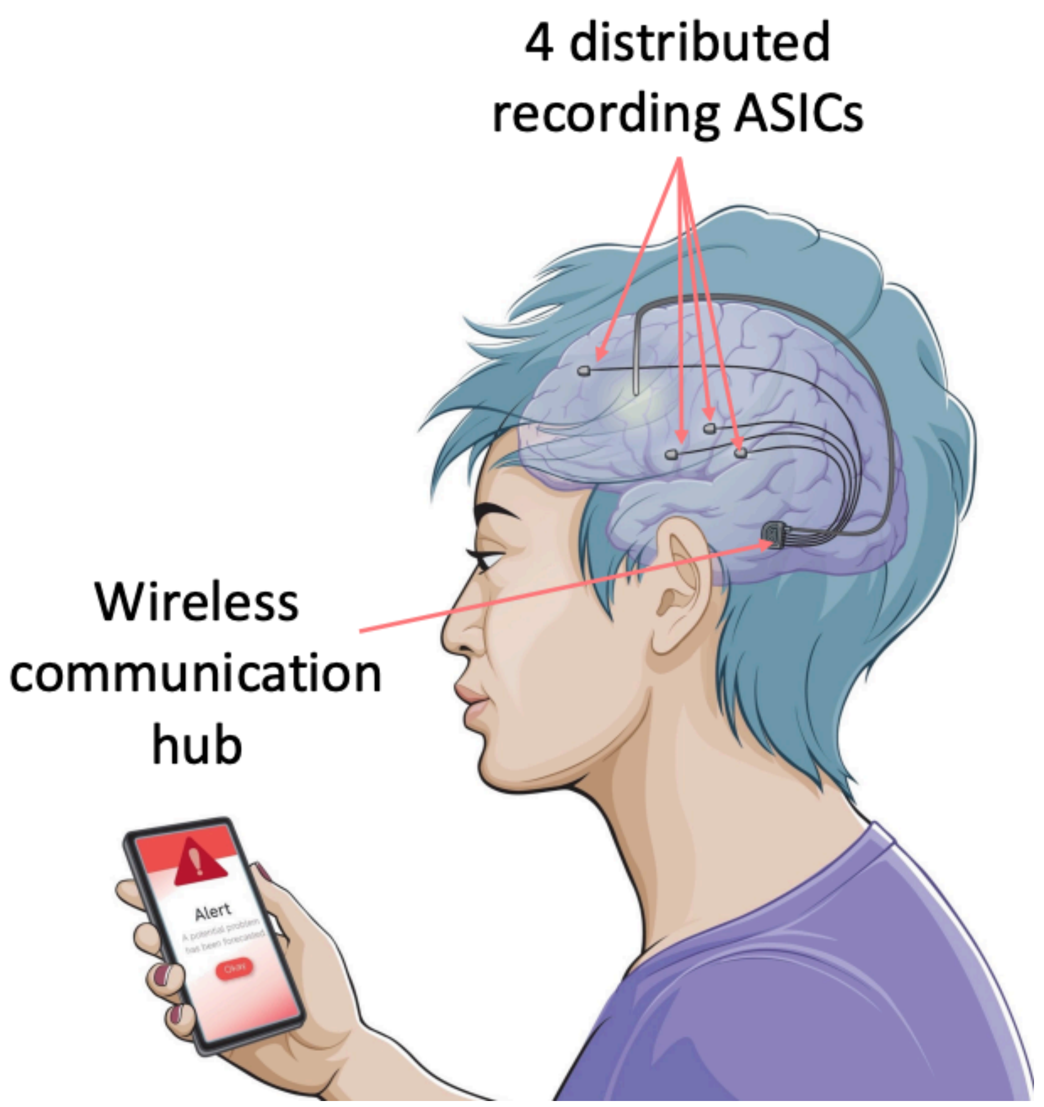
Data to shore

...and outside of HEP

Semantic segmentation for autonomous vehicles

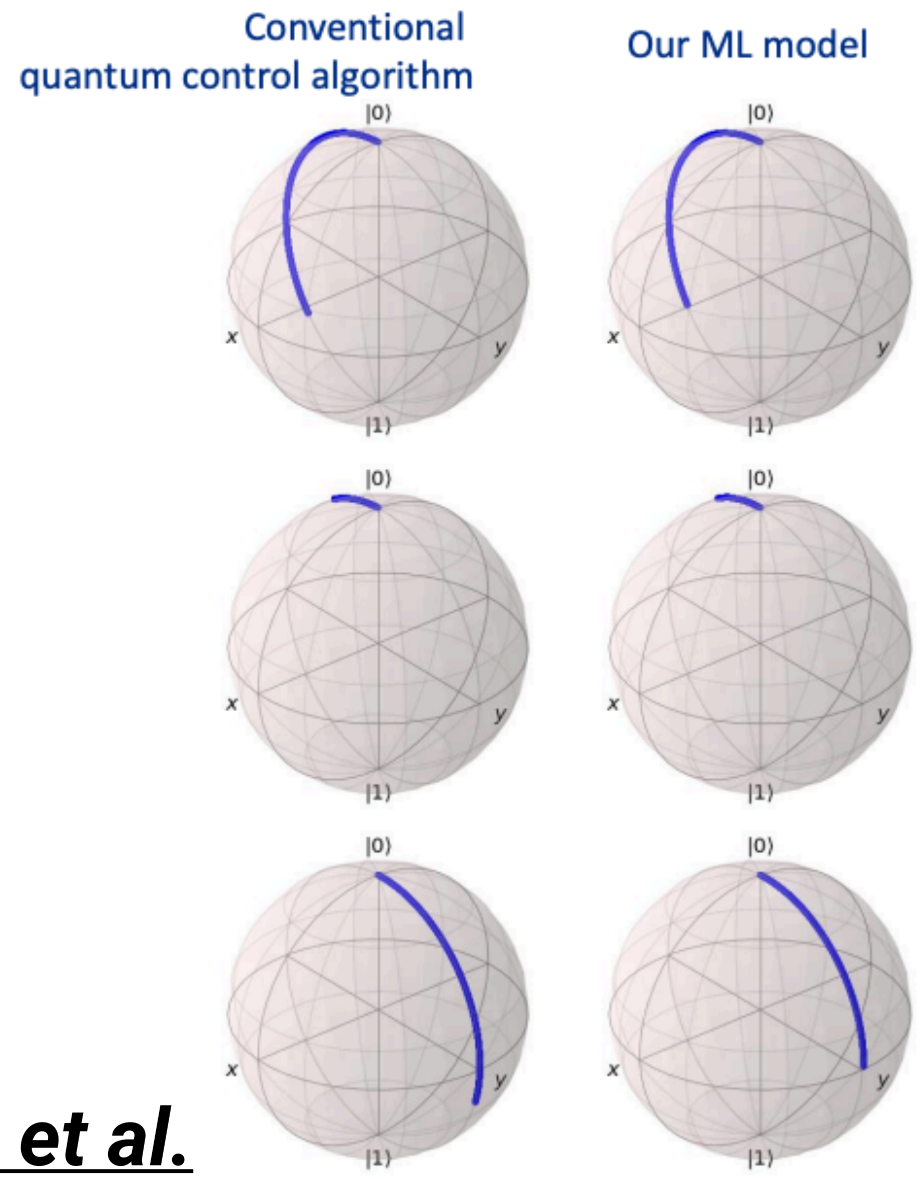
N. Ghielmetti et al.

Seizure Predicting Brain Implant



W. Lemaire et al.

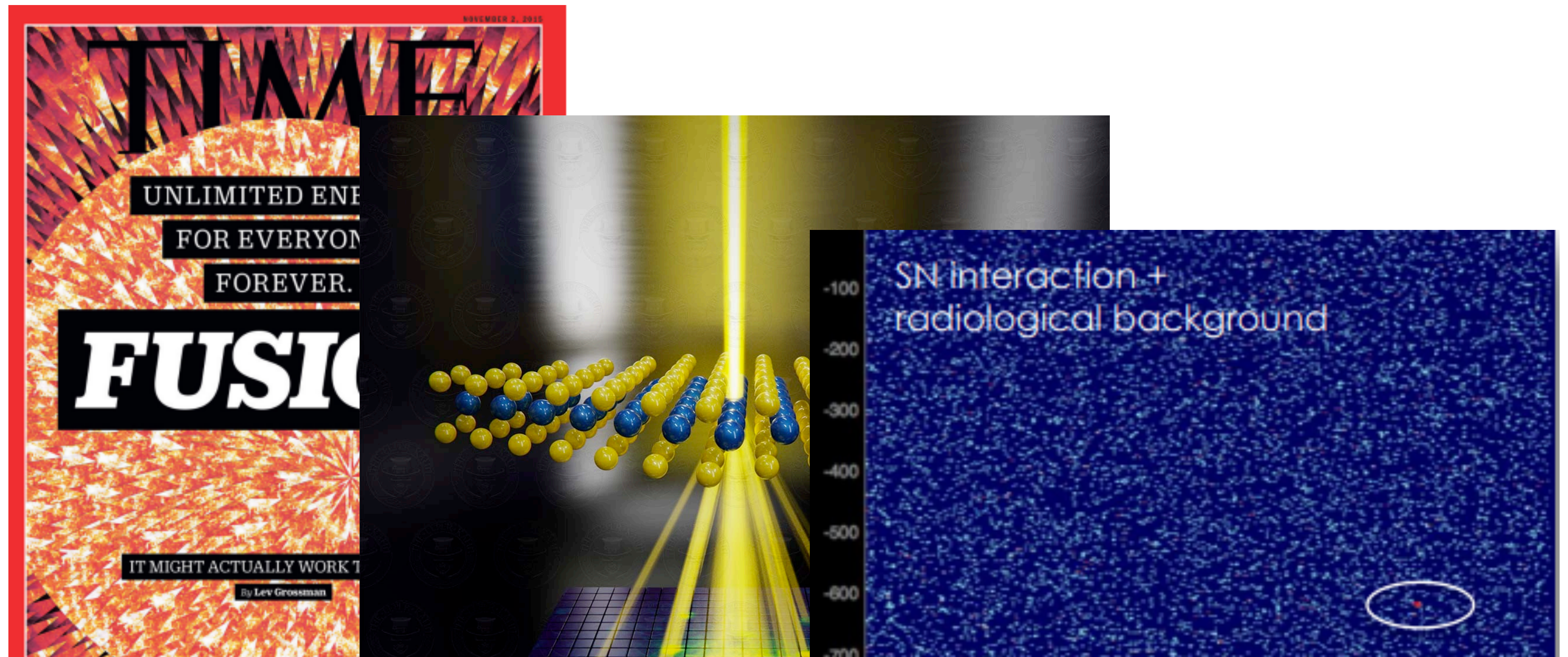
NN accelerator for quantum control

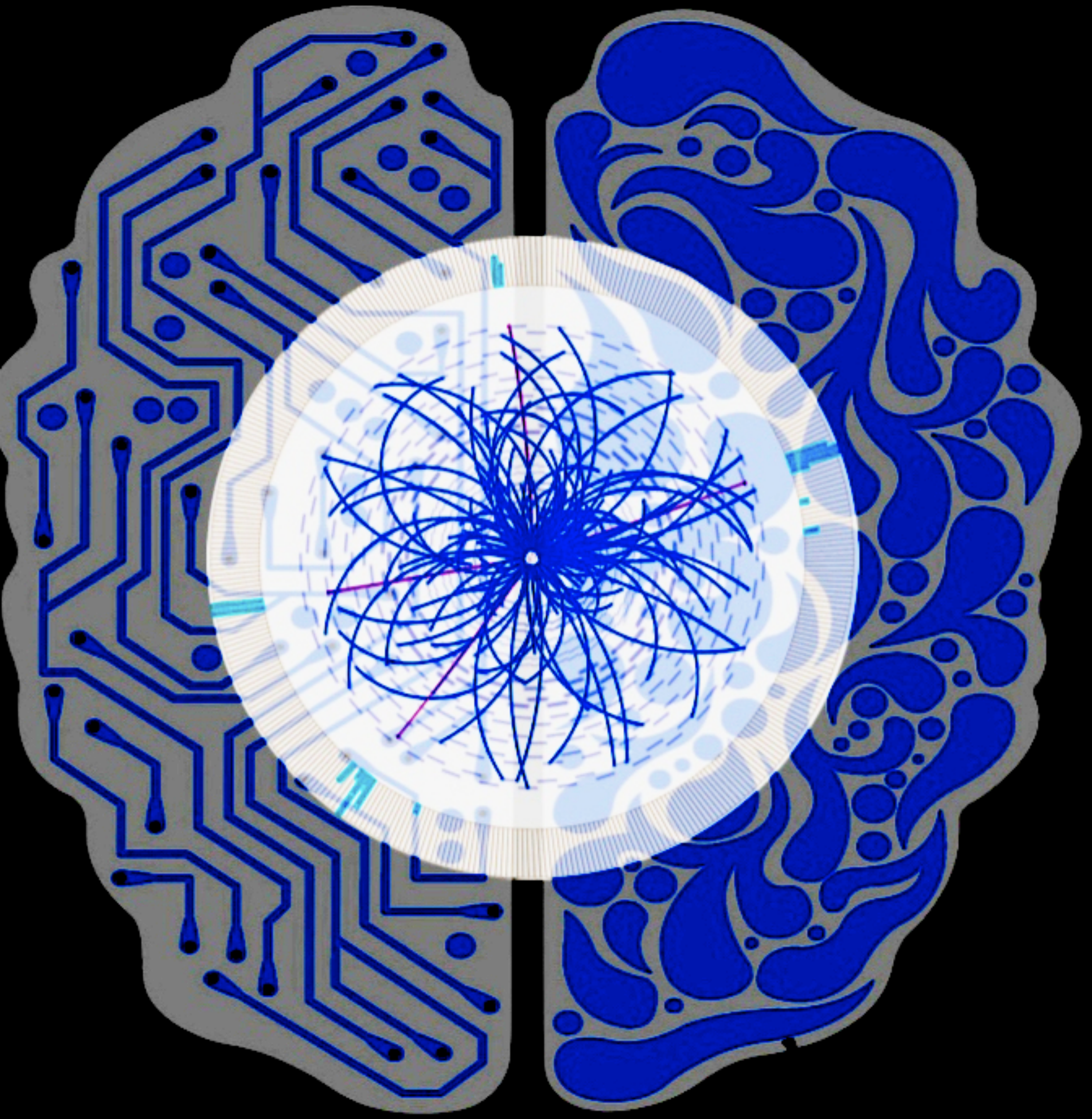


D Xu et al.

Other examples

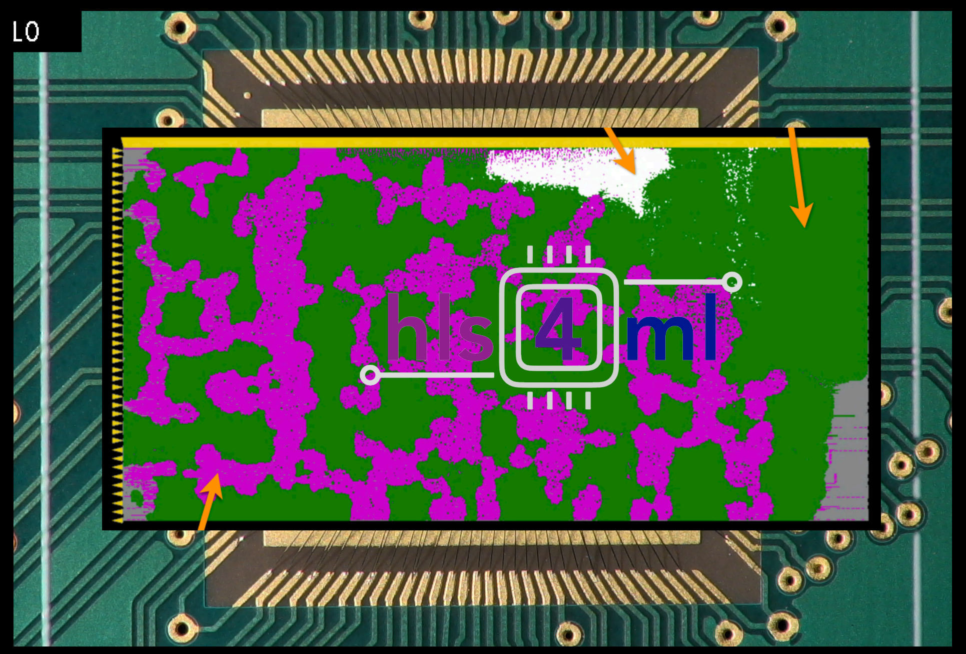
- ***For fusion science phase/mode monitoring***
- ***Crystal structure detection***
- ***Triggering in DUNE***
- ***Accelerator control***
- ***Magnet Quench Detection***
- ***MLPerf tinyML benchmarking***
- ***Food contamination detection***
- etc....





Conifer

hls4ml



Join the community:

fastmachinelearning.org

Sign up to the hls-fml group

...or come have a chat with

Patrick, Chang, Andrew and me in HPK E 29!

Extra material

Training FLOPs Scaling for SOTA CV, NLP, and Speech Models
Transformer: 750x / 2 yrs
CV/NLP/Speech: 15x / 2 yrs
Moore's Law: 2x / 2 yrs
1e+09
1e+08
1e+07
1e+06
1e+05
1e+04
1e+03
1e+02
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
YEAR
AlexNet
Seq2Seq
VGG
InceptionV3
ResNet
Xception
ResNext
DenseNet
Transformer
GPT-1
ELMo
BERT
XLNet
Megatron LM
Microsoft T-NLG
MoCo ResNet50
Wav2Vec 2.0
GPT-3

AI & Memory Wall