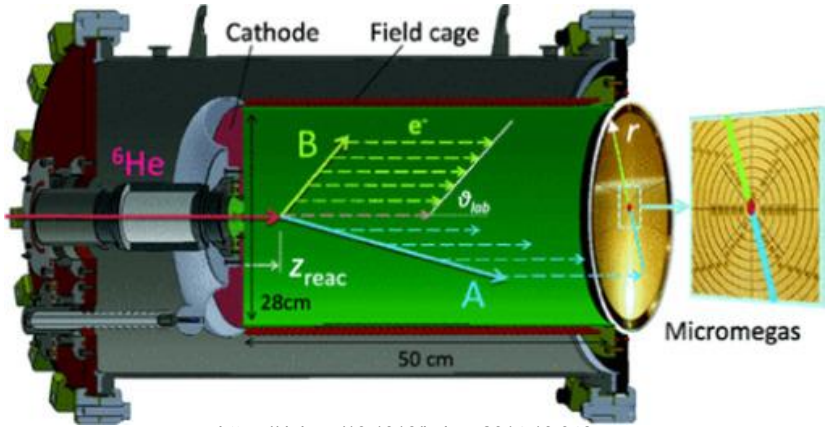
Unsupervised Learning to Build Pretrained Models for the AT-TPC

Maya Wallach[1], E. Villasana[2], R. Wimbush[2], M.P. Kuchera[2], R. Ramanujan[2], Y. Ayyad[3]

[1]Michigan State University
[2]Davidson College
[3]Universidad de Santiago de Compostela

Background

- The Active Target Time Projection Chamber (AT-TPC) is a particle detector located at the Facility for Rare Isotope Beams at Michigan State University
- FRIB runs different beam/target combinations with different experimental setups approximately every two weeks.



https://doi.org/10.1016/j.nima.2014.10.048

Motivation

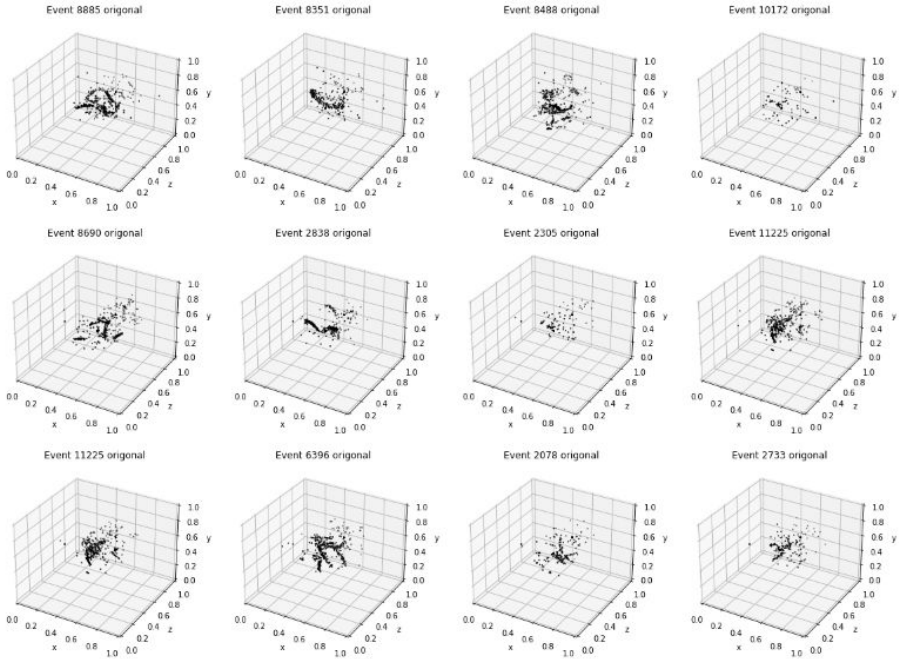
- Supervised methods require labeling for the training set
- With constantly evolving experimental parameters, this requires retraining models for each experiment.

Objective

We aim to build an unsupervised pretrained model that can be used for all upcoming experiments to accomplish various ML tasks

Background: Data

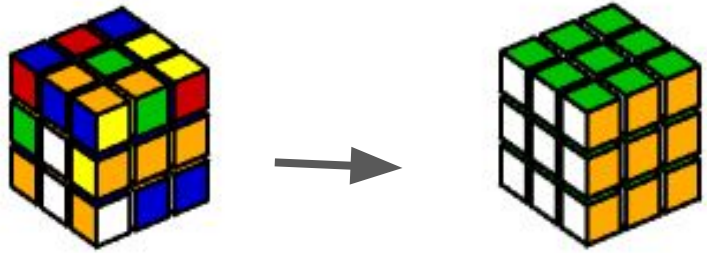
- In this project we used experimental data from two experiments
  - **Beam**: Oxygen-16, **Gas Medium**: Helium
  - **Beam**: Magnesium-22, **Gas Medium**: Helium
- The datasets are completely unlabeled and the events are represented in the form of point clouds
  - Point clouds are a collection of points in a space



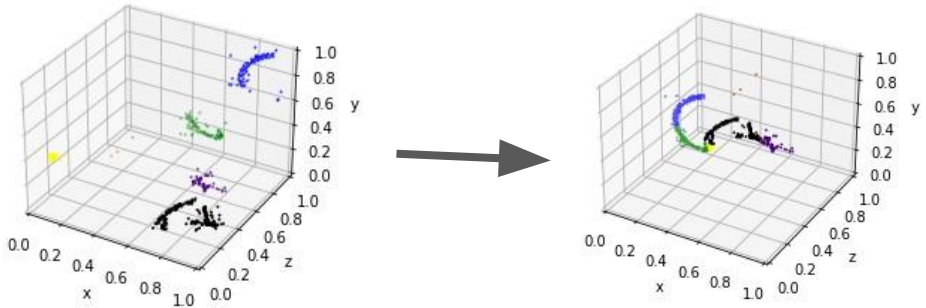
Methods: Background

- **PointNet:**
  - The architecture operates on point cloud data
  - Makes supervised, point-wise predictions

- **Unsupervised task:**
  a. Voxelize the data
  b. Have a set of events as the “control” group i.e. keep them unshuffled
  c. For every other event, translate the contents of each voxel to a different location (“scramble”)
  d. For each point in the scrambled event, the PointNet model attempts to predict its voxel of origin or in the case of an unscrambled event, predict if it should be unscrambled at all

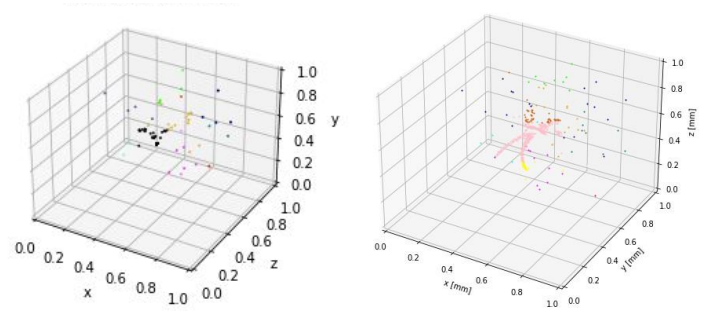


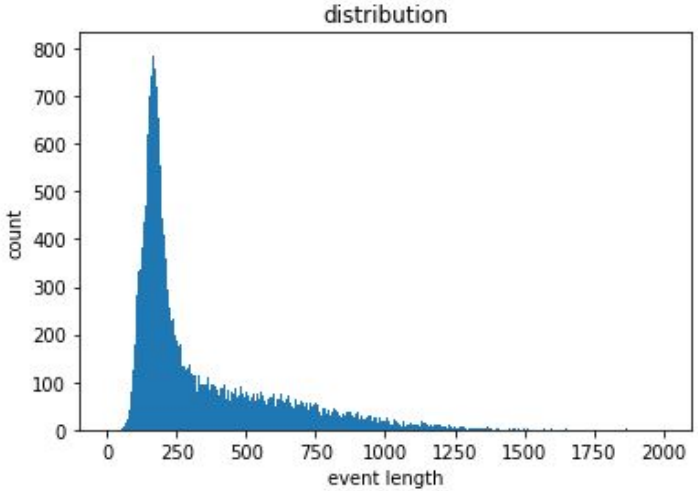
Input Output



Methods: Filtering Data

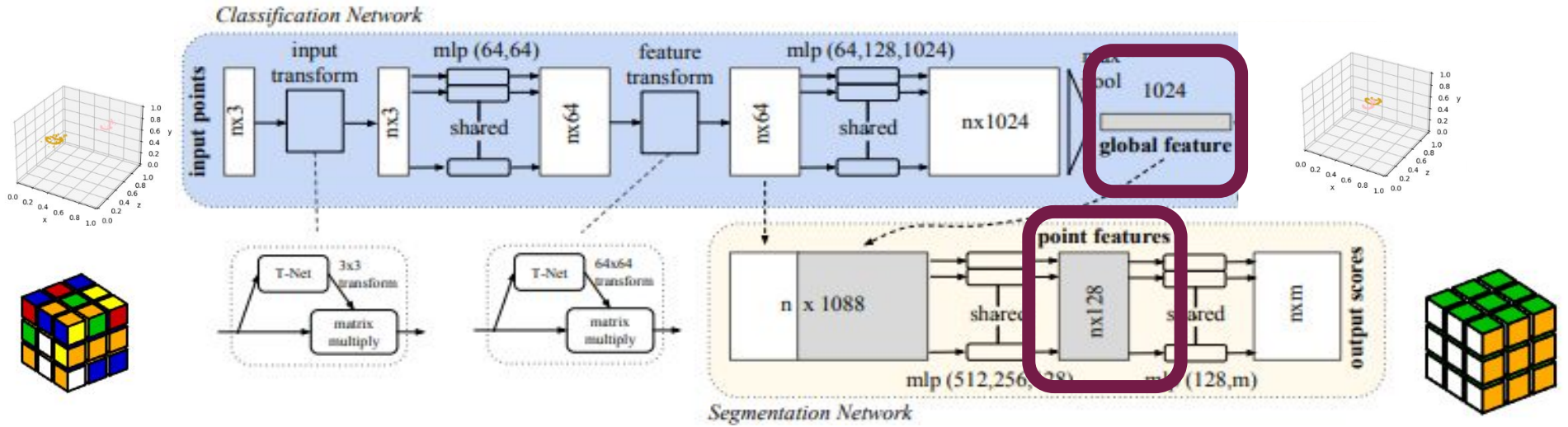
- Experimental data is populated with mostly events without reactions (these have no tracks; look random points scattered on a plane)
- This is a problem because these events are noise to our model
- Since beam events on average have less points than events with tracks we can filter out beam events easily





Methods: Approach

- Train on large amounts of unlabeled experimental data from various experiments
- We wanted to investigate the latent representations for event and track identification

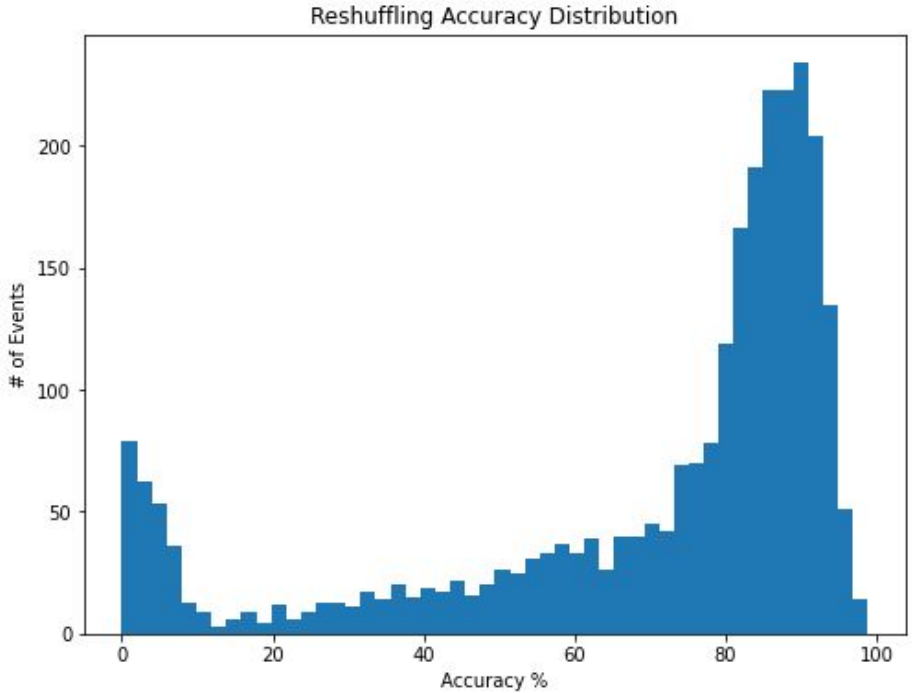


Current Project Achievements

- Train on large amounts of unlabeled experimental data
- Created a machine learning model that can reshuffle a point cloud at about 70% accuracy
- Got the model ready to have its latent spaces explored

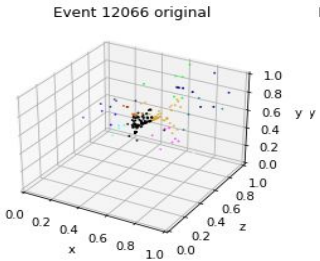
Results

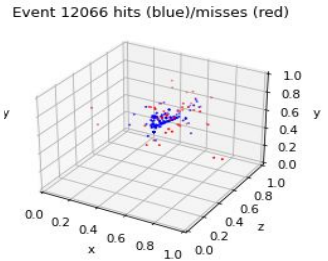
- Reconstruct accuracy is the % of the voxels reshuffled to the right place
- The mean reconstruct accuracy was 70%

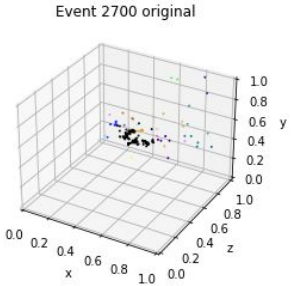


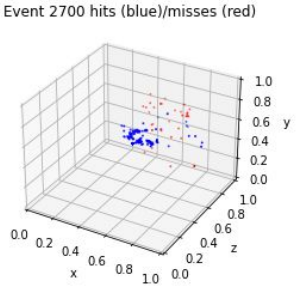
Results

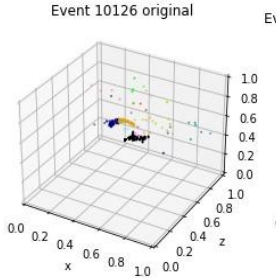
Event examples in 80-100% bin

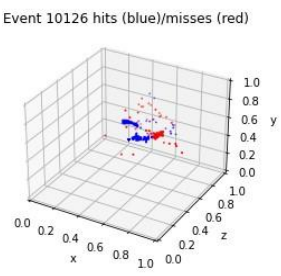




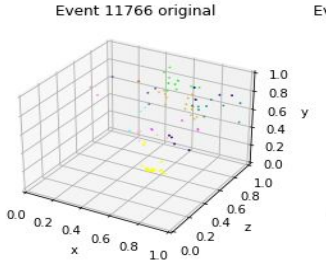


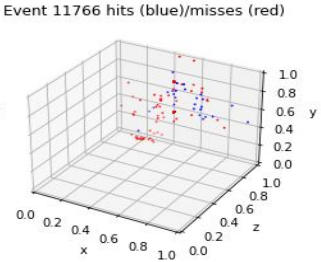


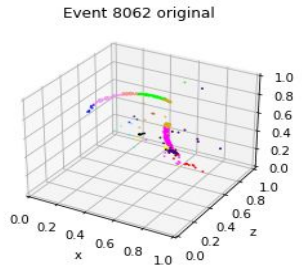


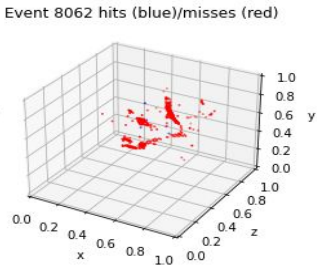


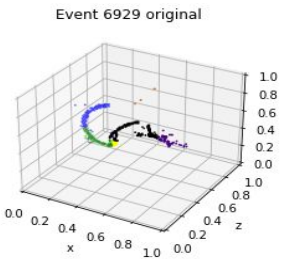
Event examples in 0-20% bin

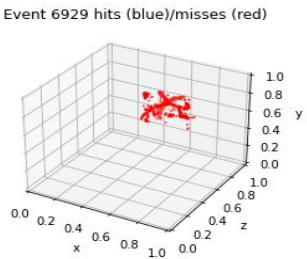






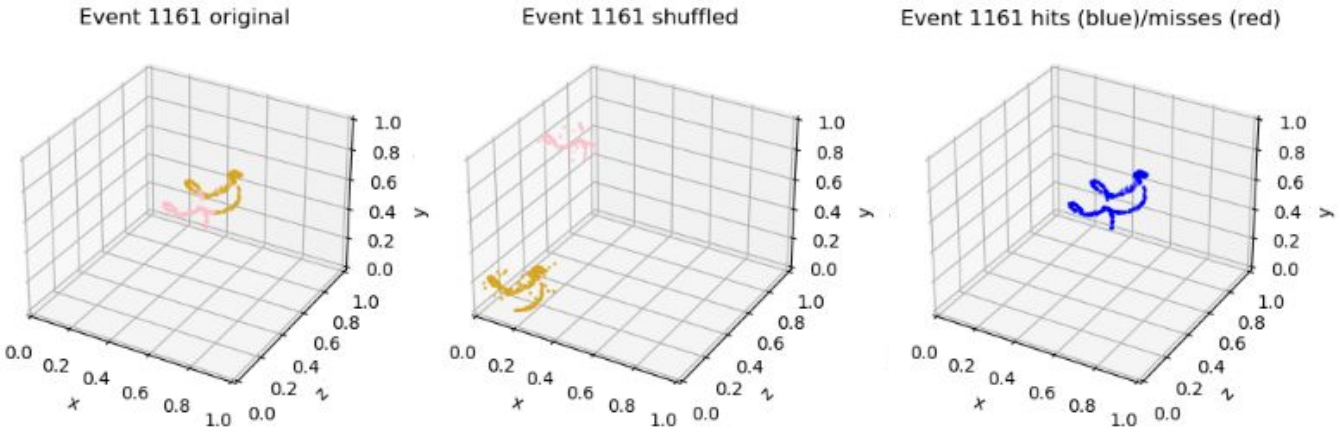






Results

^{22}Mg+α:



Next Steps

- Train on data from more experiments
- Fine-tune the model with a small amount of hand-labeled data
    - Does this match a model trained from scratch on a larger labeled dataset?

Thank you!

Point-cloud based ML

- PointNet:
  - The architecture operates on point cloud data and is permutation invariant
  - The model makes supervised point-wise predictions