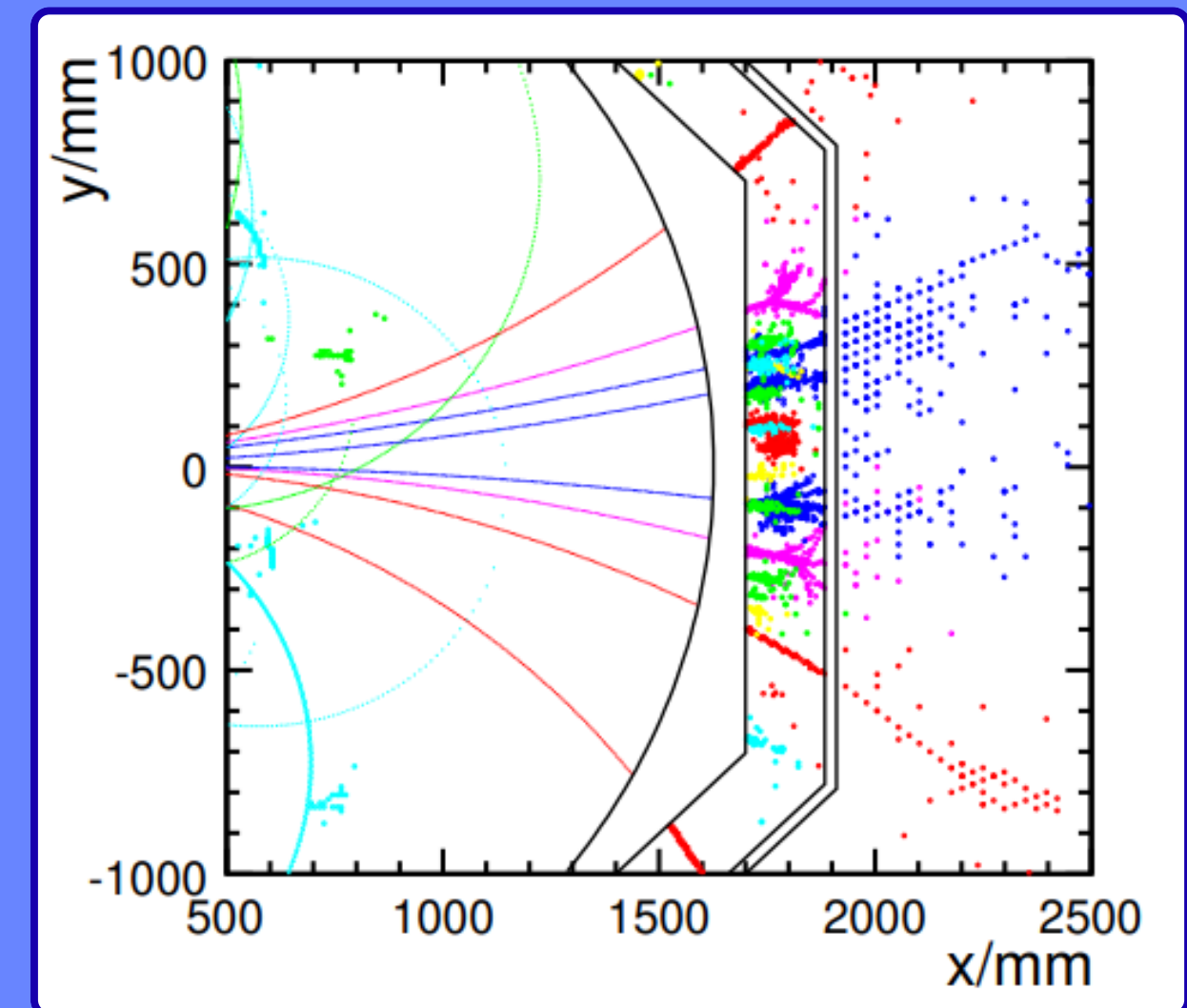


ILD Software and Analysis meeting

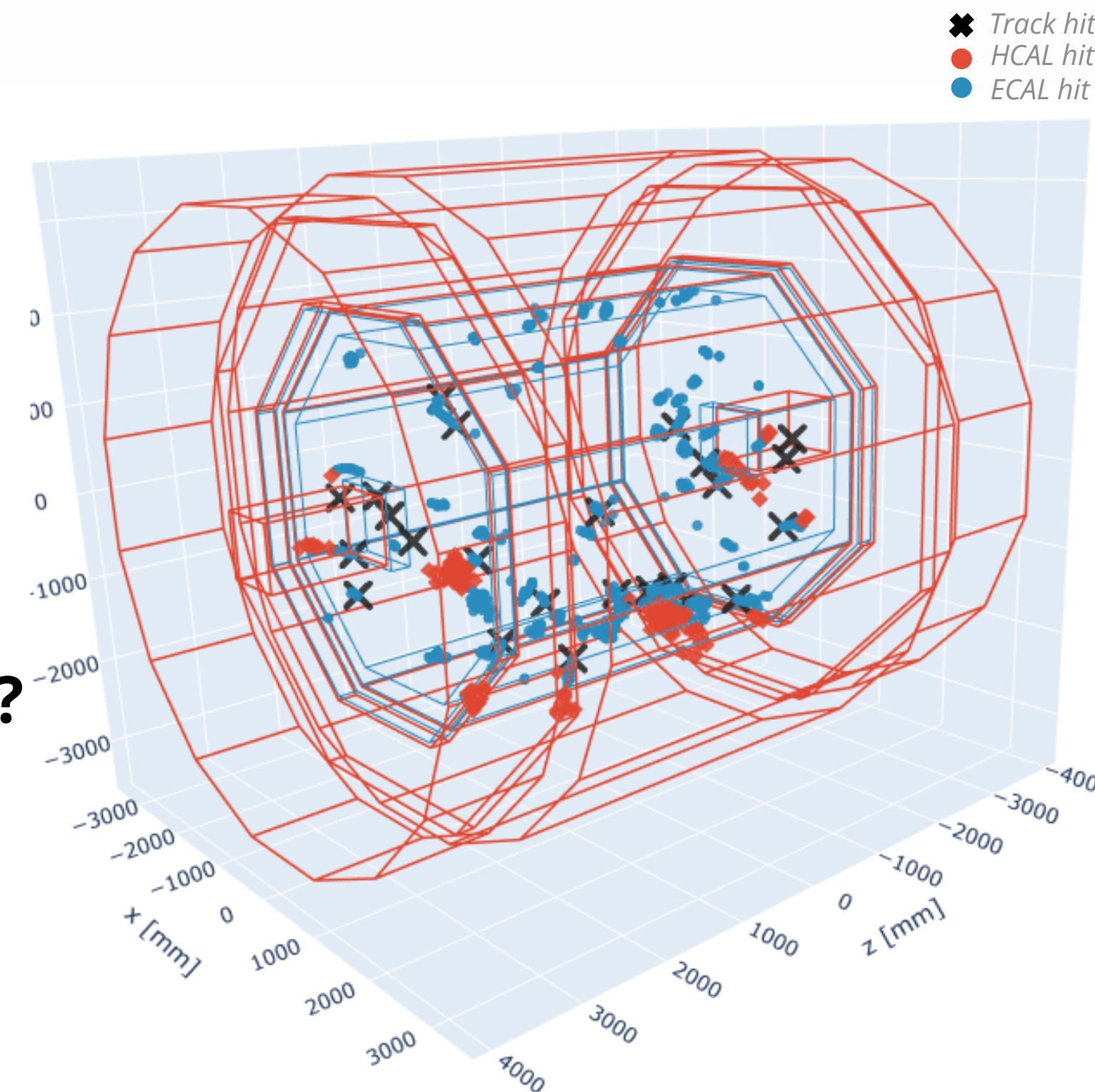
Co-authors : Cedric Trinh, Tastuki Murata, Taikan Suehara

*Time implementation for Particule Flow
Reconstruction for the ILD Using Graph
Neural Networks and Object
Condensation*



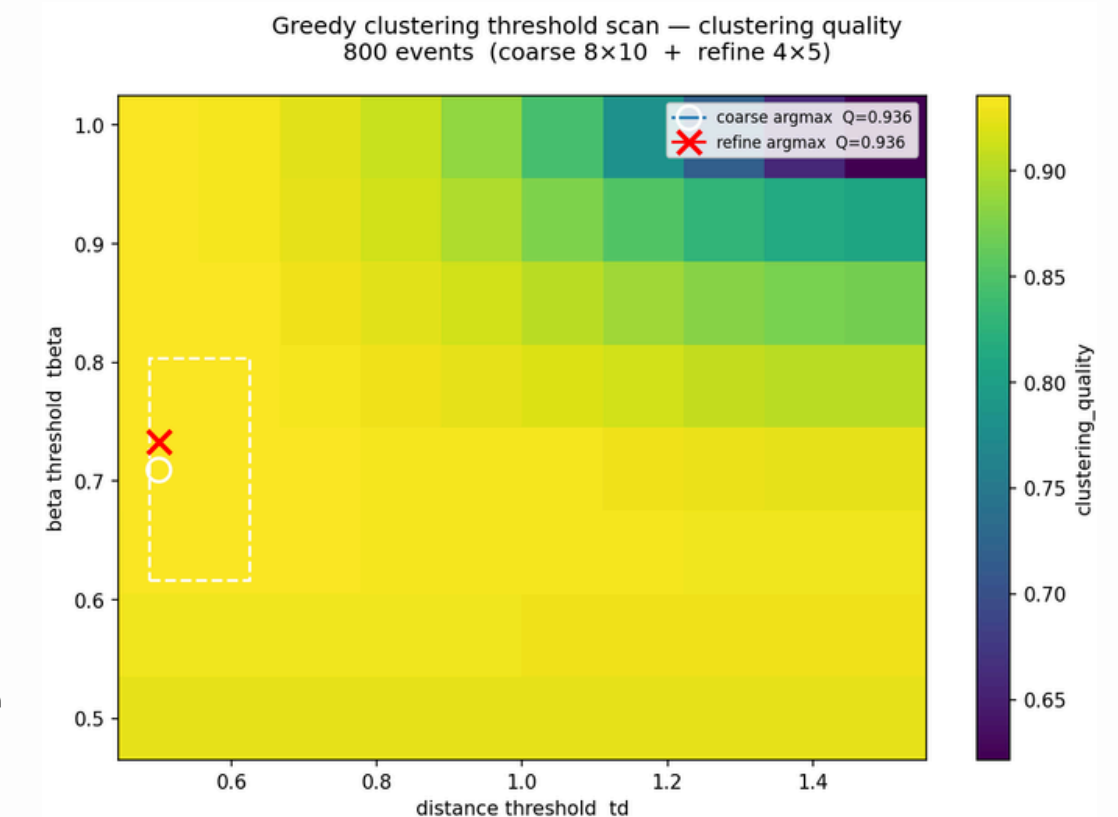
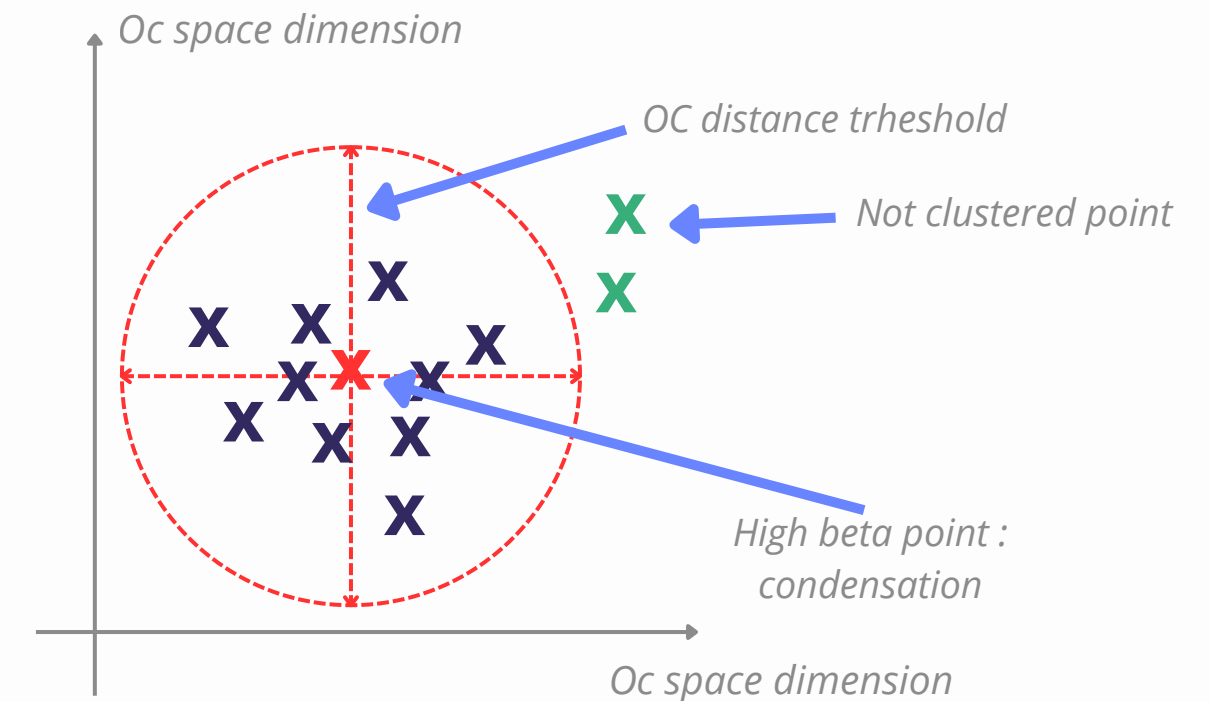
Context

- Detectors's data :
 - Track hits → Extrapolated position
 - Time of calorimeters-hits (*perfect*)
 - Not used in Pandora
 - **Easier implementation in ML ?**
- How to implement?
- Smearing/robustness of the approach



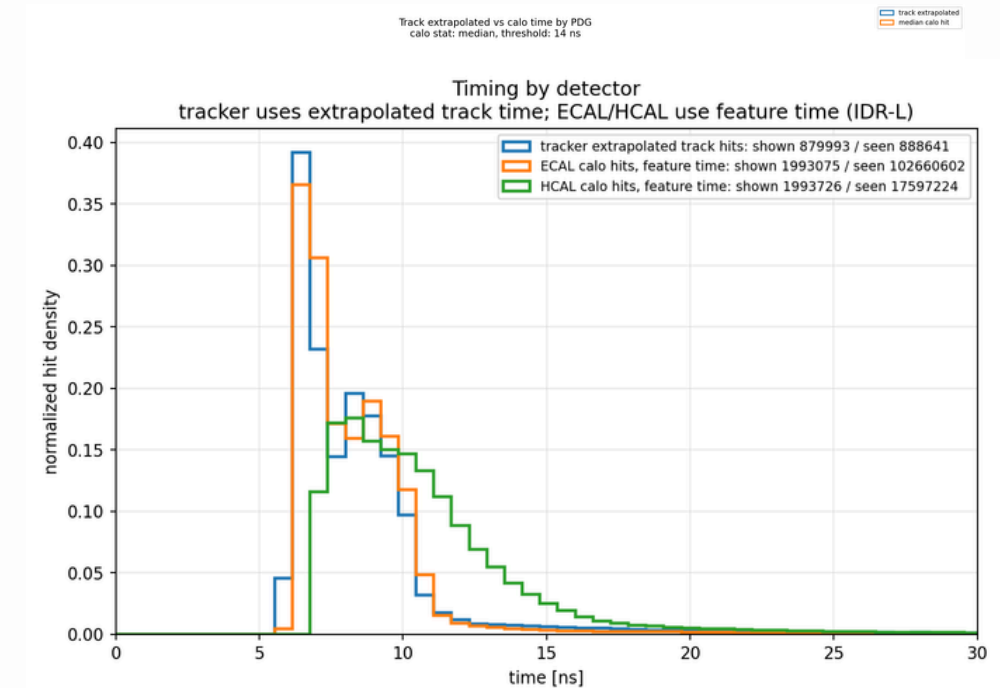
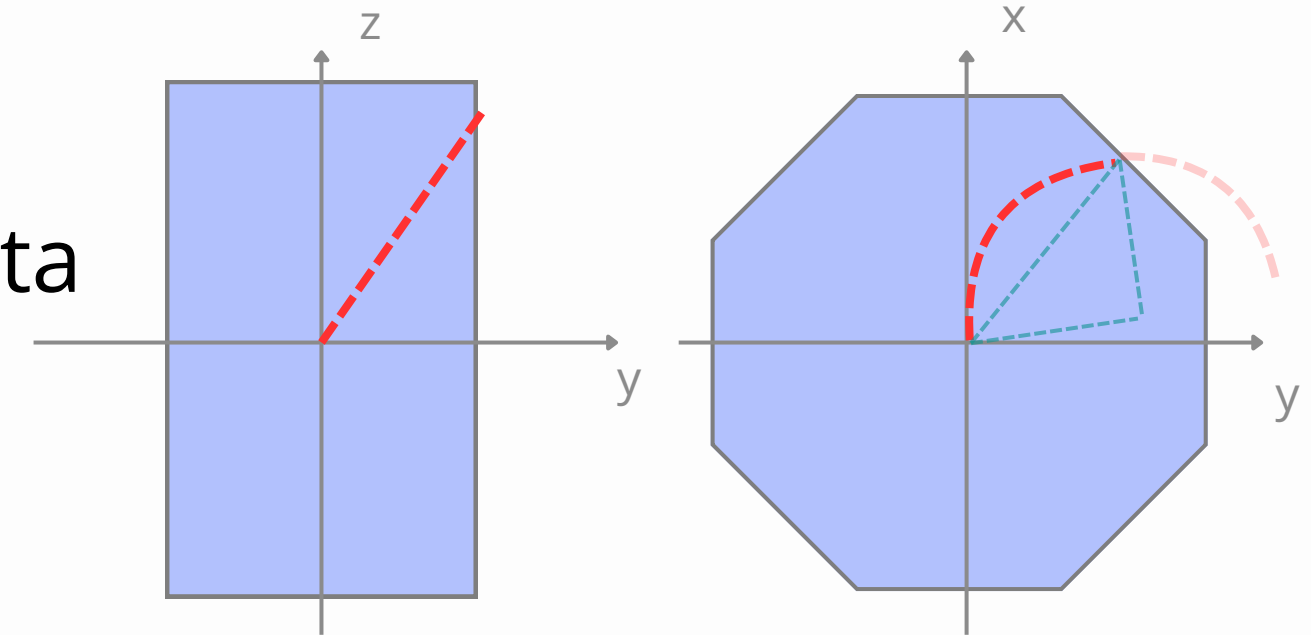
Model and clustering

- Main hyperparameters
 - 6 gravnet layers
 - oc space dimension : 6
- Naive clustering with beta & oc distance threshold (scan over validation)
 - Track hits forced to be seed
- Evaluation
 - Efficiency : rightly clustered/true hits
 - Purity : rightly clustered/ cluster's hits
 - Confusion : charged to neutral and opposite



Preprocessing

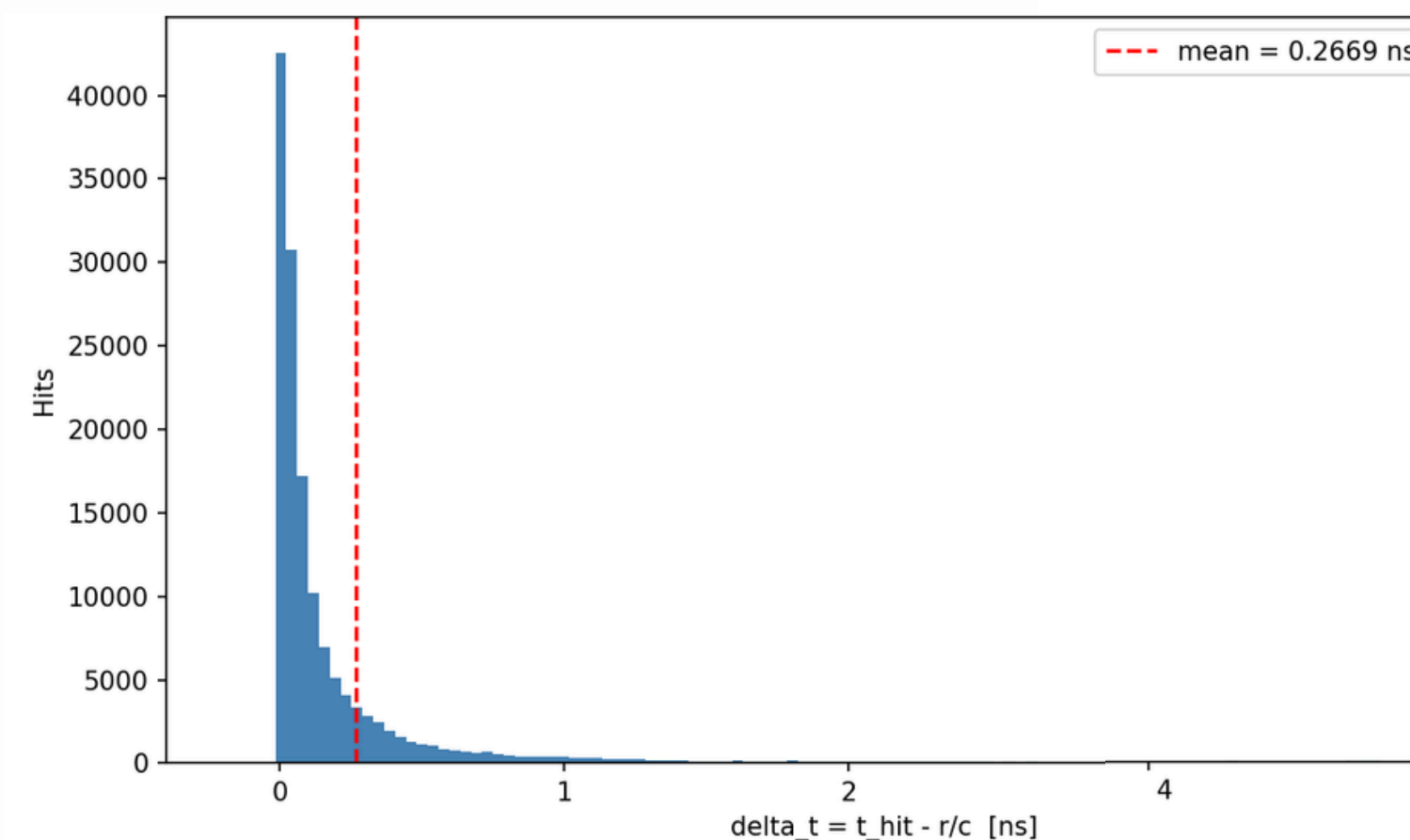
- Track hit : Final position of tracker with momenta
 - no time information → is set to 0
 - Artificial distance between calo hits
 - But is seed for clusters !
- Give an extrapolated/approximate time
 - From final position, momenta and geometry
 - $t = 2 \cdot \frac{|p|}{qBc} \arcsin \left(\frac{qBr}{2p_{\perp}} \right)$
- Make sure of the coherence of the data created



Attempted approach for time implementation

- Naive approach
- Time of flight (TOF)
 - $\Delta t = t_{hit} - \frac{r_{hit}}{c}$
 - Naturally normalised
- Transformer like approach
 - Time specialised embedding
 - $$\begin{cases} v_{2i} = \sin\left(\frac{t}{norm^{2i/D}}\right) \\ v_{2i+1} = \cos\left(\frac{t}{norm^{2i/D}}\right) \end{cases}$$

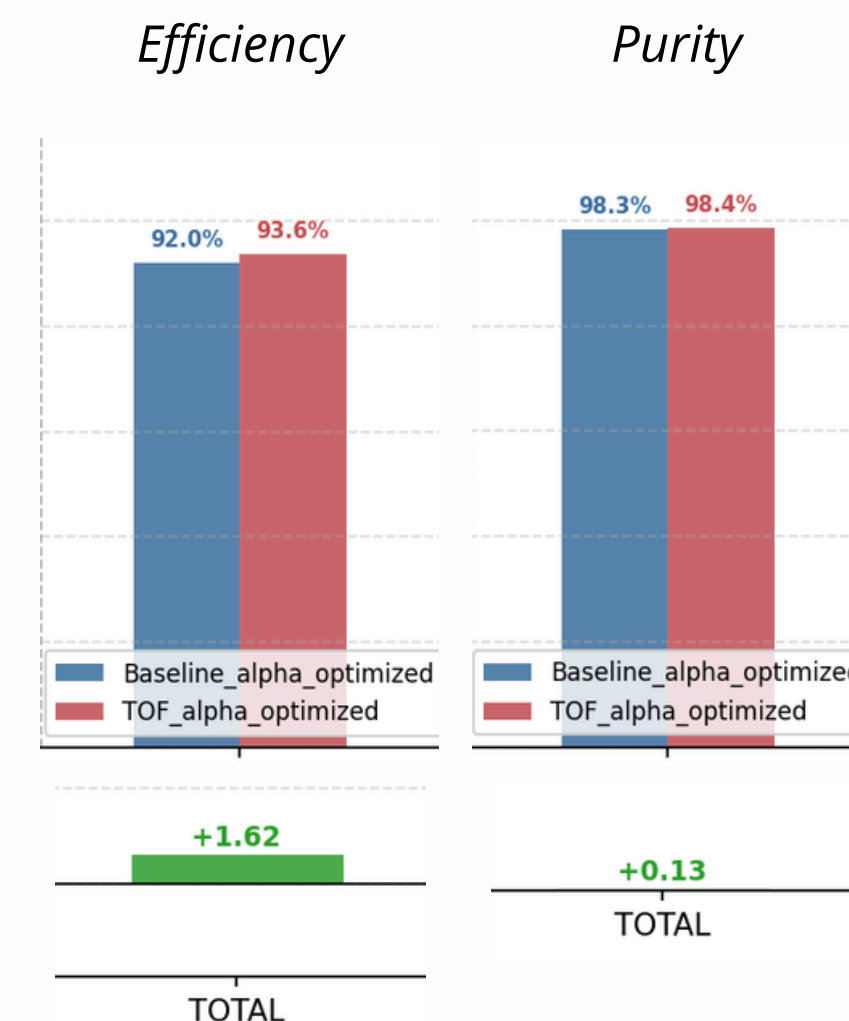
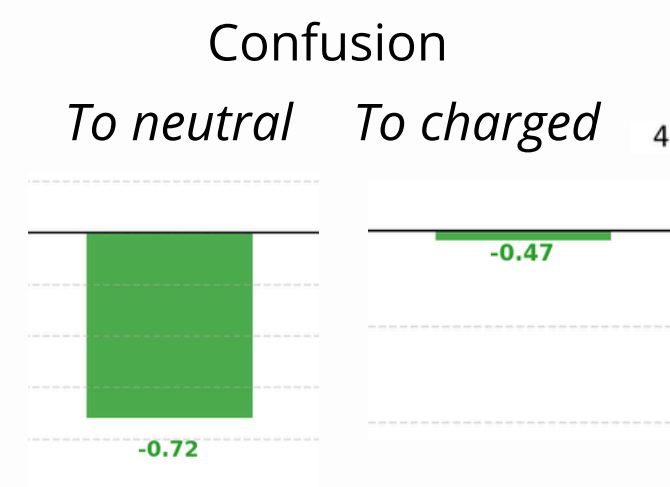
Distribution of delta_t (all hits) data: ['data', 'tc', 'tc_ntau_10GeV_10', 'train', 'ntau_10GeV_10_0.h5']



First training - Clean dataset *Tau decay*

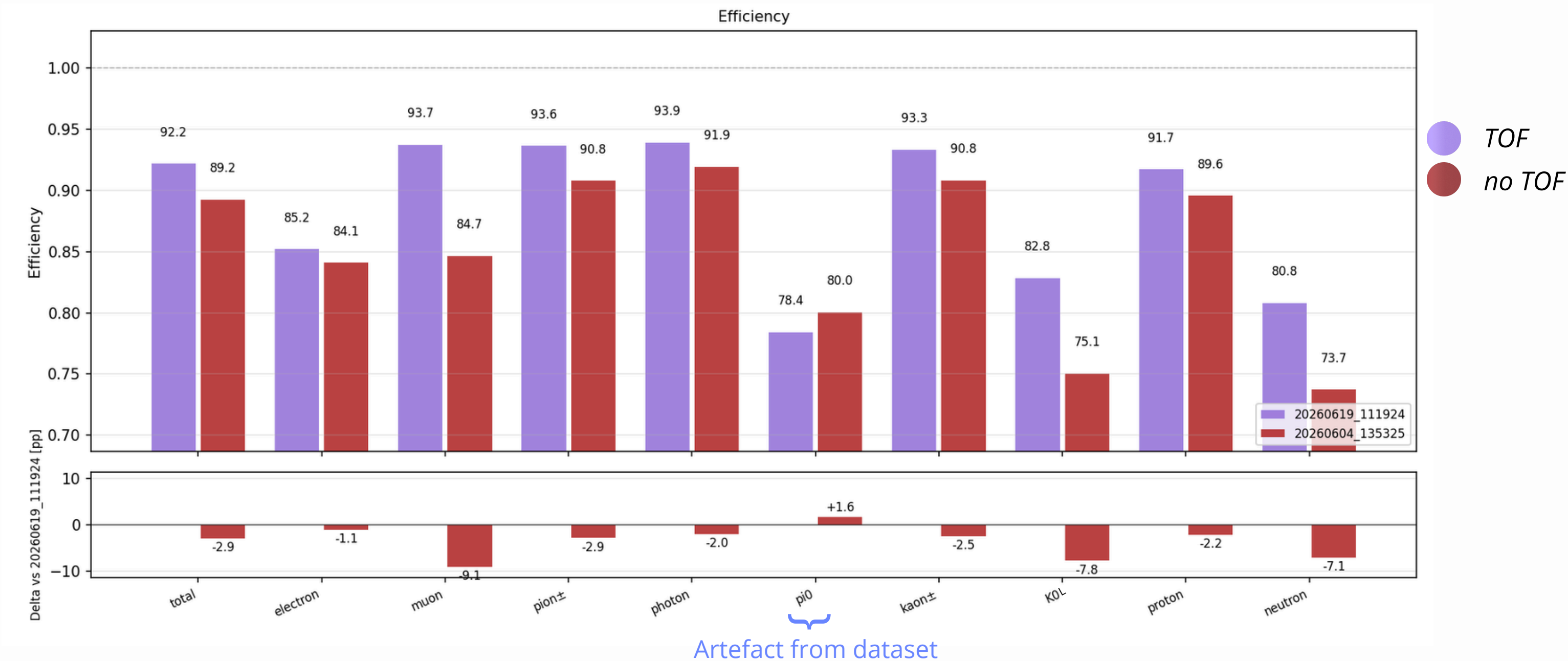
- Naive approach
 - Flodded the network
 - Worst overall performance
- Transformer like approach
 - Hard time stabilising
 - Worst overall performance
- Time of flight (TOF)
 - Promising results

→ Focus on TOF implementation

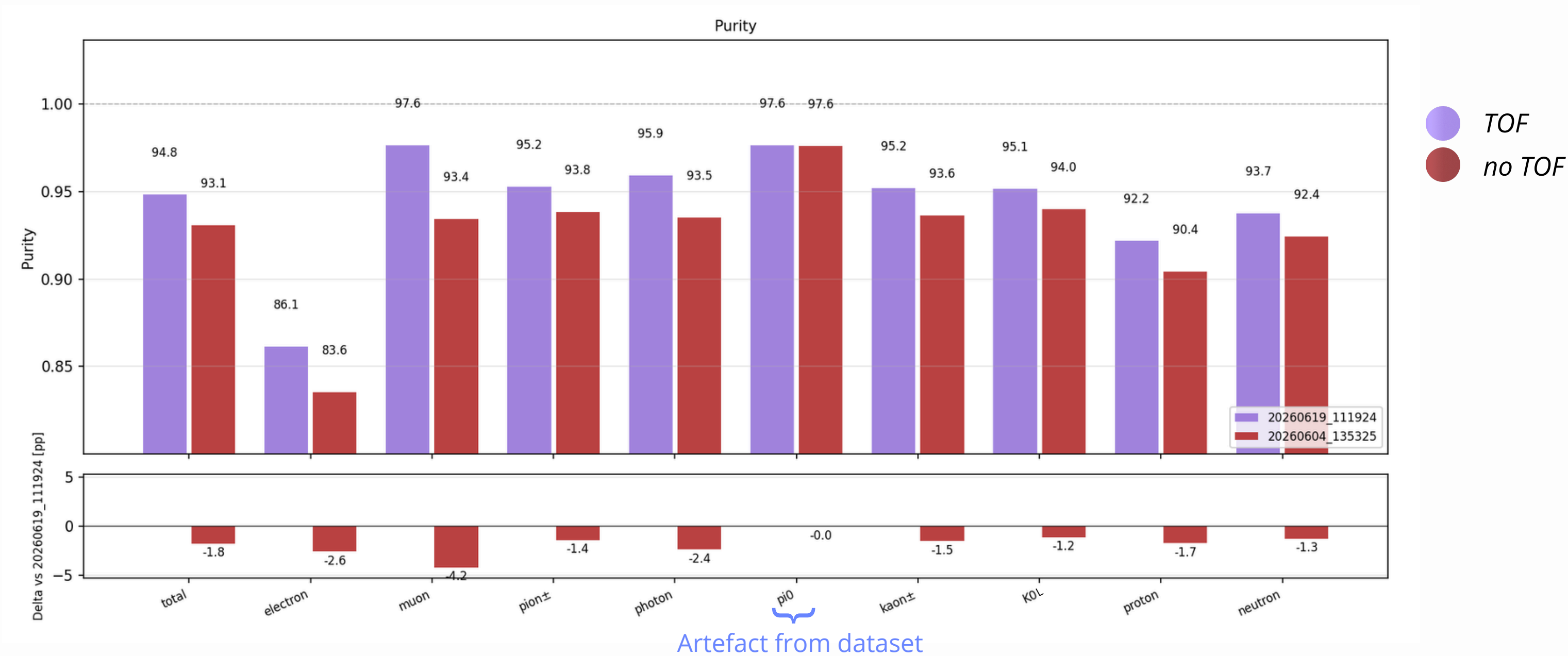


(100k events, 10 GeV x 10 taus / event → 1 million taus)

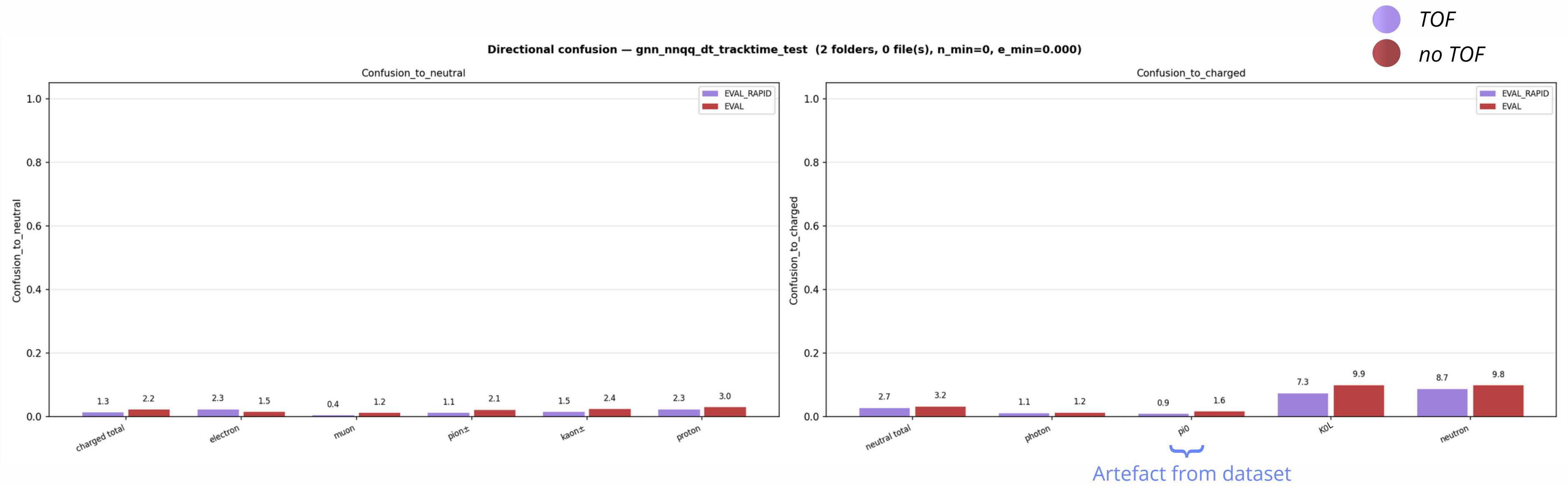
Confirming results - Dirtier dataset *$\nu\nu qq$ from Higgsstrahlung*



Confirming results - Dirtier dataset *vvqq from Higgsstrahlung*



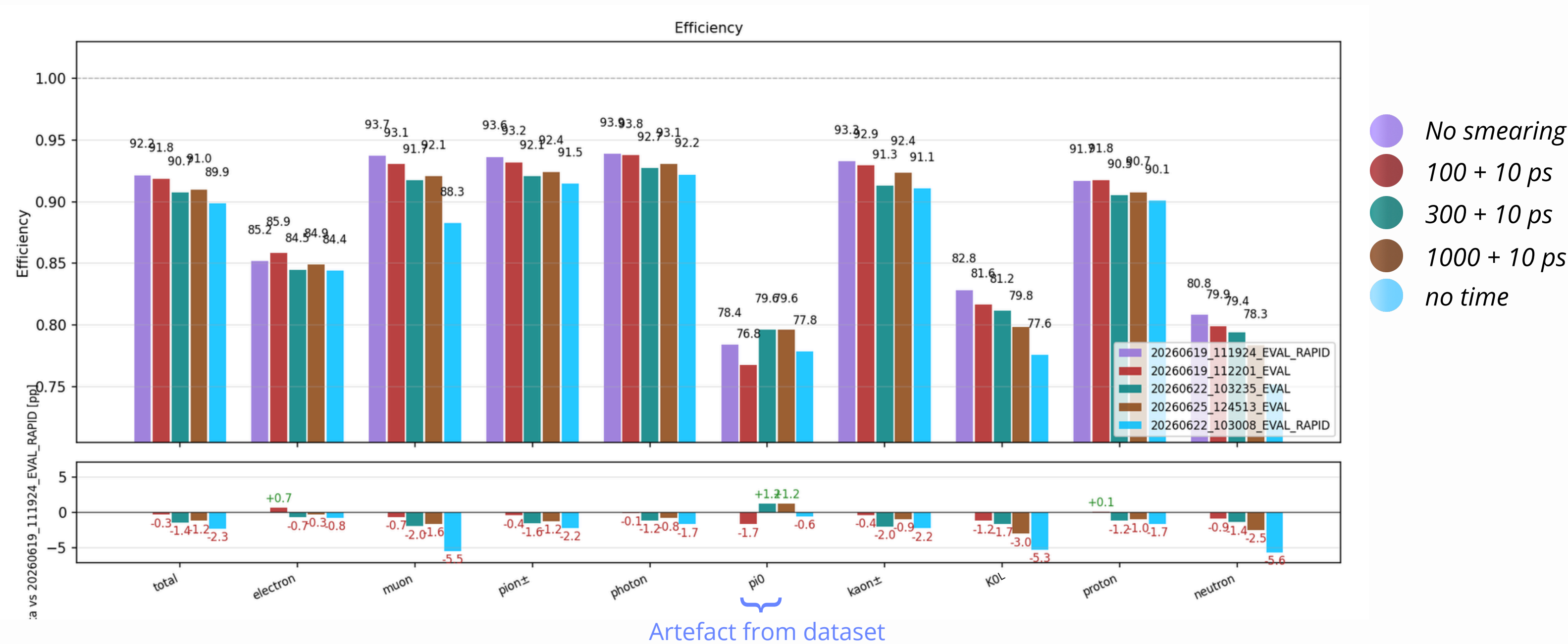
Confirming results - Dirtier dataset *vvqq from Higgsstrahlung*



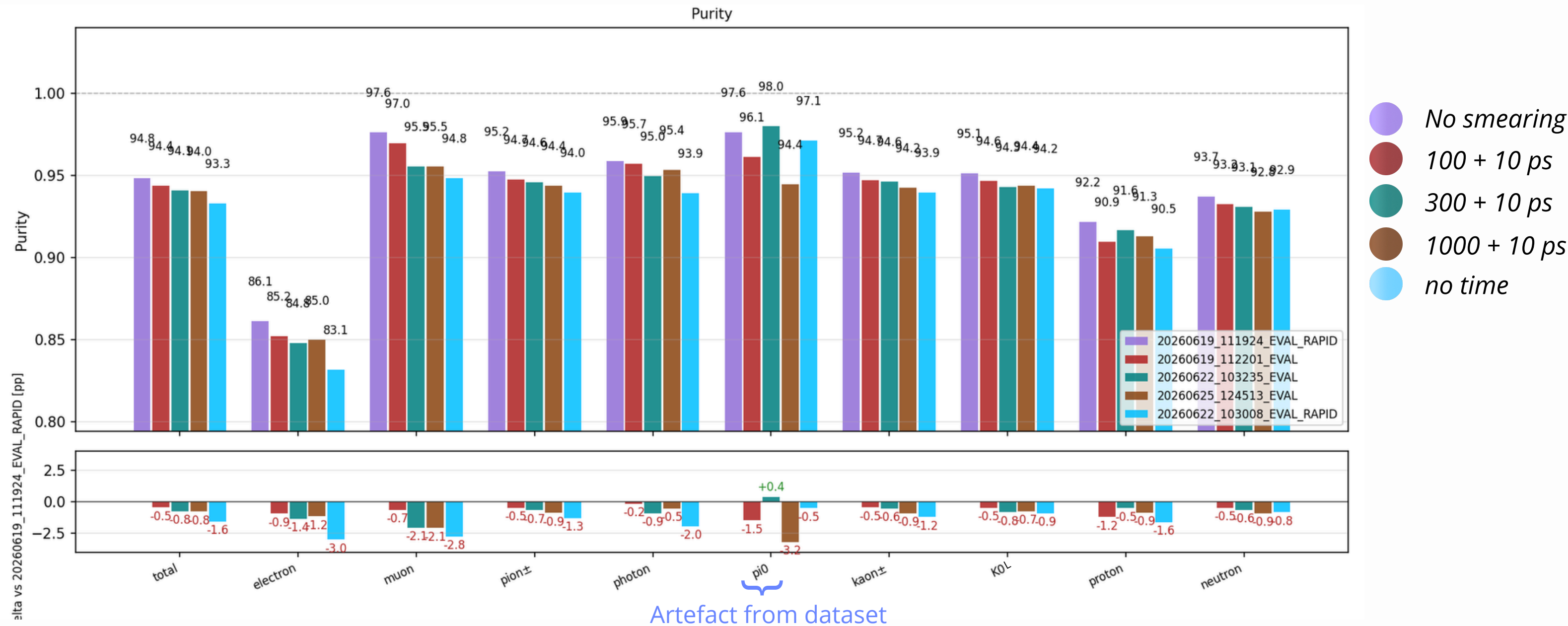
Robustness - Smearing

- Simulate uncertainties
 - From finite resolution of detectors: σ_t fixed
 - From electronic : σ_{cst}
- Poissonian approach
 - $\sigma_t = \frac{\sigma_{t \text{ fixed}}}{\sqrt{N_{MIP}}} + \sigma_{cst}$
 - Stochastic from 100ps to 1ns
 - Fixed of 10ps

Robustness - Smearing

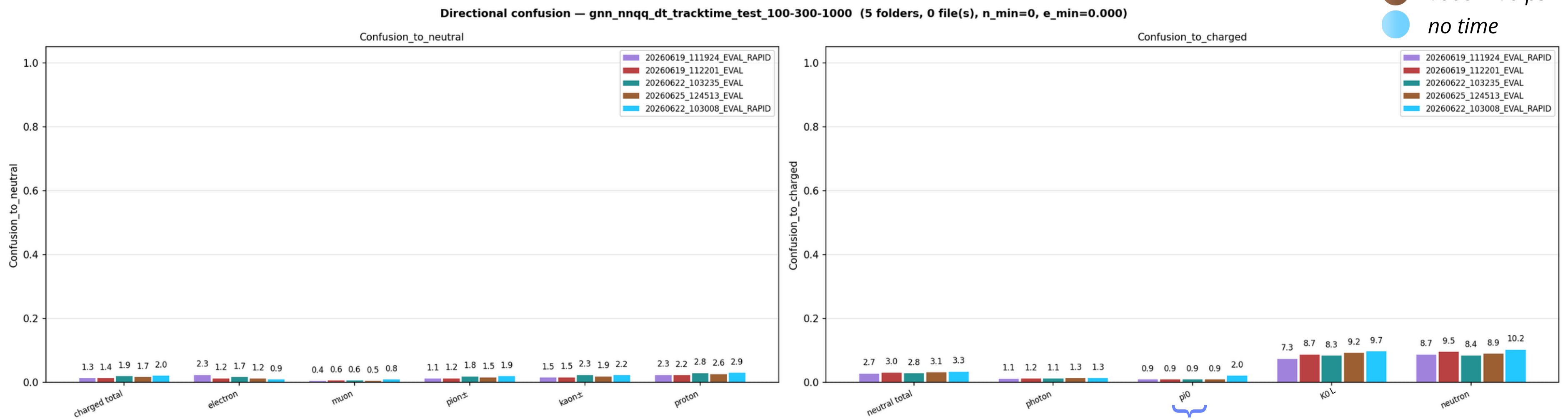


Robustness - Smearing



Robustness - Smearing

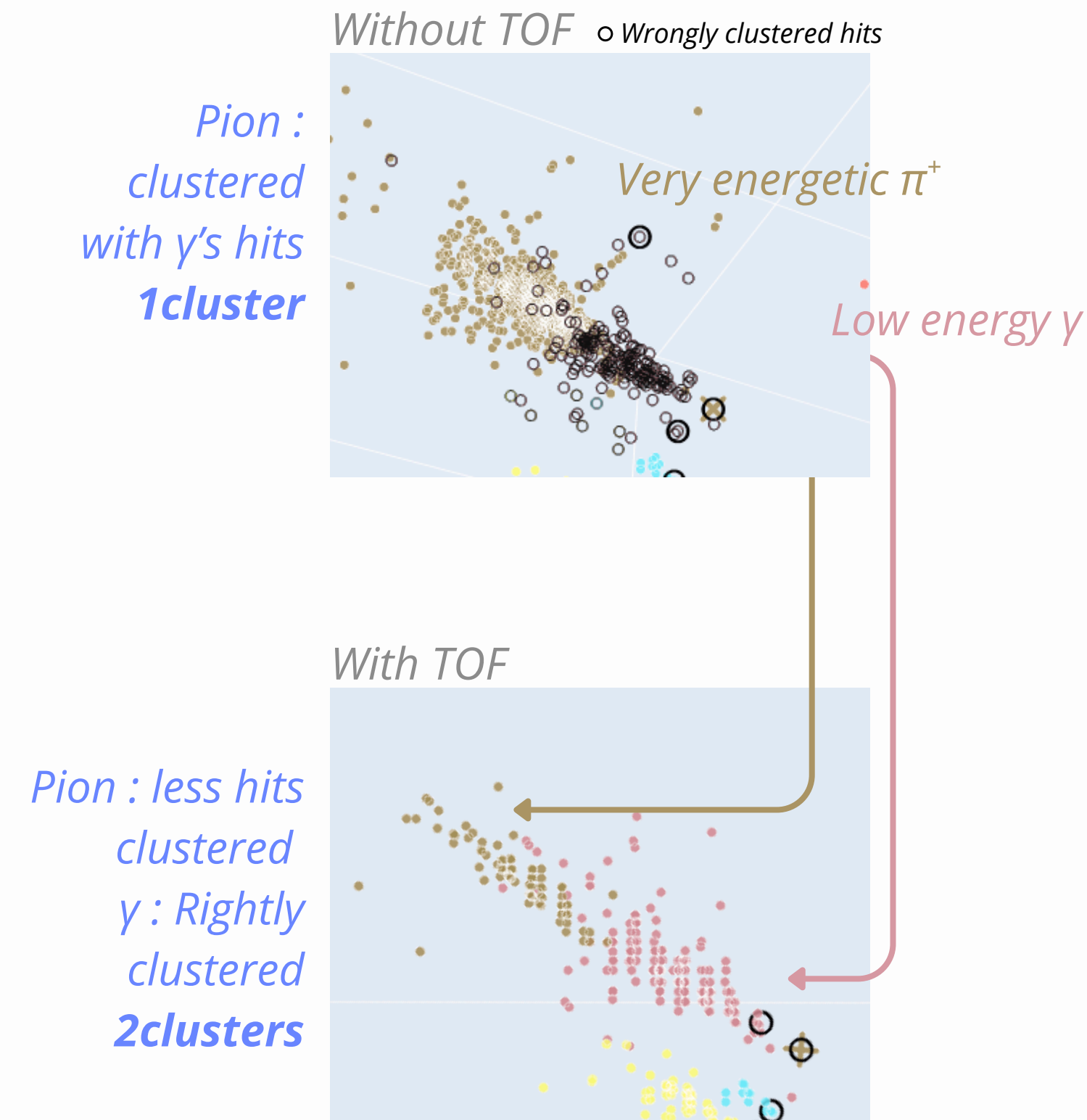
- No smearing
- 100 + 10 ps
- 300 + 10 ps
- 1000 + 10 ps
- no time



Artefact from dataset

How to influence clustering

- Improvement
 - top 10 events : improvement over average 14GeV particle
 - reunified cluster majorly impactfull
 - Regression
 - top 10 worst events : average particle momenta 32,4 GeV
 - fragmentation introduced over high energy particle
- Expected behaviour



Conclusion

- Time information usage
 - Gain in performance : (100+10) ps $\rightarrow$ *+2% eff ; +1.1% pur ; (-0.7 ; -0.3)% conf*
 - Easier with ML approach
- Perspectives
 - Mix TOF information with energy related information
 - Not clearly defined variables...
 - Optimize model architechure for TOF :
 - Hyperparameters optimisation