Deep Learning in LArTPCs

T. Wongjirad (MIT)

Workshop on Software for TPCs for Nuclear Physics Experiments

Aug. 10th, 2016
On MicroBooNE, we’ve started an effort to try and apply deep learning techniques to the problem of event reconstruction/selection on LArTPC images.

What is deep learning and how are we using it?

Proof of principle demonstrations

Where we are heading and the challenges we face
Example neutrino event from the beam

Lots of detail on location and amount of charge created in detector
Detail allows us to parse, or reconstruct, these images

Tracks tell us about the neutrino
Detail allows us to parse, or reconstruct, these images.

Tracks tell us about the neutrino.
CHALLENGES

- Full event view
- Must pick out neutrino from cosmic muon backgrounds
- Many images will not have a neutrino
- Too many images to sort through by hand
- Need to develop computer algorithms to find neutrinos
Deep learning is a subfield of machine learning where computers are taught to learn high level abstractions through a set of algorithms arranged in a graph with multiple processing layers

- e.g., a feed forward neural network with many hidden layers

We have been focusing on a particular method, convolutional neural networks (CNNs), that has grown out of the computer vision industry
Applying convolutional neural nets (CNN)

Very adept at image analysis

Primary advantages: scalable and generalizable technique

Successfully applied to many different types of problems
CNN APPROACH

- CNNs produce representations of high-level objects based on a combination of low-level features

- Consider the task of recognizing faces
  - Begin with image pixels (layer 1)
  - Start by applying convolutions of simple patterns (layer 2)
  - Find groups of patterns by applying convolution on feature maps (layer 3)
  - Repeat

- Eventually patterns of patterns can be identified as faces (layer 4)
Can think of CNNs as an extension of the common neural network

The basic unit of a neural net is the **perceptron** (loosely based on a real neuron)

Takes in a vector of inputs \((x)\). Commonly inputs are summed with weights \((w)\) and offset \((b)\) (but in principle can be any function, e.g. sigmoid, tanh)
Crude interpretation

Inputs are a list of feature we’ve extracted, e.g. momentum, dE/dx, etc.

\[ f(x) = \begin{cases} 
1 & \text{if } w \cdot x + b > 0 \\
0 & \text{otherwise}
\end{cases} \]
Form layers. Neuron in one layer uses all the outputs of previous layer as input (fully connected)

Input set of parameters for a single class

Class score, inference

Can be powerful classifier, e.g. PID, but scales poorly
Define network architecture to look for local occurrences of translation-invariant features

**Filter**

-1 0 -2 0
2 1 -1 -1
0 3 -3 -1
-1 0 1 0
1 0 2 1
1 1 1 -1
0 1 0 -1
-1 -1 -1 1
0 0 -2 0
1 -1 0 1
0 1 -3 -1
-1 1 -2 1

**Neuron output**

**Activation function**

**Dot product, add bias**

**Image**

in conv. layer, apply pattern filter at image location
CONVOLUTIONAL NEURAL NETWORK

Define network architecture to look for local occurrences of translation-invariant features

apply operation with same filter over image — looks for translation-invariant features
This core operation in a CNN is the convolutional filter – identifies the location of patterns in an image.

Here regions of light and dark are where the pattern (or its inverse) matched well within the image.
The power of CNNs and other machine learning tasks come from the fact that they learn the filters themselves!

- There is no standard feature bank
Introduction to CNNs

N Filters

Feature Maps

apply many filters

many weights!
use many layers to assemble patterns into complex image features
Typically this is a supervised learning technique. During training, provide an input image and the answer the network is suppose to output. Adjust parameters of the network until network associates input and correct output.
MicroBooNE located at Fermi National Accelerator Lab

- Sits 450 m from the start of the Booster Neutrino Beam – produces mostly muon neutrinos
MicroBooNE employs a liquid argon TPC.
Three Wire planes provide 3 views of the same interaction.
MicroBooNE, a LArTPC detector filled with 170 tons

Looking for short-baseline numu to nue oscillations

Measure interaction between neutrino and argon nucleus for future LArTPCs
There exists several CNN algorithms that perform tasks directly applicable to our problem:

- Image classification
- Object detection
- Pixel labeling

Neutrino Interaction Reconstruction

$\nu_\mu + n \rightarrow \mu + p$
RESULT: SINGLE PARTICLE CLASSIFICATION

- trained network to classify single particle MC images

Tech. issue: to keep high resolution, cropped event image using voxels with particle vertex
RESULTS: SINGLE PARTICLE CLASSIFICATION

- trained network to classify single particle MC images
- uniformly distributed in active region, generated isotropically in energy from 0.1 to 1 GeV

Figure 9: High resolution five class single particle results. Both models struggle with electron and gamma separation as well as distinguishing pions from muons. The GoogLeNet performs better than the AlexNet in all cases except for a pion.
 RESULTS: SINGLE PARTICLE CLASSIFICATION

- trained network to classify single particle MC images
- uniformly distributed in active region, generated isotropically in energy from 0.1 to 1 GeV

![Box figure 1](image1.png)

### 3.6 Particle Detection Performance

To assess the Faster-RCNN detection performance on the single particle sample, we let the Faster-RCNN network infer a set of bounding boxes per class for each high resolution event image containing one particle. Typical detection examples can be seen in figure 13. As it is done for all studies in this section, this study used the same training and validation sample described in sec. 3.1.

To quantify the Faster-RCNN detection performance on the single particle sample we infer a set of bounding boxes per class for each high resolution single particle image. To quantify this performance we compute the intersection over the union of the ground truth bounding box and the predicted box with the highest network score. This is the standard performance metric used by object detection networks to compare with one another. Intersection over union (IoU) is defined for a pair of boxes in the following way:

\[
\text{IoU} = \frac{A_1 \cap A_2}{A_1 + A_2 - A_1 \cap A_2}
\]

This quantity is unity when the predicted box and the ground truth box overlap perfectly. In other words, the predicted network box is of the same pixel dimensions. In figure 14 we plot the IoU for the different five-particle classes. We separate the detected sample into the five different particle types and break down each sample by their top classification score. The true class label is in the title of the plot, and the legend lists 1) the five particle types that were detected for the sample, 2) the number of detections in the histogram for that class, and 3) the class-wise fraction of all detections. For this plot we make a cut on the network score of 0.5. We observe good detection accuracy and ground truth bounding box overlap on the muon and proton classes. If we consider classification only, muons and...
RESULT: NEUTRINO EVENT FILTER

- Networks trained to select events with neutrino interactions
- Note: cosmic off-beam data + MC neutrino overlaid
Trained a network to place a bounding box around a neutrino interaction within a whole event view

Note: cosmic off-beam data + MC neutrino overlaid

RESULT: NEUTRINO DETECTION
Pixel-wise Particle Classification (PID + clustering)

Currently exploring deconvolution techniques

**AlexNet + DeconvNet**
Cherry-picked $\pi^-$ example (97% pixel labeling accuracy)
Fine tuned 5-Particle classification net from the paper
We have started to look into data and MC disagreement, and its impact on network behavior – helping to improve MC is one of our top priorities.

- Isolating a sample of events from the off-beam data: protons, through-going muons, Michel electrons.

- Train networks to do different tasks on the MC and see if it works on data sample and vice versa.

- Helps us gauge the effect of MC mismodeling on networks.

We’re also looking at strategies to mitigate sources of MC disagreement from detector modeling – have done studies on something called stability training.
Besides our work on MicroBooNE, others are also making progress on applying CNNs to LArTPCs

- e.g. on Dune (Fabin, Stefan, Sulej, Plonski): [https://indico.fnal.gov/conferenceDisplay.py?confId=12278](https://indico.fnal.gov/conferenceDisplay.py?confId=12278)

Successful application on other neutrino experiments as well (different detector technologies)

Shown that CNNs can identify features in LArTPC images that can allow us to extract the information we’re interested in.

More details about our results can be found in public note: http://www-microboone.fnal.gov/publications/publicnotes/MICROBOONE-NOTE-1019-PUB.pdf

Now moving towards pixel labeling (semantic segmentation) and developing side-band samples and methods to understand systematic uncertainties.
BACKUP SLIDES
Abstract

This is a reply to Joseph's comments to our paper DocDB 5905.

Question

Abstract: Have any studies been performed to see if the classification works as well, or better, when run with MC cosmics overlaid on MC neutrinos?

Answer

We have. Gabriel did this analysis. The reason that this is not already in the paper is that it is lacking a noise model and dead wires in both the cosmic or neutrino sample using slightly outdated version of LArSoft. It is also using a slightly different, though similar network than the one used in demonstration 3. The performance was quite good. In figure 1, we plot the fraction of cosmics remaining (green) and the efficiency of accepting neutrino events (blue) vs. neutrino classification score.

The performance seemed too optimistic, and we considered using real data cosmic background is more realistic to assess the applicability of this technique. Redoing this study with proper configuration and same version of LArSoft would be a significant work addition.

Comment

Figure 1: How about Fig 10 from here: http://microboone-docdb.fnal.gov:8080/cgi-bin/RetrieveFile?docid=5752&filename=main.pdf&version=3
- CNN technique also can complement/assist other reconstruction approaches

**Bottom up**

1) hits

2) clusters

3) tracks

Analogous to the “Standard” computer vision approach — we start by identifying image features to aggregate.
- **CNN technique also can complement/assist other reconstruction approaches**

Because CNN can tell us what a local region is or has, we can use this “top” level info to guide the recon. of low level objects.

1) particle/interaction

2) tracks

3) hits
### 5 Particle Classification

#### Table 2: Five particle classification rate

<table>
<thead>
<tr>
<th>Image, Network</th>
<th>$e^-$ [%]</th>
<th>$\gamma$ [%]</th>
<th>$\mu^-$ [%]</th>
<th>$\pi^-$ [%]</th>
<th>Proton [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>HiRes, AlexNet</td>
<td>73.6 ± 0.8</td>
<td>81.3 ± 0.8</td>
<td>84.8 ± 0.5</td>
<td>73.1 ± 0.8</td>
<td>87.2 ± 0.5</td>
</tr>
<tr>
<td>LoRes, AlexNet</td>
<td>64.1 ± 0.9</td>
<td>77.3 ± 0.8</td>
<td>75.2 ± 0.6</td>
<td>74.2 ± 0.8</td>
<td>85.8 ± 0.5</td>
</tr>
<tr>
<td>HiRes, GoogLeNet</td>
<td>77.8 ± 0.8</td>
<td>83.4 ± 0.7</td>
<td>89.7 ± 0.4</td>
<td>71.0 ± 0.8</td>
<td>91.2 ± 0.4</td>
</tr>
<tr>
<td>LoRes, GoogLeNet</td>
<td>74.0 ± 0.8</td>
<td>74.0 ± 0.9</td>
<td>84.1 ± 0.5</td>
<td>75.2 ± 0.8</td>
<td>84.6 ± 0.6</td>
</tr>
</tbody>
</table>

#### Table 3: Five particle classification negatives

<table>
<thead>
<tr>
<th>Image, Network</th>
<th>$e^-$ [%]</th>
<th>$\gamma$ [%]</th>
<th>$\mu^-$ [%]</th>
<th>$\pi^-$ [%]</th>
<th>Proton [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>HiRes, AlexNet</td>
<td>23.0</td>
<td>16.2</td>
<td>8.0</td>
<td>19.8</td>
<td>7.0</td>
</tr>
<tr>
<td>LoRes, AlexNet</td>
<td>29.3</td>
<td>17.6</td>
<td>11.7</td>
<td>16.5</td>
<td>7.9</td>
</tr>
<tr>
<td>HiRes, GoogLeNet</td>
<td>19.9</td>
<td>15.0</td>
<td>5.4</td>
<td>22.6</td>
<td>4.6</td>
</tr>
<tr>
<td>LoRes, GoogLeNet</td>
<td>21.0</td>
<td>21.3</td>
<td>9.4</td>
<td>19.3</td>
<td>9.1</td>
</tr>
</tbody>
</table>
Figure 19: Neutrino detection box score distribution for neutrinos (blue) and cosmics (red), normalized by area. It shows there is some discrimination only by just detection network. Box is the network prediction for the region containing the neutrino and the network score is labeled in white text above. Figures 20 and 21 are showing the results where the CNN successfully located a neutrino in an image with a high score (> 0.9). Figure 23 shows two types of mistakes: 1) finding a high score (> 0.9) bounding box in a wrong location in a neutrino event, and 2) also in a cosmic background event where there is no neutrino. In either case, the bounding box is containing an interaction topology that could be mistaken as a neutrino event. Thus, these are not boxes drawn randomly. Finally, Figure 22 shows examples of how the CNN is drawing many boxes in a cosmic background event. Boxes shown in this figure are those with neutrino scores less than 0.1.

Figure 19 shows the distribution of neutrino bounding box scores predicted by Faster-RCNN per event for both neutrino+cosmic and cosmic-only images. We can see that the network is successfully finding a more neutrino-like bounding box in neutrino events than cosmic background events. Moreover, because the network is not trained to specifically discriminate cosmic events, it finds a bounding box with a moderate score value among cosmics. This is a good sign, as it indicates that the network is not simply keying on a mere difference of data and simulation.
To quantify the Faster-RCNN detection performance on the single particle sample, we let the model perform a prediction on each high resolution single particle image. We infer a set of bounding boxes per class for each image. This results in a prediction score distribution for each class. We then calculate the intersection over the union (IoU) of the predicted bounding box with the ground truth bounding box for each class. The IoU is defined as the ratio of the intersection area to the union area between two bounding boxes.

\[
\text{IoU} = \frac{\text{intersection area}}{\text{union area}}
\]

This quantity is unity when the predicted box and the ground truth box overlap perfectly. The true class label is in the title of the plot, and the legend lists the five particle types that were detected for the sample, along with their respective IoU scores. The right plot in figure 12 shows an electron classification separation study for 1:1 mixture (6800 events total), with the goal of achieving 95% efficiency and purity in a 1:1 mixture of 5200 events taken from the validation set. The right plot in figure 12 shows a similar separation study for proton classification where score is re-normalized for 5 particle classification. Right:

- Electron: blue
- Gamma: red

As described in Sec.3.4. Orange and cyan data points are AlexNet and GoogLeNet respectively, trained with a sample of 6800 events taken from the validation set that only contain electron and gamma events. Blue and red data points are AlexNet and GoogLeNet respectively trained with a sample of 5200 events taken from the validation set that only contain muon and proton events. Note that our high resolution image has a factor of two in wire and six in time compression compared to the two types are essentially indistinguishable in the range of scores from 0.3 to 0.6. We show a similar separation study for muon identification efficiency vs. purity and proton classification where score is re-normalized for 5 particle classification. Right:

- Muon: blue
- Pion: red
RESULTS: SINGLE PARTICLE CLASSIFICATION

- trained network to classify single particle MC images
- uniformly distributed in active region, generated isotropically in energy from 0.1 to 1 GeV

GoogLeNet 5 Particle Network

Electron selection EP curve for the 5-particle classification
GoogLeNet provides better separation!

Muon/pion separation in the network score

MicroBooNE Simulation In Progress
trained network to classify single particle MC images

uniformly distributed in active region, generated isotropically in energy from 0.1 to 1 GeV

- Figure 11: Left: \(\mu\) selection EP curve for 1:1 \(\mu\) and \(\pi\) mixture (6800 events total). Blue and red data points are AlexNet and GoogLeNet respectively trained with a sample set that only contain \(\pi\) and \(\mu\) as described in Sec.3.4. Orange and cyan data points are from GoogLeNet and AlexNet respectively, trained for 5 particle classification. Right: \(\mu\) score distribution from GoogLeNet trained for 5 particle classification where score is re-normalized for \(\mu/\pi\) separation purpose.

- It is interesting to note that there is a small, but distinct, set of \(\pi\) events that follow the \(\mu\) distribution. This makes sense since the \(\pi\) has a similar mass to the \(\mu\) and decays into \(\mu\). As a result, some \(\pi\) can look very similar to a \(\mu\). A typical way to distinguish \(\pi\) is to look for a nuclear scattering, which occurs more often for \(\pi\) than for a \(\mu\). There can also be "kink" in the track at a point where the \(\pi\) decays-in-flight into a \(\mu\), although this is generally quite small. When neither is observable, the \(\pi\) looks like a \(\mu\), however when there is a kink or visible nuclear interaction involved, \(\pi\) is distinct. This can be seen by a very sharp peak for \(\pi\) in the right figure. The same reason explains why there is no \(\mu\) above 95% (with the statistics of this sample) because \(\mu\) can never be completely distinguished from those small fraction of \(\pi\) that do not carry any kink nor visible nuclear interaction.

- Figure 12: Muon Identification Efficiency vs. Purity (left) and GoogLeNet 5 Particle Network (right). The left plot shows electron selection efficiency and purity in a 1:1 mixture of 5200 events taken from the validation set. The outer-most point achieves selection efficiency of 94.3% with purity of 71.9%, although one might want to ask for a better separation with less efficiency depending on the goals of an analysis.

- The right plot shows an electron classification score distribution for both \(e\) and \(\mu\) as we did for \(\mu/\pi\). This time, however, we only show the result using five-particle classification, since we saw those networks seem to perform better, presumably for similar reasons. The left plot in figure 12 shows electron classification score distribution from GoogLeNet trained for 5 particle classification where score is re-normalized for \(\mu/\pi\) separation purpose.