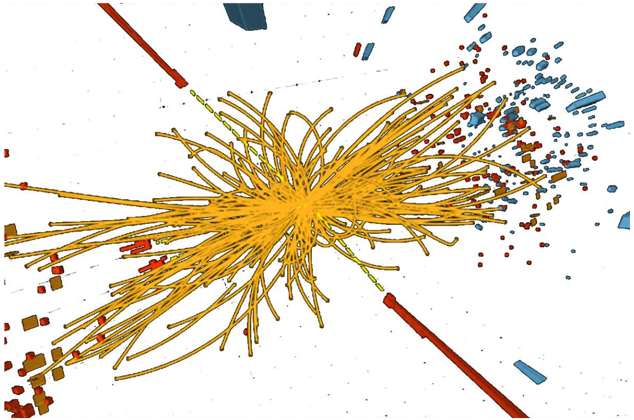
Pruning and resizing deep neural networks for FPGA implementation in trigger systems at collider experiments

D. Mascione, M. Cristoforetti, A. Di Luca, F. M. Follega, R. Iuppa, A. Saccardo

University of Trento, Fondazione Bruno Kessler, INFN TIFPA

ACAT 2022, Bari 27/10/2022

A lot of data



Collider experiments produce a **huge amount of data**.

At the Large Hadron Collider we have

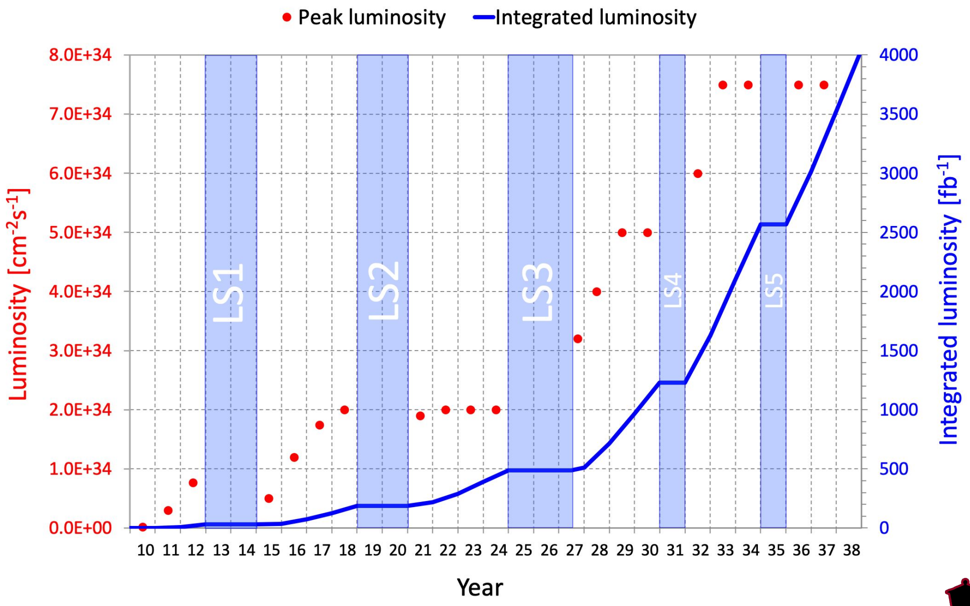
- one collision every 25 ns (= **40 Million collisions/sec**)
- **thousands of particles** emerging from each collision
- **1 MB of data** recordered at each collision by big detectors

ACAT 2022

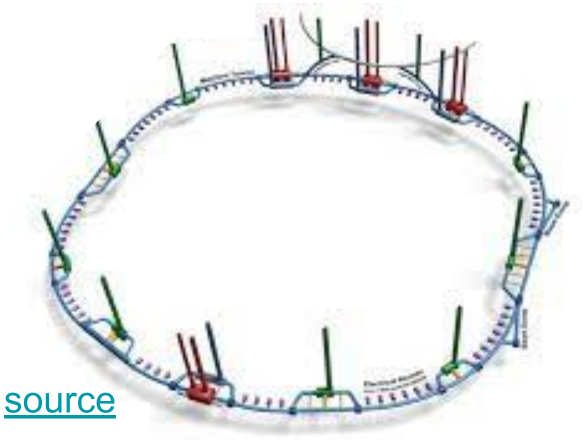
BARI

Increasing data at future colliders

The **HL-LHC** will produce more than 250 fb^{-1} of data per year and will be capable of collecting **up to 4000 fb^{-1}** (1 fb^{-1} ~ 100 million million collisions). →



source



source

← At the **FCC-hh** huge amounts of data will be produced (**O(TBytes/s) expected**).

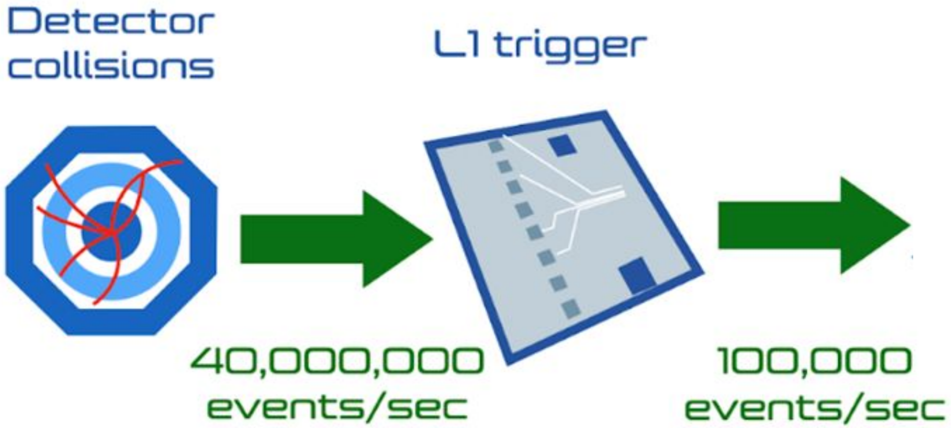
The trigger system at the LHC

Not all data produced at the LHC are stored: they are first filtered with a trigger chain

source

The trigger system at the LHC

Not all data produced at the LHC are stored: they are first filtered with a trigger chain

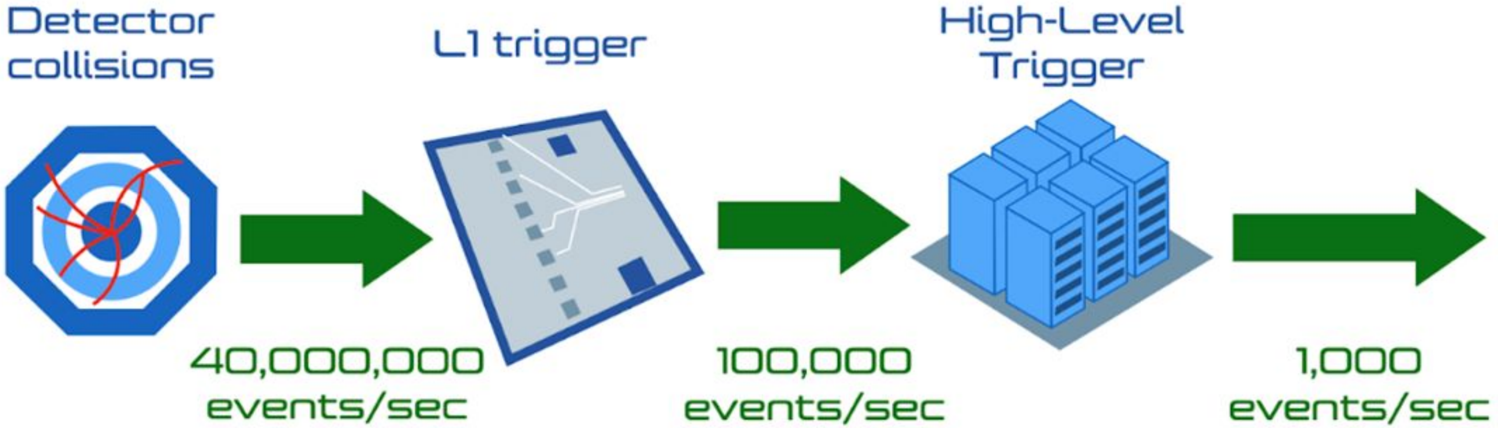


source

ACAT 2022
BARI

The trigger system at the LHC

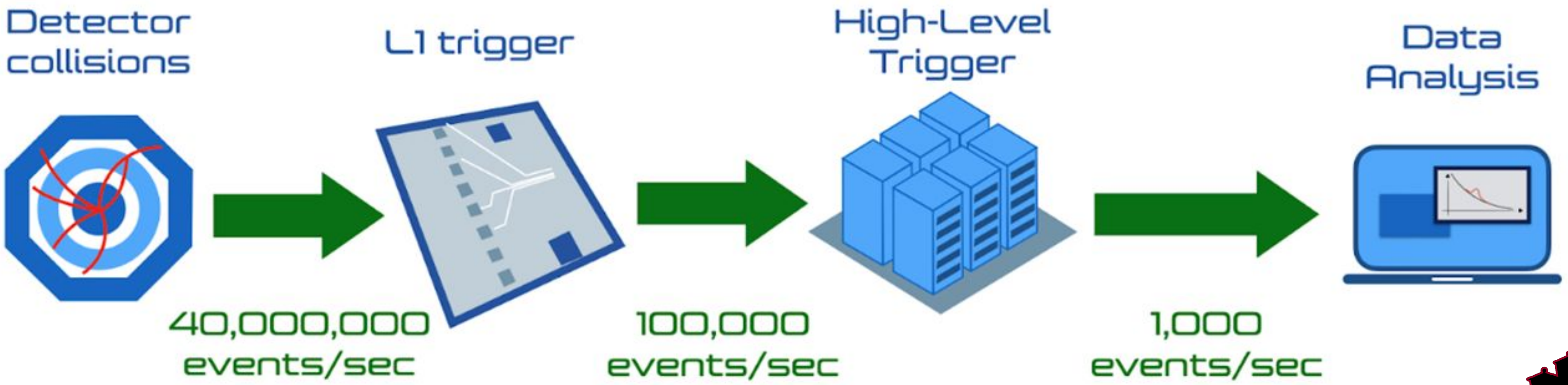
Not all data produced at the LHC are stored: they are first filtered with a trigger chain



source

The trigger system at the LHC

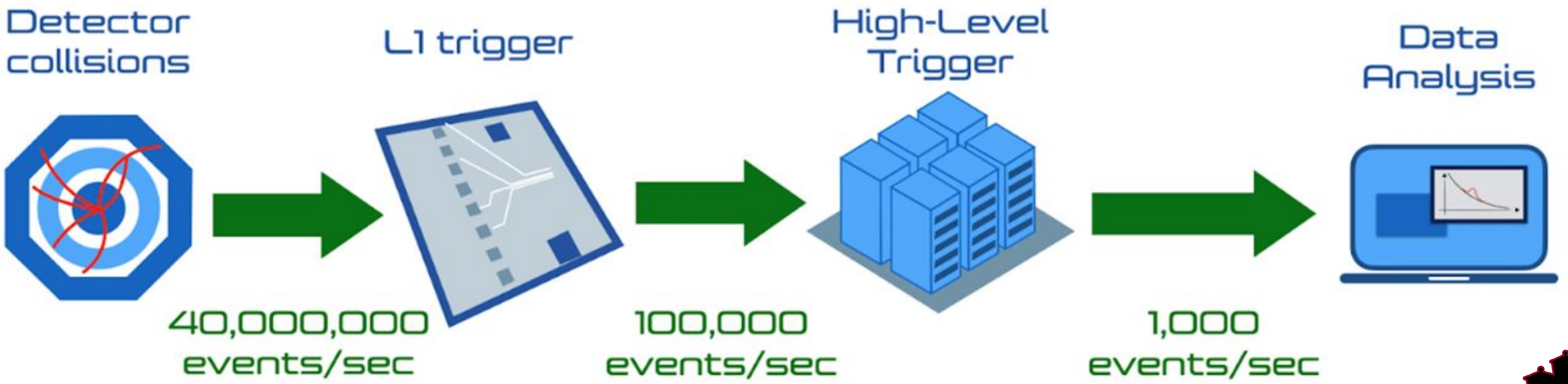
Not all data produced at the LHC are stored: they are first filtered with a trigger chain



source

The trigger system at the LHC

Not all data produced at the LHC are stored: they are first filtered with a trigger chain

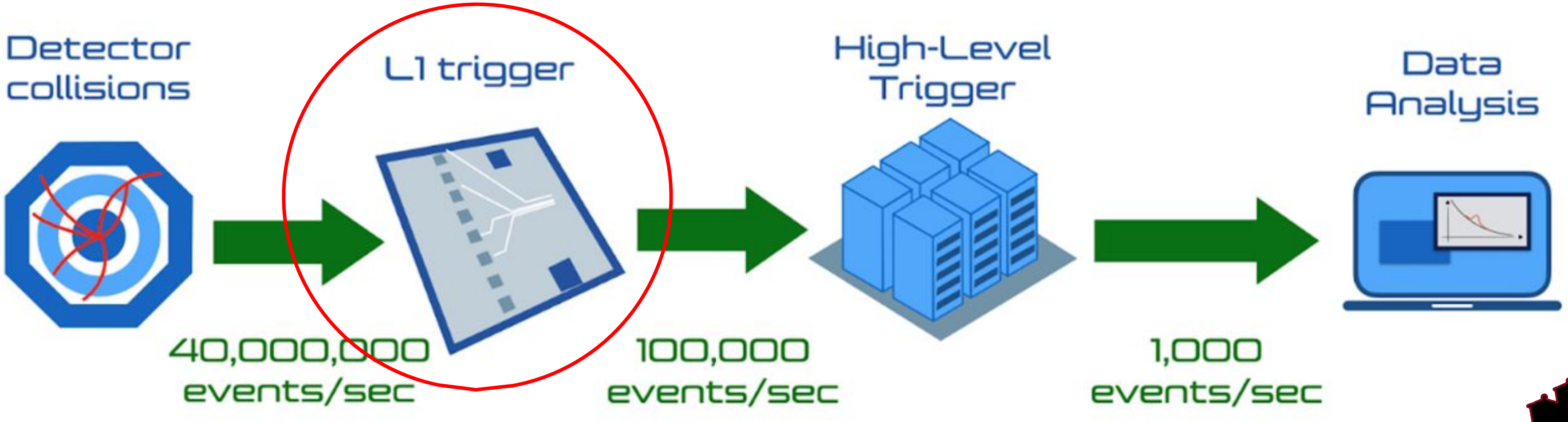


source

Events that are discarded by the trigger are **lost**!

Deep Neural Networks at rescue

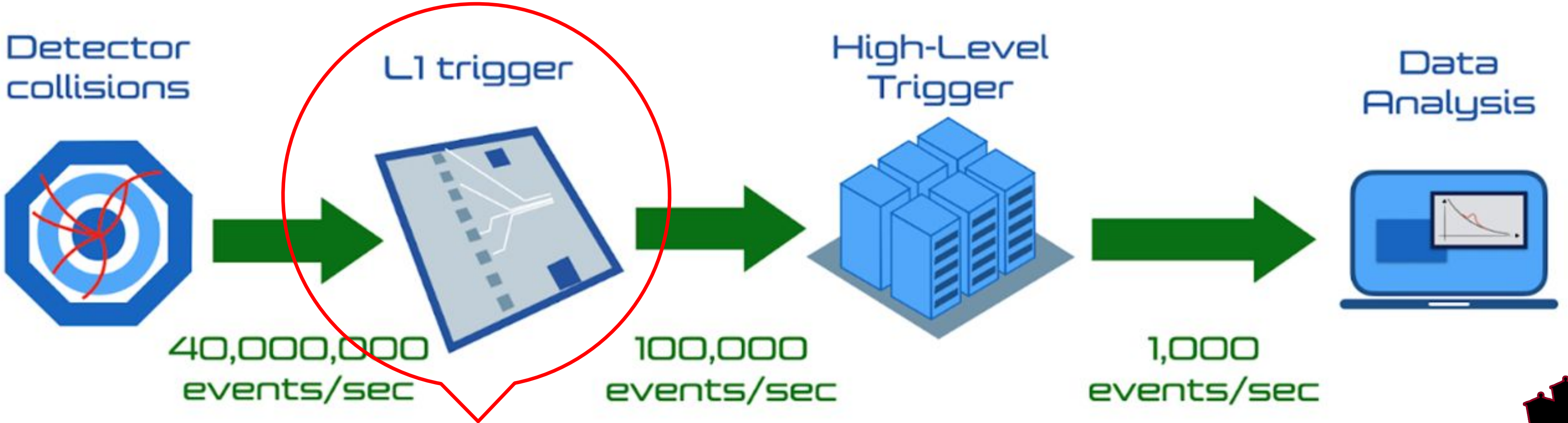
Deep Neural Networks can make a **fast event selection** in an extremely dense environment, and can therefore be used where the event selection happens.



source

Deep Neural Networks at rescue

Deep Neural Networks can make a **fast event selection** in an extremely dense environment, and can therefore be used where the event selection happens.

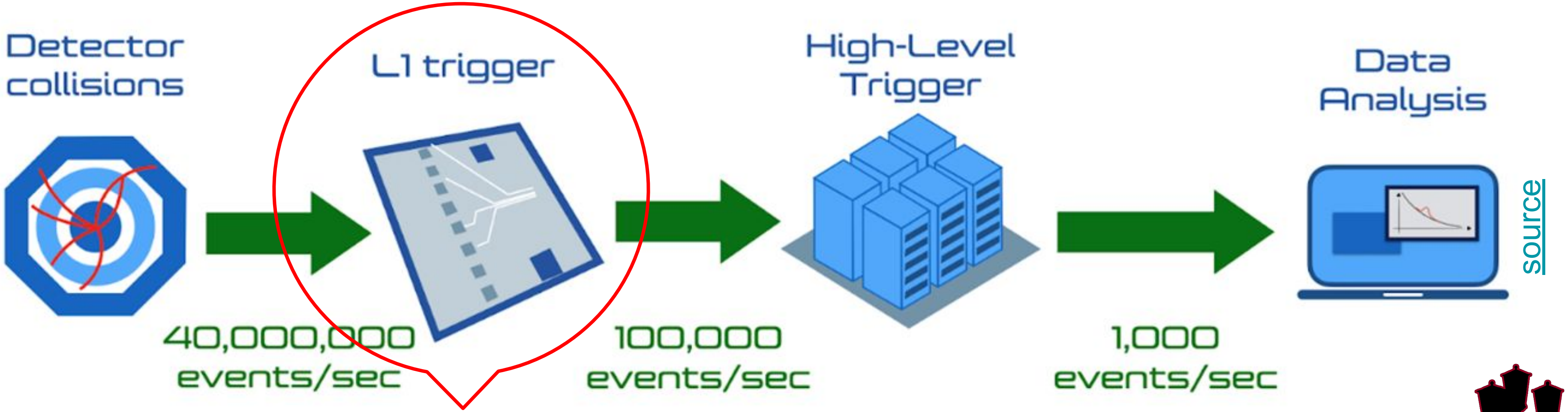


source

L1 of data processing typically uses custom hardware with FPGAs

Deep Neural Networks at rescue

Deep Neural Networks can make a **fast event selection** in an extremely dense environment, and can therefore be used where the event selection happens.

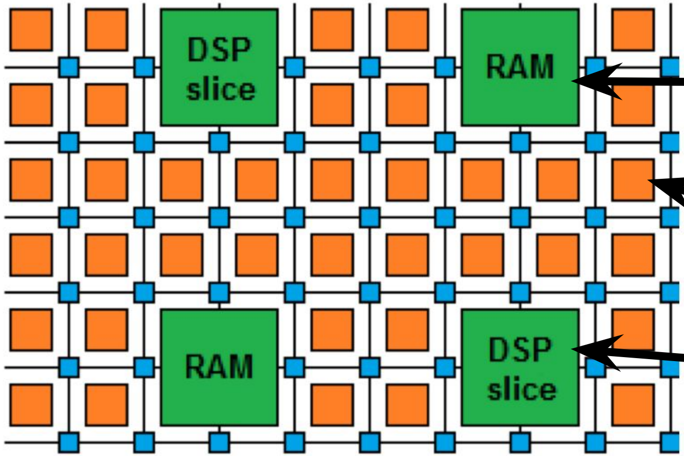


L1 of data processing typically uses custom hardware with FPGAs

Let's run Deep Neural Networks in real-time on FPGAs to improve event selection!

FPGAs

FPGAs (Field-Programmable Gate Arrays) are programmable integrated circuits.



source

Random Access Memories to store constant values

Logic cells for simple arithmetic operations

Digital Signal Processors to perform multiplications

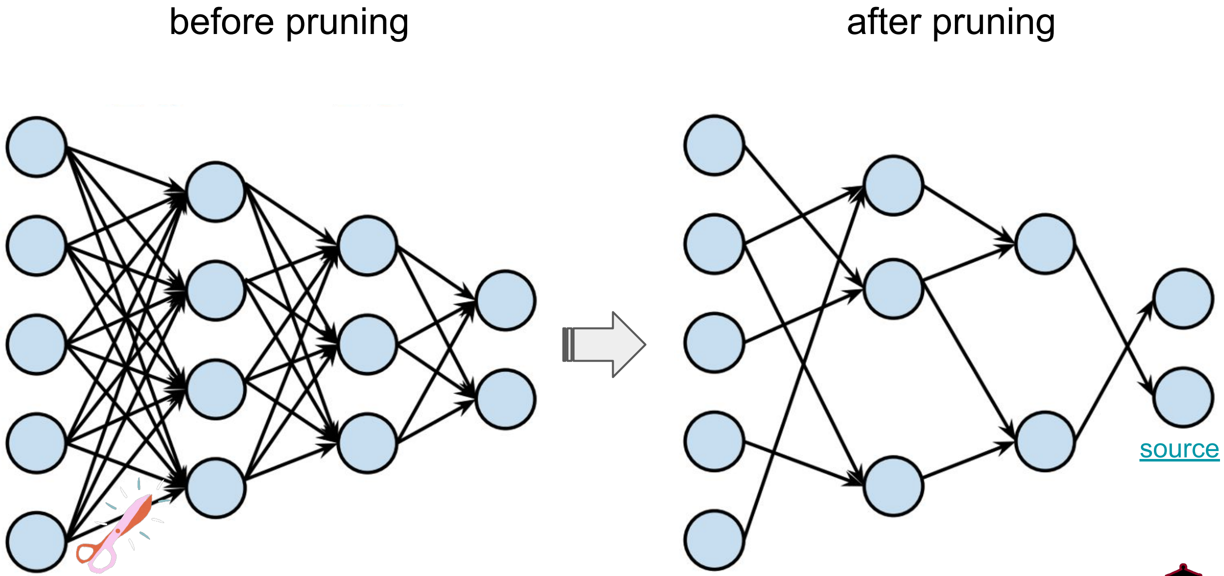
Depending on the FPGA resources available,
we should know how to **reduce the size** of a network.

ACAT 2022 BARI

Pruning

One way of **reducing** the size of a neural network is **pruning**.

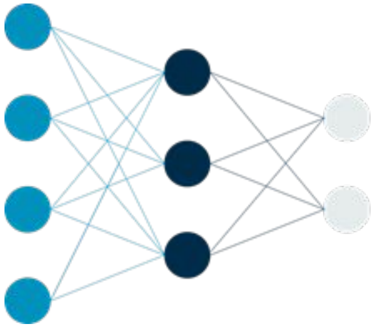
Pruning = **removing** superfluous structure



source

The usual pruning scheme

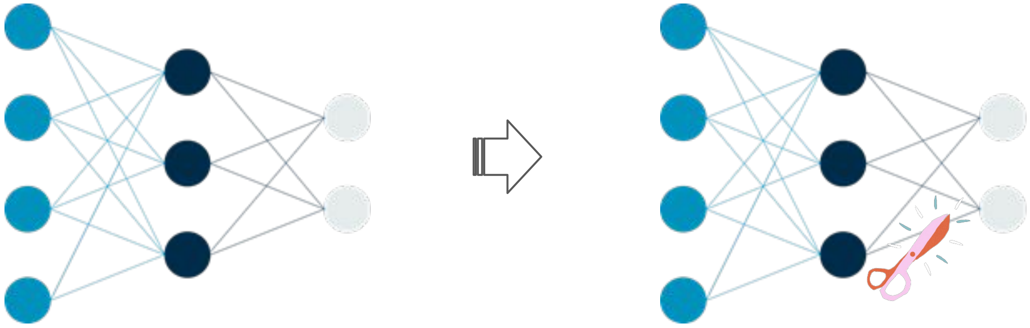
1. Train



Davis Blalock et al., *What is the state of neural network pruning?*, Proceedings of machine learning and systems 2 (2020), pp. 129–146

The usual pruning scheme

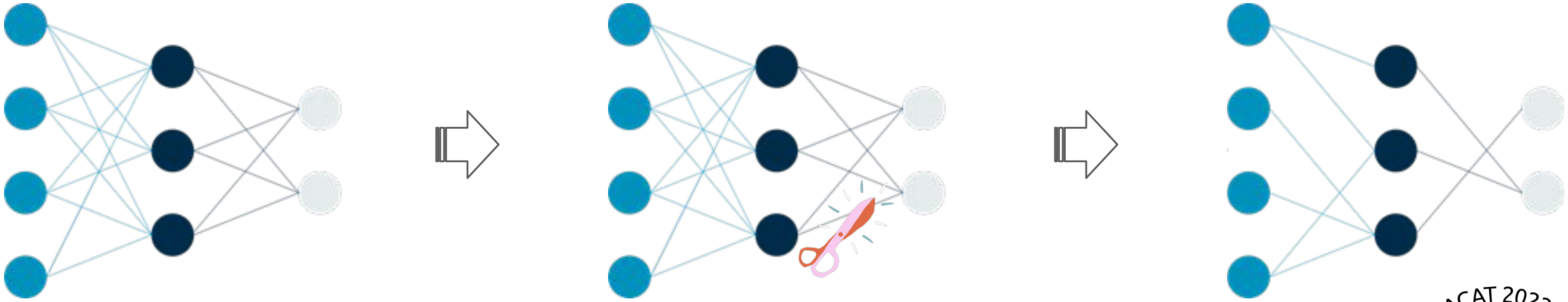
1. Train
2. Prune synapses



Davis Blalock et al., *What is the state of neural network pruning?*, Proceedings of machine learning and systems 2 (2020), pp. 129–146

The usual pruning scheme

1. Train
2. Prune synapses
3. Retrain

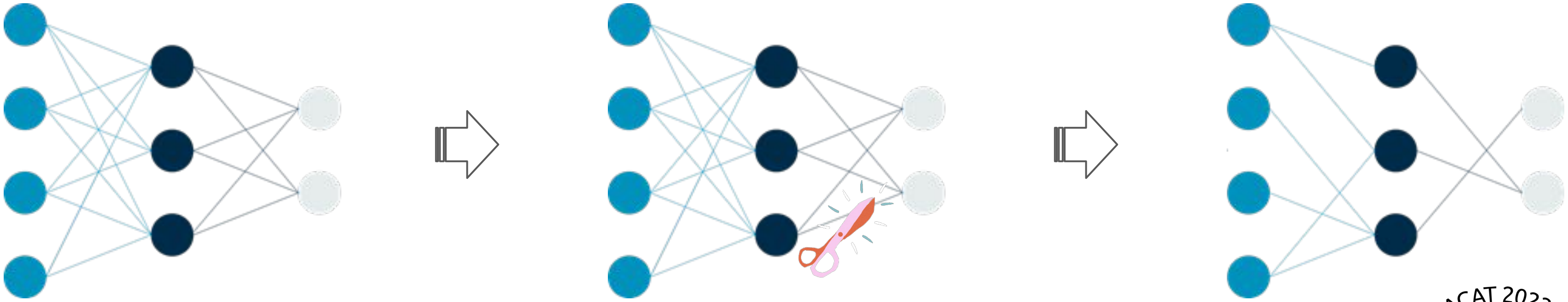


Davis Blalock et al., *What is the state of neural network pruning?*, Proceedings of machine learning and systems 2 (2020), pp. 129–146

The usual pruning scheme

Iterate (fine tuning)

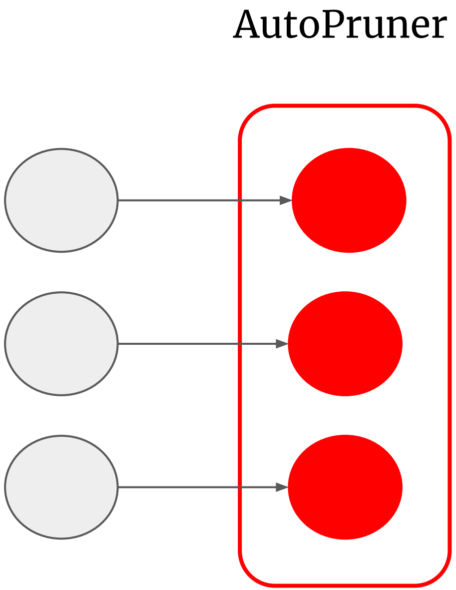
1. Train
2. Prune synapses
3. Retrain



Davis Blalock et al., *What is the state of neural network pruning?*, Proceedings of machine learning and systems 2 (2020), pp. 129–146

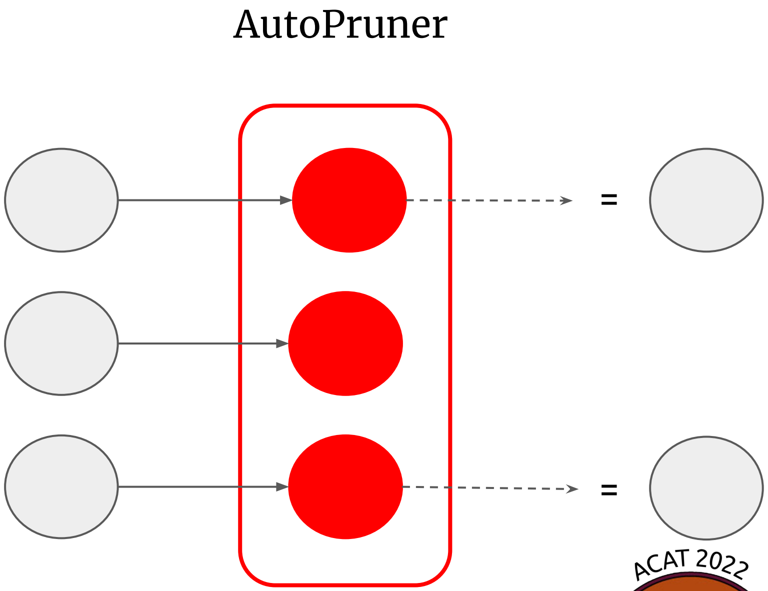
AutoPruner: a different pruning strategy

- it can prune **nodes**
- it prunes **during training**
- the number of nodes to be pruned can be determined by the **user**
- it can determine the most suitable **network architecture**



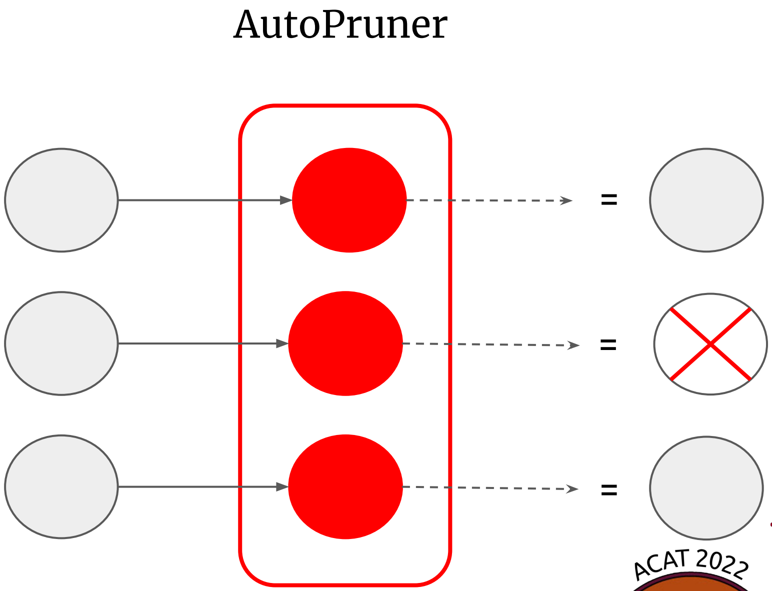
AutoPruner: a different pruning strategy

- it can prune **nodes**
- it prunes **during training**
- the number of nodes to be pruned can be determined by the **user**
- it can determine the most suitable **network architecture**



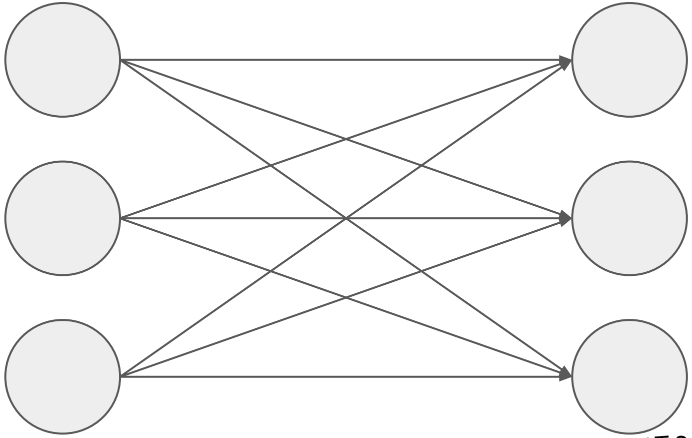
AutoPruner: a different pruning strategy

- it can prune **nodes**
- it prunes **during training**
- the number of nodes to be pruned can be determined by the **user**
- it can determine the most suitable **network architecture**



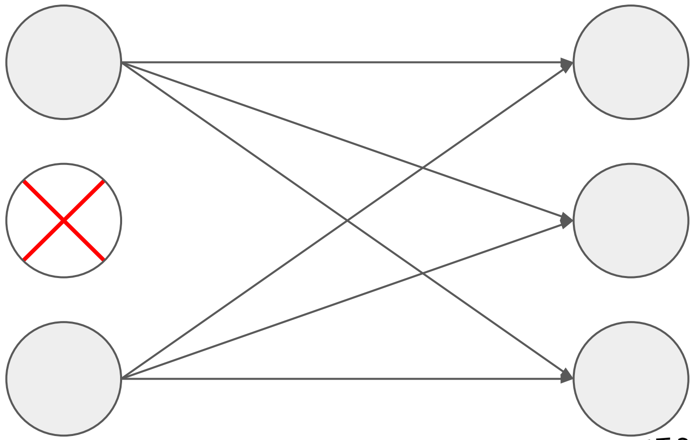
AutoPruner: a different pruning strategy

- it can prune **nodes**
- it prunes **during training**
- the number of nodes to be pruned can be determined by the **user**
- it can determine the most suitable **network architecture**



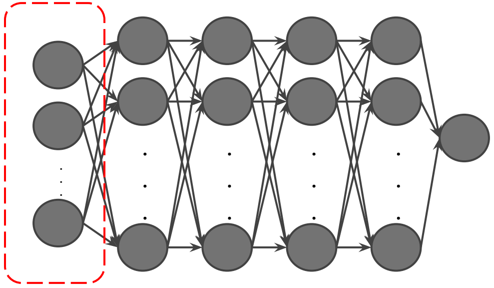
AutoPruner: a different pruning strategy

- it can prune **nodes**
- it prunes **during training**
- the number of nodes to be pruned can be determined by the **user**
- it can determine the most suitable **network architecture**

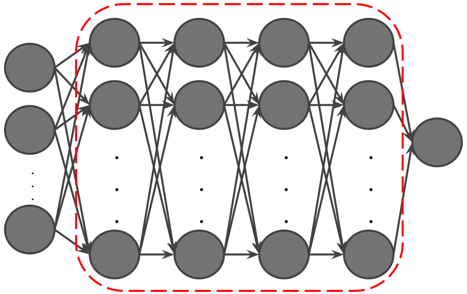


Pruning with AutoPruner

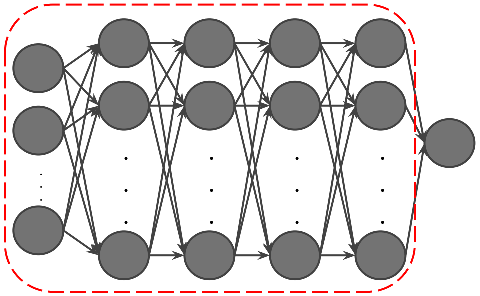
With AutoPruner you can **choose** which part of the network you want to prune



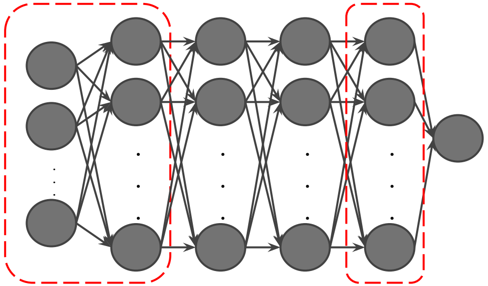
- the input layer



- the hidden layers



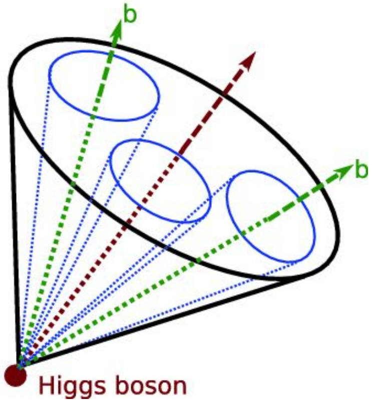
- the whole network

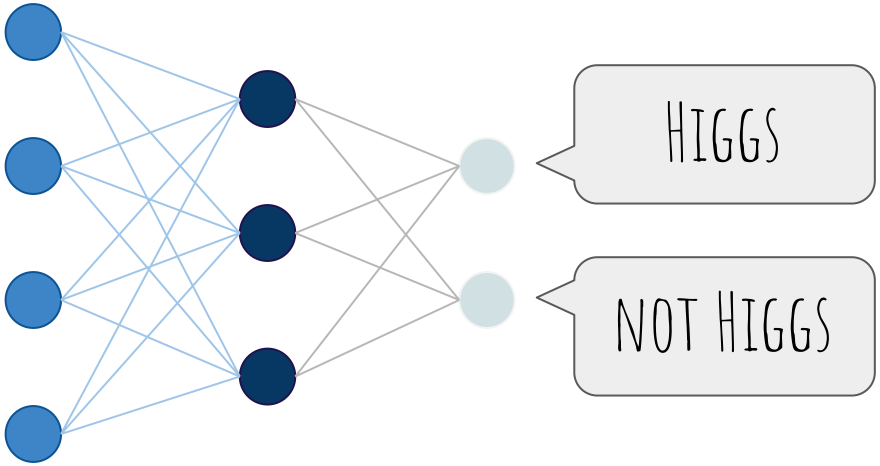


- one or more portions of the network

Use case

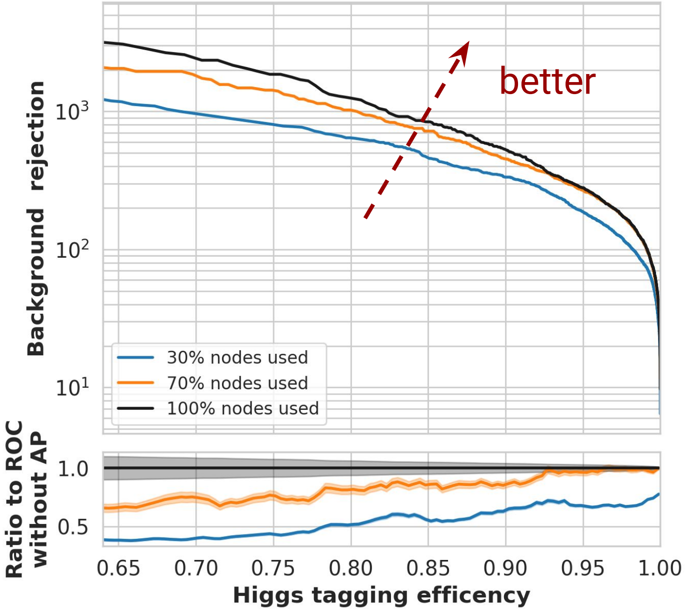
Identify jets that contain both the *b* quarks from boosted Higgs decay in *pp* collision experiments using Deep Neural Networks





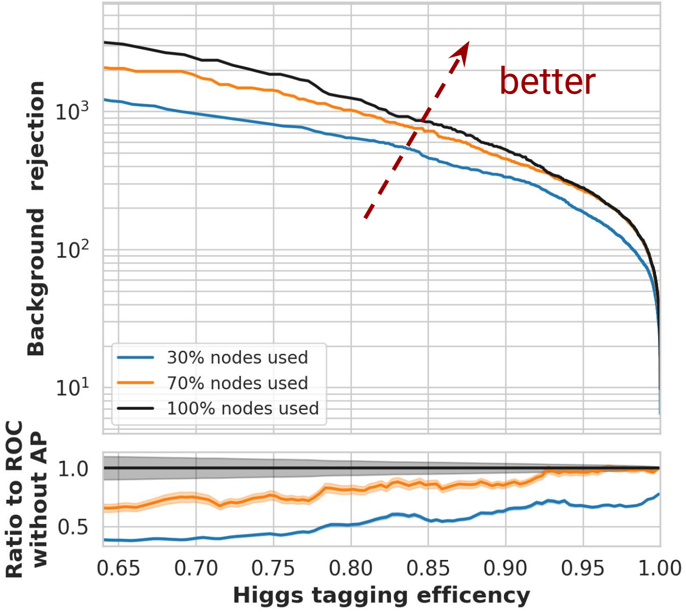
Results

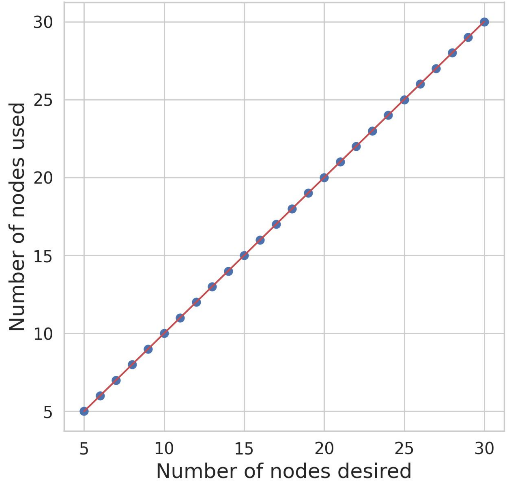
The performance increases with the percentage of nodes used, as expected: AutoPruner is really **switching off** nodes

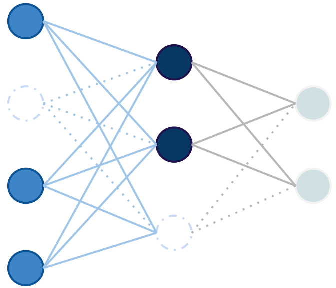


Results

The performance increases with the percentage of nodes used, as expected: AutoPruner is really **switching off** nodes

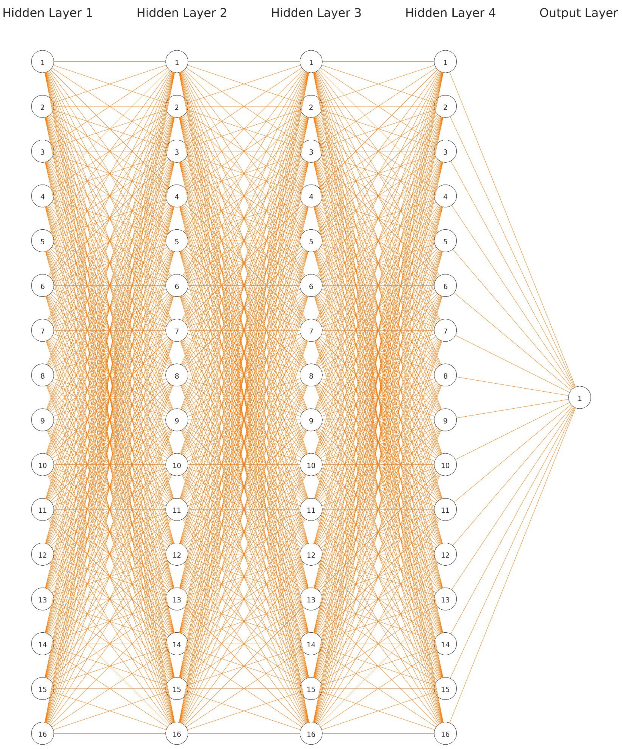




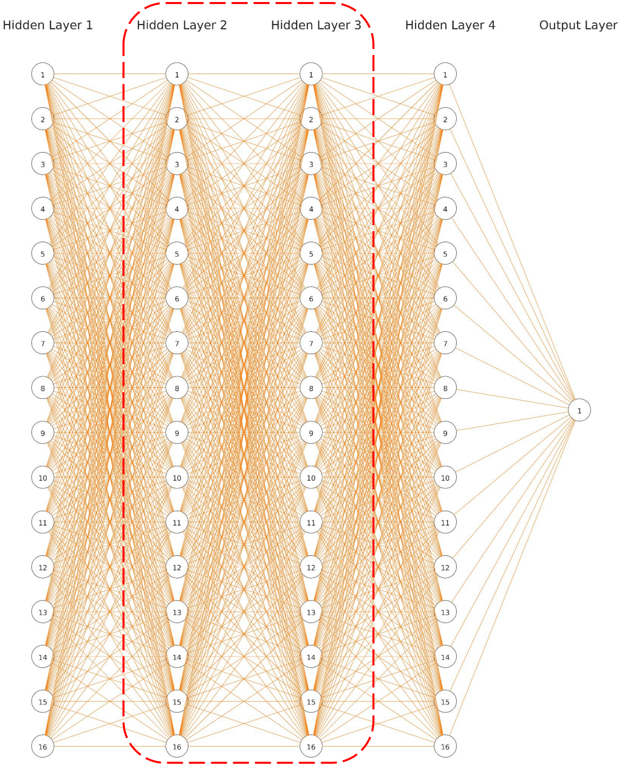


The total number of nodes used is **always** equal to the required number

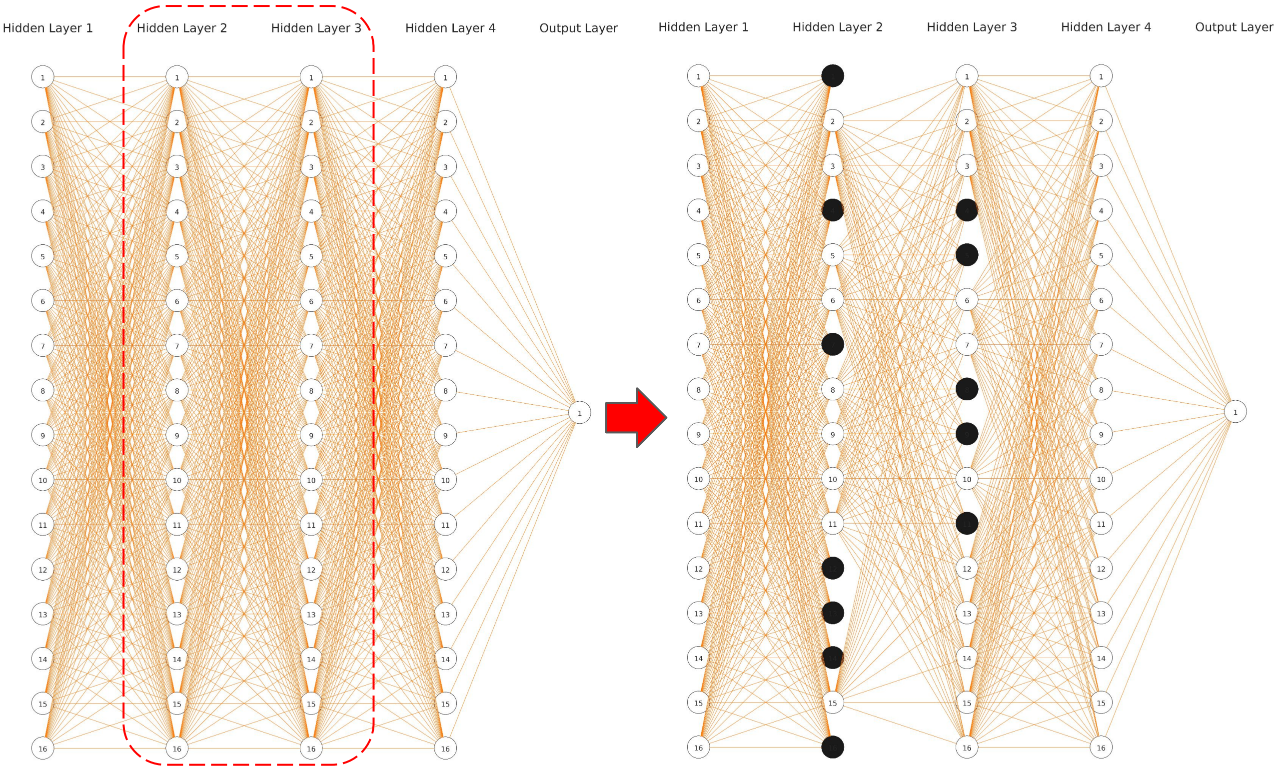
Models' comparison



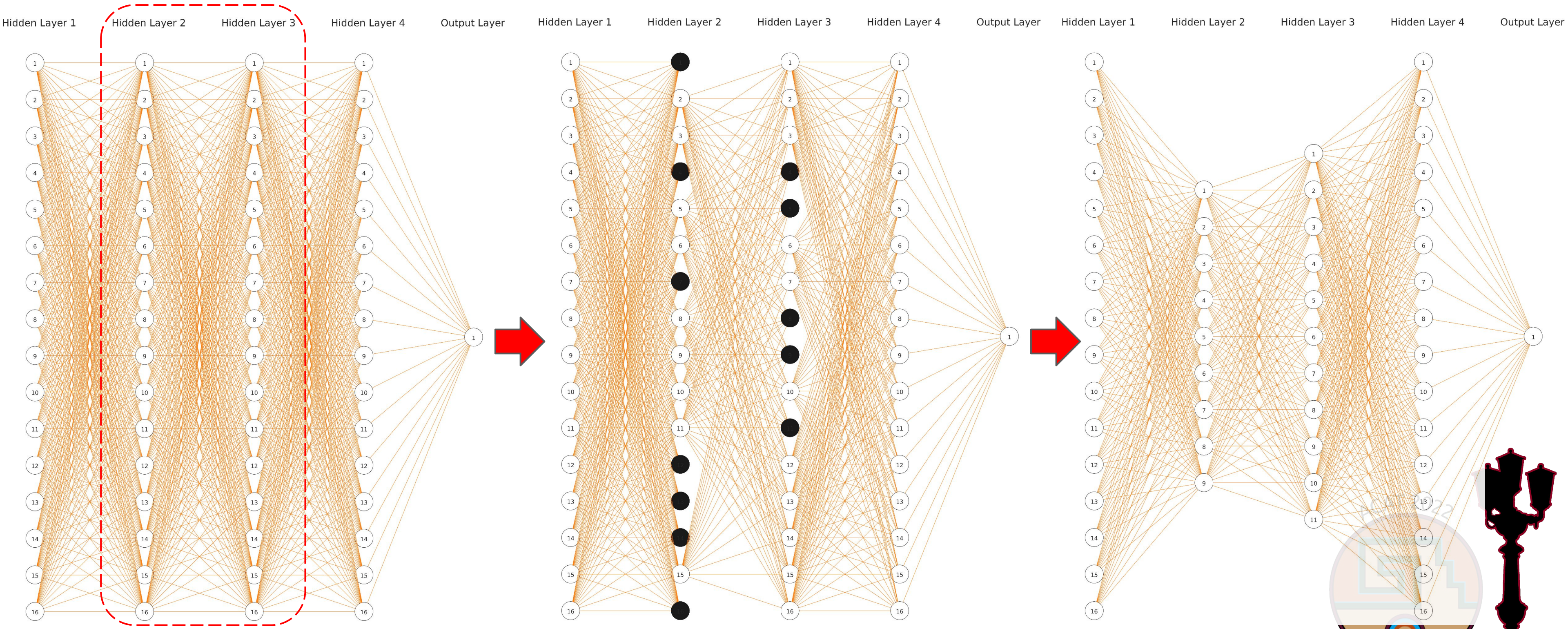
Models' comparison



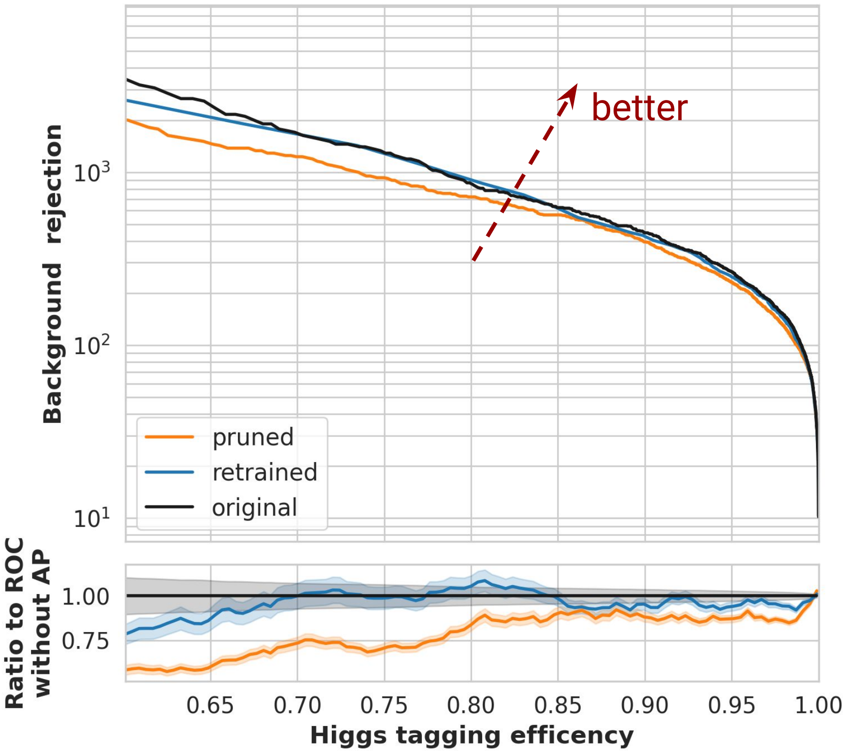
Models' comparison



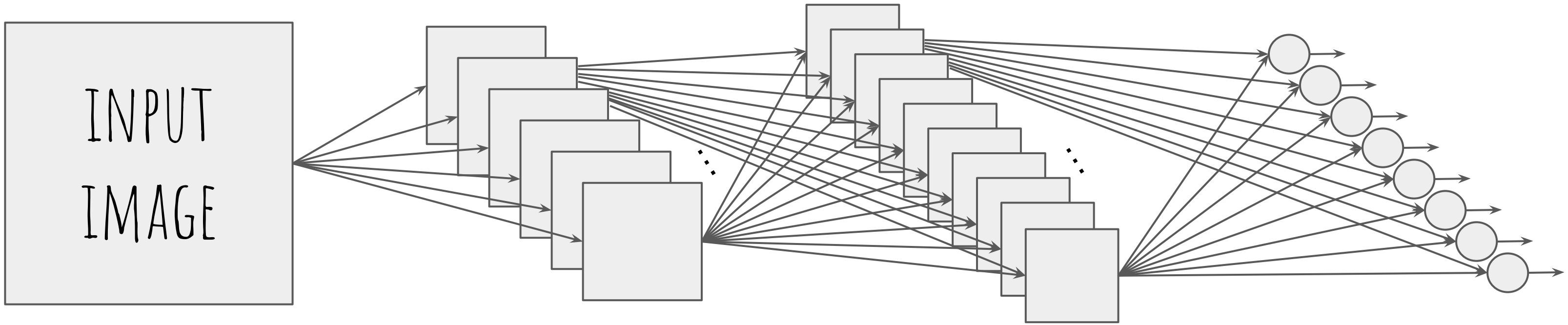
Models' comparison



Models' comparison

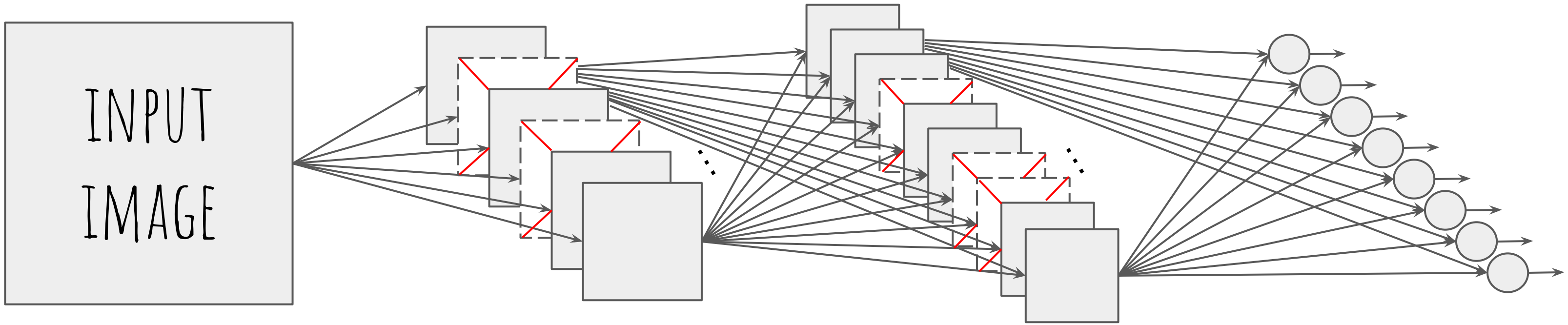


AutoPruner in Convolutional Neural Networks



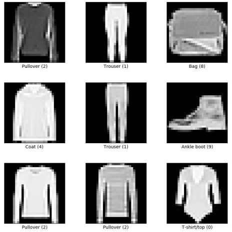
AutoPruner can be used with CNNs to prune filters during training

AutoPruner in Convolutional Neural Networks



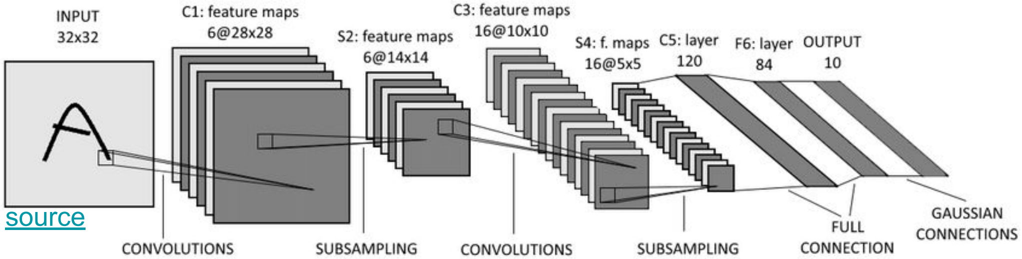
AutoPruner can be used with CNNs to prune filters during training

Preliminary results

Use case: 

source

+



source

FASHION-MNIST DATASET

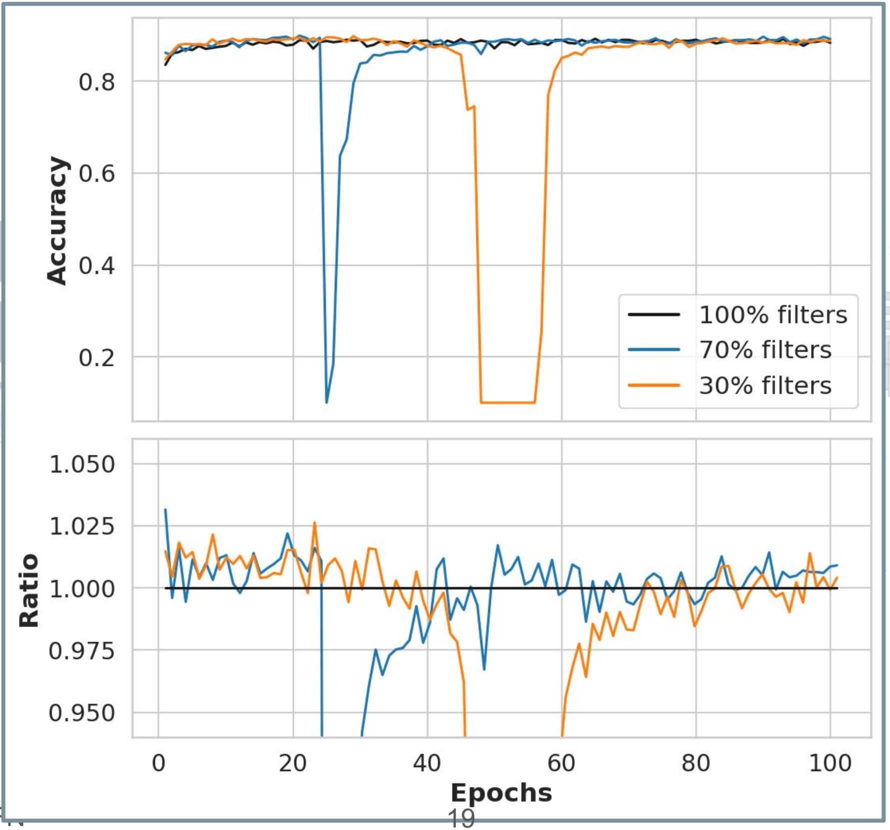
LENET-5 ARCHITECTURE

Preliminary results

Use case:

source

FASHION-M



CTURE

ACAT 2022

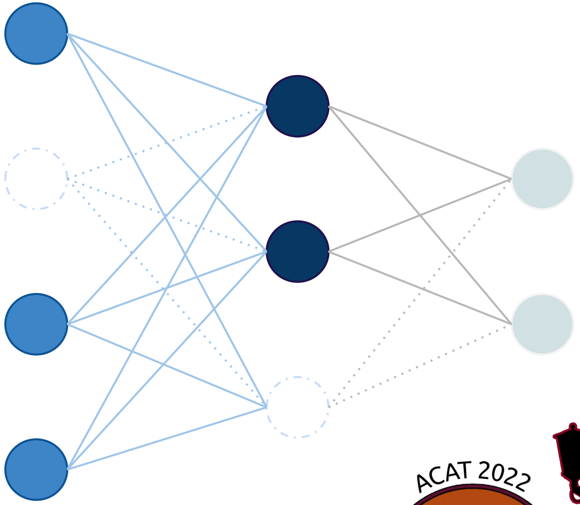
BARI

Conclusions

We introduced the AutoPruner approach to **effectively prune** Deep Neural Networks during training.

AutoPruner proved to be:

- **simple** to incorporate
- **effective** and **successful** in reducing the networks' size
- **fast** (pruning during training, no need to fine tune)
- very **understandable**



Further developments are focusing on:

- apply AutoPruner to Convolutional Neural Networks
- investigate feature selection with AutoPruner

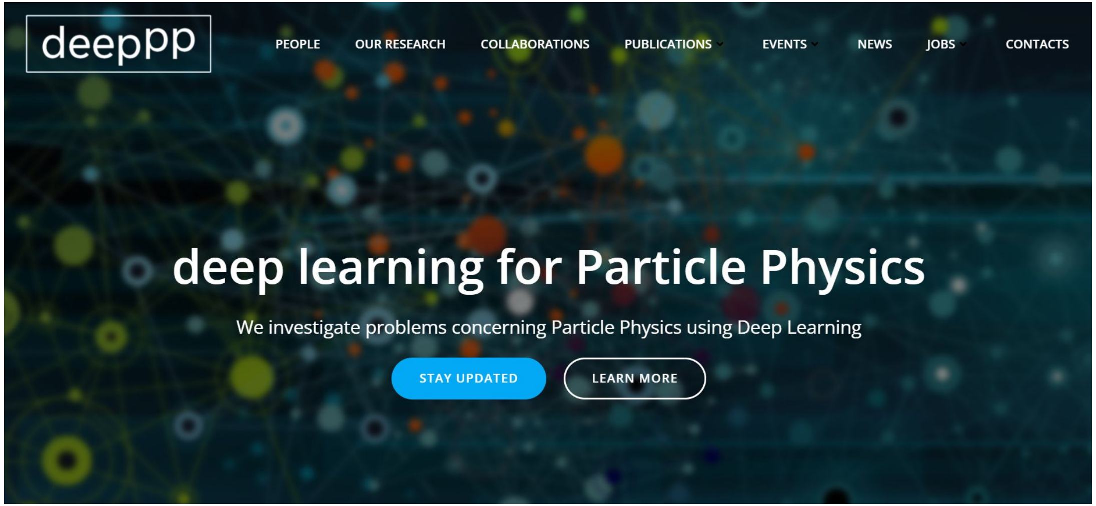
Thanks!

Want to know more about Deep Learning applications in Particle Physics?

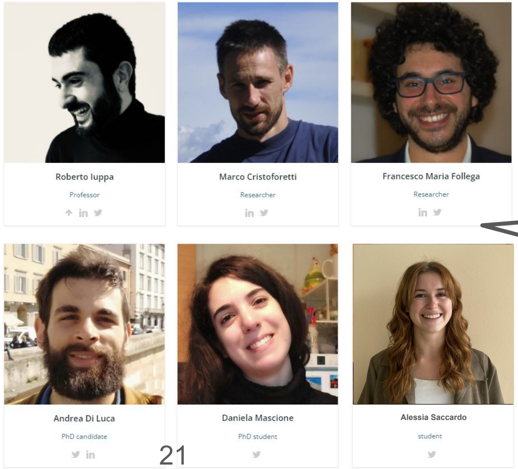
Awesome!

Visit
https://www.deeppp.eu/

deepPP



Our work



Us

Who we work with

ATLAS EXPERIMENT

TIFPA

ACAT 2022 BARI