INL/EXT-19-55616

Application of Artificial Neural Network to Prompt Gamma Neutron Activation Analysis for Chemical Warfare Agents Identification

Dongwon Lee

September 2019

The INL is a U.S. Department of Energy National Laboratory
operated by Battelle Energy Alliance

INL/EXT-19-55616

Application of Artificial Neural Network to Prompt Gamma Neutron Activation Analysis for Chemical Warfare Agents Identification

Dongwon Lee

September 2019

Idaho National Laboratory
Nuclear Nonproliferation Division
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Office of National Nuclear Security Administration
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

ABSTRACT

The Portable Isotopic Neutron Spectroscopy (PINS) is a commercialized system developed by Idaho National Laboratory (INL) to examine chemical warfare agents (CWA) non-destructively, utilizing Prompt Gamma Neutron Activation Analysis (PGNAA) techniques. The PINS system takes advantage of a high-resolution gamma-ray spectrum from a mechanically-cooled high-purity germanium (HPGe) detector, and gamma-ray peak analysis provides input to its chemical identification logic with a probabilistic decision tree (PDT). The effectiveness of the chemical identification algorithm is determined by the availability of a wide range of data to train the algorithm to identify chemical-fills with accuracy. INL has a collection of gamma-ray spectra of various chemical-fills from the field-deployed PINS systems over the years, and it was envisaged to leverage such a database with the Artificial Neural Network (ANN) technique. Therefore, an ANN-based chemical identification algorithm was developed as an independent verification of the current algorithm. The ANN-based algorithm's performance was evaluated against the U.S. Army blind test data, and results were presented and discussed in this study.

CONTENTS

FIGURES

TABLES

ACRONYMS

ANN	Artificial Neural Network
CWA	Chemical Warfare Agents
D-D	Deuterium-deuterium
FP	False Positive
FPR	False Positive Rate
FN	False Negative
FNR	False Negative Rate
HPGe	High purity germanium
INL	Idaho National Laboratory
MRC	Multiple Round Container
MSE	Mean Squared Error
PCA	Principal Component Analysis
PDT	Probabilistic Decision Tree
PGNAA	Prompt Gamma Neutron Activation Analysis
PINS	Portable Isotopic Neutron Spectroscopy
PPV	Positive Prediction Value
TP	True Positive
TPR	True Positive Rate
TN	True Negative
TNR	True Negative Rate

APPLICATION OF ARTIFICIAL NEURAL NETWORK TO PROMPT GAMMA NEUTRON ACTIVATION ANALYSIS FOR CHEMICAL WARFARE AGENTS IDENTIFICATION

1. INTRODUCTION

The Portable Isotopic Neutron Spectroscopy (PINS) is a highly fieldable system developed by Idaho National Laboratory (INL) to examine chemical munitions and containers non-destructively [**1**]. The PINS system is an application of the Prompt Gamma Neutron Activation Analysis (PGNAA) technique, and it has been successfully commercialized by ORTEC® to assay chemical warfare agents (CWA) around the world [**2**]. The PINS system collects gamma-ray spectra using either a liquid nitrogen-cooled or a mechanically-cooled high-purity germanium (HPGe) detector to resolve many gamma-ray peaks emitted from neutron capture and neutron inelastic-scattering reactions on chemical compounds. A five microgram californium-252 source or a neutron generator is used to produce neutrons (see **Figure 1**). Background subtraction, energy calibration, peak searches and peak fittings are performed on the gamma-ray peaks in the spectra collected from the CWA, then intensities and ratios of intensities of the selected peaks are used to find the most probable chemical fill in the database.

Peak analysis combined with a probabilistic decision tree (PDT) as shown in **Figure 2** has evolved to be an effective algorithm for the chemical identification through years of field operations. The splitting criteria at the nodes have been fine-tuned after analyzing many spectra of various chemical-fills. The performance of the chemical identification algorithm is highly dependent on the availability of a wide range of data that can be used to test and improve its accuracy. INL has a collection of gamma-ray spectra of various chemical-fills from the field-deployed PINS systems over the years, and it was envisaged to leverage such valuable resources with other classification methods. An algorithm based on the Principal Components Analysis (PCA) technique has shown possibility of unsupervised learning [**3**]. As part of an ongoing research effort to improve the PINS identification algorithm, an Artificial Neural Network (ANN) algorithm was developed to identify CWA. The results of CWA identification by the ANN technique are presented and discussed in this study.

2. PREPARATION OF PINS DATA FOR TRAINING AND VALIDATION

A total of 153 HPGe spectra of 21 different chemical-fills were prepared to train and validate an ANN model as summarized in **Table 1**. These gamma-ray spectra were previously collected with PINS-CF systems in the field operations and laboratory environment, and the assessed munitions (projectiles)

varied in size: bomblet, 4.2", 75mm, 105mm, 155mm etc. After training, the ANN model was tested against a holdout dataset of 23 gamma-ray spectra as listed in **Table 2**. PINS+ spectral analysis on a spectrum produces a text file with all spectral analysis results, and **Figure 3** shows a snippet from an example analysis file. This snippet shows the section of spectral analysis results on 27 elements. Some elements, however, were not directly related to chemical-fills of interest, e.g. iron, germanium and lead. Therefore, net count rate in counts per second (converted from Area/1ksec values in the analysis file) of only 15 key elements (P, O, N, As, Br, Sn, Ti, H, Si, Na, Ca, Zn, Cl, S and B) and the Cl/S ratio value were extracted to be the input values. A typical single hidden layer artificial neural network model was adopted to have the input layer of 16 nodes and the output layer of 21 nodes (21 chemical-fills to be identified) as illustrated in **Figure 4**. In order to evaluate the trained ANN model's performance on a larger dataset, another collection of data from the blind test campaign in 2018 were prepared in the same way for performance evaluation of a trained ANN model, which is discussed further in **Section 4**.

3. ANN TOOL

The OCTAVE nnet-0.1.13 package is an ANN tool of a feedforward and backpropagation multi-layer neural network model, and the parameters to construct an ANN model in this study are shown in **Table 3** [**4**]-**[**5]. All data were preprocessed in columns so that the input dataset were presented as a $R \times N$ matrix, where R is the number of input nodes (16 in this study) and N is the number of input data (153 in this study). Then, the input dataset were randomly re-ordered and a third of the shuffled input dataset were reserved as a validation dataset to tune the model. The remaining training dataset, two thirds of the initial input data, were standardized to have a mean value of 0 and a standard deviation of 1, and the training dataset's mean and standard deviation vectors ($R \times 1$ vectors) were used to standardize the validation data and the holdout dataset. Next, an ANN model was trained and validated until two conditions were met: a trained model predicted the holdout dataset correctly and its MSE (Mean Squared Error) values between the true and the predicted outputs reached the minimum value.

This procedure to train and test an ANN model was visualized in a flowchart shown in **Figure 5**, but it was not determined how many nodes in the hidden layer would be optimal for this study. When there are too few or too many hidden nodes, an ANN model is subject to under-fitting or over-fitting. Therefore, the procedure shown in **Figure 5** was repeated for various numbers of hidden nodes from 5 to 32 in order to find the best performing number of hidden nodes. The minimum MSE value of the holdout dataset was calculated at the end of the iterations for a given number of hidden nodes. **Figure 6** shows a plot of the minimum MSE values as a function of the number of hidden nodes. When the number of hidden nodes increases from 5, the MSE value decreases until the number of nodes reaches 17 and increases again past

17. As a result, the trained ANN model with 17 hidden nodes with the minimum MSE of 9.32e-4 was chosen to be used as a CWA identification algorithm. **Table 4** and

Table 4. 21×23 matrix of true identities of the holdout test data.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0

Table 5 show the true and the predicted output values with the trained model with 17 hidden nodes, respectively.

Once an ANN model was trained to meet two criteria, the key information of the trained ANN model were written to a summary file for future reference. The key information are vectors of means (a vector labeled $\boldsymbol{cMeanInput}$) and standard deviations (a vector labeled $\boldsymbol{cStdInput}$) of the training data's input values, a matrix of the input weights (a matrix labeled $\boldsymbol{IW}$), a matrix of the layer weights (a matrix labeled $\boldsymbol{LW}$), and two bias vectors (a vector labeled $\boldsymbol{IB}$ and a vector labeled $\boldsymbol{LB}$). For the trained ANN model in this study, $\boldsymbol{cMeanInput}$ and $\boldsymbol{cStdInput}$ are both 16×1 vectors, $\boldsymbol{IB}$ is a 17×1 vector, $\boldsymbol{LB}$ is a 21×1 vector, $\boldsymbol{IW}$ is a 17×16 matrix, and $\boldsymbol{LW}$ is a 21×17 matrix. It is not necessary for us to understand all

mathematics behind the ANN model provided by the OCTAVE nnet-0.1.13 package, but the output values in the output layer can be explained by a series of matrix operations with these vectors and the matrices from the trained ANN model. First, the count rates of the 15 elements and the Cl/S ratio, r_i, were standardized by

$$R_i = \frac{(r_i - \bar{r}_i)}{\sigma_i}, \qquad i = 1 \cdots 16 \tag{1}$$

where R_i is standardized count rate, $\bar{r}_i$ and σ_i are the mean and the standard deviation of r_i values in the training data set, respectively. The standardized count rates were used to calculate A_j values in the hidden layer as shown in **Figure 7**, and A_j values were calculated by

$$A_j = logsig\left(IB_j + \sum_{i}^{16} R_i \times IW_{j,i}\right), \qquad j = 1 \cdots 17 \tag{2}$$

where $logsig$ is the log-sigmoid transfer function, IB_j is the j^{th} element of the input bias vector $\boldsymbol{IB}$ and $IW_{j,i}$ is the (j, i) element of the input weight matrix $\boldsymbol{IW}$. Similarly, A_j values in the hidden layer were used to calculate O_k values in the output layer as shown in **Figure 7**, and O_k values were calculated by

$$O_k = logsig\left(LB_k + \sum_{j}^{17} A_j \times LW_{k,j}\right), \qquad k = 1 \cdots 21 \tag{3}$$

where LB_k is the k^{th} element of the layer bias vector $\boldsymbol{LB}$ and $IW_{k,j}$ is the (k, j) element of the input weight matrix $\boldsymbol{LW}$. An octave script file to perform all processes described in this section is shown in **APPENDIX A**: ANN_TEST.M.

4. BLIND TEST DATA

Data from the blind test campaign performed in 2018 were tested by the trained ANN model in order to evaluate performance of the ANN technique for PINS-CF CWA identification. The blind test data were from chemical fills in the 4.2" mortar projectiles inside 7" Multiple Round Container (MRC). A total of 93 blind test spectra collected with PINS-CF systems were available for this study, and there were 21 spectra whose chemical-fills were not included in the ANN model's training: AC (hydrogen cyanide), CN (chloroacetophenone), PS (chloropicrin), TH3 (thermate) and "empty". Also, the blind test data didn't include spectra with 7 chemical-fills included in the training: AF, BL, POP, FM, HC, SA and WP. **Table 6** shows the predicted results against the blind test data by the trained ANN model as well as those from PINS+ v6.5.2 analysis. As mentioned above, the trained ANN model was not taught to identify AC, CN, PS, TH3 and "empty" so only the 72 blind test data of 15 chemical-fills were used to create a multi-class

confusion matrix shown in **Figure 8**. WP was not one of the chemical-fills included in the blind test, but it was included in the 15 chemical-fills since some GA/GB runs were falsely predicted as WP by the trained ANN model. The numbers of true positives, false negatives, false positives and true negatives are summarized in **Table 7**. Also, true positive rates (TPR), false negative rates (FNR), false positive rates (FPR), true negative rates (TNR), accuracy and precision values for 15 chemical-fills are shown in **Table 7**. TPR, FNR, FPR and TNR are defined by

$$TPR = \frac{TP}{TP + FN}$$

$$FNR = \frac{FN}{TP + FN}$$

$$FPR = \frac{FP}{TN + FP} \quad (4)$$

$$TNR = \frac{TN}{TN + FP}$$

where TP, TN, FP and FN represent the number of true positives, true negatives, false positives and false negatives. The accuracy and the precision (also known as Positive Prediction Value, PPV) values were defined by

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

and by

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

The overall accuracy and the precision values for all 15 chemical-fills were found to be 92% and 38% by the trained ANN model. **Figure 9** shows a plot of 15 chemical-fills of the blind test data on the FPR-TPR space from positive prediction where the point (0.0, 1.0) represents a perfect classification (corresponding to 100% TPR and 0% FPR). The chemical-fills plotted above the diagonal dashed line (also known as the line of no-discrimination) were considered predicted better than those below the line. 7 chemical-fills on the bottom left of the plot had 0 true positives as summarized in **Table 7**. All 7 CK runs were mistaken for CG, and all CNB and CNS runs were incorrectly predicted to be other chemical-fills. CG, CK, CNS and CNB could be easily mistaken for each other due to their subtle differences, making them a group of high false positive and false negative rates. As a result, CG ended up with the highest false positive rate of 29%. The trained ANN model also poorly performed on BR2 and WL possibly due to weak arsenic count

rate in the blind test data. GA/GB runs were also mistaken for WP, increasing WP's false positive rate to 8%.

5. SUMMARY

The Octave nnet-0.1.13 package was used to identify chemical-fills for PINS-CF systems. An ANN model was trained and validated with a total of 153 spectra of 21 different chemical-fills in various munition sizes. A total of 23 spectra were reserved as a holdout dataset, and the number of nodes in the hidden layer were found to be 17 after searching for the least MSE value with the holdout dataset. Then, the trained ANN model was applied to the U.S. Army blind test data collected with PINS-CF systems in 2018, and the ANN model's performance evaluation results were presented in this report. The overall accuracy and the precision values for 15 chemical-fills were found to be 92% and 38%, respectively. In comparison with the ANN model, the PINS+ V6.5.2 analysis results showed remarkable 98% accuracy and 83% precision due to its sophisticated PDT algorithm.

It is speculated that higher false positive rates of CG and low true positive rates of CK, CNB and CNS could be improved by introducing the chlorine i/c ratio as the 17th input variable in the future study. It is desirable to keep growing the training and holdout datasets with a larger population to improve the ANN technique's precision, especially for BR2, WL and GA/GB. Simulated spectra of rare chemical-fills using MCNP would be a potential solution to be considered to increase the population of the training data. Finally, it would be an interesting idea to train an ANN model to identify chemical-fills in specific munition sizes, e.g. 75mm-HD or 155mm-HD, since count rates are sensitive to the munition sizes as well as the chemical compositions.

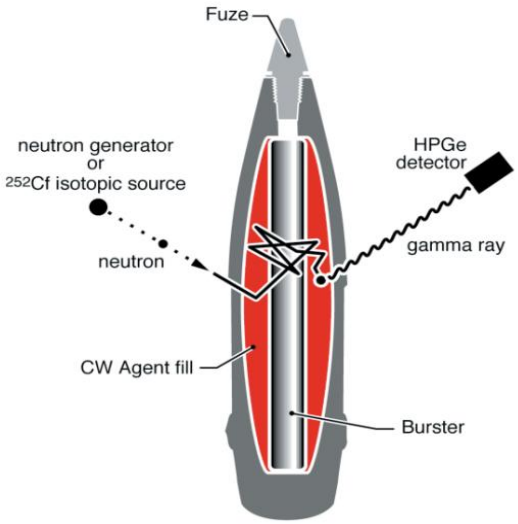

Figure 1. (Left) a schematic of the PINS system. (Right) the latest generation of the PINS system with a D-D neutron generator and a mechanically-cooled HPGe detector.

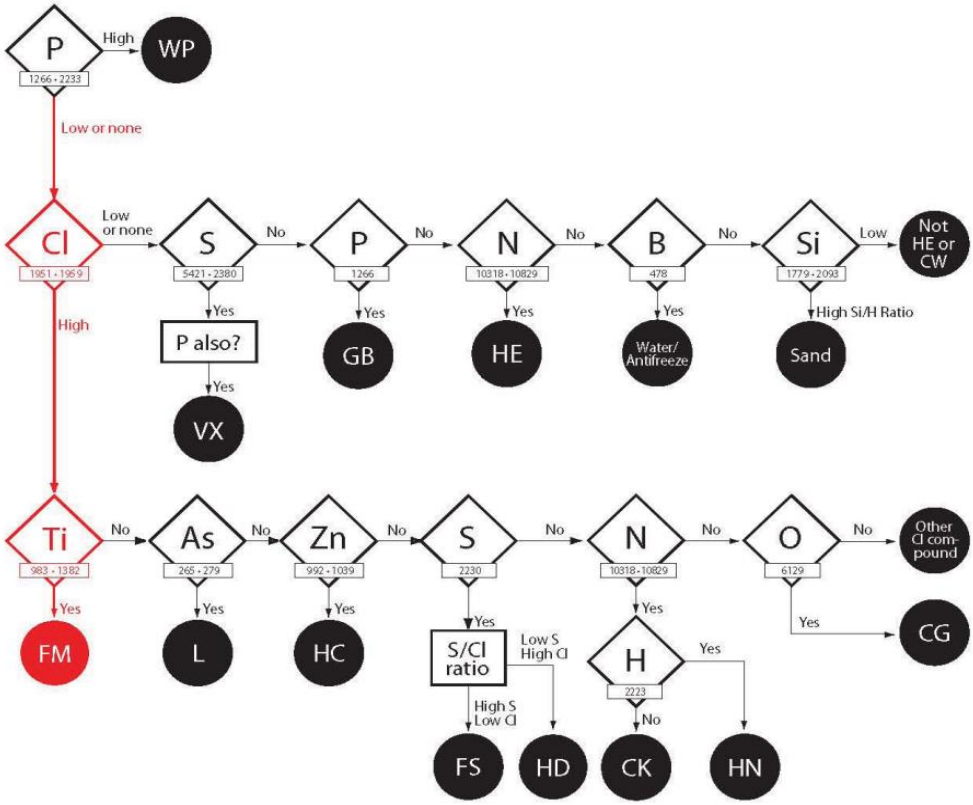

Figure 2. Probabilistic decision tree for the PINS CWA identification algorithm [2].

Isotope#=	Isotope	Detected	Confidence	Area/1Ksec	Uncert(%)
0001=	Fe	Yes	100	3563	286.45
0002=	P	No	0	-42	100.10
0003=	O	No	25	39	54.23
0004=	Al	Yes	50	644	252.50
0005=	N	No	0	-1	4.04
0006=	K	No	0	6	54.81
0007=	Ge	Yes	50	4	4.36
0008=	As	No	0	-105	435.19
0009=	Br	No	0	84	559.60
0010=	Sn	No	0	-395	227.68
0011=	Pt	No	2	358	376.10
0012=	I	No	0	15	166.37
0013=	Ti	No	0	-35	122.66
0014=	H	Yes	50	683	167.58
0015=	Ba	No	2	78	94.21
0016=	Si	No	0	79	161.74
0017=	Na	No	0	-153	300.18
0018=	F	No	0	0	0.00
0019=	Ca	No	0	-119	143.58
0020=	Zn	No	0	-14	121.25
0021=	Pb	No	0	0	0.00
0022=	Tl	No	0	0	0.00
0023=	Hg	No	0	-204	359.71
0024=	Cl	Yes	100	738	93.77
0025=	C	No	25	29	38.52
0026=	S	Yes	100	1255	94.67
0027=	B	No	0	-414	609.75

Figure 3. Snippet of the summary section from a PINS+ analysis file. This section summarizes count rates (net area per 1000 seconds) of 27 key elements. Some elements are not related to chemical-fills so only 15 count rates and the Cl/S ratio were used as input values.

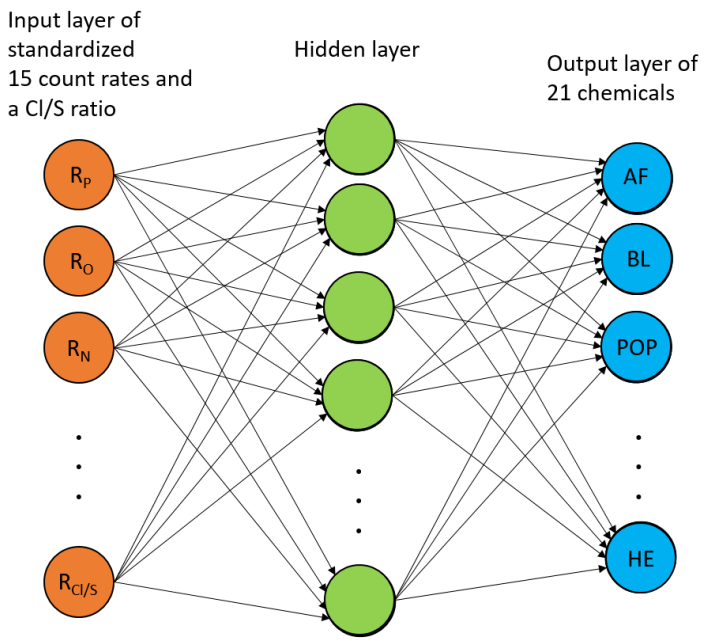

Figure 4. A model of single layer artificial neural network for PINS CWA identification. The input layer has a total of 16 nodes of count rates of the key elements and Cl/S ratio, and the output layer has 21 nodes for possible chemical-fills.

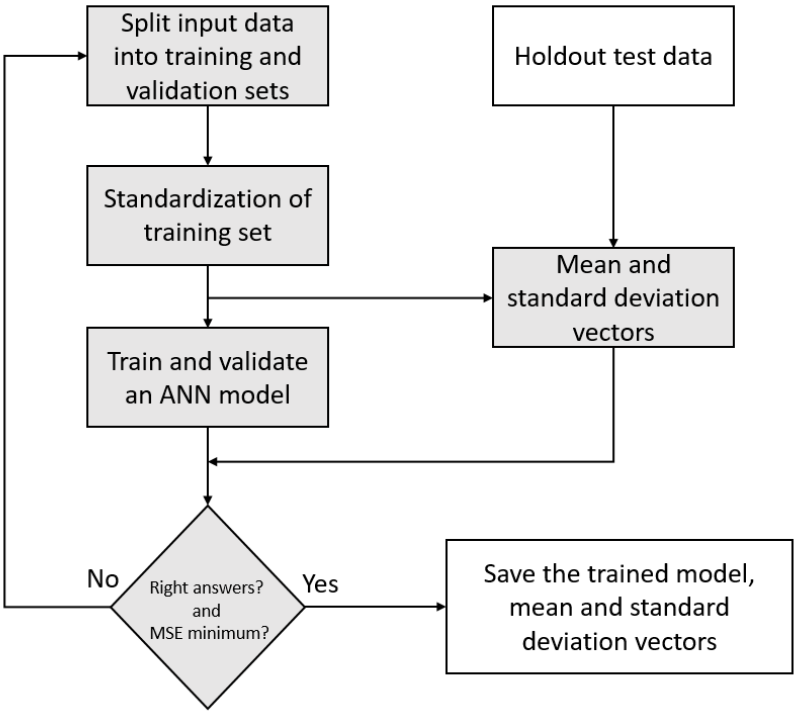

Figure 5. A procedure used in this study to train and find an optimized ANN model.

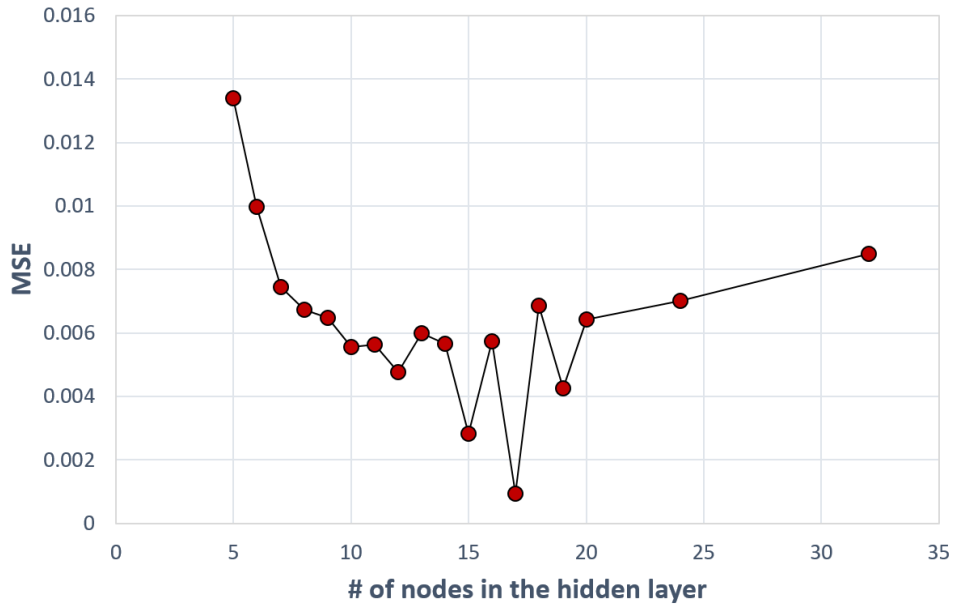

Figure 6. The optimal number of nodes in the hidden layer after the same procedure shown in **Figure 5** was repeated by varying the number of nodes in the hidden layer. Mean-squared-error values of the holdout test data were plotted as a function of the number of nodes in the hidden layer.

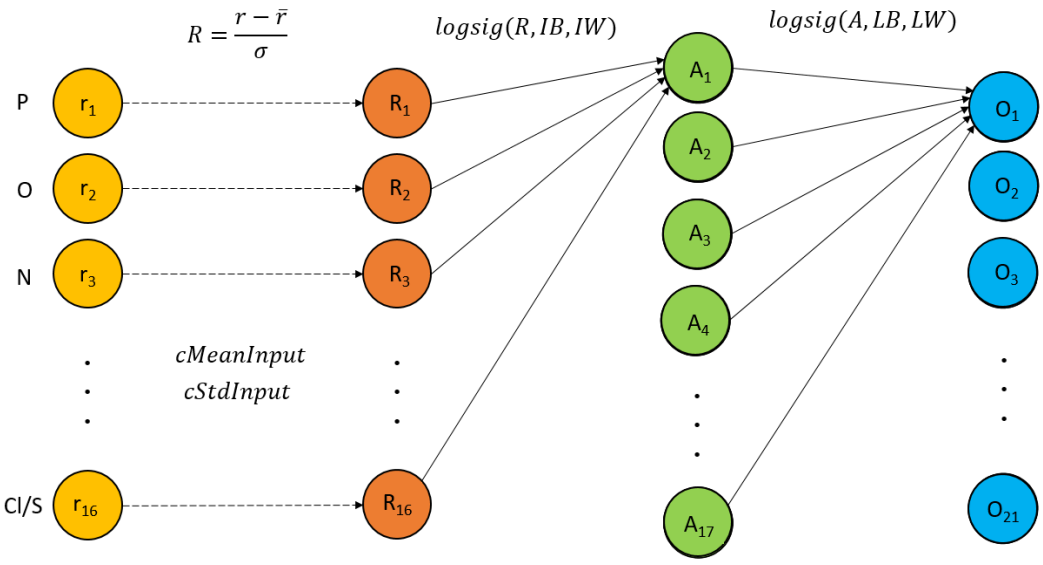

Figure 7. Conversion process of raw input values into output values in the trained ANN model. The raw input values were standardized followed by a series of matrix operations. The standardized input values contribute to each node in the hidden layer by the input weight matrix and the input bias vector, and each node in the output layer is calculated by the layer weight matrix and the layer bias vector.

	2018 blind test	Predicted														
		CG	CK	CNB	CNS	FS	HD	HN	L	BR2	WL	VX	GA/GB	WP	HE	Sand
TRUE	CG	**7**														
	CK	7														
	CNB	1			3			1								
	CNS	5	1					1								
	FS					**4**										
	HD	3					**2**									
	HN				2			**2**								1
	L	1							**4**							
	BR2								1							2
	WL	2														1
	VX											**2**	2		1	
	GA/GB											1		6	3	
	WP															
	HE/TNT														**3**	
	Sand															**3**

Figure 8. Confusion matrix of the blind test data identification results by the trained ANN model. 72 out of the 93 blind test data were tested. 5 chemical-fills were not include in this matrix because four chemical-fills (AC, CN, PS, TH3 and "empty") were not used in the training and the blind test data didn't include two practice fills (AF and BL). The overall precision was found to be 38%, and the overall accuracy of 92% was achieved by the trained ANN model.

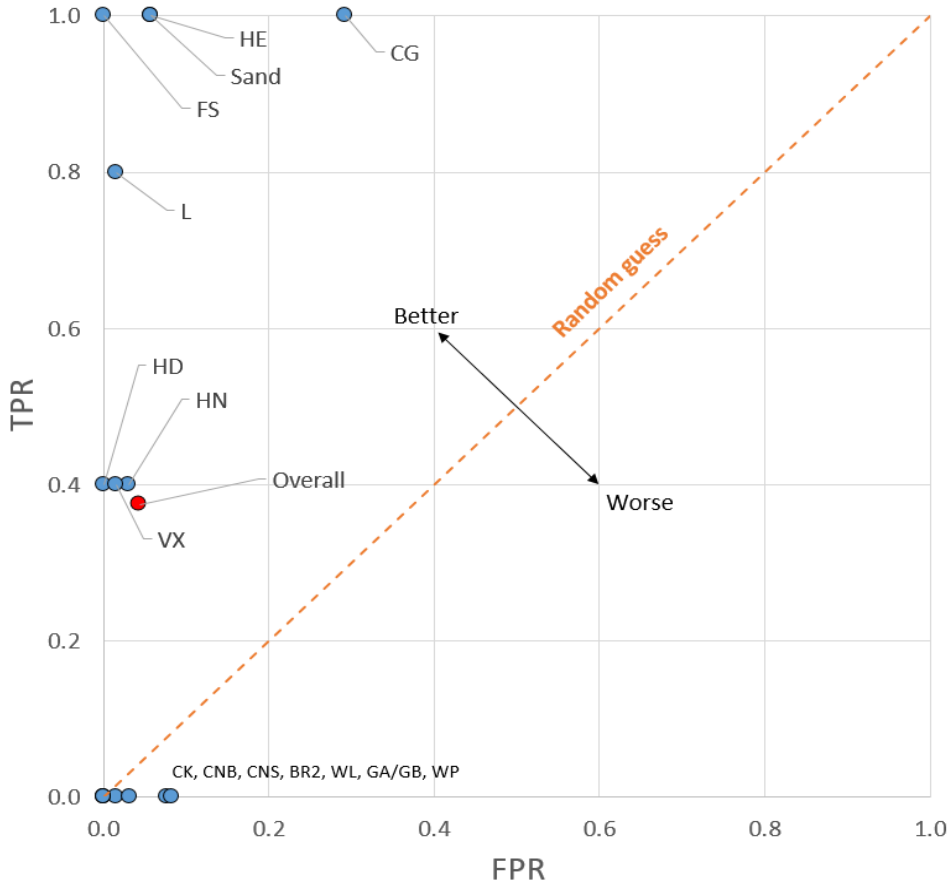

Figure 9. A plot of TPR vs. FPR values of the blind test data predicted by the trained ANN model.

Table 1. Summary of the 153 input spectra of 21 chemical-fills. These spectra were randomly split into the training (102 spectra) and the validation (51 spectra) sets.

Chemical-fills	# of spectrum	Chemical-fills	# of spectrum
AF (antifreeze)	6	HD	11
BL (Bleach)	15	HN	7
POP (Plaster of Paris)	5	L	4
Sand	16	SA	3
CG	18	BR2	3
CK	4	WL	4
CNB	7	VX	5
CNS	4	GA/GB	3
FM	13	WP	8
FS	8	HE	5
HC	4		

Table 2. The holdout test data of 23 spectra dedicated to evaluate the trained ANN model.

Test spectrum	Chemical-fills	Description
1	BL (bleach)	155mm projectile
2	POP	4.2” mortar projectile
3	Sand	75mm projectile
4	FM	75mm projectile
5	FM	Livens projector
6	FS	4.2” mortar projectile
7	HD	4.2” mortar projectile
8	HD	4.2” mortar projectile
9	CG	155mm projectile
10	CG	4.2” mortar projectile
11	CG	75mm projectile
12	L	4.2” mortar projectile
13	L	4.2” mortar projectile
14	GB	155mm projectile
15	GB	155mm projectile
16	VX	155mm projectile
17	VX	155mm projectile
18	WP	75mm projectile
19	WP	155mm projectile
20	HE (Comp. B)	105mm projectile
21	HE (Comp. B)	155mm projectile
22	HE (RDX)	75mm projectile
23	HE (TNT)	4.2” mortar projectile

Table 3. Parameters used at the time of defining a model [**4**]-**[**5].

Parameter for an ANN model	Value
Size of the training set	102
Size of the validation set	51
Size of the test set	23
Nodes in the hidden layer	Varied from 5 to 32
Nodes in the output layer	21
Transfer function	logsig (log sigmoidal)
Backpropagation training function	trainlm (nnet default)
Backpropagation weight/bias learning function	learngdm (nnet default)
Performance function	MSE (mean squared error)

Table 4. 21×23 matrix of true identities of the holdout test data.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
AF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BL	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
POP	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sand	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CG	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CK	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNS	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
FM	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
FS	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HC	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HD	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
L	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
SA	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
BR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
WL	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
VX	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
GB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
WP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0
HE	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0

Table 5. 21×23 matrix of predicted output values by the trained ANN model with 17 hidden nodes. The MSE value was calculated to be 9.32e-4.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
AF	9.7E-6	3.3E-6	1.4E-4	6.2E-4	2.2E-7	1.1E-5	9.9E-5	8.5E-5	6.1E-5	2.6E-3	1.1E-3	1.3E-6	3.3E-7	8.8E-7	1.7E-7	6.6E-7	9.3E-7	1.2E-6	2.3E-6	9.9E-5	1.5E-5	2.4E-5	1.0E-4
BL	1.0E+0	2.8E-6	1.2E-5	2.6E-4	1.8E-9	1.3E-5	1.1E-6	4.1E-6	1.1E-2	2.7E-2	1.4E-2	3.2E-5	6.8E-5	7.0E-3	1.8E-3	4.4E-4	4.5E-3	0.0E+0	0.0E+0	7.2E-5	4.1E-5	6.2E-7	4.5E-6
POP	4.7E-4	1.0E+0	2.0E-3	8.8E-7	5.7E-8	1.4E-3	5.8E-5	5.1E-5	2.4E-4	4.7E-4	8.3E-4	1.4E-4	3.9E-5	1.1E-3	1.5E-2	8.8E-3	3.0E-3	1.3E-6	1.0E-6	2.6E-4	1.4E-3	5.9E-4	5.2E-4
Sand	1.9E-9	8.3E-2	9.9E-1	7.2E-3	8.4E-6	7.3E-6	5.8E-4	2.2E-4	5.5E-1	4.2E-4	2.8E-4	1.8E-1	9.8E-4	2.4E-3	9.4E-4	1.0E-2	7.4E-3	2.5E-2	7.1E-3	1.3E-2	2.0E-1	7.8E-5	3.4E-3
CG	1.4E-3	2.9E-3	6.9E-4	1.4E-5	1.8E-4	2.0E-9	1.6E-4	1.1E-5	9.9E-1	9.3E-1	9.8E-1	5.3E-8	8.2E-8	0.0E+0	4.0E-6	0.0E+0	0.0E+0	0.0E+0	0.0E+0	7.0E-7	2.4E-7	2.4E-4	9.9E-5
CK	1.1E-3	1.4E-7	1.3E-5	4.8E-3	2.6E-7	0.0E+0	1.4E-3	1.0E-2	2.9E-4	1.7E-1	7.9E-2	4.7E-3	1.1E-3	1.4E-3	2.0E-3	2.6E-3	2.5E-4	0.0E+0	0.0E+0	1.4E-3	5.1E-5	4.8E-2	3.2E-4
CNB	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0
CNS	0.0E+0	1.2E-4	1.1E-6	0.0E+0	0.0E+0	0.0E+0	2.2E-4	4.1E-5	4.5E-4	6.8E-3	3.2E-4	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	0.0E+0	2.6E-9	1.2E-8	1.9E-3	3.6E-2	1.2E-4	1.9E-3
FM	3.7E-4	1.2E-9	4.4E-6	1.0E+0	1.0E+0	1.2E-7	1.8E-7	1.3E-7	2.7E-6	3.2E-5	2.0E-3	1.1E-8	5.4E-8	2.0E-3	5.4E-3	9.1E-7	3.7E-7	5.6E-7	1.3E-5	1.7E-3	1.4E-3	4.7E-3	1.7E-3
FS	8.5E-4	1.3E-4	5.2E-5	8.1E-5	1.3E-2	9.9E-1	1.5E-3	2.1E-3	8.8E-6	3.0E-5	8.7E-5	1.9E-3	3.4E-3	2.3E-5	3.7E-5	6.3E-7	2.1E-6	7.9E-4	6.1E-4	1.4E-4	6.5E-5	2.8E-5	4.8E-5
HC	1.9E-5	1.7E-4	1.4E-3	9.6E-5	4.4E-3	2.3E-4	6.0E-5	7.7E-5	1.2E-5	9.2E-5	2.0E-5	3.0E-5	1.2E-4	4.4E-5	4.6E-6	3.3E-4	2.0E-4	5.9E-6	8.2E-6	1.4E-3	1.1E-3	2.2E-3	5.8E-4
HD	6.9E-8	6.9E-4	2.3E-5	0.0E+0	0.0E+0	1.8E-2	1.0E+0	1.0E+0	1.7E-6	1.1E-4	3.5E-6	1.5E-3	1.1E-4	9.3E-8	3.2E-8	1.3E-9	1.3E-9	2.0E-4	1.2E-5	0.0E+0	0.0E+0	1.8E-8	0.0E+0
HN	1.3E-3	3.2E-7	3.8E-7	7.7E-4	2.0E-2	9.8E-4	1.5E-4	4.9E-4	0.0E+0	3.3E-4	1.0E-5	3.7E-8	1.9E-7	3.1E-5	2.2E-6	6.4E-7	8.1E-7	1.7E-5	2.8E-4	6.2E-4	6.3E-4	8.0E-4	3.9E-4
L	1.8E-4	6.9E-5	2.2E-5	5.3E-6	9.0E-6	3.5E-2	2.6E-5	9.2E-5	1.4E-5	2.3E-4	3.5E-4	9.9E-1	9.8E-1	6.3E-4	8.3E-4	3.5E-4	1.3E-4	2.5E-5	2.2E-5	2.2E-3	6.0E-5	9.4E-4	6.6E-4
SA	4.2E-7	1.3E-5	2.7E-2	4.6E-8	2.9E-7	3.5E-8	7.2E-9	1.7E-8	1.7E-5	2.1E-5	8.9E-6	6.6E-2	2.9E-3	1.0E-4	3.3E-5	6.3E-3	1.2E-3	8.3E-5	4.4E-5	6.8E-5	2.2E-7	2.8E-4	5.3E-5
BR	2.9E-4	6.4E-7	2.9E-4	1.7E-6	2.2E-5	2.3E-5	2.1E-4	8.8E-4	8.9E-7	3.5E-5	1.0E-6	3.1E-2	6.1E-3	2.1E-6	4.0E-7	3.3E-6	2.6E-6	1.6E-7	1.5E-7	3.4E-4	3.8E-5	1.1E-3	7.4E-5
WL	1.1E-3	2.1E-7	2.2E-4	2.0E-4	4.1E-5	1.4E-2	8.8E-4	2.7E-3	5.9E-8	3.9E-8	7.0E-7	2.4E-4	9.7E-4	4.6E-5	1.1E-5	5.5E-4	3.2E-4	2.0E-5	2.8E-6	1.4E-2	8.5E-3	8.2E-2	6.4E-3
VX	5.7E-4	1.8E-3	4.7E-5	4.3E-5	2.6E-5	1.2E-3	1.5E-4	7.1E-4	2.9E-5	1.4E-5	8.9E-6	6.6E-3	2.9E-3	5.2E-3	6.9E-4	1.0E+0	9.8E-1	3.2E-3	3.3E-4	4.5E-3	1.8E-3	3.1E-3	4.5E-4
GB	6.4E-3	1.5E-6	9.0E-7	7.7E-4	9.7E-5	1.9E-4	1.2E-7	4.0E-7	2.5E-6	5.2E-5	9.8E-4	2.3E-4	1.3E-3	9.9E-1	9.8E-1	2.6E-3	1.3E-3	9.4E-5	2.3E-3	4.9E-5	1.7E-5	4.0E-5	1.2E-5
WP	1.9E-8	9.2E-8	7.8E-6	7.5E-7	1.2E-3	3.7E-2	1.7E-7	6.3E-8	6.9E-7	1.4E-6	2.1E-5	1.7E-5	1.2E-4	2.7E-3	5.9E-3	6.8E-9	4.3E-8	1.0E+0	1.0E+0	9.1E-5	1.8E-5	2.5E-4	3.0E-4
HE	6.6E-7	8.5E-8	2.8E-4	4.0E-5	5.5E-3	6.7E-6	4.1E-9	8.0E-9	5.8E-7	1.5E-5	4.1E-5	2.2E-4	2.1E-4	5.0E-5	4.2E-5	1.6E-3	2.9E-3	3.3E-5	1.8E-4	1.0E+0	9.9E-1	9.6E-1	9.8E-1

Table 6. Summary of the 93 blind test data with PINS-CF systems. The true identities and the predictions by PINS+ and ANN technique are shown. It should be noted that AC, CN, PS, TH3 and "empty" were not used to train an ANN model in this study. The 21 blind test runs with these 5 fills are shown in gray.

Run name	True ID	Predicted ID			
		PINS+ v6.5.2		ANN	
		ID	Score	ID	Value
APG-18-593_P22_10Aug18_009	FS	FS	100	FS	1.00
BT18-001_P2_20Jul18_009	CG	CG	100	CG	1.00
BT18-002_P2_20Jul18_006	HN3	CNB	65	HN	0.32
BT18-003_P2_20Jul18_003	CK	CNS	100	CG	0.77
BT18-004_P2_20Jul18_012	AC	AC/HE	100	HE	0.2
BT18-005_P2_23Jul18_009	PS	CG	100	CG	1.00
BT18-006_P2_23Jul18_012	CG	CG	100	CG	1.00
BT18-007_P2_23Jul18_003	CK	CG	100	CG	1.00
BT18-008_P2_23Jul18_006	WL	WL	100	CG	0.22
BT18-009_P22_10Aug18_003	CG	CG	100	CG	1.00
BT18-010_P22_10Aug18_006	AC	AC/HE	100	HE	1.00
BT18-011_P2_24Jul18_003	CNS	CG	100	CG	1.00
BT18-012_P2_24Jul18_006	HN3	CN	60	Sand	0.01
BT18-013_P2_25Jul18_009	CK	CG	100	CG	1.00
BT18-014_P2_25Jul18_014	VX	VX	100	GB	0.51
BT18-015_P2_25Jul18_003	CG	CG	100	CG	1.00
BT18-016_P2_25Jul18_006	GA	GA	100	WP	0.47
BT18-017_P2_26Jul18_009	HD	HD	100	CG	1.00
BT18-018_P2_26Jul18_012	CG	CG	100	CG	1.00
BT18-019_P2_26Jul18_003	CNS	CNS	100	CG	1.00
BT18-020_P2_26Jul18_006	GB	GB	100	WP	1.00
BT18-021_P2_27Jul18_009	PS	CG	100	CG	1.00
BT18-022_P2_27Jul18_012	GA	GA	100	HE	0.98
BT18-023_P2_27Jul18_003	CG	CG	100	CG	1.00
BT18-024_P2_27Jul18_006	GB	GB	100	WP	0.63
BT18-025_P2_30Jul18_009	VX	VX	100	GB	0.79
BT18-026_P2_30Jul18_012	CK	CG	100	CG	1.00
BT18-027_P2_30Jul18_003	GB	GB	100	WP	0.42
BT18-028_P2_30Jul18_006	CG	CG	98	CG	1.00
BT18-029_P2_31Jul18_009	HD	HD	100	CG	0.99
BT18-030_P2_31Jul18_012	L	L	100	L	0.81
BT18-031_P2_31Jul18_003	PS	CG	88	CG	1.00
BT18-032_P2_31Jul18_006	CNS	CG	100	CG	1.00
BT18-033_P2_01Aug18_009	L	L	100	L	0.99
BT18-034_P2_01Aug18_012	CNB	CNB	100	CNS	0.04
BT18-035_P2_01Aug18_003	HD	HD	100	HD	0.80

Table 6. Continued.

Run name	True ID	Predicted ID			
		PINS+ v6.5.2		ANN	
		ID	Score	ID	Value
BT18-036_P2_01Aug18_006	PS	CK	99	HE	0.35
BT18-037_P2_02Aug18_009	CNS	CNS	100	CK	0.50
BT18-038_P2_02Aug18_012	CK	CG	100	CG	0.49
BT18-039_P2_02Aug18_003	GB	GB	100	WP	0.67
BT18-040_P2_02Aug18_006	GA	GA	100	HE	0.83
BT18-041_P2_03Aug18_009	FS	FS	100	FS	0.79
BT18-042_P2_03Aug18_012	PS	CG	100	CG	1.00
BT18-043_P2_03Aug18_003	GB	GB	100	WP	0.08
BT18-044_P2_03Aug18_006	VX	VX	100	VX	0.86
BT18-045_P2_06Aug18_009	CK	CG	100	CG	1.00
BT18-046_P2_06Aug18_012	AC	AC/HE	100	Sand	0.25
BT18-047_P2_06Aug18_003	HD	HD	100	HD	0.69
BT18-048_P2_06Aug18_006	GA	GA	100	HE	0.48
BT18-049_P2_07Aug18_009	HN3	HN	52	HN	0.89
BT18-050_P2_07Aug18_012	L	L	90	CG	1.00
BT18-051_P2_07Aug18_003	CN	CN	89	CG	0.63
BT18-052_P2_07Aug18_006	VX	VX	100	HE	0.43
BT18-053_P22_08Aug18_006	HN3	CNB	39	CNS	0.99
BT18-054_P22_08Aug18_009	PS	CG	100	Sand	0.64
BT18-055_P19_08Aug18_003	VX	VX	100	VX	1.00
BT18-056_P22_08Aug18_003	GA	GA	100	VX	0.92
BT18-057_P22_09Aug18_009	AC	AC/HE	100	HE	1.00
BT18-058_P22_09Aug18_012	HD	HD	100	CG	0.89
BT18-059_P22_09Aug18_003	CNS	CNS	100	CG	1.00
BT18-060_P22_09Aug18_006	L	L	100	L	0.91
BT18-061_P22_23Aug18_009	FS	FS	100	FS	0.96
BT18-062_P22_23Aug18_012	CK	CG	100	CG	1.00
BT18-063_P22_23Aug18_003	HN3	HN	45	CNS	0.98
BT18-064_P22_23Aug18_006	CNS	CG	100	CG	1.00
BT18-065_P22_14Aug18_009	CNB	CNB	57	CNS	0.97
BT18-066_P22_14Aug18_012	BR2	BR2	99	Sand	0.28
BT18-067_P22_14Aug18_003	TNT	HE	100	HE	0.98
BT18-068_P22_14Aug18_006	Empty	Sand	73	Sand	1.00
BT18-069_P22_15Aug18_009	WL	WL	100	Sand	0.14
BT18-070_P22_15Aug18_012	AC	AC/HE	100	HE	1.00
BT18-071_P22_15Aug18_003	CNS	CNS	100	HN	0.10
BT18-072_P22_15Aug18_006	BR2	BR2	80	Sand	0.48
BT18-073_P22_16Aug18_009	CNB	CNB	50	CNS	1.00

Table 6. Continued.

Run name	True ID	Predicted ID			
		PINS+ v6.5.2		ANN	
		ID	Score	ID	Value
BT18-074_P22_16Aug18_012	PS	CG	100	CG	1.00
BT18-075_P22_16Aug18_003	CN	CN	100	CG	0.08
BT18-076_P22_16Aug18_006	Sand	Sand	100	Sand	1.00
BT18-077_P22_17Aug18_009	WL	WL	100	CG	0.97
BT18-078_P22_17Aug18_012	CNB	CNB	89	CG	0.50
BT18-079_P22_17Aug18_003	Sand	Sand	100	Sand	1.00
BT18-080_P22_17Aug18_006	TH3	Unknown	0	CG	1.00
BT18-081_P22_20Aug18_009	L	L	100	L	0.94
BT18-082_P22_20Aug18_012	BR2	BR2	75	L	0.47
BT18-083_P22_20Aug18_003	HE	HE	100	HE	0.75
BT18-084_P22_20Aug18_006	Empty	Unknown	0	Sand	0.96
BT18-085_P22_21Aug18_009	TH3	Unknown	0	Sand	0.89
BT18-086_P22_21Aug18_012	CN	CN	89	CNS	0.17
BT18-087_P22_21Aug18_003	Empty	Sand	53	CG	0.99
BT18-088_P22_21Aug18_006	FS	FS	100	FS	0.96
BT18-089_P22_24Aug18_009	CNB	CNB	99	HN	0.53
BT18-090_P22_24Aug18_012	HE	HE	100	HE	0.89
Bt18-091_P22_24Aug18_003	Sand	Sand	100	Sand	1.00
BT18-092_P22_24Aug18_006	TH3	Unknown	0	Sand	0.61

Table 7. Summary of the confusion matrix for the ANN model (see **Figure 8**).

ID	TP	FN	FP	TN	$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$	$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}}$	$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}}$	$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$	$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$	$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$
CG	7	0	19	46	1.00	0.00	0.29	0.71	0.74	0.27
CK	0	7	1	64	0.00	1.00	0.02	0.98	0.89	0.00
CNB	0	5	0	67	0.00	1.00	0.00	1.00	0.93	-
CNS	0	7	5	60	0.00	1.00	0.08	0.92	0.83	0.00
FS	4	0	0	68	1.00	0.00	0.00	1.00	1.00	1.00
HD	2	3	0	67	0.40	0.60	0.00	1.00	0.96	1.00
HN	2	3	2	65	0.40	0.60	0.03	0.97	0.93	0.50
L	4	1	1	66	0.80	0.20	0.01	0.99	0.97	0.80
BR2	0	3	0	69	0.00	1.00	0.00	1.00	0.96	N/A
WL	0	3	0	69	0.00	1.00	0.00	1.00	0.96	N/A
VX	2	3	1	66	0.40	0.60	0.01	0.99	0.94	0.67
GA/GB	0	10	2	60	0.00	1.00	0.03	0.97	0.83	0.00
WP	0	0	6	66	N/A	N/A	0.08	0.92	0.92	0.00
HE	3	0	4	65	1.00	0.00	0.06	0.94	0.94	0.43
Sand	3	0	4	65	1.00	0.00	0.06	0.94	0.94	0.43
Overall	27	45	45	963	0.38	0.63	0.04	0.96	0.92	0.38

REFERENCES

[1] A.J. Caffrey, et al., "Chemical Warfare Agent and High Explosive Identification by Spectroscopy of Neutron-induced Gamma-Rays", IEEE Trans. Nucl. Sci, 39, pp 1422-1426, (1992).

[2] ORTEC's PINS3-CW Brochure, http://www.ortec-online.com/products/nuclear-security-and-safeguards/chemical-weapons-identification/pins3-cw

[3] D. Lee, et al., "Application of Multivariate Data Analysis Techniques for the Portable Isotopic Neutron Spectroscopy System", IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), Atlanta (2017)

[4] https://ir.unimas.my/id/eprint/7812/2/Octave%20MLP%20Neural%20Networks.pdf

[5] https://octave.sourceforge.io/nnet/overview.html

APPENDIX A: ANN_TEST.M

```
## Copyright (C) 2007 Michel D. Schmid  <michaelschmid@users.sourceforge.net>
##
##
## This program is free software; you can redistribute it and/or modify it
## under the terms of the GNU General Public License as published by
## the Free Software Foundation; either version 2, or (at your option)
## any later version.
##
## This program is distributed in the hope that it will be useful, but
## WITHOUT ANY WARRANTY; without even the implied warranty of
## MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
## General Public License for more details.
##
## You should have received a copy of the GNU General Public License
## along with this program; see the file COPYING.  If not, see
## <http://www.gnu.org/licenses/>.

## author: msd
##  ---- 07/2019 The original script was modified and renamed by D. Lee for this study

## load data of 176 rows
# Input file contains 37 columns. The first 16 columns are input and the last 21 columns are output
mData = load("-ascii", "cr_ann_data_compact.txt");
[nRows, nColumns] = size(mData);
mOutput = mData(:, end-20:end);
mInput = mData(:, 1:end-21);

## now prepare data column-wise
mInput = mInput';
mOutput = mOutput';

# now split the data matrix in 3 pieces, train data, test data and validate data
# last 23 demo spectra were used for the holdout test data
nTestSets = 23;

mTestInput = mInput(:, end-(nTestSets-1):end);
mTestOutput = mOutput(:, end-(nTestSets-1):end);
mInput(: ,end-(nTestSets-1):end) = [];
mOutput(:, end-(nTestSets-1):end) = [];

# True answers for the holdout test data
id = [2 3 4 9 9 10 12 12 5 5 5 14 14 19 19 18 18 20 20 21 21 21 21];

## define the number of neutrons in the hidden and the output layers
nHiddenNeurons = 17;
nOutputNeurons = 21;
```

```
# change this to lower target MSE value
target_mse = 0.002;
## start iterations till two conditions are met
do
      nTrainSets = nRows-nTestSets;
      nValiSets = floor(nTrainSets/3.0);
      order = randperm(nTrainSets);
      mInput(:, order) = mInput;
      mOutput(:, order) = mOutput;
      mInputClone = mInput;
      mOutputClone = mOutput;

      mValiInput = mInputClone(:, 1:nValiSets);
      mValliOutput = mOutputClone(:, 1:nValiSets);
      mInputClone(:, 1:nValiSets) = [];
      mOutputClone(:, 1:nValiSets) = [];
      mTrainInput = mInputClone(:, 1:end);
      mTrainOutput = mOutputClone(:,1:end);

      # standardize inputs
      [mTrainInputN,cMeanInput,cStdInput] = prestd(mTrainInput);

      # define the max and min inputs for each row
      mMinMaxElements = min_max(mTrainInputN);

     # define an ANN model structure
     MLPnet = newff(mMinMaxElements, [nHiddenNeurons OutputNeurons], {"logsig", “logsig"},
     "trainlm", "learngdm", "mse");

      ## standardize the validate data
      VV.P = mValiInput;
      VV.T = mValliOutput;
      VV.P = trastd(VV.P, cMeanInput, cStdInput);

      ## train the network
      MLPnet.trainParam.show = NaN;
      net.trainParam.epochs = 100;
      nnet.trainParam.goal = 0.00;

      [net] = train(MLPnet, mTrainInputN, mTrainOutput, [], [], VV);

      ## make preparations for net test and test MLPnet
      ## standardize input & output test data
      [mTestInputN] = trastd(mTestInput, cMeanInput, cStdInput);

      #     % simulate net
      [trainOut] = sim(net, mTrainInputN);
      [simOut] = sim(net,mTestInputN);
      [v, idx] = max(simOut(:, :));

      # calculate MSE as a measure of performance
```

```
      perf_test = sum(sum((mTestOutput-simOut).^2))/prod(size(simOut));
until (sum(abs(id-idx)) < 1 & (perf_test < target_mse))


display(perf_test);

# save ANN results of test cases to a CSV file
dlmwrite('summary.csv', simOut, ",");
# save cMeanInput vector
dlmwrite('summary.csv', cMeanInput, ",", "-append", "roffset", 1);
# save cStdInput vector
dlmwrite('summary.csv', cStdInput, ",", "-append", "roffset", 1);
# save IW matrix
dlmwrite('summary.csv', net.IW{1}, ",", "-append", "roffset", 1);
# save LW matrix
dlmwrite('summary.csv', net.LW{2}, ",", "-append", "roffset", 1);
# save IB vector
dlmwrite('summary.csv', net.b{1}, ",", "-append", "roffset", 1);
# save LB vector
dlmwrite('summary.csv', net.b{2}, ",", "-append","roffset", 1);
```