Interpretable Joint Event-Particle Reconstruction for Neutrino Physics at NOvA with Sparse CNNs and Transformers

Alexander Shmakov [*] [†]
School of Information and Computer Sciences
University of California Irvine
Irvine, CA, USA

Alejandro Yankelevich [*]
Department of Physics and Astronomy
University of California Irvine
Irvine, CA, USA

Pierre Baldi
School of Information and Computer Sciences
University of California Irvine
Irvine, CA, USA

Jianming Bian
Department of Physics and Astronomy
University of California Irvine
Irvine, CA, USA

For the NOvA Collaboration

Abstract

The complex events observed at the NOvA long-baseline neutrino oscillation experiment contain vital information for understanding the most elusive particles in the standard model. The NOvA detectors observe interactions of neutrinos from the NuMI beam at Fermilab. Associating the particles produced in these interaction events to their source particles, a process known as reconstruction, is critical for accurately measuring key parameters of the standard model. Events may contain several particles, each producing sparse high-dimensional spatial observations, and current methods are limited to evaluating individual particles. To accurately label these numerous, high-dimensional observations, we present a novel neural network architecture that combines the spatial learning enabled by convolutions with the contextual learning enabled by attention. This joint approach, TransformerCVN, simultaneously classifies each event and reconstructs every individual particle's identity. TransformerCVN classifies events with 90% accuracy and improves the reconstruction of individual particles by 6% over baseline methods which lack the integrated architecture of TransformerCVN. In addition, this architecture enables us to perform several interpretability studies which provide insights into the network's predictions and show that TransformerCVN discovers several fundamental principles that stem from the standard model.

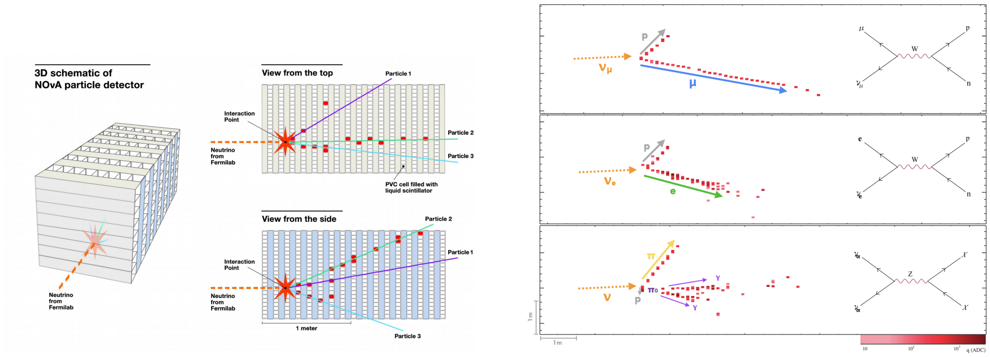
Introduction As machine learning becomes integral to physical sciences, the interpretability of deep neural networks becomes increasingly significant. The surge in particle physics data demands machine learning to extract meaningful results, and creating interpretable neural networks is essential for generating insights from this data. We focus on a common problem in particle physics known as *reconstruction*, which translates high-dimensional observations into more fundamental constructs.

Problem Statement NOvA [1], a long-baseline neutrino oscillation experiment using Fermilab's NuMI beam, comprises two liquid scintillator detectors 809km apart, which may be used to perform

[*] These authors contributed equally.

[†] Corresponding Author: ashmakov@uci.edu

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.



(a) Schematic of NOvA detector and generation of top and side views from vertical and horizontal planes respectively.

(b) Typical far detector event displays of ν_μ CC (top), ν_e CC (middle), and NC (bottom) neutrino interactions.

Figure 1: Visual representations of the input data at the NOvA Detector.

a 3D reconstruction of particle tracks. The NOvA experiment focuses on measuring neutrino oscillations by observing the disappearance of ν_μ and appearance of ν_e at the far detector through charged current interactions (Figure 1b). A previous oscillation analysis [2] used a Convolution Visual Network (CVN) [3] to classify events into categories like ν_μ CC, ν_e CC, NC, and cosmogenic background using $100 \times 80 \times 2$ images, known as *pixel-maps*, from the detector centered around the reconstructed event. Another network, ProngCVN [4], identifies individual particles, using both the complete event image and an image containing only a individual particle's energy deposition according to NOvA's reconstruction. We provide an overview of other related work in Appendix A.

Claims We introduce a hybrid convolution and transformer-based architecture known as TransformerCVN (T-CVN) that makes use of NOvA's existing particle reconstruction to simultaneously process all particles within an event in a single network. This joint input scheme makes it possible to used contextual information to improve prong reconstruction and to describe the topological features of a particle's track that leads to its classification and to analyze the relative impact of each reconstructed particle on the overall event classification.

Transformer CVN We combine Sparse CNNs with Transformers in order to address both the pixel-map sparsity and the variable prong counts in each event. We introduce *TransformerCVN*, which embeds the sparse images into a dense latent space using Minkowski sparse CNNs [5] before processing the the embeddings with transformer encoders [6] to include contextual information. The individual prong pixel-maps are associated their respective particle labels while the event pixel-map is associated with an overall neutrino interaction type.

Sparse CNN Architecture We employ a DenseNet-based architecture [7] for pixel-map embedding which includes weighted skip connections between all CNN layers to counteract network activation decay from input sparsity. We replace standard convolution and pooling in DenseNet with sparse variants of these operations [5] which only compute convolutions for non-zero center pixels and not introducing new non-zero values between these operations [8]. We use one instance of this CNN to embed prong pixel-maps, sharing weights among prongs, while a distinct event CNN processes the event pixel-map. Each $100 \times 80 \times 2$ pixel-map is then transformed into an *embedded pixel-map* latent vector who's dimensionality determined by a hyperparameter. These CNNs form the network's initial stage as depicted in Figure 2.

Transformer Architecture We process the embedded pixel-maps with a transformer encoder [6] to allow contextual information to be shared between prongs. Both prong and event embedded pixel-map are *encoded* using a single, shared transformer encoder stack, visible in Figure 2. This transformer encoder follows the canonical formulation [6], with one major exception. We elect to not add position embeddings to our latent pixel-maps to avoid imposing an inherent order over

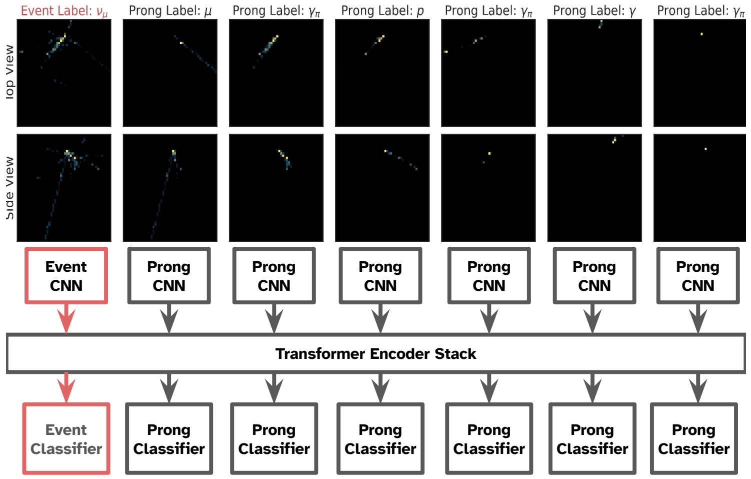


Figure 2: A complete diagram of Sparse Transformer CVN, including example pixel-maps from a ν_μ event with the event pixel-map path is highlighted in red.

pixel-maps; and, instead, simply add a *type* embedding to differentiate prong and event pixel-map by concatenating the embeddings with one of two trainable context vectors.

Classification Outputs After embedding and encoding the event and prong pixel-maps, the encoded vectors are fed through feed-forward networks to produce the final reconstruction predictions. We employ a unified weight-sharing network for all prong reconstructions, and a distinct network for event classification. The prong networks yield softmax distributions for 9 possible prong targets (detailed in) for each input prong. Meanwhile, the event network generates a softmax distribution over 10 possible interaction types, which can be condensed to represent four main event types during inference.

Training We train both the prong reconstructions and event classifications with categorical log-likelihood losses across all event and prong pixel-maps. Given the multiple classification targets per event and an imbalanced target distribution, we incorporate a focal loss [9] term to amplify weights on under-performing events. This addition enhances accuracy for secondary, non-leptonic prongs. Training specifics and hyperparameters are detailed in Appendix B. These values were selected after tuning on the architecture via Bayesian Optimization search as part of SHERPA [10], maximizing the average of event and prong accuracy.

Dataset We trained TransformerCVN on neutrino interaction simulations within the NOvA far detector, splitting the simulated events equally between the unoscillated predominantly ν_μ beam and ν_e oscillated events. The simulations employed GENIE [11] for neutrino-nucleus interactions and GEANT4 [12] for detector simulations, overlaying neutrino actions on real cosmogenic background data. The dataset derives from NOvA's 5th Monte-Carlo simulation run in 2020 [13]. Training followed the pre-selection criteria used in NOvA's prior EventCVN analysis, inclusive of cosmic ray rejection and a transverse momentum fraction cut [2]. We partitioned data into $6,316,264$ training events, $332,434$ validation events for hyper-parameter tuning, and $177,084$ testing events for model evaluation. Events had a median of 2 prongs, with 90% having 1-6 prongs, leading us to limit training events to the 10 highest-energy prongs for storage efficiency.

Event Classification Event classification targets where assigned to be one of 10 possible labels T_{event} = {ν_μ CC QE, ν_μ CC Res, ν_μ CC DIS, ν_μ CC Other, ν_e CC QE, ν_e CC Res, ν_e CC DIS, ν_e CC Other, NC, CB} where QE is quasi-elastic scattering, Res is the resonant interaction, and DIS is deep inelastic scattering, NC is neutral current events, and CB is the cosmic ray background. However, we find that the network cannot effectively separate charged current events based on their interaction type. We follow the EventCVN [3] baseline and collapse the event labels into these four basic classes

Metric	T-CVN	EventCVN
Accuracy	0.894	0.897
Precision	0.894	0.908
Recall	0.894	0.897
ROC AUC	0.982	0.984

Table 1: Event reconstruction aggregated metrics

Metric	T-CVN	Prong CVN
Accuracy	0.783	0.726
Precision	0.783	0.760
Recall	0.783	0.726
ROC AUC	0.951	0.932

Table 2: Prong reconstruction aggregated metrics.

for evaluation: $T_{EventCVN} = \{\nu_e, \nu_\mu, NC, CB\}$. Background events originally overwhelmingly dominated the training dataset, so we down-sample CB events to bring the distribution of event targets to be roughly uniform between the three signal classes and limit CB to only 10% of the training dataset.

Prong Reconstruction Prong reconstruction targets were assigned from the possible set of $T_{prong} = \{e, \mu, p, \gamma_n, \pi^\pm, \gamma_{\pi^0}, \gamma_{other}, OP, CB\}$. In an attempt to identify neutrons and neutral pions, which do not deposit energy in the scintillator, the photon class was split into $\{\gamma_n, \gamma_{\pi^0}, \gamma_{other}\}$ where γ_{π^0} refers to a photon with a mother π^0. Prongs produced in neutrino interactions corresponding to particles other than the above classes where given the other prong, OP, class. Prongs from data cosmic ray tracks do not have truth information and so were given the CB class.

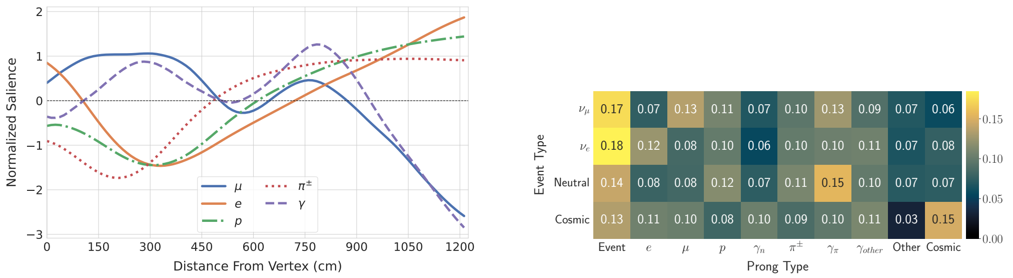
Performance We assessed TransformerCVN's baseline performance on event and prong reconstruction, presenting average metrics in Tables 1 and 2. Metrics were averaged using a one-versus-one approach with re-weighted balanced classes. Confusion matrices for both reconstructions are shown in Figures 13 and 17. Event classification achieves high accuracy of nearly 90 % with some confusion distinguishing ν_e from NC events. Prong reconstruction shows similar promise, although Protons were frequently classified, often confused with photons. We also notice that the mother-particle sub-classes are challenging to separate and are often confused for each other.

Comparison to EventCVN Event classification performance was near identical to the NOvA EventCVN, but TransformerCVN demonstrated more balanced predictions with matching precision and recall scores. Both of these models went through extensive hyper-parameter tuning and iterations and achieve similar event performance with substantially different architectures. We therefore hypothesize that both models seem to be at peak performance for the dataset and additional features or resolution will be necessary to improve event classification.

Comparison to ProngCVN TransformerCVN exhibited marked improvements in prong reconstruction. The additional context stemming from the attention mechanism boosted prong AUC for major particle types over the ProngCVN. We noted a 0.02 rise in average AUC and a near 5% hike in reconstruction accuracy. The most notable enhancement was in lepton prong reconstruction, where the μ prong AUC improved from 0.864 to 0.975 (Figure 12).

Interpretability Deep neural networks are often dubbed *black boxes* due to their opaque decision-making processes which pose challenges in understanding their predictions. However, the unified TransformerCVN architecture can shed some light on this black box. Examining the learned attention mechanism offers insights into relationships between different particles and event types. Additionally, by reverse-engineering the CNN's learned spatial structures, we gain an understanding of how the network distinguishes between particles. Through these interpretability studies, we find evidence suggesting the network learns several known principles from the standard model.

Prong Attention Maps Attention-based networks let us visualize the attention weights for each input, indicating the significance of each input to the output. Though individual attention weights can be noisy and difficult to interpret, aggregating them over multiple events yields clearer insights. Each prong is categorized by its truth labels with event pixel-maps treated as pseudo-prongs. We average the *logit*-attention scores for every prong type to estimate pair-wise significance of each type for reconstruction. Our results, presented in Figure 4, display a primarily diagonal pattern, but we notice event pixel-map inputs show higher attention scores. A detailed overview of this method is presented in Appendix C.



(a) Integrated salience map demonstrating the trace profile of different prong types.

(b) Event attention scores measuring the importance of different prongs for event classification.

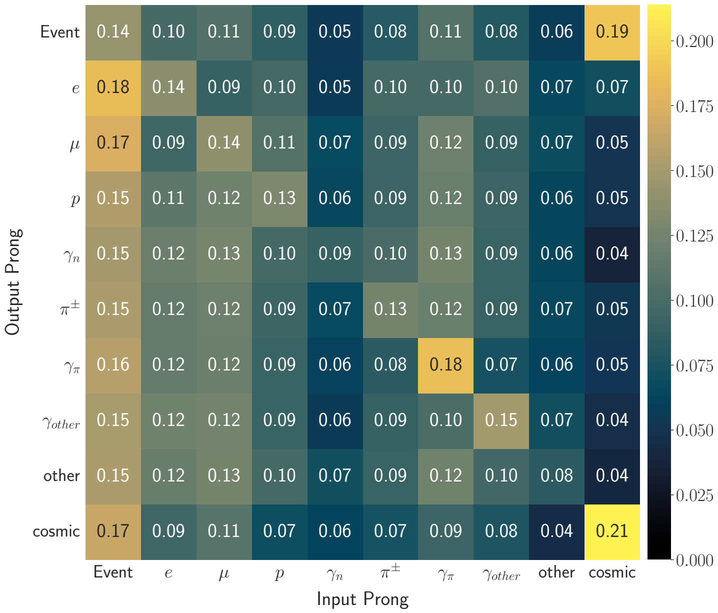
Figure 3: Aggregated Interpretation plots.

Event Attention Maps We analyze event attention scores using a similar aggregation technique, aggregating scores by the true event label instead of the prong type. Displayed in Figure 3b, we observe that leptons unique to the charged current interaction types significantly influence event-level predictions, such as elections for ν_e CC and muons for ν_μ CC.

Saliency Maps Saliency maps [14] help interpret convolution networks' learned behavior by highlighting the sensitivity of a network's output to each input pixel. These maps illustrate how a slight change in input pixel intensity affects the output. We generated saliency maps for all prong and event outputs concerning their input pixel-maps. However, like attention maps, individual event saliency are noisy due to input sparsity. To derive clear insights, we aggregated saliency maps across events, detailed in Appendix E. We present a grid showcasing saliency maps for five key prong particle labels 5. Diagonal maps highlight the saliency for each particle's associated output, while off-diagonal maps display areas most influencing predictions towards the Positive Particle label over the Negative Particle. One would expect that muons are the easiest to visually identify from the other particle classes considered here due to their tendency to leave tracks rather than showers. It is clear from the muon rows that hits near the vertex make the network more likely to classify the prong as a muon track as opposed the shower of a showering particle. μ/γ separation presents an interesting example where hits at large angles from the prong direction as opposed to hits along the direction vector far from the vertex make γ classification more likely. γ separation from e usually relies on a predicted gap between the vertex and the start of the photon's shower. Hits far the vertex therefore contribute more to γ classification. However, the e/γ separation plots appear more isotropic than μ/γ or μ/e plots, likely because both types of particles are expected to shower.

Integrated Saliency Maps We may also compare the track profile between the different classes by integrating the saliency maps presented above across the width of the detector to produce one-dimensional saliency with respect to vertex distance along the track (Figure 3a). This provides the average "pattern" we expect every prong type to form and indicates the most important regions for each prong class. We again notice that, as expected from theory, the muon track extends further than the other prong types, remaining flat for the middle third of the track. We also notice the delayed hit expected from γ tracks when compared to e.

Conclusion We present a novel neural network architecture for event reconstruction at NOvA. By combining the spatial correlation learning of sparse convolution networks with the contextual learning of transformers, we present a method for simultaneously reconstructing both individual prong labels as well as an overall event classification. This combined approach improves reconstruction accuracy over baseline methods while also providing many novel methods for interpreting the network's reasoning behind individual reconstructions. The black-box nature of neural networks is often an uneasy aspect of deep learning models, we a method for "opening the black box" provides a method for increasing trust in the network's predictions and guides our understanding of the underlying physics. Interpretable networks are critical for not just improving current physics experiments, but for guiding our understanding in designing new experiments.



	Event	e	μ	p	γ_n	$\pi^{\pm}$	γ_π	γ_{other}	other	cosmic
Event	0.14	0.10	0.11	0.09	0.05	0.08	0.11	0.08	0.06	0.19
e	0.18	0.14	0.09	0.10	0.05	0.10	0.10	0.10	0.07	0.07
μ	0.17	0.09	0.14	0.11	0.07	0.09	0.12	0.09	0.07	0.05
p	0.15	0.11	0.12	0.13	0.06	0.09	0.12	0.09	0.06	0.05
γ_n	0.15	0.12	0.13	0.10	0.09	0.10	0.13	0.09	0.06	0.04
$\pi^{\pm}$	0.15	0.12	0.12	0.09	0.07	0.13	0.12	0.09	0.07	0.05
γ_π	0.16	0.12	0.12	0.09	0.06	0.08	0.18	0.07	0.06	0.05
γ_{other}	0.15	0.12	0.12	0.09	0.06	0.09	0.10	0.15	0.07	0.04
other	0.15	0.12	0.13	0.10	0.07	0.09	0.12	0.10	0.08	0.04
cosmic	0.17	0.09	0.11	0.07	0.06	0.07	0.09	0.08	0.04	0.21

Figure 4: Aggregated attention matrices measuring the impact of different prong types to various predictions.

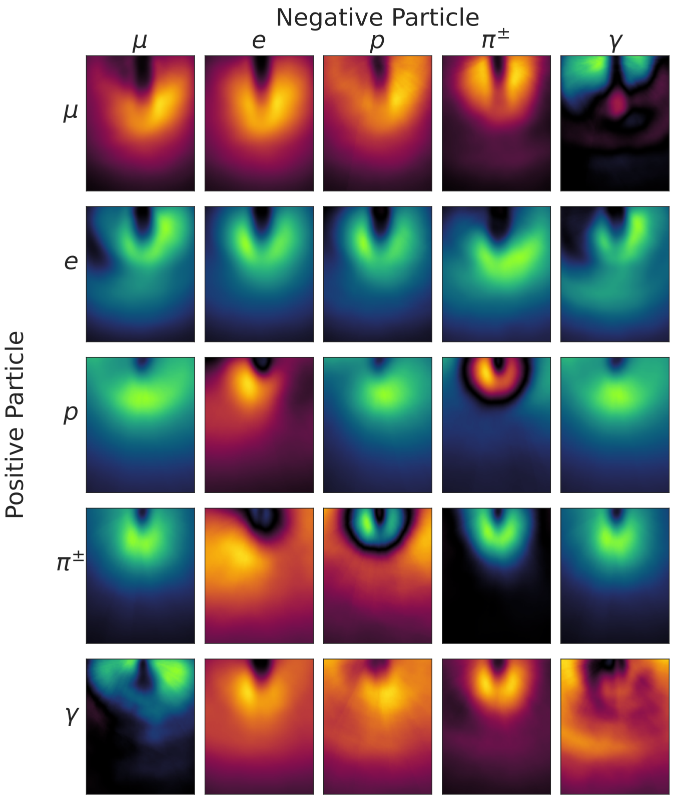


Figure 5: Grid of aggregated saliency maps and difference maps for every pair of prong types. Each row only contains pixel-maps matching the Positive Particle's class. Red indicates a positive correlation with the Positive Particle's reconstruction probability, while blue represents a negative correlation.

References

[1] D. S. Ayres et al. The NOvA Technical Design Report. 10 2007.

[2] M. A. Acero et al. An Improved Measurement of Neutrino Oscillation Parameters by the NOvA Experiment. 8 2021.

[3] A. Aurisano, A. Radovic, D. Rocco, A. Himmel, M. D. Messier, E. Niner, G. Pawloski, F. Psihas, A. Sousa, and P. Vahle. A Convolutional Neural Network Neutrino Event Classifier. *JINST*, 11(09):P09001, 2016.

[4] F. Psihas, E. Niner, M. Groh, R. Murphy, A. Aurisano, A. Himmel, K. Lang, M. D. Messier, A. Radovic, and A. Sousa. Context-Enriched Identification of Particles with a Convolutional Network for Neutrino Events. *Phys. Rev. D*, 100(7):073005, 2019.

[5] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, July 2017.

[8] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. *CVPR*, 2018.

[9] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society.

[10] Lars Hertel, Julian Collado, Peter Sadowski, Jordan Ott, and Pierre Baldi. Sherpa: Robust hyperparameter optimization for machine learning. *SoftwareX*, 2020. Also arXiv:2005.04048. Software available at: https://github.com/sherpa-ai/sherpa.

[11] C. Andreopoulos, A. Bell, D. Bhattacharya, F. Cavanna, J. Dobson, S. Dytman, H. Gallagher, P. Guzowski, R. Hatcher, P. Kehayias, A. Meregaglia, D. Naples, G. Pearce, A. Rubbia, M. Whalley, and T. Yang. The genie neutrino monte carlo generator. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 614(1):87–104, 2010.

[12] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce, M. Asai, D. Axen, S. Banerjee, G. Barrand, F. Behner, L. Bellagamba, J. Boudreau, L. Broglia, A. Brunengo, H. Burkhardt, S. Chauvie, J. Chuma, R. Chytracek, G. Cooperman, G. Cosmo, P. Degtyarenko, A. Dell'Acqua, G. Depaola, D. Dietrich, R. Enami, A. Feliciello, C. Ferguson, H. Fesefeldt, G. Folger, F. Foppiano, A. Forti, S. Garelli, S. Giani, R. Giannitrapani, D. Gibin, J.J. Gómez Cadenas, I. González, G. Gracia Abril, G. Greeniaus, W. Greiner, V. Grichine, A. Grossheim, S. Guatelli, P. Gumplinger, R. Hamatsu, K. Hashimoto, H. Hasui, A. Heikkinen, A. Howard, V. Ivanchenko, A. Johnson, F.W. Jones, J. Kallenbach, N. Kanaya, M. Kawabata, Y. Kawabata, M. Kawaguti, S. Kelner, P. Kent, A. Kimura, T. Kodama, R. Kokoulin, M. Kossov, H. Kurashige, E. Lamanna, T. Lampén, V. Lara, V. Lefebure, F. Lei, M. Liendl, W. Lockman, F. Longo, S. Magni, M. Maire, E. Medernach, K. Minamimoto, P. Mora de Freitas, Y. Morita, K. Murakami, M. Nagamatu, R. Nartallo, P. Nieminen, T. Nishimura, K. Ohtsubo, M. Okamura, S. O'Neale, Y. Oohata, K. Paech, J. Perl, A. Pfeiffer, M.G. Pia, F. Ranjard, A. Rybin, S. Sadilov, E. Di Salvo, G. Santin, T. Sasaki, N. Savvas, Y. Sawada, S. Scherer, S. Sei, V. Sirotenko, D. Smith, N. Starkov, H. Stoecker, J. Sulkimo, M. Takahata, S. Tanaka, E. Tcherniaev, E. Safai Tehrani, M. Tropeano, P. Truscott, H. Uno, L. Urban, P. Urban, M. Verderi, A. Walkden, W. Wander, H. Weber, J.P. Wellisch, T. Wenaus, D.C. Williams, D. Wright, T. Yamada, H. Yoshida, and D. Zschiesche. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303, 2003.

[13] A Aurisano, C Backhouse, R Hatcher, N Mayer, J Musser, R Patterson, R Schroeter, and A Sousa. The nova simulation chain. *Journal of Physics: Conference Series*, 664(7):072002, dec 2015.

[14] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations*, 2014.

[15] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753, Long Beach, California, USA, Jun 2019. PMLR.

[16] Alexander Shmakov, Michael James Fenton, Ta-Wei Ho, Shih-Chieh Hsu, Daniel Whiteson, and Pierre Baldi. SPANet: Generalized permutationless set assignment for particle physics using symmetry preserving attention. *SciPost Phys.*, 12:178, 2022.

[17] A. Li, Z. Fu, L. Winslow, C. Grant, H. Song, H. Ozaki, I. Shimizu, and A. Takeuchi. KamNet: An Integrated Spatiotemporal Deep Neural Network for Rare Event Search in KamLAND-Zen. 3 2022.

[18] Pierre Baldi and Roman Vershynin. The quarks of attention, 2022.

[19] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[20] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. January 2015. 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[22] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[23] P. Baldi. *Deep Learning in Science*. Cambridge University Press, Cambridge, UK, 2021.

[24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[25] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[27] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

[28] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020.

[29] Junyuan Xie, Tong He, Zhi Zhang, Hang Zhang, and Jerry Zhang. Bag of tricks for image classification with convolutional neural networks. In *CVPR 2019*, 2019.

[30] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

A Related Work

A.1 Attention and Set Classification

Several problems in Physics may be reduced to assigning classification labels to a collection of unordered objects, or a set. In the case of NOvA event reconstruction, we are interested in classifying both the underlying event as well as reconstructing the label of each individual prong. Each event may contain multiple prongs, with no inherent order to them. Therefore, the prong observations and targets define variable length sets and we may generalize the prong reconstruction task to classification over these sets.

There have recently been several developments in using attention-based methods to handle variable-length sets [15, 16, 17]. Attention provides a gating mechanism for modifying neural network activations by incorporating contextual information [18], achieving state-of-the-art results in natural language processing problems such as translation [19, 20, 21, 22], where variable-length sequences are common. Among these methods, transformers [6] stand out as particularly promising for set assignment due to their fundamental permutation invariance [15]. Transformers are especially effective at modeling variable-length sets because they can learn combinatorial relationships between set elements in polynomial time [16].

A.2 Sparse Convolutions

The scintillator pixel-maps produced by the NOvA detector present several unique challenges for machine learning. These pixel-maps are typically very sparse, with with events having, on average, 0.84% of pixels containing non-zero hit values, leaving most of the observations void of data. Convolution neural networks (CNNs) are ubiquitous element of deep learning methods for efficiently learning on image and other spatially-related collections of data [23]. This effectiveness stems from the extreme weight sharing that comes with using a small *kernel* matrix which is applied everywhere across space.

However, CNNs falter when dealing with sparse images since small kernels may contain very little data, and extremely large kernels would eliminate the benefits of weight sharing. To combat this, the spatial kernel concept has been expanded apply to sparsely distributed data, proving especially successful in 3D objective reconstruction [8, 5]. These apply the convolution operations only in regions where data exists, saving on computation and preventing the dilution of sparse values even in a predominantly zero valued image.

A.3 Interpretable Deep Learning

The black-box nature of deep neural network models stems from our inability to analytically describe the training and inference processes in all but the most simple neural networks [24]. There has recently been a surge in methods for analyzing specific aspects of neural network architectures to extract human-understandable measurements from their internal structures. Saliency maps [14] provide a method for analyzing the behaviour of CNNs by studying the model's output gradients with respect to the inputs and provide a very visual understanding of the network's behaviour near individual inputs. Similarly, transformers may be analyzed by instead focusing on the attention matrices computed during self-attention [6, 25, 21]. These *attention maps* measure the importance of different inputs, for example individual words in a language model, for determining the output of a transformer.

B Hyperparameters

We present a full list of hyperaprameters used to define the network. The CNN follows the DenseNet architecture [7] with a modified number of blocks and embedding dimensions. The prong transformer follows the canonical transformer encoder [6] architecture. We used a focal classification loss [9] with a chosen focal γ parameter. AdamW [26] and cosine annealing with warm restarts [27] with canonical parameters are used for training the network. The AdamW optimizer along with warm-restarts has shown success on both vision [28, 29] and NLP [30] tasks with transformers. We replicate this setup in our experiments. Training was performed on 4 NVidia 3090 GPUs, splitting a batch size 2048 events between the GPUs.

We optimize the hyperparameters of our neural network using the Sherpa hyperparameter optimization framework [10]. We use Bayesian Optimization with a Gaussian Process surrogate to guide the hyperparameter search process over $10,000$ short training trials.

Parameter	**Value**
CNN Embedding Dimensions	512
CNN DenseNet Blocks	5
Transformer Dimensions	256
Transformer Encoders	6
Type Embedding Dimensions	32
Focal Loss γ	1.0
AdamW Learning Rate	1×10^{-5}
AdamW Weight Decay	2×10^{-5}
Cosine Annealing Epochs	1024
Cosine Annealing Cycles	16

Table 3: Caption

C Total Attention Extraction

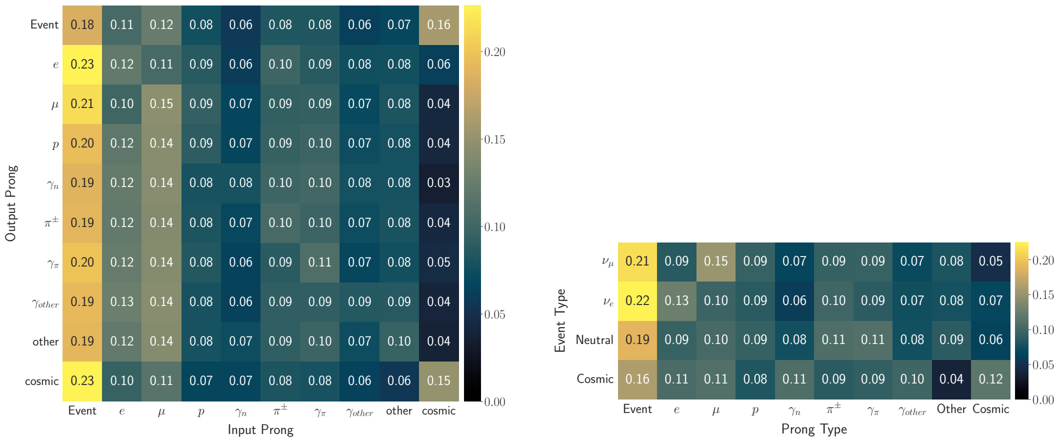
We present a method for extracting the overall attention for a single event by aggregating the attention weights of individual transformer layer.

Let $E \in \mathbb{R}^{(N+1)\times D}$ be a set of embedded pixel-maps where N is the number of prongs in the event and D is the latent dimensionality. Every transformer layer, T_i, in a K-layer transformer encoder produces a pair-wise importance score for all of the input pixel-maps: $A_i \in \mathbb{R}^{(N+1)\times(N+1)}$. We may compute a total attention score for a single event by taking the product of these importance matrices across the entire transformer encoder.

$$A = A_K A_{K-1} \dots A_2 A_1 \mathbb{I}$$

This produces a single $(N+1) \times (N+1)$ matrix who's rows all sum to one. We note that high attention values does not indicate correlation with predicting the given output, but rather the *importance* for separating different classifications.

D Attention Maps



(a) Average pairwise **Per-Prong** attention scores.

(b) Average event **Per-Prong** attention scores.

Figure 6: An Alternative aggregation method for attention matrices which just look at how each individual prong contributes to the attention score of prongs and events. This includes a downside in that prongs which appear more than once in events, such as photons, will be underrepresented since their attention will be split among other prongs with the same type. However, this does provide a more low-level view of the importance of different prongs in different classification and reconstruction tasks.

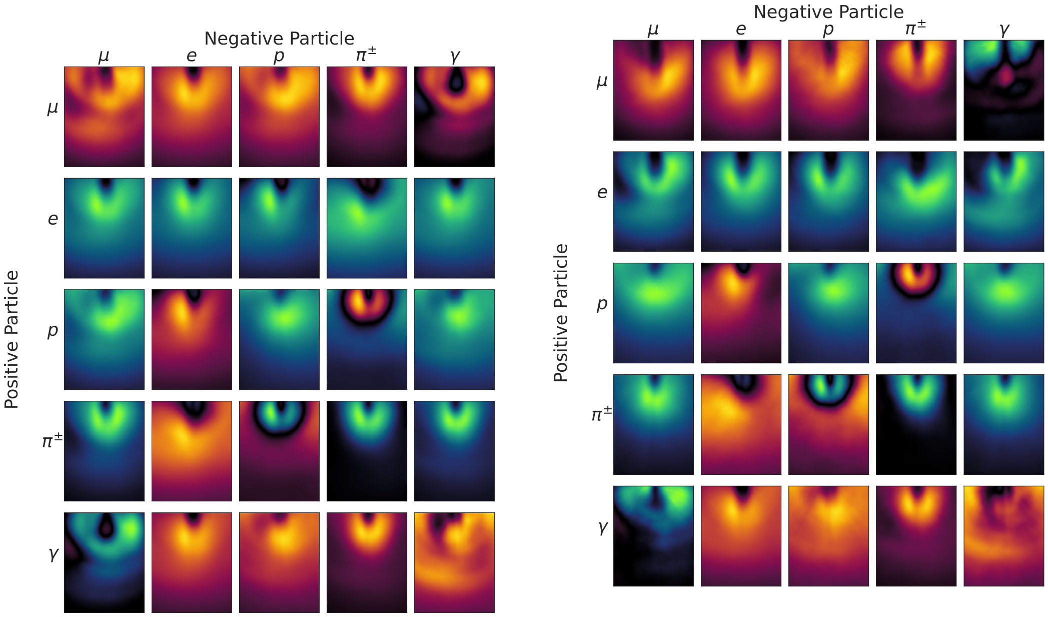
E Saliency Aggregation

We need to aggregate the individual event saliency to extract global pixel-maps with patterns identifying common areas of interest for the network. We perform this aggregation in several steps.

1. Compute the saliency of every output head for several blurred, noisy variations of each prong. We add a small uniform random noise to pixel-maps to robustly estimate the gradient near a given input, followed by a Gaussian blur with a standard deviation of 1 pixel to smooth out the discrete nature of our pixel-maps. We average the saliency across these noisy blurred inputs to provide a smooth gradient estimates. This produces 9 saliency maps for every prong, one for each prong label.
2. Translate and rotate each saliency map to align each prong's vertex (the initial location of the hit) to the top center of each map and the prong's track (the decay tail) along the vertical axis. We use vertex and direction information from the simulator's particle reconstruction, included in the NOvA MC data release.
3. Enforce a similar distribution for every prong type by limiting events to only those where the track length is less than 50 pixels (488 cm) and the reconstructed particle energy is less than 4 GeV. We do this to focus on the differentiation of similar-looking prongs, providing hints to subtle differences between different prong types instead of obvious overarching differences.
4. Average the resulting smoothed and rotated saliency maps for every type of prong to compute the gradient with respect to deflection way from the vertex for each prong type.

F Saliency Pixel Maps

In Figure 7, we present both types of saliency Pixel-maps. We first present the saliency aggregated across all events for every output of the network. We then present the saliency for only those events where the true prong type matched the positive (row) output.

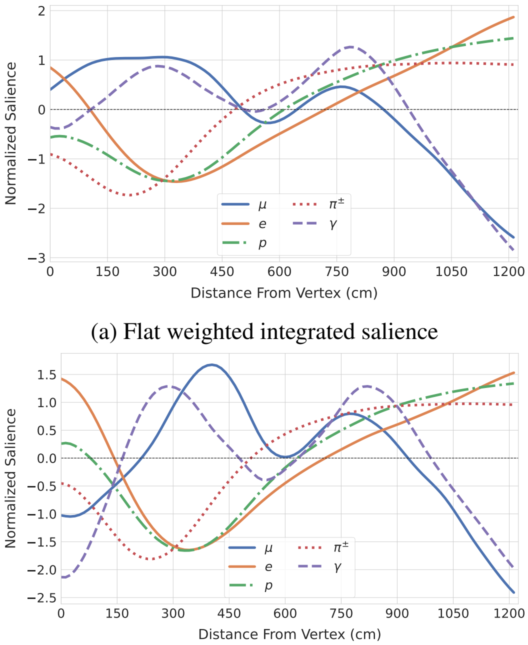


(a) All prongs examined in every saliency map.

(b) Each row only contains prongs who's truth label matches the positive particle for the given row.

Figure 7: Grid of aggregated saliency maps and difference maps for every pair of prong types. Red indicates a positive correlation with the Positive Particle's reconstruction probability, while blue represents a negative correlation.

G Integrated Saliance



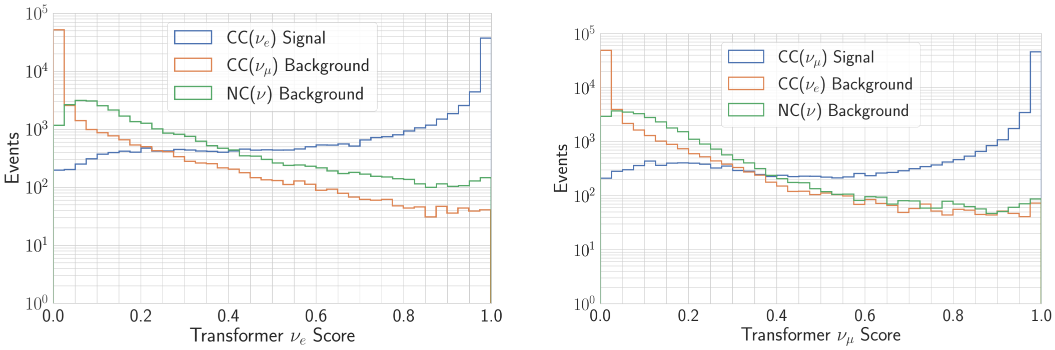
(a) Flat weighted integrated salience

(b) Gaussian weighted integrated salience.

Figure 8: Full length views of the integrated salience maps along particle tracks which include the outer regions of the track. We additionally include a Gaussian-Weighted variant which puts more importance to salience near the particle's track. We re-weighting the salience maps with a Gaussian weight with respect to distance from the track's center before integrating the maps along their width.

H Signal-Background Rejection

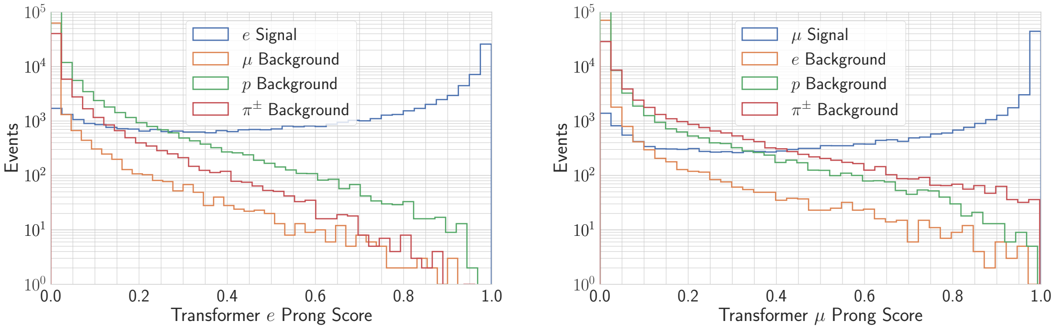
We can also use the softmax distribution outputs from both the event and prong network outputs as a method for cutting non-signal events from the NOvA data. To examine the effectiveness of this cut, we plot a histogram of the network's softmax probability of assigning a given classification for every event or prong, grouped by their ground truth values. These plots are presented in Figures 9 and 10. We notice that all of the major events and prongs achieve an order of magnitude signal-background cut after a classification probability of 0.8 while still keeping a majority of the signal data with a cut up to 0.9.



(a) Transformer ν_e Event Softmax Scores

(b) Transformer ν_μ Event Softmax Scores

Figure 9: Event signal-background rejection curves for different Event Types. Calculated as the TransformerCVN's likelihood of classifying a particular signal event as one of the background classes.

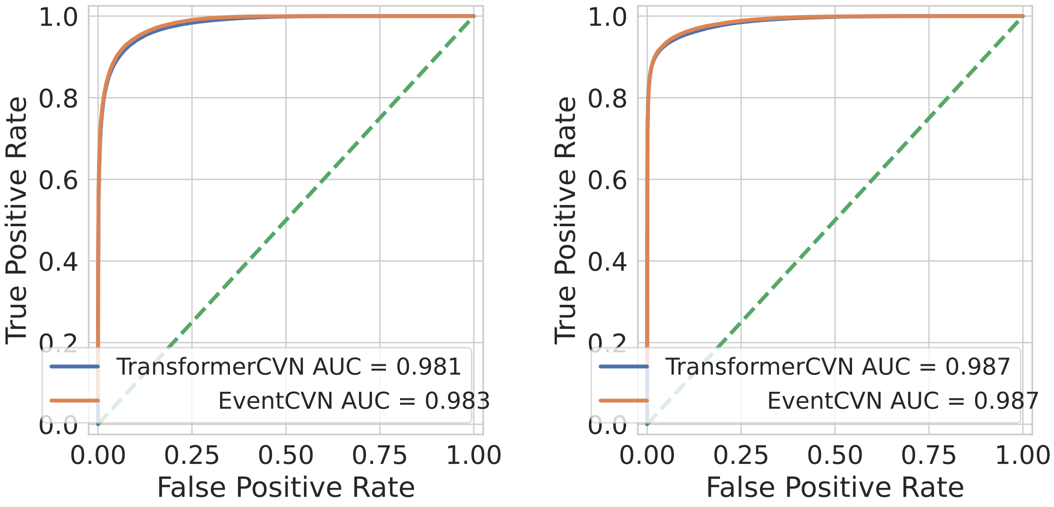


(a) TransformerCVN e Prong Softmax Scores

(b) TransformerCVN μ Prong Softmax Scores

Figure 10: Prong signal-background rejection curves for lepton prongs. Equivalent computation as the event signal-background curves, but performed with the four most common types of prongs.

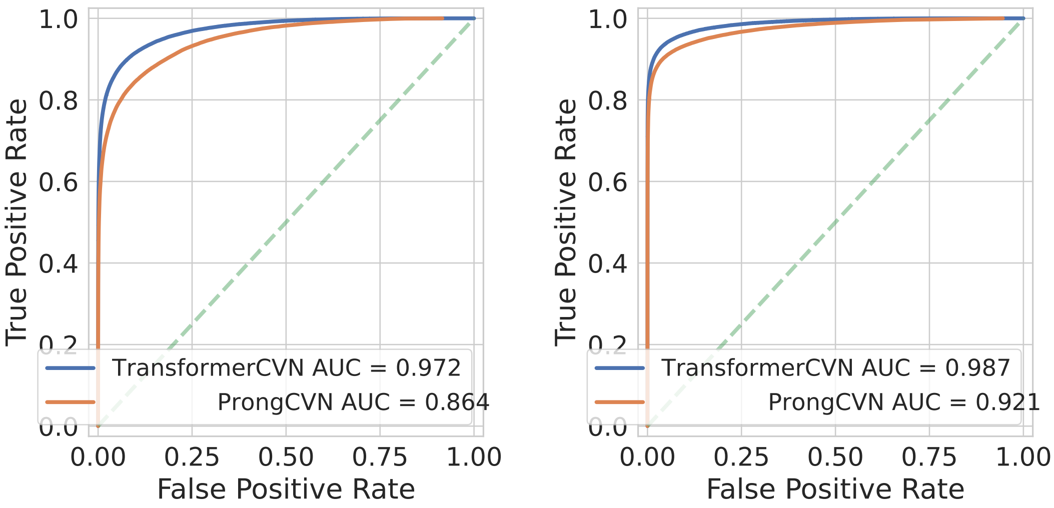
I ROC Curves



(a) ROC Curve for ν_e event reconstruction.

(b) ROC Curve for ν_μ event reconstruction.

Figure 11: Event classification ROC curves for TransformerCVN and EventCVN Baseline.

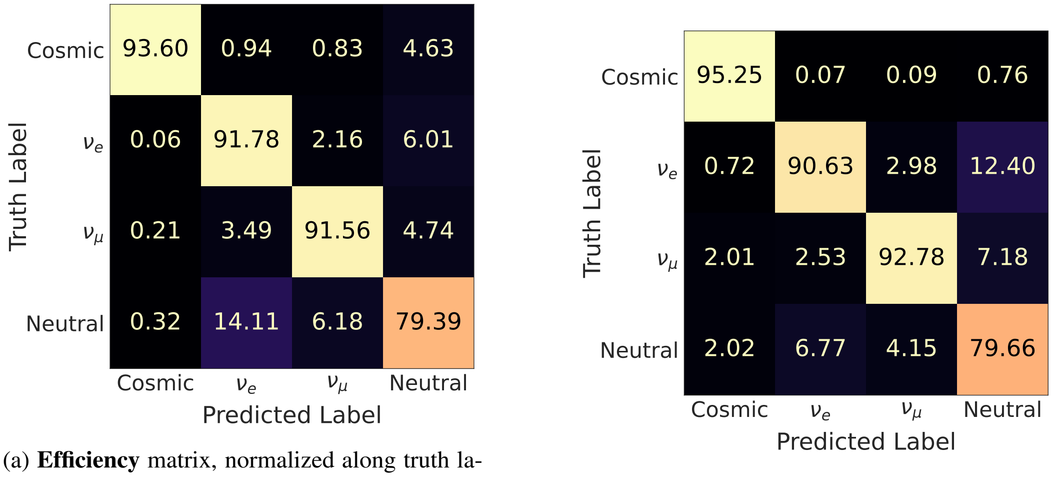


(a) ROC Curve for e prong reconstruction.

(b) ROC Curve for μ prong reconstruction.

Figure 12: Lepton Prong Reconstruction ROC curves for TransformerCVN and ProngCVN Baseline.

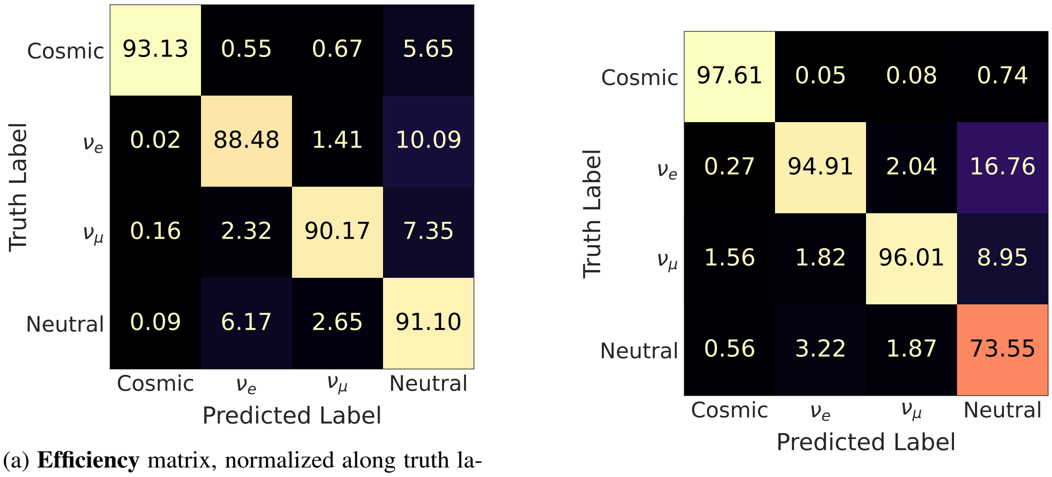
J Confusion Matrices



(a) **Efficiency** matrix, normalized along truth labels.

(b) **Purity** matrix, normalized along predictions.

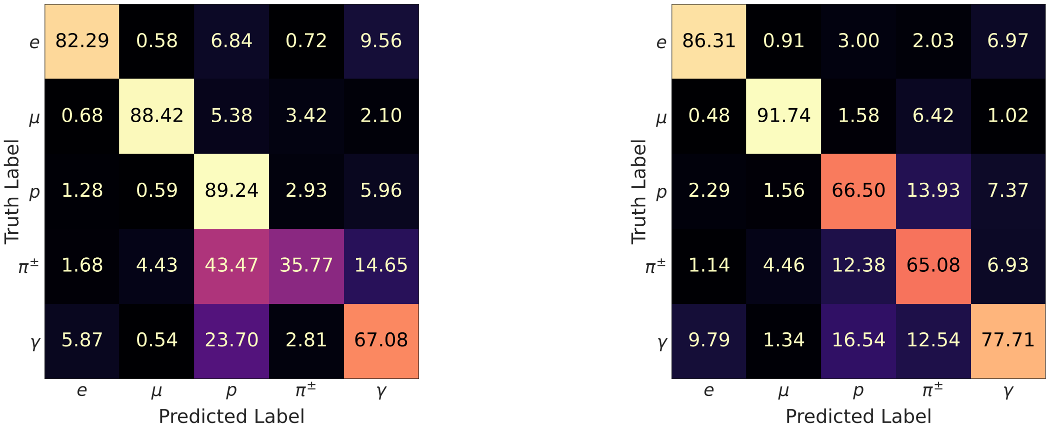
Figure 13: TransformerCVN 4 Class Event Confusion Matrices.



(a) **Efficiency** matrix, normalized along truth labels.

(b) **Purity** matrix, normalized along predictions.

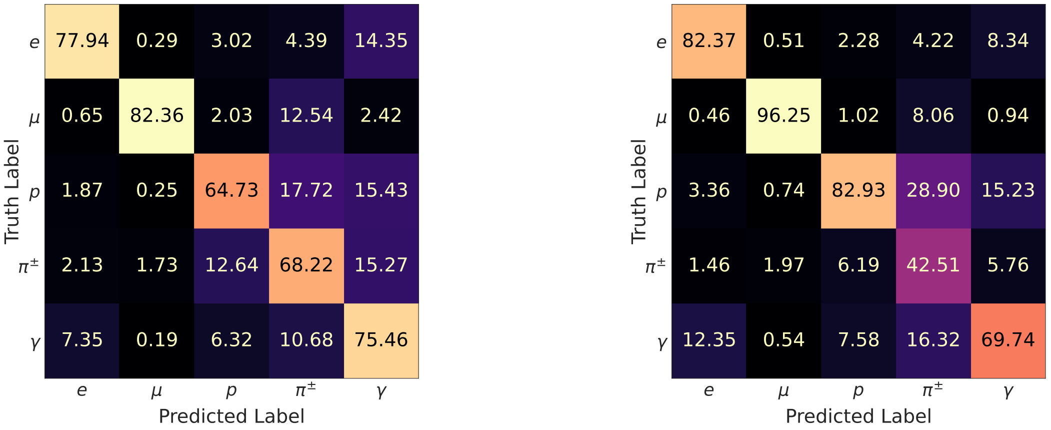
Figure 14: EventCVN Baseline 4 Class Event Confusion Matrices.



(a) **Efficiency** matrix, normalized along truth labels.

(b) **Purity** matrix, normalized along predictions.

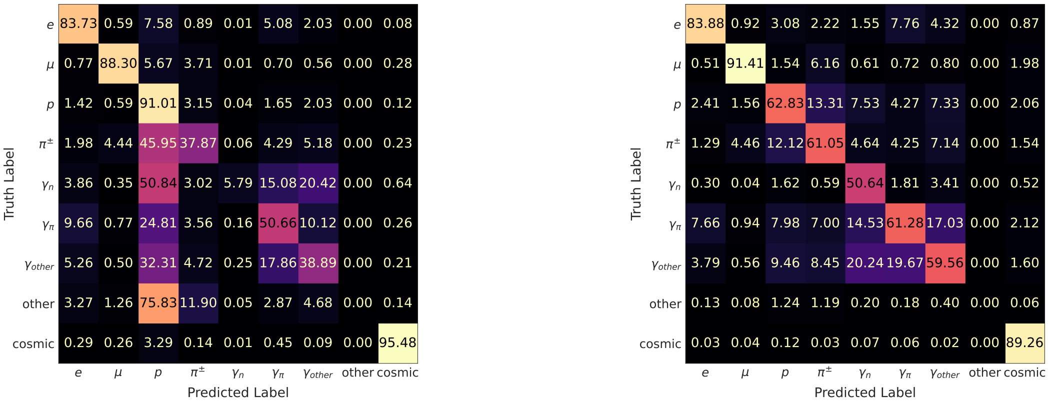
Figure 15: TransformerCVN 5-class Prong Confusion Matrices.



(a) **Efficiency** matrix, normalized along truth labels.

(b) **Purity** matrix, normalized along predictions.

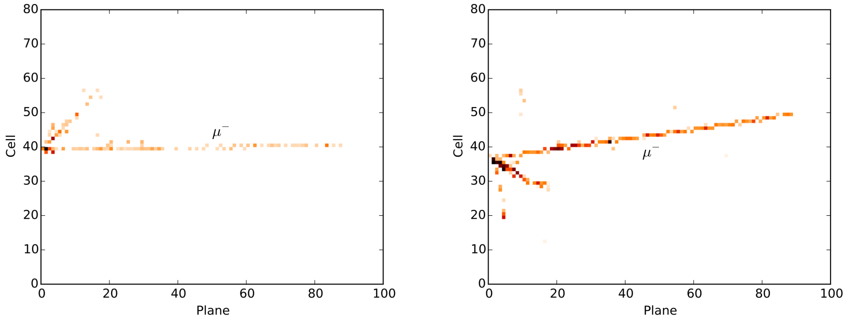
Figure 16: ProngCVN Baseline 5-class Prong Confusion Matrices.



(a) **Efficiency** matrix, normalized along truth labels.

(b) **Purity** matrix, normalized along predictions.

Figure 17: TransformerCVN Full 9-class Prong Confusion Matrices.



X-view

Y-view

Figure 18: An Example pixel-map for a ν_μ CC event. Potential neutrino interaction events are split into clusters of energy deposits called "slices." These slices are cropped such that the first cell hit along the beam direction is placed in the first column along the z-axis and the hits are centered along the x or y direction. The pixel map is then generated by filling each pixel with a value between 0 and 255 proportional to the cell's energy deposit with saturation at 278 MeV.