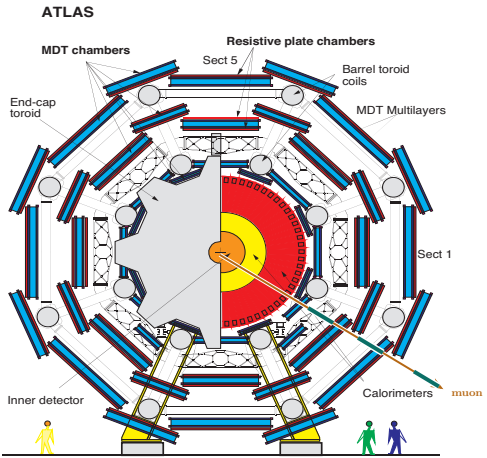
A Graph Neural Network for Muon Segment Association at the HL-LHC

CERN openlab Summer Student Programme Lightning talks

Karla Lucic
Supervisors: Davide Di Croce, Johannes Junggeburth

18 / 08 / 2026

A muon leaves a short track segment in each station



- Reconstructs the muons escaping the *pp* collision
- 16 sectors in azimuth · 3 stations (inner, middle, outer)
- A muon typically crosses 3 stations — more where large and small sectors overlap
- The toroid bends the muon between stations → momentum from curvature; within a station: straight-line segments
- Segments that belong together → tracks

The task

each bunch crossing at the HL-LHC piles up to 200 *pp* interactions on top of each other — hundreds of segments per event, only a handful from the muons we want

Score every candidate pair, where

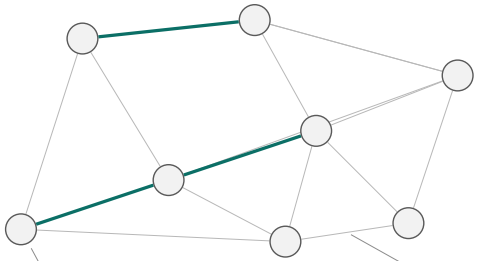
$$|s_i - s_j| \leq 1 \quad \text{and} \quad \Delta\theta \leq 35^\circ.$$

same or neighbouring sector · opening angle

Keep 99.9% of the true pairs (chosen working point), reject as much background as possible.

The event as a graph: which segments belong together?

nodes are segments · edges are pairs that *might* belong to the same muon

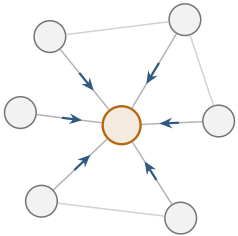


each node = one segment: 10 features
position (x, y, z) · direction (d_x, d_y, d_z)
chamber · sector · active layers
local segment count

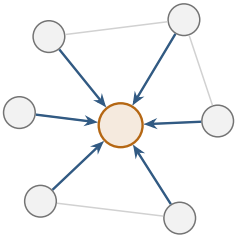
each edge = one candidate pair: 7 features
separation $\Delta\vec{x}$ and distance $|\Delta\vec{x}|$
direction alignment $\hat{d}_i \cdot \hat{d}_j$
same chamber? · same sector?

a muon leaves ∼3 segments, one per station crossed — pile-up, secondaries and reconstruction ambiguities produce the rest

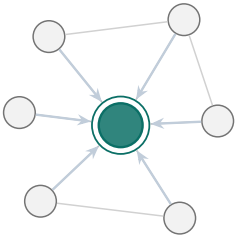
How a graph network works: message passing



1 – message: each candidate pair computes a message — a small learned function of the two segment states and the pair geometry



2 – aggregate: each segment combines the messages arriving at it



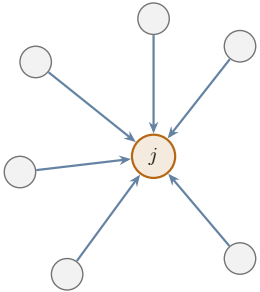
3 – update: it updates its own state, keeping what it already knew

four stacked blocks: information from up to four pairs away

The change I made: let the network weight its neighbours

Equal-size baseline

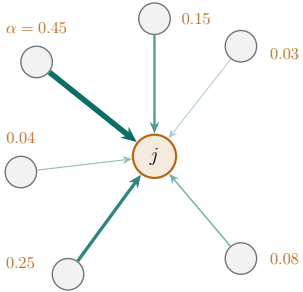
the aggregation step counts every neighbour the same



unweighted sum

This work

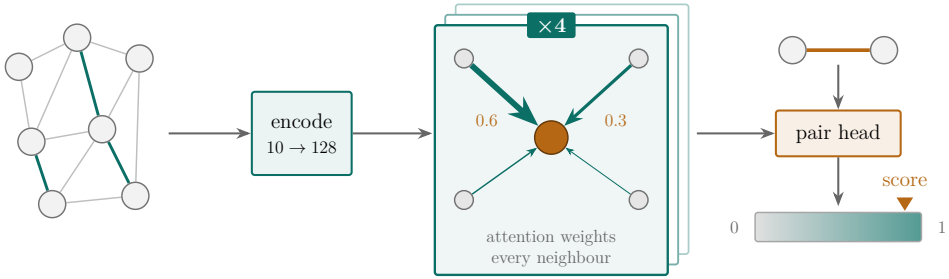
a learned weight per neighbour, recomputed in every block



learned weighted mean

the messages are learned in both — the difference is how they are aggregated

The whole network, end to end



the event graph
10 features per segment, 7 per pair

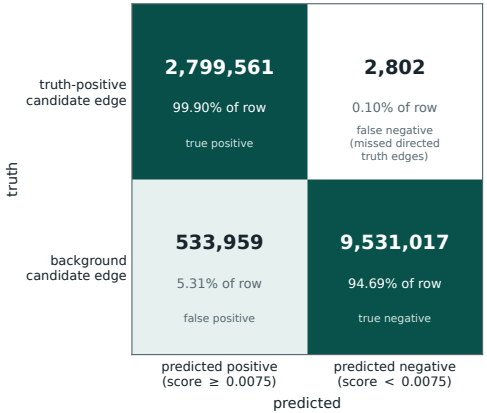
each segment becomes a 128-number description
Fourier features + MLP

four attention blocks of weighted message passing
pair geometry (7 features) enters every block

one number per pair: same muon?
from both segment states + the pair geometry a_{ij}

485,761 parameters

Result: 94.69% of background rejected, 99.9% of true pairs kept



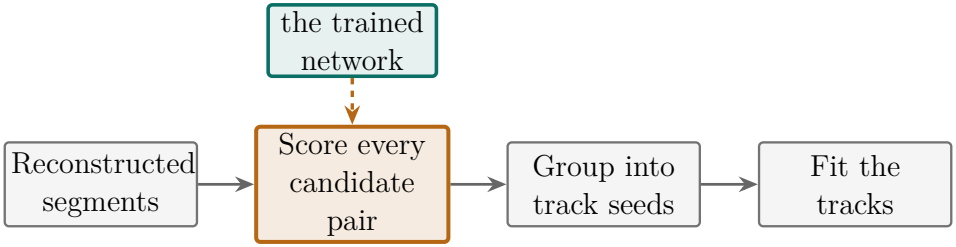
directed truth-positive vs. background candidate edges · full validation split, 275,912 events

Does the attention help?

Equal-size baseline — same data, budget, seed · development sample:

	attention	plain sum
val. loss	0.1767	0.1911
F_2 score	0.9447	0.9379
parameters	485,761	483,713

In ATLAS: segments in — groups from the same muon out



teal: built in this project

amber: trained once, then run on every event

Matches training to 1×10^{-5} on identical inputs · full algorithm in Athena: 26.6 ms/event on one CPU thread at $\langle \mu \rangle = 200$ (graph building + inference + candidates)

Integration also uncovered two input mismatches — fixes validated

Summary

Cleaner track seeding: 94.69% of background pairs rejected at 99.9% efficiency.

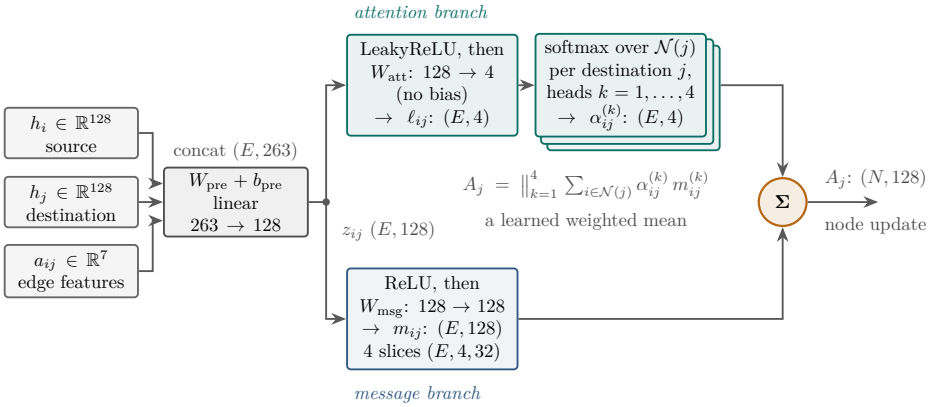
Learned attention weighting in place of an unweighted sum — beat the equal-size baseline (same budget, development sample).

Runs in ATLAS reconstruction — fixes for two detector-input bugs validated.

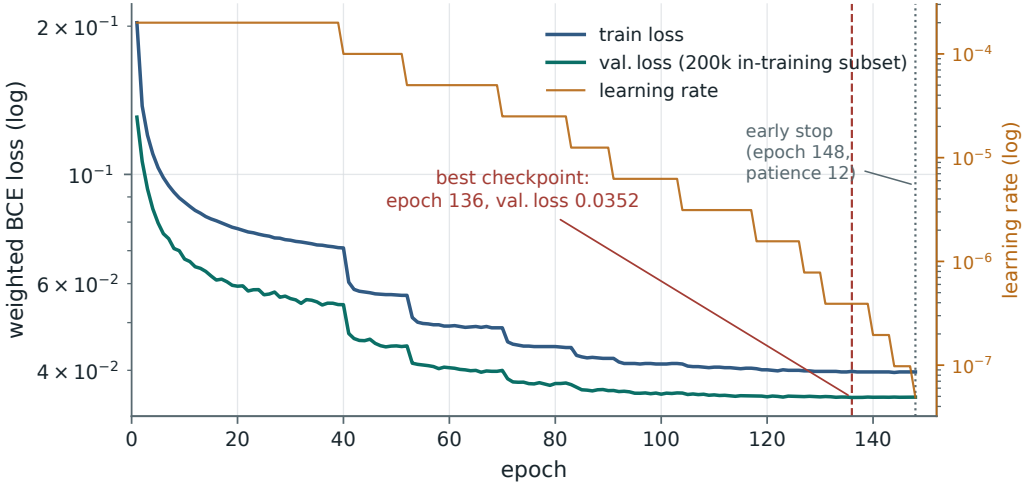
Thank you!

Karla Lucic · ATLAS Muon Software · CERN openlab

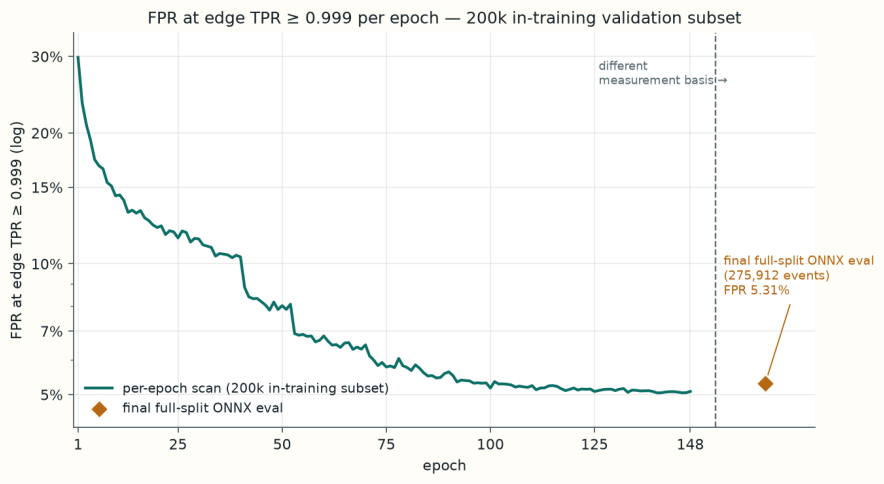
Backup: inside the attention layer



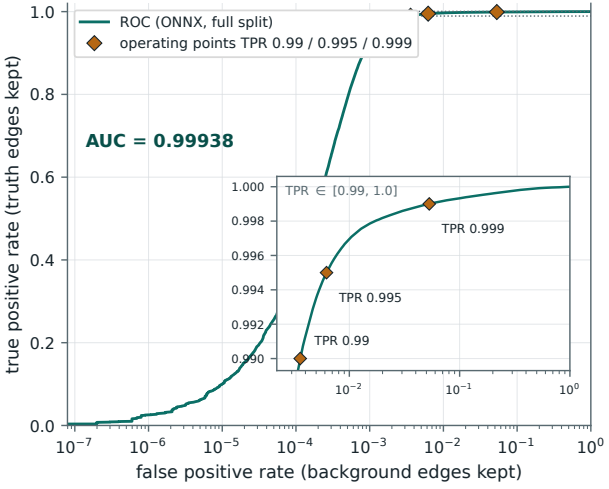
Backup: trained on 2.5 million event graphs



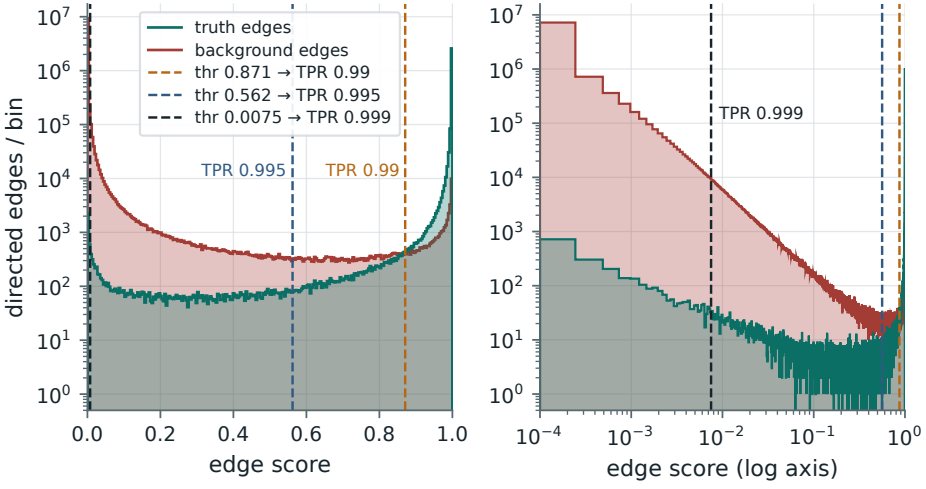
Backup: background kept at 99.9% efficiency, epoch by epoch



Backup: the ROC curve



Backup: real pairs pile up at 1, background at 0



Backup: the two architectures side by side

	published-style	this work
message-passing steps	10 (5 blocks of 2)	4 attention blocks
width of a segment state	256	128
how neighbours combine	unweighted max	learned weighted mean
pair geometry enters	only at the end	every block
parameters	2,237,185	485,761

- Several things differ at once — not a controlled comparison, and the size difference carries no performance claim
- The controlled, equal-size comparison against a plain-sum baseline is the next slide

Backup: alongside the published result

	this work	published classifier
real pairs kept	99.9%	99.9%
background rejected	94.69%	99.5%
parameters	485,761	2,237,185
validation sample	dimuon events	single muons + top-quark events
validation events	275,912	359,889

- Different samples (different background mix), different model size, different software state — not a like-for-like ranking
- A rerun of this model on the published validation sample is future work

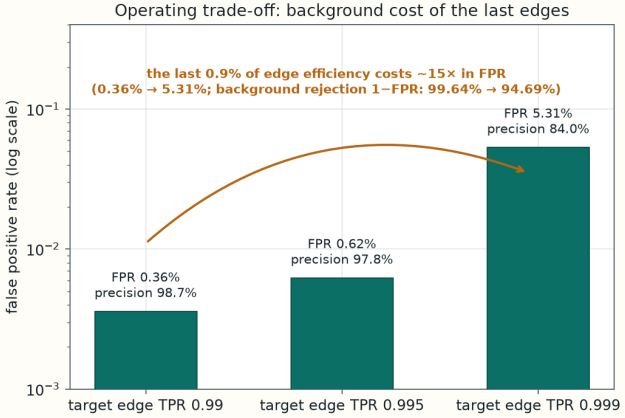
Published numbers: ATL-DAQ-PUB-2026-005

Backup: does attention pay its way?

layer	parameters	val. loss	F_2
attention (this work)	485,761	0.1767	0.9447
MPNN baseline (plain sum)	483,713	0.1911	0.9379

- Same width, same depth, same data, same training budget
- Attention adds only 2,048 parameters
- Single seed on a development sample — motivating evidence, not proof

Backup: the cost of the last 0.9% of efficiency



Per directed edge · full 275,912-event validation split (274,953 evaluated, 959 zero-edge events skipped) · exported ONNX model · best-epoch-136 checkpoint

Backup: dropping the opening-angle cut

per event, at 99.9% efficiency	with cut	without cut	change
background pairs offered	36.48	103.98	2.85×
false positives	1.94	4.79	2.48×
pairs kept	12.08	14.99	+24.1%
real pairs available	10.16	10.21	+0.55%

- Removing the opening-angle cut offers the model 2.85× more background
- It absorbs most of it: false positives rise only 2.48×, and it finds 0.55% more real pairs that the cut had been throwing away
- The two campaigns have different denominators — not a head-to-head ranking

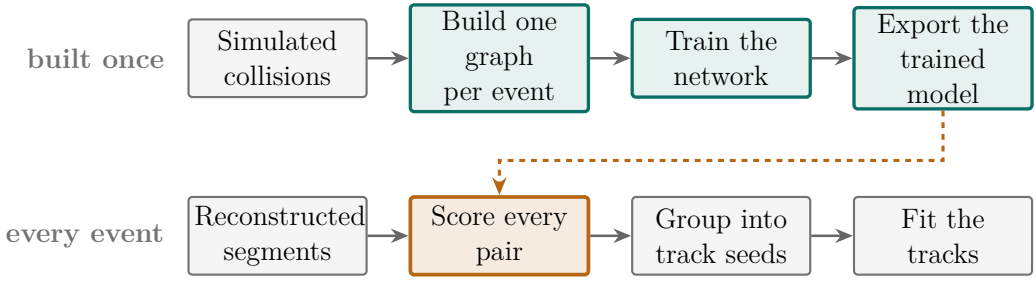
Backup: what it costs in the real chain

ms per event	edge classifier	bucket filter	seeding + fitting	whole job
no ML	—	—	839.2	2553
with cut	26.6	759.5	852.7	3668
without cut	44.4	768.7	830.4	3496

- Dropping the cut costs only +18 ms per event at the edge classifier despite 2.85× the background pairs — strongly sublinear
- The bucket filter dominates the ML chain's added cost over the no-ML baseline — not the edge classifier
- Seeding and fitting agree across all three arms: in this build the seeder never consumes the scores, so these numbers isolate the classifier chain and show no downstream saving

500 events at $\langle\mu\rangle = 200$, single thread, same machine back to back, one run per arm; algorithm-level entries, possibly nested — not an additive breakdown.

Backup: how the model is built and shipped



Sector 16 got no incoming candidate edges

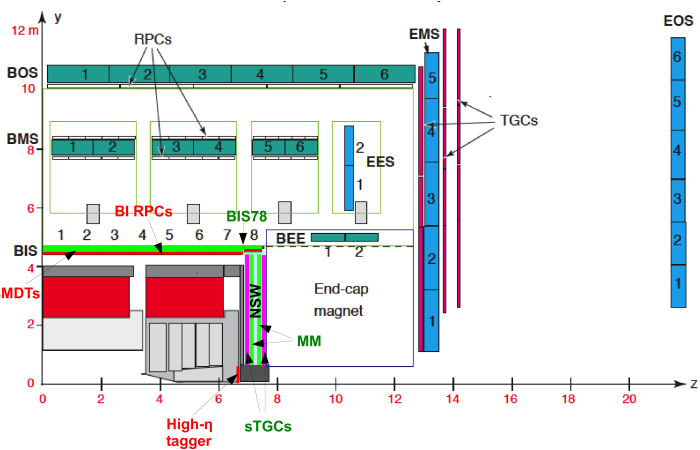
A wrap-around mistake in the graph building: 0 of 60 sector-16 segments were connected, 59 of 60 after the fix.

One feature meant two different things

The detector-layer count was defined differently in training and in reconstruction — off by roughly a factor two.

The layer-count fix is validated and ready. The sector fix changes which edges exist, so it needs the converter corrected, the data regenerated and the model retrained.

Backup: side view — the plane where the muon bends



One quadrant, beam along z · ATLAS Collaboration, JINST 19 (2024) P05063 — Fig. 1 of ATL-DAQ-PUB-2026-005

Slide 2 figure: ATLAS Collaboration, Eur. Phys. J. C 70 (2010) 875 (CC-BY)