Application of Open Set Recognition Method for Neural Network-based Models in Nuclear Power Plant

Seung Geun Kim[a*], Young Ho Chae[b], Seoryong Koo[b]
[a]Applied Artificial Intelligence Section, Korea Atomic Energy Research Institute, Daejeon, Republic of Korea
[b]Advanced Instrumentation & Control Research Section, Korea Atomic Energy Research Institute, Daejeon, Republic of Korea

Abstract: With the advancement of artificial intelligence (AI) technology, various models have been proposed for solving problems in the nuclear field. One representative problem in the nuclear field is event/accident diagnosis, and numerous classification models have been developed based on event/accident data acquired from simulations. However, in actual nuclear power plants (NPPs), there may be situations where the event is ambiguous to classify, or the event is unknown and entirely new. In these cases, most previously developed models classify such situations as one of the classes considered during their development, potentially leading to inappropriate diagnoses and the establishment of mitigation strategies.

Moreover, based on NPPs' safety strategies, which rely on diversity, independence, and redundancy, AI models should complement human operators and ensure the safety of NPPs. In this regard, AI models should be capable of determining whether training for a given situation has been conducted and transferring decision authority to human operators when the model is incapable of handling the given situation.

In this study, to provide the ability to detect untrained situations for the model, several open-set recognition methods are adopted for neural network-based models in the nuclear field. To conduct experiments while considering the characteristics of AI models in the nuclear field, a neural network-based accident diagnosis model is developed. During training, a specific accident class is neglected from the training dataset, and it is checked whether the applied open-set recognition methods are capable of detecting untrained scenarios. The experiment results have revealed that the applied open-set recognition methods are capable of detecting untrained scenarios with acceptable performances.

Keywords: Neural Network, Classification Model, Event/Accident Diagnosis, Open Set Recognition,

1. INTRODUCTION

For safe and efficient operation of nuclear power plants (NPPs), operators conduct various tasks such as monitoring, maintenance, control, and diagnosis. Among these tasks, event/accident diagnosis is conducted for the situation assessment when the plant is under abnormal or emergency condition. Accurate event/accident diagnosis is important for securing the safety of NPPs since it is essential for planning proper mitigation strategies. As artificial neural network-based artificial intelligence (AI) technology is showing outstanding performance across the various fields, many AI models have been proposed for solving problems in nuclear field [1-6], and numerous diagnosis models have been developed for event/accident diagnosis [5,6].

Most of existing classification models deduce output among the classes that are included in the training data. However, there may be situations in actual NPPs, where the event is ambiguous to classify, or the event is unknown and entirely new. In these cases, conventional diagnosis models may deduce wrong diagnosis result with high confidence for the input irrelevant to the trained classes. This may induce confusion to the operators and potentially leading to the establishment of inappropriate mitigation strategies.

Moreover, to follow the basic safety strategies of NPPs including diversity, independence, and redundancy, AI models should complement human operators. In this regard, AI models should be capable of determining whether the training for a given situation has been conducted and transferring decision authority to human operators when the model is incapable of handling the given situation.

Open set recognition (OSR) is one of the research field related to AI, which aims the identification of untrained classes. In this study, the OpenMax method [7] – which is one of the representative OSR method – was applied to the neural network-based NPP accident diagnosis model to check whether the method is capable of identifying inputs correspond to untrained classes. For the experiments, an NPP accident diagnosis model was developed with neglecting specific accident class from the training dataset and OpenMax method was applied to identify data corresponding to untrained accident class.

The rest of the paper is organized as follows. In chapter 2, brief explanations on OSR and OpenMax are provided. In chapter 3, processes of the experiments and corresponding results are presented. Chapter 4 summarizes and concludes the paper.

2. METHOD: OPENMAX

During the development of various AI models, it is generally assumed that the data used during the development (training, validation, and testing) is in the same input space with the data that will be received after the application. It is trivial that the performance of the model tends to be higher when the training data covers the larger portion of the entire input space, since most of AI models are much better at the interpolation, rather than the extrapolation. However, aforementioned assumption may not valid in real-world applications. For example, in nuclear field, most of AI models are developed based on simulation data. However, there may exists discrepancy between simulation data and actual data owing to the limitations of simulators and various factors of uncertainties. Accordingly, data used during the model development and the data that will be received after the application may not share the same input space.

Most of conventional classification models deduce output among the trained classes for every given inputs. That is, if the model is trained to classify class A, B, C, the model will always deduce one of these classes as output even when the given input is correspond to untrained class D. This problem is more emphasized for neural network-based classification models, since they tend to deduce highly confident output with near 100% classification probability for specific class. The over-confident wrong answer for untrained input may induce confusion to the users, and may result in severe consequences if the model is applied to safety-critical systems such as NPPs.

To consider the untrained class problem, various Open set recognition (OSR) methods have been proposed. OSR methods change the paradigm of classification problem from the closed-set classification to the open-set classification by granting model the ability to identify data that cannot be properly classified as one of the trained classes.

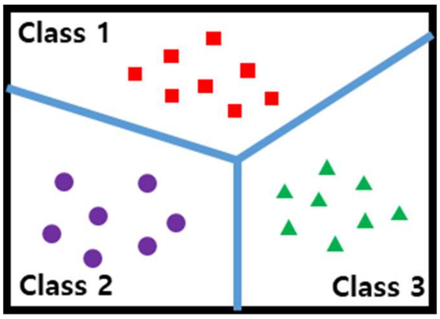


Figure 1. Schematic of the closed-set classification concept

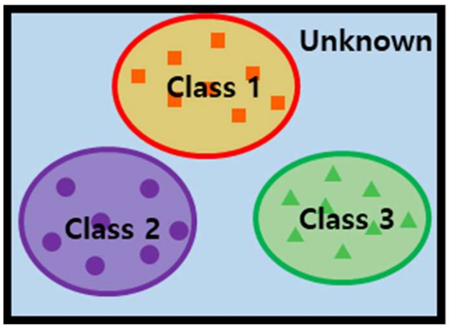


Figure 2. Schematic of the open-set classification concept

The OpenMax [7] is one of the representative OSR method with intuitive concepts, and can be applied to the trained classification models without the changes of model's structure and parameters. The OpenMax method is applied through preparation step and execution step. In preparation step, the standards for OSR is prepared based on the activation vector (AV) profiles of training data. In execution step, OSR is conducted for given input data based on the standards deduced in the preparation step. Here, AV implies the set of node values of the output layer, before the activation function calculation.

Preparation step consists of data sorting, AV profiling, and extreme value fitting sub-steps. In data sorting sub-step, only correctly classified training data are selected and used in further sub-steps.

In AV profiling sub-step, mean AV and mean distance between mean AV and AVs of each selected data are calculated for each trained class. In this study, Euclidean-cosine distance is used for distance calculation, which considers both Euclidean distance and cosine similarity. Euclidean-cosine distance can be calculated as follows.

$$EC\_distance = Euclidean\_distance \times (1 - Cosine\_similarity) \tag{1}$$

$$Euclidean\_distance = \|(V_1 - V_2)\|_2 \tag{2}$$

$$Cosine\_similarity = \|V_1\|_2 \cdot \|V_2\|_2 \tag{3}$$

Where $EC\_distance$ is Euclidean-cosine distance, V_1 and V_2 are given AVs and $\|\cdot\|_2$ implies the L2-norm.

In extreme value fitting sub-step, distribution of the distances between AVs of every selected data and mean AV of corresponding class is estimated based on extreme value theory [8]. During the distribution estimation, hyperparameter η (eta) should be determined which sets the number of values that are treated as 'extreme' value. In this study, Weibull distribution was used for distribution estimation. Probability density function of Weibull distribution can be represented as follows.

$$f(x;\lambda,k) = \begin{cases} \frac{k}{\lambda}(\frac{x}{\lambda})^{k-1}e^{-(x/\lambda)^k}, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{4}$$

Where k and λ are positive shape and scale parameter of the distribution, respectively.

Execution step consists of AV calculation, probability calculation, and AV revision sub-steps. In AV calculation sub-step, AV of the given input and its distances from mean AVs of every trained classes are calculated.

In probability calculation sub-step, probabilities that the distance to be lower than the distances between AV of the given input and mean AVs for every trained classes. If the AV of the given input is similar to the mean AV of specific class, then the probability for that class would be low, and vice versa. The calculated probability for class n is denoted as ω_n.

In AV revision step, based on the probability values ω_n, AV is revised and classification probabilities are calculated based on revised AV. Elements of AV are revised based on the elements of original AV and calculated probability values as follows.

$$v'_n = (1 - \omega_n) \times v_n \tag{5}$$

$$v'_0 = \textstyle\sum_n(\omega_n \times v_n) \tag{6}$$

$$n = 1, 2, \dots N \text{ (N: number of trained classes)}$$

Where v_n and v'_n represents the element correspond to the trained class n before and after the revision, respectively. v'_0 represents the added element correspond to the untrained class after the revision.

After the revision of AV, classification probabilities can be calculated based on Softmax function. Classification probability for class n and untrained class can be represented as follows.

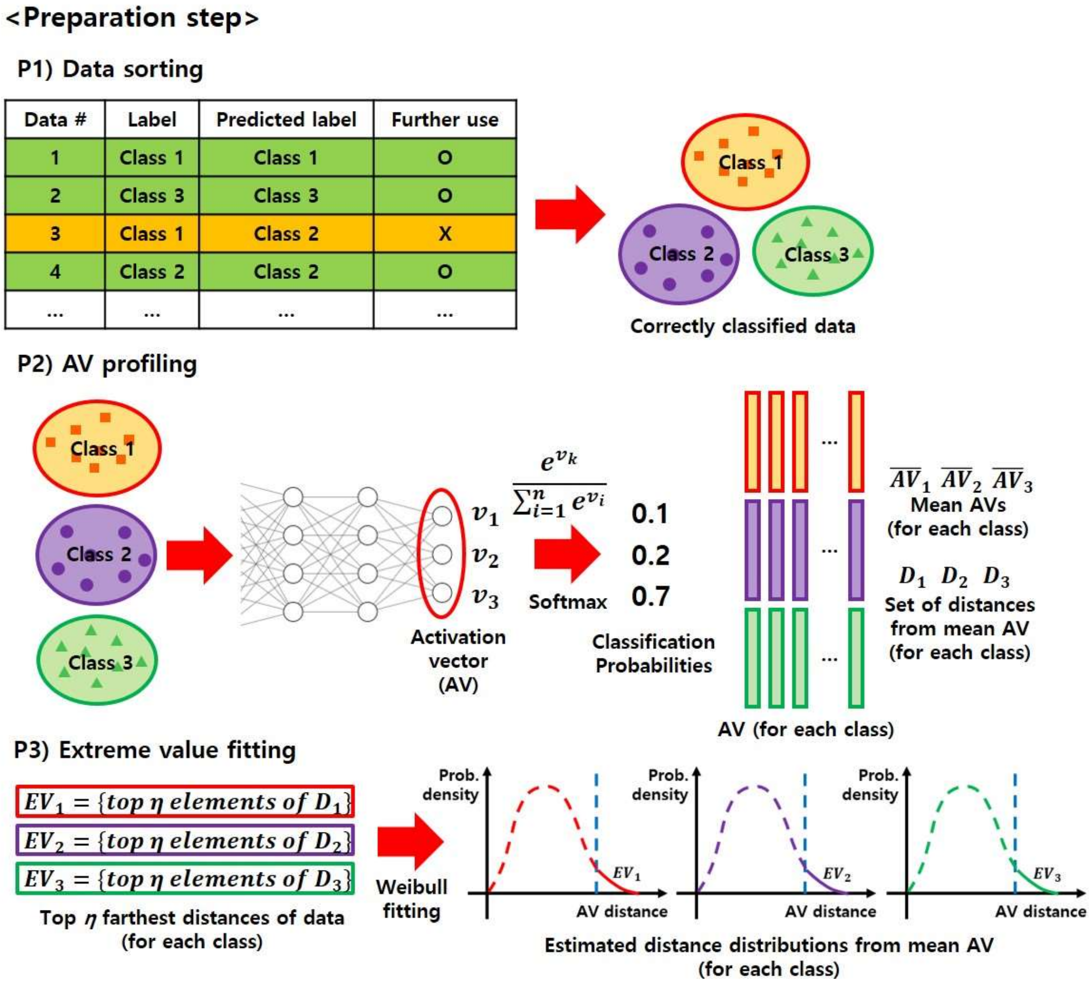
$$Pr(k) = \frac{\exp(v_n')}{\exp(v_0')+\sum_n \exp(v_n')} \tag{7}$$

$$Pr(untrained) = \frac{\exp(v_0')}{\exp(v_0')+\sum_n \exp(v_i')} \tag{8}$$

$n = 1, 2, \dots N$ (N: number of trained classes)

Where $Pr(n)$ is the revised classification probability for class n, and $Pr(untrained)$ is the classification probability for untrained class.

Figure 3 and 4 are schematics of the preparation step and execution step of the OpenMax method, respectively.



Data #	Label	Predicted label	Further use
1	Class 1	Class 1	O
2	Class 3	Class 3	O
3	Class 1	Class 2	X
4	Class 2	Class 2	O
…	…	…	…

<Preparation step>
P1) Data sorting
Class 1
Class 2
Class 3
Correctly classified data
P2) AV profiling
Class 1
Class 2
Class 3
v_1
v_2
v_3
Activation vector (AV)
$\frac{e^{v_k}}{\sum_{i=1}^{n} e^{v_i}}$
Softmax
0.1
0.2
0.7
Classification Probabilities
…
AV (for each class)
$\overline{AV}_1$ $\overline{AV}_2$ $\overline{AV}_3$
Mean AVs (for each class)
D_1 D_2 D_3
Set of distances from mean AV (for each class)
P3) Extreme value fitting
$EV_1 = \{top\ \eta\ elements\ of\ D_1\}$
$EV_2 = \{top\ \eta\ elements\ of\ D_2\}$
$EV_3 = \{top\ \eta\ elements\ of\ D_3\}$
Top η farthest distances of data (for each class)
Weibull fitting
Prob. density
EV_1
EV_2
EV_3
AV distance
Estimated distance distributions from mean AV (for each class)

Figure 3. Schematic of the preparation step of the OpenMax method

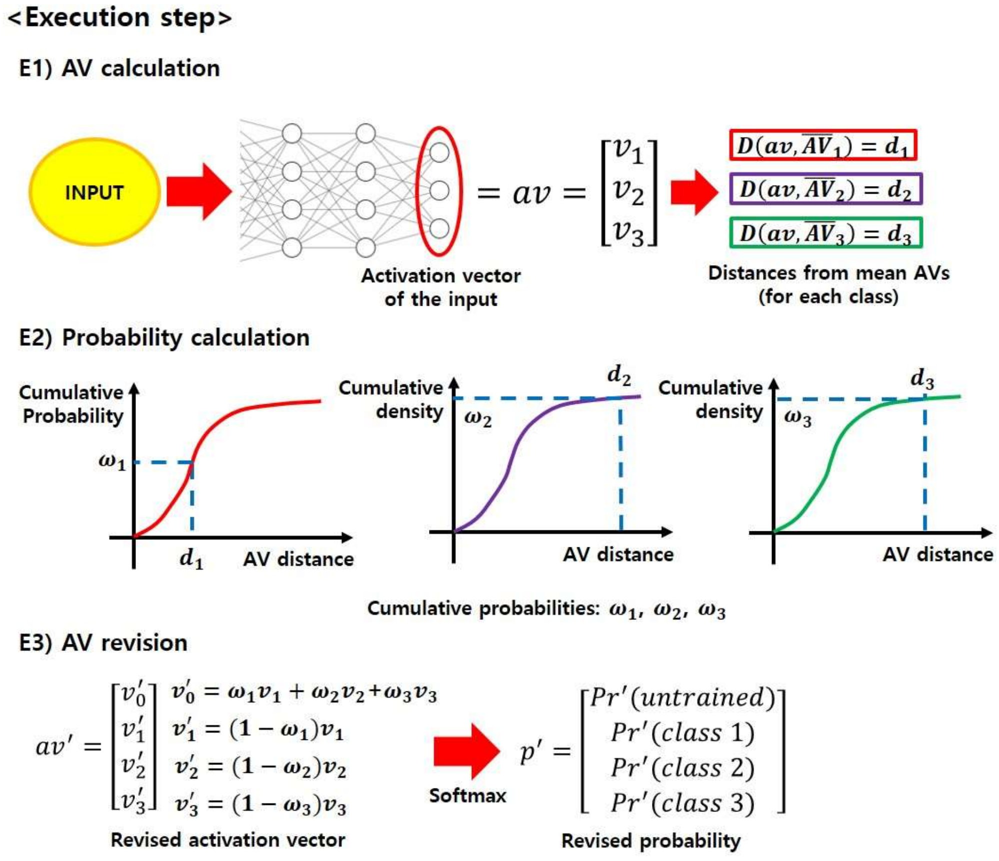

<Execution step>
E1) AV calculation
INPUT
$= av = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$
Activation vector of the input
$D(av, \overline{AV}_1) = d_1$
$D(av, \overline{AV}_2) = d_2$
$D(av, \overline{AV}_3) = d_3$
Distances from mean AVs (for each class)
E2) Probability calculation
Cumulative Probability
ω_1
d_1
AV distance
Cumulative density
ω_2
d_2
AV distance
Cumulative density
ω_3
d_3
AV distance
Cumulative probabilities: ω_1, ω_2, ω_3
E3) AV revision
$av' = \begin{bmatrix} v'_0 \\ v'_1 \\ v'_2 \\ v'_3 \end{bmatrix}$
$v'_0 = \omega_1 v_1 + \omega_2 v_2 + \omega_3 v_3$
$v'_1 = (1-\omega_1)v_1$
$v'_2 = (1-\omega_2)v_2$
$v'_3 = (1-\omega_3)v_3$
Revised activation vector
Softmax
$p' = \begin{bmatrix} Pr'(untrained) \\ Pr'(class\ 1) \\ Pr'(class\ 2) \\ Pr'(class\ 3) \end{bmatrix}$
Revised probability

Figure 4. Schematic of the execution step of the OpenMax method

3. EXPERIMENTS

3.1. Data Acquisition and Preprocessing

For the development of accident diagnosis model, data was acquired from the simulation. Compact nuclear simulator (CNS) developed in Korea atomic energy research institute (KAERI) [9] was used. Reference plant of CNS is Westinghouse 3-loop 900 MWe pressurized water reactor.

During the simulation, three kinds of accident scenarios were considered including loss of coolant accident (LOCA), steam generator tube rupture (SGTR), and main steam line break (MSLB). For the diversity, variations on tube break sizes and break locations were applied. Simulation was conducted for 20 minutes (plant time) starting from the reactor trip occurred by malfunction infusion, and 19 kinds of instrumentation signals were acquired that are considered as important for the accident diagnosis.

For the preprocessing, minimum-maximum (min-max) normalization was applied to set the range of all values of variables between 0 and 1, and unit data with 5 minutes length was generated from the data between 5 to 15 minutes (plant time) with 10 seconds interval.

As a result, totally 930, 465, and 1116 unit data were generated for the LOCA, SGTR, and MSLB accident scenarios, respectively. Among them, 70%, 15%, and 15% of unit data was used for training, validation, and testing, respectively.

Table 1. Considered accident scenarios and their variations

Accident type	Break sizes (cm^2)	Break loops	Break locations
LOCA	15, 20, 25, 30, 35	Loop #1, #2, #3	Cold leg, hot leg
SGTR	4, 8, 12, 16, 20	Loop #1, #2, #3	-
MSLB	500, 600, 700, 800, 900, 1000	Loop #1, #2, #3	Inside of the containment, Outside of the containment

Table 2. Acquired variables and their units

Variables	Units
Cold leg temperature (loops 1/2/3)	℃
Pressurizer pressure, wide range	kg/cm2
Pressurizer level	%
Steam generator pressure (loops 1/2/3)	kg/cm2
Steam generator level, wide range (loops 1/2/3)	%
Feedwater line flowrate (loops 1/2/3)	ton/hr
Steam line flowrate (loops 1/2/3)	ton/hr
Containment radiation	mRem/hr
Secondary system radiation	μCi/cc

3.2. Model Development

After the data acquisition and preprocessing, a neural network-based accident diagnosis model was developed. For simplicity, model was developed to have six fully-connected layers only.

To evenly consider the untrained class, models were separately developed with changing the neglected accident scenario from the training data. The ‘Case 1’ models have developed with neglecting MSLB data; the ‘Case 2’ models have developed with neglecting SGTR data; and the ‘Case 3’ models have developed with neglecting LOCA data.

In addition, models were also separately developed with changing the activation functions to consider the effects of the kind of applied activation function. Except the Softmax activation function at the output layer, the ‘ReLU’ models were developed with applying rectified linear unit (ReLU) activation functions; the ‘ELU’ models were developed with applying exponential linear unit (ELU) activation functions; and the ‘tanh’ models were developed with applying hyperbolic tangent (tanh) activation functions.

As a result, totally nine models were developed with changing the neglected accident scenario from the training data, and the type of activation function. Each models are denoted as “ReLU/Case 1”. Every models have achieved 100% accuracy for classifying trained accident scenarios.

$$\mathrm{ReLU}(x) = \begin{cases} x & if\ x > 0 \\ 0 & if\ x \le 0 \end{cases} \tag{9}$$

$$\mathrm{ELU}(x) = \begin{cases} x & if\ x > 0 \\ \alpha(\exp(x) - 1) & if\ x \le 0 \end{cases} \tag{10}$$

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{11}$$

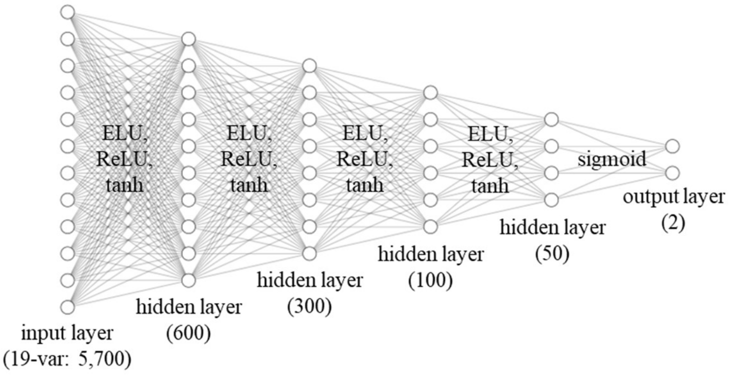


Figure 5. Schematic of the structure of developed accident diagnosis model

3.3. OpenMax Application

For the developed accident diagnosis models, the OpenMax method was applied for OSR. The experiments were repeatedly conducted with changing the hyperparameter η. Table 3 presents the best results with the highest mean accuracy.

Table 3. Best results with the highest mean accuracy

Cases	Act. Fct.	LOCA	SGTR	MSLB
Case 1 (Untrained: MSLB)	ELU	98.92% (-1.08%)	100.00% (-0%)	99.06%
	ReLU	100.00% (-0%)	98.92% (-1.08%)	37.77%
	tanh	99.28% (-0.72%)	98.57% (-1.43%)	89.38%
Case 2 (Untrained: SGTR)	ELU	100.00% (-0%)	100.00%	100.00% (-0%)
	ReLU	100.00% (-0%)	100.00%	100.00% (-0%)
	tanh	100.00% (-0%)	100.00%	100.00% (-0%)
Case 3 (Untrained: LOCA)	ELU	100.00%	99.28% (-0.72%)	98.66% (-1.34%)
	ReLU	100.00%	100.00% (-0%)	100.00% (-0%)
	tanh	100.00%	98.57% (-1.43%)	100.00% (-0%)

The experiments revealed that the application of OpenMax method enables the identification of untrained class with over 99% accuracy, except for “ReLU/Case 1” and “tanh/Case 1” models. Results were also shown that if OpenMax method is applied properly with adjusting hyperparameter η, its negative affect to the classification performance for trained classes can be minimalized.

Since hyperparameter η determines the number of ‘extreme’ value while conducting extreme value fitting sub-step, higher value of η tends to make the estimated distribution to be more emphasized for extreme values. Accordingly, OpenMax application with higher η value generally result in increased OSR performance, while it may deteriorate the classification performance for trained classes. Therefore, it is necessary to consider the trade-off relation between OSR performance and classification performance for trained classes, and find optimal hyperparameter value for practical application.

In most cases, determining optimal hyperparameter value would be difficult since data for unexpected situations are generally unavailable. As an alternative, sub-optimal η value can be found by conducting experiments similar to the experiments conducted in this study that assume specific class as untrained class.

Regarding the kind of applied activation function, the models correspond to “Case 2” and “Case 3” have shown similar performances regardless of the kind of applied activation function. However, for the models correspond to “Case 1”, model with ELU activation function have shown best performance, followed by models with tanh and ReLU activation functions. Especially, “ReLU/Case 1” model has shown poor OSR performance.

Although ELU, ReLU, and tanh activation functions are widely applied for models dealing with time-series data, the experiments revealed that the OSR performance can be varied drastically according to the kind of applied activation function. This result emphasizes the importance of comparison between various model configurations for better OSR performance, including the kind of applied activation function.

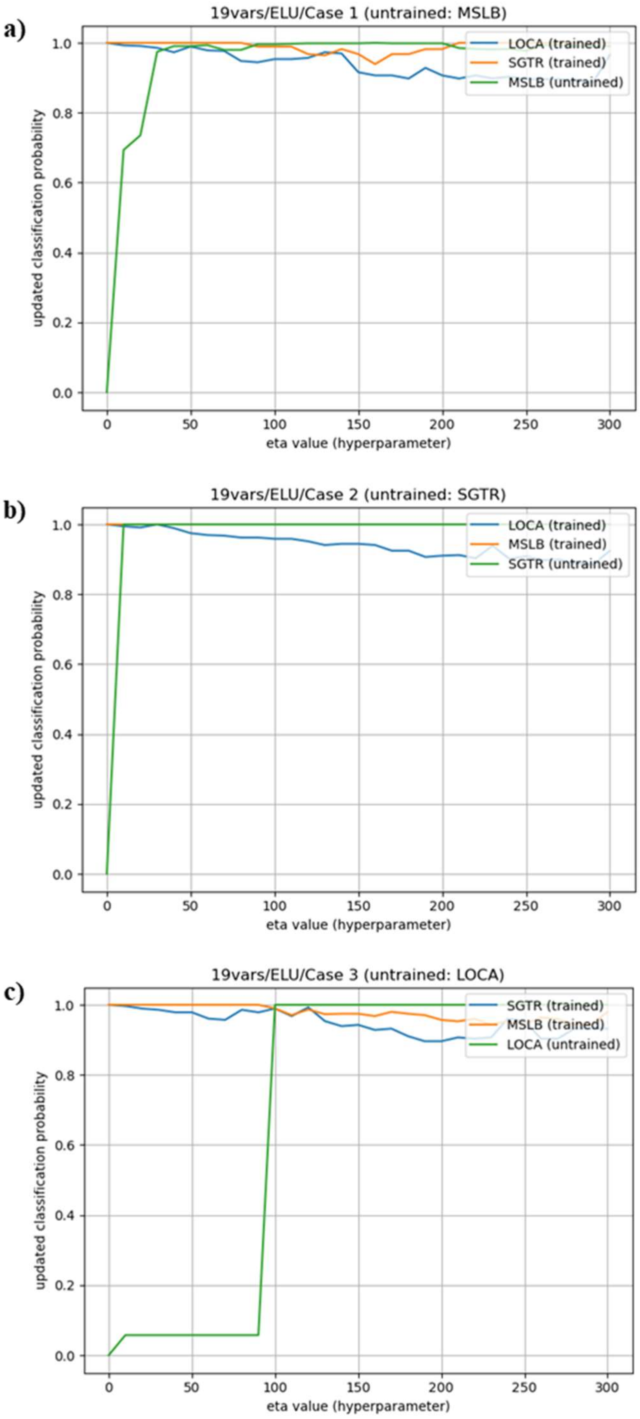


Figure 6. Examples of accuracy trends according to the hyperparameter η with changing the neglected accident cases (ELU/*). **a)** trends of “ELU/Case 1” model, **b)** trends of “ELU/Case 2” model, **c)** trends of “ELU/Case 3” model.

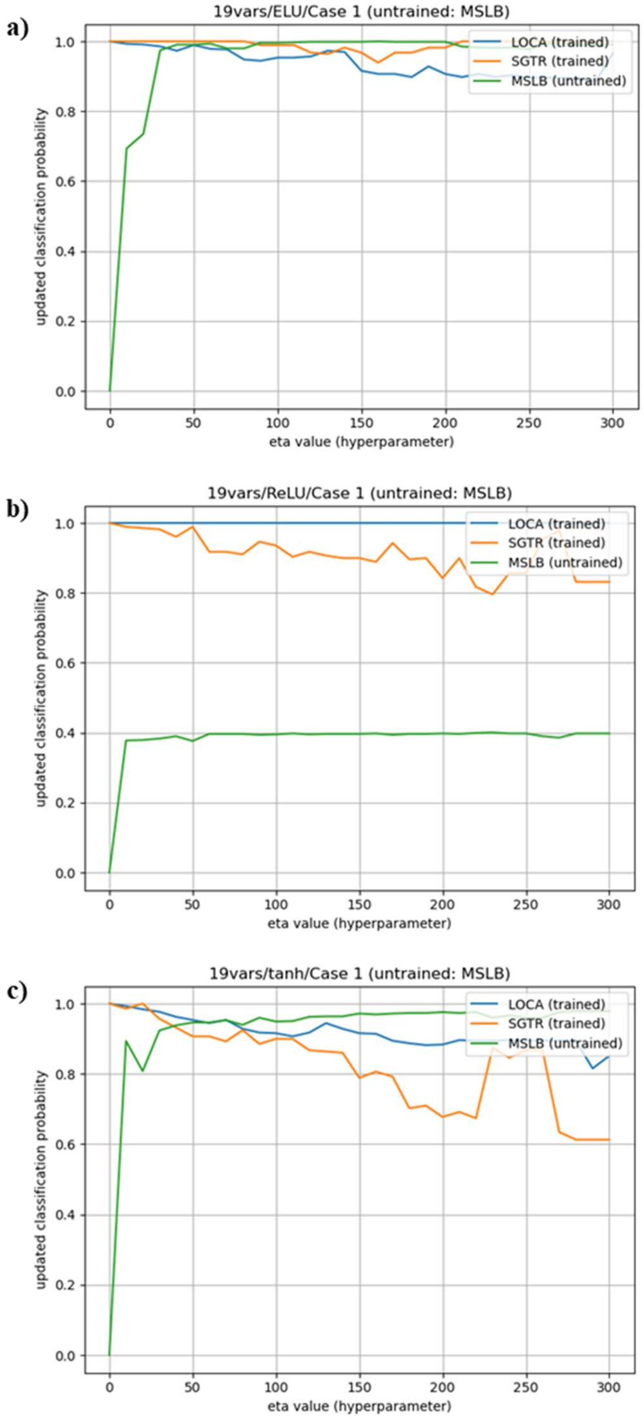


Figure 7. Examples of accuracy trends according to the hyperparameter η with changing the applied activation functions (*/Case 1). **a)** trends of “ELU/Case 1” model, **b)** trends of “ReLU/Case 1” model, **c)** trends of “tanh/Case 1” model.

4. CONCLUSION

As conventional neural-network based NPP event/accident diagnosis models always deduce output among the trained classes, they may deduce inappropriate output when the input correspond to untrained class is given. In this study, OpenMax method was adopted as one of the representative OSR method to grant model the ability for identifying input data that correspond to untrained class. The experiments were conducted based on accident diagnosis model, developed by using simulation data acquired from CNS. Several accident diagnosis models were developed with changing the untrained class by neglecting specific class from the training data, and the OpenMax method was applied for identifying inputs correspond to neglected class. The experiment results have shown that if hyperparameter tuning is properly conducted, the OpenMax method is able to conduct OSR accurately with minimalized classification performance deterioration for trained classes. Furthermore, from the several results that shown relatively poor OSR performance, it was revealed that the type of activation function may heavily affect OSR performance of the OpenMax method. Therefore, it is necessary to conduct experiments for finding optimal hyperparameter and comparing various model structures including the type of activation function, to achieve high OSR performance in future applications of the OpenMax method.

Although this study has found the significance of the activation function and hyperparameter value on the model performance, further studies are necessary to find the reasons for the varying performances. Therefore, the OpenMax method's OSR performance for the models with more complicated structures will be investigated as future works. In addition, the comparative studies on the OpenMax method and other OSR methods will be conducted.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) (No. RS-2022-00144150)

References

[1] Choi Y., Yoon G., and Kim J., Unsupervised learning algorithm for signal validation in emergency situations at nuclear power plants, Nuclear Engineering and Technology, 54.4, 1230-1244, 2022.

[2] Kim S. G., Chae Y. H., and Seong P. H., Development of a generative-adversarial-network-based signal reconstruction method for nuclear power plants, Annals of Nuclear Energy, 142, 107140, 2020.

[3] Ryu S., et al., Probabilistic deep learning model as a tool for supporting the fast simulation of a thermal-hydraulic code, Expert Systems with Applications, 200, 116966, 2022.

[4] Lee D., Arigi A. M., and Kim J., Algorithm for autonomous power-increase operation using deep reinforcement learning and a rule-based system, IEEE Access, 8, 196727-196746, 2020.

[5] Chae Y. H., et al., Graph neural network based multiple accident diagnosis in nuclear power plants: Data optimization to represent the system configuration, Nuclear Engineering and Technology, 54.8, 2859-2870, 2022.

[6] Shin J. H., et al., Approach to diagnosing multiple abnormal events with single-event training data, Nuclear Engineering and Technology, 56.2, 558-567, 2024.

[7] Bendale A. and Boult T. E., Towards open set deep networks, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[8] Coles, Stuart, et al., An introduction to statistical modeling of extreme values, Vol. 208, London: Springer, 2001.

[9] Kwon K. C., et al., Compact nuclear simulator and its upgrade plan, 1997.