Annals of Nuclear Energy 178 (2022) 109373

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene

Deep learning techniques for in-core perturbation identification and localization of time-series nuclear plant measurements

Antonios Papaoikonomou [a], James Wingate [b], Vasudha Verma [c], Aiden Durrant [d], George Ioannou [a], Tasos Papagiannis [a], Miao Yu [b], Georgios Alexandridis [a], Abdelhamid Dokhane [c,*], Georgios Leontidis [d], Stefanos Kollias [a,b], Andreas Stafylopatis [a]

[a] School of Electrical & Computer Engineering, National Technical University of Athens, Zografou, 157 80, Greece
[b] University of Lincoln, Machine Learning Group, School of Computer Science, Brayford Pool, Lincoln, United Kingdom
[c] Paul Scherrer Institut, Laboratory for Reactor Physics and Thermal-Hydraulics, Forschungsstrasse 111, 5232 Villigen PSI, Switzerland
[d] University of Aberdeen, Department of Computing Science & Interdisciplinary Centre for Data and AI, AB24 3UE, Aberdeen, United Kingdom

ARTICLE INFO

Keywords:
Convolutional neural networks
Recurrent neural networks
Deep learning
Perturbation identification
Perturbation localization
SIMULATE-3K
Self-supervised domain adaptation

ABSTRACT

The use of machine learning in the field of reactor safety and noise diagnostics has recently seen great potential given the advancements made in computational tools, hardware and noise simulations. In this work we demonstrate how deep neural networks, specifically recurrent and convolutional neural networks can be trained in a synthetic setting and aligned to operate on real plant measurements to recover perturbation type and origin location from time-series signals. We first utilize the vast quantities of synthetic data generated from the extended SIMULATE-3K codes, simulating a Swiss 3-loop pre-KONVOI reactor to train our networks under a variety of differing perturbation settings. Additionally, we extend these approaches to operate in the setting of unsupervised real plant measurements, where information about the true perturbation characteristics is unknown. As such, we show the applicability of a self-supervised domain adaptation approach to correctly align the representations learned by the neural network between both the synthetic and real detector readings to more concretely classify and localize perturbation. We validate our approaches under a number of experimental analyses showing successful performance in both simulated and synthetic domains.

1. Introduction

The continuous advance in the field of nuclear physics, aging of western fleets and the planned increase in population of nuclear power plants (NPPs) in Europe has led to an ever-growing necessity of non-intrusive core monitoring operations. The basic tasks of monitoring include managing the conditions of the core, ensuring the steady performance of the power plant and guaranteeing the overall safety of the system. Several attempts have been made over the past years towards the direction of detecting abnormal behavior in the core, predominately through measuring the neutron flux inside and outside of the core and then processing and analyzing the obtained signals (Ma and Jiang, 2011).

In this work, the aforementioned objective is being sought through the application of machine learning techniques and more specifically, through deep learning approaches that are employed in an effort to tackle the issue of identifying the type of perturbations that might be occurring within the core, as well as their source. For this task, time domain signals are processed through recurrent neural networks (RNNs) and convolutional neural networks (CNNs) (Goodfellow et al., 2016), with different architectures being evaluated in simulated perturbations produced by the Paul Scherrer Institute's (PSI) SIMULATE-3K code (Grandi, 2015; Chionis et al., 2017).

Extending from the simulated setting to the application on real operational reactor measurements, we additionally advocate the use of self-supervised domain adaptation (Sun et al., 2019; Durrant et al., 2021) to minimize the difference between simulated perturbations and plant measurements. Specifically, given the unsupervised nature of real-plant measurements, we utilize inherent structural properties common in both the simulated and real signals to align the embeddings learned by our deep neural networks without human annotated labels. We empirically show the alignment of these domains whilst maintain strong validation results in the simulated setting.

The paper is organized as follows. Section 2 overviews the literature regarding nuclear core monitoring, as well as, some machine learning methods used in similar tasks. Section 3 presents the source and structure of the simulated perturbations used in the experiments. In

* Corresponding author.
E-mail address: abdelhamid.dokhane@psi.ch (A. Dokhane).

https://doi.org/10.1016/j.anucene.2022.109373
Received 1 March 2022; Received in revised form 28 June 2022; Accepted 3 August 2022

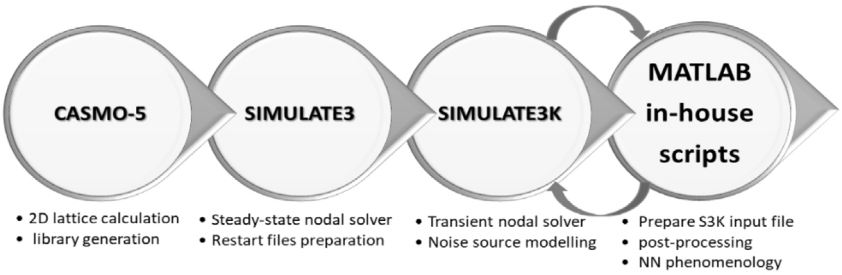

Fig. 1. PSI methodology for neutron noise modeling and analysis.

Section 4, a methodology for perturbation identification & localization based on CNNs & RNNs is presented and the main components of the system are analyzed, while in Section 5 perturbation identification is realized by one and two dimensional convolutions and self-supervised domain adaptation. Finally, Section 6 summarizes and discusses any possible future work resulting from this paper.

2. Related work

Anomaly detection plays a vital role in monitoring and safeguarding NPPs. In recent years, multiple works have been published in this research field. Many approaches of fault detection diagnosis regarding NPPs are summarized in Ma and Jiang (2011). The state-of-the-art methods can be divided in two main categories; (i) model-based techniques and (ii) model-free ones. In the latter case, the statistical properties of NPP measurements are studied in order to identify the presence of out-of-the-ordinary activity within the NPP, that could qualify as an anomaly, while in the former case, measurement data are used to infer more abstractive reasoning, known as *modeling*. The approaches presented in this work qualify as model-based techniques; nevertheless, other types of methods may exist, based on different types of NPPs and datasets.

In the majority of the relevant literature, signals of neutron flux measured in the nuclear core are utilized in order to identify various perturbations. More specifically, researchers employ model-based techniques that use either shallow or deep neural networks, in a similar fashion to the work presented in the current manuscript. Examples include (Maurya and Toshniwal, 2014), where support vector data description (Tax and Duin, 2004) has been employed in order to find anomalies in a dataset of NPP measurements. Neutron flux signals have also been used for the task of locating the source of the perturbation. Two recent approaches (Caliva et al., 2018; Ribeiro et al., 2018) employing deep learning (DL) techniques have been shown to be very efficient in detecting the perturbations' location; initially the original time-domain data are transformed to the frequency domain and are subsequently fed into a CNN (LeCun et al., 2015) that predicts possible anomalies. The authors additionally propose a clustering algorithm, used in conjunction with a denoising autoencoder, that successfully adapts to the peculiarities of the specific task.

Another important core monitoring task is to identify the perturbation type. Again, most approaches are model-based techniques, employing DL methods, like is the work presented in this manuscript. For example, in Tagaris et al. (2019), a methodology is proposed that transforms the time-domain signals into wavelet-based scaleograms. A deep CNN is subsequently used to classify the transformed signals into multiple categories. Additionally, random noise has been added to the signals, at various amplitudes, ensuring the generalization capabilities of the model and guaranteeing its robustness. Similarly to the aforementioned technique, in Tasakos et al. (2021), the original signals are used to produce spectrograms that compress useful information from long time series. The transformed data are provided to a multioutput unique neural network architecture, which is able to predict both the location and the type of perturbation. This approach has been tested in simulated perturbation, as well as actual plant measurements.

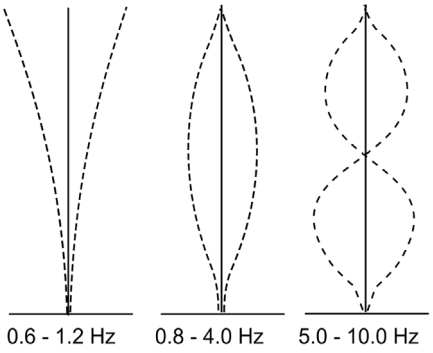

Fig. 2. Three modes of fuel assembly vibrations.

In Ioannou et al. (2021), a methodology is presented, where the single perturbation time-domain signals are used for training a one-vs-all scheme of a set of one dimensional (1D) CNN and long short-term memory (LSTM) neural networks. This framework is later utilized to identify multiple perturbations in core signals. A noise analysis is also carried out to improve the whole robustness of the proposed method. Another technique using time-domain signals is proposed in Pantera et al. (2022), where the employed datasets consist of simulations of the Czech VVER-1000 pressurized water reactor (PWR). The neural network-based model has been trained for perturbation localization and has also been evaluated on actual plant measurements.

3. Simulated perturbations

The simulated perturbations used in the context of the current work have been performed in the time-domain, using PSI methodology (Chionis et al., 2020) for neutron noise modeling and analysis, based on the commercial nuclear reactor codes CASMO5 (Ferrer, 2015), SIMULATE3 (Grandi, 2011) & SIMULATE-3K (Pohlus J and Girardin, 2018) (see Fig. 1).

Out of the various possible noise sources, a certain type of mechanical vibration of reactor internals, i.e. fuel assembly (FA) vibration is one of the most plausible one, which is evaluated here, along with thermal–hydraulic (TH) fluctuations of inlet coolant temperature and flow. The most realistic and significant modes of fuel assembly vibrations are cantilevered mode, where the fuel assembly is clamped-free at the top but fixed at the bottom; and the C-shaped and the S-shaped modes at, respectively, where the fuel assembly is fixed at both the top and the bottom (Thie, 1981). The three modes of fuel assembly vibrations are illustrated in Fig. 2.

The realistic fuel assembly vibrations are primarily modeled with the help of the delta-gap model of CASMO5 and the improved assembly vibration model of SIMULATE-3K (Chionis, 2020; Chionis et al., 2020). The model expresses the reactor anomalies or perturbations as

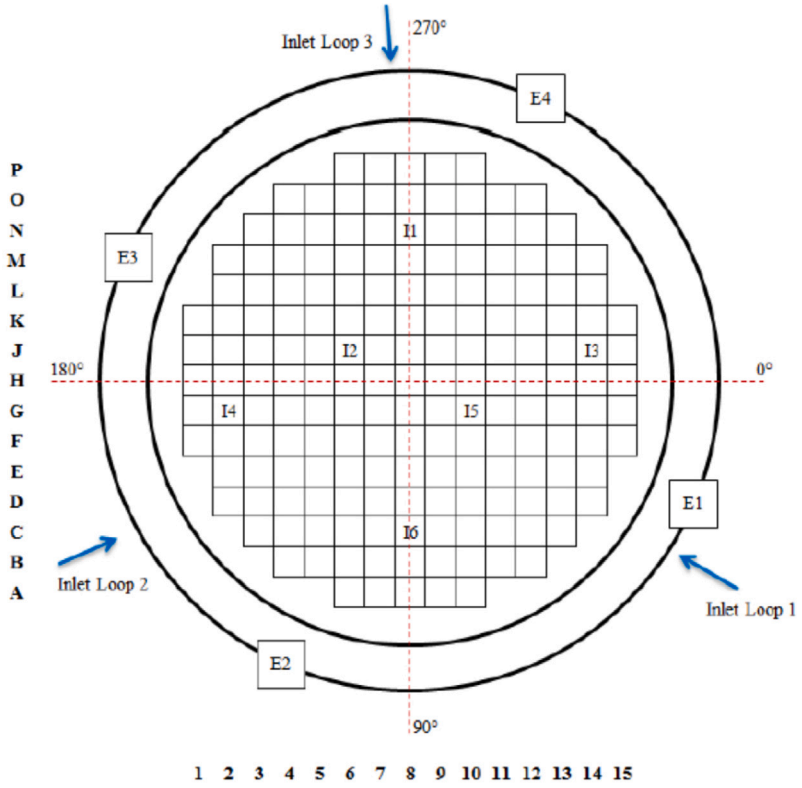

Fig. 3. Radial layout of the PWR core.

fluctuations of macroscopic cross sections. As a fuel assembly vibrates laterally, the surrounding water gap on either side of the oscillating assembly changes, i.e. there is an increase in water-gap in one-direction and an equal amount of decrease in the opposite direction. CASMO5 is employed to generate cross sections accommodating for the varying water gap widths using the "delta-gap model". CASMO5 performs an additional delta-gap branch calculation together with the default automated case matrix. Each fuel assembly is equally divided into axial nodes, which are further subdivided into 2×2 planar sub-nodes. The cross sections from CASMO5 are homogenized within each node and adapted into a readable library format for the codes SIMULATE3 and SIMULATE-3K. SIMULATE3 solves the two-group three-dimensional diffusion equation for every node corresponding to certain operating conditions of interest and produces restart files required by the downstream code SIMULATE-3K. The transient nodal code SIMULATE-3K imitates time-dependent vibrations of the fuel assemblies in x- and y-directions, using the assembly vibration model by modifying the water gap-widths between any two assemblies, based on the input characteristics and patterns of vibrations. This function is supported by in-house MATLAB-based supplementary scripts, which allow the user to impose dynamic water-gap widths, corresponding to different vibrational conditions in terms of choice of vibrating assemblies, vibration amplitude, phase and frequency and patterns such as synchronized or unsynchronized vibrations. Further options include choice of oscillation type such as random, step-wise, sinus patterns and predefined functions representative of the three vibration modes of the fuel assemblies (Verma et al., 2021). This is done by creating an input vector of the axial shape by assigning factored coefficients between zero and one to each axial node. The fuel assembly displaces each of the axial nodes by a width that is calculated using the coefficients and the water-gap widths at every time-step, thereby, simulating the vibration of the fuel assembly in a given axial pattern. With the help of the support file, SIMULATE-3K performs transient full core calculations to obtain three-dimensional, time-dependent, two-group fluxes and associated neutron noise.

The simulated perturbations comprise of neutron noise signals corresponding to various scenarios, which are calculated for a Swiss pre-KONVOI PWR (Pohlus J and Girardin, 2018). It is a 3-loop reactor containing 177 fuel assemblies and a set of 36 in-core and 8 ex-core detectors, as shown in Fig. 3, that supply the noise signals used for training, testing and validating the machine learning architectures. The in-core detectors are located in 6 axial and 6 radial locations, while the ex-core detectors are located in 4 radial and 2 axial locations. As the core instrumentation in such a PWR is limited, and fewer than the possible locations of the perturbations, machine learning techniques are required and explored over conventional signal unfolding techniques.

In this case, every noise scenario is simulated for core conditions of middle-of-cycle 39 (MOC39) and beginning-of-cycle 40 (BOC40). These conditions relate to the amount of fuel burned in the reactor. The first dataset for MOC39 contains 537 simulated perturbation scenarios, while the second dataset for BOC40 contains 358 perturbation scenarios. Each simulation scenario has a duration of 100 sec and a time-step of 0.01 sec, and includes signals corresponding to detector responses of both in-core and ex-core detectors. A time step of 0.01 s is a standard time step that captures the details of the dynamics of the coupling between neutronics and thermal hydraulics for the neutron noise case, since it is not a very fast transient. The choice of the duration (100 sec) on the other hand, ensures the convergence of the static solution and also to minimizes the statistical error of the results of processing the signal in the time and frequency domains. The specific scenarios considered are outlined on Table 1.

In case of scenarios S1-S3, each of the 177 fuel assemblies in the core is vibrated individually, thereby resulting in 177 subscenarios for each case. Simulations are performed at a nominal frequency of 1.2 Hz for S1 & S2 and of 5 Hz for S3, for a 1 mm displacement amplitude in the x-direction. Temperature fluctuations are at ± 1 °C for S4, while flow fluctuations are at $\pm 1\%$ for S5. The central 5×5 FA in scenarios S6-S9 vibrates at 1.2 Hz in the x-direction, with a 1 mm displacement amplitude. The co-occurring thermal–hydraulic fluctuations are ± 1 °C

Table 1
Simulated perturbations for MOC39 and BOC40.

Scenario	Description	MOC39	BOC40
S1	Single FA vibrating in cantilevered mode	yes	yes
S2	Single FA vibrating in C-shaped mode	yes	yes
S3	Single FA vibrating in S-shaped mode	yes	no
S4	Random fluctuations in inlet temperature	yes	yes
S5	Random fluctuations in inlet flow	yes	yes
S6	Simplistic lateral vibration of a central 5 × 5 FA cluster & random thermal–hydraulic fluctuations	yes	yes
S7	Cantilevered vibration of a central 5 × 5 FA cluster & random thermal–hydraulic fluctuations	yes	yes
S8	C-shaped vibration of a central 5 × 5 FA cluster & random thermal–hydraulic fluctuations	yes	no
S9	S-shaped vibration of a central 5 × 5 FA cluster & random thermal–hydraulic fluctuations	yes	no

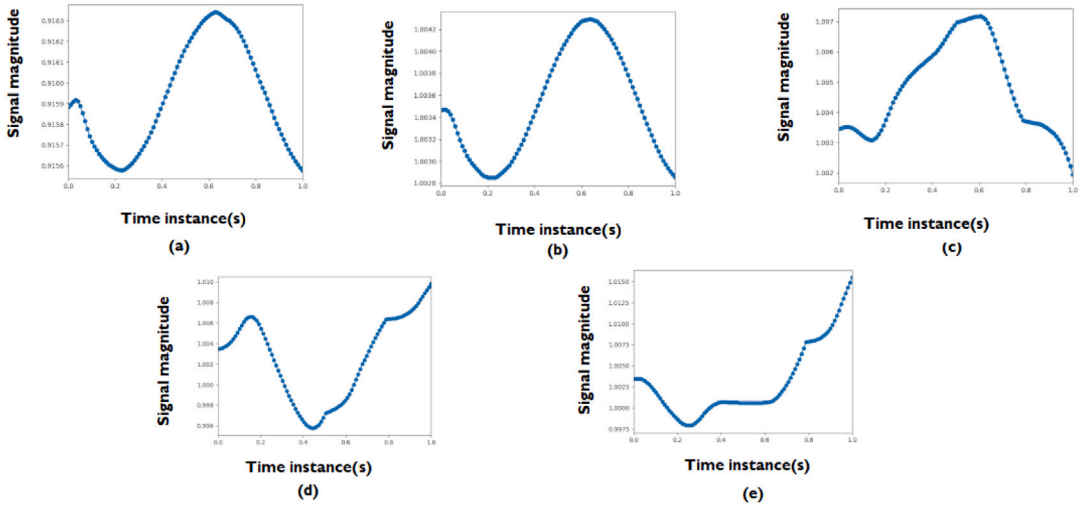

Fig. 4. Visualization of recorded sensor signals for different perturbations (a). Vibration of one FA in cantilevered mode (b). Vibration of one FA in S-shaped mode (c). Random fluctuation in inlet temperature (d). Random fluctuation in inlet flow (e). Simplistic lateral vibration of central 5 × 5 FA & thermal–hydraulic fluctuations.

in temperature and ±2% in flow and they are introduced synchronously in the three loops.

The simulated perturbations are further processed for evaluation by the machine learning models. A sliding window method is applied for data augmentation on both datasets, with a step of 1 sec and an overlap rate of 50%, which yields 31,524 training samples, 32,043 validation samples and 32,043 test samples for MOC39 (the respective sizes for BOC40 are 20,982–21,326 and 21,326 respectively). Finally, Fig. 4 displays sensor recording examples for some perturbations.

4. CNN & RNN networks for perturbation identification and localization based on temporal signals

In order to both identify and localize perturbations given temporal signals recorded by in-core and ex-core neutron detectors combinations of CNN & RNN networks have been developed. Initially, a recurrent neural network is applied to extract representative features by exploiting the temporal dependencies between samples; the extracted features are then refined by a convolutional neural network which aims to learn the spatial dependency between the detectors in the core to assist localization and are finally fed into a fully connected neural network to make regression and softmax classification and source localization. The overall approach is depicted in Fig. 5.

4.1. Long Short-Term Memory units

RNNs can capture temporal dependencies between a sequence of data for a variety of tasks (e.g., sequence classification, temporal series forecasting, etc.) (Goodfellow et al., 2016). In the context of the current work, RNNs are exploited to model the temporal dependencies between the (temporal) input signals. Traditional RNNs suffer from both gradient vanishing and gradient explosion (Goodfellow et al., 2016), which limit their efficiency, especially when processing long sequential data. Currently, the most popular way to overcome the aforementioned limitation is to adopt a new architecture incorporating LSTM units, which are a special type of a recurrent network and are also used in this work (Sherstinsky, 2020).

As illustrated in Fig. 5, a number of distinct LSTM units are included in the proposed model. There are a series of arithmetic operations associated with each of them, summarized in the following set of Equations

$$
\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i)\\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f)\\
c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)\\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o)\\
h_t &= o_t \tanh(c_t)
\end{aligned}
$$

where x_t, o_t and h_t represent LSTM input, output and state associated with the data sample at time instance t, c_t is the LSTM cell value representing encoded historical information obtained from previous data samples before t, i_t is the activation vector of the input/update gate, f_t is the activation vector of the forget gate, $\sigma(\cdot)$ and $\tanh(\cdot)$ represent sigmoid and hyperbolic tangent functions, respectively. Other parameters represent weights and biases.

The LSTM-based model encodes the input sequence into a series of feature vectors, which incorporate the temporal dependency information between consecutive time instances in this sequence. These

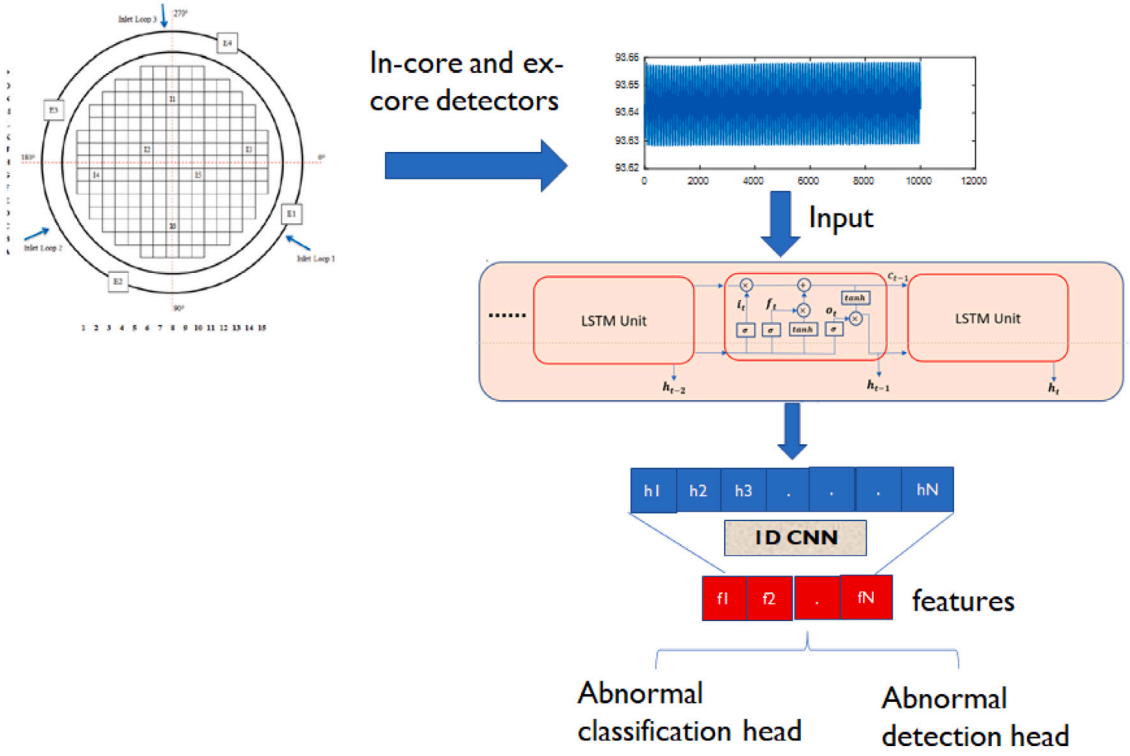

Fig. 5. CNN/RNN-based approach for perturbation identification and localization.

extracted feature vectors are then fed into the convolutional neural network for further processing.

4.2. One dimensional convolutional neural network

The output of the previous stage is subsequently fed into a 1D-CNN for further refinement. The 1D-CNN used in our work contains multiple one dimensional convolutional layers, with each of them performing Eq. (1) below

$$O_t^k = f(\sum_{i,j} w_k^{i,j} I_{t+i}^j + b) \tag{1}$$

where O_t^k represents the kth output corresponding to the time instance t, I_t^j represents the jth input at t, $w_k^{i,j}$ and b represent the convolutional kernel weight and bias respectively and $f(\cdot)$ represents an activation function (e.g., rectified linear unit, sigmoid, softmax, etc.).

The output of the 1D-CNN is fed into a feed-forward network with two heads for perturbation identification and localization, respectively. Network training is performed using the stochastic gradient descend algorithm, which minimizes the class-weighted loss function of Eq. (2)

$$L = -\frac{1}{M}\sum_{i=1}^{M} c_i \{\frac{\lambda_1}{C}\sum_{j=1}^{C}[y_i^j \log(\hat{y}_i^j) + (1-y_i^j)\log(1-\hat{y}_i^j)] - \lambda_2 \|y_i^l - \hat{y}_i^l\|\} \tag{2}$$

where C and M are the class number and total temporal sample number, respectively, y_i^j and y_i^l represent the class and localization for the ith sample and c_i represents the class weight associated with the ith sample, which is inversely proportional to the number of samples in the class.

4.3. Experimental results

The proposed architecture is evaluated on the datasets and scenarios outlined in Section 3. As it is common in supervised learning tasks, the networks are trained on the training dataset, hyper-parameter values are determined by the validation dataset and finally tested and results are reported on the test set. Table 2 summarizes the performance of perturbation classification (in terms of *classification accuracy* for distinguishing between different perturbation scenarios) and localization (in terms of the *root mean square error* - RMSE) by different architectures on the test sets. The gated recurrent units (GRUs) (Cho et al., 2014) are an extension of LSTMs that add a forget gate, but overall have fewer parameters, since they lack an output gate. Multiple evaluation trials have been made and the mean of the respective metrics, accuracy and RMSE, are calculated. Accuracy is used for measuring the performance of perturbation identification and is defined as the fraction of the correctly identified perturbations over all identified perturbations. RMSE, on the other hand, is used for those perturbations where localization is also important (e.g. single vibrating FA) and is defined according to Eq. (3)

$$RMSE = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - d_i)^2} \tag{3}$$

where M is the total number of training instances, y_i is the predicted location of the ith perturbation and d_i is the actual location of ith perturbation. Therefore, a 99% classification accuracy designates that the system is able to classify correctly 99% of the perturbation types. In case of perturbation localization, a RMSE of 1 means that the predicted location of the vibrating FA is, on average, one FA away from the actual location (Fig. 8). In total, the results demonstrate that, compared to other models, the combined CNN/RNN based approaches always achieve the most accurate performance for classification with the smallest RMSE for localization, on both datasets.

In order to evaluate the robustness of each system, white noise has been introduced to the simulated perturbation signals (Fig. 6). Table 3 summarizes the performance of the perturbation classification (in terms of accuracy) and localization (in terms of RMSE) under added white noise, for the same models examined on Table 2.

From Tables 2–3, we can see that the CNN-RNN framework can always achieve the best performance for perturbation classification and localization, either under the presence or the absence of white noise in detector signals. Nevertheless, we observe that the added white noise

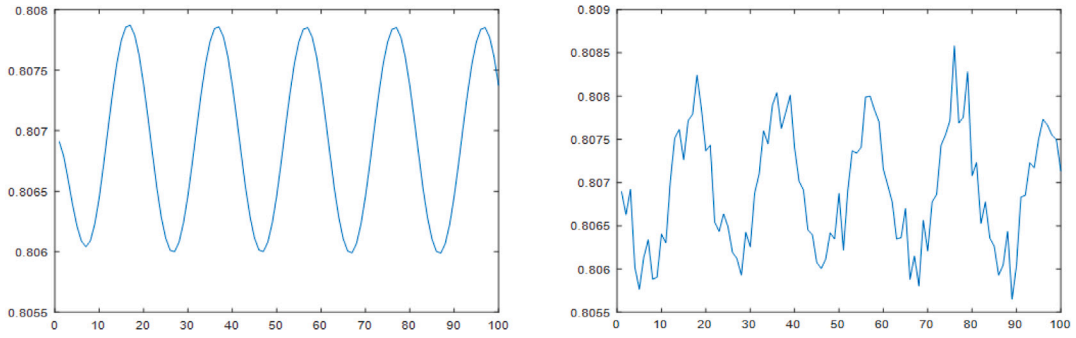

Fig. 6. The original signal (left) and the signal after adding noise (right).

Table 2
Classification accuracy and localization RMSE obtained by different methodologies on the MOC39 & BOC40 test datasets (Section 3).

Models	BOC40		MOC39	
	Accuracy	RMSE	Accuracy	RMSE
LSTM	93.72% ± 0.19%	2.39 ± 0.04	85.68% ± 1.70%	1.92 ± 0.17
GRU	98.45% ± 0.23%	1.76 ± 0.13	98.09% ± 0.27%	1.49 ± 0.13
CNN	99.90% ± 0.10%	0.92 ± 0.07	99.38% ± 0.12%	0.86 ± 0.09
CNN + LSTM	99.90% ± 0.10%	0.39 ± 0.05	99.59% ± 0.17%	0.45 ± 0.04
CNN + GRU	99.90% ± 0.10%	0.47 ± 0.05	99.46% ± 0.14%	0.44 ± 0.03

Table 3
Classification accuracy and localization RMSEs obtained by different methodologies on the MOC39 & BOC40 test datasets (Section 3) with added white noise.

Models	BOC40		MOC39	
	Accuracy	RMSE	Accuracy	RMSE
LSTM	86.04% ± 0.31%	3.51 ± 0.06	89.38% ± 1.61%	3.78 ± 0.11
GRU	85.60% ± 0.48%	3.66 ± 0.08	89.08% ± 0.64%	3.92 ± 0.10
CNN	84.56% ± 0.55%	4.09 ± 0.05	90.75% ± 0.27%	4.23 ± 0.08
CNN + LSTM	88.59% ± 0.37%	3.24 ± 0.06	93.12% ± 0.17%	3.02 ± 0.04
CNN + GRU	88.30% ± 0.25%	3.22 ± 0.04	92.96% ± 0.22%	3.24 ± 0.06

Table 4
Classification accuracy obtained for different loss functions on the MOC39 test dataset (Section 3).

Scenario	Loss without class weight	Loss with class weight
S1	99.88% ± 0.04%	99.92% ± 0.02%
S2	99.97% ± 0.01%	99.96% ± 0.01%
S3	53.32% ± 9.18%	58.80% ± 7.38%
S4	99.71% ± 0.57%	99.58% ± 0.65%
S5	38.86% ± 19.10%	95.62% ± 1.48%
S6	80.04% ± 27.76%	95.05% ± 2.21%
S7	16.90% ± 16.89%	92.82% ± 4.34%
S8	99.99% ± 0%	99.99% ± 0%
S9	55.61% ± 9.34%	54.03% ± 7.90%
Overall	99.38% ± 0.12%	99.73% ± 0.02%

affects the BOC40 dataset to a greater extend, compared to the MOC39 dataset. This difference in performance is attributed to the fact that the BOC40 training sample size is smaller (~66% of the training sample size of MOC39, according to Section 3) and therefore the models are not as robust to noise, when trained on this dataset. Next, we evaluate the classification performance of the CNN/RNN models under different loss functions, which include: (i) a class weighted loss function as in Eq. (1) and (ii) a loss function without class weight (c_i term in Eq. (2)).

The results of this experiment on the MOC39 test dataset are summarized on Table 4. It can be seen that the weighted loss outperforms the non-weighted one for classifying the majority types of perturbations. The overall classification accuracy (both mean and standard deviation) is improved by adopting the weighted loss. Moreover, the classification accuracy is drastically improved for classes with few data samples (such as S5 and S7).

5. Perturbation localization based on one and two dimensional convolutions and self-supervised domain adaptation

The second approach considered in this work is to directly encode time-domain detector signals using one and two dimensional CNNs for perturbation localization. The introduction of 2D CNNs acts as a "bridge" between previous implementations (Durrant et al., 2021) using 3D CNNs and frequency analysis for localizing multiple perturbations with the 2D CNN implementation and results being provided as a point of reference for the evaluation of the localization performance. The detail of the analysis is at the fuel assembly level; that is, we are interested in identifying the radial location (or the vicinity) of a single vibrating fuel assembly, therefore only the relevant scenarios of Section 3 have been considered (single fuel assembly vibrating in cantilevered, C-shaped or S-shaped mode, i.e. S1-S3). Prior to encoding both simulated perturbations and plant measurement signals, they have been checked for the existence of linear trend and have been detrended, wherever trend was identified.

5.1. Using convolutions for perturbation localization

The detrended signals have been subsequently encoded using four distinct convolutional architectures; the first three originating from the literature, while the fourth has been developed in the context of the current work. The first two have been developed in the framework of vibration signal analysis of mechanical parts (Chen and Lee, 2021).

The first architecture encodes the one dimensional detector signal using two dimensional convolutional operations. In order to achieve this, the spectrogram of each signal is firstly computed, which is a 2D structure. Then, four convolutional layers (Fig. 7(a)) are applied on the spectrograms of varying filter size (4, 8, 16 and 32, respectively) and kernel size (9 × 9 for the first two, 4 × 4 for the latter two), but with a fixed stride (2 × 2). All encoded detector signals are provided as input to a fully connected architecture comprised of two hidden layers (of 64 and 32 neurons, respectively and with the rectified linear unit as the activation function). The output layer predicts the radial location of the single vibrating fuel assembly and its neurons have the softmax activation function.

The second one encodes each detector signal through the successive application of four one dimensional convolutional filters of size 8, 16, 32 and 64, as depicted in Fig. 7(b). The kernel size is 30 and the stride is set at 1. Then all encoded detector signals are provided as input to a fully-connected feed-forward neural network that is comprised of two hidden layers; the first one having 128 neurons and the second one 32. All neurons of both layers have the rectified linear unit as the activation function, while the output layer neurons (that predict the vibrating fuel assembly) have the softmax activation function.

The third architecture to be considered has also been developed in the framework of vibration signal analysis for fault diagnosis (Zhang et al., 2017). It consists of five successive convolutional (of increasing filter size — 16, 32 and 64) and pooling layers of fixed width (Fig. 7(c)).

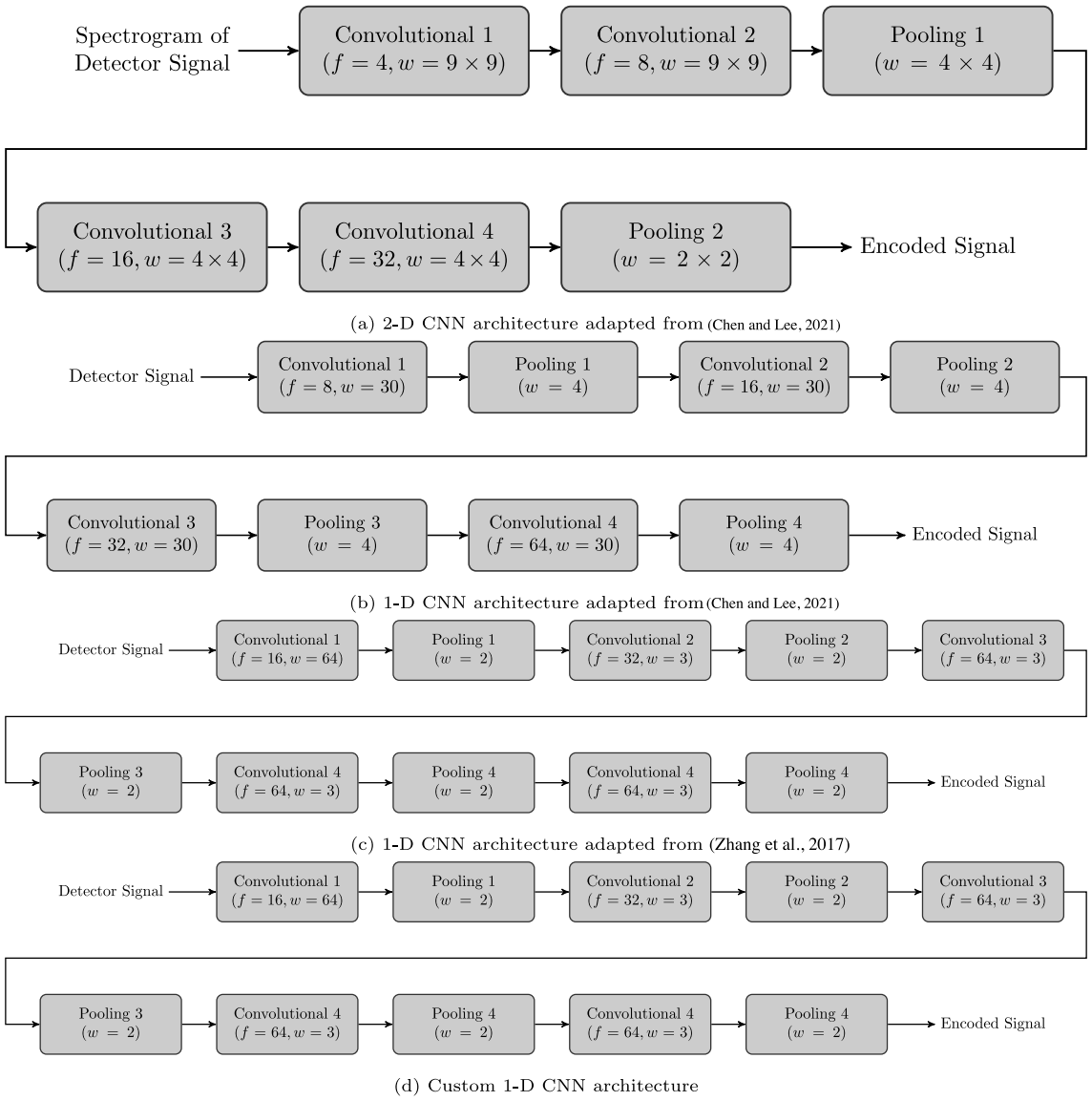

Fig. 7. Convolutional architectures for encoding detector signals.

The size of first convolutional kernel is 64, while the rest are of size 3. After each convolutional layer and fully-connected layer, batch normalization is used to improve system performance. The encoded signals are then provided to a fully-connected feed-forward neural network comprised of a single hidden layer of 100 neurons and an output layer with sofmax activation functions that performs the prediction task.

Finally, the fourth architecture (Fig. 7(d)) is proposed in the framework of the current work. It is a merge between the previous architectures (Figs. 7(b) and 7(c)), simplifying the (Fig. 7(c)) as it contains one fewer convolutional and pooling layer, with all other elements remaining the same.

All of the aforementioned architectures have been trained on simulated perturbations pertaining to a single fuel assembly vibrating in cantilevered, C-shaped and S-shaped mode for both of the examined cycles in Section 3 (MOC39 and BOC40). Each model has been trained for 150 epochs, with a batch size of 32, using the Adam optimizer (Kingma and Ba, 2015). Since the systems' output is the radial location of the vibrating fuel assembly, the loss function used is the RMSE, which quantifies the distance between the predicted fuel assembly and the actually vibrating one. Table 5 summarizes the results of the examined architectures for the studied perturbations during the two cycles.

From the table above, it can be seen that all of the considered architectures can identify the vicinity of the occurring single fuel assembly perturbation for all scenarios on both datasets. More specifically, the proposed custom 1D-CNN architecture outperforms the other 1D-CNN approaches on the majority of the scenarios in both the MOC39 and the BOC40 datasets while at the same time being on par with the 2D-CNN approach on the MOC39 S2 dataset. Finally, Fig. 8 below visualizes an example prediction for the S2 scenario on the MOC39 dataset.

5.2. Self-supervised domain adaptation

We have demonstrated in the previous sections the capability of machine learning, specifically deep recurrent neural networks and 1-dimensional convolutional neural networks to classify and localize perturbations in the simulated setting. However, as identified in Durrant et al. (2021) it is not applicable to simply make predictions on real plant measurements with these networks trained on solely synthetic data. Although the simulation tools used provided data that is extremely accurate to the real-plant setting, there is inevitably some differences that occur that make it difficult for our network to interpret, this is referred to as domain shift (Luo et al., 2019). In practice, neural networks will learn representations of the input signals that

Table 5
Localization accuracy (in RMSE) for different single FA perturbations and test datasets.

Dataset	Scenario	2D-CNN (Chen and Lee, 2021)	1D-CNN (Chen and Lee, 2021)	1D-CNN (Zhang et al., 2017)	Custom
MOC39	S1	1.81 ± 0.04	3.17 ± 0.14	2.20 ± 0.28	$2,63 \pm 0.45$
MOC39	S2	1.75 ± 0.20	2.81 ± 0.33	2.19 ± 0.28	1.90 ± 0.11
MOC39	S3	1.63 ± 0.16	3.19 ± 0.35	2.69 ± 0.34	2.28 ± 0.41
BOC40	S1	2.15 ± 0.38	3.35 ± 0.11	3.30 ± 0.43	3.44 ± 0.27
BOC40	S2	2.02 ± 0.46	3.10 ± 0.18	2.51 ± 0.44	2.42 ± 0.46

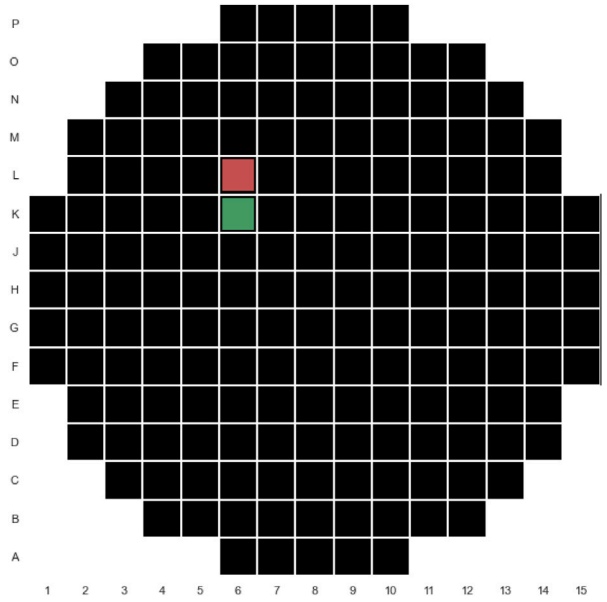

Fig. 8. Radial cut of the reactor using simulated scenarios. The (single) vibrating FA is located at K06 (shown in green), while the prediction is for L06 (shown in red).

only correspond to the data it is trained on (i.e. the simulated data), when given a signal from a different distribution – for example our real plant measurements – the network will produce a representation that occupies a different region in the embedding space. Therefore, to make accurate predictions in the real-plant setting we need to adapt our network to align its representations to lie in the known space occupied by the simulated signals.

To achieve this we employ the use of the self-supervised domain adaptation (SSDA) (Sun et al., 2019) approach presented in Durrant et al. (2021) in which we aim to align the representations of synthetic and real domains without the need for human annotations, which are near impossible to obtain for real measurements. Instead, we utilize auxiliary tasks alongside our main task (perturbation classification) to align the representations by optimizing common objectives between domains. The auxiliary tasks are constructed from the data itself without the need for human annotation, and are defined to specifically assist the network to capture representations that are known to be common between both domains, i.e. structural or meta-data. Essentially, we train our network to predict identical outputs for the auxiliary tasks given both a simulated and real sample which in tern constructs a mapping from input to output which is largely identical for both domains. We employ an auxiliary task where we randomly remove a portion of the input signal across all detectors and task the network to predict the missing portion of the signal.

The architectural design of the self-supervised domain adaptation is depicted in Fig. 9, where the 1-dimensional CNN is given both a simulated and real plant signal which has been augmented randomly according to the auxiliary task setting aforementioned, its output is then given to the auxiliary tasks and the simulated only given to the main classification task. The loss function to simultaneously optimize is then a weighted sum of all the tasks, this is given as follows,

$$\mathcal{L}(y_s, y_{\text{aux}}, \hat{y}_s, \hat{y}_{\text{aux}}) = \lambda_{\text{CE}} \cdot \mathcal{L}_{\text{CE}}(y_s, \hat{y}_s) + \lambda_{\text{aux}} \cdot \mathcal{L}_{\text{aux}}(y_{\text{aux}}, \hat{y}_{\text{aux}}) \tag{4}$$

where y and $\hat{y}$ represent the predicted and target value respectively, $\mathcal{L}_{\text{aux}}$ refers to the auxiliary task, $\mathcal{L}_{\text{CE}}$ denotes the main softmax cross entropy classification task, λ_{CE} and λ_{aux} are multiplicative weightings of each task, and the subscripts s, aux denote the input and output for the main classification task and auxiliary task. At inference time only the main classification task is employed disregarding the auxiliary task and the augmentations that are associated with it.

5.2.1. Experimental results

The self-supervised domain adaptation methodology has been trained in a similar manner to that described in Section 5.1, although with the 1-dimensional CNN network being pre-trained (initialized with the parameters from training on only simulated data from Section 5.1) and then fine-tuned via the self-supervised domain adaptation procedure. This fine-tuning allows full leverage of the simulated data, to provide better downstream classification performance. In order for both datasets (real and simulated) to be in accordance, the real measurements gathered from the 3-loop Swiss pre-KONVOI reactor have been re-sampled and detector usage has been aligned to be the same.

The aforementioned network has been trained on both simulated perturbations and real plant measurements of the 3-loop Swiss pre-KONVOI BOC40, using the Adam optimizer (Kingma and Ba, 2014) for 100 epochs and with a learning rate of 0.001. We select the weighting factors empirically to be $\lambda_{\text{CE}} = 0.5$ and $\lambda_{\text{aux}} = 0.5$, equally balancing the optimization between the classification task and the alignment of the representations. In Fig. 10, we demonstrate how the network correctly minimizes the error of both the classification and auxiliary task losses during training, under both the simulated and real data. This promising finding shows how our network is correctly able to learn the underlying common features regardless of domain, giving us a good indication of alignment of the learned representations.

The trained model has shown to produce aligned representations through the method of SSDA, which results in a model that can be employed on the plant measurements to make predictions to be used as an indicator of possible perturbation located in the core. We provide the predictions on the real plant measurements in Fig. 11, identifying the fuel assembly vibration scenario being present at J5. Although not fully validated given the lack of human annotations, these findings are promising and further validation through traditional signal processing means provides potential for future work.

6. Conclusion

We show in this work the capabilities of machine learning methods to appropriately uncover temporal and spatial patterns enabling the detection of reactor perturbation type, and origin location from time-series neutron detector readings. We focus on training our neural networks with large quantities of synthetic time-series signals produced by the extended SIMULATE-3K codes, under a number of different perturbation scenarios. Specifically, a 3-loop Swiss pre-KONVOI reactor is modeled, which we use to validate our methodologies under the simulated setting before making extensions to most applicably and

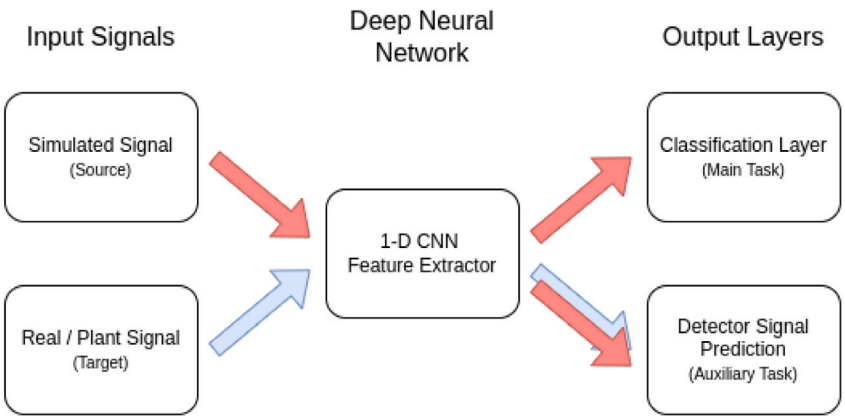

Fig. 9. Visual depiction of the self-supervised domain adaptation training procedure. Both the real and simulated signals are passed to the 1-D CNN feature extractor producing a feature representation of each input. The representations from both domains are passed to the auxiliary task network to compute the auxiliary loss, while only the simulated representation with corresponding target annotations are passed to the classification network.

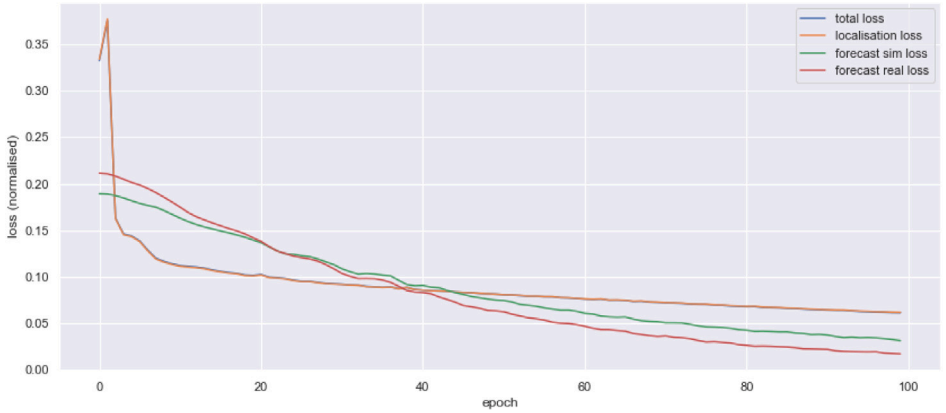

Fig. 10. Normalized train losses on the combined S1 and S2 data (BOC40).

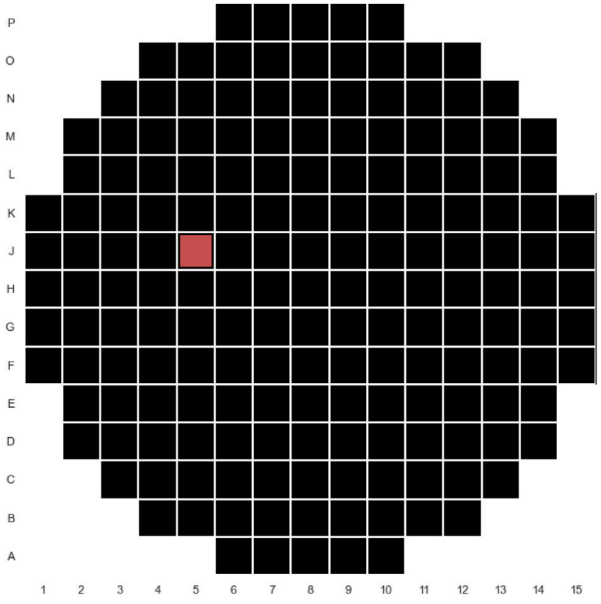

Fig. 11. Prediction (indication) of possible vibrating FA provided from the 1D-CNN, fine-tuned via the SSDA method on plant measurements from BOC40 (shown in red).

accurately make predictions on real plant measurements of the same reactor architecture.

Furthermore, we employ the use of domain adaptation methods to help mitigate the inevitable phenomena of domain shift between the real and synthetic plant measurements. Specifically, our method aims to align the embeddings learned by the network in a self-supervised fashion given the absence of annotations (true perturbation type and location) in the real-plant setting meaning supervised alignment and training is not possible. Our self-supervised method achieves this alignment by learning common features present in both the synthetic and real data through structural and non-semantic tasks that do not require human annotations. We demonstrate promising results in both the simulated pre-training, showing the capability of machine learning methods to unfold reactor transfer functions given time-series neutron detector readings. Additionally, we also provide promising results that such methods pre-trained on simulated data can provide promising predictions in the real plant setting through means like self-supervised domain adaptation.

Fundamentally, our proposed methodologies demonstrate how machine learning systems could potentially help provide support to nuclear safety engineers and operators via early alerts and perturbation characteristic retrieval. However, our work has certain limitations, such as lack of real plant data and corresponding annotations, which is something that will form our future work. We also plan to work on retrieving additional perturbation characteristics and origin location in real plant setting, and to experiment with additional auxiliary tasks (e.g. removal and subsequent prediction of missing detector and its signal) to improve alignment. Lastly, the interpretability of our AI systems via uncertainty measurements (Bhatt et al., 2021), explainable reasoning (Linardatos et al., 2021) and accountability (Naja et al., 2021) are necessary components for the wide-spread approval and

adoption by nuclear operators as well, providing solid foundations for future work.

CRediT authorship contribution statement

Antonios Papaoikonomou: Investigation, Methodology, Validation, Visualization, Writing – original draft. **James Wingate:** Investigation, Validation, Visualization, Writing – original draft. **Vasudha Verma:** Investigation, Methodology, Validation, Visualization, Writing. **Aiden Durrant:** Investigation, Methodology, Validation, Visualization, Writing. **George Ioannou:** Investigation, Methodology, Validation, Visualization, Writing. **Tasos Papagiannis:** Investigation, Methodology, Validation, Visualization, Writing. **Miao Yu:** Investigation, Validation, Visualization, Writing. **Georgios Alexandridis:** Funding acquisition, Conceptualization, Supervision, Data curation, Investigation, Methodology, Validation, Visualization, Writing – original draft. **Abdelhamid Dokhane:** Funding acquisition, Supervision, Validation, Visualization, Writing – review & editing. **Georgios Leontidis:** Funding acquisition, Supervision, Writing – review & editing. **Stefanos Kollias:** Funding acquisition, Writing – review & editing. **Andreas Stafylopatis:** Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

The research conducted has been made possible through funding from the Euratom research and training programme 2014–2018 under grant agreement No. 754316 for the "CORe Monitoring Techniques And EXperimental Validation And Demonstration (CORTEX)" Horizon 2020 project, 2017–2021.

References

Bhatt, U., Antorán, J., Zhang, Y., Liao, Q.V., Sattigeri, P., Fogliato, R., Melançon, G., Krishnan, R., Stanley, J., Tickoo, O., et al., 2021. Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In: Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society. pp. 401–413.

Caliva, F., De Ribeiro, F.S., Mylonakis, A., Demaziere, C., Vinai, P., Leontidis, G., Kollias, S., 2018. A deep learning approach to anomaly detection in nuclear reactors. In: 2018 International Joint Conference on Neural Networks. IJCNN, IEEE, pp. 1–8.

Chen, H.-Y., Lee, C.-H., 2021. Deep learning approach for vibration signals applications. Sensors 21 (11), http://dx.doi.org/10.3390/s21113929, URL https://www.mdpi.com/1424-8220/21/11/3929.

Chionis, D., 2020. Development of Advanced Methodologies for Monitoring and Modelling of Neutron Noise in Modern LWR Cores. EPFL, Lausanne, p. 228. http://dx.doi.org/10.5075/epfl-thesis-10219, URL http://infoscience.epfl.ch/record/279775.

Chionis, D., Dokhane, A., Belblidia, L., Ferroukhi, H., Girardin, G., Pautz, A., 2020. Development and verification of a methodology for neutron noise response to fuel assembly vibrations. Ann. Nucl. Energy 147, 107669. http://dx.doi.org/10.1016/j.anucene.2020.107669, URL http://www.sciencedirect.com/science/article/pii/S0306454920303674.

Chionis, D., Dokhane, A., Belblidia, L., Pecchia, M., Girardin, G., Ferroukhi, H., Pautz, A., 2017. SIMULATE-3K analyses of neutron noise response to fuel assembly vibrations and thermal-hydraulics parameters fluctuations. In: M&C 2017-International Conference on Mathematics & Computational Methods Applied To Nuclear Science & Engineering, At Jeju, Korea. pp. 289–297.

Cho, K., van Merrienboer, B., Bahdanau, D., Bengio, Y., 2014. On the properties of neural machine translation: Encoder-decoder approaches. CoRR arXiv:1409.1259.

Durrant, A.M., Leontidis, G., Kollias, S., Torres, A., Montalvo, C., Mylonakis, A., Demaziere, C., Vinai, P., 2021. Detection and localisation of multiple in-core perturbations with neutron noise-based self-supervised domain adaptation. In: The International Conference on Mathematics and Computational Methods Applied To Nuclear Science and Engineering.

Ferrer, R., 2015. CASMO-5 Methodology Manual. Tech. rep., (SSP-08/405 - Rev 4).

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, http://www.deeplearningbook.org.

Grandi, G., 2011. SIMULATE-3K Models & Methodology. Tech. rep., (SSP-98/13 Rev. 7).

Grandi, G., 2015. SIMULATE-3K Input specification (SSP-98/12, Rev. 17). Waltham, USA. Studsvik Scandpower Inc.–2013.

Ioannou, G., Tagaris, T., Alexandridis, G., Stafylopatis, A., 2021. Intelligent techniques for anomaly detection in nuclear reactors. EPJ Web Conf. 247, 21011. http://dx.doi.org/10.1051/epjconf/202124721011.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. URL http://arxiv.org/abs/1412.6980.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436.

Linardatos, P., Papastefanopoulos, V., Kotsiantis, S., 2021. Explainable ai: A review of machine learning interpretability methods. Entropy 23 (1), 18.

Luo, Y., Zheng, L., Guan, T., Yu, J., Yang, Y., 2019. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2507–2516.

Ma, J., Jiang, J., 2011. Applications of fault detection and diagnosis methods in nuclear power plants: A review. Prog. Nucl. Energy 53 (3), 255–266. http://dx.doi.org/10.1016/j.pnucene.2010.12.001, URL http://www.sciencedirect.com/science/article/pii/S0149197010001769.

Maurya, C.K., Toshniwal, D., 2014. Anomaly detection in nuclear power plant data using support vector data description. In: Proceedings of the 2014 IEEE Students' Technology Symposium. IEEE, pp. 82–86.

Naja, I., Markovic, M., Edwards, P., Cottrill, C., 2021. A semantic framework to support AI system accountability and audit. In: European Semantic Web Conference. Springer, pp. 160–176.

Pantera, L., Stulík, P., Vidal-Ferràndiz, A., Carreño, A., Ginestar, D., Ioannou, G., Tasakos, T., Alexandridis, G., Stafylopatis, A., 2022. Localizing perturbations in pressurized water reactors using one-dimensional deep convolutional neural networks. Sensors 22 (1), http://dx.doi.org/10.3390/s22010113, URL https://www.mdpi.com/1424-8220/22/1/113.

Pohlus J, U.P.J., Girardin, G., 2018. Report about the measurement results of Goesgen. Tech. rep., CORTEX deliverable D4.1.

Ribeiro, F.D.S., Calivá, F., Chionis, D., Dokhane, A., Mylonakis, A., Demaziere, C., Leontidis, G., Kollias, S., 2018. Towards a deep unified framework for nuclear reactor perturbation analysis. In: 2018 IEEE Symposium Series on Computational Intelligence. SSCI, IEEE, pp. 120–127.

Sherstinsky, A., 2020. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D 404.

Sun, Y., Tzeng, E., Darrell, T., Efros, A.A., 2019. Unsupervised domain adaptation through self-supervision. arXiv preprint arXiv:1909.11825.

Tagaris, T., Ioannou, G., Sdraka, M., Alexandridis, G., Stafylopatis, A., 2019. Putting together wavelet-based scaleograms and convolutional neural networks for anomaly detection in nuclear reactors. In: Proceedings of the 2019 3rd International Conference on Advances in Artificial Intelligence. ICAAI 2019, Association for Computing Machinery, New York, NY, USA, pp. 237–243. http://dx.doi.org/10.1145/3369114.3369121.

Tasakos, T., Ioannou, G., Verma, V., Alexandridis, G., Dokhane, A., Stafylopatis, A., 2021. Deep learning-based anomaly detection in nuclear reactor cores. In: Proceedings of the International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering (M&C 2021), Online. pp. 2026–2037.

Tax, D.M., Duin, R.P., 2004. Support vector data description. Mach. Learn. 54 (1), 45–66.

Thie, J.A., 1981. Power reactor noise. URL https://www.osti.gov/biblio/5239676.

Verma, V., Chionis, D., Dokhane, A., Ferroukhi, H., 2021. Studies of reactor noise response to vibrations of reactor internals and thermal-hydraulic fluctuations in PWRs. Ann. Nucl. Energy 157, 108212. http://dx.doi.org/10.1016/j.anucene.2021.108212, URL https://www.sciencedirect.com/science/article/pii/S0306454921000888.

Zhang, W., Peng, G., Li, C., Chen, Y., Zhang, Z., 2017. A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. Sensors 17 (2), http://dx.doi.org/10.3390/s17020425, URL https://www.mdpi.com/1424-8220/17/2/425.