

Deep Learning for Signal and Background Discrimination in Liquid based Neutrino Experiment

Rui Li^{a,1}, Zhengyun You^{a,2}, Yumei Zhang^{b,3}

^aSchool of Physics, Sun Yat-sen University, Guangzhou 510275, China

^bSino-French Institute of Nuclear Engineering and Technology, Sun Yat-sen University, Guangzhou 510275, China

Email: ¹lirui47@mail2.sysu.edu.cn, ²youzhy5@mail.sysu.edu.cn, ³zhangym26@mail.sysu.edu.cn

Abstract. In high energy physics experiments, efficient data analysis tools are required to extract interesting information from the massive data. For large-scale liquid-based neutrino experiments, neutrino signals are usually overwhelmed in huge backgrounds. By constructing a liquid neutrino detector toy model, we generate simulation data in Geant4 [1] and run reconstruction for signal background discrimination. The low-level Photo Multipliers (PMT) hits are also projected to a 2D plane to create visualization outputs for classification. With the 2D images as input, we use the Convolutional Neural Network (CNN) as a specific application, which has shown remarkable performance in signal and background discrimination and outperforms those with high-level reconstruction outputs. The method is expected to be used in the neutrino experiments such as JUNO with further study.

1. Introduction

In high-energy physics (HEP) experiments, a huge number of events are recorded in the detectors in every second. The signals are usually overwhelmed by huge backgrounds. To search for the interesting physics in the huge data, it is essential to discriminate the signal events from backgrounds with some characteristic features being extracted. Neutrino experiments are good examples in HEP experiments where signal and background discrimination play an important role. Due to the extremely low cross section of neutrinos' interactions with matters, neutrino experiments are known for their low signal rates and high backgrounds, which include the cosmic ray muons, radioactive backgrounds from detector materials and environments, as well as accidental backgrounds, such as two independent signal events coinciding within the same event. To increase the statistics of signal events, neutrino experiments are usually built with large-scale liquid detectors, such as the Super-Kamiokande [2], DayaBay [3] and JUNO [4] experiments.

The traditional algorithms in HEP experiments often require extracting features from raw event data so that we can get better performance in event classification and physics analysis. For example, some critical physical quantities, such as the transverse momentum, secondary vertex and energy of the tracks in an event will be reconstructed. However, even if physicists spend a great deal of time on manually constructing the features, the result is still not promising because these algorithms are not good at solving

non-linear problems. More or less, some information in the raw event data will be lost after data reconstruction.

Deep learning is a newly flourishing machine learning algorithm which shows superior performance on solving non-linear problems [5]. Much efforts have been taken in various kinds of HEP experiments and software developments [6, 7]. In all these applications, CNN is a powerful tool in image processing and recognition, which makes it very suitable for event classification. Some implementations in HEP experiments have been realized in the recent years [8, 9].

In this paper, we focus on signal and background discrimination in the large-scale liquid-based neutrino experiments and application of CNN on the image-based event output for classification. In section 2, we introduce Neural Network and CNN in deep learning. In section 3, the implementation of a toy liquid neutrino detector model is introduced, as well as the production of simulation data and classification with reconstruction outputs. In section 4, we discuss the 2D imaging of the PMT hits distribution in event outputs and application of CNN with the image input for signals and radioactive backgrounds discrimination. The classification results are compared with the results from the traditional Multilayer Perceptron (MLP) method using reconstruction data. Finally, in section 5, we make a summary and discuss the potential application of the method in future neutrino experiment.

2. Deep Learning

2.1. Neural Network

Neural network is a machine learning algorithm intellectually inspired by human brain. MLP [10, 11] is one of the traditional neural networks. In MLP, the basic element is a perceptron, which does exactly what a biological neuron do - turn to a specific state due to the input. For each input variable, the perceptron has a corresponding weight. Each perceptron has its own activation function and bias. Say the bias of the perceptron is b and the activation is the sigmoid function, a typical output of the perceptron with three inputs is $\left[e^{(w_1x_1+w_2x_2+w_3x_3+b)} \right]^{-1}$.

Neural network is the combination of connected perceptrons. As shown in the left plot of Figure 1, perceptrons are fully connected layer by layer. The left layer in purple represents the input layer, the number of the perceptron is decided by the number of inputs. Layers in blue represent the hidden layers. The total amount of hidden layers and perceptron can be adjusted. While training the neural network, different combinations need to be tried to find out which one outperforms the others. The green layer represents the output layer, which gives prediction of the neural network.

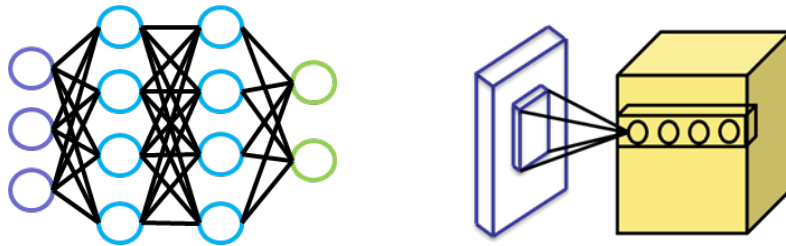


Figure 1. Basic structure of neural network (left) and the structure of a convolutional layer (right).

2.2. Convolution Neural Network

Convolution neural network [12] is inspired by the biological process in visual cortex. Instead of processing the information pixel by pixel, visual cortex does it in a part by part manner. It is widely used in the field of computer vision and has shown extraordinary performance.

In a convolutional layer, the neurons are no longer fully connected to its previous layer. There are a set of learnable filters in the convolution layer. During the forward process, the filter slides across the input and computes the dot product between the weights of filter and the input. When we slide the filter

over the whole picture, we get something like a feature map of the original input. The basic structure of a convolutional layer is shown in the right plot of Figure 1.

3. Implementation in a Neutrino Detector Model

3.1. Detector Modelling

Neutrino experiments usually use a large amount of materials as the detector. In measurement of the neutrinos/antineutrinos from accelerators or reactors in nuclear power plants, inverse beta decay (IBD) $\bar{\nu}_e + p \rightarrow e^+ + n$ is one the most commonly used reaction for neutrino detection. The detecting materials are usually liquids, such as pure water (Super-Kamiokande, DayaBay), or liquid scintillator (KamLAND, JUNO). To detect the single photons from Cherenkov light or scintillation light, photon-multiplier (PMT) are widely used in neutrino experiments.

A toy neutrino detector model has been designed to simulate the large-scale liquid-based neutrino experiments. In this toy model, the detector is a spherical container with the diameter of 30 meters and is filled with liquid scintillator (linear alkylbenzene) for neutrino detection. The surface of the sphere is covered by 20-inch PMTs to measure the scintillation light from the liquid in the container. The number of the PMTs is about 10,000. A sketch of the detector model is shown in Figure 2. (Please note that the size of the PMTs and the size of container are not of the same scale but just for illustration.)

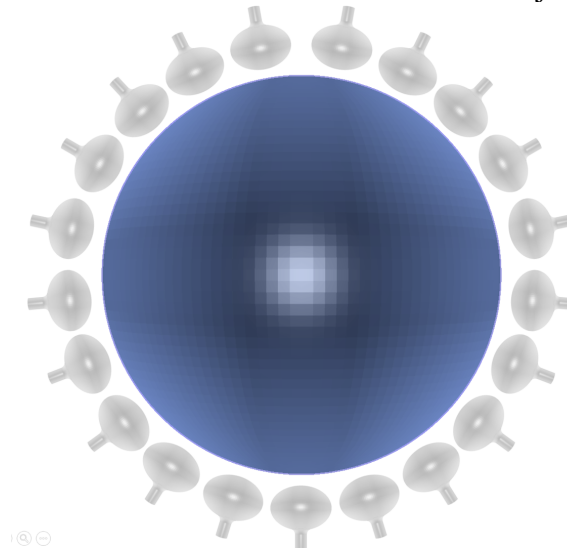


Figure 2. Sketch of the toy detector model. The blue sphere is the liquid container with grey PMTs arranged on its surface for photon detection.

3.2. Signals and Backgrounds

In this study, we focus on the discrimination of IBD signals and radioactive backgrounds. The signals are from neutrinos' IBD interaction with two final particles, a positron and a neutron. They both deposit energy in the liquid scintillator and produce optical photons, which will be received by the PMTs to reconstruct the vertex and energy of the event.

Due to the low signal counting rates in most neutrino experiments, background control is one of the most important and difficult works. The background events may come from different kinds of sources, including cosmic ray muons, radioactive background and accidental backgrounds. Neutrino experiments are usually build in deep underground with heavy overburdens above. Usually only high energy muons can penetrate the rocks on top of the detector, so the characteristics of cosmic ray muons are quite different from signals. In this study, we focus on discrimination of signals and radioactive backgrounds. In neutrino experiments, the level of radioactive backgrounds is greatly dependent on the purity of the environments and detector materials. The most common radioactive backgrounds come from the radioactive elements such as ^{238}U , ^{232}Th , ^{40}K and ^{222}Rn , which widely exist in the environment rocks,

PMT glasses, detector liquid due to impurity and the containers. According to different environment of a specific neutrino experiment and materials being used in a specific detector, radioactive backgrounds can be greatly different.

3.3. Simulation and Reconstruction

The detector model has been implemented in a Geant4 based detector simulation program. The signal IBD events are generated with event vertex randomly distributed in the container of liquid scintillator. Its final particles, the positron and neutron deposit energy after flying a short distance and produce scintillation lights. The photons transport in the liquid container and then produce hits in the PMT on the surface of the container. The background events are produced not only in the liquid scintillator, but also come from the container itself and the PMTs surrounding it.

The hits from all fired PMTs in an event will be written out for reconstruction. In neutrino experiments, event vertex and energy of the neutrino are two important variables for further physics analysis. In vertex reconstruction, we use the Charge Centered algorithm, in which the weighted position center of all fired PMTs after correction is set to be the vertex of the neutrino interaction point. For energy reconstruction, a Maximum Likelihood algorithm is used to fit the PMT hits distribution and get the most probable value of the reconstructed energy.

3.4. Classification with Reconstruction outputs

The reconstructed event vertex and event energy, together with other characteristic variables such as the total number of the PMT hits, can be used to discriminate IBD signals from radioactive backgrounds. The IBD signals can only be produced with their vertices in the liquid scintillator, while the radioactive background can also be produced from the container and PMTs near surface. The radioactive contributions from PMTs are greatly dependent on the materials of PMT glasses. In this way, the event vertex information can be used to separate signals and backgrounds, although there may be some overlaps in their distributions. The energy spectrum of reactor neutrinos is also different from those from radioactive elements decays, which makes it possible to use the event energy as a variable in discrimination. So long as the event vertex and energy are accurately reconstructed, the signal and background can be classified to some extent.

To test the performance of signal and background discrimination with event reconstruction outputs, we generate 10,000 signal IBD events and 10,000 radioactive background events, run through the detector simulation and event reconstruction to produce the variables for classification. The 5 reconstruction output variables to be used as input for classification are: reconstructed energy, number of total hits and the three-dimensional Euclidean coordinate of reconstructed vertex.

A traditional neural network MLP is used to study the discrimination power with the reconstruction outputs. The architecture of the network is shown in Figure 3. The input layer has 5 inputs, which correspond to the 5 reconstruction output variables. The number of hidden layer is 3, with the number of neurons in each layer being 50 or 100. The output has only one neuron to determine whether this event is signal or background.

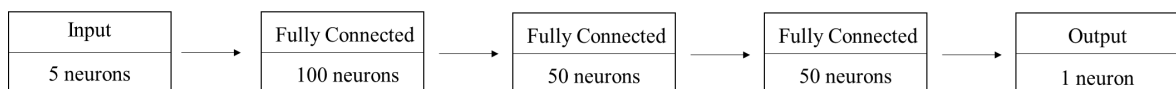


Figure 3. Architecture of the traditional neural network MLP.

The total 20,000 events of signals and backgrounds are separated into two samples, 18,000 events for training and 2,000 events for verifying. In training, we choose the Area Under the Curve (AUC) as the index of performance. The training result for AUC is 0.856, which will be used as a baseline for comparison with the CNN classification result.

4. Application of CNN for classification

4.1. 2D event display

The distribution of the fired PMT hits on surface of the spherical container can be projected to a θ - ϕ 2D plane to construct the image of the hits distribution of an event. For the PMTs with multiple hits, the total charges of all hits on the PMT are accumulated and the hit time is set to the earliest arrival hit's time. According to the difference on the production vertex of the event and the characteristic energy due to different particles in signal and background, the 2D projection event display can be used as graphical inputs for classification.

Figure 4 shows the typical 2D event display of a signal IBD event (left) and a radioactive background event (right). The scale of energy deposition is at MeV level. The signals are usually produced in the liquids in the container with a flatter hits distribution on the surface, while the backgrounds are produced closer to the surface with a sharper hits distribution on the surface. With the characteristics in 2D images, they can be used as inputs of CNN for classification. The images are low-level data with all raw information being kept. In comparison to the method using reconstruction outputs, no event reconstruction processes are required. It not only saves the time and computing resources, but also independent of the precision of any reconstruction algorithm.

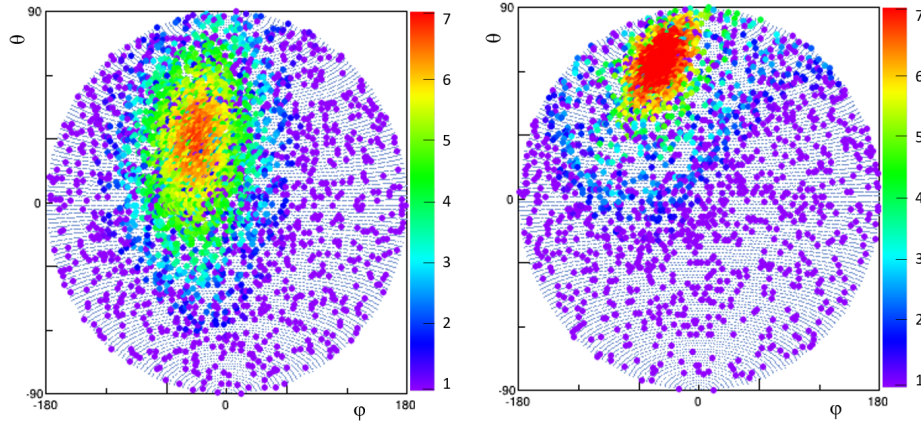


Figure 4. 2D projection of the hits distribution in a signal event (left) and a background event (right). The deposit energy in the events is on the scale of MeV.

4.2. CNN architecture

The size of input image is 314 by 329 pixels, which fully map the PMT arrangements on the surface of container. A CNN is constructed for classification with the image inputs. The architecture of the network is shown in Figure 5. The original 2D projection pictures are converted into RGB format and feed into it. There is a total of 5 hidden layers in the CNN. The first three hidden layers' structures are the same, they are all composed of two convolutional layers with 10 5×5 filters and one max-pooling layer with a 2×2 filter and its stride is set to be 2. The last two hidden layers are fully connected and each of them has 50 neurons. The output layer's activation function is the sigmoid function, the threshold is set to be 0.5 to determine whether the input is signal or background.

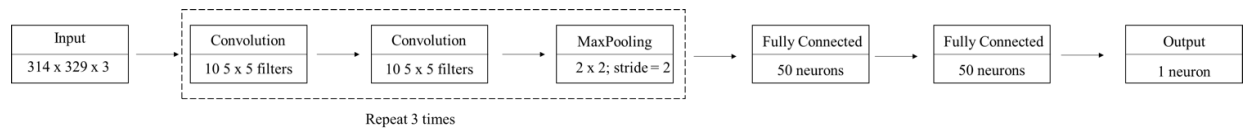


Figure 5. Architecture of CNN neural network

4.3. Training

The CNN model is trained on a GPU server with 4 NVIDIA Tesla K80 video cards. The same 20,000 event data samples in previous MLP tests are used in the CNN training. Instead of using the reconstruction output as input variables, the 2D hits projection images from each event are used as inputs

of CNN network. In training, we use the Batch-Normalization [13] and Dropout [14] strategy to speed up the training and prevent overfitting. We also adopt ADAM as the optimizer [15]. The training lasts for 30 epochs before it converges. The training curve for CNN is shown in Figure 6.

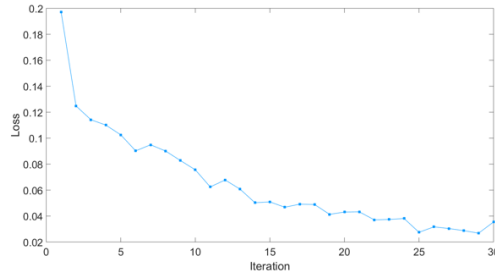


Figure 6. Convergence curve of the CNN in training.

4.4. Performance

The event classification results from MLP and CNN are listed in Table 1. We adopt AUC as the evaluation metrics to compare the classification power in different methods. As shown in Table 1, the CNN achieves much higher accuracy than MLP, which means that using CNN can significantly improve the performance of signal background discrimination without requiring any high-level feature. The classification accuracy is 97.4% for signals and 96.0% for backgrounds, respectively.

Table 1. Performance of event

	AUC	Variance
MLP	0.856	<0.001
CNN	0.975	<0.001

5. Summary

Deep learning has shown its powerful learning ability in different fields. It is skilful in handling massive data and finding the non-linear relationship between inputs and outputs. By applying the Convolutional Neural Networks in signal and background classification for a liquid neutrino detector model, we demonstrate that the low-level hits output can be transformed into images for event classification with CNN. Without knowing the high-level features of events from reconstruction, the signal and background discrimination power of CNN outperforms the traditional MLP method with reconstruction results. The toy detector model is designed to be a simplified version of the neutrino experiment JUNO. With further study, this method is expected to be applied in the realistic neutrino experiments in the future.

The authors would like to thank the supports from National Natural Science Foundation of China (11405279, 11675275), the Strategic Priority Research Program of Chinese Academy of Sciences (XDA10010900) and the Recruitment Program of Global Experts.

References

- [1] S. Agostinelli et al. "GEANT4-a simulation toolkit", *Nucl. Instrum. Meth.*, A506.3 (2003):250-303.
- [2] Y. Fukuda et al. (Super-Kamiokande Collaboration), "The Super-Kamiokande detector", *Nucl. Instrum. Meth.*, A501 (2003) 418-462.
- [3] F.P. An et al. (Daya Bay Collaboration), "A side-by-side comparison of Daya Bay antineutrino detectors", *Nucl. Instrum. Meth.*, A685 (2012) 78-97.
- [4] Z. Djurcic et al., (JUNO Collaboration), "JUNO Conceptual Design Report", *arXiv:1508.07166*.
- [5] P. Baldi, P. Sadowski, and D. Whiteson. "Searching for exotic particles in high-energy physics with deep learning." *Nature Communications* (2014):4308.

- [6] E. Racah et al., "Revealing Fundamental Physics from the Daya Bay Neutrino Experiment using Deep Neural Networks", *arXiv:1601.07621*.
- [7] A. Bagoly et al., "Machine learning developments in ROOT", *J. Phys. Conf. Ser.* 898 (2017) no.7, 072046.
- [8] K. Simonyan, A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", *arXiv:1409.1556*.
- [9] A. Aurisano. et al., "A Convolutional Neural Network Neutrino Event Classifier", *JINST* 11 (2016), P09001.
- [10] F. Rosenblatt, "Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms", Spartan Books, 1961.
- [11] R. Reed and R. Marks, "Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks", A Bradford book, MIT Press, 1999.
- [12] Y. LeCun, et al., "Backpropagation applied to handwritten zip code recognition", *Neural Comput.* 1 (1989) 541–551.
- [13] Ioffe, Sergey, and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", *arXiv:1502.03167*.
- [14] Krizhevsky, Alex, I. Sutskever, and G. E. Hinton. "ImageNet classification with deep convolutional neural networks" *International Conference on Neural Information Processing Systems* Curran Associates Inc. 2012:1097-1105.
- [15] Kingma, Diederik P, and J. Ba. "Adam: A Method for Stochastic Optimization" *Computer Science* (2014).