Machine Learning in Baikal-GVD: Noise Filtering and EAS Suppression via Neural Networks with Domain Adaptation

This work was supported by the Russian Science Foundation under grant no. 24-72-10056.

Matseiko Albert
MIPT, INR RAS
matseiko.av@phystech.edu

Quarks-2026, May

Outline

0. *Baikal-GVD experiment: brief description*

1. *Baikal-GVD ML processing pipeline*

2. *Developed NN models: metrics and real data validation*
   - *Background EAS Events (pre)filtering*
   - *Signal Hits Filtering*
   - *ν-candidate extraction*

Baikal-GVD

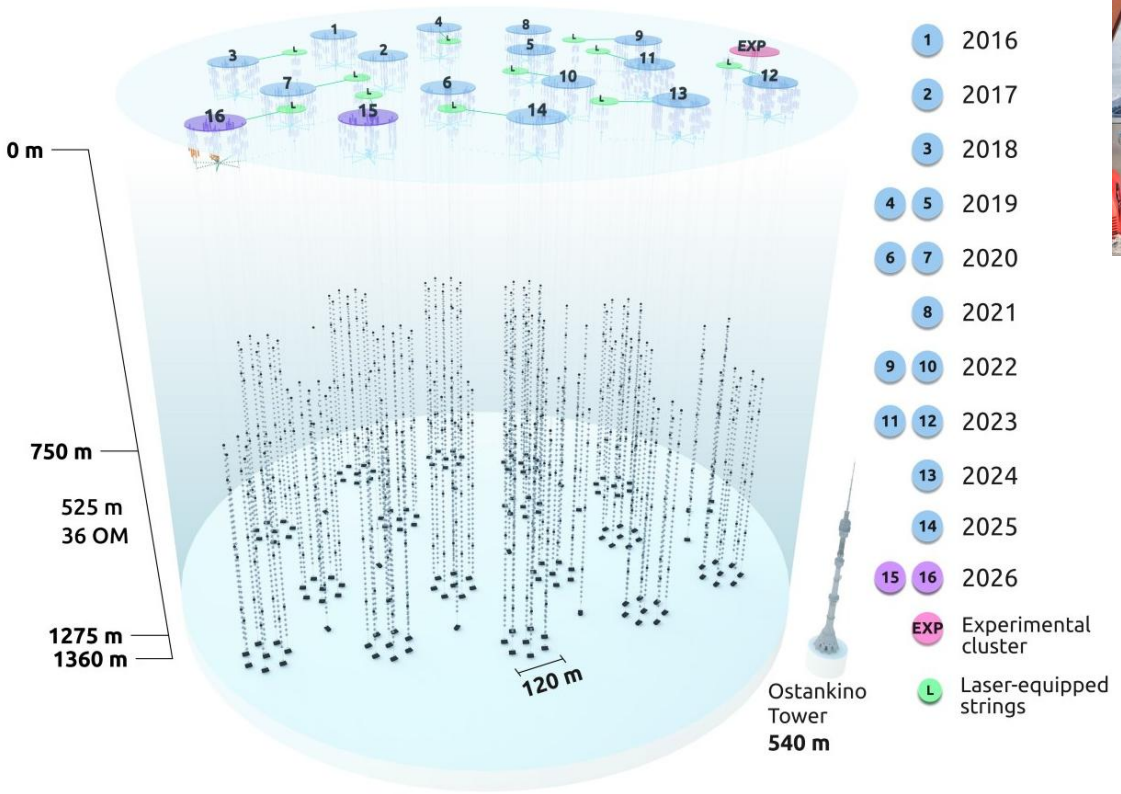
Setup process

Optical modules

Form «strings»

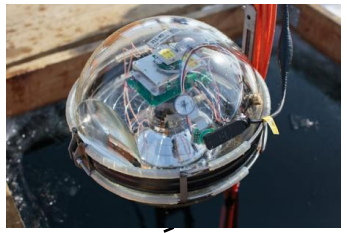
→

Strings form «clusters»



>16 clusters are setup by 2026

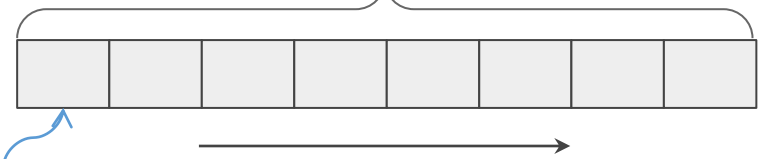
Event representation for NN



Single cluster event
(not multicluster yet)

Causal (time-ordered)

Hits



1 **hit**
Q, t, x, y, z

ordering by time t

Transformer NN

Uses *self-attention mechanism*

-->

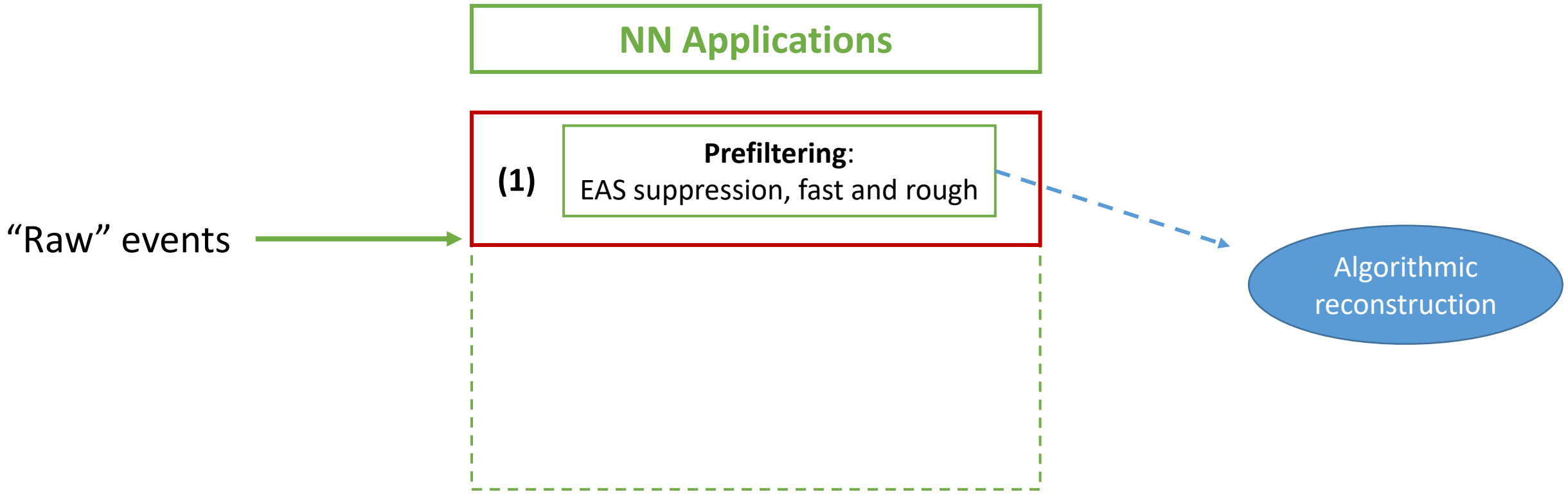
Finds important relations between individual hits

-->

Helps to retrieve sophisticated dependencies

Training with **MC-simulated** data

ML processing pipeline



(1)

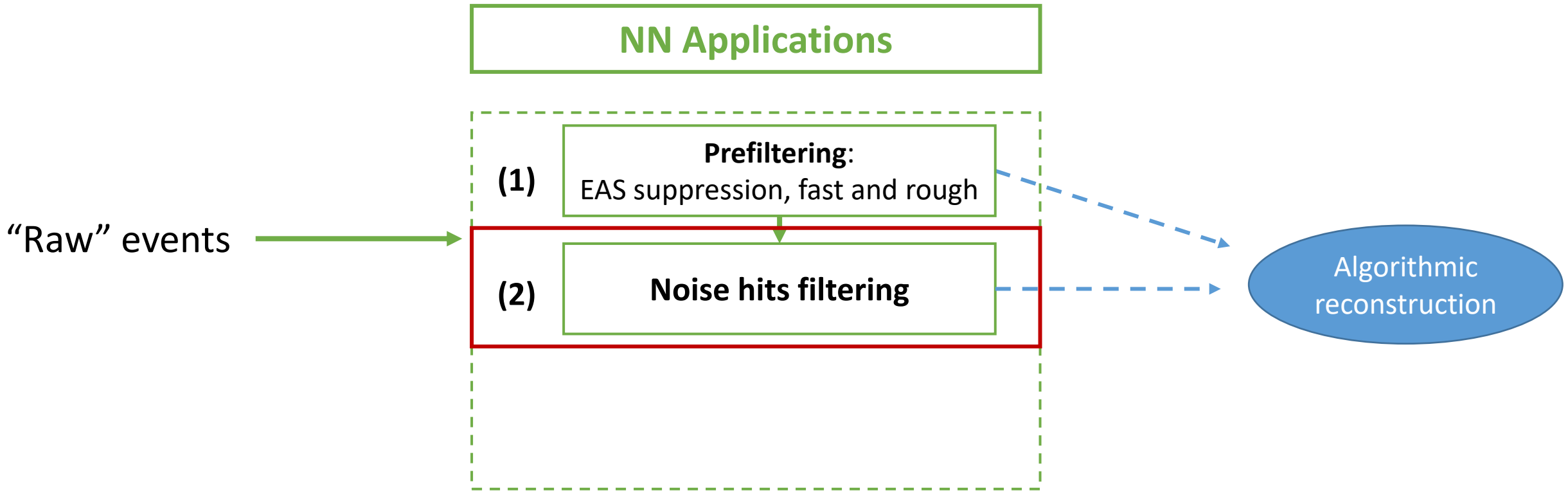
Problem: 1 ν per 10^6-10^7 events.

NN solution: Prefilter Model

suppresses background by factor 1000 while keeping 99% ν-events

Also may speed up standard reconstruction!

ML processing pipeline



(2)

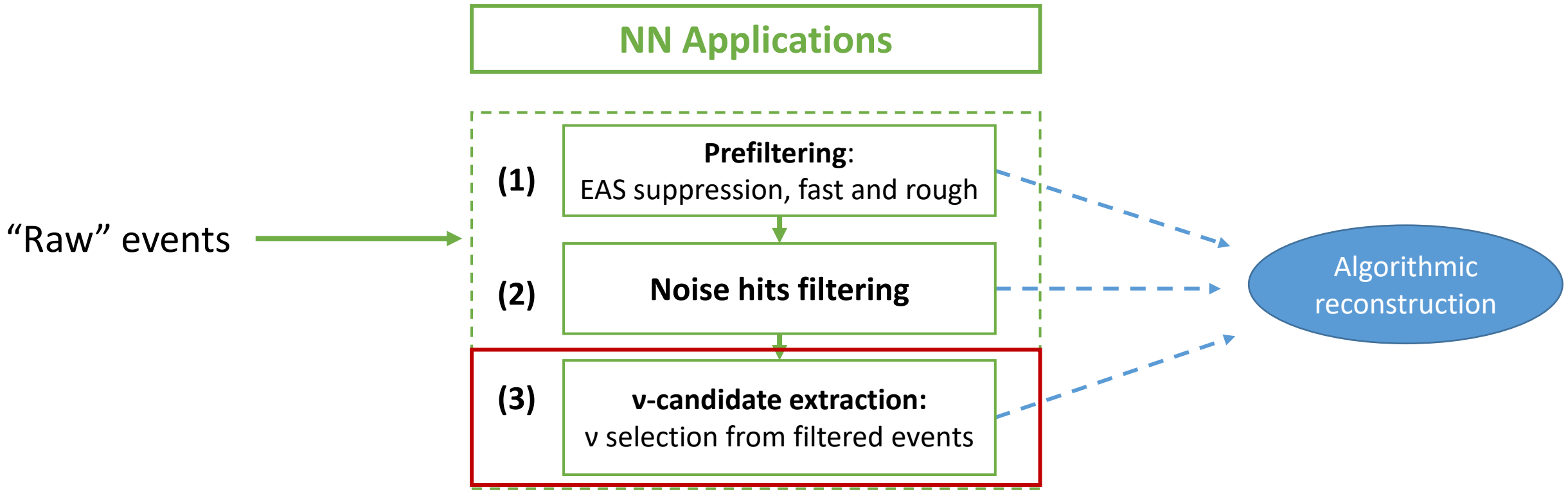
Problem: 85%-90% of hits are due to water luminescence.

NN solution: Hit-level Filter

suppresses noise hits with Recall=Precision=95%

Also may improve usual reconstruction!

ML processing pipeline



(3)

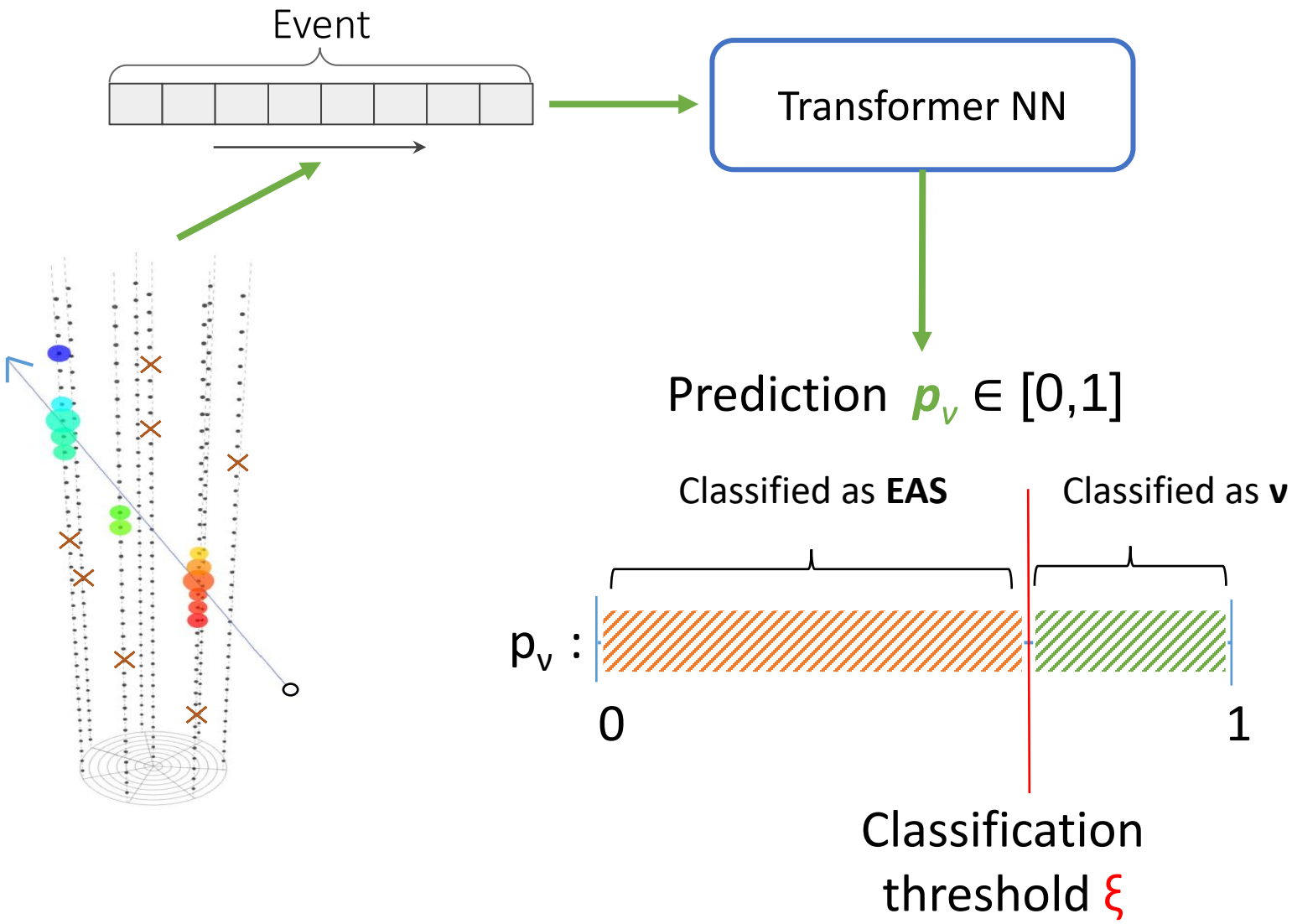
Problem: still many background events in data.

NN solution: ν-candidate Extractor

uses signal hits to select **ν**-events **precisely**

Selected **ν**-candidates may be passed through usual reconstruction!

Prefilter model



MC datasets events #

	EAS	ν_{atm}	ν_{cosmo}
Train	1M	0.5M	0.5 M
Val	50k	25k	25k
Test	600k	100k	100k

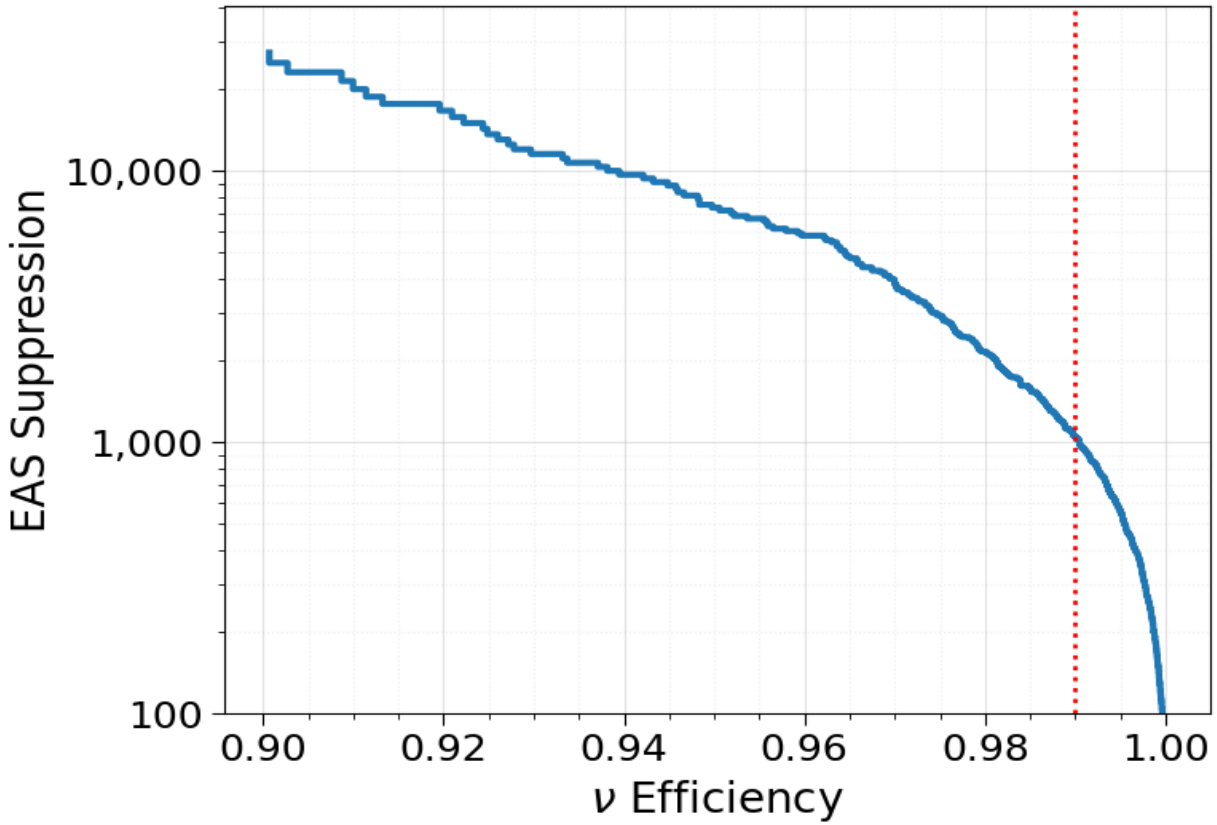
Metrics having ξ fixed:

- **EAS** suppression factor (FPR^{-1})
- **ν** events efficiency (TPR)

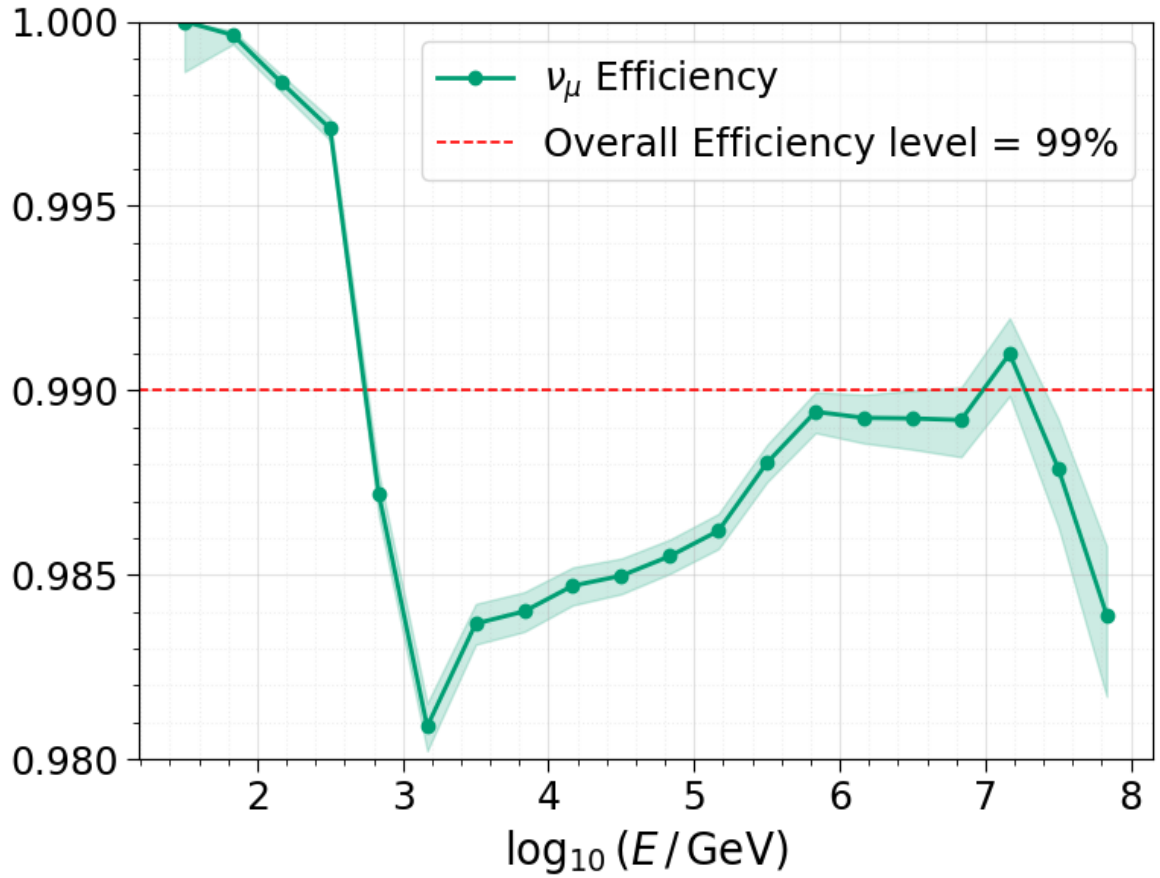
Prefilter model: metrics

Results for MC ground truth quality cuts:

- # signal hits ≥ 8
- # signal strings ≥ 2



EAS Suppression VS ν Efficiency (scanning ξ)
*At ξ = 0.761: suppresses EAS 1000 times while saving 99% **ν***



***ν** Efficiency VS **ν** Energy*
@ξ = 0.761

Prefilter model:
Domain Adaptation (DA) from MC to real data

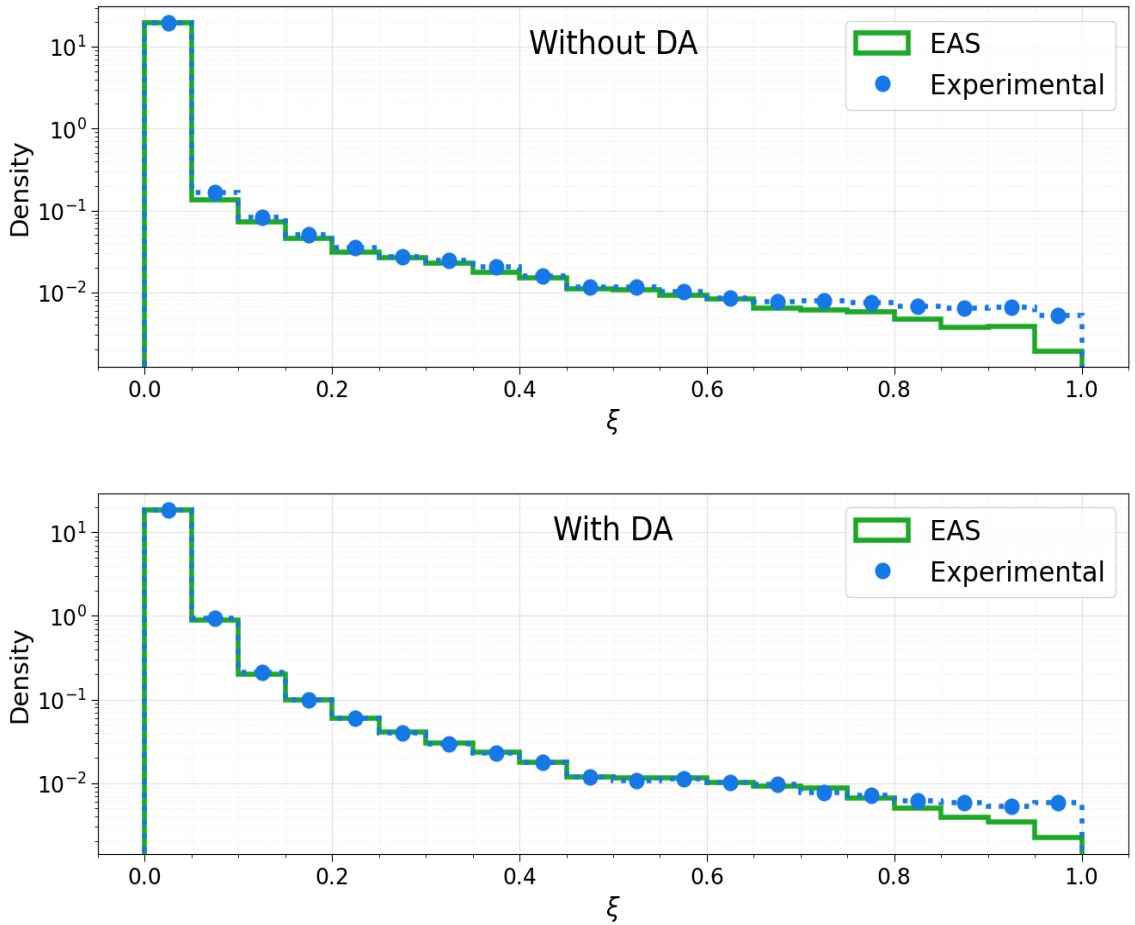
DA technique from arXiv:1505.07818

Uses (indirectly) real data while training on MC!

	Real Events
Train	255k
Val	45k

Applying standard cuts via **algorithmic** reconstruction (to select physically same EAS subsets):

- # of signal hits ≥ 8 and # of signal strings ≥ 3;
- covariance matrix convergence status equal to 3 (fully successful fit);
- number of minimizer calls < 350 (convergence quality);
- log10 phit > −9;
- log10(F/(nhits − 5)) < 2, where F is the minimizer objective value;
- ntriplets/nhits > 0.1 (track-topology requirement);
- zdist > 70 m;
- zcenter < 220 m;
- $\theta_{rec} \in (80^\circ, 180^\circ)$;
- $\sigma\theta < 2.5$ rad.

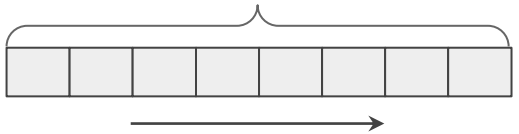


Prediction distributions for MC (EAS) and real (Experimental) dataset

Better correspondence with DA (?)

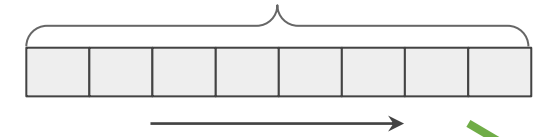
Noise hits filter

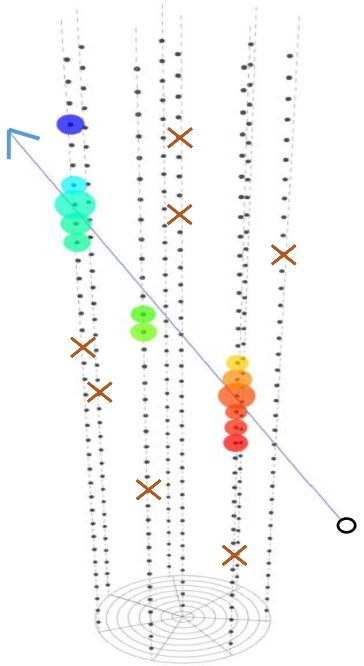
N pulses in event

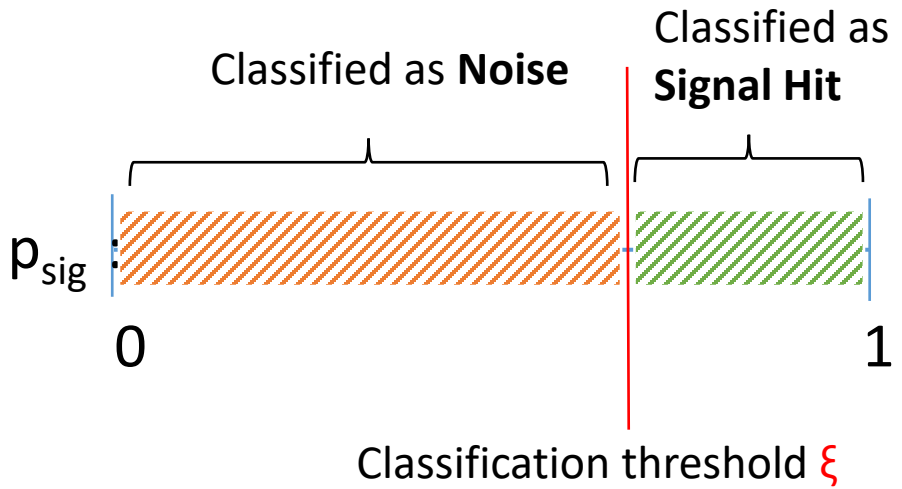


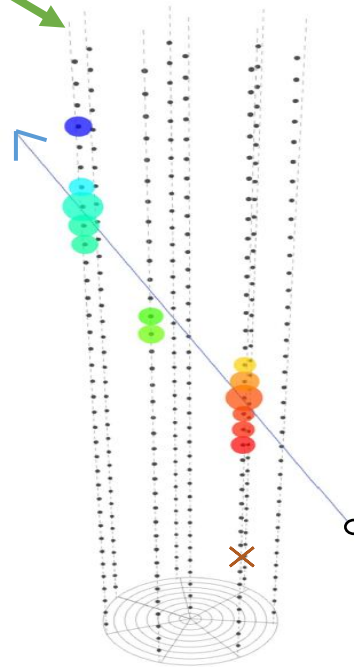
Transformer NN

N predictions $\boldsymbol{p}_{sig} \in [0,1]$









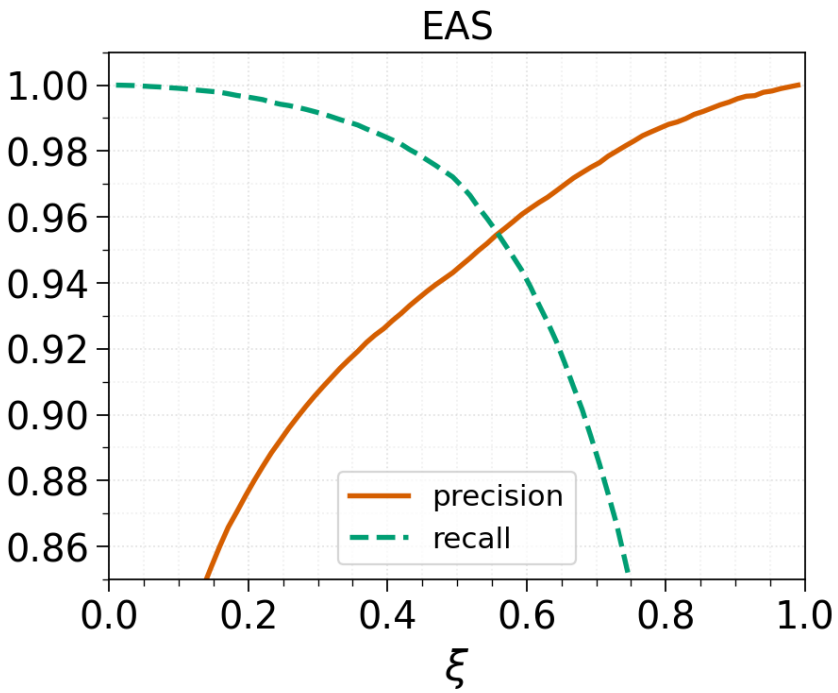
Metrics after ξ is fixed:

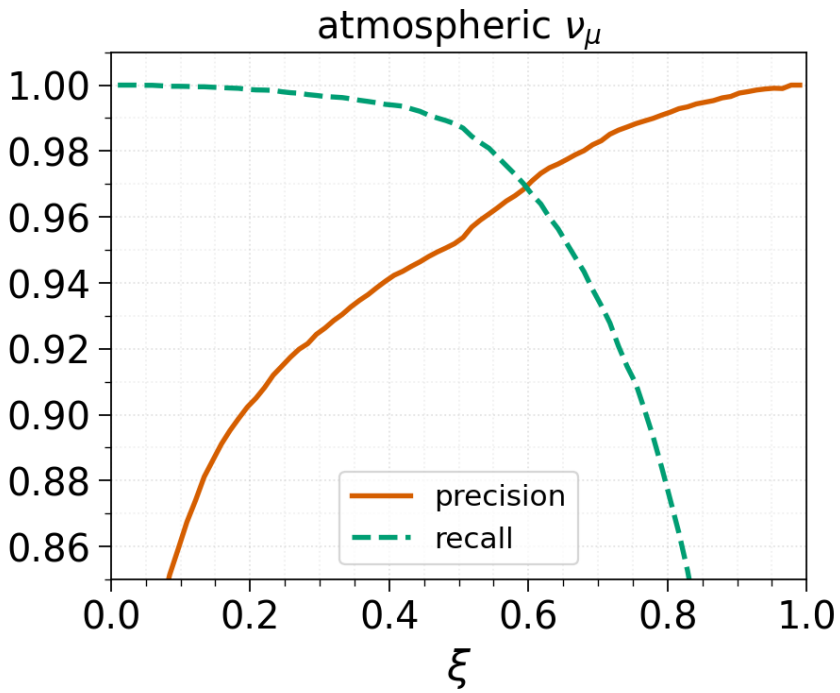
- **Precision** = TP/(TP+FP)
- **Recall** = TP/(TP+FN)

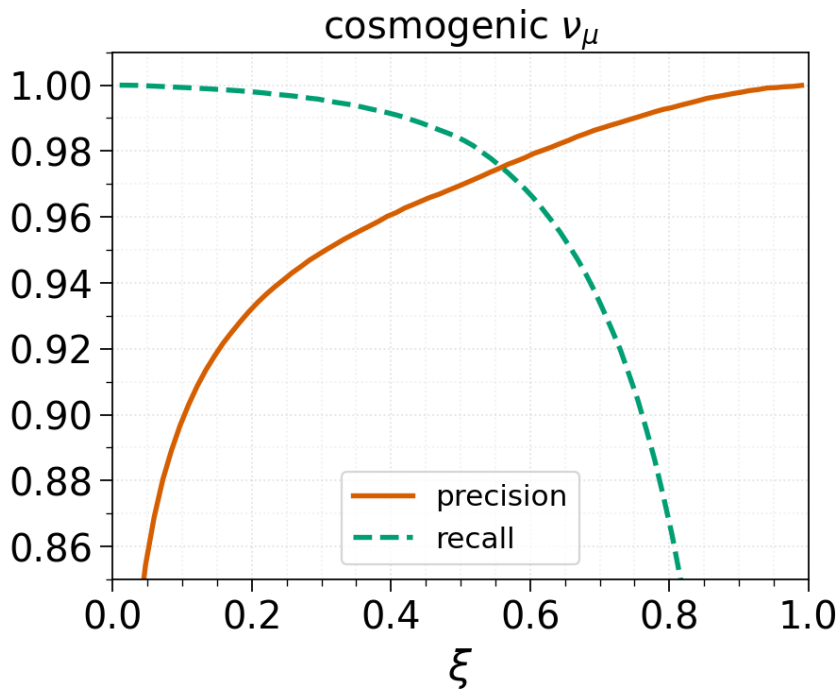
Noise hits filter: metrics

Results for quality cuts **via the model**:

- # signal hits ≥ 8
- # signal strings ≥ 2





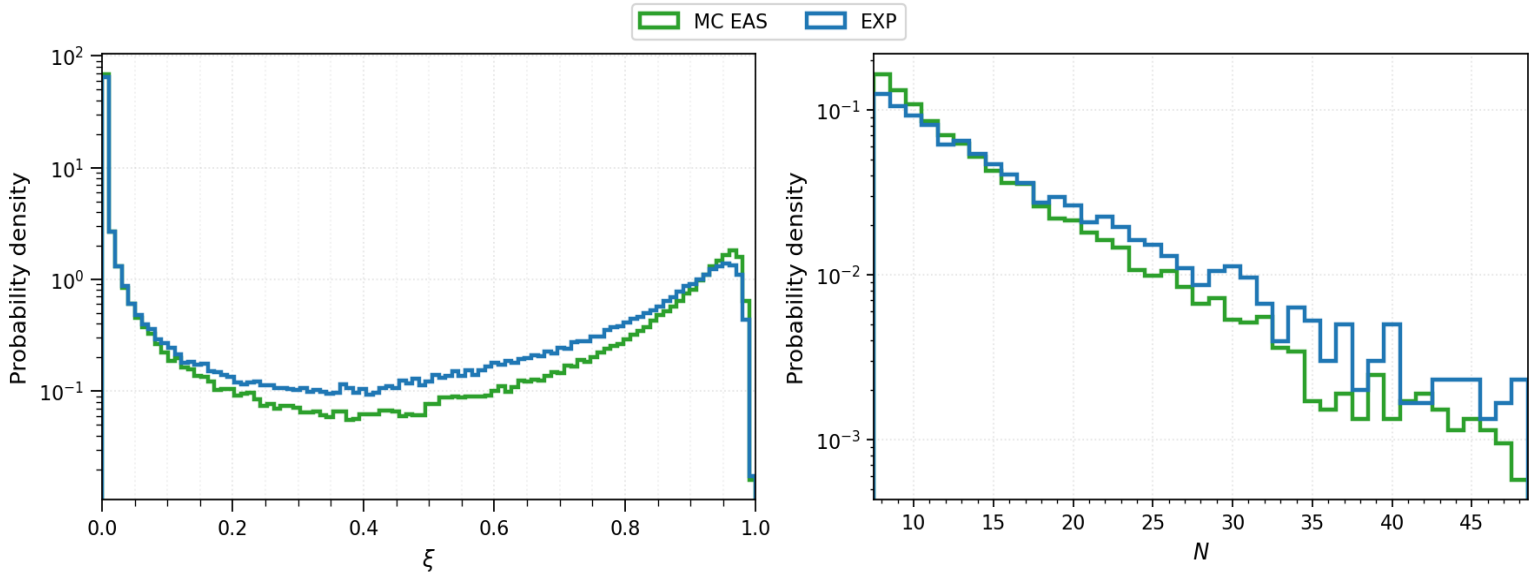


Both precision and recall exceed 95% at ξ = 0.55 for all event types.

Surpasses standard “scan-fit” algorithm [arXiv:2108.00208]

Noise hits filter:
Domain Adaptation (DA) from MC to real data

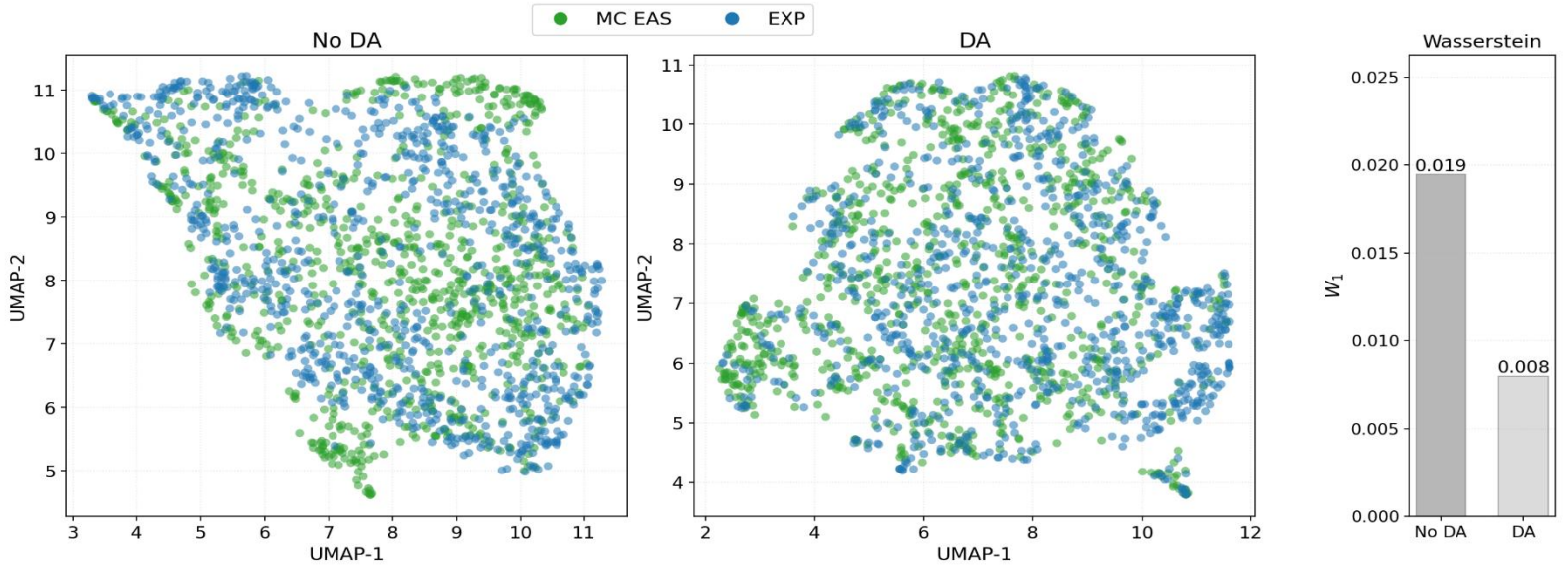
Same DA technique from [arXiv:1505.07818]



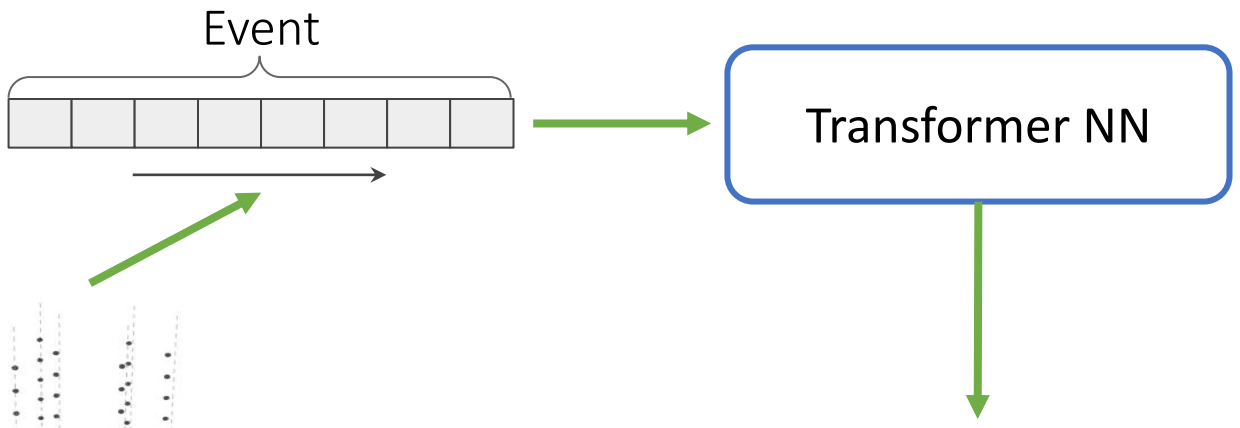
p_{sig} distributions and signal hits multiplicity for MC (EAS) and real data after cuts h8s2 **by the model itself** (trained **with** DA)

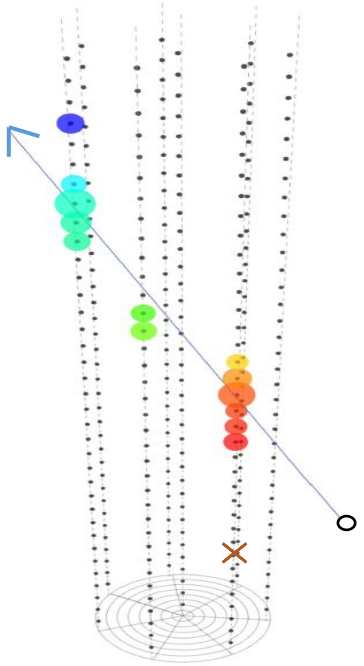
UMAP of the last encoder layer output to compare cases with and without DA

Wasserstein distance is computed between p_{sig} distributions



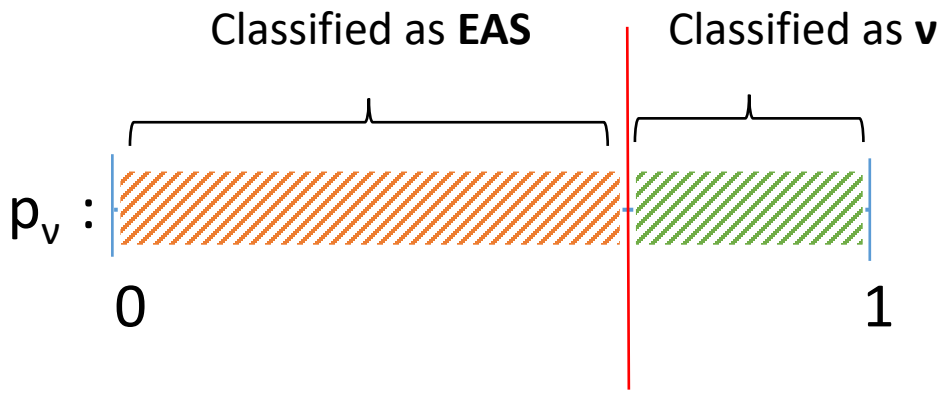
ν-candidate Extractor





After noise hits filter model with threshold 0.8

Prediction $p_\nu \in [0,1]$



Soft cut applied: signal hits ≥ 5

MC datasets events #

	EAS	ν_{atm}	ν_{cosmo}
Train	2.4M	1.2M	1.2 M
Val	200k	100k	100k
Test	23M	100k	100k

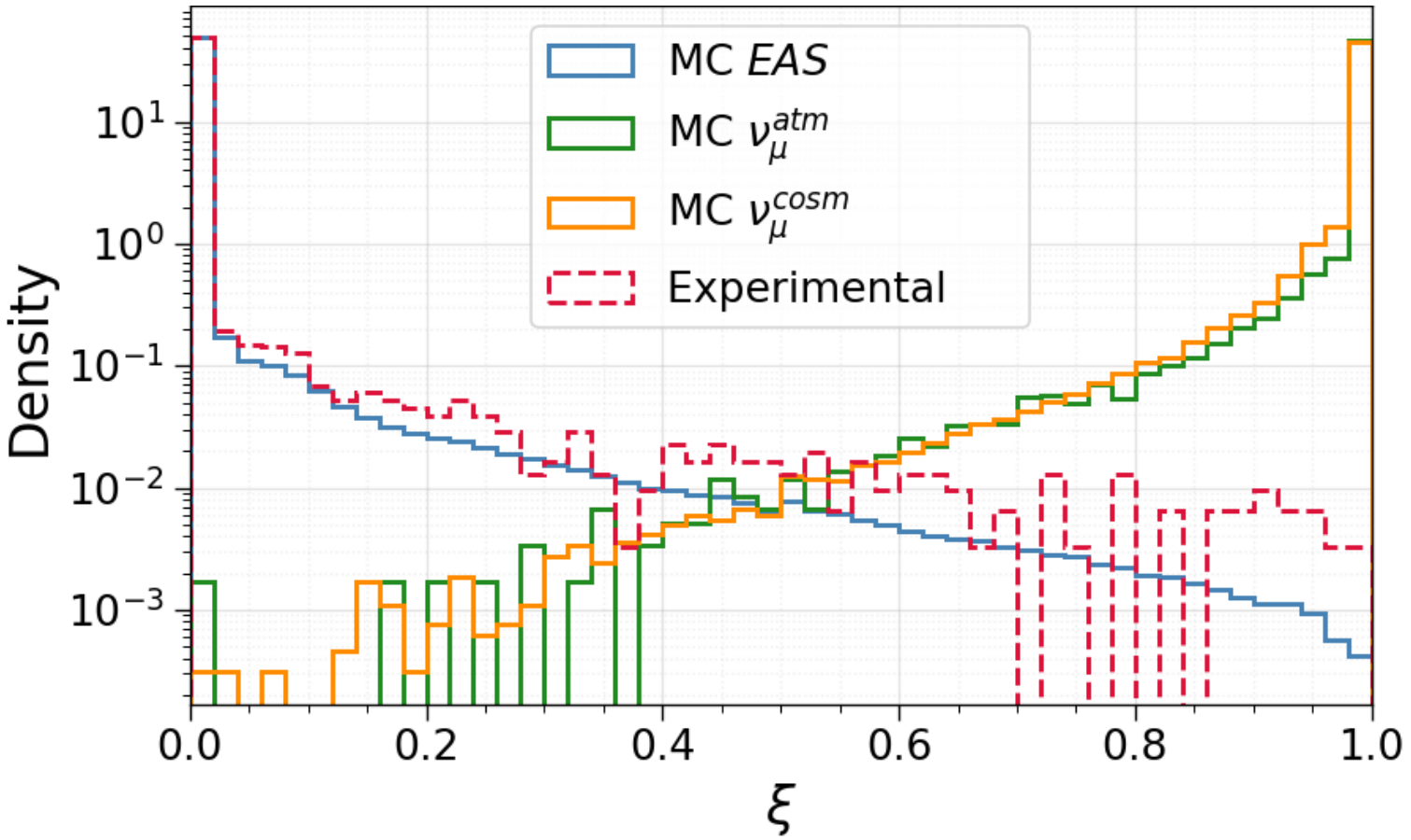
Metrics having ξ fixed:

- **EAS** suppression factor (FPR^{-1})
- **ν** events efficiency (TPR)

ν-candidate Extractor: real data (preliminary)

Same DA technique from arXiv:1505.07818 was applied

Quality cuts: # signal hits ≥ 8, # signal strings ≥ 3. MC EAS suppression of 1M times working point.



Prediction distribution for real and MC data

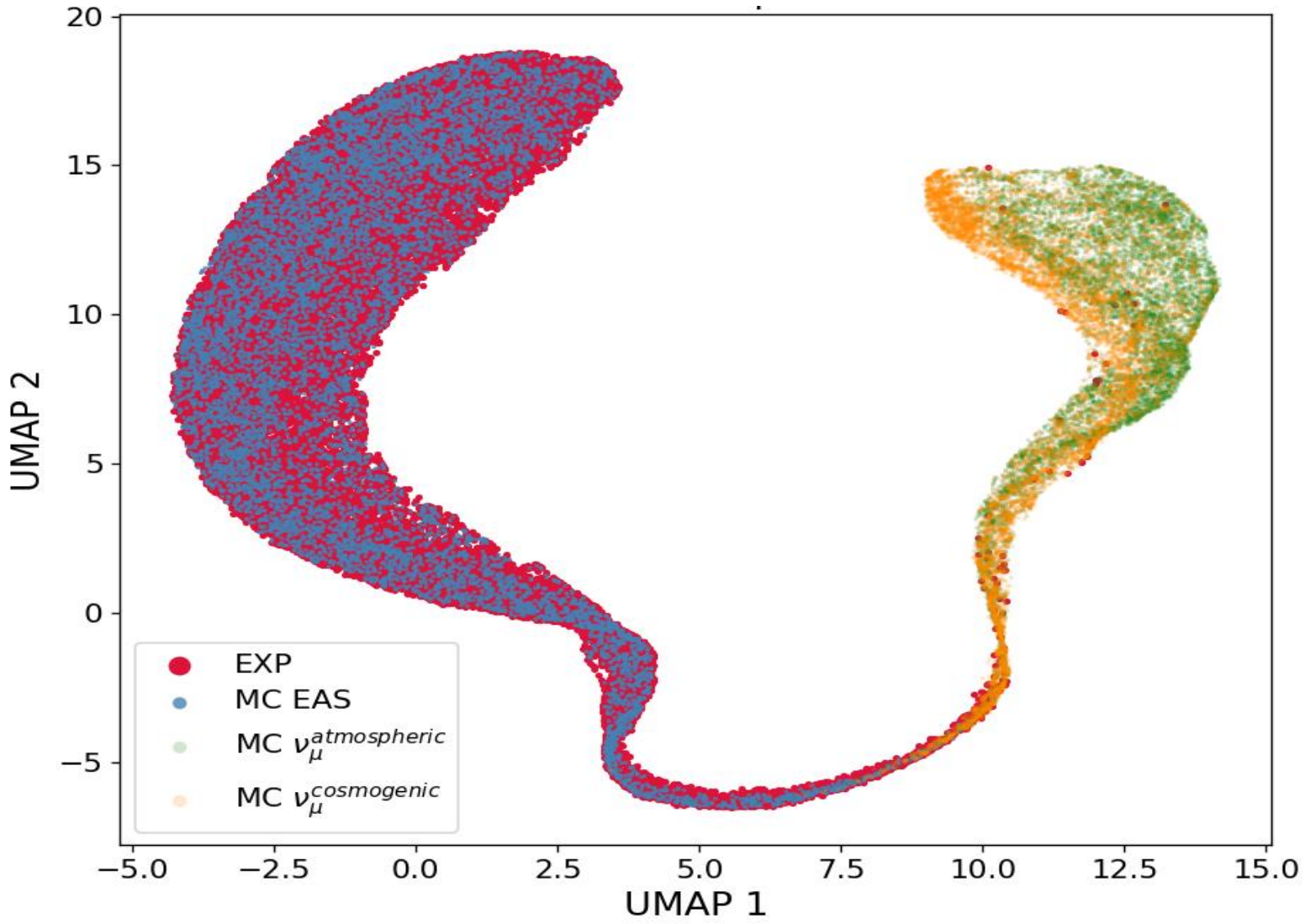
- At ξ=0.995 we suppress MC EAS by factor 10^6, saving 80% of MC ***ν*** *events*
- Close MC EAS and real data distributions (pipeline works!)
- But not exactly: some false ***ν*** candidates with ξ>0.99 in ≈ 15,000 experimental data events sample.

ν-candidate Extractor: real data (preliminary)

Same DA technique from arXiv:1505.07818 was applied

Quality cuts: # signal hits ≥ 8, # signal strings ≥ 3. MC EAS suppression of 1M times working point.

At the UMAP representation red "experimental" dots are found deep in the MC ***ν*** events area → strange sign to work with.



Summary

1. Progress in 3 ML tasks was achieved (MC metrics)s

I. Events Prefilter	***II. Noise hits filter***	***III.* ν-candidate extractor**
Preserves 99% h8s2 **ν** events Suppresses background in 1000 times.	95% Recall and Precision for all hits	Suppresses background by factor 10^6 Preserves 80% **ν** events

2. Made steps towards real data processing
   - Domain Adaptation makes models more robust while switching from MC to real data
   - However some domain mismatch was found for **ν-candidate extraction** task

3. What's next?
   - Figure out and solve the domain mismatch (MoE, soft labeling)
   - To solve downstream reconstruction tasks:
     - Energy
     - Direction of Arrival

Thanks!
Please ask your questions!

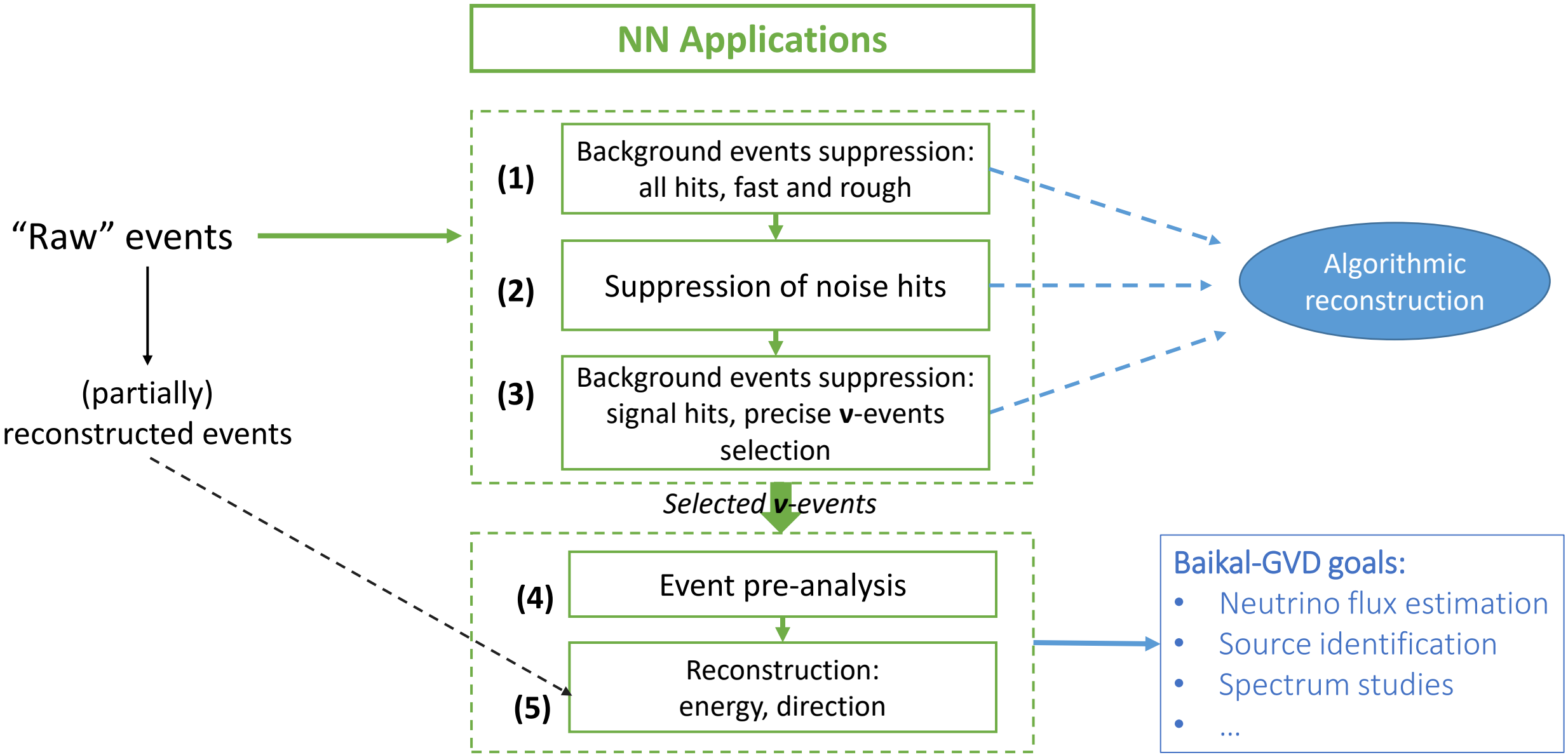
Contacts:
matseiko.av@phystech.su
t.me/AlbertMac280

Code and models:

github.com/ml-inr/Baikal-ML

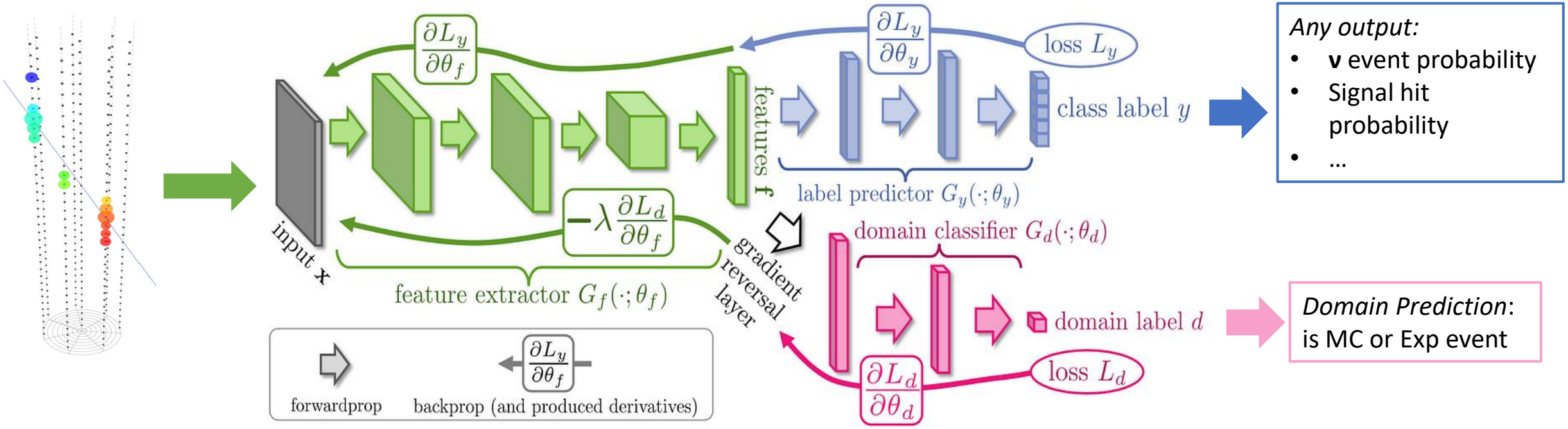
Backup

Full ML processing pipeline



Domain Adaptation Technique

DA scheme:
domain-adversarial training with a gradient reversal layer



Any output:
- **ν** event probability
- Signal hit probability
- …

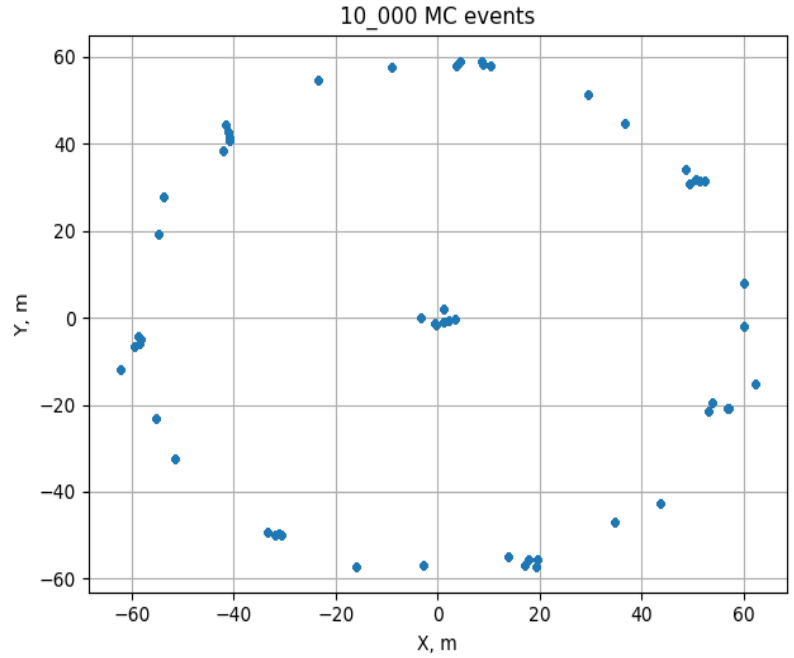
Domain Prediction:
is MC or Exp event

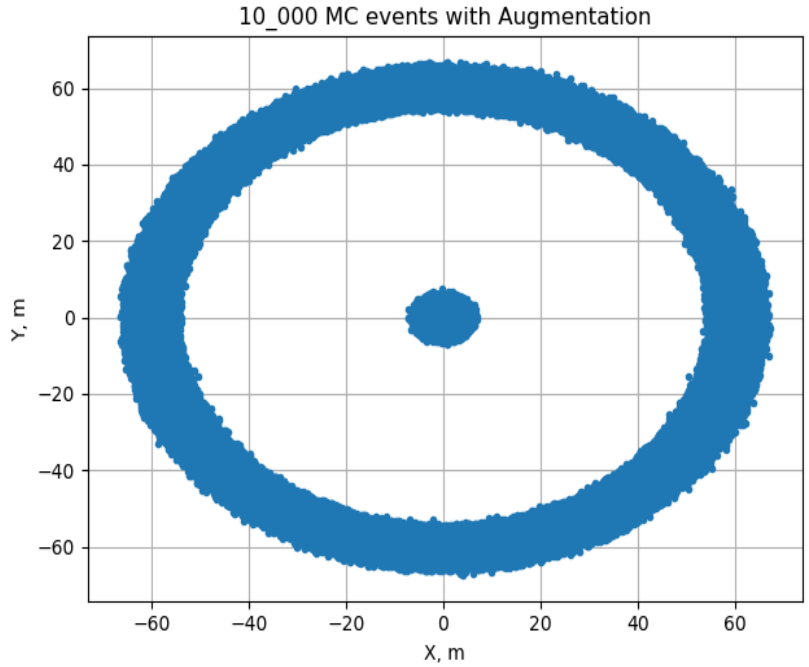
Source arXiv:1505.07818

Fast pre-filter: dataset

Data augmentations:

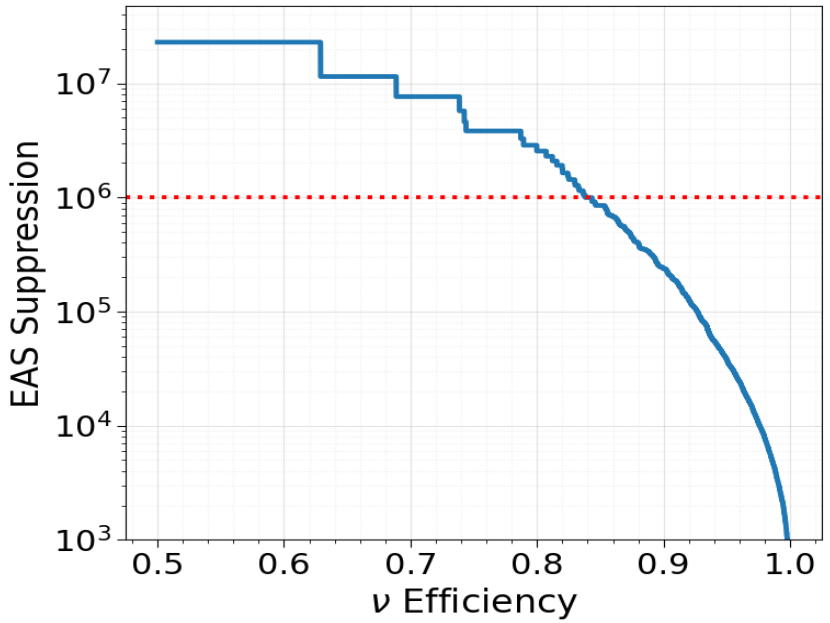
- Random gaussian noise to all inputs.
  STDs:
  - 1 m for **x** and **y**
  - 2m for **z**
  - 5 ns for **t**
  - 0.05 pe for **Q**
- Random rotations in x-y plane





ν-candidate Extractor: MC metrics

Results for data after quality cuts:
- # signal hits ≥ 8
- # signal strings ≥ 3



EAS Suppression VS ν Efficiency (scanning ξ)
*Capable of suppressing EAS by 10^6 times while saving >80% **ν***

***ν** Efficiency VS azimuth angle Θ and Energy @ suppression factor of 10^6*