Transactions of the Korean Nuclear Society Spring Meeting
Jeju, Korea, May 9-10, 2024

Two-stage prediction model of nuclear power plant parameter trends for preventing significantly erroneous predictions

Donghee Jung, Junyong Bae, Seung Jun Lee*
Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan, 44911
**Corresponding author: sjlee420@unist.ac.kr*

Keywords : Nuclear power plant, deep learning, trends prediction, operator action

1. Introduction

Nuclear power plant (NPP) operators should check the plant status from indicators such as various sensors and alarms. In emergency situations, operators must identify rapidly changing NPP parameters, make appropriate diagnoses according to procedures, and conduct appropriate mitigation actions. However, if inappropriate actions are conducted, it can lead to core damage like the TMI-2 accident [1]. For this reason, research on human error reduction measures is necessary to reduce the burden on operators.

If future trends of NPP key parameters such as temperature, pressure, and water level that change due to operator actions can be predicted in real-time, it is possible to verify appropriate actions and make early corrections to inappropriate actions. To predict the trends, Bae et al. (2021) implemented deep learning models with long short-term memory (LSTM) layers and a multi-input multi-output (MIMO) strategy [2]. The model successfully predicted NPP key parameter trends according to operator actions in emergency situations in most cases. However, some inaccurate predictions were large enough to interfere with the operators' judgment, as presented in Fig.1.

The current research proposes a methodology that prevents significantly erroneous predictions when predicting future trends of NPP parameters following operator action. To this end, we divided the prediction model into two stages: accident trend prediction and operator action evaluation. As each stage dedicates to impact of accidents and the impact of actions on changes in NPP key parameters, respectively, the overall prediction accuracy is expected to be enhanced and significantly erroneous predictions can be prevented. To show feasibility, we compared the proposed two-stage model and the MIMO-LSTM model from the previous study.

2. Methods

In this section, the motivation and framework of the proposed two-stage prediction model are described. The prediction model is separated into Accident Trend Predictor and Operator Action Impact Evaluator.

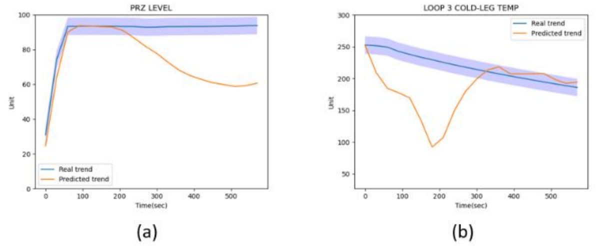


Fig. 1. Significantly erroneous prediction cases of the MIMO-LSTM model. The real trend is shown as a blue line, and the predicted trend is shown as an orange line. The blue area is the 5% error range band.

2.1 The motivation for model separation

We implemented the MIMO-LSTM model, which showed sufficient performance in the previous study, and analyzed the causes of prediction failure. Model specifications and datasets were identical to the previous study. Fig.1 (b) shows the loop 3 cold-leg temperature trend in a scenario without operator intervention in a $60cm^2$ loss of coolant accident (LOCA) situation. Even without operator intervention, the model made a significantly inaccurate prediction on the temperature trend, which sharply decreases and recovers. Due to the black-box property of the artificial neural network, it is difficult to determine the cause of such inaccurate predictions. However, because it is a supervised learning-based model, the cause of prediction failure can be found in the training dataset. Among the training dataset, we plot loop 3 cold-leg temperatures of scenarios in which various operator actions were taken in the 10 - $50cm^2$ LOCA situation. Among them, the only scenario with a sharply decreasing primary temperature was the instance when the operator stopped the RCP under the $50cm^2$ LOCA situation, as shown in Fig.2(a). Note that despite the LOCA situation with the same operator action, the temperature did not decrease rapidly as shown in Fig. 2(b) when the break size was less than $40cm^2$.

For this reason, we assumed that the model predicted the rapid decrease in temperature based on the break size. Ultimately, such prediction errors may occur because the impact of accidents and actions are not clearly distinguished. Therefore, we concluded that if we separate the model into two stages and train them to predict the trends following accidents and actions separately, significantly erroneous predictions can be prevented.

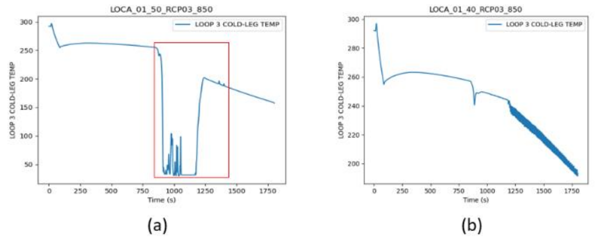


Fig. 2. Training data analysis to find the cause of the sharp decrease in temperature.; (a): 50 cm^2 LOCA scenario with action to stop RCP03; (b): 40 cm^2 LOCA scenario with action to stop RCP03.

2.2 Framework

The proposed model was developed to achieve a clear distinction between the effects of the accident and action. Therefore, it is necessary to understand where each effect is revealed in the data we deal with. When an accident occurs as an initiating event, the operator then performs an appropriate action. Until the operator action starts, the influence of the accident will appear in the data, and the impact of the operator action will appear in the data between just before the start and just after the completion of the action. In this context, the proposed model evaluates and combines the effect of the accident/action with different inputs for each stage.

Fig. 3 illustrates the framework of the proposed two-stage prediction model. Let t be the time just after the completion of the action and let $t\text{-}1$ be the time just before the start of the action. Since the N input parameters at the time point $t\text{-}2$ and $t\text{-}1$ do not include information from the action, the Accident Trend Predictor can draw the base trend ([$t+1$:$t+H$], M) as much as H time steps of M output parameters by assuming that there is no operator intervention using input_1 ([$t\text{-}2$, $t\text{-}1$], N). On the other hand, since the N input parameters at the time points $t\text{-}1$ and t contain information changed by the action, sufficient information is given to evaluate the action impact. The Operator Action Impact Evaluator learns both the base trend drawn from input_1 and the action impact information obtained from input_2 ([$t\text{-}1$, t], N) to predict the future trend ([$t+1$:$t+H$], M) reflecting the operator action impact.

2.3 Accident Trend Predictor

As previously mentioned, Accident Trend Predictor is a module that predicts trends excluding the effects of the operator action performed immediately before. In strict terms, the Accident Trend Predictor can evaluate not only the impact of an accident (initiating event) but also the automatic actions by the I&C system and previously performed actions in multi-action scenarios.

Accident Trend Predictor is an independent deep learning model, and training data must be organized separately. Rather than using all parts of the data extracted through the simulator, the time window is moved and indexed only in the time area after the final action is taken. This work serves to isolate the Accident Trend Predictor from learning the impact of additional, potential actions after the final action of the training scenario.

This module receives input from a test scenario and outputs a base trend that excludes the impact of the final action. This base trend then moves on to the next stage and is used as additional input for the Operator Action Impact Evaluator.

2.4 Operator Action Impact Evaluator

Operator Action Impact Evaluator is another deep learning model that evaluates the impact of a test scenario's final action on parameter future trends. Because plant status data before and after the final action of the test scenario is accepted as input_2, the impact is evaluated based on the change.

The base trend used as another input is received from the Accident Trend Predictor. In order to clearly distinguish the learning target, the layer of the pre-trained Accident Trend Predictor is set to untrainable during the training of the Operator Action Impact Evaluator.

The evaluator is trained by reflecting the base trend and input_2, and outputs the final future trend of NPP parameters.

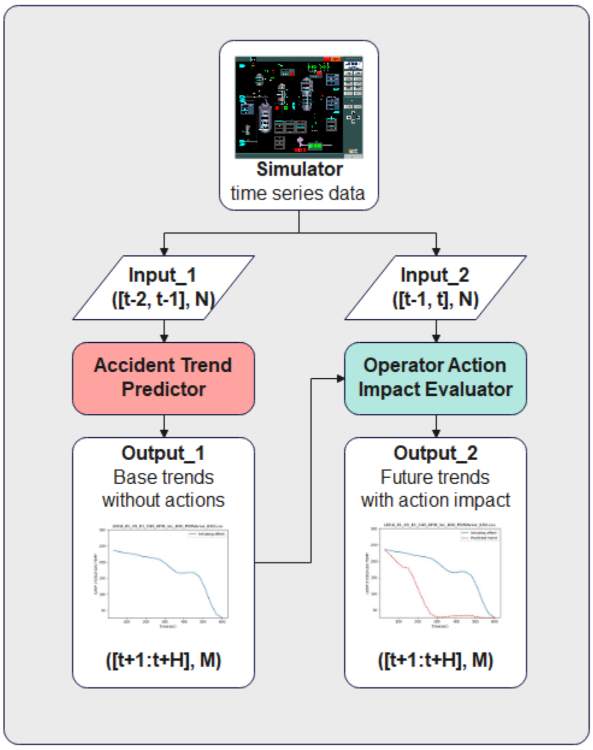


Fig. 3. The framework of the proposed two-stage prediction model

3. Case study

We conducted a case study to evaluate the feasibility of the proposed two-stage prediction model. The same data as the previous study was used, and MIMO-LSTM, which achieved the best performance in the comparison study in the previous study, and MIMO-MLP (Multilayer Perceptron) were compared with the proposed model. To implement them, we utilized Python and its libraries including Keras API. and TensorFlow [3].

3.1 Description of datasets

Data were acquired from the Compact Nuclear Simulator (CNS), a simplified simulator of a Westinghouse 1000MWe 3-loop plant [4].

Output parameters were 25 parameters illustrated in the critical safety function (CSF) tree of CNS emergency operating procedures. There are a total of 109 input parameters, including 25 output parameters in Table I and 84 parameters consisting of valve state, component state, instrument values from sensors, and important signals.

Each model predicts future trends for 20 time steps with 2 or 3 time steps as input. Since the time interval between each time step is 30 seconds, the trends of output parameters are predicted for 10 minutes.

Table I: Output parameters from the CNS [2]

	Plant Parameters (units)
1	POWER RANGE PERCENT POWER (%)
2	INTERMEDIATE RANGE START-UP RATE (DPM)
3	INTERMEDIATE RANGE NEUTRON LEVEL (A)
4	SOURCE RANGE START-UP RATE (DPM)
5	CORE OUTLET TEMPERATURE (◦C)
6	LOOP 1 HOT-LEG TEMPERATURE (◦C)
7	LOOP 2 HOT-LEG TEMPERATURE (◦C)
8	LOOP 3 HOT-LEG TEMPERATURE (◦C)
9	PRESSURIZER PRESSURE (kg/cm2)
10	STEAM GENERATOR #1 NARROW LEVEL (%)
11	STEAM GENERATOR #2 NARROW LEVEL (%)
12	STEAM GENERATOR #3 NARROW LEVEL (%)
13	FEEDWATER #1 FLOW (m3 /hr)
14	FEEDWATER #2 FLOW (m3 /hr)
15	FEEDWATER #3 FLOW (m3 /hr)
16	STEAM GENERATOR #1 PRESSURE (kg/cm2)
17	STEAM GENERATOR #2 PRESSURE (kg/cm2)
18	STEAM GENERATOR #3 PRESSURE (kg/cm2)
19	LOOP 1 COLD-LEG TEMPERATURE (◦C)
20	LOOP 2 COLD-LEG TEMPERATURE (◦C)
21	LOOP 3 COLD-LEG TEMPERATURE (◦C)
22	CONTAINMENT PRESSURE (kg/cm2)
23	CONTAINMENT SUMP WATER LEVEL (m)
24	CONTAINMENT RADIATION (mRem/hr)
25	PRESSURIZER LEVEL (%)

To extract NPP operation data, three accidents were assumed: LOCA, a steam generator tube rupture (SGTR), and a spurious reactor trip. A total of 1153 operation data were obtained by applying operator actions that could occur through analysis of each emergency operation procedure. All the scenarios outlined in Table II have different operator action timing, degree, and correctly performed prerequisites.

Table II: Simulated emergency operation scenario [2]

LOCA (865 scenarios)	Leak of reactor coolant due to 10 cm^2, 20 cm^2, 30 cm^2, 40 cm^2, or 50 cm^2 break in the cold leg
Operator action	1. Auxiliary feedwater 2. flow control 3. Reactor coolant pump stop 4. PORV shut-off valve open 5. PORV open 6. Safety injection signal reset 7. Safety injection pump stop 8. No action
SGTR (200 scenarios)	Single-tube or double-tube ruptures in a steam generator
Operator action	1. Auxiliary feedwater flow control 2. PORV shut-off valve open 3. PORV open Reactor coolant pump stop 4. Main steam line isolation 5. Secondary side relief valve manual open 6. Contaminated steam line isolation 7. No action
Spurious Trip (98 scenarios)	Unintended reactor trip due to a malfunction of the reactor protection system
Operator action	1. Auxiliary feedwater flow control 2. PORV shut-off valve open 3. PORV open Reactor coolant pump stop 4. No action

3.2 Design of deep-learning models

The architectures of MIMO-MLP and MIMO-LSTM for comparison are presented in Table III. The input of the Multilayer perceptron (MLP) flattens (2, 109) into one dimension and receives (218,), and the MIMO-LSTM model receives (2, 109) as input. On the other hand, the proposed model presented in Fig. 4 receives an input in the shape of (3, 109), splits it into two parts, and applies it to each module. Although the layers used in each module can be used in various ways, the same LSTM layers were used to check the differences in the model's structural aspect. The shape of the final output is both (20, 25), which predicts the same range of parameter trends, but the proposed model can additionally output a base trend (output_1) assuming there is no operator intervention.

Table III: ANN structures of the 3 models.

	ANNs	Input shape	Hidden layers	Cells/ hidden layer	Output shape
MIMO MLP	25	(218,)	8	200	(20,)
MIMO LSTM	25	(2,109)	4	100	(20,)
Proposed model	25	(3,109)	4	100	(20,)

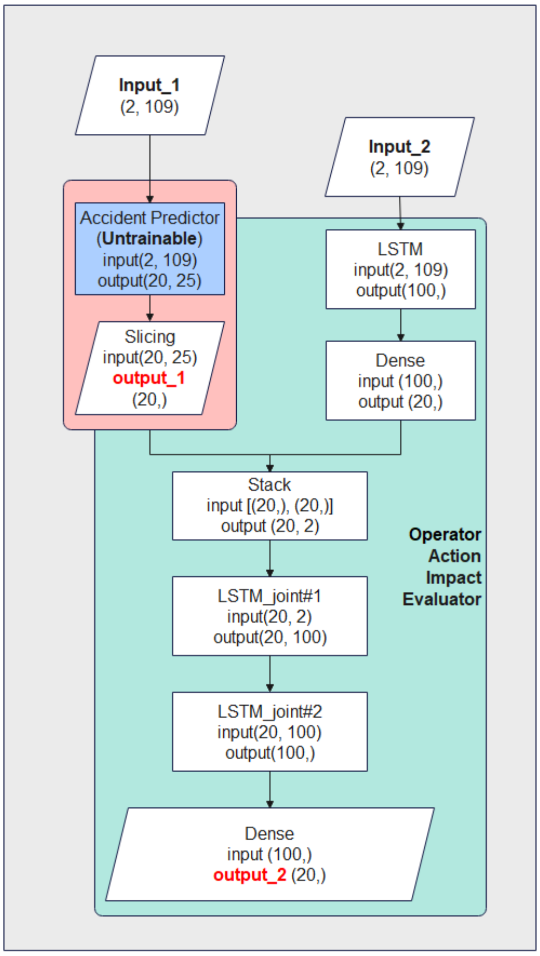


Fig. 4. The architecture of the proposed model. Accident Trend Predictor is pre-trained and further training is limited.

3.3 Error metrics

We calculated the error between the actual and predicted values. Root mean squared error (RMSE), mean squared error (MSE), and mean absolute error (MAE) were used as error metrics and are defined in Eqs (1) to (3), respectively.

We calculated 20 points for each 25 variables in 35 test scenarios, for a total of 17,500 points, and listed them in Table IV. The RMSE, MSE, and MAE of the proposed model scored the lowest at 0.0158, 0.0003, and 0.0048, respectively.

$$RMSE(y,\hat{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1}$$

$$MSE(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

$$MAE(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3}$$

Table IV: Error metrics of the 3 models.

	MIMO-MLP	MIMO-LSTM	Proposed model
RMSE	0.0866	0.0213	**0.0158**
MSE	0.0075	0.0005	**0.0003**
MAE	0.0502	0.0061	**0.0048**

3.4 Trend prediction

Fig 5 is an example of trend prediction. The real trend is indicated by a blue solid line, and the predicted value is indicated by a red dotted line. The blue shaded area represents the 5% error range, and the red shaded area represents the 10% error range, obtained from the maximum and minimum values of each parameter.

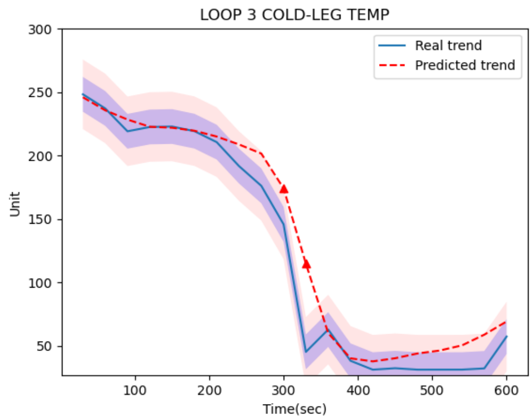


Fig. 5. Example of trend prediction results.

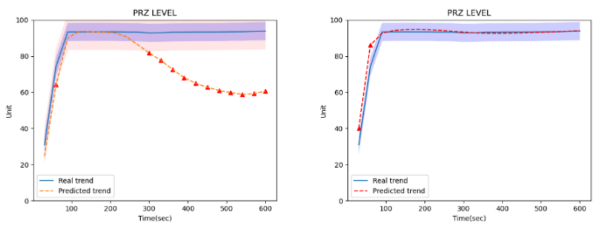
Three success criteria were defined to evaluate each predicted trend: The predicted trend is *Accurate* when all points are within the 5% error range. The predicted trend is *Mostly Accurate* when there are a maximum of 3 points outside the 5% error range. The predicted trend is *Acceptable* when there are a maximum of 2 points outside the 10% error range.

We evaluated a total of 875 trends in 25 parameters across 35 test scenarios and listed them in Table V. For all three criteria, the proposed model achieved the highest accuracy of 93.60%, 97.71%, and 99.31%, respectively.

Table V: Percentage of successful trend prediction.

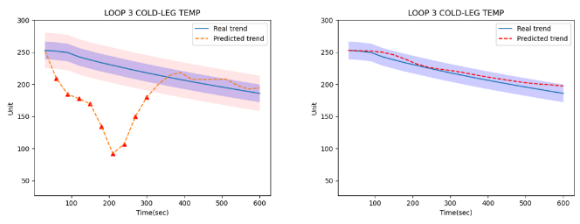
	MIMO-MLP	MIMO-LSTM	Proposed model
Accurate	43.54%	91.66%	**93.60%**
Mostly Accurate	48.23%	96.57%	**97.71%**
Acceptable	73.60%	98.63%	**99.31%**

In addition to the overall accuracy improvement, substantial improvements have been made in the proposed model for significantly erroneous prediction cases of the MIMO-LSTM model. Fig. 6 shows pressurizer level trends predicted by (a) the MIMO-LSTM model and (b) the proposed model when an operator action error that opens the PORV occurs in a LOCA situation with a 45cm^2 rupture size. Fig. 7 shows the prediction of loop 3 cold-leg temperature trends using (a) the MIMO-LSTM model and (b) the proposed model in a LOCA situation of 60cm^2 rupture size. In both cases, the MIMO-LSTM model showed errors larger than 30%, but the proposed model predicted most points well within the 5% error range. These show that the proposed model recognizes simulated accident and operator action well in test scenarios.



(a) (b)

Fig. 6. 45cm^2 LOCA with PORV action error, pressurizer level trends predicted by (a) MIMO-LSTM model and (b) Proposed model.



(a) (b)

Fig. 7. 60cm^2 LOCA, loop 3 cold-leg temperature trends predicted by (a) MIMO-LSTM model and (b) Proposed model.

4. Conclusions

Deep learning technology is being used as a useful tool to predict NPP parameter trends. In the previous study, a model combining the MIMO strategy and LSTM layer shows great performance in predicting NPP key parameter trends according to operator action. However, significantly erroneous prediction cases exist, which may interfere with the operator's judgment. As a result of training dataset analysis, it was found that the model did not distinguish well between the effects of accident and action in the input scenarios.

To solve this problem, we proposed a two-stage trend prediction deep-learning model to reduce significantly erroneous prediction cases. In the first stage, the Accident Trend Predictor reflects the impact of accidents on parameter changes by predicting a base trend assuming no final operator action is performed. In the second stage, the Operator Action Impact Evaluator predicted parameter trends by reflecting the final operator action information in the base trend received from the previous stage. Through a case study, we compared the proposed model with MIMO-LSTM and MIMO-MLP models and showed that the proposed model scored the highest accuracy while preventing significantly erroneous prediction cases.

This model is expected to reduce human errors and reduce the operators' burden by verifying operator actions through real-time prediction of NPP key parameter trends. Furthermore, by expanding to long-term predictions, it will be possible to measure the time it takes for core damage to occur for each operator action. That time can be used as a quantitative action evaluation factor, which will become the basis for developing an optimal operational action selection model.

ACKNOWLEDGEMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-RS-2022-00144042)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT). (No. RS-2022-00144328)

REFERENCES

[1] J. M. Broughton, P. Kuan, D. A. Petti, and E. L. Tolman, "A Scenario of the Three Mile Island Unit 2 Accident," Nuclear Technology, vol. 87, no. 1, pp. 34–53, 1989.

[2] J. Bae, G. Kim, and S. J. Lee, "Real-time prediction of nuclear power plant parameter trends following operator actions," Expert Systems with Applications, vol. 186, 115848, 2021.

[3] F. Chollet, "Keras: The python deep learning library," Astrophysics Source Code Library, ascl:1806, 2018.

[4] J. C. Park, K. C. Kwon, H. H. Cha, W. M. Park, S. J Song, K. W. Seh, and Y. C. Joo, "Equipment and Performance Upgrade of Compact Nuclear Simulator", KAERI/RR1967/1999 KAERI: Daejeon, 1999.