Nucl. Sci. and Tech., Vol.11, No. 2 (2021), pp. 01-08

Nuclear Science and Technology

Journal homepage: https://jnst.vn/index.php/nst

Applying machine learning method in neutrons and gamma-rays identification according to their pulse shapes

Le Xuan Chung[1], Vi Ho Phong[2], Le Tuan Anh[1], Nguyen Duc Ton[1], Nguyen Hoang Phuc[1] and Bui Duy Linh[1]

[1]*Institute for Nuclear* Science *and Technology, VINATOM, P.O.Box 5T-160, Nghia Do, Hanoi, Vietnam*

[2]*Hanoi University of Science, 334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam*

Abstract: Neutrons and gamma-rays from a ^{152}Cf source have been measured and separated based on the time of filght (TOF) technique. Their pulse shape characteristics measured by EJ-299-33 scintillator were used to train an artificial neural network (ANN) in a machine learning method. Afterwards, the ANN was used to predict another set of pulse shape data to identify neutron and gamma-ray events. Comparing to the charge-comparison method, the ANN gave better identification. This result proves a potential application of machine learning method in the nuclear data analysis.

Keywords: *ANN, machine learning, neutrons, gamma-rays, time of flight.*

I. INTRODUCTION

Artificial intelligence (AI) is widely applied in many aspects of society. Its aim is to enable computers to simulate human intelligence based on logic, rules, decision trees, and machine learning. In which, machine learning (ML) is an AI subfield including deep learning (DL) which contains a stack of hidden layers. The operation of ML can be based on an artificial neural network which mimics human brain [1]. In this manner, the ANN is trained by a sufficiently enormous data (given inputs with given outputs). Afterwards, it can find out the logic and be able to predict the outputs of new inputs. Therefore, the ANN has an advantage that it does not require an unambiguously mathematical input/output relationship.

In nuclear physics study, machine learning has been applied in for decades [2-3] and recently became intensive thanks to the computer's fast calculating speed and large storage. The ANN models were applied to derive successfully nuclear charge radii [4], nuclear mass in neutron star [5], mass and binding energy [6], ground-state energy and the ground-state point-proton root-mean-square radius along with their extrapolation uncertainties [7], or automatic feature extraction in heavy ion collisions [8].

Concerning neutron and gamma-ray, many nuclear structural information, such as spin and parity [9], half-lives [10, 11], et cetera, can be obtained by detecting them. In such studies, the detection of neutrons is normally accompanied by gamma-rays. This fact demands the identification of them. The well-known method is charge comparison (CC) which relies on the neutron and gamma-ray pulse shape difference (almost in the tail components) as seen in Fig. 1. In this method the charge ratio of the pulse-to-total waveform were calculated [12-13]. According to these difference, neutrons and gamma-rays can be identified.

As an alternative approach, similar to the work reported in [3, 14], this paper presents the application of ML method in neutron and gamma-ray identification via their waveform difference. Firstly, an experiment to measure and identify neutrons and gamma-rays using

https://doi.org/10.53747/nst.v11i2.355
Received 26 July 2021, accepted 30 September 2021

time of TOF technique was performed. Thanks to the experimental availability, the neutron and gamma-ray waveform data were digitized suitable for applying the ML method. 15729 data samples were used to train the ANN and 43460 left for testing. The results were compared to those obtained from charge comparison method.

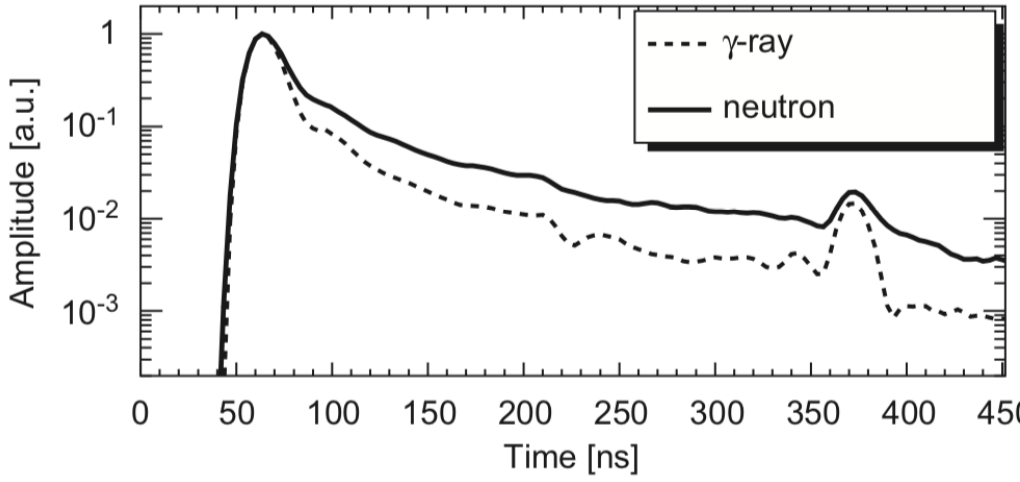

Fig. 1. Neutron and gamma pulse shapes normalized to the same amplitude. The difference appears in the tail components at about 70 ns. This figure is taken from [13]

II. TOF MEASUREMENT AND DATA PREPARATION

The main experimental setup is presented in Fig. 2. A ^{252}Cf neutron source which also emits gamma of up to several MeV energies was placed in front of an EJ-299-33 scintillator [15]. Another scintillator of the same type was placed around 1.2 m away from it. They were coupled to high-optic-efficiency H11265 Hamamatsu photomultiplier (PMT) operated at 1000 V. These two detectors provided start and stop signals, respectively. Their PMT signals were fed to 2 channels of a CAEN V1730 flash ADC. The trigger was active only when both detectors detected signals and the signal in the second one was above a given threshold.

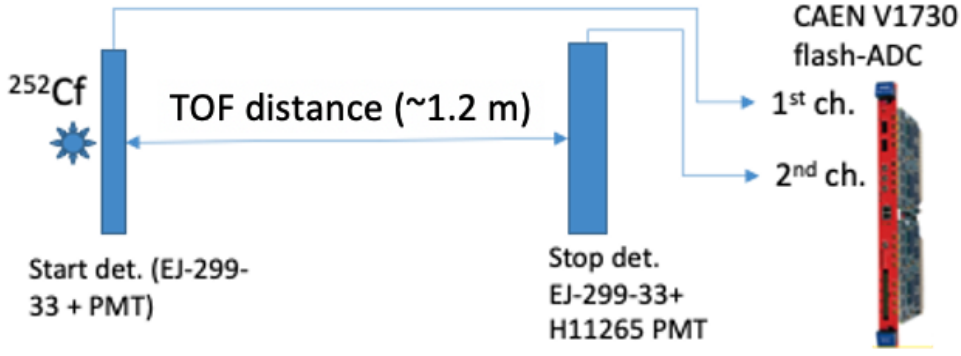

Fig. 2. Experimental setup for neutron and gamma-ray TOF measurement, see text for details

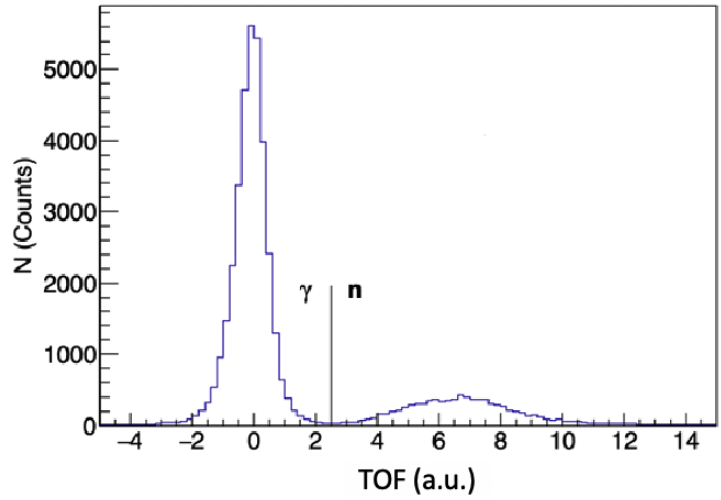

Fig. 3. Uncalibrated time of flight of neutrons and gamma-rays in about 1.2 m

The TOF was extracted by constant fraction discrimination (CFD) algorithm [16]. The time resolution in our experiment was 185 ps full width half magnitude (FWHM).

The uncalibrated TOFs of neutrons and gamma-rays from the ^{252}Cf source is presented in Fig. 2. Two peaks are observed at around 0 and 7 (a.u.). The first one corresponds to gamma-rays because they have the speed of light and the left to neutrons with slower speed. These two components can be delimited by a vertical line at 2.5 (a.u.) as shown in this figure.

The flash ADC digitized the entire signal waveforms from the detectors. The digital data were analyzed by two methods to identify ones induced by neutron and gamma-ray: CC and ANN pulse shape discriminations (PSD). For the ANN PSD, firstly the waveform was digitized into 400 samples (equivalent to an 800-ns-time window) whose amplitudes were denoted as X_i (i=0-399). According to TOF (see Fig. 2) they were certainly identified as neutron or gamma-ray tagged as 1 and 0, respectively, forming an "Activity" matrix. The first 5 events' digital data structures are illustrated in Tab. 1.

Table 1. The first 5 digital data structures. Each data sample is a vector of 400 elements ($X_{0\text{-}399}$). The last "Activity" column implies neutron and gamma-ray events tagged as 1 and 0, respectively

X0	X1	X2	X3	X4	X5	X6	X7	X8	X9	...	X397	X398	X399	Activity
0.037903	0.037659	0.038086	0.038025	0.037903	0.038025	0.038025	0.037903	0.037964	0.037720	...	0.039062	0.038635	0.038208	1
0.037720	0.037964	0.037903	0.037476	0.037781	0.037720	0.038025	0.037964	0.037659	0.037598	...	0.037781	0.037598	0.037720	1
0.037964	0.037659	0.037781	0.038025	0.038086	0.037964	0.037781	0.037903	0.038208	0.037903	...	0.038391	0.037964	0.037964	1
0.037537	0.037781	0.037537	0.037903	0.037842	0.037781	0.037842	0.037842	0.038025	0.037964	...	0.038147	0.037903	0.037903	0
0.037781	0.037781	0.037903	0.037964	0.037964	0.037781	0.037903	0.037720	0.037537	0.037903	...	0.037903	0.037537	0.037903	0

III. PSD METHOD AND RESULT

3.1. Charge comparison

As mentioned above, in the CC PSD, the charge ratio of the pulse-to-total waveform were calculated. The pulse charge (Q_p) and the total charge (Q_t) were integrated within the short and long gates, respectively. This method is

illustrated in Fig. 3 where the width of such gates were optimized using the commonly used figure-of-merit (FOM) that quantify the discrimination capability of the CC PSD results.

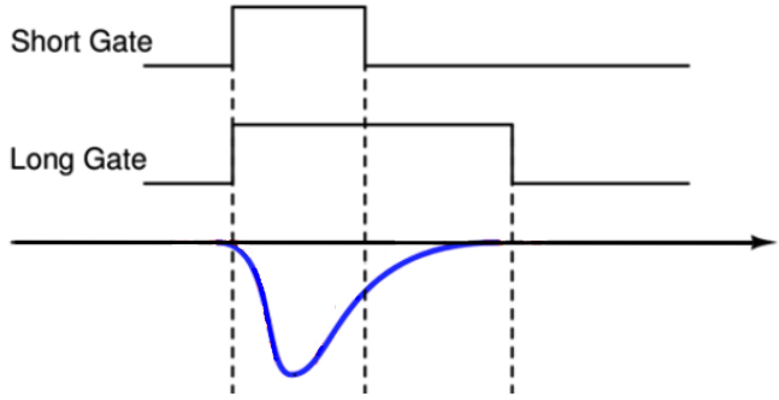

Fig. 3. Illustration of charge comparison method. The long gate covers the whole component. While the short one covers almost the pulse

In the next step, the charge ratio of the pulse-to-total waveform (PSD_{CC}) was calculated as $PSD_{CC}=(Q_t-Q_p)/Q_t$. The result is presented in Fig. 4. Due to the fact that PSD_{CC} is characterized by neutron and gamma-ray, according to the figure, these events are delimited by the vertical line at 0.34.

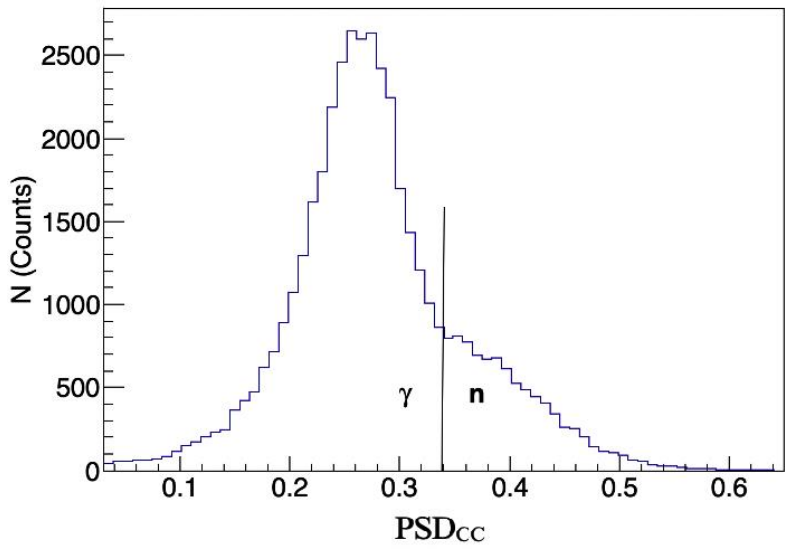

Fig. 4. Charge ratio of the pulse-to-total waveform (PSD_{CC}). Neutron and gamma-ray components are delimited by the vertical line at 0.34

3.2. Artificial neural network

An ANN model has been built using "Sequential" model, the simplest Keras neural network type. Keras is high-level Deep Learning Application Programming Interface [17] which is bundled in Tensorflow [18], an open source Machine Learning Platform. The model construction is illustrated by part of the Python code as follows:

```
import tensorflow as tf
from tensorflow import keras
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(400,)),
    keras.layers.Dense(80, activation=tf.nn.relu),
    keras.layers.Dense(20, activation=tf.nn.relu),
    keras.layers.Dense(1, activation=tf.nn.sigmoid),
])
```

The first is “Flatten” layer acting as input layer. Its role is to convert each input in to 1D array. In this layer, the input shape should be specified, for example “input_shape = 400” for data structure in Tab. 1.

There are 2 hidden “Dense” layers consisting of 80 and 20 neurons, respectively. They use the “relu” activation function [19]. Each layer manages its own weight matrix connecting the neurons to their inputs. When all neurons in a layer are connected to everyone in the previous layer, it is call “Dense layer”. If input data is passed, it computes the perceptron h as [1]:

$$h_{W,b}(\mathbf{X})=\phi(\mathbf{XW}+\mathbf{b}) \quad (1)$$

where, **X** represents the input matrix (see Tab. 1 for an example), **W** contains all the connection weights, the bias vector **b** contains all connection weights between bias neuron and the artificial neurons, ϕ is the activation function [19].

The last one is output layer consisting of a single neuron. Because as mentioned in section II, the output (PSD_{ANN}) is a number between 0 and 1 the current problem belongs to a binary classification. Therefore, the “Sigmoid” activation function [19] was chosen.

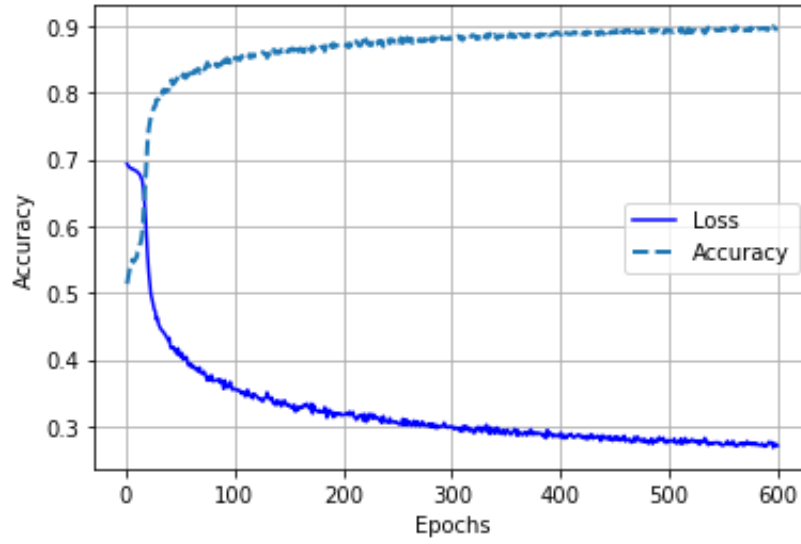

Fig. 5. Training accuracy versus epochs

The above “Sequential” model was trained by 15729 data samples from the experiment. The training process was carried out through 600 epochs to assure the accuracy conversion as shown in Fig. 5. One can see that the training accuracy steadily increases, while the training loss decreases. The model accuracy on the training set was 91.45 %.

After training, the ANN model was applied to predict neutrons and gamma-rays according to their pulse shapes digitized as X_i (i=0-399). 43460 experimental data samples were used. The ANN outputs (denoted as "PSD_{ANN}") are shown in Fig. 6. The neutron and gamma-ray components are delimited by vertical line at PSD_{ANN}=0.5.

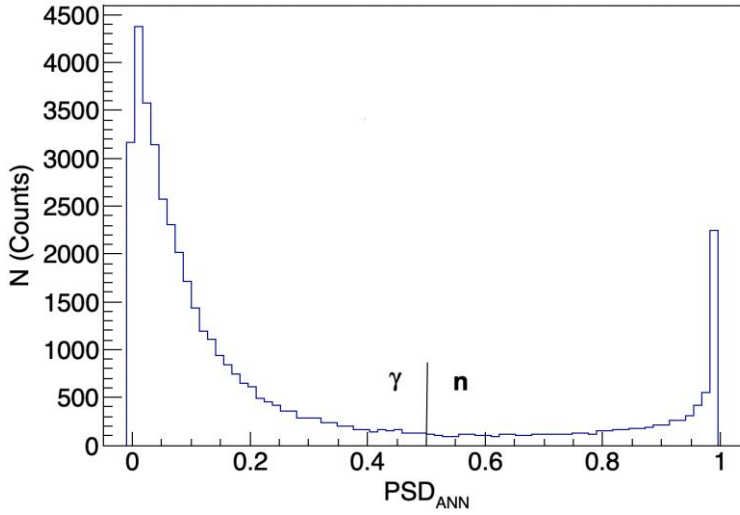

Fig. 6. ANN pulse shape discrimination (PSD_{ANN}) for n and γ delimited by a line at 0.5

3.2. Method evaluation

To evaluate CC and ANN methods, the TOFs of gamma-ray and neutron events selected by their condition on PSD_{CC} and PSD_{ANN} are plotted in Figs. 7 and 8, respectively. For gamma-rays, the identification accuracy is nearly the same in both methods, ~95 % in Fig. 7. While for n identification, the ANN's (89.26 %) is much better than the CC's (79.60 %). The accuracy is defined as the ratio of true-to-total event number. The true events in Fig. 7 are with TOF smaller than 2.5 (a.u.) and vice versa for the true neutron events on Fig. 8.

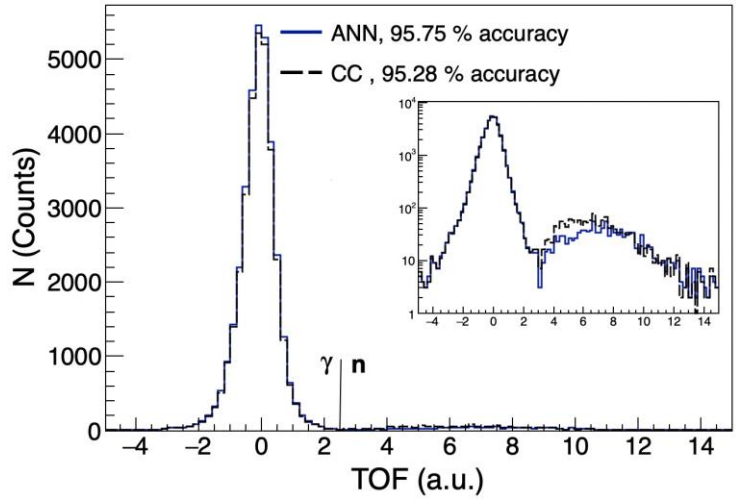

Fig. 7. TOF of gamma-ray events selected by ANN and CC methods. The inset is the same figure but in logarithmic scale for y axis. See text for more details

For gamma-ray identification, the dashed line is higher in the TOF range from 2.5-7 (a.u.) but lower than the solid line with TOF > 7 (a.u.), see the inset in Fig. 7. This means that the CC method is worse for fast neutron discriminating than the ANN one. This fact is also observed in Fig. 8. The contrast behaviors of the dashed and solid lines show that the ANN method discriminates fast neutrons better than the CC one does. Note that identifying fast neutrons is more challenging because they are close to the gamma components in the TOF method.

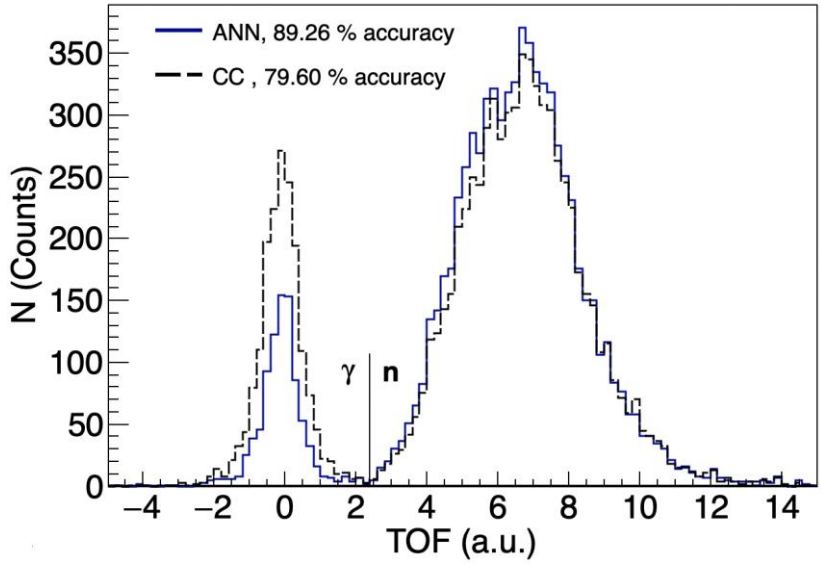

Fig. 8. TOF of neutron events selected by ANN and CC methods. See text for more details

IV. CONCLUSION

The neutrons and gamma-rays were measured and their TOFs in about 1.2 m were determined to identify them, accordingly. These events were also discriminated by CC and ANN methods via their waveforms induced on the EJ-299-33 scintillator. In the latter, the artificial neural network with 2 hidden layers of 80 and 20 neurons, respectively, was successfully built and trained. The results showed that gamma identification accuracy in both methods is similar. While for neutron identification, the ANN accuracy is better than the CC's. In both cases, the ANN identifies fast neutrons better than the CC does. As the result, the ANN method is potentially applicable in nuclear data analysis.

ACKNOWLEDGMENT

This work is supported by the Vietnam Atomic Energy Institute under the Grant No. CS/21/04-02.

REFERENCES

[1]. Aurélien Géron, *"Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow"*, O'Reilly Media publishing house, September, Second Edition, 2019.

[2]. S.Gazula and J.W .Clark, and H. Bohr, Nuclear Physics A540, 1, 1992.

[3]. E. Ronchi *et al.*, Nuclear Instruments and Methods in Physics Research A 610, 534, 2009.

[4]. S. Akkoyun, T. Bayram, S. O. Kara, and A. Sinan, Journal Physics G: Nuclear Particle Physics 40, 055106 (2013).

[5]. R. Utama, J. Piekarewicz, and H. B. Prosper, Physical Review C 93 (2016) 014311.

[6]. L. Neufcourt, Y. Cao, W. Nazarewicz, and F.Viens, Physical Review C 98 (2018) 034318.

[7]. G. A. Negoita *et al.*, Physical Review C 99 (2019) 054308.

[8]. Long-Gang Pang, Nuclear Physics A 1005 (2021) 121972.

[9]. B. D, Linh *et al.*, *"Investigation of the ground-state spin inversion in the neutron-rich ^{47,49}Cl isotopes"*, submitted to Physical Review C (2021).

[10]. V. H. Phong *et al.*, Physical Review 100 (2019) 011302(R).

[11]. L. X. Chung *et al.*, Nuclear Science and Technology 9, 48, 2019.

[12]. https://indico.cern.ch/event/737461/contributions/3730613/contribution.pdf

[13]. P.-A. Söderström and J. Nyberg, R. Wolters, Nuclear Instruments and Methods in Physics Research A 594 (2008) 79.

[14]. P.-A. Söderström *et al.*, Nuclear Instruments and Methods in Physics Research A 916 (2019) 238.

[15]. S. Nyibule *et al.*, Nuclear Instruments and Methods in Physics Research Section A 768 (2014) 141.

[16]. https://www.caen.it/download/?filter=CoMPASS

[17]. https://keras.io

[18]. https://www.tensorflow.org/

[19]. https://keras.io/api/layers/activations/