Exploring Deep Set Neural Networks for Modeling Neutron Multi-Scatters
REU Program at Columbia University - Nevis Labs

Adeep Sen

Clemson University

August 02, 2026

Abstract

This work investigates whether a Deep Set neural network can approximate the contribution of each neutron-scatter cluster to the primary reconstructed S2 signal in XENONnT simulation. The model receives an unordered, variable-size set of clusters from each event and predicts a continuous value, p_{main}, for every cluster. A gated-sum Deep Set using event-relative inputs achieved a held-out test mean squared error of 0.0606 and $R^2 = 0.753$, substantially improving on both an electron-count argmax benchmark and a per-cluster multilayer perceptron. The improvement is strongest for the common endpoint targets, $p_{\text{main}} = 0$ and $p_{\text{main}} = 1$. Fractional targets remain the primary failure case: their test mean squared error is nearly twice the endpoint value, and predictions do not follow the identity line across the fractional range. Feature engineering and pooling studies identify event-relative electron and drift-time information as the main source of improvement. Event-weighted training and binary cross-entropy variants did not improve overall regression performance. The gated-sum model is therefore a useful ongoing study of event-context modeling, rather than a final solution for accurately reconstructing fractional p_{main} values.

Contents

1 Introduction

1.1 Dark Matter

Dark matter is one of the largest unsolved mysteries in modern physics. Although it is estimated to make up approximately 27% of the universe, it is invisible and has not yet been directly detected. Despite not knowing its composition, there is a wide range of observational evidence supporting its existence. For example, dark matter provides the additional gravitational attraction needed to explain the motion of stars within galaxies and the ability of galaxies to remain gravitationally bound. Its presence can also be inferred through gravitational lensing, in which light is bent by more mass than can be accounted for by visible matter alone.

The search for dark matter includes a wide range of theoretical candidates spanning roughly 50 orders of magnitude in mass. The XENON collaboration searches primarily for Weakly Interacting Massive Particles, commonly known as WIMPs. WIMPs are hypothetical, electrically neutral particles that are expected to interact through gravity and possibly the weak nuclear force. Because these interactions are extremely rare, detecting a WIMP requires a large, highly sensitive detector with very low background contamination.

1.2 The XENONnT Dark Matter Detector

The XENONnT dark matter detector is a dual-phase liquid xenon time projection chamber (TPC) designed to search for WIMP interactions [1, 2]. When a particle enters the detector and interacts with a xenon atom, the collision deposits energy into the liquid xenon. This energy produces prompt scintillation light and releases electrons from the surrounding atoms. The prompt scintillation light is detected by arrays of photomultiplier tubes and forms what is known as the S1 signal.

The released electrons are guided upward through the liquid xenon by an applied electric field. At the top of the TPC, the electrons are extracted into a thin layer of gaseous xenon, where a stronger electric field accelerates them and produces a second burst of proportional scintillation light. This second signal is known as the S2 signal. Information from the S1 and S2 signals can be used to reconstruct the location and energy of an interaction. The horizontal position is estimated using the pattern of light observed by the photomultiplier tubes, while the vertical position is determined from the time delay between the S1 and S2 signals.

Different types of particles produce different forms of energy deposition in the detector. Interactions caused by beta particles and gamma rays are generally classified as electronic recoils, while interactions involving neutrons or WIMPs are classified as nuclear recoils. The ratio of the S2 signal to the S1 signal provides a useful method for distinguishing electronic recoils from nuclear recoils. However, neutron and WIMP interactions both produce nuclear recoils and can therefore generate very similar detector signals.

The primary difference between the two is that neutrons scatter relatively easily and may interact multiple times while traveling through the detector. In contrast, a WIMP is expected to interact at most once because of its extremely small interaction probability. Neutron interactions are therefore an important source of background in the WIMP search. Accurately modeling the distribution of this background is necessary for determining whether a potential signal is consistent with a WIMP interaction or with a known background process [3].

Neutron-background distributions can be constructed using a combination of detector calibration data and simulation. A full-chain simulation of neutron multi-scatter events can be performed using Geant4 and FUSE. These simulations model the physical neutron interactions, the detector response, and the resulting reconstructed signals with high fidelity. However, this process has a significant computational cost, making it difficult to generate the large numbers of events needed for repeated studies.

There is therefore a need for a less computationally expensive method of modeling neutron multi-scatter events. In particular, it is useful to predict how individual neutron-scatter clusters contribute to the primary reconstructed S2 signal using only basic information about each cluster. A fast machine-learning model could learn this relationship from previously generated full-simulation events and then approximate the detector response without repeating the entire simulation chain.

1.3 Introduction to Deep Sets

To reconstruct the primary S2 signal from the properties of each scatter cluster, this project uses a neural network. Each neutron event contains a variable number of clusters, with each cluster described by features such as its horizontal position, mean drift time, drift-time spread, and electron count. The desired output is a value referred to as p_{main}, which represents the fraction of the primary S2 signal contributed by each cluster.

A standard neural network is not naturally suited to this problem. Events contain different numbers of clusters, and the prediction for one cluster may depend on the structure of the entire event. In addition, the ordering of clusters in the input should not affect the final prediction. A model should produce the same result regardless of whether an event is represented as $\{x_1, x_2, x_3\}$ or $\{x_3, x_1, x_2\}$. This property is known as permutation invariance.

To address these requirements, this work uses an architecture known as a Deep Set [4]. Deep Sets are designed to operate on unordered collections of varying size. A general permutation-invariant set function can be expressed as

$$f(X) = \rho\left(\sum_{i=1}^{N} \phi(x_i)\right), \tag{1}$$

where $X = \{x_1, \ldots, x_N\}$ is the set of clusters in an event. The function ϕ applies the same neural network to every cluster and produces a learned representation of each cluster. These representations are then combined using a permutation-invariant pooling operation, such as a sum. The function ρ uses the pooled representation to produce an output based on the overall event.

For this project, the architecture must produce a separate prediction for every cluster rather than a single prediction for the full event. Each cluster representation is therefore combined with a pooled event-level representation before being passed through a final prediction network. This allows the model to use both the properties of an individual cluster and the context provided by all other clusters in the event.

The goal of this work is to determine whether a Deep Set neural network can accurately predict the contribution of each neutron-scatter cluster to the primary S2 signal. The performance of the Deep Set model is compared with simpler baseline methods that do not use event-level context. The effects of feature engineering, pooling strategy, and model complexity are also investigated in order to identify the most effective architecture for approximating the full neutron simulation.

2 Dataset Characterization

The original simulation sample contains 2,704,607 neutron events and 10,259,121 scatter clusters. The 13 cm fiducial cut retains 2,044,825 events and 6,838,889 clusters. The retained events are typically small: the median event contains two clusters, the mean multiplicity is 3.34, 95% of events have at most 10 clusters, and 99% have at most 18 clusters. Although the largest event contains 89 clusters after the cut, the distribution is strongly concentrated at low multiplicity. This makes an event-based set representation computationally practical while preserving the physically meaningful variation in the number of scatters.

The label distribution is also highly imbalanced. In the fiducial sample, 3,610,619 clusters have $p_{\text{main}} = 0$ and 3,050,965 have $p_{\text{main}} = 1$. Only 177,305 clusters, or 2.59% of the sample, have a fractional label. This imbalance explains why a model can obtain a favorable overall MSE while still performing poorly on the fractional subset. It also motivates reporting separate endpoint and fractional metrics throughout this study.

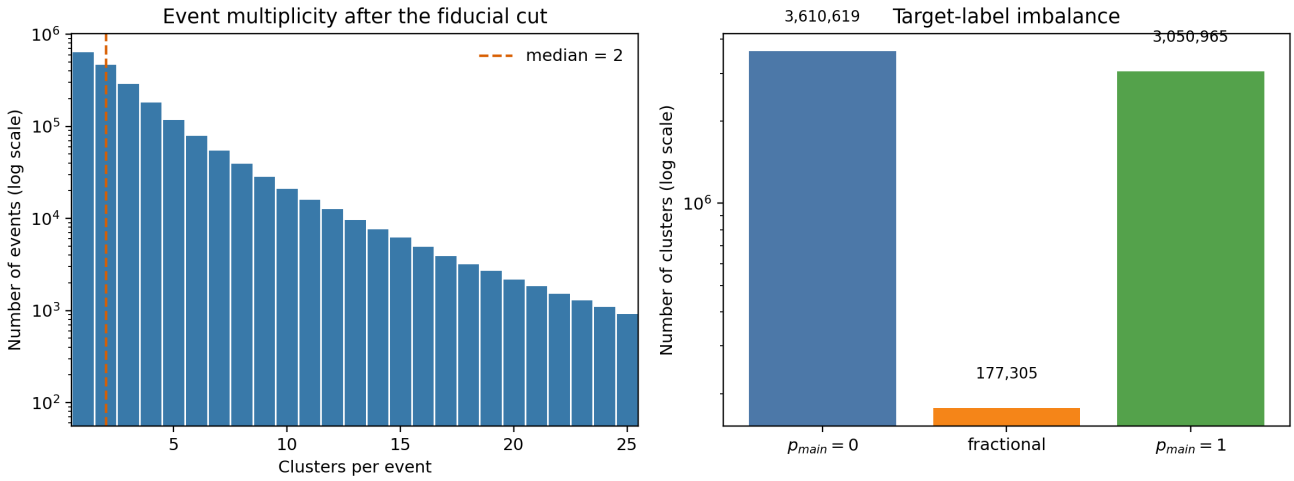

Figure 1: *Dataset characteristics after the fiducial cut. Left: event multiplicity on a logarithmic scale. Right: the p_{main} target distribution. Fractional labels are rare compared with the zero and one endpoints.*

3 Methodology

3.1 Simulation Data and Preprocessing

The model was trained using simulated neutron multi-scatter events. Each row of the data corresponds to one scatter cluster and includes the cluster's horizontal position, x and y, the number of electrons reaching the liquid-gas interface, the mean drift time, and the spread of the drift time. Clusters from the same neutron interaction share an event number and are treated as a single set. The training target is p_{main}, the fraction of the primary reconstructed S2 signal contributed by each cluster.

The original sample contained 2,704,607 events and 10,259,121 clusters. Before training, an event-level fiducial cut was applied to remove a simulation artifact near the top of the TPC. Any event containing a cluster with a mean drift time below 192,600 ns was removed. This cut removed 659,756 events and left 6,838,889 clusters. The remaining p_{main} values were restricted to the physical range from zero to one.

The data were divided by event number into 70% training, 15% validation, and 15% test samples. Splitting by event rather than by individual cluster ensures that clusters from the same neutron interaction do not appear in more than one sample. All input features were standardized using a StandardScaler fit only to the training data. The same scaling parameters were then applied to the validation and test data.

3.2 Input Features

The baseline model used the five cluster-level features described above. Additional features were constructed using only information available within each event. These features provide the model with simple relative comparisons between clusters without using the target value. The

final feature set contained 14 inputs: the five original cluster features, the logarithm of the event multiplicity, the cluster electron fraction relative to the event sum and maximum, the electron-count rank within the event, the difference between the cluster drift time and the event mean, the drift-time rank, the x and y offsets from the event centroid, and the distance from that centroid.

These event-relative features were included because the contribution of a cluster to the primary S2 signal depends not only on its individual properties but also on how it compares with the other clusters in the same event. All engineered features were calculated separately within each event, so the model can use them during inference without access to the true p_{main} values.

3.3 Deep Set Model

Each event was represented as a variable-size set of clusters. The model first applies the same encoder network, ϕ, to every cluster. The encoder contains five fully connected layers with 256 hidden units per layer and produces a 128-dimensional representation for each cluster. GELU activation functions and dropout were used between layers.

To construct event-level information, the encoded clusters were passed through a learned gate. The gate produces an independent sigmoid weight for each cluster, and the weighted cluster representations are summed to form a gated event embedding. Using independent sigmoid gates allows multiple clusters to contribute strongly to the event representation, which is appropriate for neutron multi-scatter events. The gated event embedding was then combined with the representation of each individual cluster.

The combined cluster and event representations were passed to a prediction network, ρ, consisting of four fully connected layers with 512 hidden units per layer. The network produces one output for each cluster. A sigmoid function was applied to this output so that the predicted p_{main} value remains between zero and one. Since the same encoder and prediction network are applied to every cluster and the event information is obtained through a sum, the model is unchanged if the order of the clusters is rearranged.

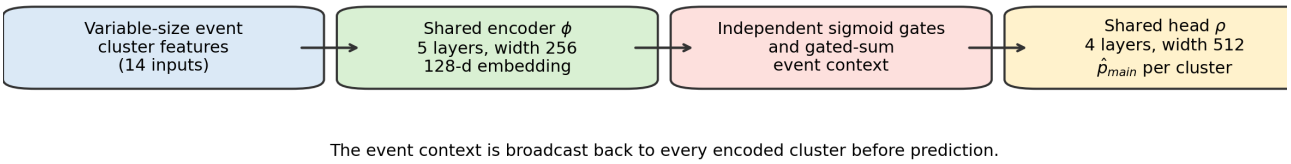

Figure 2: *Gated-sum Deep Set architecture. The shared encoder maps each cluster to a learned representation. Independent sigmoid gates form a permutation-invariant event context, which is combined with every cluster representation before the shared prediction head produces* $\hat{p}_{\mathrm{main}}$.

3.4 Training and Model Selection

The model was trained to minimize the mean squared error between the predicted and simulated p_{main} values. Adam optimization was used with a learning rate of 3×10^{-4}, a weight decay of 1.08×10^{-5}, and a batch size of 1024 events. Training was allowed to continue for up to 300 epochs, with a learning-rate scheduler and early stopping based on the validation mean squared error.

The best checkpoint was selected using the minimum validation p_{main} mean squared error. The held-out test set was not used to choose input features, pooling strategies, or network sizes. The final model configuration was selected by comparing the validation performance of the baseline features, the event-relative feature sets, and several pooling strategies. The final configuration used all 14 features and gated-sum pooling because it achieved the lowest validation mean squared error among the corrected-data architecture comparisons.

4 Results

4.1 Comparison with Benchmarks

The gated-sum Deep Set was evaluated on the held-out event split using the checkpoint selected by validation mean squared error. The test split was not used to select the architecture, input features, or training procedure. Table 1 compares the model with two simpler benchmarks. The electron-count argmax assigns $p_{\text{main}} = 1$ to the cluster with the largest electron count in each event and assigns zero to all other clusters. The per-cluster MLP uses the five raw cluster inputs but cannot compare one cluster with the other clusters in the same event.

The gated-sum Deep Set has a test mean squared error of 0.0606 and $R^2 = 0.753$. This is a substantial improvement over the electron-count argmax benchmark, which has mean squared error 0.2293 and $R^2 = 0.064$, and over the per-cluster MLP, which has mean squared error 0.1862 and $R^2 = 0.240$. The Deep Set also improves strict event-level identification of the main cluster from 0.726 for the electron-count baseline and 0.721 for the MLP to 0.740. These results show that electron count is a useful first approximation, but that information about the full event is needed for accurate per-cluster regression.

Model	**Test MSE**	**Test MAE**	R^2	**Strict event accuracy**
Electron-count argmax	0.2293	0.2322	0.064	0.726
Per-cluster MLP	0.1862	0.3744	0.240	0.721
Gated-sum Deep Set	**0.0606**	**0.1193**	**0.753**	**0.740**

Table 1: *Held-out test performance. The Deep Set checkpoint was fixed using validation performance before this evaluation. Lower MSE and MAE are better; higher R^2 and event accuracy are better.*

4.2 Endpoint and Fractional p_{main} Predictions

The overall result is dominated by endpoint labels. Of the 1,027,230 clusters in the held-out test sample, 1,000,683 have $p_{\text{main}} = 0$ or $p_{\text{main}} = 1$, while only 26,547 have values strictly between zero and one. The gated-sum model performs much better on the endpoint targets than on the fractional targets. Its endpoint mean squared error is 0.0591 and mean absolute error is 0.1158. For fractional targets, the mean squared error rises to 0.1169 and the mean absolute error rises to 0.2496.

Figure 3 illustrates this difference. The Deep Set predictions are concentrated near the correct values for the large populations at zero and one. In contrast, the predictions for intermediate true values are broadly distributed and do not follow the red identity line. The model is therefore useful for identifying clusters that are likely to make no contribution or the dominant contribution to the primary S2 signal, but it does not yet reconstruct the continuous fractional contribution reliably.

Figure 4 compares the MLP and Deep Set separately in the two target regimes. The Deep Set lowers error in both regimes, but the fractional error remains much larger than the endpoint

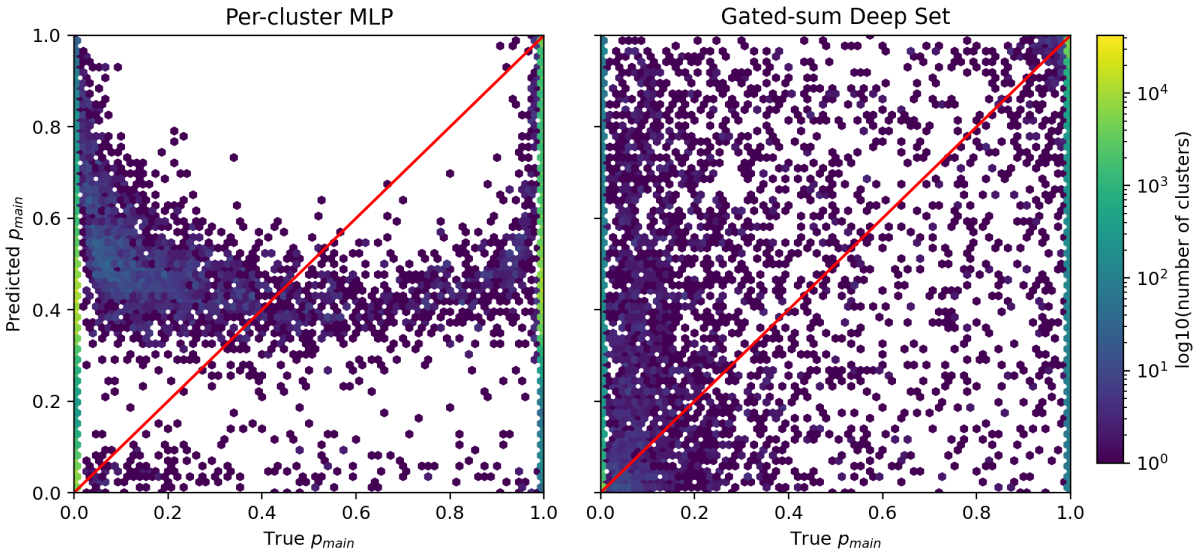

Figure 3: *Held-out predicted-versus-true p_{main} distributions for the per-cluster MLP and the gated-sum Deep Set. The red line is the ideal prediction. The Deep Set more clearly separates the zero and one target populations, but both plots show a broad prediction distribution for fractional targets.*

error. This behavior is important because an acceptable overall regression score can hide poor performance on the smaller, physically interesting fractional population.

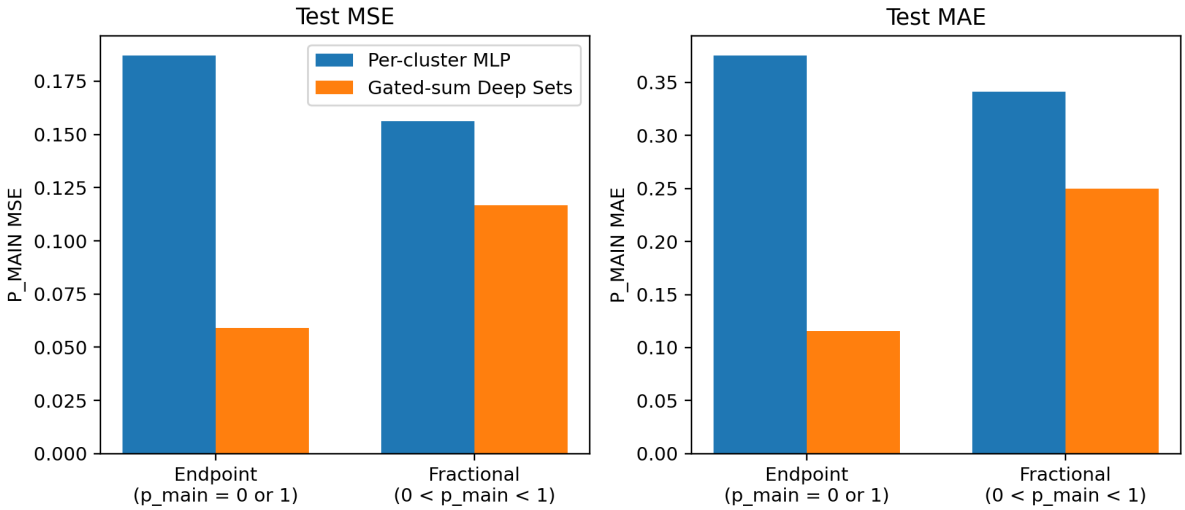

Figure 4: *Held-out MSE and MAE for endpoint and fractional p_{main} targets. The Deep Set improves both regimes relative to the per-cluster MLP, but the fractional-target error remains the main weakness.*

4.3 Audit of Failure Cases

The fractional cases are not simply a small extension of the endpoint problem. A grouped permutation audit of the gated-sum checkpoint shows that electron count remains the most important source of information for both all targets and fractional targets. However, the importance ranking changes in the fractional subset. Drift-time spread becomes the third most important original input for fractional targets, while the transverse coordinates contribute less than they do

overall. This suggests that the within-cluster drift-time distribution contains information that is particularly relevant to the fractional cases.

The audit also indicates that the available input variables do not fully distinguish all of the ambiguous events. A nearest-neighbor study in the 14-dimensional engineered feature space found substantial disagreement between labels in some local neighborhoods, especially in regions associated with fractional targets. This is consistent with missing explanatory information or label stochasticity, but it does not prove that a fixed error floor has been reached. The current input set can predict the common endpoint behavior well while still failing to resolve the physical differences that determine a fractional contribution.

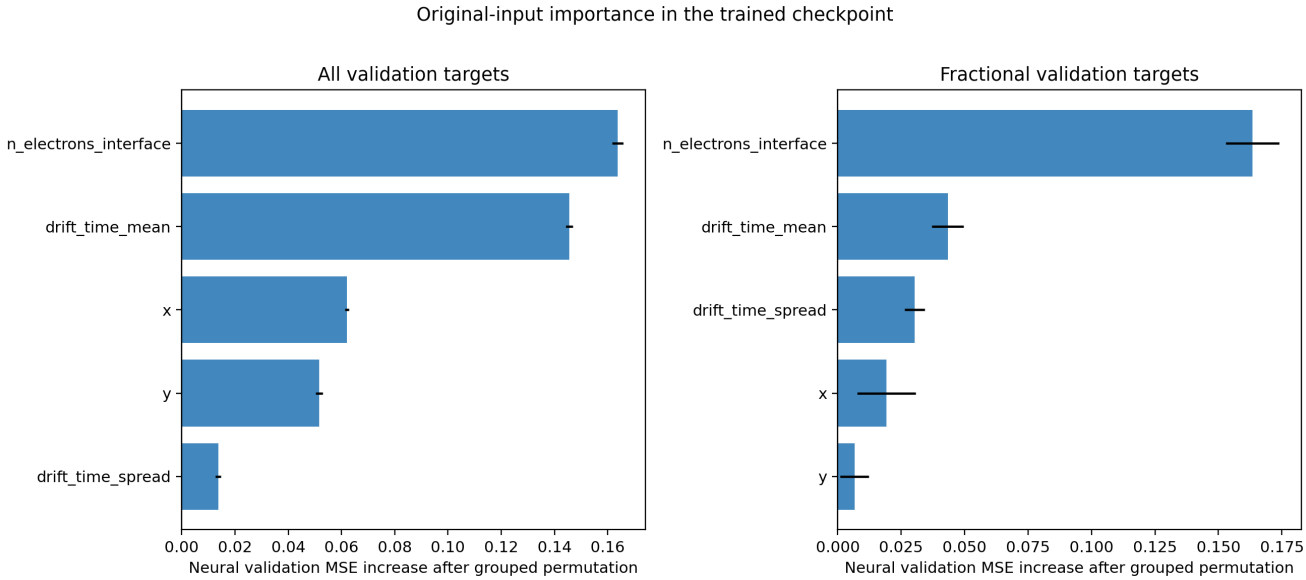

Figure 5: *Grouped permutation importance for the gated-sum validation checkpoint. Permuting a source variable also recomputes event-relative features derived from that variable. Electron count is dominant in both panels; drift-time spread becomes more important for fractional targets.*

To investigate this ambiguity further, a tree-based nearest-neighbor diagnostic was performed in the standardized 14-dimensional engineered feature space. For the all-event-relative inputs, the mean variance of the true labels among ten nearest training neighbors was 0.12595. This is lower than the corresponding five-feature value of 0.16851, showing that event-relative inputs make similar events more consistent. However, the remaining variance is still substantial, and regions with large neighbor-label variance also have larger prediction error. The diagnostic therefore supports the conclusion that the fractional failure cases are associated with incomplete information in the available observables, rather than only with insufficient network width.

4.4 Feature Engineering and Pooling Studies

The largest architecture improvement came from adding event-relative features, rather than from making the network wider. Using the 14-input feature set with ordinary sum pooling reduced the validation MSE from 0.06520 for the five-raw-input baseline to 0.06073. These features include electron fractions and ranks, drift-time offsets and ranks, event multiplicity, and relative transverse geometry. They make explicit the comparisons between clusters that a per-cluster model cannot make.

Several pooling options were then evaluated using the same corrected data, event split, optimizer, and validation metric. The independent gated-sum pool achieved the lowest validation MSE. It is only a small improvement over ordinary sum pooling, and the result should be repeated over additional seeds before it is treated as a definitive pooling result. More complex pooled statistics and a much wider network did not justify their additional parameters.

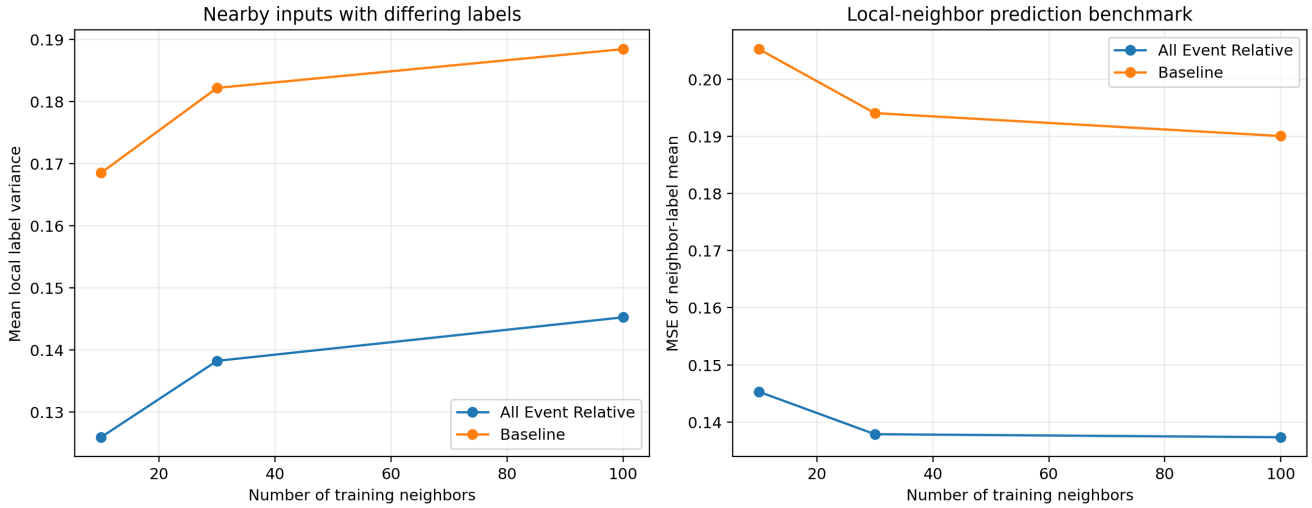

Figure 6: *Conditional-label-variance diagnostic for tree-based models. Event-relative features reduce local label disagreement and prediction error, but ambiguous neighborhoods remain. This is evidence consistent with missing explanatory inputs or stochastic labels, not a proof of an irreducible error floor.*

Configuration	Parameters	Best validation MSE
Five raw inputs with sum pooling	1,217,665	0.06520
14 event-relative inputs with sum pooling	1,219,969	0.06073
14 inputs with sum/mean/max pooling	1,351,041	0.06051
14 inputs with gated-sum pooling	1,236,610	**0.06014**
14 inputs with sum/mean/max plus gated pooling	1,433,218	0.06020
14 inputs with a wider network	4,864,769	0.06211

Table 2: *Validation-only feature and pooling comparison. The gated-sum result is promising but should be confirmed with more seeds.*

4.5 Attempted Training-Data Tuning

Because fractional labels are rare, an event-weighted sampling study was used to expose the model to fractional events more frequently during training. An event was considered fractional if it contained at least one target between 0.001 and 0.999. These events made up 7.75% of training events. Weighting them by a factor of two increased their expected sampling rate to 14.39%, and weighting them by a factor of four increased it to 25.15%.

Neither weighting strategy improved the overall validation regression objective. With the same all-event-relative inputs and sum/mean/max pooling, weight two obtained validation MSE 0.06142 and weight four obtained 0.06314, compared with 0.06051 for the unweighted sum/mean/max model. A binary-cross-entropy-with-logits variant with weight four was also worse for the MSE target, with a final validation MSE of approximately 0.0632. These tests do not show that reweighting can never help fractional predictions; rather, the simple event-level weighting choices tested here traded away overall regression performance and were not retained.

5 Summary and Conclusions

The results show that event context is useful for predicting p_{main}. The gated-sum Deep Set substantially outperforms an electron-count argmax rule and a cluster-only MLP on the held-out test sample. It is especially effective for the common zero and one labels, where it learns a useful approximation of whether a cluster is absent from or dominant in the primary S2 signal.

The main limitation is the model's inability to accurately learn the fractional p_{main} values. Fractional predictions have approximately twice the endpoint MSE, and their predicted-versus-true distribution remains broad rather than following the identity line. This failure persists even though the model improves on the MLP in the fractional subset. The results should therefore be interpreted as a successful demonstration that Deep Sets can use event-level information, not as a final continuous reconstruction model.

The next study should focus directly on the fractional cases. First, simulation truth variables or reconstructed observables that distinguish ambiguous fractional neighborhoods should be audited and added only if they are available in the intended inference setting. Second, the fractional target distribution could be modeled with a dedicated conditional head, a mixture-density or uncertainty-aware loss, or a loss that evaluates fractional calibration separately from endpoint classification. Third, full-feature sum and gated-sum pooling should be repeated over additional random seeds to establish whether the small gated-pooling gain is reproducible. Finally, all future comparisons should retain the event-level split and reserve the held-out test data for a single final evaluation.

6 Acknowledgements

I would like to thank my advisor, Pueh Leng Tan, for their guidance, feedback, and support throughout this project. I also thank Professor Elena Aprile, the members of the XENON collaboration, and Nevis Laboratories for the opportunity to contribute to this work. This material is based upon work supported by the National Science Foundation under Grant No. PHY-2349438.

A Experiment Configuration and Reproducibility

All model comparisons used the same event-level split and training-only feature standardization. The held-out test split was used only for the final benchmark comparison. Validation MSE was used for learning-rate scheduling, early stopping, and checkpoint selection. This separation is important because the test performance in Table 1 is intended to estimate generalization rather than guide further architecture choices.

Setting	**Value**
Input features	14
Encoder depth / width	5 / 256
Latent dimension	128
Prediction-head depth / width	4 / 512
Dropout	0.0485
Optimizer	Adam
Learning rate	3×10^{-4}
Weight decay	1.08×10^{-5}
Batch size	1024 events
Gradient-clip norm	1.0
Maximum epochs	300
Model-selection metric	Validation p_{main} MSE

Table 3: *Configuration of the gated-sum Deep Set study.*

The analysis is reproducible from the event-level data preparation, feature definitions, model-training scripts, and saved checkpoints in the project repository. The figures in this report are generated from frozen checkpoints and archived validation or test outputs. The code records the random seed, feature set, fiducial-cut statistics, and checkpoint metric for each experiment so that

a future study can repeat the selected configurations and compare new fractional-target methods against the same baseline.

References

[1] E. Aprile et al. The XENONnT dark matter experiment. *The European Physical Journal C*, 84(8), August 2024.

[2] E. Aprile et al. XENONnT analysis: Signal reconstruction, calibration, and event selection. *Physical Review D*, 111(6):062006, March 2025.

[3] E. Aprile et al. XENONnT WIMP search: Signal and background modeling and statistical inference. *Physical Review D*, 111(10):103040, May 2025.

[4] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. *CoRR*, abs/1703.06114, 2017.