Event filtering and mitigation of simulation biases using machine learning

IPA workshop on Machine Learning for particle physics and astrophysics 2023

Marta Babicz

University of Zurich

21.03.2023

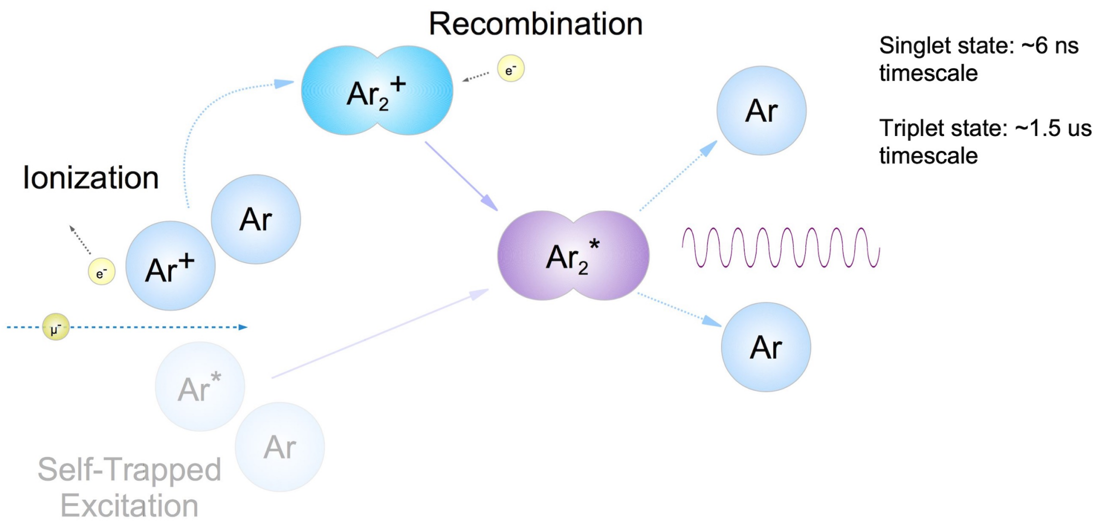
Introduction

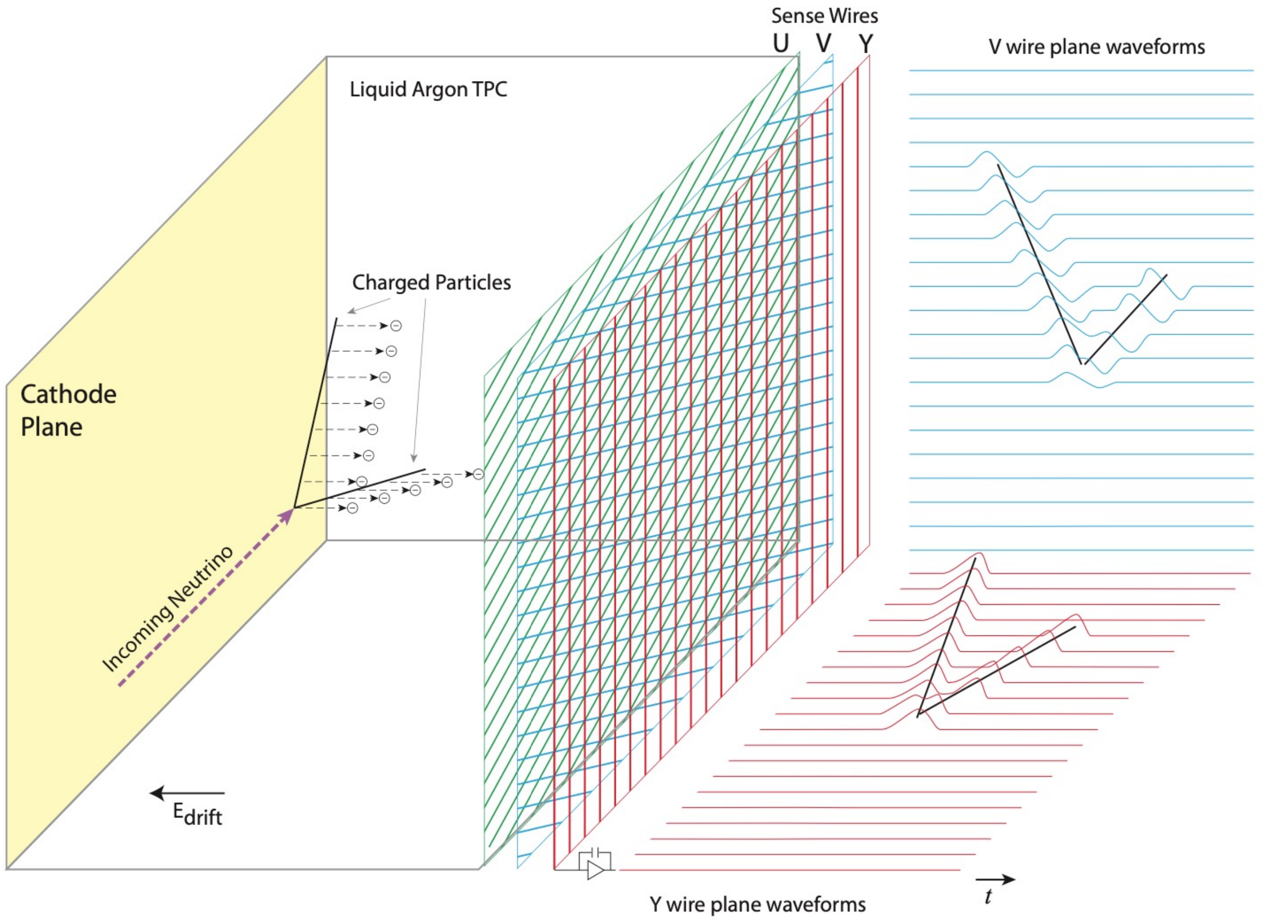
- Motivation: Current and future LAr-TPC* key challenge: identifying neutrino interactions from the pervading cosmic-ray background.
- Goal: to reduce this background using 3D Convolutional Neural Network (CNN) trained on low-level timing information from the scintillation light signal.
- Challenge: uncertainties from scientific modelling when applying ML algorithms trained on simulated data to *real data*
- Solution: use domain-adversarial neural network (DANN) to make CNN robust against systematic uncertainties
- Further improvement: semi-supervised domain adaptation using labelled events from target distribution

** LAr-TPCs are particle detectors that use liquid argon to capture images of rare physical events, such as neutrino interactions, by detecting small electrical signals and flashes of light produced when these particles pass through the liquid argon.*

Liquid Argon Time Projection Chambers

The operational principle: when a charged particle interacts in LAr, it induces atomic excitation and ionization, followed by recombination both leading to the emission of scintillation photons ($\lambda = 128\ nm$)

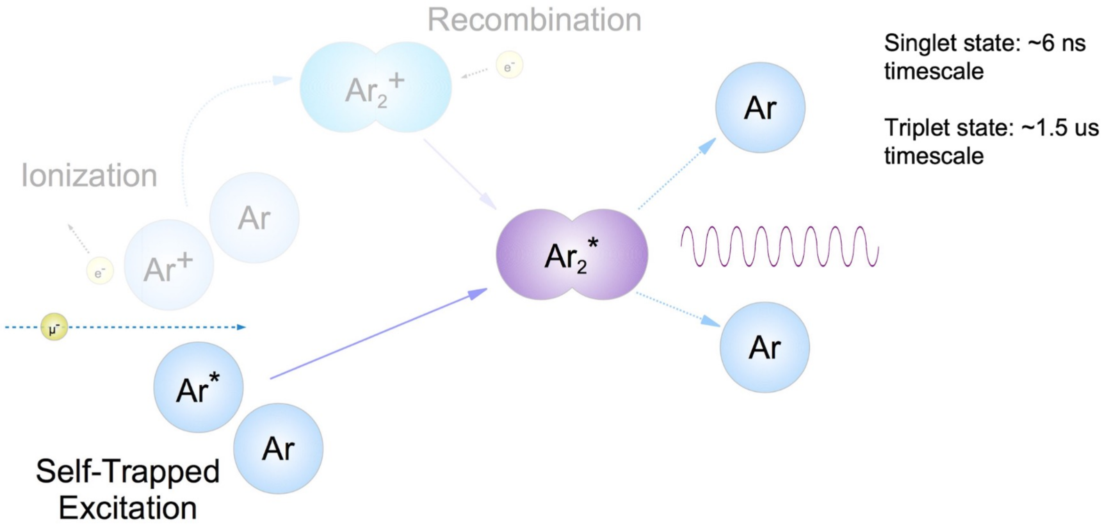


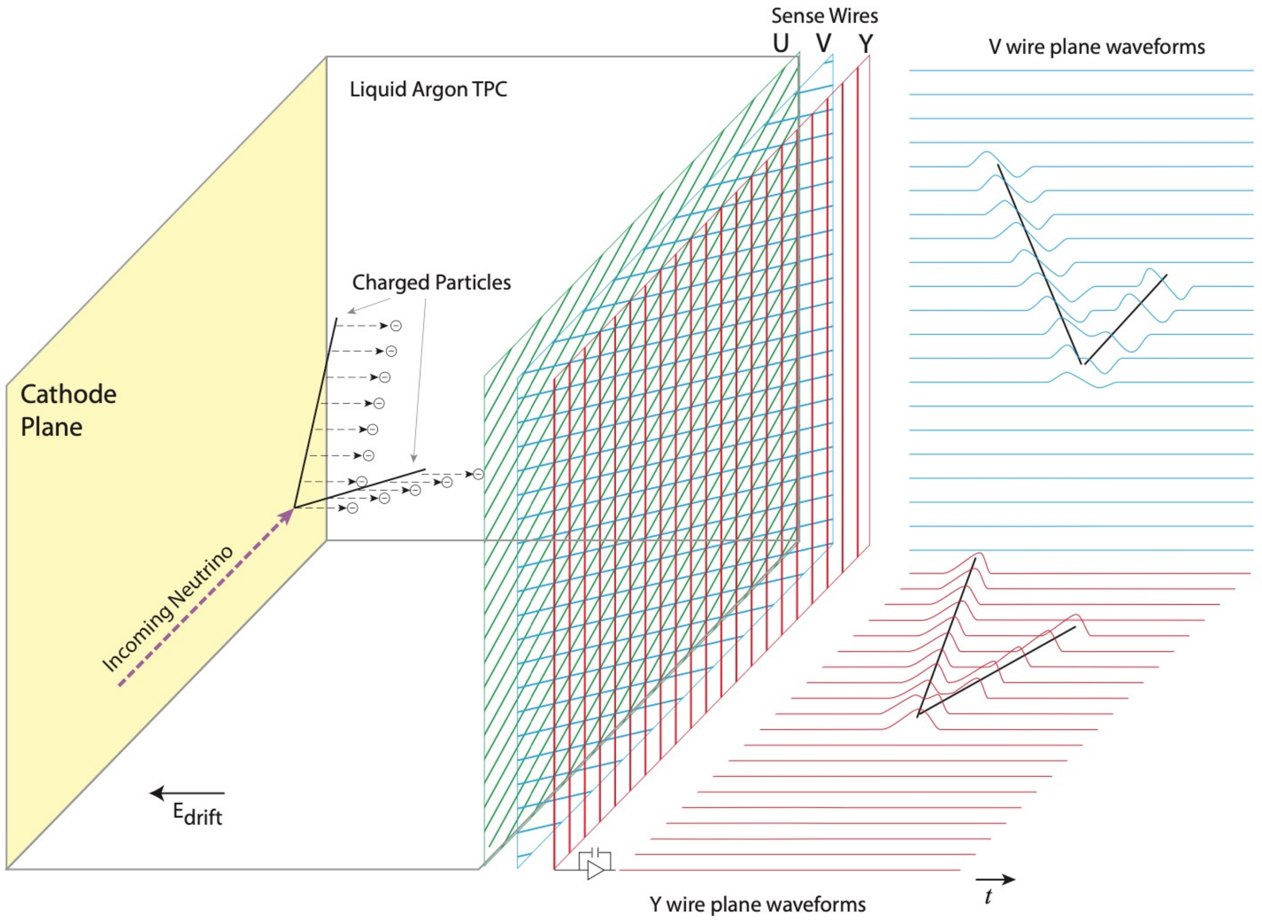


Detectors using the liquid Argon time projection technique: ICARUS, ArgoNeut, MicroBooNE, SBND, ProtoDUNE, DUNE.

Liquid Argon Time Projection Chambers

The operational principle: when a charged particle interacts in LAr, it induces atomic excitation and ionization, followed by recombination both leading to the emission of scintillation photons ($\lambda = 128\ nm$)

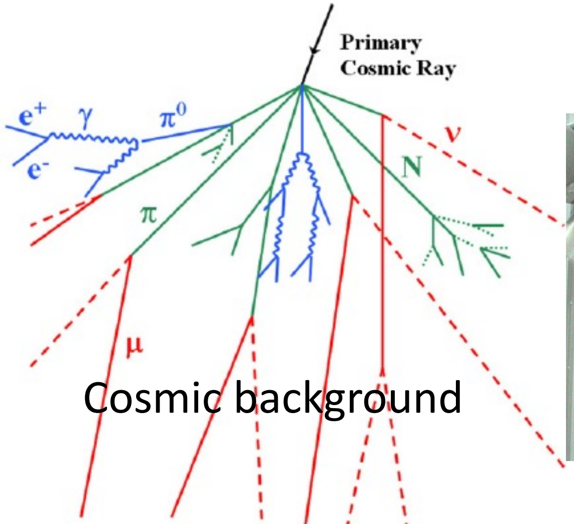




Detectors using the liquid Argon time projection technique: ICARUS, ArgoNeut, MicroBooNE, SBND, ProtoDUNE, DUNE.

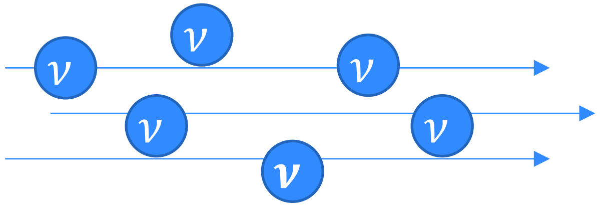
Case study: ICARUS detector

LAr TPCs are powerful neutrino detectors, but they can also detect other particles, such as cosmic rays, which can be mistaken for neutrino signals.

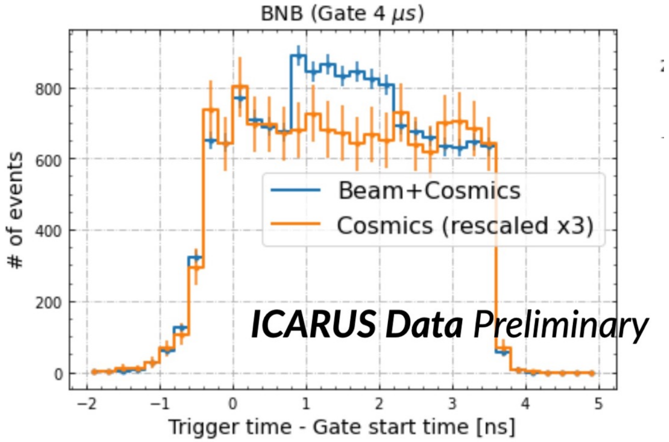


360 PMTs

Neutrino beam

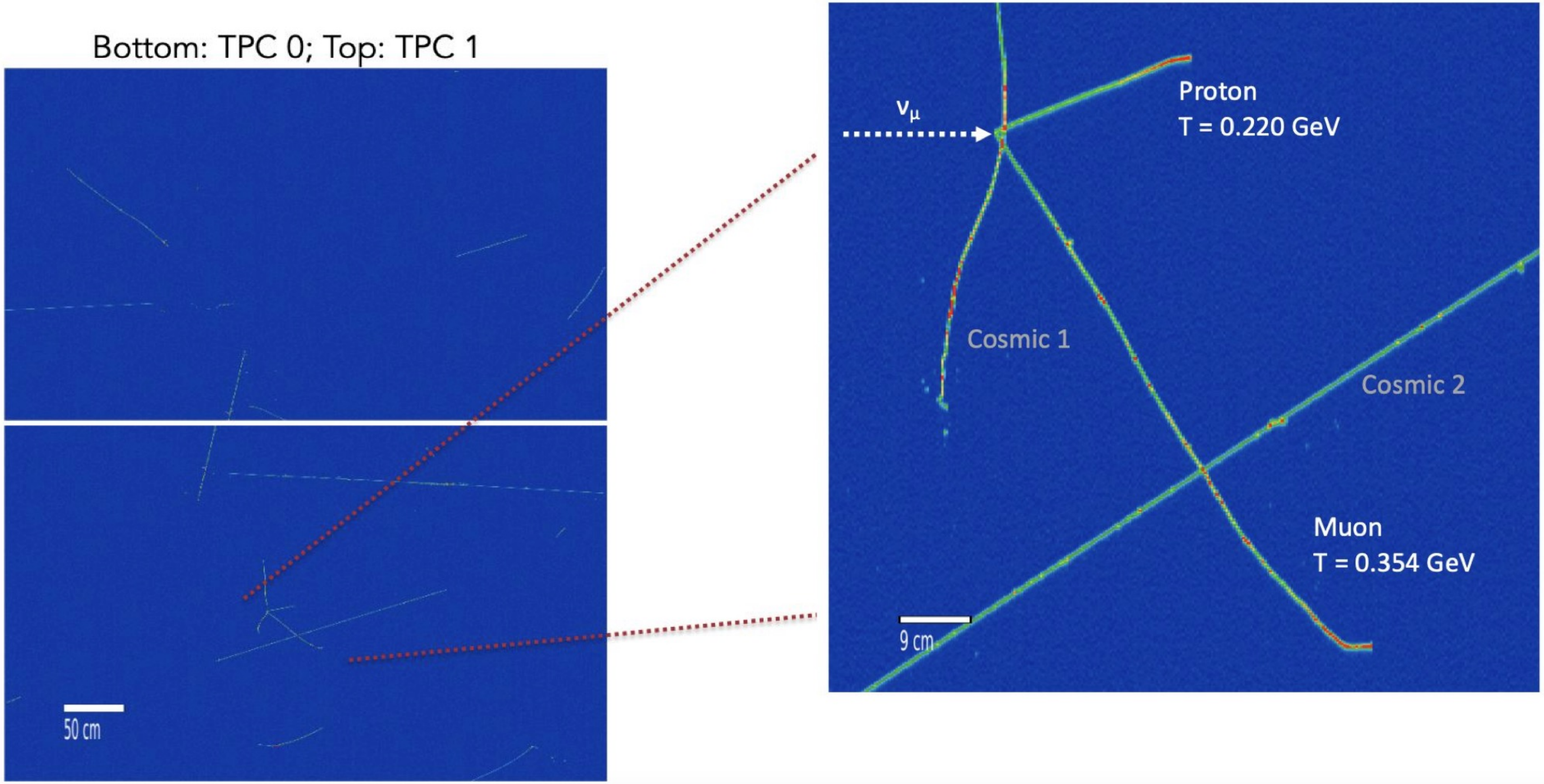


Excess of PMT light signal in correspondence with neutrino beam observed.



LAr TPC image example

Simulated BNB neutrino event
Muon neutrino charge current interaction (Energy = 0.697 GeV)

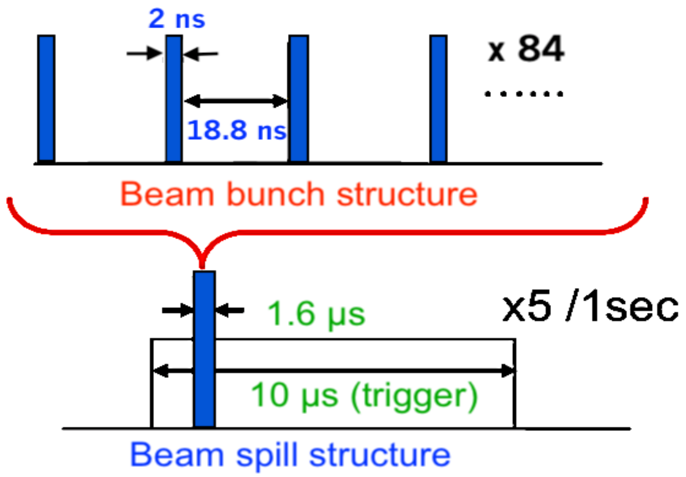


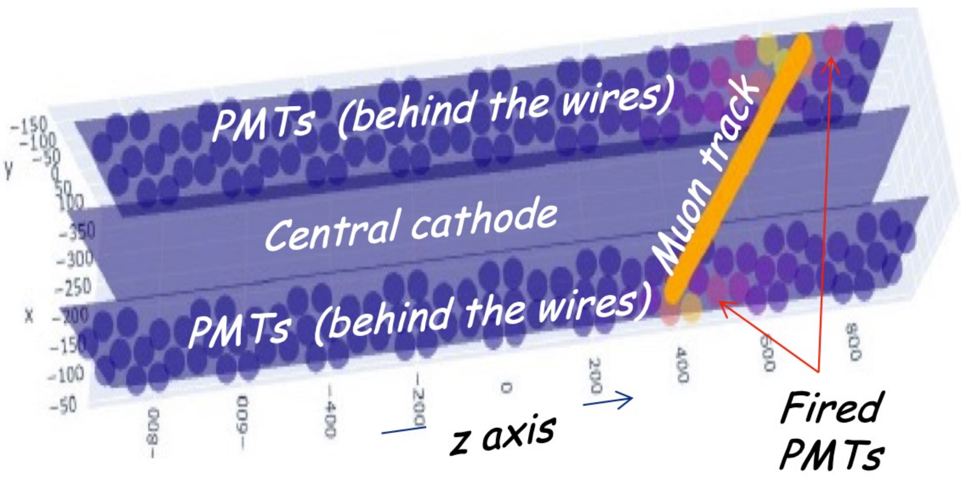
Methodology

1. Within the neutrino beam spill window over three times more cosmic backgrounds are expected than neutrino interactions in the ICARUS detector.

2. The aim is to reduce this background using the information available from 360 ICARUS PMTs:
    - **3DPosition:** of each pair as the point half way between them
    - **OpeningTime:** time at which a light pulse is detected by the PMT
    - **NOpenings:** number of light pulses detected by the PMT within a given time window

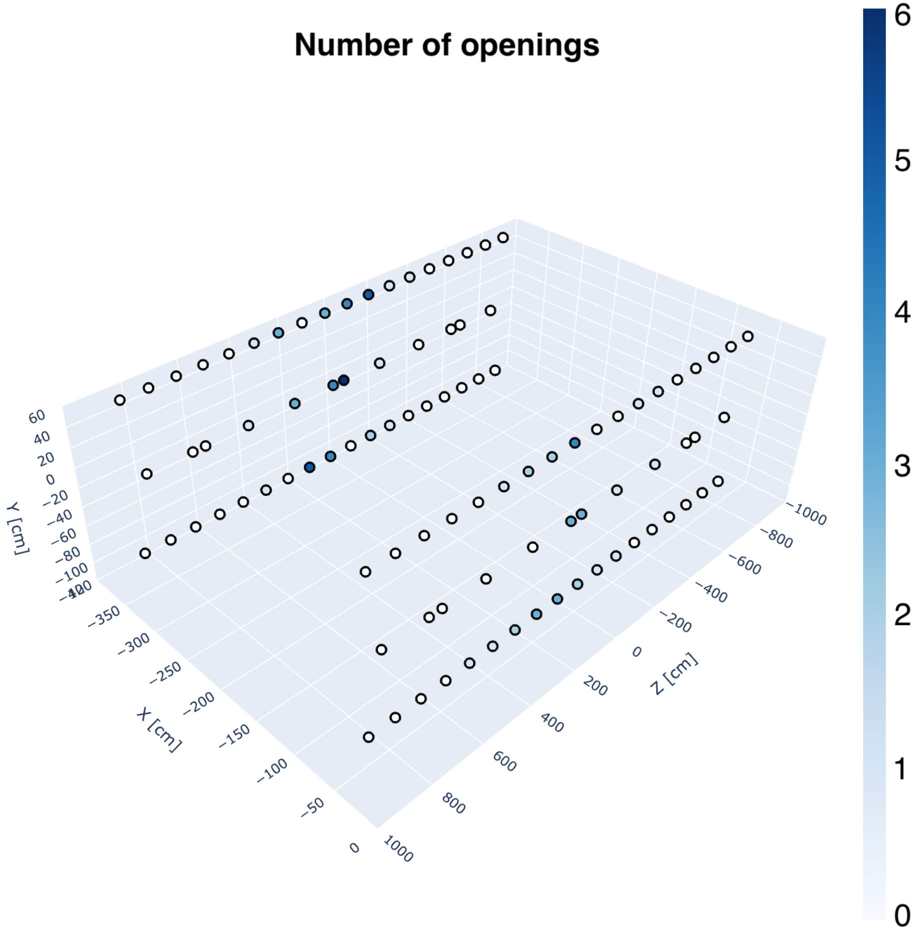
3. To generate simplified images for event filtering.

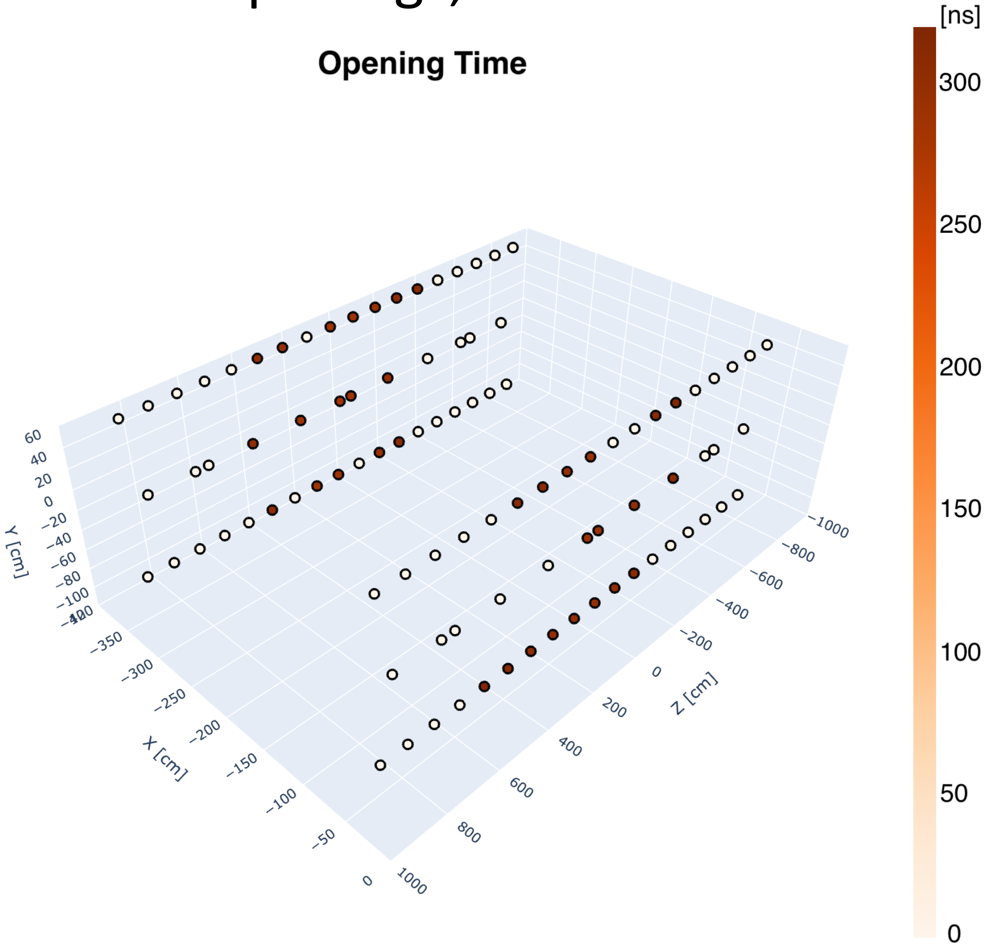




Training data

The simulated PMT data is presented as images, where the position of each PMT pair, alongside its opening time and a number of openings, are stored.

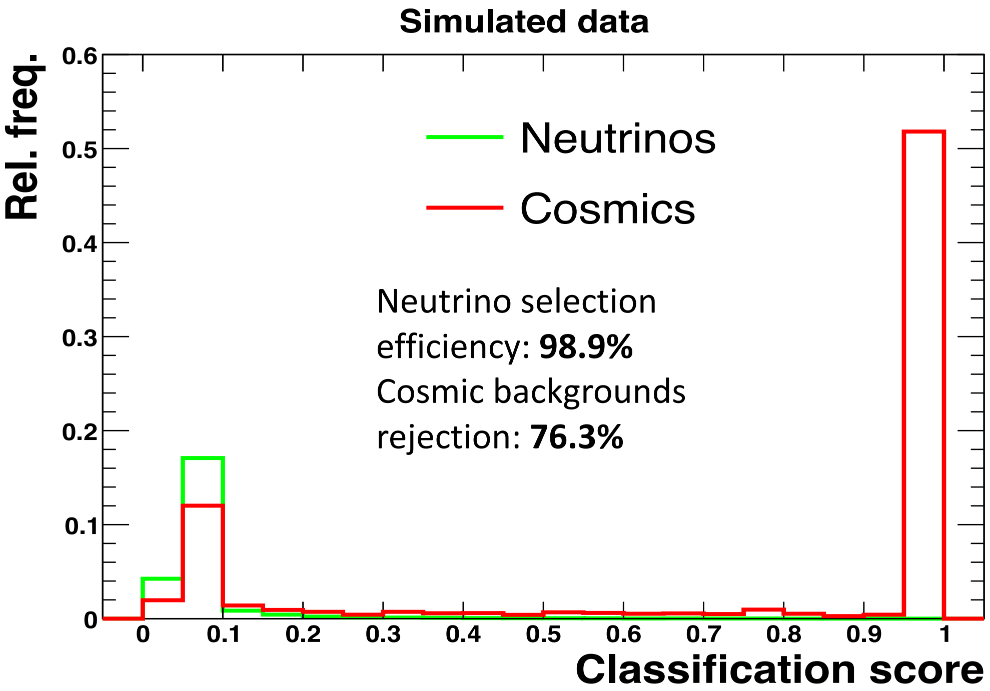


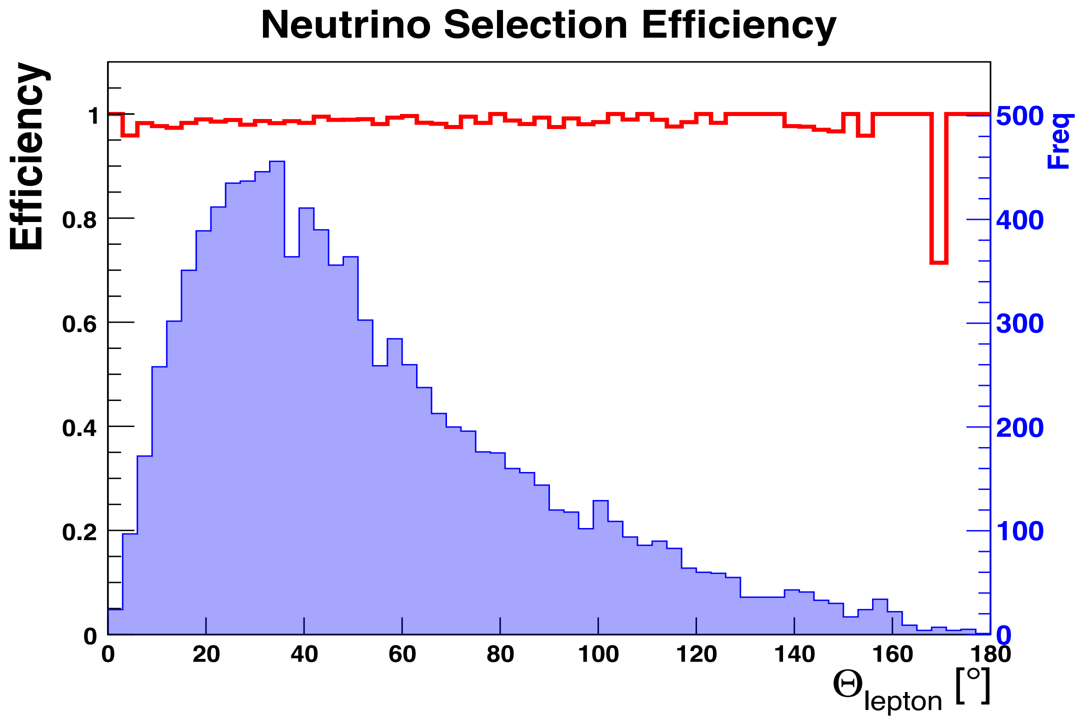


Event filtering in ICARUS using PMTs

- Once trained, the output of the CNN is a score for each event between 0 (neutrino-like) and 1 (cosmic-like).
- The charged-current selection efficiency is found to be flat (i.e., unbiased by kinematics) in various tested observables.

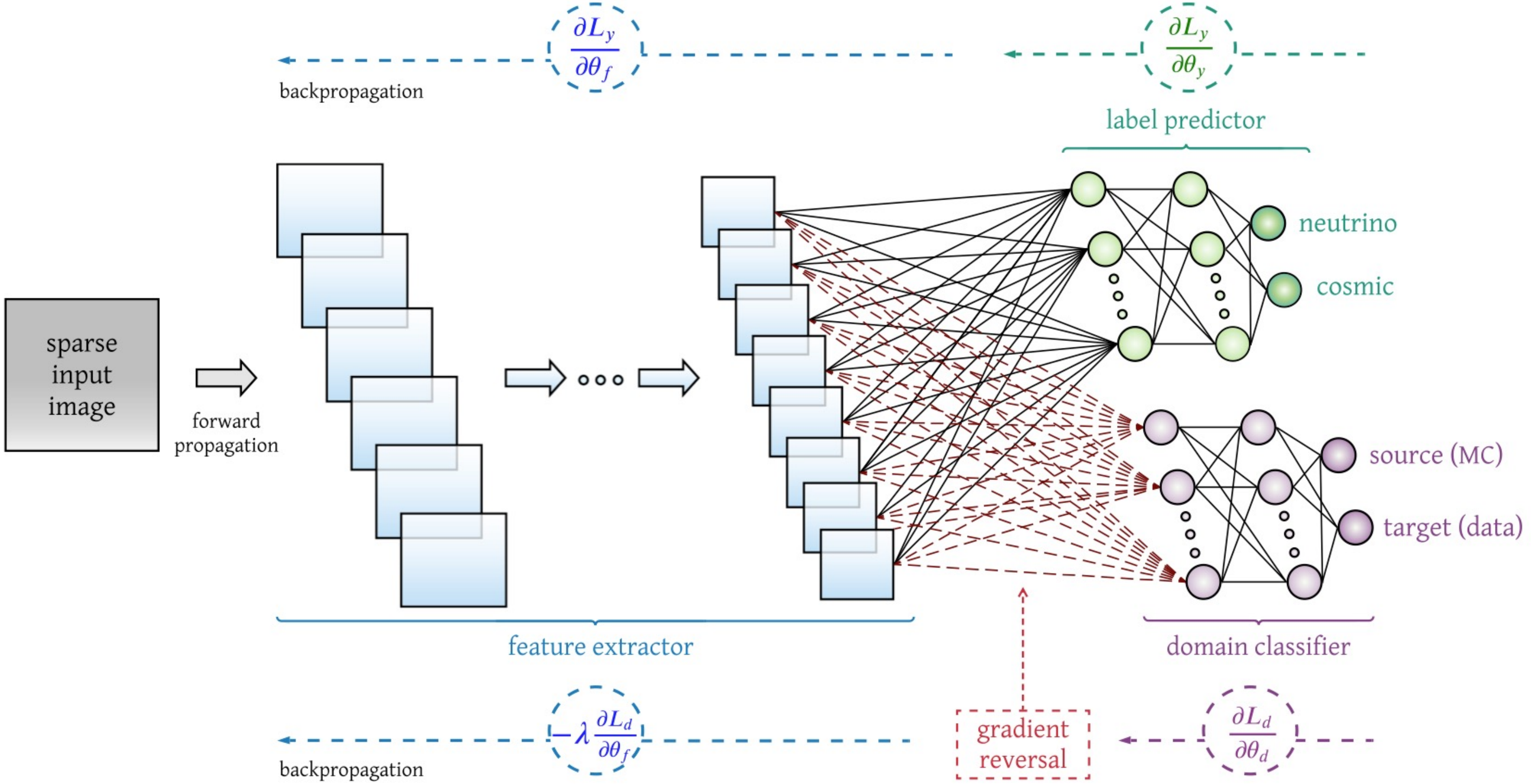
M. Babicz *et al PoS ICRC2021 (2021) 1075*





Simulation bias

To mitigate the simulation dependence, the Domain Adversarial Neural Network (DANN) has been introduced on top of CNN.



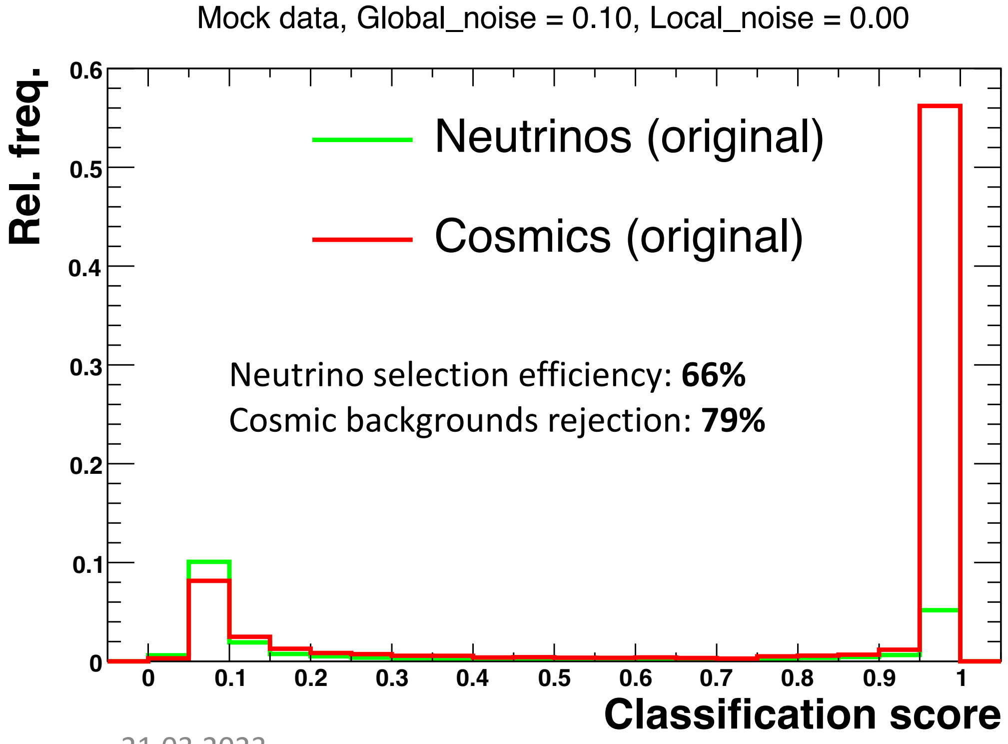
domain classifier tries to distinguish between the source and target domains, while the feature extractor tries to confuse the domain classifier by learning domain-invariant features

Reducing the simulation bias

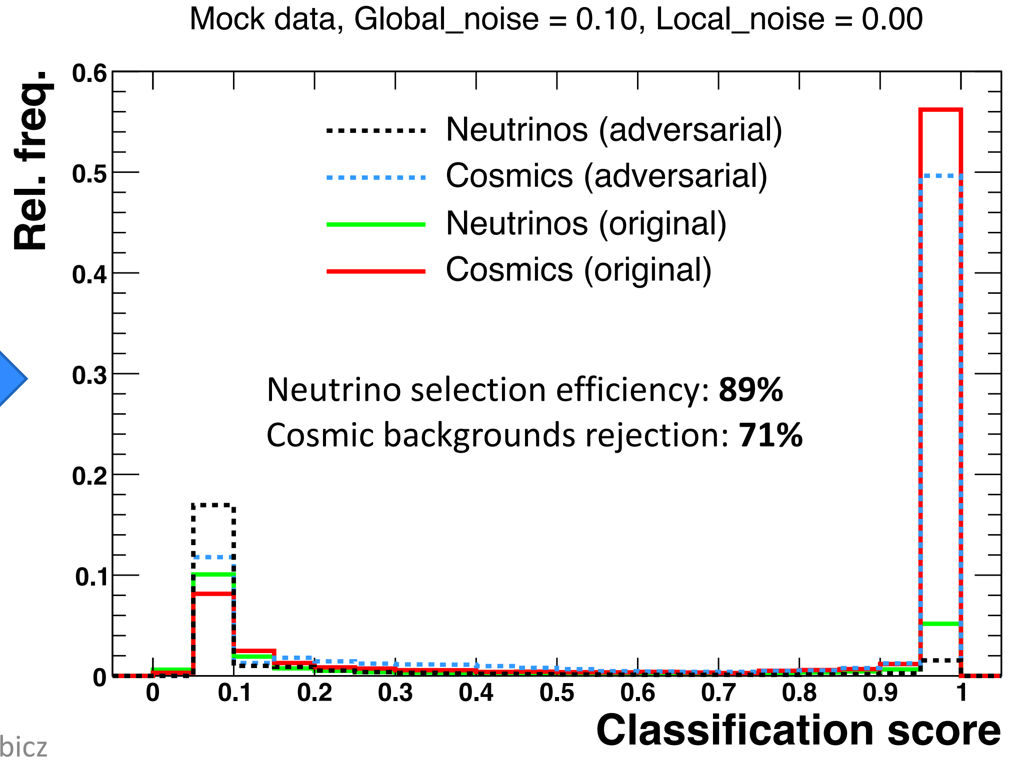
- CNN algorithms trained on simulation, but applied to real data, face uncertainties due to imperfect modelling.
- To test this, mis-modelling bias (as global/local noise) was introduced to the simulation.

  M. Babicz et al *Phys.Rev.D* 105 (2022) 11, 112009

- To mitigate the simulation dependence, the Domain Adversarial Neural Network (DANN) has been introduced instead of CNN.
- Improvement of neutrino selection efficiency while keeping the cosmic rejection factor at a similar level was obtained.



After DANN



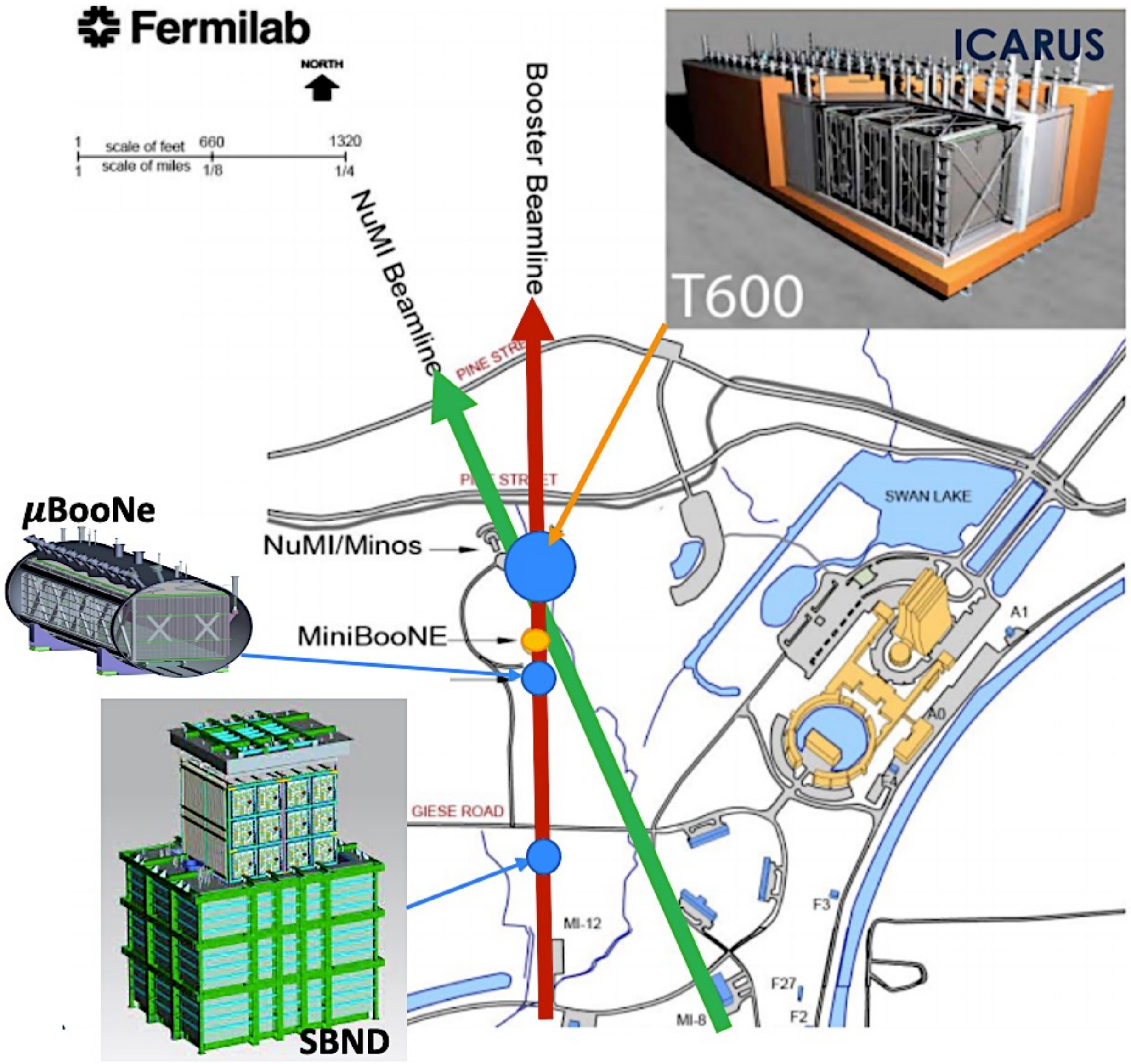
Conclusions

- A ML-based event filter to separate neutrino interactions from cosmic background has been developed using the PMT information.
- The event filter successfully rejects the vast majority of the cosmic background while efficiently selecting neutrino interactions.
- A way to mitigate potential biases from imperfect input simulations by applying Domain Adversarial Neural Networks (DANNs) was developed.
- DANNs can learn a feature representation that is invariant to the shift between the source domain (simulated data) and the target domain (real data), potentially improving the accuracy and efficiency of event filtering in LAr TPCs.
- This method was trained on real data and tested on eye-scanned data and showed an increase in efficiency for selecting neutrino candidates.
- The results showed that the average CNN score for selected neutrino candidates was 0.31 with an efficiency of 95%.
- The application of the DANN to the same list of events resulted in an average score of 0.28 and an efficiency of 99%.

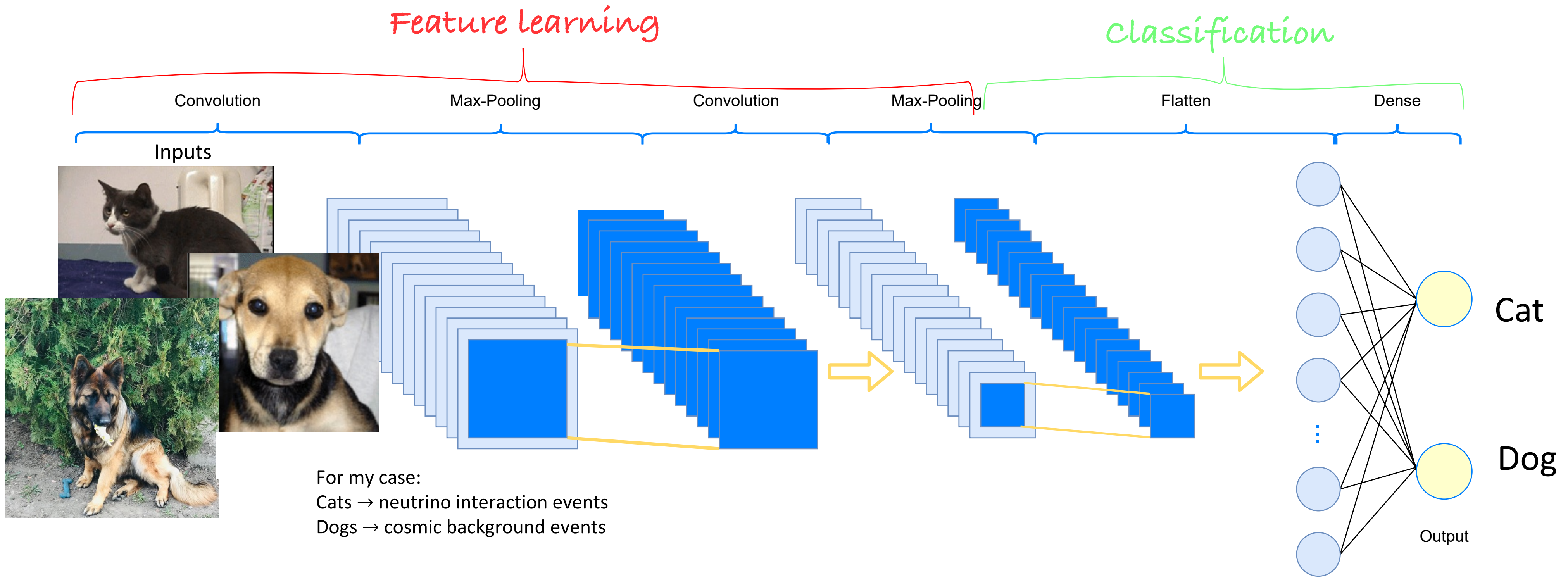
Short Baseline Neutrino program at Fermilab

Program aimed at final check of LSND and MiniBooNE observations

- Use of three liquid Argon time projection chambers (LArTPCs) located at the Fermilab Booster Neutrino Beamline (BNB).
- Same detector technology: reducing systematic uncertainties to the % level.
- All the three detectors are on the surface and thus are exposed to a high cosmic background.
- Although the latest MicroBooNE's results show no sign of the sterile neutrino, to definitively solve this puzzle we still have to wait for the other two detectors.



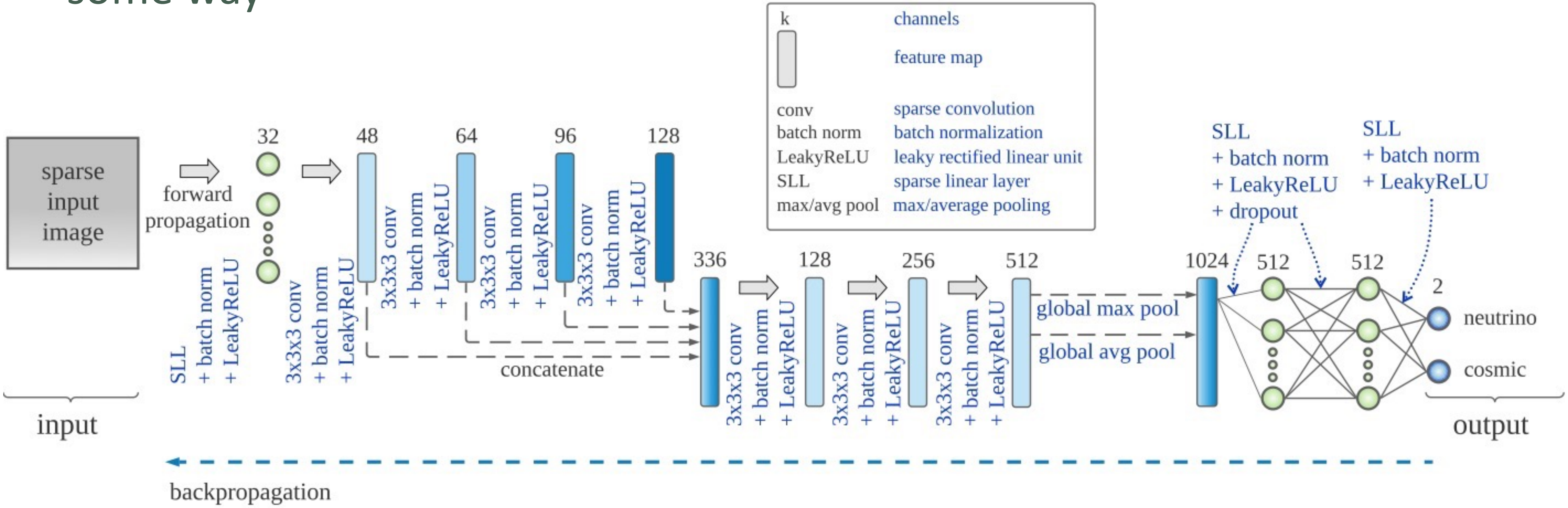
Convolutional Neural Network



1. **Convolution** – performed on the input data with the use of filter to produce feature maps.
2. **Max-pooling** - a down-sampling of the feature map representation reducing its dimensionality.
3. Finally, the feature maps will be **flatten** into a single column vector to form a fully connected (**dense**) layer.

Our CNN implementation (in reality)

- Goal: to extract features from the images that allow to classify them in some way



One huge advantage of using CNNs: no need to do a lot of pre-processing on images.