Neutrino Classification Through Deep Learning

María Fernanda Romo-Fuentes[1⋆], Luis Eduardo Falcón-Morales[1]

1 Department of Science and Engineering, Tecnológico de Monterrey

⋆ mferomof@gmail.com , † luis.eduardo.falcon@itesm.mx

22nd International Symposium on Very High Energy Cosmic Ray Interactions (ISVHECRI 2024) Puerto Vallarta, Mexico, 8-12 July 2024
doi:10.21468/SciPostPhysProc.?

Abstract

Neutrinos are a type of subatomic particle whose study is expected to allow us to gain a better understanding of cosmic phenomena and the universe itself. The study of these particles begins with the collection of data provided by their passage through a Water Cherenkov detector followed by determining properties such as its energy, track direction and class. In this project we implemented four deep learning methods: VGG19, ResNet50, PointNet and Vision Transformer, to classify neutrino events as one of three classes: gamma, electron and muon, with the objective of determining which algorithm works better, in comparison among themselves and with state of the art methods, which include custom Convolutional Neural Networks or deep learning algorithms. Our results show that among the implemented methods, ResNet 50 yielded the best results, with an accuracy of 72.48% and an Area Under the Curve for the efficiency plot of 0.71. These results were obtained by employing the largest dataset available which showed the importance of having a big enough representation of all types of events in the analysis.

Received Date
Accepted Date
Published Date

1 Introduction

Studying neutrinos is expected to provide significant insights into the universe's history and functioning since these highly elusive particles have no charge and thus interacting minimally with matter along their path of travel directly from their origin up to a detector, or more importantly, they present a behavior called oscillation, in which they can be measured to have a different flavor from the one they actually have [1,2].

Because neutrinos are difficult to observe, we need special detectors in which we can manage to catch information about their passing. These detectors, called Water-Cherenkov detectors, work on the principle of catching the Cherenkov radiation produced when neutrinos collide with the charged particles in water, which makes them travel at a speed higher than the speed of light in the medium. [1,3]. The light produced by the Cherenkov radiation, which usually comes in the form of a ring or cone, is gathered by the sensors located along the walls of the detectors for studying. These sensors are called Photo-Multiplier Tubes (PMTs) which, to increase the amount of information collected for each event, are used as a single circular unit composed by 19 of them and thus receive the name of multi-PMTs (mPTs) [3].

The analysis of the data begins by determining features of the detected event such as the track direction of the neutrino, its energy before colliding with the particles in the water, and the its class, which is the main topic of the present research, where four deep learning methods: VGG19 [4], ResNet50 [5], PointNet [6] and Vision Transformer [7], were implemented to identify which works better for the task of identifying the class of simulated neutrino events corresponding to a Water-Cherenkov detector called Intermediate Water Cherenkov Detector (IWCD). State of the art research shows that for a custom Convolutional Neural Network (CNN) we get an accuracy of approximately 70% [8] while for a ResNet50 model with different hyper-parameters to the ones shown in this project and by applying cuts to the data we get an Area Under the Curve (AUC) for the efficiency plot of 0.77 at best [9].

In this project we found that the model which provided better results was ResNet50 with an accuracy of 72.48% and an AUC for the efficiency plot of 0.71, while also minimizing the needed for computational resources. Moreover we observed that, independently to the models, the bigger the dataset the better the results as then, we have enough samples of different types of events within each class to assure the employed architectures can learn them. Finally, for hyper-parameter tuning, we had to employ smaller samples of the largest dataset to make the process resource efficient given its size, and from this we determined that the samples should be taken randomly and be unordered, assuring the models learned better.

In this paper we have 5 sections, in section 2 we talk about the employed methodology and the data we used, to then, show and explain our results in section 3. After this, in sections 4 and 5 we provide the discussion and conclusions of the developed project, respectively.

2 Methods

The followed methodology and a description of the employed dataset is found in this section.

2.1 Data

The employed data was simulated and corresponds to the IWCD tank, a Water-Cherenkov detector 8m tall, with a diameter of 10m and 536 mPMTs along its walls, at the moment the events were simulated. All events are of a single ring type, which means we have only one class per event, which can be one of three: gamma, electron or muon. This data is stored within two supercomputing clusters: CADS located at Universidad de Guadalajara in Mexico, and Cedar located at Simon Fraser University in Canada. The number of events per class for each of the dataset located in these two clusters can be seen in Table 1. The number of events per particle at CADS are given as a range because we have multiple datasets with a size ranging between the given numbers while in Cedar we have only one dataset.

2.2 Methodology

The overall methodology followed to obtain the models used to classify the neutrino events can be observed in the diagram in Figure 1. As can be seen, the process of training, validating and testing to get the final models was done recursively as we had to tune their hyper-parameters, by using the values obtained for the metrics listed in the last block of the diagram. This tuning was done employing either the smaller datasets, which had 9k events per class or by taking a smaller sample of the larger datasets, to reduce training time and usage of computational resources.

Table 1: Number of events per class per supercomputing cluster

Database	Class	Number of events per particle
	gamma	9k to 3M
CADS	electron	9k to 3M
	muon	9k to 3M
	gamma	∼8M
Cedar	electron	∼8M
	muon	∼3M

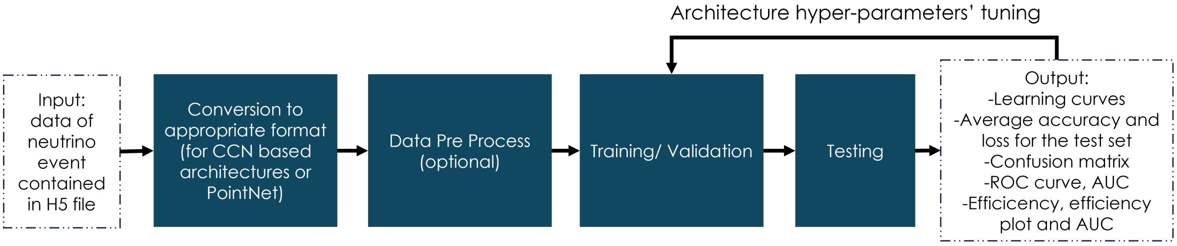

Figure 1: Block diagram of the followed methodology

As a result of the process depicted in the diagram, we obtained 9 different models by employing one of the 4 implemented deep learning architectures: VGG19, ResNet50, PointNet and Vision Transformer, considering different training parameters or modifications made to the dataset so as to try and improve the values of the classification metrics. These modifications were:

- Separating the dataset into muon and not muon class and then separating the not muon class into gamma and electron. This was done because all the models have no problem identifying the muon class because of its higher energy, but easily misclassify the gamma and electron classes, whose energies tend to overlap. Therefore, the muon class was removed from the beginning so that the models can focus in learning how to differentiate between gamma and electron.

- Considering only gamma and electron we filtered out those events in which a minimum number of pixels were not different from zero after processing the input data, since there are events that had sparse hits and did not form a ring. It was only applied to the smaller datasets and considering the objective of improving the evaluation metrics of the models but, even though it did improve the results it also made the models biased to the events that form a ring making it not useful for classifying real events, where the number of hits varies considerably. Nonetheless, we found that by employing the largest datasets we can mitigate the effect that events with sparse hits have on the learning of the models.

For parameter tuning and obtainment of all the models the data was divided randomly into training, validation and testing, by taking 70, 15 and 15% of the data for each set, respectively.

3 Results

To evaluate the models obtained from applying the process shown in Figure 1 we got different evaluation metrics but, considering only the general objective of the project which was about finding which architecture worked better for the classification of neutrino events, we only show two metrics for the gamma and electron classes, as the muon class is easily separable: the efficiency plots with their respective AUC considering the electron class as signal and the gamma class as background, and the ROC curves with their AUC in a one vs. the rest approach.

Then, in Figure 2 we can see the efficiency plot and AUC obtained from considering the electron class as signal and the gamma class as background for each of the following 9 models:

- A VGG19, a ResNet50 and a PointNet model, each trained with 9k events per class and considering the three classes.
- A Vision Transformer and a ResNet50 model, each trained with 300k events per class and considering only the electron and gamma classes.
- A Vision Transformer model trained with the events left after filtering 300k events per class by pixels and considering only the electron and gamma classes.
- A Vision Transformer model using pretrained weights and applying fine tuning to the last layers using the data from the previous point.
- A ResNet50 model using pretrained weights and applying fine tuning to the last layers using 300k events per class consdering only the electron and gamma classes.
- A ResNet50 model trained on the Cedar dataset, considering the three classes.

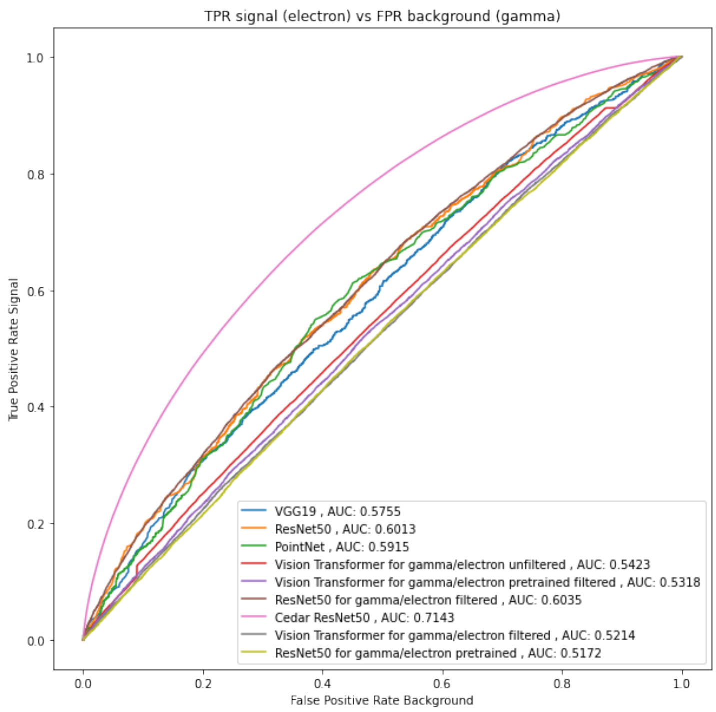

Figure 2: Efficiency plot for all models employed in the classification of the gamma (background) and electron (signal) classes

As we can see, the best results were obtained by employing the ResNet50 model to classify the Cedar dataset, since we had a curve with an AUC of 0.71. These results were confirmed by the ROC curves and AUC values shown in Figure 3, where the curve that reaches a value of 1 for the true positive rate with a smaller false positive rate for both classes, which also meant a greater AUC value, was also this ResNet50 model. Overall we can see that ResNet50 did well with all the modifications applied to the data, except for when using pretrained weights.

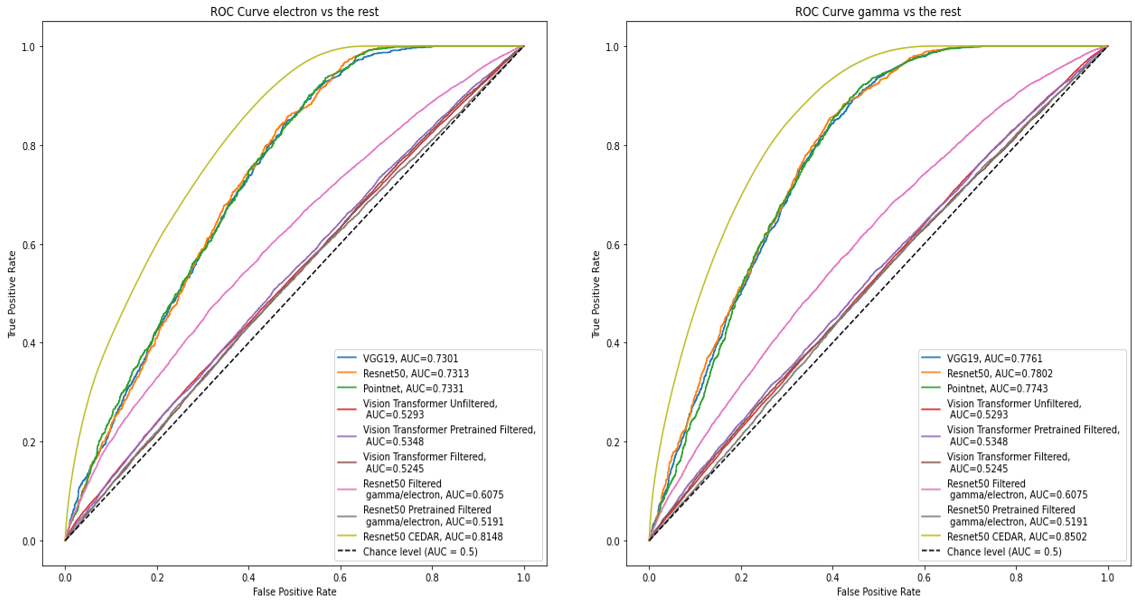

Figure 3: ROC curves and AUC for the gamma and electron classes in a one vs. the rest approach

4 Discussion

Our research has shown that out of all the models we applied to the task of classifying neutrino events, ResNet50 provided the best results, something that was consistent throughout all the applications and modifications that were made to the dataset, except when using pretrained weights, which showed that pretrained weights are useful depending on the application. Moreover, the results considerably improved when the largest dataset was employed which also solved the situation of having biased data after applying the filtering by number of pixels, showing that events with sparse hits do not pose a problem for the classifiers as long as we have a large enough sample of these types of events.

We also found that when sampling from the largest datasets, it is essential to take random samples and we should not sort by the values of any of the variables which describe the event, thus allowing the representation of all types of events in the sample, this is, those that form rings and those that have sparse hits.

As for the filtering of events with sparse hits, we ought to mention that the classification metrics did improve since we only analyzed events that did form a ring. Nonetheless, we cannot assure that we will only have these types of events since features like energy, track direction or location within the tank where the collision of the neutrino with the charged particles in the water occurred could provoke detection of sparse hits. Therefore, further research should be done so that we can conclude what are the conditions of an event with detection of sparse hits so that we can assure if we should filter these events or not.

5 Conclusion

While the main objective of our research was to determine which of the proposed deep learning architectures worked better for the classification of neutrino events, we could also observe how important getting to know our data was to get conclusive results, since we could grasp that the gamma and electron classes were harder to separate and, though we could improve our results by filtering events with sparse hits, this was not ideal but we could solve it by increasing the number of events. As for the models, ResNet50 provided the best and most consistent results except when using pretrained weights, obtaining an accuracy of 72.48% and an AUC for the efficiency plot of 0.71 for our best model, results comparable with state of the art methods. Further research should be done to the conditions surrounding the detection of events that form a ring and those that do not so that this could be taken into account during analysis.

Acknowledgements

We thank Tecnológico de Monterrey and CONAHCyT for all the support given for the development of this project.

Funding information The present project was supported by the Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCyT) through the grants given to the projects CBF2023-2024-427 and CF-2023-G-643.

References

[1] S. Pastor Carpi, *Los neutrinos*, Editorial CSIC Consejo Superior de Investigaciones Científicas, ISBN 978-84-00-09864-3, OCLC: 923058992 (2000).

[2] S. S. Berkman, *$\nu\mu$ CC1π+ Events Produced in the T2K Beam at Super-Kamiokande* (2018), doi:10.14288/1.0365713, Publisher: University of British Columbia Version Number: 1.

[3] HK Collaboration, *Hyper-Kamiokande: Peering into the Universe and its elementary particles from underground*, Tech. rep., Hyper-Kamiokande (2015).

[4] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition* (2015), 1409.1556.

[5] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition* (2015), 1512.03385.

[6] C. R. Qi, H. Su, K. Mo and L. J. Guibas, *Pointnet: Deep learning on point sets for 3d classification and segmentation* (2017), 1612.00593.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, *An image is worth 16x16 words: Transformers for image recognition at scale* (2021), 2010.11929.

[8] S. Cuen-Rochin, E. de la Fuente, L. Falcon-Morales, R. Gamboa Goni, A. K. Tomatani-Sanchez, F. Orozco-Luna, H. Torres, J. Lozoya, J. A. Baeza, J. L. Flores, B. Navarro-Garcia, B. Veliz *et al.*, *Machine Learning Application for Particle Physics: Mexico's Involvement in the Hyper-Kamiokande Observatory*, In D. Oliva, E. H. Houssein and S. Hinojosa, eds., *Metaheuristics in Machine Learning: Theory and Applications*, vol. 967, pp. 583–608. Springer International Publishing, Cham, ISBN 978-3-030-70541-1 978-3-030-70542-8, doi:10.1007/978-3-030-70542-8_23, Series Title: Studies in Computational Intelligence (2021).

[9] N. Prouse, *Machine Learning for Reconstruction (& detector simulation & calibration)* (2023).