ANALYSING DETECTOR IMAGES FROM THE NOvA EXPERIMENT USING CONTEMPORARY MACHINE LEARNING TECHNIQUES

UNIVERSITY COLLEGE LONDON, GOWER ST.

YUWEI ZHU

Abstract – Simulated NOvA experiment detector image data and corresponding metadata, created using GEANT4 and GENIE were analyzed with contemporary machine learning techniques. A binary classification network, inspired by recent works and the GooGLeNet network architecture designed for the Imagenet competition, was created and tuned towards separating charged – current muon neutrino events from other background data from the detector. Further regression analysis was done on the metadata, to reconstruct the incoming neutrino and resulting lepton energies, which are crucial to determining the probability of neutrino oscillations. Finally, more complex multiclass classification tasks of: the number of protons/pions present in the detector image, and classifying the interaction mode were attempted and the results documented.

I. Introduction

Experimental history

The recent progress in research around neutrino oscillations has been groundbreaking for modern physics, and will continue to have massive impact on future discoveries and our understanding of the universe. The journey to finding this quantum mechanical phenomenon started during the pursuit of answers to a “solar neutrino problem” which was first encountered in the mid-1960s [1]. The basic fusion reactions that take place inside the sun produces neutrinos [1], some of which eventually penetrate underground beneath the Earth’s surface. A solar neutrino detector was devised and created, which would go on to run for 3 years [1], yielding surprising results. The cosmic ray level of neutrino flux was measured to be significantly lower than the standard model predicted at the time [1], which caused much speculation.

Around half a century later, a breakthrough came with the discoveries of the Kamiokande and Super-Kamiokande (SK) Collaborations [2]. Analysis was completed on atmospheric neutrino data collected from 33.0 kiloton-years, or 535 days exposure of the SK detector. One particular ratio R measured by the detector, calculated using equation 1 where μ and e are the number of muon-like and electron-like events found by the detector (data) and Monte Carlo (MC) simulation, was eventually cited as evidence for neutrino oscillations [2]. R took values significantly smaller than expected. No combination of uncertainties in the data or known theories at that time was able to explain the data [2].

$$R \equiv \frac{(\frac{\mu}{e})_{Data}}{(\frac{\mu}{e})_{MC}} \tag{1}$$

The main theory to explain the data was a two-neutrino oscillation model either between muon neutrinos and tau neutrinos, or between muon neutrinos and a new, non-interacting “sterile” neutrino, consistent with the observed flavor ratios and zenith angle distributions over the entire energy region [2]. The electron neutrino was left out of the model due to its negligible activity in atmospheric mixing, which is the data SK was collecting.

The current neutrino oscillation model describes the process of a neutrino being created in a certain flavor – tau, muon and electron – and changing into another neutrino of a different flavor over a period of time, with a certain probability, described in equation 2. This can happen several times during a single journey of the neutrino.

$$P_{\alpha \to \beta} = sin^2_{(2\theta)}\, sin^2 \left(1.27 \frac{\Delta m^2 L}{E} \frac{[ev^2][km]}{[GeV]}\right) \quad (2)$$

The other theory of sterile neutrinos was recently made highly unlikely through the MiniBooNE experiment, which found "severe constraints" on the standard 3 + 1 sterile neutrino hypothesis [3]. Several null oscillation results [4 – 8] from various experiments strongly disfavored the sterile neutrino hypothesis, and while adding more sterile flavors to the hypothesis improves it, it only does so marginally [3].

The follow on from this experiment, called MicroBooNE, aims to examine problems that couldn't be solved by MiniBooNE. It is currently running, and the full results are yet to be seen.

Background for the NOvA experiment

The focus of this paper is solely based on data from the NOvA experiment. This experiment was created with the goal of taking measurements of neutrino oscillation parameters [9], some of which are shown in equation 2. The experiment also seeks to understand a phenomenon not predicted by the current standard model of physics.

The experiment is conducted by Fermilab [10]. The neutrino beam is "created" at their Main injector. The setup works by firing neutrinos created from the injector towards two detectors; one "near" detector which is around 1km away from the source of the beam, and a "far" detector around 810 km away [10]. The beam of neutrinos is initially created by firing protons at a graphite target – upon collision, there is a creation of many different fundamental particles including pions [10]. These pions eventually decay into muons and muon neutrinos. Both detectors contain 344 thousand cells of highly reflective PVC filled with plastic scintillator [10]. The scintillator is made out of solvent molecules and fluor molecules, which work in tandem to emit photons once charged particles collide with the scintillator molecules. Charged particles are created through the collision of the neutrino, and the energy of these particles is recorded as they come to rest. Using the patterns of scintillation light, it can be determined which neutrino flavor caused the interaction and its energy [10].

These interactions within the detector can be divided into two main types of events. One of these is called "charged-current" event, and the other is called "neutral-current" event [11]. In charged-current (CC) events, which happen via the exchange of W+/W- bosons [11], the neutrino interacts to become its charged lepton partner. In a neutral current (NC) event, which happens via the exchange of Z bosons [11], the neutrino just scatters inside the detector.

The charged-current events, can be further divided into three categories. These are the quasi-elastic (QE), resonant (RES), and deep-inelastic scattering (DIS) categories [11]. Each interaction mode dominants in various neutrino energies [12]. Specifically, QE events are known to dominate at low energies, RES at medium energies, and finally DIS, which dominates at large neutrino energies [12].

Experiments in high-energy physics use a plethora of different analysis techniques to the properties of various subatomic particles [14]. Recently, following the developments in the field of deep learning, there have been several implementations of machine learning algorithms in the form of neural networks to aid data analysis of the NOvA experiment [13-14], with reasonable success. It is thought that the classification of interactions/events inside a detector shares many similarities with image recognition, in the sense that topological features are useful in both tasks [14].

In certain areas, such as cosmic background rejection, neural networks already outperform traditional rejection algorithms in all sample areas [13]. In recent years, convolutional (CNNs), adversarial and recurrent neural networks (RNNs) have been used in the field of high-energy physics for a variety of different tasks [14].

A traditional neural network consists of a sequence of layers, and each layer is made up of neurons. The connections between individual neurons gives rise to hyperparameters such as weights and biases [15]. The neural network learns by tuning these hyperparameters to minimize a loss function, which is a function which describes the distance between the current model prediction and the label/answer [15]. It does this through an algorithm called backpropagation, which calculates the gradient of the loss function with respect to every hyperparameter, and conveniently updates each hyperparameter in the opposite direction to this gradient after every forward pass of the network, reducing loss [15].

The basic inner workings of a CNN can be described as a sliding "kernel" which creates a new image by multiplying pixel values of the image with the values in the kernel [15]. CNNs can also be implemented with other techniques, known as pooling or 1x1 convolutions, to reduce the dimensionality of the image data, which reduces the computational load [15]. Pooling also reduces the sensitivity of the network to the absolute location of certain features of the image.

A recent paper published in 2016 outlines a particular method to classify detector events taking inspiration from the GoogLeNet architecture [13], which performed very well in the ImageNet competition task of that year.
There is a "sub-network" present in the GoogLeNet architecture, that will be used in the network used in this paper, called an "inception module" which extracts features of different scales and concatenates the result. This is shown in figure 1.

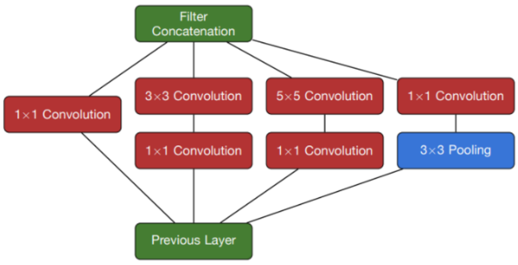

Figure 1: An inception module

The LRN layers stand for “local response normalization”, which normalizes the output of the previous layer.

II. Methodology

The dataset is given in the form of .h5 files and can be found at ref. [16]. The detector images are given as a concatenated array of two separate images: one image shows a top view (x -z plane) of the detector, and one shows the side (y-z plane) view.

One particular approach to handling this image data would be to pass it as a single input into the network, but a more robust way would be to separate the images and process them separately.

The second approach is a method called mixture of experts, where multiple networks are used to process different data, before the “judgements” of each expert, or network, is combined. The overall network decides what weight to give to each of the “experts” and the judgement of each is combined to determine the final decision of the network. In this case, two such experts could be used, one to process the x – z plane view of the detector, and another to process the y – z plane. This illustrated in figure 2.

The most common network architecture used throughout this experiment is a simplified version of the network in figure 2.

The entire project will be conducted on a Windows Surface Pro 6, with 8GB RAM, Windows 64-bit operating system, and Intel Core i5 – 58250U CPU 1.60GHz. The code is run on the Google Colab IDE, where the project has access to a NVIDIA-SMI P100 GPU, with runtime set to High-RAM mode.

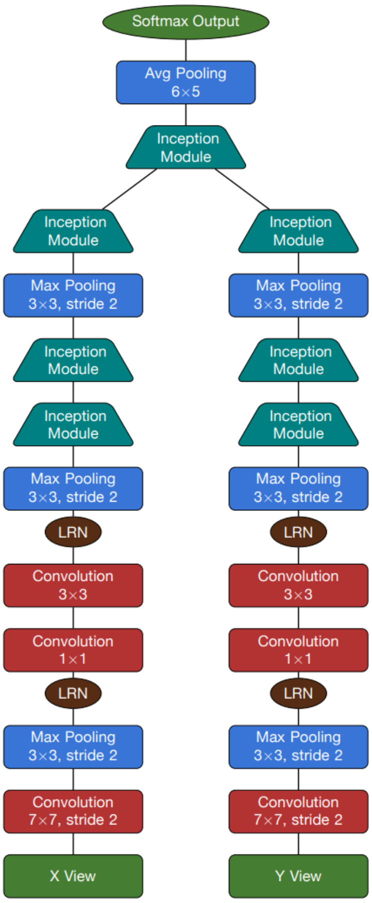

Figure 2: GoogLeNet inspired network architecture [13]

Classifying CC muon neutrino events and background events

The first step in any deep learning/supervised learning task is obtaining training data, and preferably lots of it. For context, one of the most known and widely used data sets, the MNIST handwritten digits dataset contains around 60000 images. For this experiment, UCL provided a collection of total 403 files of simulation of detector images from the NOvA, simulated GENIE and GEANT4 [16]. Each file contains varying amounts of data, but there are usually around 7000 x 2 images, plus 16 pieces of metadata corresponding to each image, such as interaction type, final state, lepton energy, neutrino energy, and more. Not all of the metadata will be utilized.
For each machine learning task, it will not appropriate to use every single piece of data and

image in the dataset, due to data balancing, but also the diminishing returns in model accuracy for the increase in needed computational resources. For the first task, classifying CC neutrino events, a total of 30 files were used.

The files are initially composed of roughly 88% CC muon neutrino events, with only 12% background events. This data imbalance is a common problem in machine learning. Training a ML model on such biased data often results in disastrous results. The ML model usually ends up learning the majority about the dominant class, and struggles to classify the minority class.

A model can achieve up to 89% accuracy on a test set, which is a deceptively good result. However, after reviewing the model's prediction, it was found that the model was predicting one class only, i.e., predicting only CC events. Due to the data imbalance, the model achieved high accuracy, as the test set happened to contain around 88 – 89% CC events. In other words, the model coincidentally achieved a very good result. This result becomes trivial, however, when you consider the fact that the model is just baseline; predicting the same result no matter the input. This same model would theoretically get around 0% accuracy for a test set consisting of only NC events.

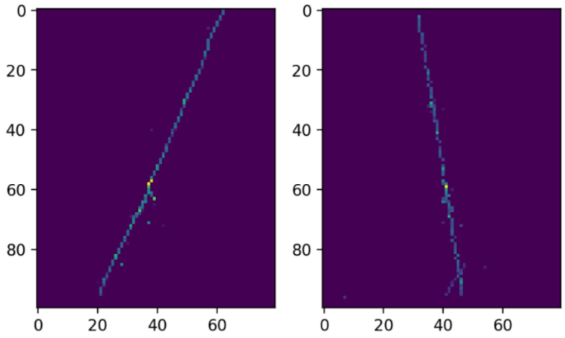

Figure 3: A pair of detector images

So, it was incredibly important to preprocess the training data to make it suitable to be passed to the model. This process involved implementing a function that only extracted the same amount of each unique class from each data set. Although a lot of background events were filtered out, the final dataset used for training was now a 50/50 split between CC/background events. This resulted in around 48 thousand images, so around 24 thousand per class. A new labelling system was implemented, by changing the 'interaction' number of each image/event into a new, Boolean variable, which represented a CC event with 1, and a background event as 0. In this case, all interaction types from 0 to 3 inclusive were classed as CC events, and the remaining – 4 – 15 were background.

Image data was reshaped into a format suitable for CNNs (None, 100, 80, 1). Following this, around 1% uniform noise was added to the training data, to promote model generalization. After, the image data was shuffled to prevent the test, training and validation sets from containing data that was not representative of all of the data set. Then, the image pixels were normalized.

The data set was split 85% 5% and 10% for the training, validation and test sets respectively. The validation set was kept as small as possible, as it doesn't have a large role to play except to indicate the performance of the model on the test set during training, so we can save the best weights. It is always optimal to have as large a training set as possible to prevent overfitting. It can be noted that a validation set was not even used in other similar experiments [13 -14].

Regularization techniques were employed, such as dropout, 40% just before the dense layer at the end of the model. Dropout prevents overfitting by removing particular features learnt by the model randomly, to reduce the number of useless features/features specific to training data. The LRN layers deal with a concept called "lateral inhibition", which comes from neurobiology. It involved normalization the output of the previous layer, especially effective when used in tandem with RELU activations, as they are "unbounded". This has the effect of boosting the sensitivity of already excited neighbors, and dampens the neighboring neurons, which effectively aims to increase sensory perception [15].

There was a total of 4 network architectures tested, to compare and contrast results. All models were created using Keras Functional API, which is used to create more complex models with multi-input/multi output models. A basic densely connected network was trialed, as well a standard CNN model architecture. The similarity between these simpler models is the fact that they process both images as a concatenated input. The last two models trialed were the more complex mixture of experts' models, one a simplified version of figure 1, and the last one is the most complex version, with three experts, taking three separate inputs, the x view, y view and the concatenated input.

III. Data analysis

After the data sets were prepared, the models were trained and then tested. The performance of the four networks is shown in table 1.

Network	Best accuracy observed
Dense	79%
Standard CNN	83.3%
GoogLeNet	84.5%
Adapted GoogLeNet	85.4%

Table 1: 4 different network performances on the classification task

As expected, the performance of the GoogLeNet inspired networks performed much better on this classification task.

There is, of course, a difference in computational resources and time required for each model, which increases down table 1. The few percentage points increase from the standard CNN to the GooGLeNet inspired architectures may not be a huge improvement on this scale, but for the NOvA experiment, which has access to more computational resources, the improvement could be beneficial. Also, as the detector will process millions of images over time, the few percentage points could be a difference of a few thousand images.

Neutrino energy

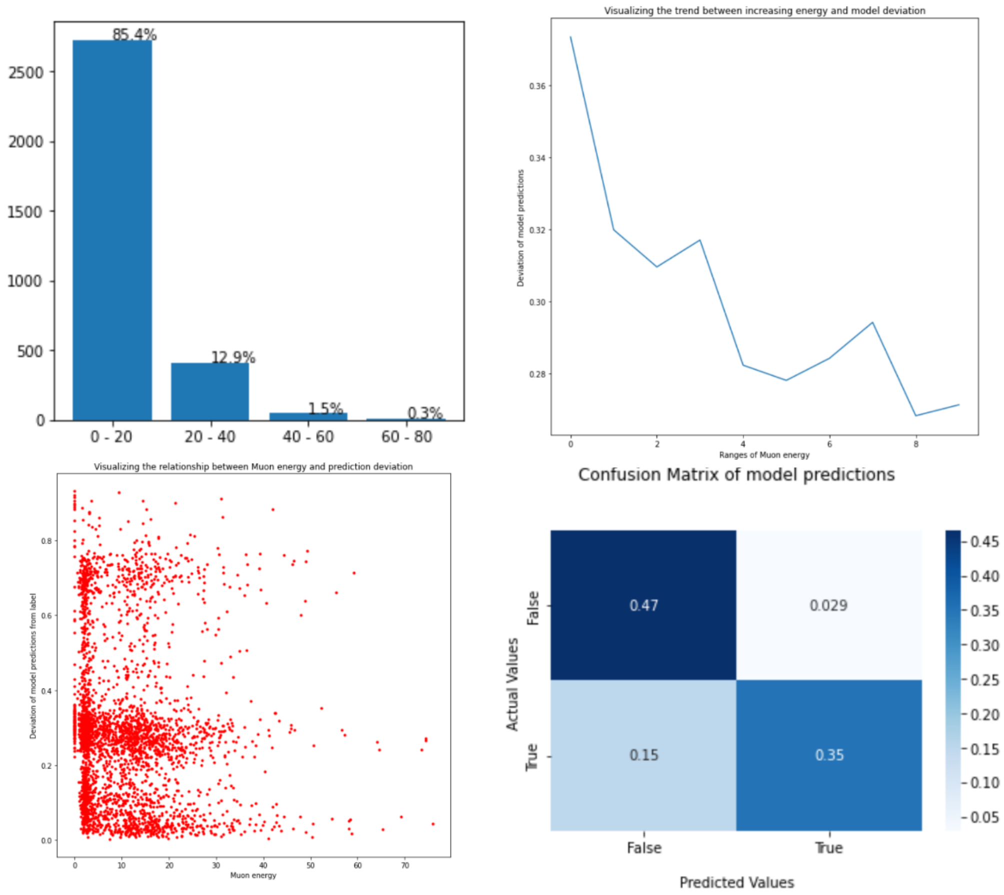

Figure 4: Analysis graphs on neutrino energy metadata and how neutrino energy affects network performance

From the confusion matrix, a useful and popular way to display the performance of a network, shown in figure 4 bottom right, it can be seen that around 82% of all images were identified correctly. The other quadrants show the percentage of images that were classified wrongly; separated into false positives – 2.9% - and false negatives – 15%. Overall, it should be remarked that in the context of this experiment, false negatives are more problematic than false positives. This is due to the aims of this study – to explore the capabilities of contemporary machine learning methods on the task of classifying NOvA neutrino event. If an algorithm such as the ones created in experiment were to completely automate the process of sorting and finding certain event data, then it is preferrable to have false positives. To illustrate this concept, imagine a situation where many of these images were being analyzed – if false negatives are reported, it would require scientists to look through the images that were dismissed by the model anyway, due to the possibility that a particular event was missed by the algorithm, making the model almost redundant. However, in the other case, if the model never reported false negatives, but some false positives, then at least the model has reduced the data set that needs to be sorted through.

Observing figure 4 bottom left, which portrays the relationship between neutrino energy and model prediction deviation, a few interesting features can be seen. Firstly, there is a clumping of lower energy neutrinos, which is the dense region near x = 0, which is a trend also confirmed in figure 4 top left. After analyzing the energy values, there are a few zero energy neutrinos. This is impossible, of course, but these represent more "subtle" neutrino interactions where the detector equipment was not sensitive enough to measure the energy of the neutrino.

There is also a clear clumping of deviations about 0, 0.3 and 0.7. It is uncertain why these deviation values are so popular with the model, but it could be due to a lot of images being quite similar in terms of features. The clumps also illustrate one other interesting point – there is a group of images, probably with similar features or similar underlying metadata that cause the model to consistently perform badly. Viewing figure 5, bottom right, there is a representation of the predictions of the network. There are clearly two main clump regions, where the model "prefers" to predict. This is partly due to the activation function on the final dense/output layer, which uses sigmoid. Sigmoid is a curve function that takes max and min values at zero and one. This particular function "prefers" to predict close to one and zero limits, due to the nature of its shape. Only a small range of values near zero make the network predict "middle" values between zero and one.

There is also a subtle pattern – the clumping of data points at the higher deviations is slightly reduced as the energy of the neutrino increases. This feature is emphasized more in the graph in the top right of figure 4. This graph depicts the relationship between neutrino energy and average model deviation. The averages were calculated based on even splits in the data – for example, the first average is calculated over the model deviation of the lowest 10% of neutrino energies, the second 10 – 20%, etc. There is a clear negative trend, which implies that the model performs better on neutrinos with higher energy, compared to lower energy neutrinos. Overall, between the group of lowest energy neutrinos and highest group, there is a 10% difference in model deviation.

This result is expected. Neutrinos are generally very hard to detect due to having no charge. Neutrinos have a variety of different energies, depending on the process that formed it – high energy reactions create high energy neutrinos. The higher the energy of a neutrino, the more interacting they are, as opposed to low-energy neutrinos, such as the neutrinos originating from the big bang, which are only weakly interacting. From examining the detector images by eye, there is a clear trend for high energy neutrinos as opposed to low energy ones. High energy neutrinos tend to "penetrate" further into the detector, leaving longer, straighter tracks. Presumably, this is easier for the model to identify, resulting in higher accuracies on high energy neutrino images. For lower energy neutrinos, the detector images are less distinctive; often the detector images for this group tend to have less activity, containing shorter, more curved tracks that do not travel far into the detector. This trend is not consistent of course, and occasionally, there are high energy neutrinos that also leave shorter tracks characteristic of the low energy neutrinos, which is likely where the model loses accuracy.

Lepton energy

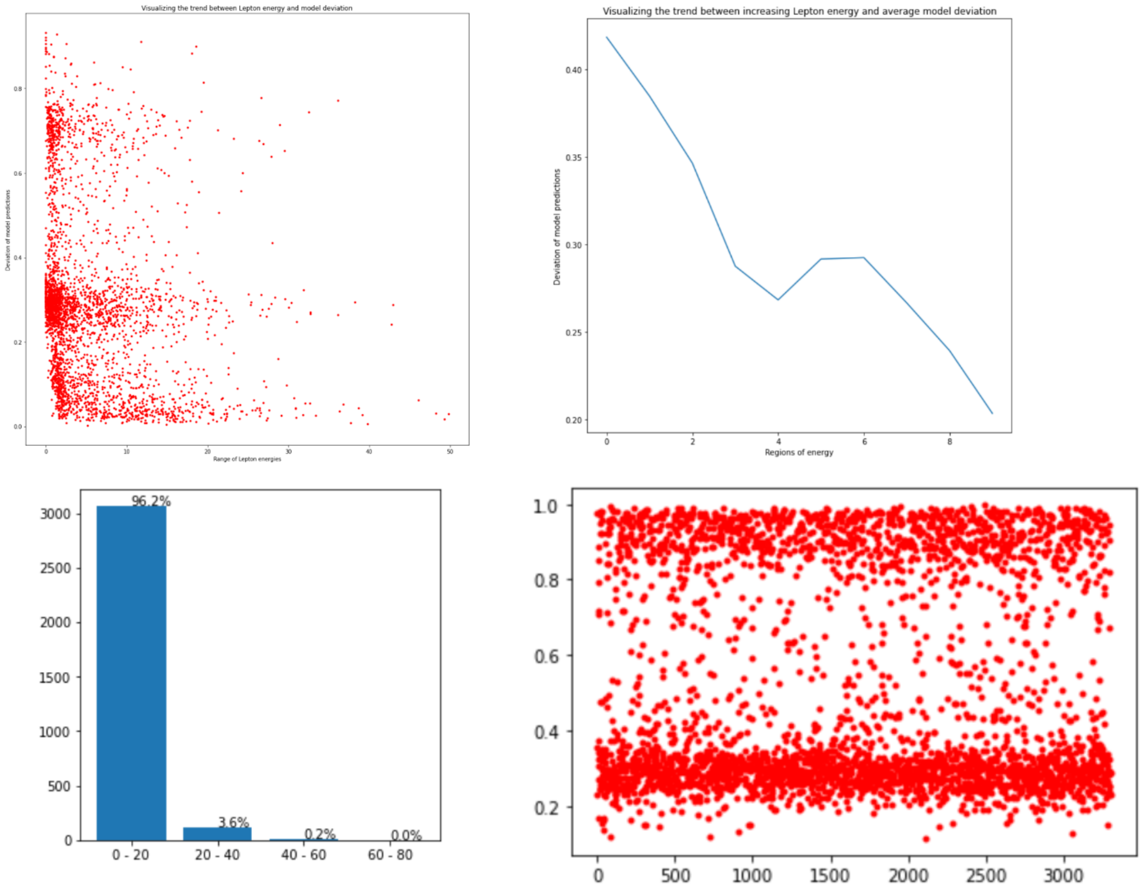

Figure 5: Analysis graphs on lepton energy metadata and how lepton energy affects network performance

Figure 5 shows the results to completing similar analysis on the lepton energy. The graphs show quite intuitive results. Leptons are created by neutrino charged current interactions. Therefore, by conservation of energy, higher energy neutrinos that undergo interaction is more likely to produce higher energy leptons. However, the conservation energy also means that the lepton energy cannot exceed the neutrino energy. In fact, if additional particles/photons are created, the lepton energy should be less than the neutrino. This is all in agreement with the graphs; looking at the top left, a similar patten is seen here as with the neutrino energies. The graph in the bottom left shows that the distribution of lepton energies is skewed even further towards zero – indicating overall less energy. The maximum energy is also around 50 GeV, whereas the highest energy neutrino was between 70 and 80 GeV. Lastly, looking at the graph in the top right, the same trend is followed by leptons; higher energy generally results in better model performance. As it has been concluded that higher lepton energies correspond to the higher energy neutrino events, this conclusion is not surprising.

Interaction modes

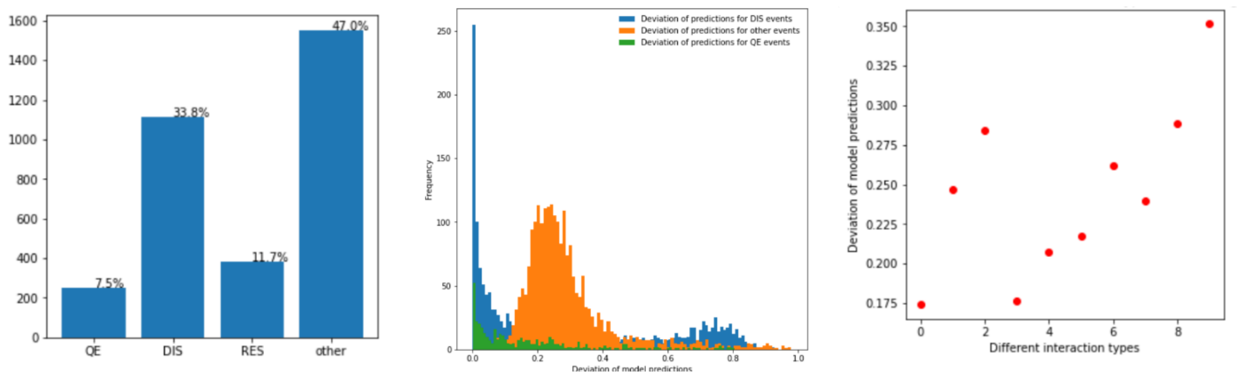

Figure 6: Analysis graphs on interaction modes and how interaction modes affects network performance

Interaction modes	Average model deviation
QE	0.1761
RES	0.22447
DIS	0.2819
other	0.2899

Table 2: Average model performance (defined by deviation of pred. to label) of each interaction mode

In this section I will analyze the model performance based on QE, RES, DIS and background category "other" which refers to CC interactions that cannot be classed as the three above, neutral current interaction, or cosmic background.

$$v_\mu + \mathrm{n} = \mu^- + p \tag{3}$$
$$\overline{v_\mu} + \mathrm{p} = \mu^+ + n \tag{4}$$

$$v_\mu + \mathrm{n} = \mu^- + p + \pi^0 \tag{5}$$
$$v_\mu + \mathrm{n} = \mu^- + n + \pi^+ \tag{6}$$

$$v_\mu + \mathrm{N} = \mu^- + X \tag{7}$$
$$v_e + \mathrm{e} = \mathrm{e} + v_e \tag{8}$$

Equations describing QE interactions are shown in equation 3 & 4, RES shown in equations 5 & 6, and finally DIS interaction in equation 7. Equation 8 indicates the interaction of an electron neutrino. Tau neutrinos are not present in our entire data set. Observing figure 6, the left graph shows the composition of the images used in the test set. As can be seen, the data set is dominated by events in the "other" category, followed by DIS events. The minority classes in the data would be the QE and RES events.

Table 2 shows the average deviation for each interaction mode. It is very clear that there is a pattern in model deviation, when compared to the interaction mode. The model performs better on QE events on average, which are generally considered to be “cleaner” events, leaving two tracks in the detector image usually. The performance of the model is second best for RES events, then DIS and finally the “other” category. This is a reasonable result, as it would be assumed that “messier” events such as DIS would be more difficult for the model to understand and interpret. It could also be assumed that messier events have less order and regularity compared to QE events, which would also affect the performance of the model. This is also shown in the histogram, where the distribution of model deviations for each interaction mode is shown. This is interesting, because part of the DIS set is well classified (the blue distribution peaks around zero deviation), similar to the performance of QE interactions, but seemingly the DIS deviation is much higher due to a smaller group of images that produce model deviations in the range 0.6 – 0.8. It is uncertain why this is the case, but it can be speculated that it is due to DIS events being dominant at higher energy, where the model performs better. However, DIS events which happen at lower energies could account for the DIS clump of higher deviations.

Energy reconstruction tasks

Energy reconstruction is crucial for the NOvA experiment, as neutrino oscillations and differential cross-sections are functions of neutrino energy [17]. More specifically, the energy of incoming neutrino particles is a parameter of oscillation probability, as shown in equation 2. Therefore, the accuracy of neutrino energy construction constrains the precision to which physicists can estimate the neutrino oscillation parameters. Traditionally, particle energies are reconstructed by adding up or fitting to the hits on detector readout units, and event energy is reconstructed as a function of particle energies in the event [17].

In this additional experiment, the aim is to “reconstruct” the energy of the neutrino that produced a certain interaction by feeding a neural network two images of the detector as before, but instead of binary classification, the network will estimate the initial energy of neutrino in units of GeV.

For this task, there are two architectures that could be trialed – the same architecture as before (GoogLeNet), or an LSTM network inspired by figure 7. The LSTM belongs to the larger umbrella of RNNs, which stands for recurrent neural network. The LSTM, and the family of RNNs are networks that specializes on time series/sequential data, and has achieved impressive results in a selection of tasks, such as language modelling, speech-to-text transcription, machine translation and much more.

The basic idea of RNNs is to achieve some preservation of previous data in earlier neurons. Whilst in other neural networks, the output of previous layers is no longer used later in the network, the RNN preserves past data, by having a mechanism which “reinjects” old data into every following cell.

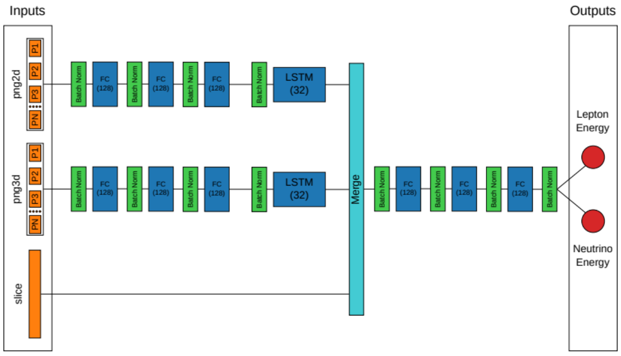

Figure 7: Energy reconstruction model [14]

The theory behind using LSTMs for this particular task is that the detector images can be viewed chronologically – the image itself describes the path/motion of particular particles within the detector. Therefore, knowing that the particles "enter" from the top edge of the image, it is feasible to feed the network an image row by row starting at the top edge of the image, and to process the rows of pixels as a video of sorts, that describe the movement of the particles.

For this task, we will not normalize the label data, as there is no suitable "maximum" value to divide the data by. Instead, the metric used by network to judge performance will be mean absolute error (MAE). Another suitable metric would be mean squared error (MSE), which penalizes large deviations more than small ones, and can be square rooted to get an estimate of the absolute deviation as well. In this case, MAE was used so the final result could be compared directly to absolute values to gauge the performance of the network in understandable terms.

Both the LSTM and CNN were trialed. The LSTM network trained faster, but fell short on MAE of 6.9, as opposed to the much slower CNN, which achieved MAE of 6.2.
Similar models were built and trained; the results are shown in table 3.

Task	Final metrics
Predicting Y	
Predicting neutrino E	Mae: 6.22, 0.6 std
Predicting lepton E	Mae: 0.19, 0.72 std

Table 3: Final metrics on energy recontruction models

Predicting neutrino flavor and interaction mode

Task	Final metrics
Predicting neutrino flavor	68.4% accuracy
Predicting interaction mode	46% accuracy

Table 4: final metrics on predicting neutrino flavour and interaction mode

The final result of these tasks was less successful, shown in table 4. Due to the nature of these classification tasks, which are more specific, with more classes to learn and identify, naturally, would perform worse using similar network architectures and similar size of data sets.

The prediction of neutrino flavor is directly comparable to the results of another CNN developed to combat a similar task see ref [13] achieved an accuracy just above 70%. That experiment had access to more resources, so comparatively speaking, the model to predict neutrino flavor was quite successful.

The interaction mode task requires the identification of 4 different classes. The baseline model would achieve an average of 25% accuracy, so the model isn't a huge improvement.

Predicting the number of protons and pions

One of the more challenging tasks for this experiment, this task requires a suitable dataset that contains the image data, as well as labelled data which corresponds to the number of protons and pions in the image.

There are several metrics that may be interesting to look at in this case. The metrics that are related to this problem are: the interaction mode (QE, DIS, RES and other), the number of prongs present in the final state, the parent metric, the final state metric, and the particle metric.

After conducting analysis on the metric, the parent and particle metrics were discarded, after it was found that the metrics were not correlating with other metrics in an understandable way. It is unlikely that the metrics were representing PDG particle codes, as not only were the number constrained to a maximum value of 391, there were an abundance of zeros, which is not a classified particle, and ones, which correspond to down quarks, which are not known to be the biproduct of QE or RES interactions. It is more likely that these metrics represented the number of parent and final state particles. Regardless, the metrics are still not very useful for this task.

Nevertheless, the final state, final state prong number and interaction mode, when used in tandem, provided a very interesting result. Observing equations 3 and 4, which describe two possible Quasi-Elastic interactions, it can be seen that in one case, one charged particle is produced (an antimuon) and in the other case, two charged particles are produced (a proton and a muon). After analyzing data for QE events, a pattern in the data was found. Many of the interactions produced only 1 prong – or one charged particle – and a minority of events produced 3, 5 and in rare cases, 13. In the cases with 5 prongs occur from electron neutrino interactions. The cases with 13 prongs will be ignored, as it cannot be determined what interaction has occurred. It can be speculated that equation 3 and 4 are representing the first two rows of table 3. Although only two charged particles are created in equation 4, an extra prong/track could arise from the decay of the muon.

Interaction mode	prongs	Final state
QE	1.0	v_μ, 1 track
QE	3.0	v_μ, 3+ tracks
QE	5.0	v_e, 2 tracks
QE	13.0	Unknown

Table 5: sample metadata from selected QE interactions

Similar deduction techniques were used on RES interactions, where the majority of events contained three prong events. Both equations 5 & 6 could correspond to these events, but equation 5 is unique as it produces a neutral pion, which usually causes "showers" in detector images. So, it is likely that resonant charged current events with 3 prongs and 1+ showers are producing 2 protons/pions, as according equation eq. 5. Equation 6, which only produces a positive pion, can be approximated to RES interaction with 3 prongs, but no showers. DIS and "other" interaction modes should be counted as producing no protons/pions.

Next, the data was balanced across 0, 1 and 2 proton/pion events. The labels were then one hot encoded, which is a method of converting each class into a sparse vector – all zeros, with only 1 one – corresponding to the correct label. This can then be passed to a network with 3 dense output neurons, and softmax activation function. The softmax function confines the sum of all the outputs from the network to equal 1, in other words, it gives the probability of each class.
The final accuracy for this task reached 43% accuracy.

Conclusion

This experiment has been insightful into the different machine learning methods available to complete certain tasks. There were trade-offs between networks, such as the faster but less precise LSTM network, and the slower but more precise GoogLeNet CNN. This realistic machine learning task shows the true side of machine learning; the unbalanced data sets, the lack of certain meta data such as tau neutrino interactions and uncleaned data. This task also illustrates the strengths of machine learning but also it provides insight into a future where machines can complete data analysis completely without human guidance.

The experiment has several points of improvement. If possible, use a machine with a lot of RAM and a fast GPU. There are also many advantages to running models on a Linux OS instead of windows. If more RAM and GPU is not available, it would be wise to implement a pipeline that trains a model using chunks of the full dataset at a time, only loading some data onto the RAM at a time, instead of loading the full data set. Implementing one of the above improvements would allow the project to gain access to a larger dataset, which is one of the best ways to prevent overfitting and improve model performance.

References

[1] John N. Bahcall and Raymond Davis,Solar Neutrinos: A Scientific Puzzle,Science,191,4224,264-267,1976, [online] https://www.science.org/journal/science available at: https://www.science.org/doi/abs/10.1126/science.191.4224.264

[2] Y. Fukuda et al. (Super-Kamiokande), Phys. Rev. Lett. 81, 1562 (1998), arXiv:hep-ex/9807003 [hep-ex].

[3] J. R. Jordan, Y. Kahn, G. Krnjaic, M. Moschella, and J. Spitz, Phys. Rev. Lett. 122, 081801 (2019), arXiv:1810.07185 [hep-ph].

[4] B. Armbruster et al. (KARMEN), Phys. Rev. D65, 112001 (2002), arXiv:hep-ex/0203021 [hep-ex].

[5] K. Abe et al. (Super-Kamiokande), Phys. Rev. D91, 052019 (2015), arXiv:1410.2008 [hep-ex].

[6] P. Adamson et al. (MINOS), Phys. Rev. Lett. 107, 011802 (2011), arXiv:1104.3922 [hep-ex].

[7] G. Cheng et al. (SciBooNE, MiniBooNE), Phys. Rev. D86, 052009 (2012), arXiv:1208.0322 [hep-ex].

[8] M. G. Aartsen et al. (IceCube), Phys. Rev. Lett. 117, 071801 (2016), arXiv:1605.01990 [hep-ex].

[9] M. A. Acero et al., (The NOvA Collaboration), "*An Improved Measurement of Neutrino Oscillation Parameters by the NOvA Experiment*.", 18 Aug 2021, [online] arXiv.org. Available at: https://arxiv.org/abs/2108.08219

[10] "How does NOVA work?", [online], available: https://novaexperiment.fnal.gov/how-does-nova-work/

[11] Grover, Deepika and Saraswat, Kapil and Shukla, Prashant and Singh, Venktesh, "Charged-current deep-inelastic scattering of muon neutrinos", Physical Review C, vol 98, Dec 2018, available at: https://arxiv.org/abs/1808.00287

[12] J. A. Formaggio and G. P. Zeller, Rev. Mod. Phys. 84, 1307 (2012)

[13] Aurisano, A. and Radovic, A. et al., Journal of Instrumentation, "A convolutional neural network neutrino event classifier", vol 11, pages p09001, Sep 2016, available at: https://arxiv.org/abs/1604.01444

[14] Psihas, F. and Niner, E. and Groh, M. and Murphy, R. and Aurisano, A. and Himmel, A. and Lang, K. and Messier, M. D. and Radovic, A. and Sousa, A., “Context-enriched identification of particles with a convolutional network for neutrino events”, Physical review D, Oct 2019, available at: https://arxiv.org/abs/1906.00713

[15] Chollet F., “Introduction to convnets”, Introduction to Deep Learning with Python, Manning Pub. Co., 2018, chapter 5, pgs.117 -177

[16] http://www.hep.ucl.ac.uk/undergrad/0056/other/projects/nova/

[17] Baldi, Pierre and Bian, Jianming and Hertel, Lars and Li, Lingge, “Improved energy reconstruction in NOvA with regression convolutional neural networks”, vol 99, Physical Review Dm Jan 2019, available at: https://arxiv.org/abs/1811.04557