Summary of Machine Learning Applications for the COHERENT Collaboration

Jacob Daughnetee
University of Tennessee

NPML
July 14th, 2020
**Coherent Elastic Neutrino-Nucleus Scattering**

- Clean prediction from the Standard Model – D. Freedman 1974
- Cross-section increases with energy as long as coherence condition is satisfied \((q \leq R^{-1})\)
- Largest of all SM neutrino cross-sections at 1-100 MeV scale
- NC mediated: all flavors of neutrino can scatter via CEνNS
- Sensitive Standard Model Probe
- Applications: Dark Matter Experiments, Supernovae, Monitoring

\[
\sigma_{tot} = \frac{G_F^2 E_v^2}{4\pi} \left[ Z \left( 1 - 4\sin^2\theta_W \right) - N \right]^2 F^2(Q^2)
\]

*Cross section may be high, but the signal is in the form of a low-energy nuclear recoil!*
CEνNS Efforts

- Gaseous Spherical Proportional Counters
- Ge and Zn Bolometers
- HPGe
- Super CDMS-style Ge
- Al and Ca Bolometers
- Si CCD
- LAr TPC
- LAr
- LXe TPC
- Accelerator
- HPGe
The COHERENT Collaboration
COHERENT Program

Multi-target program to measure CEνNS cross section over wide range of $N$

Staged approach: *Observation* -> *Precision*
**COHERENT at the SNS**

- **Spallation Neutron Source (SNS):**
  - 1.4 MW pulsed 1 GeV proton beam on Hg target
  - Pulsed at 60 Hz with 400 ns FWHM
  - Pion decay-at-rest (DAR) neutrino source.

- Detectors located between 20-30 m from target in neutron quiet basement corridor (Neutrino Alley).

- Multiple detectors currently operating measuring either CEvNS or backgrounds.

*Staged approach: Observation -> Precision*
Identifying Charged-Current Events in NaIvE

- Prototype 185 kg NaI[Tl] detector array currently deployed at the SNS.
- CNN is used to distinguish cc interactions on $^{127}$I from backgrounds such as cosmic muons.
- Ongoing work, see next talk by Peibo An for more information!
Neutrino Cubes

- Detectors searching for neutrino-induced neutrons; a potential CEvNS background.
- Excited nuclei in target material (Fe, Pb) can emit neutrons which can produce nuclear recoil events in embedded detectors.
- NIN process yet to be observed; relevant to SNe nucleosynthesis and as SNe detection channel (HALO).

\[
\begin{align*}
\text{(CC)} & \quad \nu_e + ^{208}\text{Pb} \to ^{208}\text{Bi}^* + e^- \\
& \quad ^{208-\gamma}\text{Bi} + x\gamma + yn \\
\text{(NC)} & \quad \nu_x + ^{208}\text{Pb} \to ^{208}\text{Pb}^* + \nu'_x \\
& \quad ^{208-\gamma}\text{Pb} + x\gamma + yn
\end{align*}
\]
Neutrino Cube Event Discrimination

$^{252}$Cf Calibration Data

- Excited states of scintillator molecule emit light at two different timescales.
- Nuclear recoils and gamma/electron events populate these excited states in different ratios – allows for pulse shape discrimination (PSD).
- Tail to full integral ratio is a common discrimination metric for organic scintillators.
- Performance degrades and populations blend together as photon statistics become more discrete.
- Machine learning methods can use more event information!
Collecting a Training Dataset

- Time-tagged source allows for the isolation of NR events from ER.
- Less stringent standard PSD cut included to reduce contamination from gammas contemporaneous with neutrons.
- Calibration runs performed for all eight scintillator cells currently in operation.
BDT Input Variables

- Fraction of total integral accumulated at different time intervals.

- Information from each level of Haar Wavelet transformation also used for a total of 15 variables.

\[
\frac{\int_0^x WF}{\int_0^{400} WF}
\]

Where \( x \) is \{20, 26, 38, 44, 50, 140, 220, 340\}
BDT Performance / Future Steps

- Use of BDT in place of Std PSD cut gave marginal improvement for some cells; very little in others.

- NIN recoil spectrum increases rapidly at lower energies; so any increase in acceptance is good for sensitivity.

- Early tests of CNNs on NUBE waveforms look promising:
CENNS-10

• Loaned from J. Yoo et al from Fermilab.

• Single-phase liquid Ar scintillation detector located 28 m from SNS target (~2 x 10⁷ ν / s )

• Engineering Run: Dec 2016 -> May 2017
  • 80 keVnr threshold
  • No Pb shielding
  • Analysis Results -> Phys. Rev. D100 (2019) no. 11, 115020

• First Production Run: July 2017 -> December 2018
  • Dramatically improved light yield results in lower threshold (20 keVnr)
  • 2x 8” Hamamatsu PMTs with 18% eff @ 400 nm
  • Tetraphenyl butadiene (TPB) wavelength shifter coating Teflon walls and PMT glass.
  • 24 kg fiducial volume.

Liquid Ar Scintillation

- Bright scintillator (40 photons/keVee)
- Well-known nuclear quenching factor
- Emission timescales:
  - 6 ns (singlet)
  - 1.6 μs (triplet)
- Electron recoils (ER) and nuclear recoils (NR) yield different ratio in exited state populations -> Pulse Shape Discrimination (PSD)
- Scintillation light wavelength: 128 nm (requires wavelength shifting)
- Benefit of using liquid noble gas – Scalability
- LAr detectors used for neutrino beams, dark matter, coherent elastic neutrino-nucleus scattering (CEvNS).
Event Discrimination

• Use of PoT signal from SNS greatly reduces steady-state backgrounds, BUT 1 Hz/kg of $^{39}$Ar events still a large background:

<table>
<thead>
<tr>
<th>Data Events</th>
<th>3752</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fit CEvNS</td>
<td>159 ± 43 (stat.) ± 14 (syst.)</td>
</tr>
<tr>
<td>Fit Beam Related Neutrons</td>
<td>553 ± 34</td>
</tr>
<tr>
<td>Fit Beam Unrelated Background</td>
<td>3131 ± 23</td>
</tr>
<tr>
<td>Fit Late Beam Related Neutrons</td>
<td>10 ± 11</td>
</tr>
<tr>
<td>$2\Delta(-\ln L)$</td>
<td>15.0</td>
</tr>
<tr>
<td>Null Rejection Significance</td>
<td>3.5$\sigma$ (stat. + syst.)</td>
</tr>
</tbody>
</table>

Analysis A Fit Results

• Standard PSD technique for Ar scintillation is ratio of integral in first 90 ns to total integral (F90)

• Potential issues near threshold:
  • Discrete photon pulses widen dispersion of band.
  • Value of F90 much more susceptible to fluctuations.

• Can we use more (all) waveform information?
Applying a 2D CNN

• Convolutional neural networks typically work on 2-d images; but there is some support for 1d neural network in pyKeras.

• Recurrence plots are often used to visualize periodic features in N-dimensional phase spaces.

\[ R(i,j) = ||\tilde{x}(i) - \tilde{x}(j)|| \]

• Distance is limited to some number of gradations which in the following instances is set to 128.

• Due to large size of peak w.r.t. other samples, square root of distance was used instead.
Training the Network

• Time-tagged DT data makes for an excellent source of NR waveforms with little accidentally contamination form ER band.

• Selection criteria for training samples:
  
  **NR**
  - $20 < \text{NPE} < 600$
  - $0.4 < F90 < 0.81$
  - $-0.6 < \text{Tag Time} < 0.1$

  **ER**
  - $40 < \text{NPE} < 600$
  - $0.15 < F90 < 0.4$
  - $^{57}\text{Co}$ calibration data

• Approximately $1e5$ events for both event samples.
Training the Network

- Event waveforms are truncated to 1024 samples, and then down sampled twice to yield an array of 256 samples.

- Recurrence map then generated which has shape (256,256), which is fed to the CNN.

```python
initializer = initializers.glorot_normal()

model = Sequential()
model.add(Conv2D(64, kernel_size=(3,3), activation='relu', input_shape=(256,256,1)))
model.add(MaxPooling2D(pool_size=(2,2), strides=(2,2)))
model.add(Conv2D(32, kernel_size=(3,3), activation='relu'))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.4))
model.add(Dense(2, activation='sigmoid'))
model.compile(loss="sparse_categorical_crossentropy", optimizer='adam', metrics=['accuracy'])
```
Training the Network

- Few epochs required to train on this dataset.
- Can use model output to classify new data.
- Classification results are either binary class decision (Sig,Bkg) **OR**
  - Score for each category (0->1) where summation of scores equals unity.
- Variable used for cuts: Signal Score – Bkg Score (-1->1 with > 0 being signal classification).
Evaluation on Calibration Data

- Sig
- Bkg

AmBe
Analysis Dataset (Beam OFF)
Internally Triggered Bkg

AmBe Event Classification (Sig Cut 0.05)
Analysis Data Event Classification (Sig Cut 0.05)
Bkg Event Classification (Sig Cut 0.05)

Signal Score – Bkg Score > 0.05
Evaluation on Separate DT Dataset

Bkg Classified

Sig Classified
1D CNNs are ideal for time-series data; should be a more appropriate tool for waveform classification.

- No need to down-sample to speed up computation for 1D.

- Network architecture similar to that used by Griffiths et al. in arXiv:1807.06853

- Incorporation of large fractional dropout layer found to be necessary to prevent overtraining.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Channels</th>
<th>Filter Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conv</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>MaxPool</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>Conv</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>MaxPool</td>
<td>–</td>
<td>2</td>
</tr>
<tr>
<td>Flatten</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Dense</td>
<td>64</td>
<td>–</td>
</tr>
<tr>
<td>Dropout</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Dense</td>
<td>2</td>
<td>–</td>
</tr>
</tbody>
</table>
Evaluation on Separate DT Dataset

No Cuts

Sig Classified
**CNN vs Standard PSD Cut**

- Significant increase in number of lower energy events preserved; Preliminary NR efficiency suggests better NR acceptance.

- Check against background data to see if improved signal retention comes with greater background acceptance: **ER events are leaking in.**

- Two Issues:
  - ER Training dataset is skewed to higher energies; need more low energy ER events for training!
  - ER contamination in NR training dataset may be contributing to ER acceptance at low energy.

- Proper estimation of contamination in training sample in progress.

- To Do: validation of performance in simulation.

- Future DT or $^{252}$Cf calibrations with conditions optimized for machine learning training.
Future Applications

- Currently applying 1D CNN to combined top and bottom PMT waveforms; possibly more to be gained by using both waveforms individually.

- 1D CNN can be generalized for future LAr detector (CENNS-750) which will feature far more detection channels.

- Machine learning for more than just event discrimination; use in event reconstruction may prove useful (corrections for detector non-uniformity, handling of saturated channels, etc).

![Ton-Scale NaI[Tl]](image)

![SiPM assembly](image)
Summary and Outlook

• Machine learning currently limited to event discrimination uses for now, but plans in place to explore event reconstruction (particularly for successor detectors).

• Standard PSD methods in scintillation detectors begin to degrade with low photo-statistics.

• Signal spectrum for CEvNS (and other NR signals) is steeply rising at low energy; harsh charge integration cuts eliminate potential signal.

• CNN trained using time-tagged DT data is able to distinguish events at low energy without strict cut in F90-space.

• Low energy performance affected by contamination of signal training dataset and lack of problematic ER background events in bkg training dataset.

• Similar networks can be used in other COHERENT scintillation detectors.

• Results are still preliminary, but initial results are promising!
Auxiliary Slides
Calibrations

• Calibrations performed using multiple gamma sources ($^{57}$Co, $^{241}$Am, $^{83m}$Kr).

• Observed light yield: $4.6 \pm 0.4$ p.e./keVee

• 9.5% resolution at 41.5 keVee

• Linearity of detector response over energy range of interest.

• Global fit to LAr nuclear quenching data to provide keVnr->keVee conversion.

Neutron Calibrations

• AmBe – Used to measure NR response in detector and model CEvNS signal.

• DT Generator – Used to confirm veracity of external neutron simulations.
1D CNN Architecture

```python
model = Sequential()
model.add(Conv1D(filters=8, kernel_size=10, activation='relu', input_shape=(1024,1)))
model.add(MaxPooling1D(pool_size=2))
model.add(Conv1D(filters=16, kernel_size=10, activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(Flatten())
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.8))
model.add(Dense(2, activation='softmax'))
model.compile(loss="sparse_categorical_crossentropy", optimizer='adam', metrics=['accuracy'])
```