Work Presentation

Machine Learning Active-Target Time Projection Chamber (ATTPC)

Rujuta Karanjikar

A59573979

Summer 2020

1 Introduction

This Summer, I worked with Dr.Yassid Ayyad, nuclear physicist at the Facilities of Rare Isotope Facility at Michigan State University. I worked on the computational side of the project - Active Target Time projection Chamber. It was quite a learning experience as along with the research, I participated in a two week virtual European Center for theoretical studies' TALENT program 2020.[1]

In modern science, probability and statistics play a key role due to the collection of large amounts of data. This deluge of data calls for automated methods of data analysis, which is exactly what machine learning provides. For the same purpose, the Nuclear TALENT course was initiated virtually from 22nd June to 3rd July. Central theme of the course was to provide better conceptual study of Machine learning and data analysis and its applications in nuclear physics experiments. Beta decay experiment and ATTPC experiments were discussed as the main nuclear experiments in this course.

Students were expected to have a deeper understanding of the following:

- Statistical data analysis, theory and tools to handle large data sets.
- A solid understanding of central machine learning algorithms for supervised and unsupervised learning, involving linear and logistic regression, support vector machines, decision trees and random forests, neural networks and deep learning (convolutional neural networks, recursive neural networks, etc.)

- Be able to write codes for linear regression, logistic regression and use modern libraries like Tensorflow, Pytorch, Scikit-Learn in order to analyze data from nuclear physics experiments and perform theoretical calculations
- A deeper understanding of the statistical properties of the various methods, from the bias-variance tradeoff to resampling techniques. [2]

Nuclear study is vital for our understanding of nuclear science and its many fundamental applications - national security, medicine, astrophysics, nuclear energy, etc. The National Superconducting Cyclotron Laboratory (NSCL) at Michigan State University produces beams of rare isotopes for national and international researchers to study nuclei at the limits of existence. My line manager at FRIB, Dr.Yassid Ayyad, works in the superconducting laboratory for multiple ongoing experiments. One of them being ATTPC using different particles and to find out various different reactions due to nuclear decay and scattering. The lab's Active-Target Time Projection Chamber (AT-TPC) is a gas-filled detector that acts as both the detector and target for high-efficiency detection of low-intensity, exotic nuclear reactions. Because the gas target also acts as the detector, the AT-TPC is highly efficient, providing nearly 4π angular coverage. Reactions can be measured over a wide range of energies as the beam loses energy in the gas [3]. Just like the ATTPC proton events in the Ar - 40 data [4], this project is using the Mg-22 database for downsampling and training this model.

2 Problem Statement

Classifying an event from the 3D plots of collected data/recent data sets from the ATTPC project. This would be a definitive filter if successful results are seen after running the model, impacting the overall effectiveness of the ATTPC Project.

3 Methods

Analysis: After analysing the problem statement and understanding the Project from the 3 research papers, created 3D plots using MATLAB and classified them using intensity and tried to spot the patterns and events from the plots.

X-Pos	Y-Pos	Z-pos	Time	Amplitude
-73.6235	62.3091	414.4	134	306.15
-76.0777	60.9599	420.8	138	354.15
-78.5318	62.3091	427.2	142	158
-80.9859	60.9599	432	145	286.6
269.953	14.3382	433.6	146	472.2
211.054	31.3408	433.6	146	569.5
-83.44	56.7093	446.4	154	219.55
-90.8024	49.5571	473.6	171	214.05
-93.2565	48.208	480	175	135.4

Figure of the sample data

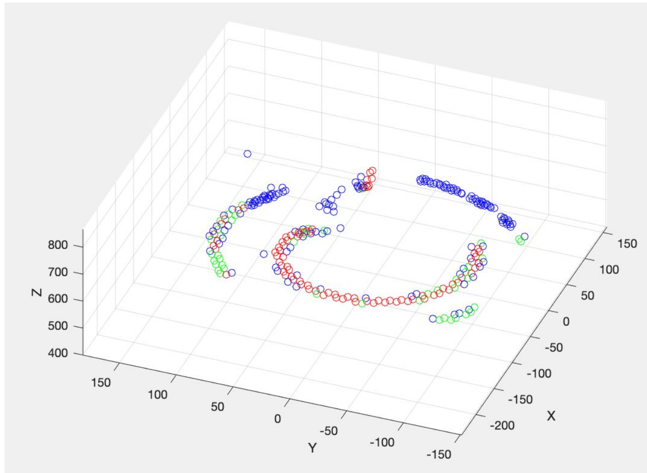

Figure for MATLAB 3D Plot of the data

On researching, exploration and analysing object detection to detect spiral forms in 3D by detecting curves in 2D plots in scikit[5] and rotating the plot appropriately, pre-existing Machine Learning models and Ryan Strauss Git (https://github.com/ATTPC/event-classification/tree/master/data-processing), there are several methods that can be used in machine learning for this classification problem of supervised learning.

Since this data is in a complex HDF5 format, downscaling and pre-processing the data becomes essential. On using the train.py code from the CNN folder after preprocessing the data from the data.py code from the utils folder, pytpc framework caused some errors.

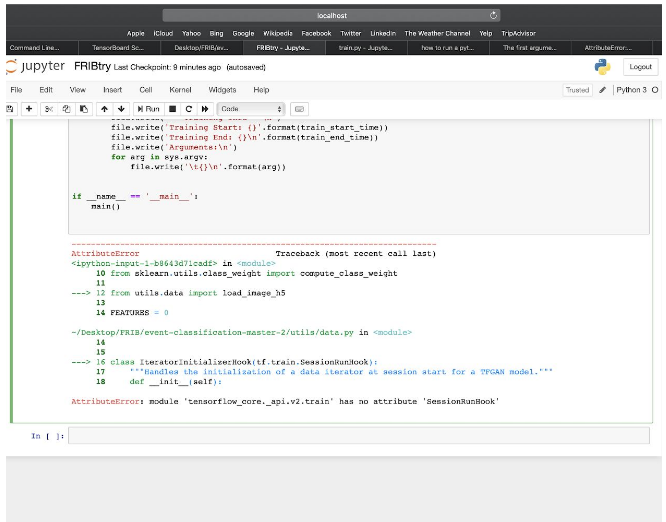

Figure showing an attempted code error

Down sampling and Preprocessing: Using the downsampling code in the Git file, I wa able to downsample the given Mg 22 data into 128*128 HDF5 file. [6]

Figure to plot the event data

Training the model: Model was successfully trained with 100 epochs and learning rate as 0.00001 only to find the min loss to be 0.6-0.7 and accuracy to be 0.53.

```
688
Epoch 98/100
35/35 [==============================] - 176s 5s/step - loss: 0.7016 - accuracy: 0.5545 - val_loss: 0.7131 - val_accuracy: 0.4
771
Epoch 99/100
35/35 [==============================] - 175s 5s/step - loss: 0.6947 - accuracy: 0.5580 - val_loss: 0.7130 - val_accuracy: 0.4
812
Epoch 100/100
35/35 [==============================] - 175s 5s/step - loss: 0.7036 - accuracy: 0.5491 - val_loss: 0.7132 - val_accuracy: 0.4
708

In [18]: result = CNN_model.evaluate(test_data, test_label, verbose=2)

13/13 - 13s - loss: 0.6986 - accuracy: 0.5263

In [19]: predictions = np.argmax(CNN_model.predict(test_data[:]),axis=1)

In [20]: print(classification_report(test_label, predictions))

              precision    recall  f1-score   support

           1       0.52      0.81      0.63       200
           2       0.56      0.24      0.34       199

    accuracy                           0.53       399
   macro avg       0.54      0.53      0.48       399
weighted avg       0.54      0.53      0.48       399


In [21]: plt.plot(CNN_history.history['loss'])
         plt.plot(CNN_history.history['val_loss'])
         plt.title('Model Loss')
```

Figure to show the classification report (Improvisation required)

4 Results

I have tried using a Sequential model in this trial. Many more CNN types can be used to train this model. My training can be found in the jupyter notebook named 'ATTPC_Classify_updated' in https://github.com/Rujuta219/Ru219 . Alternative file, ‘ATTPC_Classify’ contains an alternative method using CNN that was used to classify Ar 40 dataset. Shifting the axes in the 3D data also makes a difference in the results.

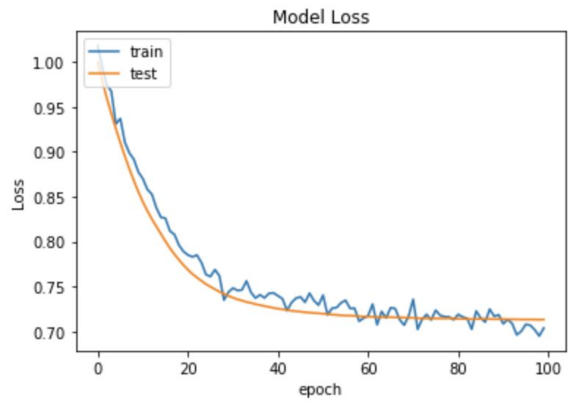

Figure for Model Loss (Reduction of loss achieved)

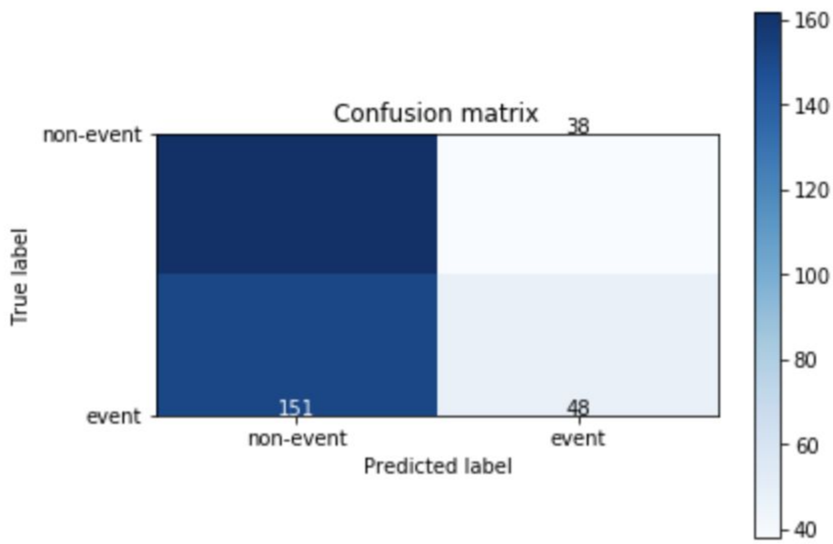

Figure for Confusion Matrix

I was successfully able to downsample the data to fit the size of the input file for training.

5 Conclusion

Training the model and getting machine learning involved in classification before analysing the data (i.e. Run-time application) is the bright beginning of a new era of applications of Deep learning. Its use is proven to be beneficial in many applications and adapting Artificial intelligence in all the realms of experimenting, analyzing and providing solutions to a crucial issue is the innovative start of the future of science.

This is just one variation of the method used for the ATTPC Mg-22 data. Results for this variation are not the best but more focussed work in this promising area (classification of ATTPC events to reduce the time required for sorting of data) could certainly help improve the F1 score and the accuracy of the model. In this variation too, changing the learning rate and executing the training is important. Numerous variations are possible with the same data. This project has a lot of scope of improvement as different parameters, *larger cleaner varied datasets* can be tried for better accuracy. Optimal position, model type and learning rate must be identified with further experimenting.[7]

6 References

[1] Gazzoli Barbara, European Center for theoretical research in nuclear physics. https://ectstar.fbk.eu/node/4474 Accessed on 6/12/20.

[2] Hjorth-Jensen Morton, M.P. Kuchera, R. Ramanujan. Machine Learning and Data Analysis for Nuclear Physics, a Nuclear TALENT Course at the ECT 2020. https://github.com/NuclearTalent/MachineLearningECT . Accessed on 8/4/20.

[3] J. Bradt, D. Bazin, F. Abu-Nimeh, T. Ahn, Y. Ayyad, S. Beceiro Novo, L. Carpenter, M.

Cortesi, M.P. Kuchera, W.G. Lynch, W. Mittig, S. Rost, N. Watwood, J. Yurkon, Commissioning of the Active-Target Time Projection Chamber, Nuclear Inst. and Methods in Physics Research, A. 875 (2017) 65– 79. doi:https://doi.org/10.1016/j.nima.2017.09.013

[4] M.P. Kuchera, R. Ramanujan, J.Z. Taylor, R.R. Strauss, D. Bazin, J. Bradt, Ruiming Chen, Machine learning methods for track classification in the AT- TPC, Nuclear Inst. and Methods in Physics Research, A. 940 (2019) 156– 167. doi:https://doi.org/10.1016/j.nima.2019.05.097.[5] Aurelien Geron, Hands‑On Machine Learning with Scikit‑Learn and TensorFlow, O'Reilly. March 2017.

[6] Lexi Weghorn, REU 2020, https://github.com/lexweg7/REU2020/tree/master/22Mg(alpha%2C%20proton) Accessed on 8/4/20.

[7] Andrew Ng. Fundamentals of machine learning. https://www.coursera.org/learn/machine-learning, 2014. Accessed on 8/2/20.

Additional website references used for background research:

[1]https://towardsdatascience.com/watching-machine-learning-models-fitting-a-curve-c594fec4bbdb

[2]https://machinelearningmastery.com/how-to-implement-a-machine-learning-algorithm/

[3] https://www.dataquest.io/blog/learning-curves-machine-learning/

[4]https://scikit-learn.org/stable/auto_examples/index.html#model-selection

[5]https://arxiv.org/abs/1808.05882

[6]https://www.sciencedirect.com/science/article/pii/S0168900217311798

[7]https://arxiv.org/abs/1807.03513