Neutrino Experimentation: with Machine Learning*

Mohamed Ali†
University of Washington's T2K Experimentation
(Dated: March 9, 2021)

In this paper we will first introduce neutrinos, their connection to computer learning algorithms namely machine learning and deep learning. we will then explore previous neutrino experiments that have been exposed to deep learning. We will look at how they have used these computer algorithms within their unique experimental setups. Finally, we will review our own T2K experimentation, and recommend portions of our experimentation where we can experience an increase in efficiency or accuracy in results by the implementation of deep learning algorithms. We will then discuss the T2k results and then speak on where our deep learning algorithms was used if used at all.

Usage: University of Washington M.S Physics student's research physics 600.

I. INTRODUCTION: MACHINE LEARNING'S ROLE IN NEUTRINO EXPERIMENTATION

A. Introduction to Neutrino Experimentation

Neutrinos are special particles that are many times lighter than other fermions, in fact there is no direct measurement for a non-zero neutrino mass. Neutrinos are also neutral meaning they have no charge and therefore also feel no charge. This means that neutrinos interact very weakly amongst themselves.

Neutrinos are considered the most exclusive and also the most abundant for of particle in our universe, we simply do not notice them because of their weird state. In fact, we swim in a sea of neutrinos much like how a fish would swim in a sea of water. "Approximately 60 billions of solar neutrinos, produced in the core of the sun by nuclear reactions, cross each square centimeter of our body every second" (Lipari). It is estimated that sun and the closes supernova have been the first neutrino measuring devices however new detectors would be able to detect neutrinos from other sources. Neutrinos are the hope of the future with regards to it assuming the responsibility of it acting as the messenger for distant objects within our galaxy. Our biggest problem from which this can be a reality is how small the neutrino interaction cross section is.

Neutrinos have two different types of interactions; "they can couple with a Z 0 boson, changing their 4-momentum but keeping their identity (neutral current interactions), or they can couple with a W ± boson "transforming" into one of the charged leptons e ±, µ ± or ± (charged current interactions)" (Lipari).

Bruno Pontecorvo has observed a phenomenal effect with regards to neutrinos; the phenomena of flavor oscillations. Neutrinos exhibit three neutrino "flavors"; each neutrino completes a doublet with a corresponding lepton which represent the three neutrino or charged lepton pairs. Their label represents the corresponding charge lepton which is produced in each association.

B. Introduction to Machine Learning

Machine Learning is a computational method used to perform tasks. Deep learning, which is a subset or a type of machine learning, is capable of performing more complex tasks. Deep learning is a subset of machine learning which means it still designed to take in an input and through a series of algorithms which produces an output based on the given inputs. What these algorithms do is essentially "learn", it does this through the use of something called a neural network to find the association between inputs and outputs. A neural network is simply a three-layer process comprised of the inputs, the learning stages and the output. The input layer takes in data, usually pre-processed to be able to be read by the learning stage. These are the computational methods used heavily within the analytic portions of neutrino experiments. In general machine learning is simply an artificial neural network (ANN) where neurons are interconnected in a network and each neuron performs a mathematical task. Therefore, each output of a neuron is the input of the next neuron, and together this network achieves learning.

These ANNs can be layered as well so that each layer can produce an output which is fed into the next layer of ANNs. This multi-layer perception (MLP) is an example of a feed forward network. The number of hidden layers and their interconnected array of neurons allows the network to perform complex tasks. This process of learning happens iteratively, and this iteration continues until the network's ability to approximate the function's behavior no longer capable of improvements, this of course is when the closest approximation to a solution can be given.

* A footnote to the article title
† http://www.Second.institution.edu/~Charlie.Author

C. Machine Learning vs. Deep Learning

Deep learning is a little more complex than traditional machine learning; deep learning uses more nonlinear operations so that mapping from results to input variables is a little harder to map. These processes have been made easier through the use of GPUs; a process that especially dominates major upcoming fields in computer engineering such as computer vision. One of the most common deep learning algorithms employed for pattern recognition is CNN (Conventional Neural Network). Key components of CNNs are kernels or image filters; these are image filters set up as two-dimensional arrays or matrices that takes in an input image and produces an output image with highlighted features. Producing a CNN is done by establishing some important layers such as the initial layer where conventional layers serve the purpose of feature extraction, followed by several training layers processes to extract features that are most useful for the desired task. There is also some optimization that happens during this layer to reduce images needed to be sampled. The final layer s of the network then performs the classification or regression task using the extracted features as input.

Ultimately deep learning algorithms differ from machine learning by the complexity of the problems this program is solving. Although Deep learning is a subset of machine learning, deep learning algorithms differ through the depth of the neuron network and the amount of layering present when building the CNN. While machine learning almost always requires structured data, deep learning requires layers of ANN (artificial neural networks). Also, deep learning algorithms do not require being taught like a machine learning algorithm, this simply means that there is no need for human intervention as the multilevel layers in the neural network arranges data in a hierarchy of different concepts. This ultimately makes the network learn from its own mistakes provided that the data quality is sufficient for the above-mentioned process. The task of identifying signals and reconstructing physical characteristics in interactions in particle detectors is similar to the usual task of identifying images, a dominant use for deep learning in computer engineering today, making deep learning perfect for analytic neutrino physics.

D. Computational Methods in Neutrino Experiments

Particle physics experiments require the analysis of large amount of data, complicated modeling and some heavy statistics. Similarly, neutrino Physics also require analysis of large data sets. This includes the need for sophisticated modeling and the computation of some complicated statistics. This has really been where the marriage of neutrino experiments and ML has happened especially with the emergence of neutrino oscillations.

With the emergence of neutrino oscillations and the rapid scaling and use of deep learning, it seems inevitable that there would be the marriage between neutrino physics experiments and deep learning. "The SNO experiment was the first to explore the use of neural networks in neutrino physics,13 using feedforward networks, a type of artificial neural network to classify events based on hit pattern features". Although deep learning algorithms were first deemed not the best for the complex statistical work needed in particle physics, were it really performed was in event identification especially in neutrino detector data. Tools like feedforward networks, MLPs and decision trees have really dominated the arena of particle physics, especially with regards to experimental design such as; hardware triggers, energy estimation, reconstruction, and signal selection. The greater performance possible by deep learning tools have been the main reason of their implementation in particle physics over previous other classical methods.

II. DEEP LEARNING IN NEUTRINO EXPERIMENTATION

As we know industry has a huge impact on deep learning, mainly as it dictates the import tools needed with regards to the climate of the industry at that time. With the growing emergence and significance of what were deemed academic areas of interest within major tech companies, the marriage of academia and for-profit industries is higher than ever before. This has caused a academic institutions to take advantage of investments made by corporations to develop areas of interest with regards to areas of interest by these major for profit companies. This has been the cause with the use of deep learning within neutrino experimentation. This growth in interest within computer science has further pushed the marriage between neutrino physics and deep learning along.

A. Challenges of using Deep Learning in Neutrino Physics

B. Adaptability of Methods

We know that some of the most commonly used algorithms for neutrino experimentation is algorithms derived for image recognition. This makes sense given that a lot of neutrino experimentation setups involve two-dimensional images mapping.

Although there are similarities between the typical neutrino experimentation set up with regards to data input types and even core objectives and classical image recognition, there are some important differences; obviously these differences call for important adaptations to be made to conventional image recognition algorithms. These adaptations can range from tasks as simple as converting detector data into image-like tensor inputs or can be as intensive as like implementing an entire network redesign for these algorithms to be applicable to your specific neutrino experimentation. Of course, we must keep in mind that the more drastic the adaptation of a classical deep learning algorithm to your specific input and output, the more drastic the performance of the algorithm with respect to your task. Although a significant amount of work is required upfront with respect to making sure your algorithm performs as expected, this is necessary because of how unique your particular neutrino experiment can be. One example of how deep learning adaptation has been implemented is the GoogLeNet CNN architecture. GoogLeNet was the first implementation of convolutional layers in CNNs in a non-sequential way. GoogLeNet is a 22-layer deep convolutional neural network that's a variant of the inception network, a Deep Convolutional Neural Network developed by researchers at Google. Another experiment that shows how unique the customizations of a deep learning algorithm can be is the NEXT experiment.

The NEXT experiment is a successful application of GoogLeNet, this experiment is a background rejection network. How it works is the NEXT experiment has cylindrical time projection chambers with photon and charge detection serving as detectors. "Photomultiplier tubes collect a light signal and silicon photomultipliers (SiPM) collect an electroluminescence signal from drifted charges inside the detector". Now the data from SiPM redout is reduced to a 3D model of dimensions x, y and z so that the training inputs of the CNN can resemble 2D image data. This simple use was found to outperform traditional methods of reconstruction by a factor of 1.2 to 1.6. Actually, as a result of the success GoogLeNet has found with the implementation of neutrino experimentation, many other neutrino experiments started to address their needs and how deep learning can improve the efficiency of their unique experimentation. Despite the differences between classical and neutrino data, deep learning proved better than other classical methods in its ability to conform to a specific experimentation's uniqueness efficiently.

1. *Quantifying Bias Uncertainties*

There are some risks of applying machine learning to neutrino experimentation, one of which being the neural network will extend beyond what is necessary or even what is wanted. This problem is really referring to the possibility that the data set that was set to be used at the beginning of the neural training has information or structured in some way that would result in incorrect bias. The results, of course, would just throw up of the results which would render the entire algorithm pointless. This problem is further exaugurated when dealing with tasks involving feature extraction. Since the networks used for detector data analysis are typically trained on simulated datasets of the event of interest, the particle interactions and other physical processes; it is off utmost importance to quantify bias and limit this as much as possible. This can be done through careful choices in the construction of the input data. It is possible that biases exist and is unknown to the developer.

2. *Network interpret-ability*

Another challenge in using ML to interpret results is the inevitable tradeoff between performance of the algorithm and the ability to interpret results. As a model grows larger and deeper often the problem of results validity becomes ever so apparent. One example highlighted this problem in a scenario where "Boosted decision trees, for example, are low-level machine learning models. They can often inform the user of the relative importance of each input into the model but may not have the accuracy that can be achieved with deeper models. CNNs on the other hand, have achieved state of the art performance on many tasks, but the features extracted by the convolutional layers are abstract and challenging to interpret" (src1). In neutrino experiments it is very important to be able to connect your core objective or task with regards to the construction of your deep learning model. A much deeper understanding of the physical features used by the network would be much better for minimizing the inefficiencies in the performance of the algorithm. Dimensionality reduction is a common method of interpreting the features extracted by a network. The Daya Bay Reactor Neutrino Experiment, for example, is designed to detect anti-neutrinos produced by two nuclear reactors. The advantages of a CNN and it's properties were vital in this experimentation to separate inverse beta decay events, which is the signal of interest in this experimentation, from noise in the detector. They use t-Distributed Stochastic Neighbor Embedding (t-SNE) to transform the network into two dimensions. This method uses non -linear transformation to "reduce the dimensionality of data in a way that maintains the distance between points local to one another". Another method of dimensionality reduction is Principal Component Analysis (PCA). This method uses a linear change basis where "the new basis has vectors along the dimensions of maximum variation in the data". Usually, you only need to explain a few of the basis vectors with this method, the new basis vectors are orthogonal which simply means they each have a unique contribution to the variation in the data. PCA is usually used to limit the number of input data to highlight the input values specifically needed for the task.

3. *Computational System Constraints*

The latest developments in deep learning are largely driven by improvements in GPU technology where the many computations needed for large networks can be done in parallel. The many computations performed by a neural network is compounded by the amount of data collected in particle physics experiments. Modern neutrino experiments record billions of events which require evaluation by many reconstruction and analysis algorithms further increasing the number of computations a network must make at every layer. Many experiments perform these evaluations on large-scale computing grids on CPUs. While neural networks have significantly expanded the capabilities of many neutrino experiments, the data intensive nature of particle physics experimentation has produced a limitation to the widespread use of very deep neural networks. Here we consider three methods to alleviate this concern. One potential solution is to expand the availability of GPUs. Small GPU clusters used for training neural networks are becoming more common. However, these are not enough to match the production needs of many neutrino experiments. Larger availability of GPU clusters would enhance the ability of experiments to utilize large neural network-based algorithms. Machine learning based methods often show significant improvements over traditional methods and one-way to improve performance is to maximize the primary task algorithm, but the implementation of multi-task algorithms could be a promising way to enhance the total physics output from an individual algorithm.

III. DEEP LEARNING IN NEUTRINO EXPERIMENTATION: MINERVA

A. Data-set

The data set for this experiment was made up of almost two million simulated events, these events were simulated through the use of a GENIE Neutrino Monte Carlo generator. The representation of each one of these events was in the form of a two-dimensional array of energy values, a matrix. Normalization was also applied so that the greatest energy response was set to a value of 1.0 for every image. The data set was then split into a series of x, y and z view images where the x view images where set to 127x50 and the y and z images were set to 127x25; all of these representing a compressed view of the full MINERvA detector. The orientation of the images was such that the left side represented the side where the beam enters the detector. "A segment ranging from 0 to 10 encoding the vertex region, as seen in Figure 3 and as a floating-point Z value, ranging from 0 to 8000 corresponding to the precise vertex location. Therefore, the problem may be cast as both a segment classification task and a regression task"; and so the data set was set. Their core objective remains to be locating the vertex in the generated image, the most important motivation for the use of deep learning is simply vertex reconstruction and so deep learning's ability to infer complex patterns leads to better or more accurate classification accuracy.

B. Method

Convolutional neural networks were used within this experiment, which is the best method to use with regards to segment classification. Columnar structure is used and the weights of each convolutional layer within each column is independently learned. Within each column contains "four convolutional layers, four max pooling layers, a fully connected layer" they then applied a rectified linear unit (ReLU) activation function to both convolutional layer and connected layer of each column. They implemented a dropout at a rate of 0.5 at each fully connected layer; the number of outputs steadily increased after each convolutions layer. The best performing network found was then an Adagard solver was applied with two hundred thousand iterations on a small batch of about five hundred. They then used stochastic gradient descent solver or SGD to isolate improvements in performance. This process also held true for the task of vertex reconstruction to regression. The created final fully connected layer had a single output which represented the Z location. Finally "two methods of encoding the Z location as a target value were explored: (1) naive encoding of the Z location as the position within the detector in millimeters and (2) removing sparse or empty regions from the data and normalizing the remaining regions between 0 and 1".

C. Results

IV. T2K EXPERIMENTATION

A. T2K Introduction

B. T2K Introduction

The T2K experimentation is a long baseline neutrino oscillation experiment which is designed to facilitate the mixing of the muon neutrino with other species in the hope that we can shed light on the neutrino mass scale. The very first long baseline experimentation was designed to look specifically at the electron neutrino appearance from the muon neutrino and then measuring the angle theta. This angle theta represents the last unknown mixing angle in the lepton sector. T2K uses a detector to measure neutrino rates at a distance of about three hundred kilometers from the accelerator. The experimentation also includes a neutrino baseline and a near detector.

C. Deep Learning's place in T2K

D. T2K Result's Review

The `widetext` environment will make the text the width of the full page, as on page **??**. (Note the use the `\pageref{#1}` command to refer to the page number.)

a. Note (Fourth-level head is run in) The width-changing commands only take effect in two-column formatting. There is no effect if text is in a single column.