# Data-driven fault diagnosis in nuclear power plant(NPP) utilizing CNN-based Bi-LSTM models

**Authors:** Xie, Dr. Mingliang, Dr. Wei Dai, Dai, Dr. Wei

**Date:** 2025-12-17T09:57:21+00:00

## Abstract

Accurate and long-term prediction of reactor parameters during severe accidents is critical for emergency response and accident management in Nuclear Power Plants (NPPs). However, existing data-driven methods are often hindered by error accumulation in long-sequence forecasting and fail to account for dynamic external interventions (e.g., operator actions). To address these challenges, this study proposes an integrated risk prediction framework. The framework first employs a dual-stage deep network for early fault classification. Subsequently, a prognostic model utilizing a CNN-BiLSTM-Attention architecture is introduced. A novel autoregressive injection strategy is developed to dynamically integrate secondary control signals—such as valve actions and safety injection status—into the iterative prediction loop, thereby stabilizing long-term inference and ensuring consistency with system operations. The model was trained and validated on a dataset generated by RELAP5 simulations of an M310 reactor under various accident scenarios (e.g., LOCA). Experimental results demonstrate that the proposed framework significantly outperforms standard CNN and LSTM baselines. It achieves high-fidelity predictions for key parameters (e.g., core outlet temperature, system pressure) over an extended 14,400 s horizon, while maintaining robust early fault classification capabilities. This approach provides a reliable, physics-aware decision-support tool for operator intervention during accident progression.

## Full Text

## Preamble

Data-Driven Fault Diagnosis in Nuclear Power Plants Utilizing CNN-Based Bi-LSTM Models

Mingliang Xie,[1],[2] Hui Du,[3] Yu-Qing Chen,[1] Lei Yu,[1] Wei Wei,[2] and Wei Dai[3],*
[1]College of Nuclear Science and Technology, Naval University of Engineering, Wuhan 430033, China [2]China Nuclear Power Operation Technology Corporation, LTD, Wuhan 430223, China [3]School of Mathematics and Physics, China University of Geosciences, Wuhan 430074, China

Accurate and long-term prediction of reactor parameters during severe accidents is critical for emergency response and accident management in nuclear power plants (NPPs). However, existing data-driven methods are often hindered by error accumulation in long-sequence forecasting and fail to account for dynamic external interventions (e.g., operator actions). To address these challenges, this study proposes an integrated risk prediction framework. The framework first employs a dual-stage deep network for early fault classification. Subsequently, a prognostic model utilizing a CNN-BiLSTM-Attention architecture is introduced. A novel autoregressive injection strategy is developed to dynamically integrate secondary control signals—such as valve actions and safety injection status—into the iterative prediction loop, thereby stabilizing long-term inference and ensuring consistency with system operations. The model was trained and validated on a dataset generated by RELAP5 simulations of an M310 reactor under various accident scenarios (e.g., LOCA). Experimental results demonstrate that the proposed framework significantly outperforms standard CNN and LSTM baselines. It achieves high-fidelity predictions for key parameters (e.g., core outlet temperature, system pressure) over an extended 14,400 s horizon, while maintaining robust early fault classification capabilities. This approach provides a reliable, physics-aware decision-support tool for operator intervention during accident progression.

**Keywords:** Nuclear Power Plant, Deep Learning, Fault Diagnosis

## Introduction

Despite three serious nuclear accidents undermining confidence in nuclear power, over 440 nuclear power plants are operational worldwide today [**?**, **?**, **?**]. The issue is not if a nuclear accident will occur, but when. Therefore, accurate early-stage fault detection and diagnosis, along with appropriate remedial measures, are crucial for the safe operation of nuclear power plants, reducing accident impact and economic losses. Relying solely on operators for early-stage fault diagnosis can lead to serious consequences due to human error [**?**]. For example, the 1979 Three Mile Island accident resulted from an operator' s misdiagnosis, causing reactor core damage [**?**]. To ensure safer and more reliable operations, automatic fault diagnosis and detection methods are necessary.

A number of automated fault diagnosis and detection (FDD) methods have been applied to nuclear power plants today, including hardware redundancy-based, model-based, signal processing-based, and data-driven machine learning methods [**?**, **?**, **?**]. Compared to traditional FDD methods, data-driven machine learning approaches do not require prior experience and expertise, but only historical data for training, which is more aligned with practical industrial application scenarios. The use of data-driven machine learning methods has grown rapidly in recent decades [?, ?, ?].

In recent years, artificial intelligence methods have been applied to FDD and trend prediction in nuclear power plants, representing a subset of the data-driven machine learning approach. For example, Artificial Neural Networks (ANNs) are used for reliability assessment of passive systems [?] and accident prevention systems [?]. Convolutional Neural Networks (CNNs) have been proposed for crack detection [?] and diagnosing flow-accelerated corrosion-induced pipe thinning [?]. Recently, accident diagnosis systems based on recurrent neural networks have been developed that are tolerant to sensor faults [?, ?]. A new fault diagnosis scheme based on CNN and LSTM—a special type of recurrent neural network—has been proposed and verified using different operational states along with four different fault types and one blind case [?]. Additionally, a CNN-LSTM hybrid model has been used for fault diagnosis [?].

These data-driven machine learning methods for FDD mainly categorize fault types or monitor outliers, but do not provide early incident warnings. Thus, another research category uses data-driven machine learning to predict trends, aiding operators in diagnosis and response. For example, LSTM neural networks are used for long-term trend prediction of steam generators [?] and reactor coolant pumps [?]. While successful, these predictions span days, whereas faults can escalate to serious accidents in minutes to hours, limiting early fault diagnosis and reaction. A cost-sensitive LSTM (CS-LSTM) method predicts water level trends in reactor pressurizers [?], but it fails to consider correlations between multiple parameters, offering limited predictive time and assistance during emergencies. Nine prediction models based on Multilayer Perceptron, RNN, and LSTM were tested for real-time prediction of multiple nuclear plant parameters [?]. Despite considering parameter correlations, their predictions still provide only short-term trends.

In this study, we propose an integrated Fault Diagnosis and Risk Prediction Framework based on deep learning architectures, validated primarily on Loss-of-Coolant Accident (LOCA) scenarios. The framework comprises two synergistic components: first, a Dual-Stage Spatiotemporal Network is introduced for the rapid and accurate pre-classification of accident types at the early stage. Building on this diagnosis, the second component utilizes a CNN-BiLSTM model to execute long-term prognosis. By employing a multi-input multi-output structure combined with a multi-step autoregressive strategy [?, ?, ?, ?, ?, ?], the model is capable of simultaneously forecasting the trends of 31 critical physical parameters—including main system pressure, core outlet temperature, and containment pressure—in real-time. A key breakthrough of this approach is the extension of the reliable prediction horizon to 3600 s, enabling full-spectrum coverage of various accident stages. Furthermore, unlike static forecasting models, this framework explicitly integrates operator interventions (dynamic signal injection) into the inference loop. This dynamic capability provides operators with long-term, action-responsive trend predictions, significantly enhancing the reliability of fault diagnosis and emergency decision-making.

## II. Data Preparation

This section focuses on the generation of LOCA accident data, the preprocessing scheme for training data, the basic structure of the model, the training methodology, and the prediction approach. The dataset was computed by the RELAP5 simulation software based on designed fault sequences. The dataset is routinely screened, and wavelet noise reduction is applied to remove oscillatory noise. The dataset is then divided into training and test sets according to an 8:2 ratio. The normalized data is used to train the model based on the set hyperparameters. A multi-step iterative prediction approach was used to forecast 72 reactor parameters for future periods after the fault.

### A. Accident Simulation

In this paper, the M310 nuclear power plant is used as an example. RELAP5 is employed to simulate the operation of the nuclear power plant, and the LOCA accident is simulated by inserting a breach in the primary coolant loop of the reactor [**?**, **?**, **?**]. RELAP5 is a premier thermal-hydraulic simulation tool used to analyze the dynamic response of reactor systems following operational transients. Through precise design of input boundary conditions, it enables simulation of distinct accident types: Loss-of-Coolant Accidents (LOCA) defined by rapid depletion of reactor coolant inventory; Steam Generator Tube Ruptures (SGTR) leading to containment bypass via tube failure; Main Steam Line Breaks (MSLB) causing excessive system overcooling; Station Blackout (SBO) characterized by complete loss of all AC power sources and subsequent failure of active safety systems; and Loss of Main Feedwater (LOFW) resulting in degradation of the steam generators' heat removal capability.

Input cards have been created according to the key parameters of the M310 nuclear power plant in Table 1 and input into RELAP5 to obtain simulated LOCA data for training and testing the neural network model. The LOCA is initiated through the sequence of events designed in Table 2 , assuming the accident occurs from time $t_1$. At $t_1$, the nuclear power plant loses power. At $t_2$, a main transformer ground fault occurs while the main pumps are stopped, the reactor is shut down, the turbine is shut down, and primary feedwater is stopped. At $t_3$, auxiliary feedwater begins operation. At $t_4$, a random-sized breach occurs in the primary coolant loop. At $t_5$, the High-Pressure Safety Injection System (HPSIS) is activated. At $t_6$, the HPSIS fails. Eventually, coolant is lost from the primary coolant loop through the breach. This process was simulated using RELAP5, yielding a total of 700 LOCA simulations with run times ranging from 400 s to 50,000 s, depending on the size of the inserted breach.

Each dataset was recorded at 10-second intervals during reactor operation for a total of 72 parameters, including main system pressure, heat pipe section temperature, reactor core outlet temperature, pressure vessel wide-range water level, pressurizer pressure, steam generator pressure, and containment upper compartment pressure.

### B. Data Pre-Processing

To improve model training efficiency and prediction accuracy, the data obtained from RELAP5 must be pre-processed, and data that do not meet requirements must be discarded before training the CNN-BiLSTM model. The data are first pre-screened: samples are considered too short if the accident duration time t $\leq$ 600 s, and no LOCA is considered to have occurred if there is no loss of coolant or if the pressure of the main system is maintained above 15.5 MPa throughout the accident. After screening, 370 samples remain available for model training.

Missing values for a parameter in a sample are replaced with the average value of that parameter across the sample. The data are then randomly divided into training and test sets in a 9:1 ratio. The 333 training samples contained data sliced to a length of 240 s, resulting in 1,173,251 training data points using a sliding window of 10 s.

Figure 1 [Figure 1: see original paper] illustrates the original simulation data versus the denoised signal for steam generator pressure. By analyzing the RELAP5 simulation results, it was found that the steam generator pressure exhibited severe oscillation during the late stage of LOCA. Due to coolant inflow into the steam generator during the late accident stage, pressure increases until reaching the limit, at which point the relief valve opens to release pressure, causing a decrease. The relief valve then closes, but continuous coolant inflow causes pressure to rise again, resulting in a repetitive process that creates violent oscillations around the relief valve threshold. Such oscillating signals may hinder the model' s ability to learn useful information. Therefore, the data was smoothed using wavelet noise reduction, which is only effective for high-frequency noise and does not affect the values of other parameters during the process [**?**].

As illustrated by the red curve in Figure 1, the noise reduction process significantly attenuates the oscillations in SG pressure while preserving data integrity with minimal distortion. The overall trend remains consistent, and the evolutionary characteristics of the parameters are preserved. Testing revealed that noise reduction significantly increases model learning efficiency, as evidenced by the decrease in training loss shown in Figure 11 [Figure 11: see original paper] of the Appendix.

## III. Methodology

### A. Integrated Diagnosis and Prediction Framework

Figure 2 [Figure 2: see original paper] presents the comprehensive framework of the proposed methodology, which systematically integrates data generation, fault diagnosis, and long-term prognostic prediction. The process begins with establishing a high-fidelity fault dataset utilizing RELAP5 simulations, followed by data preprocessing to standardize inputs.

For fault identification, the system employs a Dual-Stage Spatiotemporal Network (top-right panel). This module extracts deep feature representations through cascaded CNN-BiLSTM blocks and an Attention mechanism to accurately classify accident types (e.g., LOCA, MSLB, SGTR, LOFW, SBO) at the onset of a transient.

Upon identifying the fault type, the system activates the Disaster Prediction Module. As illustrated in the "Prediction Method" diagram (bottom-right), predictions are generated using a CNN-BiLSTM architecture governed by a multi-step iterative strategy with dynamic signal injection. This strategy explicitly categorizes the 72 system parameters into 31 physical state parameters (to be predicted) and 41 signal parameters (control variables). In each predictive step, the CNN-BiLSTM model forecasts the physical state for the subsequent 40 seconds. These predictions are then concatenated with the corresponding signal parameters to reconstruct the complete system state. This fused vector updates the sliding window input for the next iteration, enabling the model to maintain physical consistency and stability over extended prediction horizons.

### B. Design of Neural Network Models

To implement the integrated diagnosis and prediction framework, two specialized variations of the CNN-BiLSTM architecture were designed. The first is the Fault Pre-classification Model, which adopts a Dual-Stage Spatiotemporal Network architecture [**?**, **?**, **?**]. Unlike standard single-pass models, this classifier incorporates a unique feature reconstruction stage that refines latent representations before passing them to the deep classification stage. This hierarchical design ensures robust identification of distinct accident types (e.g., LOCA, SGTR, MSLB, LOFW, SBO) by effectively filtering noise and enhancing critical feature patterns from raw sensor data.

Building upon this architectural foundation, the Disaster Prediction Model is specifically optimized for continuous multi-step forecasting. This model was implemented using the PyTorch framework, and its core architecture comprises a CNN layer, a Bi-LSTM layer, and a self-attention layer [**?**, **?**, **?**, **?**, **?**, **?**]. Functionally, the CNN layer first extracts deep spatial coupling patterns among multiple physical parameters from the reshaped 2D feature matrix. Subsequently, the BiLSTM encodes the sequence to precisely capture the long-range temporal dependencies inherent in accident evolution. Finally, an attention mechanism introduces adaptive weighting to dynamically focus on the critical time points that contribute most significantly to the prediction.

The detailed structure of the prediction model is illustrated in Figure 3 [Figure 3: see original paper]. The architecture connects the CNN, BiLSTM, and Attention layers sequentially via Tanh activation functions. Specifically, the CNN module contains two convolutional layers and two pooling layers, interconnected by ReLU activation functions. The input data is convolved, pooled, and reshaped into an n × 72 matrix before being passed to the BiLSTM layer. The BiLSTM module, consisting of 6 hidden layers, processes this sequence to capture temporal dynamics. The output is then forwarded to the attention layer, where variable weights are recalculated to highlight salient features before the final result is generated by the output layer.

For the prediction task, the model input is defined as a 24 × 72 matrix, representing historical data of 72 reactor parameters over the past 240 s. The output is a 4 × 31 matrix, forecasting the 31 key physical parameters for the subsequent 40 s. The model comprises a total of 105,184 trainable parameters. Hyperparameters were optimized based on dataset size and task complexity; specific settings are detailed in Table 3 . Training utilized a batch size of 256, a learning rate of 0.0005, and 250 epochs, employing the Adam optimizer. The loss function is defined as Mean Squared Error (MSE), calculated as:

$$MSE = \sum (\hat{x}_i - x_i)^2$$

where n represents the number of observations, while $\hat{x}_i$ and $x_i$ denote the predicted values and ground-truth simulation values for sample i, respectively. Before training, the dataset is normalized and partitioned into batches to ensure efficient model convergence.

**Table 3. Hyperparameters for model training**

| Parameter | Value |
|---|---|
| Batch size | 256 |
| Epoch | 250 |
| Learning rate | 0.0005 |
| Weight Decay | 0.00001 |
| Optimizer | Adam |
| Loss function | Mean Square Error |

### C. Prediction Model Training

A specialized multi-step autoregressive training strategy was implemented to enhance the model' s stability for long-term forecasting [**?**, **?**, **?**]. This approach executes a recursive loop of four iterations within each training batch to explicitly simulate the error propagation process inherent in sequential prediction. During the i-th iteration of this loop (i  {0, 1, 2, 3}), the model processes the current input matrix, representing 240 seconds of history for all 72 parameters, to forecast the 31 physical parameters for the subsequent 40-second interval. The deviation between these predictions and ground truth values is quantified using MSE, denoted as $L_i$. To mitigate the impact of cumulative errors in later prediction stages, a progressive weighting factor $w_i$ is applied to the loss of each successive iteration.

The training phase employs a physics-informed state reconstruction mechanism to dynamically update the input sequence. The generated predictions for the 31 physical parameters are concatenated with the ground truth values of the 41 signal parameters for the corresponding timeframe. This fusion reconstructs the complete 72-parameter state vectors using accurate external intervention signals. These reconstructed vectors are then appended to the end of the input sequence via a sliding window mechanism, discarding the oldest 40 seconds of data to form the input for the subsequent iteration. Upon completion of the four recursive iterations, backpropagation and parameter optimization are performed based on the accumulated total weighted loss $L_{total}$, formulated as:

$$L_{total} = \sum w_i L_i, \text{ with } w_i = 1 + 0.2(i+1)$$

In this formulation, the coefficient $w_i$ increases with the iteration step, thereby assigning higher penalties to prediction errors occurring further in the future [**?**]. This methodology conditions the model to capture the evolutionary trends of the 31 physical parameters under specific control actions while simultaneously learning to correct potential errors introduced by its own prior predictions.

### D. Disaster Prediction

In real accident scenarios, reactor parameter information may not be available after the accident occurs. Therefore, the model must predict reactor parameter trends over extended future periods using only short-term pre-accident information. Additionally, considering that manual and automatic remedial interventions will occur after the accident, the model should be able to insert the effects of these interventions into the input during the prediction process. Consequently, the CNN-BiLSTM model adopts a multi-step iterative prediction method that can forecast future trends of 31 physical reactor parameters over any length of time using information from 72 reactor parameters over the 240 s before the accident, with optimization focusing primarily on the 3600 s period after accident initiation.

The prediction process is shown in Figure 2 (bottom-right panel). During prediction, the 72 reactor parameters are divided into two categories: physical parameters (including main system pressure, reactor core outlet temperature, steam generator pressure, etc.) totaling 31 parameters, and signal parameters (including feedwater opening signals, high-pressure injection opening signals, and various valve opening/closing signals) totaling 41 parameters, which are primarily Boolean-type. When a nuclear power plant fault occurs, the model is activated and reactor parameters are read as input for the first round of prediction.

For the first prediction step, a matrix representing 240 seconds of history for all 72 parameters, combined with temporal encoding, serves as the input. The model then generates an output matrix representing the predicted values of the 31 parameters for the subsequent 40 seconds. Under the assumption that the 41 signal parameters remain constant in the absence of external updates, the model output is concatenated with these signal parameters to form a complete state matrix. This new data block is appended to the input sequence while discarding the oldest data points to maintain a sliding window. Combined with updated temporal information, this constructs the input for the subsequent prediction cycle. By recursively advancing 40 seconds at each step, the model achieves continuous long-term forecasting of reactor parameter evolution.

## IV. Results Analysis

This section presents experimental results, organized to first demonstrate fault diagnosis capabilities and subsequently evaluate long-term parameter prediction performance. Initially, the model' s efficacy in early accident detection and identification of diverse fault types is analyzed. Following this, the training outcomes of the proposed CNN-BiLSTM prediction architecture are detailed, including comparative analysis against benchmark CNN, LSTM, and CNN-LSTM models. Finally, the predictive capability of the model for 31 key reactor parameters during LOCA scenarios is assessed across varying break sizes and prediction horizons.

### A. Fault Diagnosis

The recognition ability of the CNN-BiLSTM model in the early stage of accidents was tested first. Data for several accident types—SGTR, MSLB, LOFW, and SBO—were calculated via RELAP5 simulation and used to train the accident classifier. The training results are shown in Figure 4 [Figure 4: see original paper] and Figure 5 [Figure 5: see original paper]. The model' s recognition ability for these five accident types was tested, with results shown in Table 4 . These results demonstrate that the CNN-BiLSTM model is capable of accurately identifying different types of faults at the early stage, which is a prerequisite for selecting the appropriate prediction model.

**Table 4. Classification accuracy**

| Accident | Accuracy |
|---|---|
| LOCA | 100.0% |
| SGTR | 100.0% |
| MSLB | 100.0% |
| LOFW | 93.48% |
| SBO | 100.0% |

### B. Parameter Predictions

Following accurate diagnosis, the parameter prediction model is activated. The primary model was trained according to structural specifications and hyperparameter settings defined in Section III. Training and validation losses were monitored across all epochs to evaluate convergence. Figure 6 [Figure 6: see original paper] illustrates the model training loss, with the red curve representing performance on the training set and the blue curve indicating loss on the test set. The loss is obtained via MSE. The figure shows that loss decreases rapidly in initial training rounds, reaching a minimum of 0.000253 after 200 training rounds on the test set. After 250 training rounds, the model' s loss on the training set converges to 0.000205. The minimal discrepancy between training and test set losses indicates strong generalization capability.

To ensure rigorous comparison, baseline models were trained using identical hyperparameter configurations. Training losses for CNN, LSTM, CNN-LSTM, and CNN-BiLSTM models are compared in Table 5 .

**Table 5. Train loss on different models**

| Model | Epoch | Parameter Loss |
|---|---|---|
| CNN-LSTM | 250 | (value not specified) |
| CNN-BiLSTM | 250 | 0.000205 |

The overall performance of the CNN-BiLSTM model on the test set was evaluated using the test measurement RRMSE :

$$RRMSE_t = \sqrt{\frac{\sum_{i=1}^{n}(x_i^t - \hat{x}_i^t)^2}{\sum_{i=1}^{n} x_i^t}}$$

where n is the number of observations, and $\hat{x}_i^t$ and $x_i^t$ are the predicted and simulation true values at t seconds on sample i, respectively. Results are shown in Table 6 . The model' s prediction performance on the test set is stable, with minimal average deviation from simulation results. RRMSE $\leq$ 20% for the full prediction time and RRMSE $\leq$ 10% for most of the time, while regulator temperature predictions achieve RRMSE $\leq$ 5%.

**Table 6. RRMSE on test sample**

| Feature | 1000s | 1500s | 2000s | 2500s | 3000s | 3500s | 4000s |
|---|---|---|---|---|---|---|---|
| Main system pressure | 4.12% | 6.64% | 11.8% | 14.5% | 13.0% | 11.9% | 15.3% |

| Feature | 1000s | 1500s | 2000s | 2500s | 3000s | 3500s | 4000s |
|---|---|---|---|---|---|---|---|
| Pressure vessel wide range water level | 3.59% | 7.88% | 5.45% | 4.92% | 6.50% | 5.55% | 6.87% |
| Core outlet temperature | 1.15% | 7.04% | 2.37% | 10.1% | 4.83% | 3.78% | 3.06% |
| Regulator temperature | 1.16% | 1.07% | 0.71% | 0.84% | 0.90% | 0.77% | 0.66% |
| Steam generator pressure | 1.91% | 4.14% | 3.58% | 7.21% | 8.35% | 7.46% | 4.11% |
| Containment vessel pressure | 4.87% | 7.84% | 9.56% | 9.85% | 7.53% | 7.10% | 7.40% |

The predictive effectiveness of the CNN-BiLSTM model was further evaluated across different accident samples. Since the progression timeline from fault initiation to potential core meltdown depends heavily on breach size, datasets of varying durations were selected to assess model performance under different dynamic conditions. Specifically, durations of 0-5000 s, 5000-20000 s, and >20000 s represent large-breach, medium-breach, and small-breach LOCAs, respectively. The simulation protocol involves loss of offsite power at 400 s, followed by breach initiation at 640 s, after which model inference commences. Evaluation focused on prediction of six key reactor parameters: main system pressure, pressure vessel wide-range water level, core outlet temperature, pressurizer temperature, primary loop SG pressure, and containment pressure.

Figure 7 [Figure 7: see original paper] and Figure 8 [Figure 8: see original paper] show model predictions of reactor key parameters at 0 s and 2000 s after the occurrence of a break in a large-breach LOCA accident, with a prediction length of 3600 s. Blue curves show RELAP simulation results, and red curves show model predictions. In Figures 7(b) and 7(c), model predictions describe this period well, with core temperature rising due to coolant loss, leaving the core bare in the 1000-2000 s interval, and falling as injection opens to refill the core. In Figures 8(b) and 8(c), the model also describes well the continued coolant loss from auxiliary feedwater failure about 3000 s after the accident, and the complete coolant loss after 4000 s, resulting in the reactor core being exposed again and temperature rising until meltdown.

Figure 9 [Figure 9: see original paper], together with Figure 15 [Figure 15: see original paper] and Figure 16 [Figure 16: see original paper] in the Appendix, illustrate model predictions for a medium-breach LOCA initiated at 0 s, 5000 s, and 10000 s post-breach, respectively, with a 3600 s prediction horizon. Blue curves represent RELAP simulation ground truth, while red curves denote model predictions. Similarly, Figure 17 [Figure 17: see original paper], Figure 18 [Figure 18: see original paper], and Figure 19 [Figure 19: see original paper] in the Appendix depict reactor key parameter predictions for a small-breach LOCA at the same intervals of 0 s, 5000 s, and 10000 s post-breach, with a constant prediction length of 3600 s. The comparison between RELAP simulation results (blue) and model predictions (red) demonstrates tracking capability across different accident scales.

Model performance for longer prediction periods was also tested. The same samples were chosen, with results shown in Figure 20 [Figure 20: see original paper] (see Appendix) and Figure 10 [Figure 10: see original paper], where the model predicts key reactor parameter trends for 14,400 s after breach insertion. Blue curves show RELAP simulation results, and red curves show model predictions.

These plots demonstrate that CNN-BiLSTM model predictions of changes in six key parameters over time after LOCA-type accidents with different breach sizes are in good agreement with RELAP5 simulation results. The model accurately predicts the process from coolant loss leading to an exposed reactor core, followed by temperature rise and eventual core meltdown. The model also agrees with simulation results in predicting different accident stages. The results in Figure 20 of the Appendix and Figure 10 show that the CNN-BiLSTM model is stable and can accurately predict reactor parameter evolution for up to 4 hours.

## Conclusion

This paper proposes an integrated Fault Diagnosis and Risk Prediction Framework based on deep learning architectures to enhance safety management in nuclear power plants. Validated on typical accident sequences simulated by RELAP5 for an M310 reactor, the framework demonstrates superior performance in both early fault identification and long-term prognostic forecasting.

First, the Dual-Stage Spatiotemporal Network exhibited robust generalization capabilities in the pre-classification task. By utilizing limited data from the early accident stage, the model accurately identified five distinct accident types: LOCA, SGTR, MSLB, LOFW, and SBO. Recognition accuracy reached 100% for LOCA, SGTR, MSLB, and LOFW, and exceeded 93% for SBO. Furthermore, the model successfully distinguished between different break sizes in LOCA scenarios, providing a reliable foundation for selecting appropriate downstream prediction models.

Second, the Disaster Prediction Model, underpinned by a CNN-BiLSTM architecture and multi-step autoregressive strategy with dynamic signal injection, effectively addressed the challenge of long-term prediction instability. Experimental results confirm that the model can accurately forecast trajectories of 31 key physical parameters. Specifically, the model maintained an average Relative Root Mean Square Error (RRMSE) of less than 10% over a standard 3600 s horizon. Moreover, stability tests demonstrated that the model effectively converges and tracks accident progression for extended periods up to 14,400 s (4 hours), accurately capturing critical phases from coolant loss to potential core meltdown.

The CNN-BiLSTM model' s ability to quickly identify accident type and accurately predict reactor state can assist nuclear power plant emergency responders in understanding reactor condition, accident severity, and accident development trends, enabling them to take more effective and appropriate remedial measures. Furthermore, this predictive framework paves the way for future development of Autonomous Intervention Models based on Deep Reinforcement Learning [**?**], aiming to evolve from passive decision support to active, intelligent accident mitigation. This work provides a reliable, physics-aware decision-support tool, advancing the field from passive monitoring toward active, intelligent mitigation during accident progression. Future research will focus on extending the framework to a broader spectrum of accident scenarios, optimizing the model for real-time deployment, and exploring its integration with plant digital twin systems.

## Bibliography

[1] Z. Gu, Annals of Nuclear Energy 120, 682 (2018). [2] A. Adamantiades and I. Kessides, Energy Policy 37, 5149 (2009). [3] Q. Wang, R. Li, and G. He, Renewable and Sustainable Energy Reviews 90, 90 (2018). [4] S. M. Friedman, Bulletin of the Atomic Scientists 67, 55 (2011), https://doi.org/10.1177/0096340211421587. [5] P. Le Bot, Reliability Engineering & System Safety 83, 153 (2004), Human Reliability Analysis: Data Issues and Errors of Commission. [6] B. Qi, J. Liang, and J. Tong, Energies 16 (2023), 10.3390/en16041850. [7] T. Shorthill, H. Bao, H. Zhang, and H. Ban, Nuclear Technology 208, 892 (2022). [8] K. E. Holbert and K. Lin, Science and Technology of Nuclear Installations 2012, 421070 (2012). [9] G. Hu, T. Zhou, and Q. Liu, Frontiers in Energy Research 9 (2021), 10.3389/fenrg.2021.663296. [10] H. Song, X. Liu, and M. Song, Applied Energy 341, 121077 (2023). [11] M. El-Sefy, A. Yosri, W. El-Dakhakhni, S. Nagasaki, and L. Wiebe, Nuclear Engineering and Technology 53, 3275 (2021). [12] R. Solanki, H. D. Kulkarni, S. Singh, P. Varde, and A. Verma, Annals of Nuclear Energy 144, 107487 (2020). [13] L.-C. C. Po, Nuclear Technology 206, 505 (2020), https://doi.org/10.1080/00295450.2019.1641877. [14] F.-C. Chen and M. R. Jahanshahi, IEEE Transactions on Industrial Electronics 65, 4392 (2018). [15] Y. H. Chae, S. G. Kim, H. Kim, J. T. Kim, and P. H. Seong, Annals of Nuclear Energy 143, 107501 (2020). [16] J. Choi and S. J. Lee, Sensors 20 (2020), 10.3390/s20205839. [17] M. Lin, J. Li, Y. Li, X. Wang, C. Jin, and J. Chen, Energy 282, 128905 (2023). [18] H. A. Saeed, M. jun Peng, H. Wang, and B.

wen Zhang, Progress in Nuclear Energy 118, 103066 (2020). [19] C. Zhang, P. Chen, F. Jiang, J. Xie, and T. Yu, Energies 16 (2023), 10.3390/en16062934. [20] Y. Wang, J. Liu, H. Liao, H. Wang, and Z. Zhang, Annals of Nuclear Energy 227, 111913 (2026). [21] C. Zhang, P. Chen, F. Jiang, J. Xie, and T. Yu, Energies 16, 2934 (2023). [22] J. Xing, W. Li, S. Deng, P. Duan, and X. Ma, in Journal of Physics: Conference Series, Vol. 2425 (IOP Publishing, 2023) p. 012038. [23] Y. Chen, M. Lin, R. Yu, and T. Wang, Science and Technology of Nuclear Installations 2021, 8839867 (2021). [24] B. Wang, Y. Jiang, M. Lin, and Q. Wang, Nuclear Engineering and Design 436, 113998 (2025). [25] R. Xu, Y. Wang, M. Peng, H. Cui, H. Wang, and C. Kuang, in International Conference on Nuclear Engineering, Vol. 88216 (American Society of Mechanical Engineers, 2024) p. V001T01A032. [26] Z. Wang, P. Sun, and X. Wei, in The Proceedings of the International Conference on Nuclear Engineering (ICONE) 2023.30 (The Japan Society of Mechanical Engineers, 2023) p. 1644. [27] H.-P. Nguyen, J. Liu, and E. Zio, Applied Soft Computing 89, 106116 (2020). [28] H.-P. Nguyen, P. Baraldi, and E. Zio, Applied Energy 283, 116346 (2021). [29] J. Zhang, X. Wang, C. Zhao, W. Bai, J. Shen, Y. Li, Z. Pan, and Y. Duan, Nuclear Engineering and Technology 52, 1429 (2020). [30] J. Bae, G. Kim, and S. J. Lee, Expert Systems with Applications 186, 115848 (2021). [31] S. B. Taieb, G. Bontempi, A. Sorjamaa, and A. Lendasse, in 2009 International Joint Conference on Neural Networks (2009) pp. 3054–3061. [32] J. Bae, G. Kim, and S. J. Lee, Expert Systems with Applications 186, 115848 (2021). [33] I. E. Livieris and P. Pintelas, Neural Computing and Applications 34, 19453 (2022). [34] Y. Xiao, J. Liu, and Q. Su, Progress in Nuclear Energy 155, 104517 (2023). [35] H. Kim and J. Kim, in 31st European Safety and Reliability Conference (ESREL' 21) (2021) pp. 818–824. [36] H.-P. Nguyen, P. Baraldi, and E. Zio, Applied Energy 283, 116346 (2021). [37] A. Mangal, V. Jain, and A. Nayak, Progress in Nuclear Energy 61, 1 (2012). [38] M. Wang, Y. Wang, W. Tian, S. Qiu, and G. Su, Annals of Nuclear Energy 150, 107836 (2021). [39] M. Xie, Y. Chen, L. Yu, W. Wei, Z. Xie, and F. Shan, in 2023 11th International Conference on Information Systems and Computing Technology (ISCTech) (IEEE, 2023) pp. 384–389. [40] G. Y. Park, J. Park, and P. H. Seong, Nuclear Technology 145, 177 (2004). [41] Z. Cui and C. Zhao, IFAC-PapersOnLine 53, 17035 (2020). [42] Y. Liu, C. Gong, L. Yang, and Y. Chen, Expert Systems with Applications 143, 113082 (2020). [43] Y. Xiao, H. Yin, Y. Zhang, H. Qi, Y. Zhang, and Z. Liu, International Journal of Intelligent Systems 36, 2036 (2021). [44] L. Shan, Y. Liu, M. Tang, M. Yang, and X. Bai, Journal of Petroleum Science and Engineering 205, 108838 (2021). [45] J. Zhang, L. Ye, and Y. Lai, Mathematics 11 (2023), 10.3390/math11091985. [46] C. Ren, H. Li, J. Lei, J. Liu, W. Li, K. Gao, G. Huang, X. Yang, and T. Yu, Nuclear Technology 209, 1365 (2023). [47] F. Dong, S. Chen, K. Demachi, M. Yoshikawa, A. Seki, and S. Takaya, Nuclear Engineering and Design 404, 112161 (2023). [48] T. Zhang, Q. Jia, C. Guo, and X. Huang, Energies 16, 6745 (2023). [49] G. Qian and J. Liu, Progress in Nuclear Energy 155, 104502 (2023). [50] D. A. Ejigu and X. Liu, Progress in Nuclear Energy 161, 104729 (2023). [51] J. Kaur, K. S. Parmar, and S. Singh, Environmental Science and Pollution

Research 30, 19617 (2023). [52] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, International Journal of Forecasting 36, 1181 (2020). [53] J. Lei, C. Ren, W. Li, L. Fu, Z. Li, Z. Ni, Y. Li, C. Liu, H. Zhang, Z. Chen, et al., International Journal of Energy Research 46, 21467 (2022). [54] J. Bae, J. M. Kim, and S. J. Lee, Nuclear Engineering and Technology 55, 3277 (2023).

## Appendix

### A. Key Parameter Predictions of LOCA Scenario

Figure 11 [Figure 11: see original paper]. Comparison of training loss on raw and denoised datasets.

Figure 12 [Figure 12: see original paper]. RRMSE of parameter predictions starting at 0s.

Figure 13 [Figure 13: see original paper]. RRMSE of parameter predictions starting at 5000s.

Figure 14 [Figure 14: see original paper]. RRMSE of parameter predictions starting at 10000s.

Figure 15 [Figure 15: see original paper]. Model prediction results of reactor key parameters starting at 5000s after medium-breach LOCA.

Figure 16 [Figure 16: see original paper]. Model prediction results of reactor key parameters starting at 10000s after medium-breach LOCA.

Figure 17 [Figure 17: see original paper]. Model prediction results of reactor key parameters starting at 0s after small-breach LOCA.

Figure 18 [Figure 18: see original paper]. Model prediction results of reactor key parameters starting at 5000s after small-breach LOCA.

Figure 19 [Figure 19: see original paper]. Model prediction results of reactor key parameters starting at 10000s after small-breach LOCA.

Figure 20 [Figure 20: see original paper]. Long-term trend prediction of key reactor parameters at 14400s in medium-breach LOCA.

*Note: Figure translations are in progress. See original paper for figures.*

*Source: ChinaXiv —Machine translation. Verify with original.*