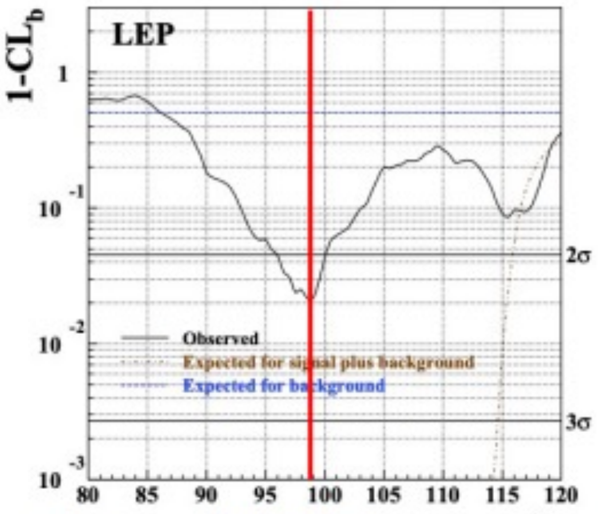
Improving the diphoton event selection with GNNs

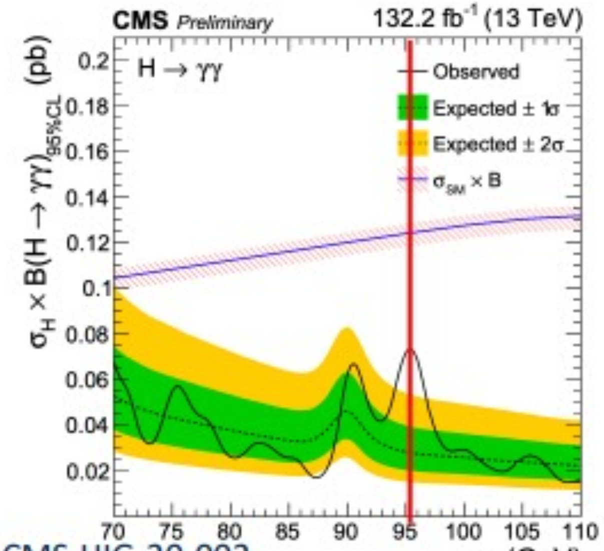
Speaker: Isai Roberto Sotarriva Alvarez

Jinnouchi laboratory, **Tokyo Institute of technology**

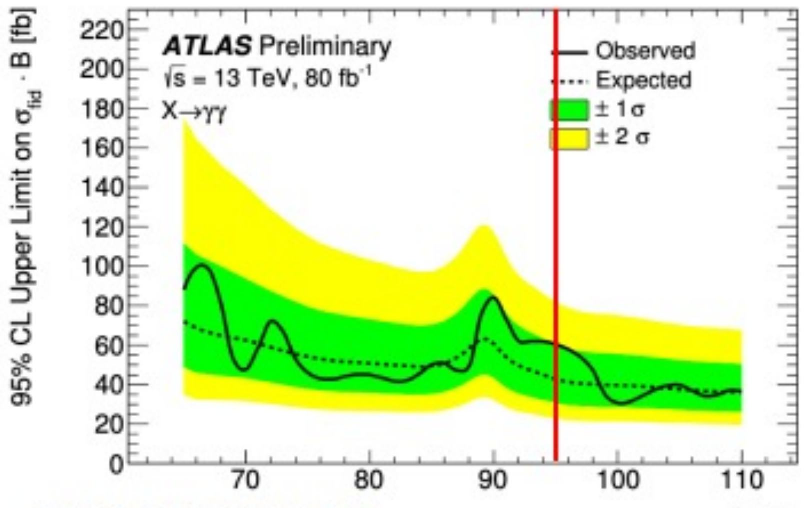
Introduction



arXiv: hep-ex/0306033



CMS-HIG-20-002

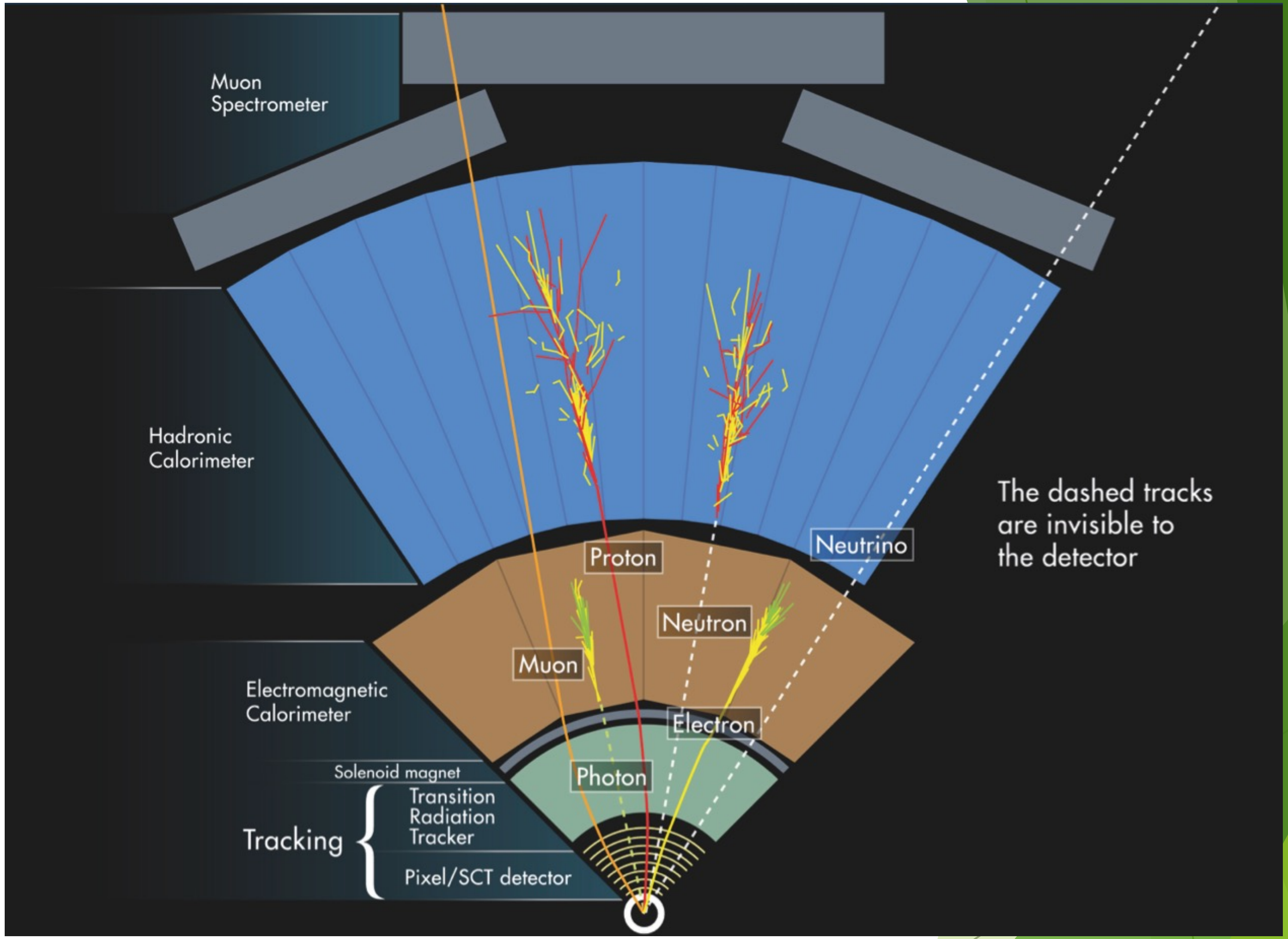


ATLAS-CONF-2018-025

- The existence of an additional spin-0 boson similar to the 125GeV Higgs but with a lower mass is predicted by some beyond the standard model (BSM) theories like supersymetry
- Analysis on Run 2 data from multiple experiments present a small excess around the 90-100GeV region.

  (Could this be a sign of a low mass counterpart of the Higgs?).

ATLAS



Vanden Broeck. Renilde, [ATLAS - Visualising particles. Visualiser les particules], 2019

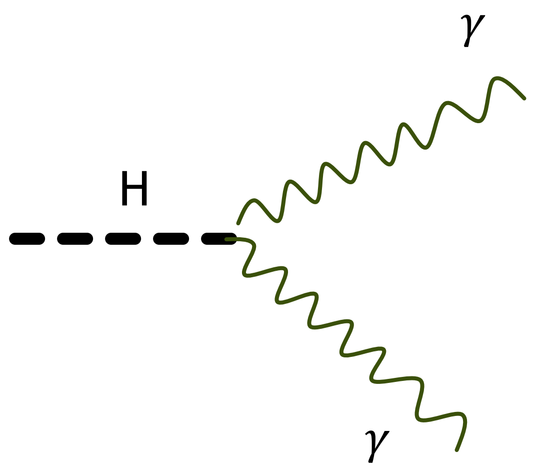
- The ATLAS experiment is one of the 2 general purpose detectors at the LHC.

- Objective of the project:
- Test the predictions of the Standard Model.
- Search for new physics beyond the Standard Model.

The ATLAS experiment relies on layers of specialized detectors for the particle reconstruction

Ideally photons are reconstructed only on the electromagnetic calorimeter. However, they might “convert” into an electron-positron pair before reaching the electromagnetic calorimeter.

Low mass $[60 - 110]GeV$ $\mathrm{H} \to \gamma\gamma$ analysis



For this analysis we want to:

a) Determine if an event contains any diphoton decay.
b) Identify the pair of photons originating from a diphoton decay process.

- On this analysis we are looking for a decay of an invisible particle H into 2 photons.
- On a single event there might or not be any pair of photons coming from this diphoton decay process.
- On each event there might be many background photon candidates from processes unrelated to the diphoton decay.
- Information regarding the photon is available from the reconstruction. Some valuable variables for this analysis are:
  - p_T: Transverse momentum (The component of the momentum on the direction perpendicular to the beam axis).
  - P: The 4-momentum of the particle.
  - Position of the conversion vertex.
  - Any variables related to the quality of the reconstruction (number of hits on the innermost pixel layer, n hits on all the pixel layers, shower shape variables, etc.)

Low mass $[60-110]GeV$ $\mathrm{H} \rightarrow \gamma\gamma$ analysis challenges

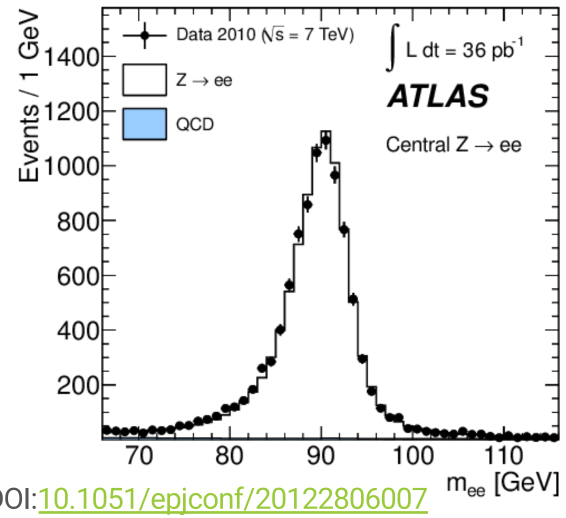
Background rejection is key. There are 2 main sources of background in our analysis:

- Non-resonant

  QCD jets misidentified as photons.

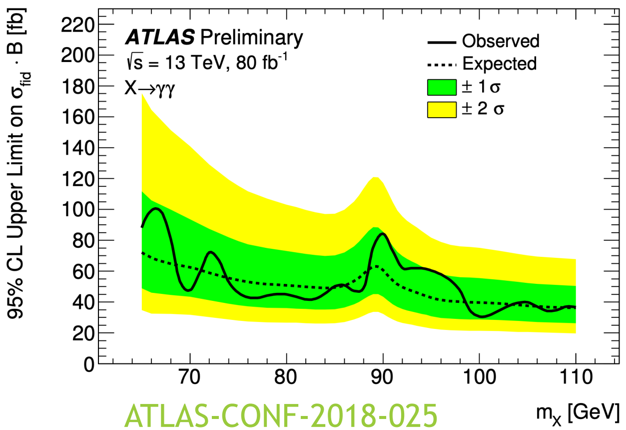
  ➢ Method of rejection: Kinematic cuts and isolation criteria

- Resonant Drell-Yan dielectron processes:



DOI:10.1051/epjconf/20122806007

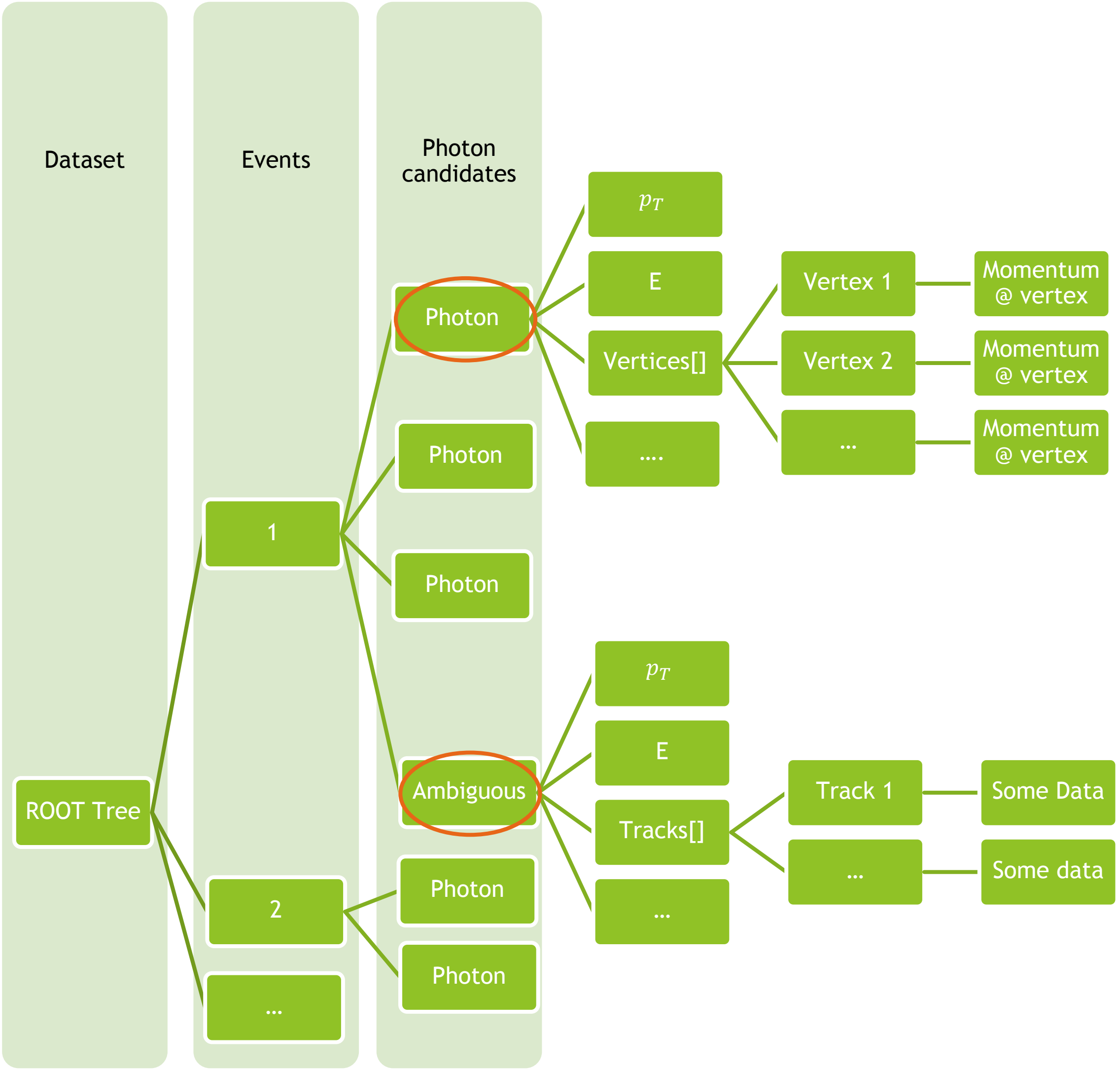
The Z—> ee peak is located at around $91GeV$, and is clearly visible on the expected diphoton curve on all previous analysis



ATLAS-CONF-2018-025

➢ Method of rejection: BDT (RUN2 only)

How does our data look like?



- There are 2 kinds of photon candidates in the events:

-Photons: Almost 100% guaranteed to be photons

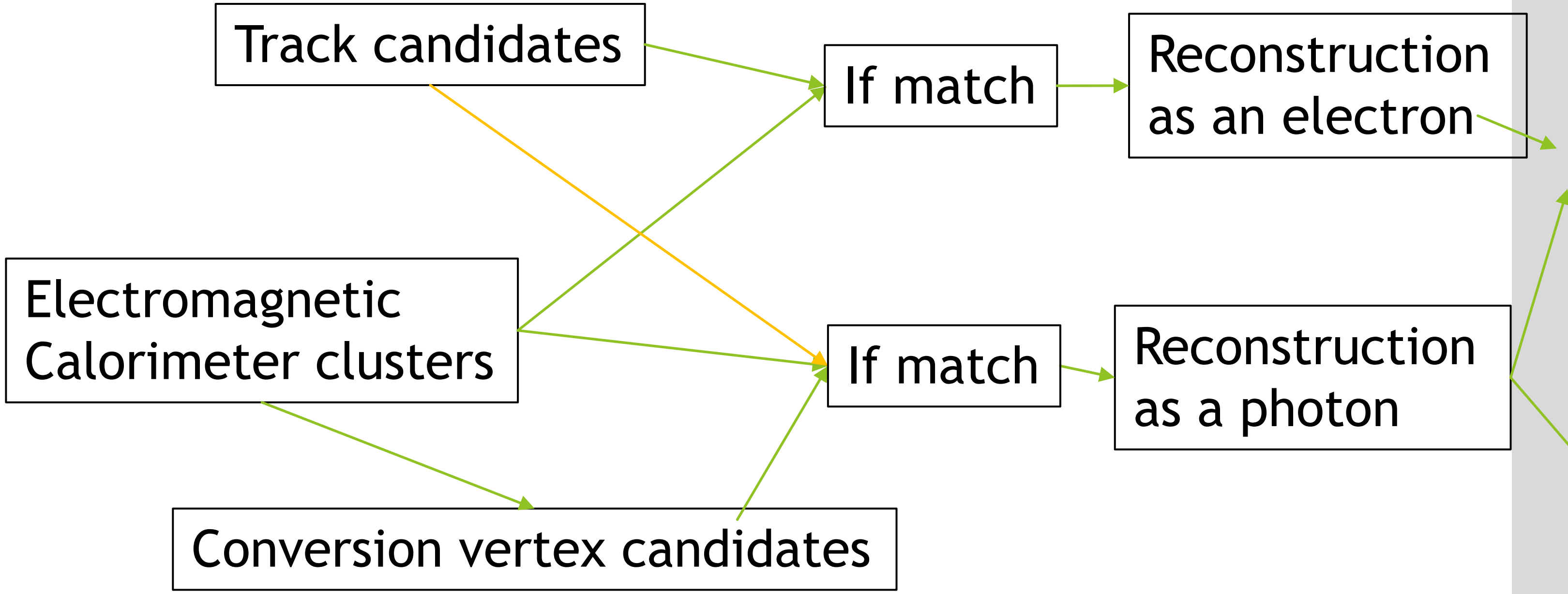
-Ambiguous: Could be photons or electrons.

- Each candidate can contain 2 types of variables:

-Scalar: Can be expressed by a single value

-Arrays: Each element of the array is an array of scalar values.

Photon Vs. ambiguous objects



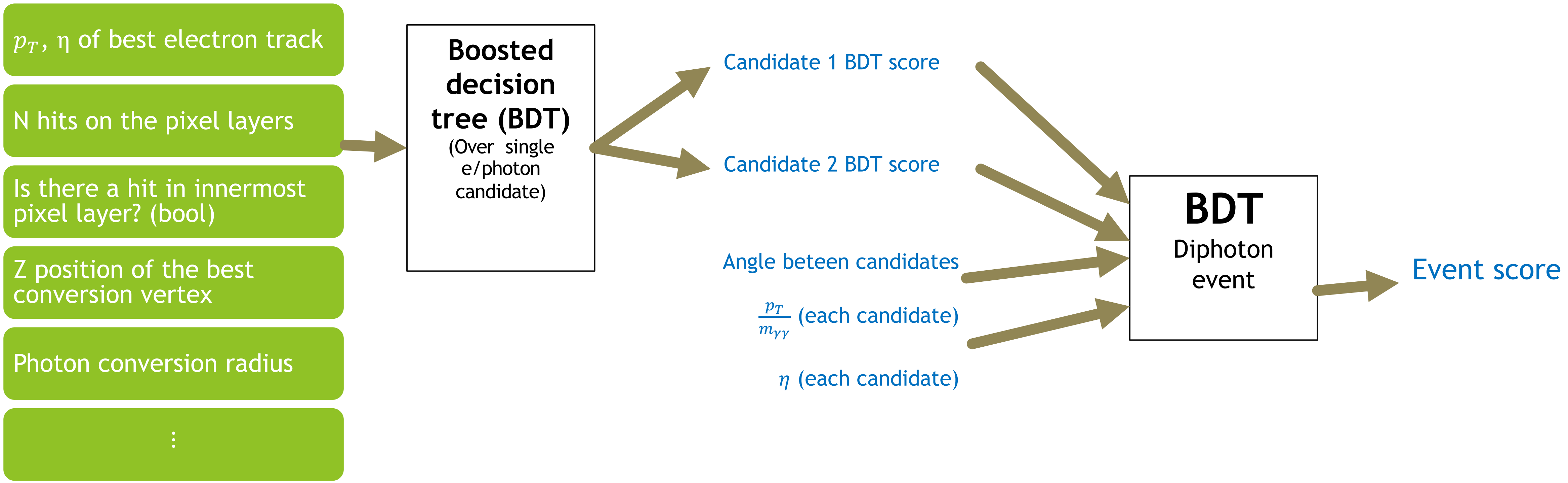
Object generated

If both reconstructions exist:
Ambiguous={Electron reconstruction ,Photon reconstruction}

If only photon reconstruction exists:
Photon={Photon reconstruction}

An ambiguous object represents a particle that could be interpreted as an electron or as a photon. We are not sure of the true identity of the particle (electron or photon).

Diphoton event selection in RUN2



Opportunities for improvement

The number of variables used for classification could be increased

The second BDT is limited to only the e/γ BDT score.
(If the BDT could use all the data from the reconstruction instead of just a summary it should result in better predictions)

This is done only for the 2 candidates with the highest p_T. Why not let the method choose the pair itself?

- p_T, η of best electron track
- N hits on the pixel layers
- Is there a hit in innermost pixel layer? (bool)
- Z position of the best conversion vertex
- Photon conversion radius
- ⋮

Boosted decision tree (BDT)
(Over single e/photon candidate)

Candidate 1 BDT score

Candidate 2 BDT score

Angle beteen candidates

$\frac{p_T}{m_{\gamma\gamma}}$ (each candidate)

η (each candidate)

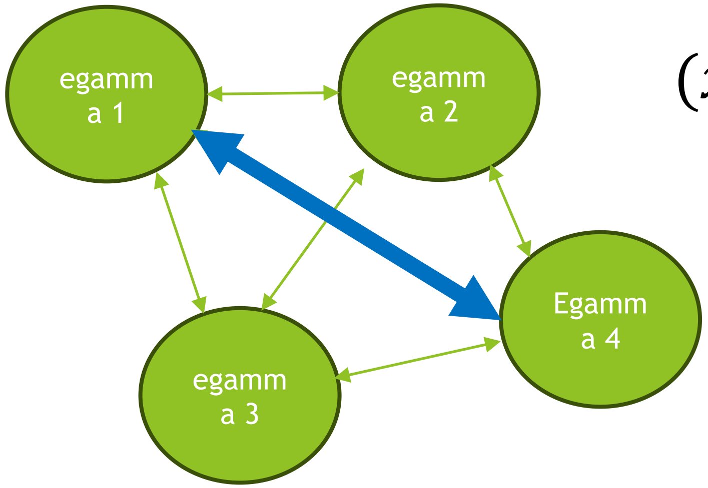
BDT
Diphoton event

Event score

Not well suited for processing tracks and vertices (arrays)

Creating a basic GNN for diphoton event selection

We want to do **edge prediction:**



Rest mass (in the hypothetical case of a diphotons decay)

TARGET FOR EDGE PREDICTION
Indicated if the pair of candidates is the result of a diphoton decay.
0 if not, 1 if it is.

This is done only for the 2 candidates with the highest p_T. Why not let the method choose the pair itself?

With a GNN all the possible combinations are considered.

$$(x_\alpha, x_\beta) = [m_{\gamma\gamma}, \Delta\omega] \rightarrow [isX \rightarrow \gamma\gamma]$$

Angle between the photon pair

Egamma is a vector of features
Each node is a photon candidate (photon or ambiguous object)

Label indicating both nodes come from a diphoton decay

The graph is constructed by connecting every node with each other. (fully connected graph)

Dealing with the variants of the e/gamma object (Photons and ambiguous)

The second BDT is limited to only the e/γ BDT score.
(If the BDT could use all the data from the reconstruction instead of just a summary it should result in better predictions)

To allow the GNN to accept both photons and ambiguous candidates as nodes we need to convert both objects into some common ground (let's call it egamma).
This is the job of the **encoder.**

The Egamma intermediary does not only allow for much more than one feature to be passed on, but also the features can be determined by the model to minimize the information lost given the number of features.

Photon ⇒ Photon encoder

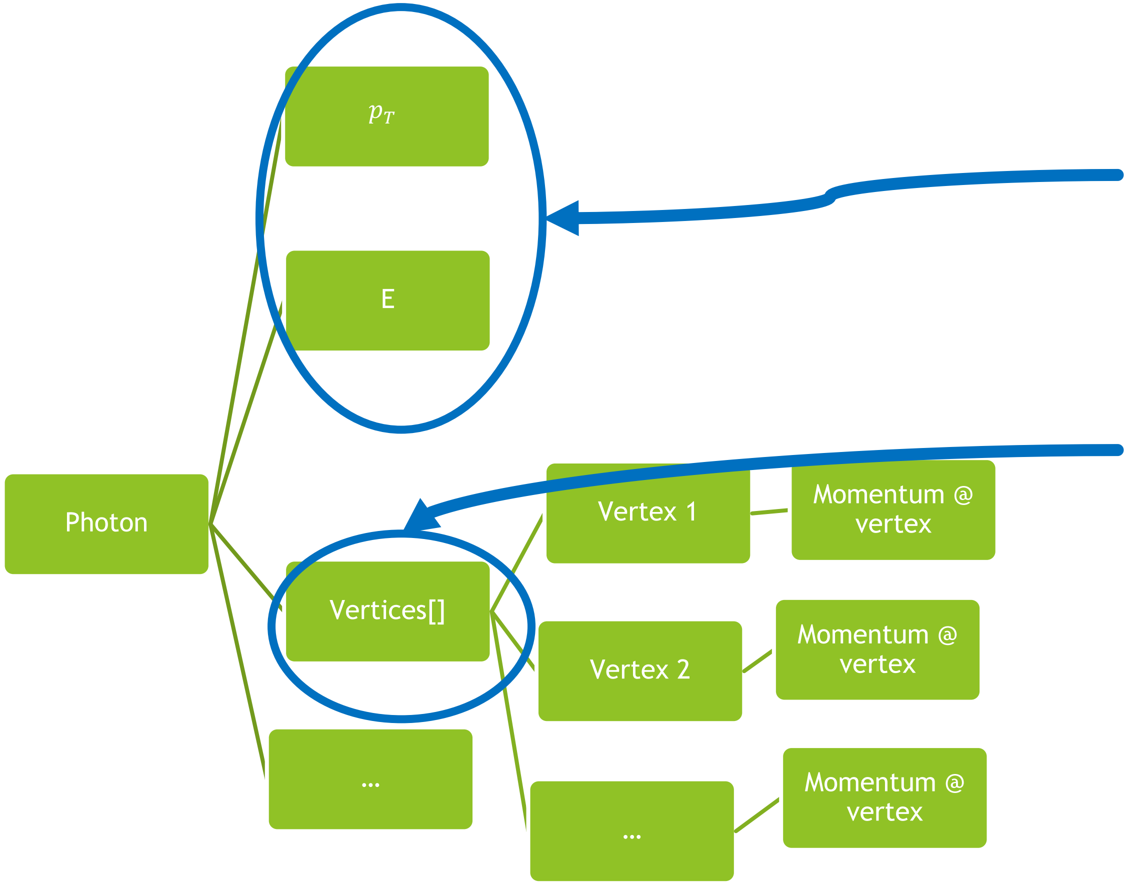
Ambiguous ⇒ Ambiguous encoder

Egamma=$[c_1, c_2, c_3, c_4, \ldots, c_m]$

Array of n features. It is essentially a ML compressed version of the input.

Encoder

There are many alternatives for this:
-MLP encoder: Concatenate encoded arrays (local) with scalar (global) features
-Transformer-encoder: Use attention to merge both encoded arrays (local) and scalar (global) features.



Both the photon and the ambiguous object features can be divided into 2 groups:

-Scalars: They can be added directly to an array of features

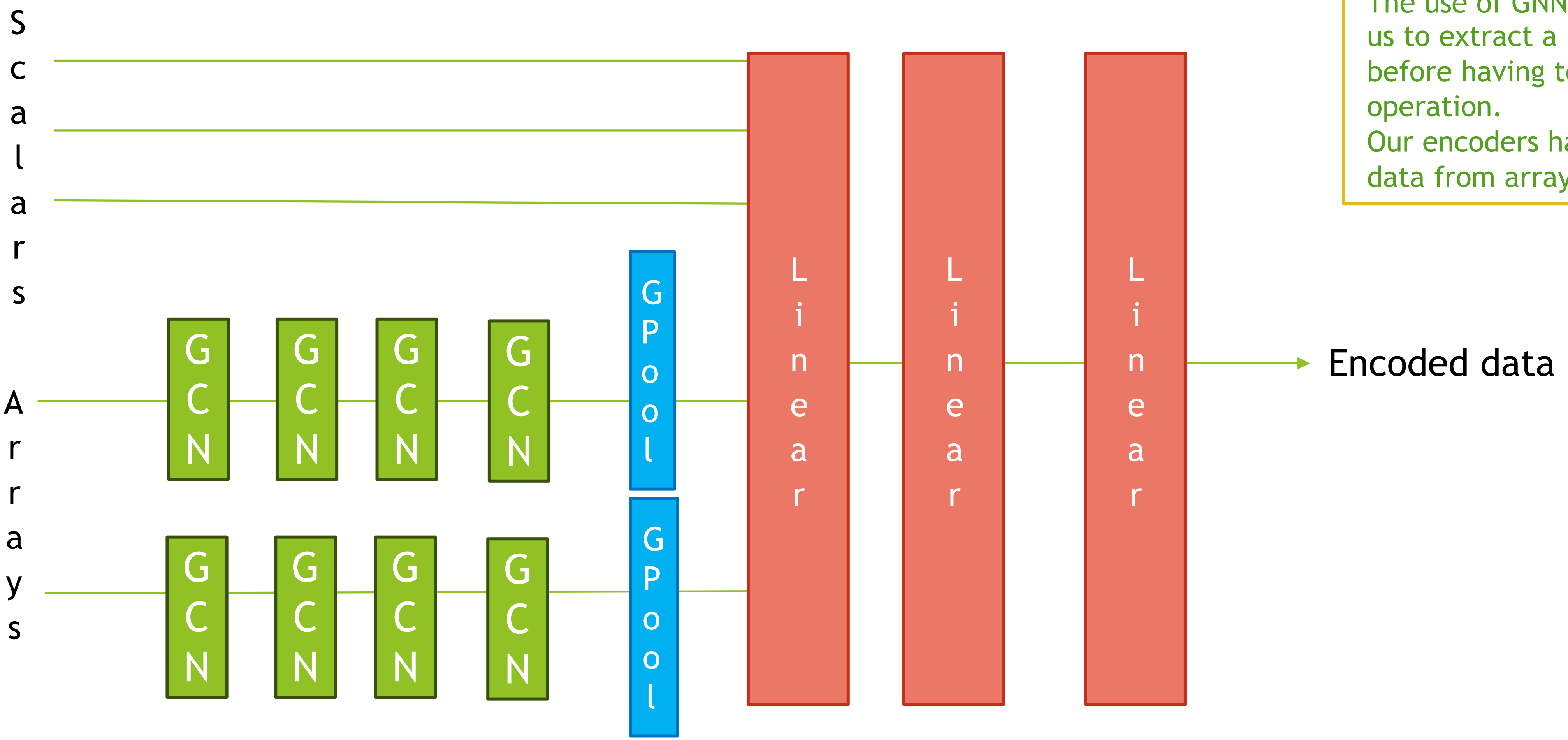
-Arrays: They need to be preprocessed and summarized somehow (this process it is on itself its own encoder inside our encoder). For this we can use an GNN.
The number of elements on an array is variable

- The summary of the array must always have the same number of features.

Dealing with arrays (encoder within an encoder)

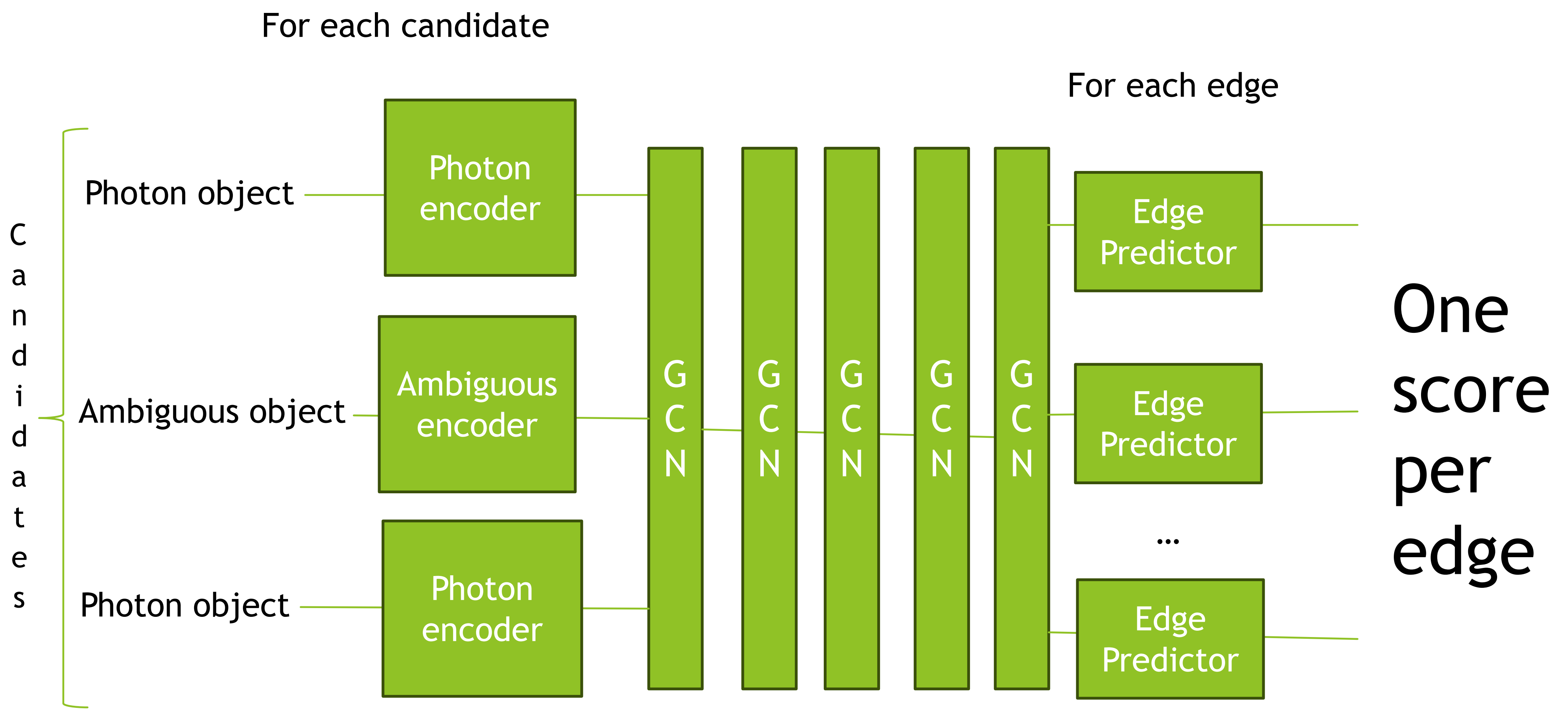
Not well suited for processing tracks and vertices (arrays)

The use of GNNs on the encoder allow us to extract a little bit more data before having to do any pooling operation.
Our encoders have no problem reading data from arrays.



*GCN is TorchGeometric graph convolutional layer
**Gpool is a graph mean pooling layer

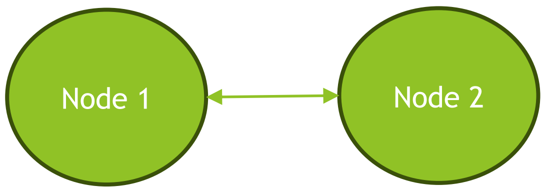
Final model overview



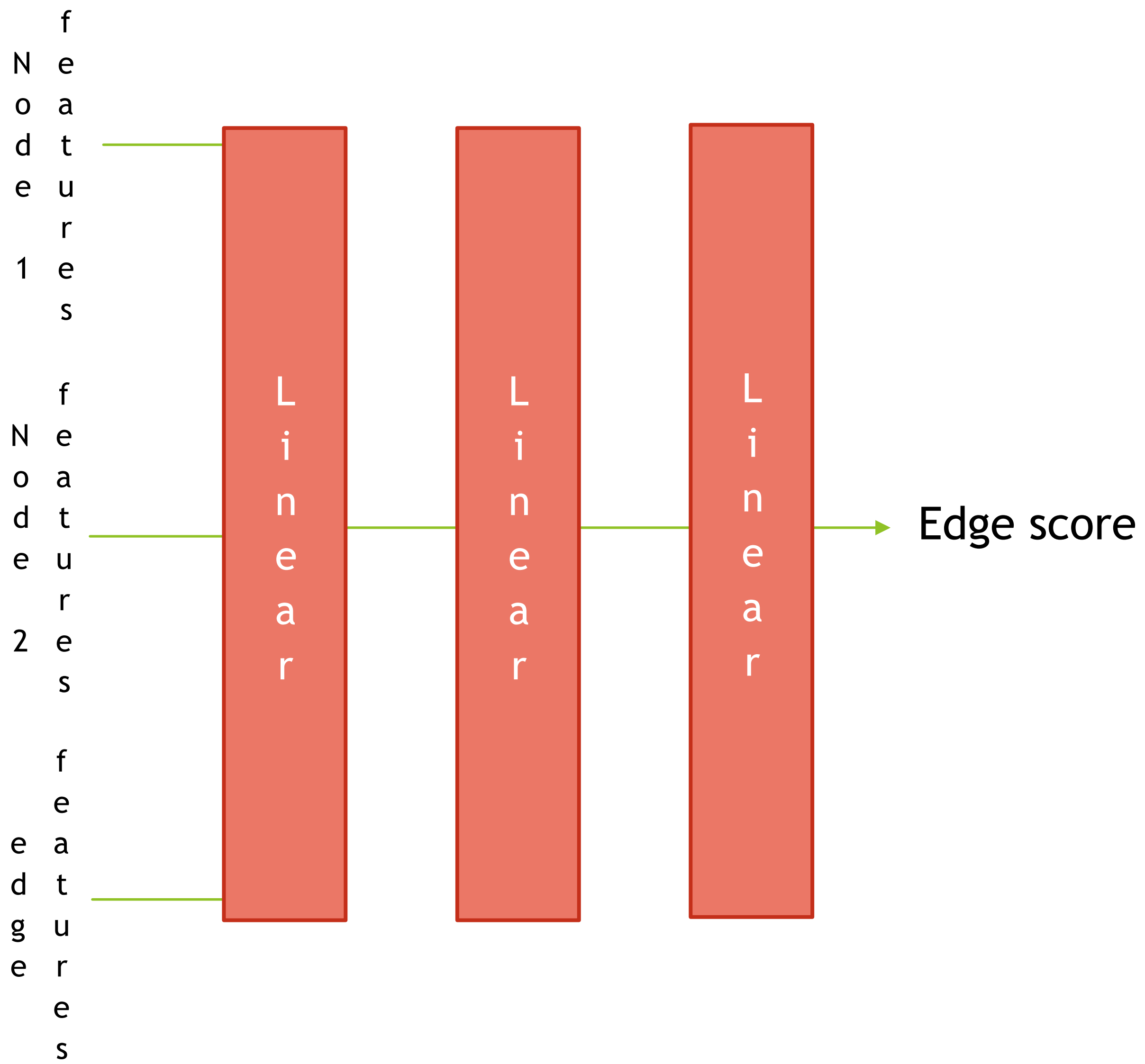
*GCN is TorchGeometric graph convolutional layer
**Gpool is a graph mean pooling layer

Edge predictor

The edge predictor is applied over a single edge.



And returns the edge score. (How likely is this specific pair to have come from a diphoton decay)



Summary

- Previous experimental results have shown hints of the possible existence of a second (BSM) Higgs boson with a mass of about $95GeV$.
- A significant part of the $\mathrm{H} \rightarrow \gamma\gamma$ analysis consists of minimizing the effect that the background has on the data. For this purpose, the $\mathrm{H} \rightarrow \gamma\gamma$ community is interested in using ML for improving the background rejection.
- On the RUN 2 analysis a two steps BDT approach was used to identify events with a diphoton signal versus fakes coming from electrons masquerading as photon (fake photons).
- A list of possible areas of improvement to the RUN 2 approach were discussed. And as shown later in the presentation GNNs happen to do a better job in all the right areas.
- The concept of encoder was presented, and it was shown how the encoders for both Photon and Ambiguous objects could be defined.
- Finally, a bird's eye vie of the architecture was shown.

Back up

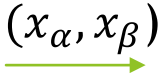
Graphs

A simple graph is composed of 2 main elements:

Nodes: A node represents an object.

(For simplicity, all nodes in a graph have the same number of features, in this case n.)

Example: $X_1 = [a1, a2, a3, a4, \ldots, an]$

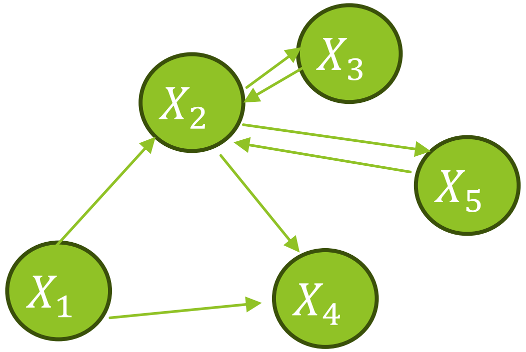


Edges: An edge represents a relationship or connection between 2 objects.

Note: Each edge can have a direction

(For simplicity, an all edges will have the same number of features, describing the possible relationship)

Example: $(x_\alpha, x_\beta) = [b1, b2, b3, b4, \ldots, bn]$

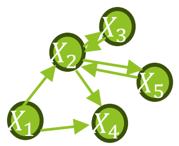


Graph neural networks (GNNs)

- A graph neural network (GNN) is a type of artificial neural network designed to process and analyze data structured as graphs, representing relationships between interconnected nodes
- A GNN operates on a similar way to an image convolution, but instead of working over a 3D table (an image) a GNN operates over a graph.
- A GNN can be used to do:

Graph prediction

Tries to predict a value about the entire graph.



$\Rightarrow$ Y=?

Node prediction

Tries to predict the value of one or more unknown node features.

$X_1 = [a1, a2, a3, a4, \dots, an] \to [Y]$

Edge prediction

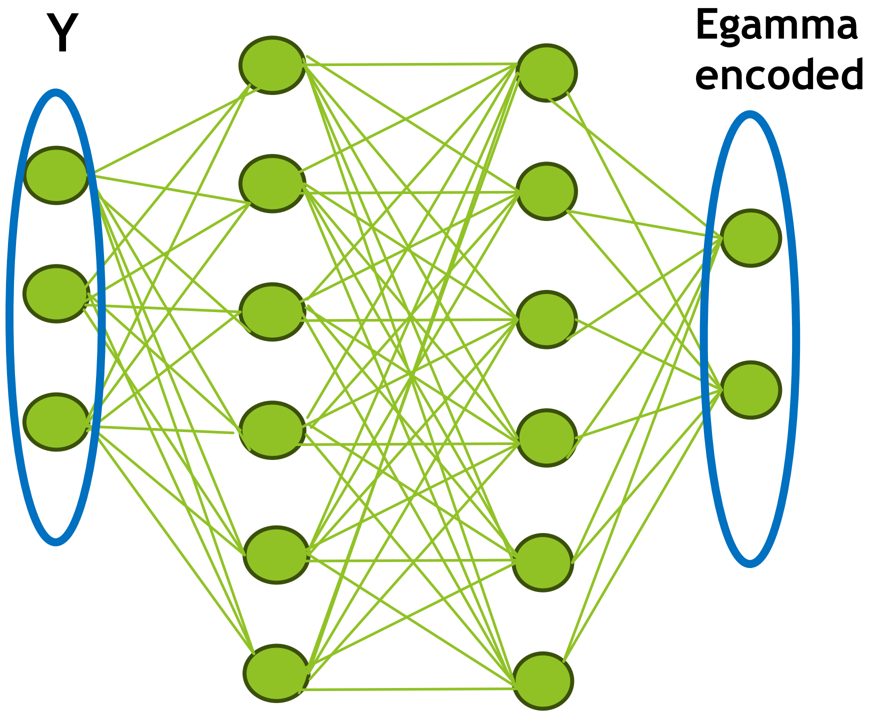
Tries to predict the value of one or more unknown edge features.

$\overrightarrow{(x_\alpha, x_\beta)} = [b1, b2, b3, \dots, bn] \to [Y]$

Creating the naive encoder

First create a vector y (size can always be calculated before hand)

Then run it through a MLP neural network



Y= All scalars + preprocessed output of all arrays

See previous slide

Graph neural networks current status at ATLAS

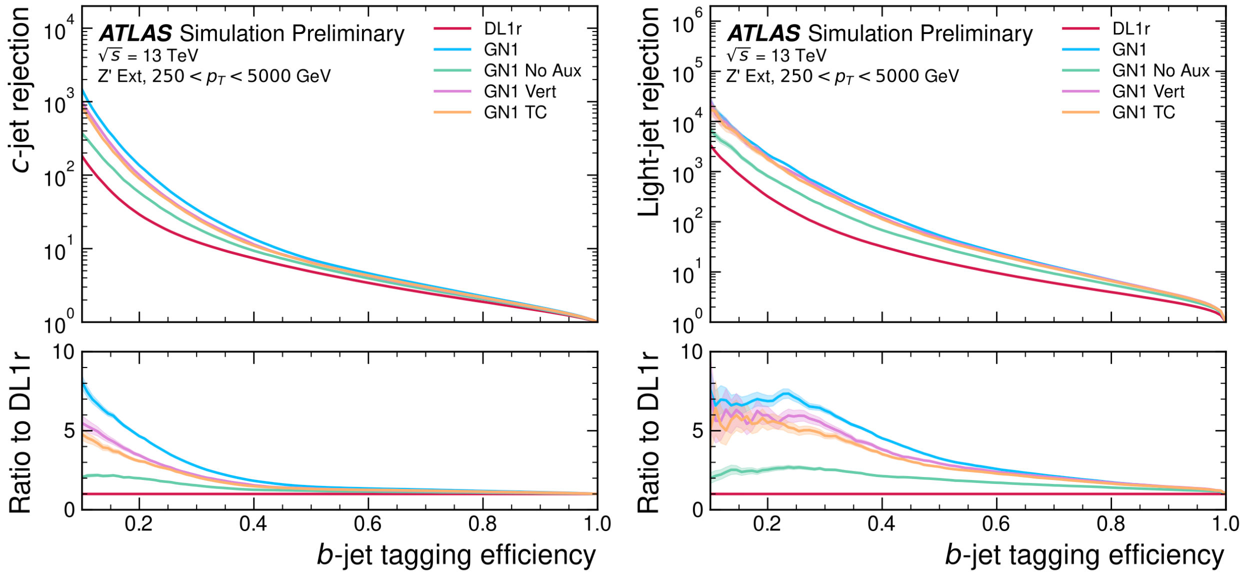
- GNNs are currently not natively supported by TMVA
- Jet flavour tagging in ATLAS has already adopted the GNN approach with some success

SALT

ATLAS official flavour tagging framework

Most of the tools needed for using GNN on the analysis are already availible on SALT.

Maybe we can borrow some of their tools.



The flavour tagging community has observed some improvements after moving from deep learning to GNN.