Rotor dynamics informed deep learning for detection, identification, and localization of shaft crack and unbalance defects

Weikun Deng, Khanh T.P. Nguyen, Kamal Medjaher, Christian Gogu, Jérôme Morio

Rotor dynamics informed deep learning for detection, identification, and localization of shaft crack and unbalance defects

Weikun Deng[a], Khanh T. P. Nguyen[a], Kamal Medjaher[a], Christian Gogu[b], Jérôme Morio[c]

[a]*Laboratoire Génie de Production, LGP, Université de Toulouse, INP-ENIT, 47 Av. d'Azereix, 65016, Tarbes, France.*

[b]*Institut Clément Ader (ICA), Université de Toulouse, ISAE-SUPAERO, UPS, CNRS, INSA, Mines Albi, 3 rue Caroline Aigle, 31400 Toulouse, France*

[c]*ONERA/DTIS, Université de Toulouse, F-31055 Toulouse, France.*

Abstract

In this paper, a new convolution long short-term memory networks model (CNN-LSTM), rotor finite element mimetic neural network (RFEMNN), is proposed and used for the diagnostics of rotor unbalance and shaft crack faults. RFEMNN aims to accomplish the recognition and localization of faults in the rotor structure. In this context, "Mimetic" refers to two levels: the topology of the structural division of the rotor finite element model through custom layers design and the topology of finite element solution process data flow by designing interlayer connections. The "Mimetic" theory is a new paradigm of physics-informed structure that enhances the physics consistency of machine learning (ML) and does not require complete and analytic physical knowledge with all known parameters. To train RFEMNN, this paper proposes a multi-label and multi-task supervised learning approach with one-hot encoded fault type labels, fault location labels, and vibration behavior labels. These labels are also involved in the training process of other tasks through the proposed physics-informed structure. The effectiveness of the proposed model is validated through a series of experimental platform tests on different rotor layouts and fault combination conditions. Several evaluation metrics are proposed to calculate the RFEMNN performance in a hierarchical 10-fold validation of the experimental data. The average test results show that the comprehensive diagnostics accuracy (on fault identification and error location aspects) is 94.7%, which is better than the benchmark models in the literature.

Keywords: Rotor finite element model, Physics mimetic layer, Physics mimetic connection, Multi-label and objective learning, Fault recognition, Fault location.

1. Introduction

Rotating equipment is the foundation of energy production and transmission. Typical rotating machinery operating conditions, such as wind and gas turbines, are challenging and harsh. Their components are sophisticated and

Email address: weikun.deng@enit.fr (Weikun Deng)

highly coupled. Therefore, an unexpected component defect can quickly lead to secondary failures and affect others. Due to manufacturing and assembly errors, rotor unbalances are almost inevitable and frequently occur in practice [1]. Besides, under the excitation of unbalance, the accumulation of strain energy from other potential secondary failures, such as shaft cracks, will snowball until catastrophic consequences. These potential failures can expose the structure to unpredictable fatigue failure and permanent structural deformation. Much research has been conducted on this topic with Prognostics and Health Management (PHM) technologies, trying to provide new fault diagnostics methods supported by innovations from physics-based or data-driven approaches. Tthe existing works based on Machine learning (ML) provide promising solutions for fault detection and diagnostics (FDD) of rotor unbalance and shaft cracks. For example, a long short-term memory (LSTM) network is used to detect and classify the rotor unbalance fault and other structural and electrical failures [2]. A support vector machine (SVM) is combined with the advanced signal processing methods in [3] for rotor-shaft surface crack classification. Besides, K-nearest neighbors, naive Bayes, artificial neural networks, and deep learning are all standard data-driven methods for rotor fault diagnostics [4]. However, the performance of these methods strictly depends on the quantity and quality of condition monitoring data. In addition, the number of faulty signals is generally limited. As an alternative, physics-based methods (PBM) and even their digital twin models have been primarily developed to detect and localize unbalance and crack defects [5, 6]. PBM provides credible fault detection and diagnostics by capturing and analyzing parameter or system behavior changes [7], which include finite element models (FEM) and other mechanism modeling methods allowing simulation of normal and fault state responses. However, in practice, the rotor aging and the rotor operating conditions' diversity may affect the performance of the PBM. There is an increasing discrepancy between the actual rotor behavior and its physics model. Also, understanding the fault mechanism may be limited and insufficient to construct trustworthy degradation models [8]. Moreover, model updates can not inherently correct modeling errors [9].

Despite a large number of in-depth failure mechanism research for PBM models, their idealized assumptions are difficult to achieve. Besides, pure ML models can only find correlations or nonlinear mappings of inputs, but not physical causal relationships (hence they are called "black boxes"), making their decisions hard to understand. Therefore, Physics-informed machine learning (PIML) is an exciting way to achieve a compromise between physical and ML models. For leveraging the advantages of different approaches while minimizing their limitations for better results in PIML for rotor fault diagnostics, we need to overcome the following issues:

1. how to build PIML under biased conditions caused by sparse and noisy data?
2. facing combined failures of rotor unbalance and shaft cracks, how to effectively use rotor responses to detect and localize the defects?
3. facing the diversity of rotating system structures, how to build an effective PIML model that can be applied to different rotor layouts?

The above three issues are intertwined and interrelated. It is essential to interweave data-driven and PBM at the most fundamental level, seeking commonalities between PBM and ML and exploiting them to build specific PIML methods [10]. Hence, this paper aims to develop a new PIML method that brings the following contributions:

1. Solutions for mimicking physics in architectural PIML structures.
2. The first work (to our knowledge) on a single PIML model with a composite function dealing with the early detection, identification, and localization of rotor defects.
3. An Rotor Finite Element Simulation Neural Network (RFEMNN), can be applied to multiple rotor structures to diagnose unbalance and shaft crack defects.

This paper is structured as follows. In Section 2, relevant works on developing PIML in industrial systems are discussed and synthesized. Next, we present the novel framework RFEMNN with ML data processing pipelines in Section 3. Section 4 compares the performance of RFEMNN and typical ML models from the existing literature through an experimental case study. Finally, conclusions, open challenges, and future research work are presented in Section 5.

2. Related works

The term "PIML" was formally introduced at the Los Alamos PIML workshops during 2016-2020 [11]. In the field of PHM, PIML specifically refers to the organic combination of ML and prior knowledge of the physics of failures related to the system dynamics behaviors. It has become an effective way to alleviate the shortage of training data, improve model generalization and ensure the physical consistency of the results in different "Informed" ways. According to the different "Informed" paradigms, "PIML" can be divided into three categories:

1. Physics embedded in the construction of health indicators for establishing the ML input (named *PI_input*),
2. Physics embedded in the algorithmic structure of ML (named *PI_structure*),
3. Physics (failure laws and system dynamics behavior) constrained ML objective function design (named *PI_loss*).

Considering the existing "Informed" frameworks, and to the best of our knowledge, currently, available literature on the topic does not address all three issues mentioned in Section 1. However, some inspiring works on PIML-based fault diagnostics and PIML-based surrogate models are reviewed in the following paragraphs and synthetically presented in Table.1.

Table 1: Summary of PIML in fault diagnosis
(FD: fault detection, FDT: fault detection and trace, FDR: fault detection and recognition, CM: performance condition monitoring, DNN: deep neural networks, VAE: variational auto-encoders, CNN: convolution neural networks, DCAE: deep convolutional autoencoders.K-SVD: K-means singular value decomposition)

Ref.	Application	Taxonomy	Informed way	ML model
[12]	Gearbox FDT	PI_input	Vibration inherent cyclostationary characteristics	Autoencoder
[13]	Electromechanical load CM	PI_input	Feature space load separability prior evaluating	SVM & DNN
[14]	Gearbox FDR	PI_input	Health-adaptive physics time-scale representation embeded input module	CNN
[15]	Industrial production FD	PI_input	Time-series derivative weighting for perturbation values	VAE
[16]	Aircraft composite structure FDR	PI_input	A numerical solutions of Lamb waves	CNN
[17, 18]	Crack FDR	PI_structure	Defect representation based on NN connection with focused guided wave	Siamse NN
[19, 20, 21]	Crack FDR	PI_structure	DNN approximates or encodes the measurements as a solution to the model	DeepONet or codec NN
[22]	Bearing FDR	PI_structure	Interpretable weights based envelope spectrum to highlight fault frequencies	Supervised learning dichotomy
[23]	Turbofan FDR	PI_structure	Gas path performance responsed based model calibration	DNN and VAE
[24]	Off-shore and aerostructures FDR	PI_structure	Physics-informed prior mean function	Gaussian process regression
[25]	Aircraft wings fatigue FDT	PI_structure	Walker crack propagation model embeded recurrent NN cell	RNN
[26, 27, 28, 29, 11]	Grid and Buses FD	PI_structure	Physics spatial or spectrum associativity-based graph convolution	Graph NN
[30]	Bearing FDR	PI_structure	Physics convolutional kernel for fault feature similarity comparison	CNN
[31]	Drill pipe FD	PI_structure	Embedding hydraulic coefficient relationship between two DNNs	DNN

[32, 33]	Bearing FDR	PI_structure	Dynamic wavelet informed layer	CNN & Resnet
[34, 35, 36]	Structure damage FDR	PI_structure	Damage-induced structural changes are estimated by output changes of NN	Neural differential equations
[37]	Bridge damage identification	PI_loss	Difference between the update of FEM and NN	NN_FEM parallel
[38]	Bearing & Milling data FDR	PI_loss	Dominant frequency reconfiguration loss	DCAE
[39]	Bearings FDR	PI_loss	Physical threshold model for bearing health	CNN
[40]	High impedance FD	PI_loss	Voltage-current elliptical trajectory reconstruction error	CAE
[19, 41, 42]	Metal plate damage	PI_loss	Regular loss term generated by the residuals of the governing equations	K-SVD & Dictionary & NN
[43]	Building damage FDR	PI_loss	Data transfer based on physical similarity	Adversarial NN

Table.1 is organized according to the three basic paradigms: "PI_input", "PI_structure", and "PI_loss". Those studies have seen a proliferation of applications from material and component damages to system faults. However, none of them deal with the problem of combining fault detection, identification, and localization. Regarding the first group, the existing "PI_Input" approaches can be seen as an extension of the traditional "feature engineering" or "simulation-based data augmentation" processes using physics knowledge to assist data processing. They provide an input space defined by physics knowledge related to faults representation, which is a soft constraint rather than guiding the algorithm to dig deeper into the potential knowledge of the raw data [44]. Their principle remains mapping features extracted from state monitoring data to degradation phenomena. However, after the physics meaningful health indicators have been processed by ML, further changes in these indicators and the action mechanisms are difficult to interpret when applied to the dynamic behavioral representation of PHM case studies [45]. Thus, the first contribution of our work is further exploring how to enhance the physics consistency and accomplish a higher level of "physics-informed" in the construction of physics-informed input space.

Considering the "PI_structure" category, some studies extended the idea of "PI_input" to construct additional layers that allow embedding physics knowledge into ML structures for extracting valuable features [32, 33, 30]. Beyond feature extraction, PIML can estimate missing physics information [25] and unobserved process variables [23]. More specifically, some "PI_structure" approaches propose integrating "Hard Constraint Projections (HCP) [46]" with ML. Accordingly, NN has an HCP layer that employs an activation function or cell structure based on domain knowledge to ensure that the network's predictions conform to the equations-based physics constraints. In addition to basic

ML module design, it is possible to embed physics knowledge by designing links between ML modules. For example, in the turbofan degradation prediction [47], three cascaded NNs can represent the PDE-based system dynamics. In general, the serial structure of NN is suitable when the failure model is uncertain because it facilitates the adaptability of each sub-module to extract useful information [48]. In contrast, the parallel structure of NN accomplishes hybrid estimation with high accuracy but needs better extrapolation capabilities. Therefore, the originality of the approach proposed in this paper lies in the fact that RFEMNN topologies the rotor finite element structure through customized layers and the finite element-based fault diagnostic process through customized inter-layer connections.

Regarding the third group, "PI_loss", the relevant studies add a physics non-consistency item in the loss calculation when training ML models. This item can be considered as a test of the model output concerning the laws of physics (such as the rules of thumb for cutter head wear [49] or the reconstruction error of physical components like fault frequency [38]). Besides, "PI_loss" can be derived by the difference between PBMs output and ML output [39]. To design "PI_loss", it is necessary to answer the questions of "which" labels to introduce in the implementation and "how" to calculate the losses corresponding to these labels. Considering the first term "which", most existing studies focus on fault classification issues but do not investigate fault localization problems. In order to localize the damage, the labels in the dataset can be extended with the corresponding damage locations [50, 51]. Furthermore, the model can perform multi-task learning for fault type classification and fault localization regression purposes. Multi-task learning (MTL) is not new for ML. However, for PIML, an interesting question is how to design representative labels that connect and interact with each other, enabling both regression and classification tasks based on physical knowledge. Addressing this question brings another originality to this paper. In addition, the second term "How", related to calculating the losses corresponding to these labels, will be investigated in the framework proposed in this paper in the MTL process.

3. Proposed PIML approach

To address the challenges in section 1 and show the innovation points which have been discussed in section 2, we propose a rotor dynamics finite element model informed deep learning framework for detection, identification, and localization of shaft crack and unbalance defects. It is called Rotor finite element mimetic neural network (RFEMNN). An overview of the implementation of the proposed RFEMNN model is shown in Fig. 1.

In the RFEMNN model, the input includes the raw vibration acceleration signal and physics parameters (i.e., shaft geometry and components material parameters). These parameters are used in step ③ to build the FEM mimetic NN structure. The proposed RFEMNN architecture is intended to facilitate the optimal calibration process of the RFEMNN model thanks to the ML's nonlinear fitting ability in the supervised multi-task learning-training process. The framework diagnostics results are fault type, fault location, and predicted temporal vibration features.

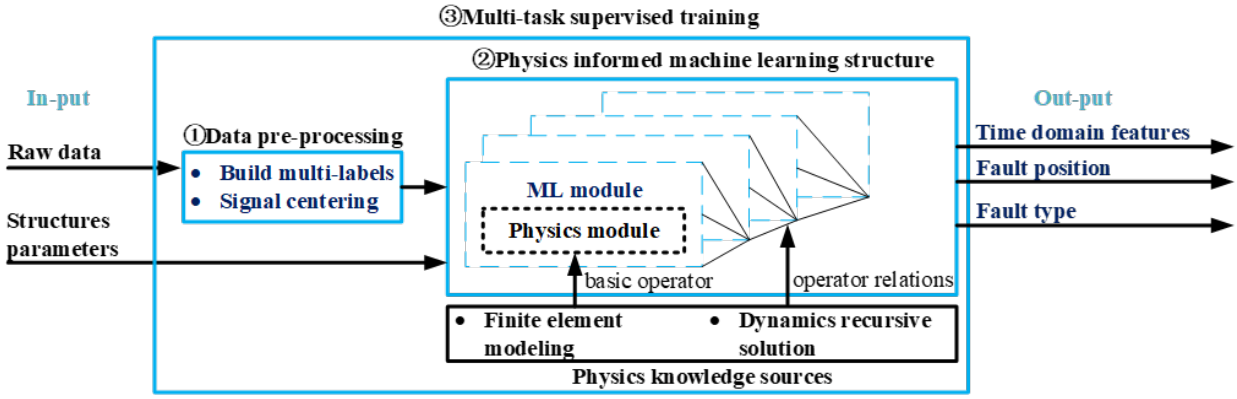

Figure 1: Overview of the proposed RFEMNN architecture

The RFEMNN model of Fig. 1 includes three steps: 1) data pre-processing (presented in Section 3.1), 2) construction of PIML model (presented in Section 3.2), and 3) multi-task supervised training (presented in Section 3.3). Every step in the RFEMNN model aims to effectively exploit known physical knowledge related to the rotor dynamics problem. Generally, the rotor dynamics are governed by Eq. (1):

$$M \underset{known}{\ddot{q}} + Sq + D\dot{q} = F(x,t), \ddot{q} \tag{1}$$

Where M, S, D, and $F(\cdot)$ are respectively the mass, the stiffness, the damping and the excitation force of the rotor. Besides, q is the vibration displacement, while $\dot{q}$ and $\ddot{q}$ are the velocity and the acceleration of the rotor. Among those parameters, only the $\ddot{q}$ is monitored in this work by vibration signals collected using accelerometers, thus it is denoted as $\underset{known}{\ddot{q}}$. This work assumes that Eq. (1) is obtained based on a finite element discretization of the problem. The proposed RFEMNN aims to mimic the physical relations between $[M, S, D, F]$ and q by customizing layer and layer connections to incorporate the mass finite element matrices M, the stiffness finite element matrices S. The damping finite element matrices D into the structure of the CNN-LSTM network. Then, RFEMNN is trained to determine the fault type and the defect position in the MTL process.

3.1. Data pre-processing

The monitoring vibration signals are split into small segments before input to the RFEMNN. The length of those segments is an important parameter that can affect the model performance. A too-short segment needs to contain more information, while a too-long segment can lead to excessive computational resource requirements. Therefore, Algorithm 1 is proposed to split data in different segments with an appropriate length by considering the relations among the NN input unit number, the rotor shaft element length, and the sampling frequency. Given $scale$ is an integer with a base number of 2, the whole data splitting algorithm is follows:

Algorithm 1 Determining the sliding window step in sample splitting.

Input: Frequency sampling (fs), lengths of different considered rotor shafts ($Shaft\_length\_groups$), and $sample\_length$. T_fs is theoretical sampling frequency. N is the number of divided axis segments.

Output: Sliding window step, N

Function Data slice
    T_fs=fs× scale
    Min_unit length =$\lceil 1/T\_fs \rceil$
    Sliding window step= $(fs - sample\_length)/(T\_fs/sample\_length - 1)$
    N=$Shaft\_length\_groups/Min\_unitlength$
    return Sliding_window_step, N

The fault type, location, and rotor behavior features are defined as labels for the multi-task learning (MTL) process.

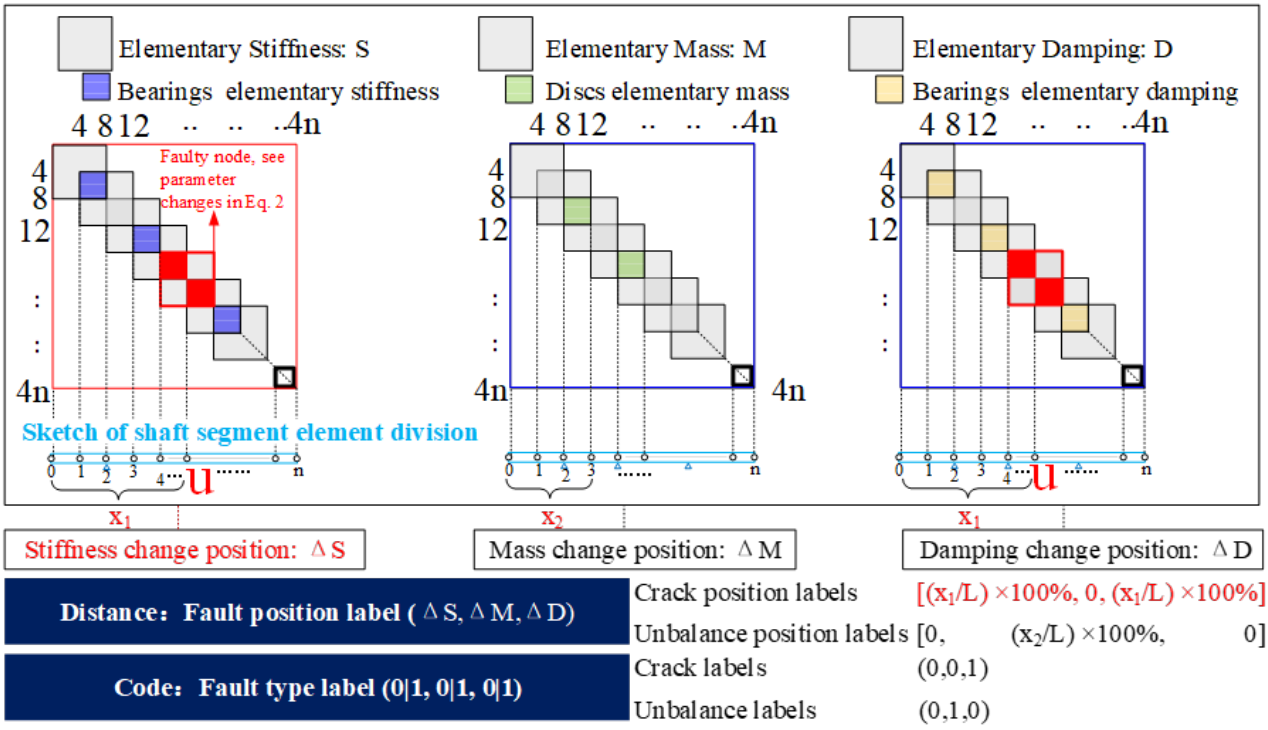

Figure 2: Relation between the matrices of the structural parameter, fault type, and fault location labels

Firstly, the healthy state(1, 0, 0), unbalance defect(0, 1, 0), shaft crack (0, 0, 1), and the combined defects (of rotor unbalance and shaft cracks)(0, 1, 1) are respectively represented by a binary one-hot encoding.

Secondly, the defect distance gives the location label (such as x_1, x_2 in Fig. 2, their units are in meters). Because in the experimental tests, several rotor structure layouts with different shaft lengths are considered, it is necessary to normalize the defect distance by the corresponding whole shaft length L. RFEMNN is to design the NN structure to mimic the relationship between the structural model and the fault location, as shown in Fig. 2. For example, the

shaft crack mainly affects the change in the bending stiffness and structural damping of the rotating shaft, which is represented by the nodal value changes in some specific small matrices in S like a constant symmetric matrix s_u in Eq. (2). When a failure occurs, s_u becomes a variable related to rotating speed rpm and time t. Different faults affect different types of structural matrices. Then at the RFEMNN output, the fault location is the regression of the structural parameters variation in the M, S, and D mimetic NN layers.

$$
\begin{aligned}
&\overset{Healthy}{} \\
S_u = EI/(L^3) &\begin{bmatrix}
\ldots & & & & & & & & \\
& 12 & & & & & & & \\
& 0 & 12 & & & & & & \\
& 0 & -6L_n & 4L_n^2 & & & & & \\
& 6L_n & 0 & 0 & 4L_n^2 & & & & \\
& -12 & 0 & 0 & -6L_n^2 & 12 & & & \\
& 0 & -12 & 6L_n & 0 & 0 & 12 & & \\
& 0 & -33L_n & -L_n^2 & 0 & 0 & 6L_n & 4L_n^2 & \\
& 6L_n & 0 & 0 & 2L_n^2 & -6L_n & 0 & 0 & 4L_n^2 \\
& & & & & & & & \ldots
\end{bmatrix} \\
&\Downarrow \\
&\overset{Faulty}{} \\
S_u = f(EI/L^3 &\begin{bmatrix}
\ldots & & & & & & & & \\
& 12 & & & & & & & \\
& 0 & 12 & & & & & & \\
& 0 & -6L_n & 4L_n^2 & & & & & \\
& 6L_n & 0 & 0 & 4L_n^2 & & & & \\
& -12 & 0 & 0 & -6L_n^2 & 12 & & & \\
& 0 & -12 & 6L_n & 0 & 0 & 12 & & \\
& 0 & -33L_n & -L_n^2 & 0 & 0 & 6L_n & 4L_n^2 & \\
& 6L_n & 0 & 0 & 2L_n^2 & -6L_n & 0 & 0 & 4L_n^2 \\
& & & & & & & & \ldots
\end{bmatrix}, rpm, t)
\end{aligned}
\tag{2}
$$

Finally, the rotor vibration behavior label is used for guiding the training to keep the response behavior consistent. This label is the temporal statistic features considered in this paper: the margin factor, impulse factor, peak factor, wave factor, kurtosis, skewness, RMSE, variance, and mean. Their details can be consulted in the reference [52].

3.2. Building physics informed machine learning structure

This subsection describes how to incorporate FE model into CNN-LSTM seamlessly. Remarkably, the proposed RFEMNN builds a customized layer to construct the basic operators similar to the elements of the FEM (see Section

3.2.2 for more details). In addition, the design of RFEMNN layer connections allows driving the data flow similar to the derivation process of the rotor FEM solution (see Section 3.2.3 for more details).

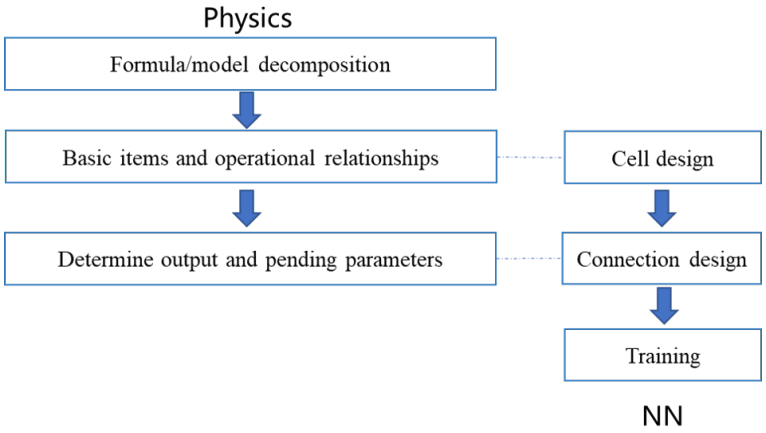

Figure 3: Step ②-Specific steps for embedding physics knowledge into neural networks

Fig. 3 shows the specific steps for embedding physics knowledge into neural networks structure. It includes:

- the topologically similar structure that can be obtained by customizing the ML module and by replacing the original ML input-output module partially or completely with the physics model,
- the topologically similar physics derivation process can be obtained by customizing the ML module connection and by changing the original ML interlayer connections partially or wholly with the physics symbolic derivation process.

3.2.1. Overview of the proposed RFEMNN model

Fig. 4 presents an overview of the proposed RFEMNN model. Condition monitoring data, i.e., vibration signals, are fed at the input of the NN model while the rotor information, such as the rotating speed and the whole rotor shaft length, are used as additional parameters to construct the FEM cell of the NN model. The RFEMNN outputs provide diagnostics information such as fault types and position as well as the rotor vibration features.

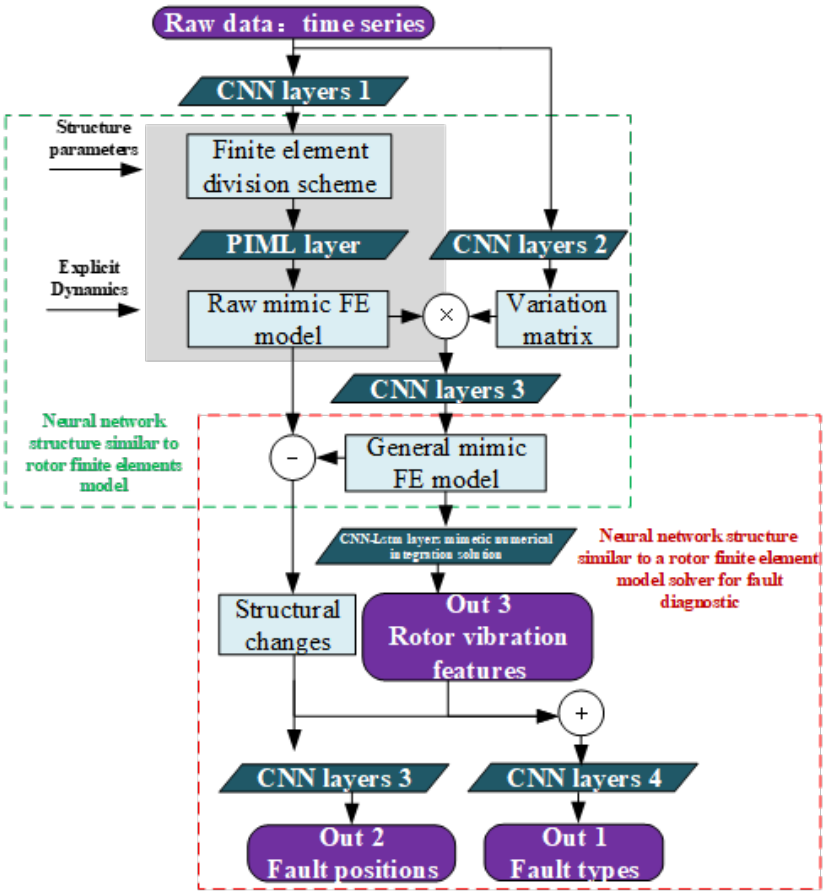

Figure 4: Schematic diagram of the proposed RFEMNN model

This RFEMNN model consists of two principal parts: similar structure topology (framed by the green dash line) and similar derivation topology (crafted by the red dash line). The first part aims to capture the essential physics operators and their mimetic relations. In contrast, the second part seeks to build the FEM solution process-based relationship between the mimic NN module and the final outputs. The finite element division ratio of the shaft elements is provided by "CNN layers 1". Meanwhile, three RFEMNN outputs (i.e., fault positions, fault types, and vibration features) are the results of three data flows coupled in the proposed model.

The output "Out 3", which represents rotor vibration features, is defined by a codec-like structure. The difference between the predicted features obtained at the "Out 3" and the real features extracted from the raw vibration signals is used as the constraints of the dynamic system behaviors. "Out 3" also contributes to the supervised learning process of the CNN layers 2, 3, and the CNN-LSTM layer. In detail, "CNN layers 2" uses vibration signals as input to predict the variation of the parameters in the "Raw mimic FE model part". It enables the "Raw mimic FE model" to be further generalized to a more expressive "General mimic FE model". "General mimic FE model" represents a pool of potential structural possibilities of the different layouts. Then, its output will be used as the input of the CNN-

LSTM layer to extract the time features of vibration signals. “General mimic FE model”, “Raw mimic FE model”, and “Structure changes” are the foundation of the RFEMNN, and they share parameters and information so that the different tasks are coupled with each other.

“Out 2”, i.e., fault position, is obtained through a regression process. In fitting the system’s dynamic behavior, the “Variation matrices” will iterate their own parameters to minimize the difference between the “General mimic FE model” and the real system behavior function. We assume that the “Raw mimic FE model” is the rotor finite element matrices with no fault conditions and it is created by the rotor FEM. The difference between the raw and the general mimic FEM represents the degradation of the system structure from no-fault to fault conditions. The location of the defect is then determined based on variations regression of the structural parameter of different mimetic shaft nodes in the “Structure changes”. This physics knowledge is embedded into the hidden layer by calculating the “Structure changes”. The “Structure changes” output is then fed into the “CNN layers 3” to get the position-related coding features, i.e., “Out 2".

“Out 1", i.e., fault types, is provided at the output of “CNN layers 4" which uses the “Structure changes” output and the vibration time features, “Out 3", as the input. “Structure changes” denotes changes in the mimetic neural matrices M, K, D, gives additional physics information to enhance the precision of the conventional diagnostics results which could be obtained only based on a pure data-driven process (i.e., time features extracted from vibration signals).

3.2.2. Customizing layers to mimic the finite element model

The specific steps for customizing the design layer “PIML layers’ module” of the mimetic FEM are as follows:

1. Unit generation: Using the output of “CNN layers 1" as the input for “PIML layer", the bending & shear stiffness matrices and the inertia mimetic NN tensor are generated in the discrete variational form of FE.

2. Unit assembly: Assembly of the stiffness and mass cell tensor into the overall stiffness tensor by using the corresponding position filling.

The structure of the “PIML layers’ module” is shown in Fig. 5. It consists of the “PIML layer” to topology the basic elements in FEM. The output of the “PIML layer” are processed by a restock to nonlinear fit the “Raw mimic FE model”. The “Raw mimic FE model” consists of the mimic mass, stiffness, and damping layers, as shown in Fig. 5.

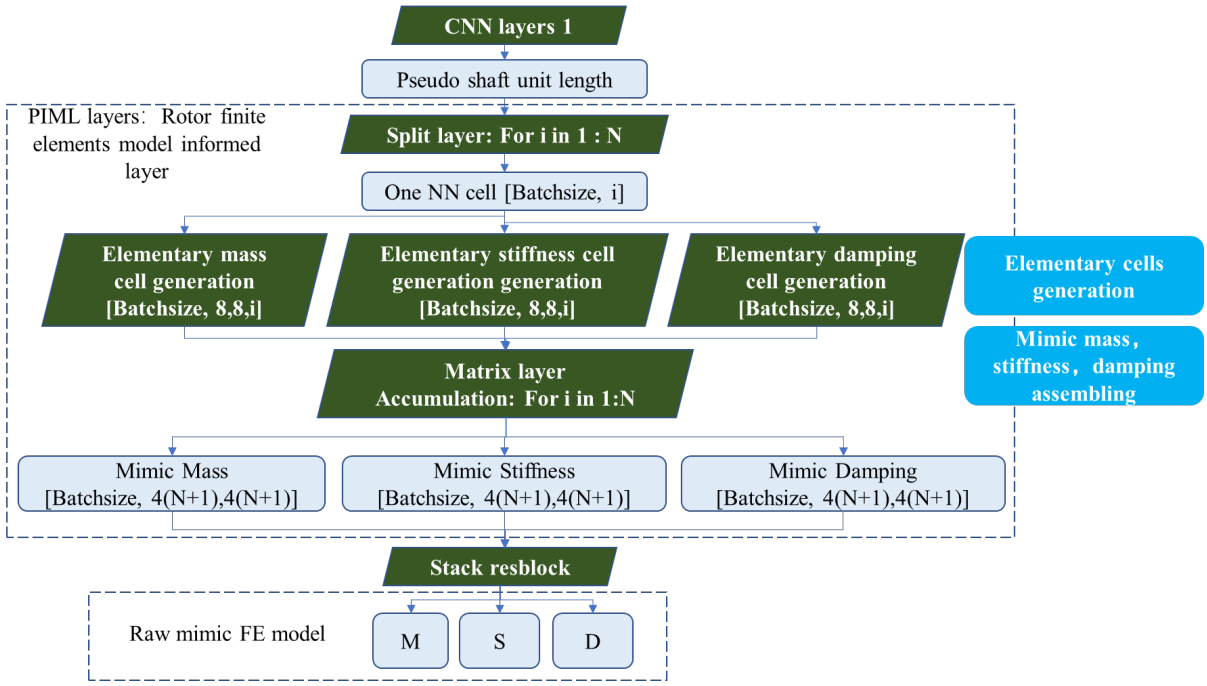

Figure 5: Structure of PIML layers

The raw data are processed in the "CNN Layers 1" to generate N neural network cells. They represent the ratio of the length of each shaft segment to the total shaft length. This ratio is multiplied by the total shaft length and the input of the subsequent "PIML layer", i.e., "Pseudo shaft unit length". Under the Euler-Bernoulli beam assumption, the structural characteristics of one shaft element are represented by the elementary mass M, stiffness S and damping D parameters. Each shaft element involves two nodes (endpoints of the shaft elements). Each node has 2 degrees of freedom, lateral displacement ν and in-plane turning angle φ, each degree of freedom is divided into horizontal and vertical directions by x and y orthogonal decomposition, i.e., ν_x, ν_y, φ_x and φ_y, as shown in Fig. 6. As a result, each shaft element can be represented by 3 matrices with a dimension of 8×8, called "Elementary Mass matrices", "Elementary Stiffness matrices" and "Elementary damping matrices", as shown in Fig. 5.

In "Elements assembly" step, after generating all the elements, the proposed RFEMNN assembles the mimic finite element NN cells into mimic structural topology layers, see Fig. 7, according to the shaft unit order $n = 0, 1, ..., N$. This process obeys the finite element assembly rule based on the fact that the displacement at the right endpoint of a shaft element is the same at the left endpoint of the next shaft element. It outputs a finite element NN layer topology that consists of the total stiffness, the total mass and the total gyroscopic tensor of the entire shaft.

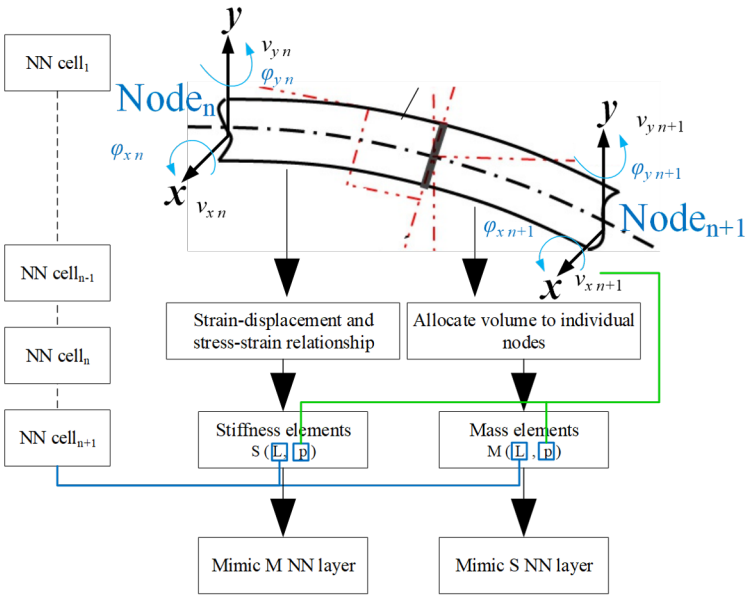

Figure 6: Generation and assembly of NN cells for simulating finite elements.

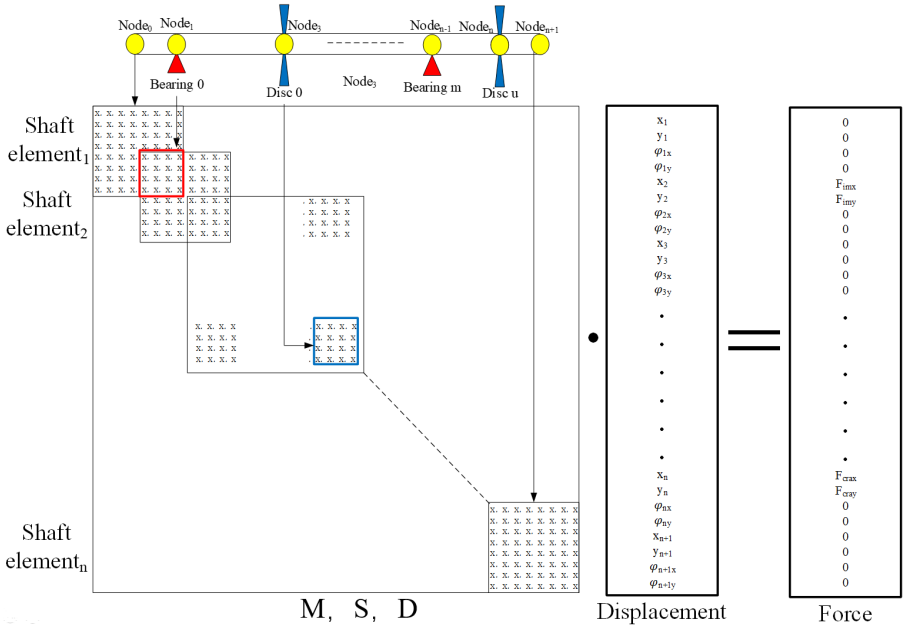

Figure 7: Assembling finite element NN layer.

The discs' mass is added directly to the "Mimic Mass layer" in the disc nodes, and the stiffness of the bearings is added to the "Mimic Stiffness layer" in the bearing nodes. Since exact values of the disc block mass and bearing stiffness are not available, these values are initialized by the default values of the simulation and then adjusted by training.

3.2.3. Customizing layer connections to mimic FEM solution process

The motions of the rotor in the faulty cases can be expressed by Eq. (3), where $q(t)_{fault}$ denotes the vibration displacement. Note that in this equation, only $\ddot{q}(t)$ is monitored by vibration signals while the fault excitation Δf and the operational excitation $F(x,t)$ remain unknown. However, they can be expressed by $\Delta m'$, $\Delta s'$ and $\Delta d'$, which is equivalent to the deterioration change in the structural parameters as shown in [53].

$$\begin{cases} (M+\Delta \mathrm{m})\,\ddot{q}(t)_{fault} + (\mathrm{S}+\Delta\,\mathrm{s})q(t)_{fault} + (\mathrm{D}+\Delta \mathrm{d})\dot{q}(t)_{fault} = \mathrm{F(x,\ t)}+\Delta \mathrm{f} \\ \quad (M+\Delta m')\,\ddot{q}(t)_{fault} + (\mathrm{S}+\Delta \mathrm{s'})q(t)_{fault} + (\mathrm{D}+\Delta \mathrm{d'})\dot{q}(t)_{fault} = 0 \\ \Rightarrow [\mathrm{failure}_{type},\ failure_{position}] = \mathrm{h(G(M,\ D,\ S),\ G'(\Delta m',\ \Delta D',\ \Delta S'),\ \Delta f)} \end{cases} \tag{3}$$

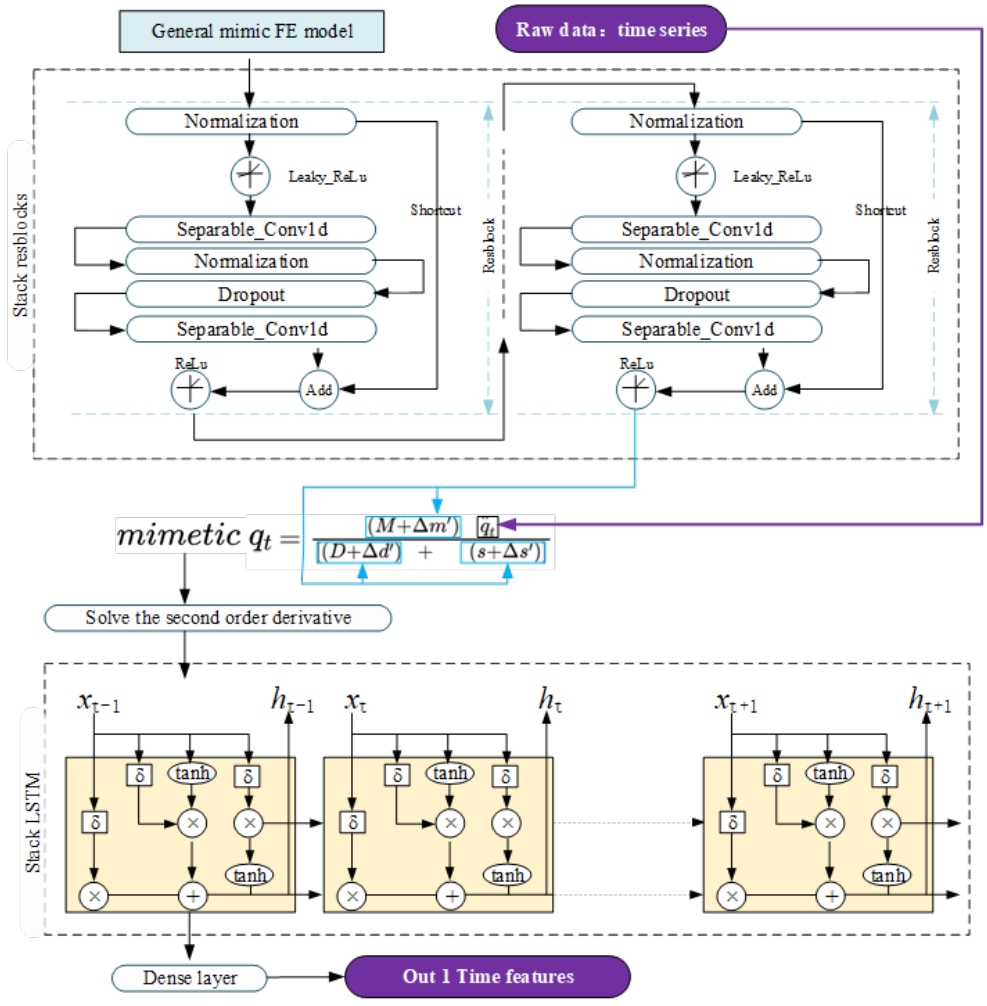

Figure 8: Structure of CNN-LSTM layer to mimic the FEM solution process

It is not trivial to analytically address a partial differential equation as the one in Eq. (3), by taking into account

the combination of structural deterioration and fault excitation. But this equation can be approximately solved by establishing the stacking of multiple hidden layers in the RFEMNN, i.e., by using the NN layers module to capture the implicit function $h(\cdot)$ presented in Eq. (3). In detail, the function $h(\cdot)$ can be built to mimic the Newmark-β method [53] that transfers the equations Eq. (1) into implicit recurrence relations, as shown in Fig. 8.

After processing by stacking residual blocks, the mimetic M, S, D tensor in the "General mimic FE model" can be used to represent $M + \Delta m', S + \Delta s', D + \Delta d'$ in Eq. (3), and to calculate $mimetic\ q_t$ as shown in Fig. 8. Then, the second-order derivatives $\ddot{q}_t$ and the recovered time features of the vibration signals can be approximately obtained by training the LSTM and dense layers.

3.2.4. Selection of physics parameters

Table.2 summarizes the physical parameters involved in the RFEMNN model and presents their determination method.

Table 2: Physics parameters in RFEMNN.

Symbol	**Parameter meaning**	**Determination method**
ρ	Structure density	Design parameter
R_d	Shaft section outer diameter	Field measurement or Design parameter
r_d	Shaft section inner diameter	Field measurement or Design parameter
th_i	Thickness of the disc	Field measurement or Design parameter
s_{ji}	Support stiffness	Estimation or Design parameter
d_{ji}	Support damping	Estimation or Design parameter

The support stiffness and damping come from the coupling of the bearing and the support structure. In practice, it is difficult to obtain their exact values but can be adjusted during the training by the "variation matrices" presented in Fig. 4.

3.3. Multi-task supervised training

The proposed model will be trained in a multi-task learning (MTL) process and constrained by a multi-loss function shown in Eq. (4). In this equation, L_t denotes the loss of the fault type recognition while L_m and L_s are the location error of the unbalance fault and shaft crack fault. L_v represents the loss concerning the reconstruction of temporal vibration features. Note that the binary cross entropy is used to metric L_t while the mean square error is used to metric L_m, L_s, and L_v. α_t, α_m, α_s, and α_v are respectively the weights of the corresponding loss.

$$L = \alpha_t L_t + \alpha_m L_m + \alpha_s L_s + \alpha_v L_v \tag{4}$$

The major challenge of this multi-task supervised training comes from the joint optimization process for all tasks (identification of fault types, fault position, and temporal vibration features). Particularly, the definition of the overall

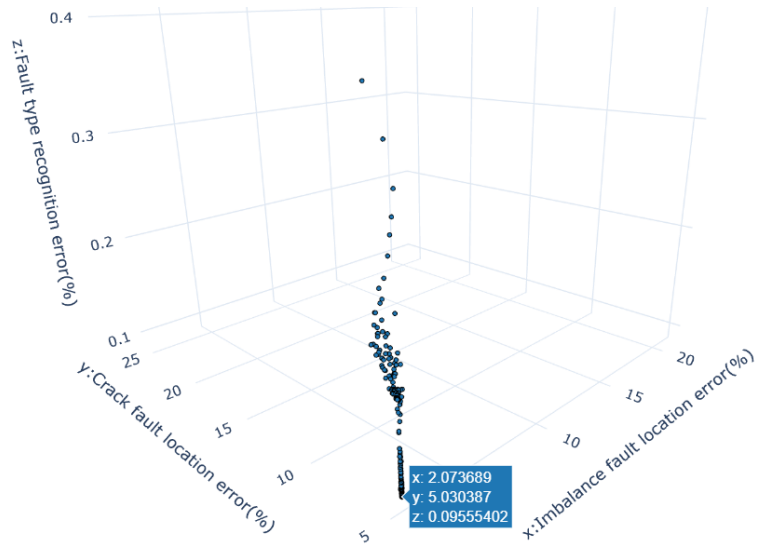

Figure 9: Selection of the optimized models in 3 aspects: fault identification, unbalance localization, and crack localization.

loss needs to be carefully balanced to avoid situations where one or more tasks have a dominant influence on the network weights iteration. To do this, it is necessary to address the following two issues.

1. Different scales of output magnitudes.
   Considering different orders of magnitude in the RFEMNN outputs, the loss weights α_t, α_m, and α_s are set to 100 to balance the influence of varying output magnitudes. They correspond to the fault type and the fault position. In contrast, the loss weight α_v corresponding to the vibration features is set to 1.

2. Conflicts when optimizing the overall and local loss functions.
   For diagnostics, L_m and L_s correspond to the accuracy of fault localization. L_t represents the fault identification accuracy. Then, L_m, L_t, and L_s are the main factors to decide the diagnostics performance of the trained model. However, optimizing the total loss function L presented by Eq. (4) does not guarantee that its local loss functions (L_m, L_s) are simultaneously optimized to the best. So we use L to drive the training process, save the training results after every three epochs in a model pool, and monitor L_v to decide the stopping time (e.g., the training process will be stopped if L_v is not improved after 50 epochs). After the training process, the model pool is thoughtfully investigated to select the best model that has the best performance in all three aspects: L_m, L_s, and L_t, as shown in Fig. 9.

4. Case study

In this section, the performance of the proposed model is investigated on a real test bench for diagnostics of unbalanced faults and shaft cracks. In addition to considering single and combined defects, we validated RFEMNN under different operating conditions and with different rotor configurations.

4.1. Description of the case study

4.1.1. Experiment platform

Fig. 10 shows the platform used to test the performance of the proposed RFEMNN model. This PT 500 machinery diagnostics platform [54] is driven by a three-phase AC motor with a rated speed of 3000 rpm. The setup accessories are two rigid long shafts and one rigid short shaft, three supports consisting of two roller bearings and one ball bearing, two discs with counterweight holes and a flange coupling. These accessories allow constructing different rotor structure layouts by modifying the position of the supports and discs. Two accelerometers can be placed in different directions on different supports to collect vibration signals through a data acquisition card NI-DAQ 9174 with a sampling frequency of 4096 Hz.

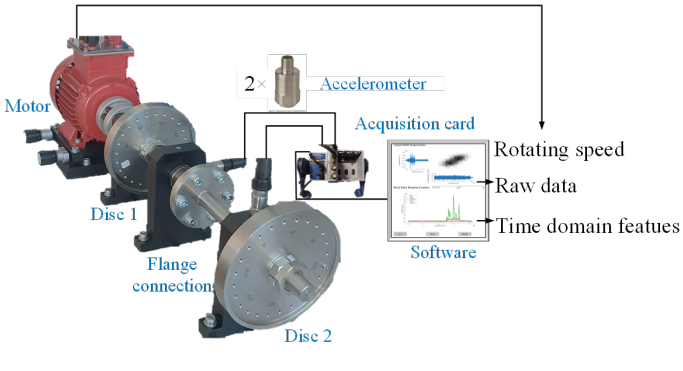

Figure 10: Schematic diagram of the experimental platform.

4.1.2. Multi-faults experiments

In this work, the unbalance fault is artificially created by adding a 2 g screw in a counterweight hole on the disc, as shown in Fig. 11. Different unbalance positions are investigated by changing the positions of the screw on the disc. The number of screws can simulate unbalance degree.

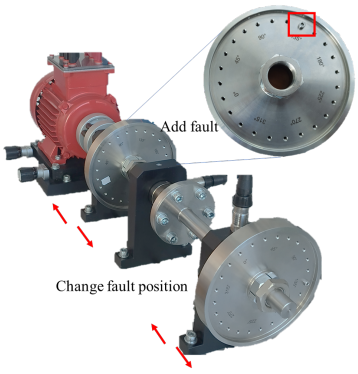

Figure 11: Schematic diagram of the unbalance fault experiment.

In the shaft-cracking experiments, the flanged shaft consists of a long and a short one. Different crack positions are simulated by connecting the long or short shaft to the rotor shaft (as in Fig. 12) .

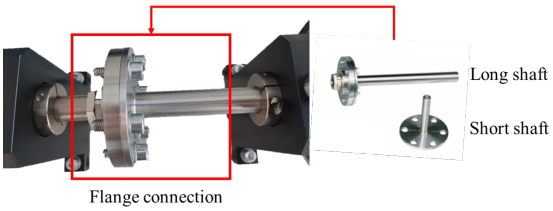

Figure 12: Schematic diagram of the shaft-cracking experiment.

Table.3 summarizes different experiments investigated in this work. We consider four health states of the rotor: healthy (H), unbalance (U), crack (C), and a combined fault (U&C). The positions of unbalanced defects and shaft cracks in different experiments are varied along the shaft length. Also, to highlight the robustness of the proposed model, different rotor structures (with 3 shaft lengths and 4 layouts, see Fig.13) are tested by varying the rotating speed.

Table 3: Overview of the experimental setup.

Health state	**Shaft length** (m)	**Rotor structure**	**Rotating speed** (rpm)	**Fault position** (m)	**Samples**
H_1	0.355	Layout A	1200, 1500, 1800, 2100, 2400, 2700, 3000	-	300
U_1	0.355	Layout A	1200, 1500, 1800, 2100, 2400, 2700, 3000	0.175	300
H_2	0.409	Layout B	1500, 1800	-	40
C_1	0.409	Layout B	1500, 1800	0.120	40
U_2	0.409	Layout B	1500, 1800, 2400	0.290	60
H_3	0.605	Layout C	1200, 1500, 1800	-	60
C_2	0.605	Layout D	1500, 1800, 2400	0.355	60
U_3	0.605	Layout D	1500, 1800, 2400	0.207	60
U_4	0.605	Layout D	1500, 1800, 2400	0.110	60
U_5	0.605	Layout D	1500, 1800, 2400	0.155	60
U&C_1	0.605	Layout C	1200, 1500, 1800, 2100, 2400	C:0.155, U:0.586	60
U&C_2	0.605	Layout C	1200, 1500, 1800, 2100, 2400	C:0.355, U:0.175	60

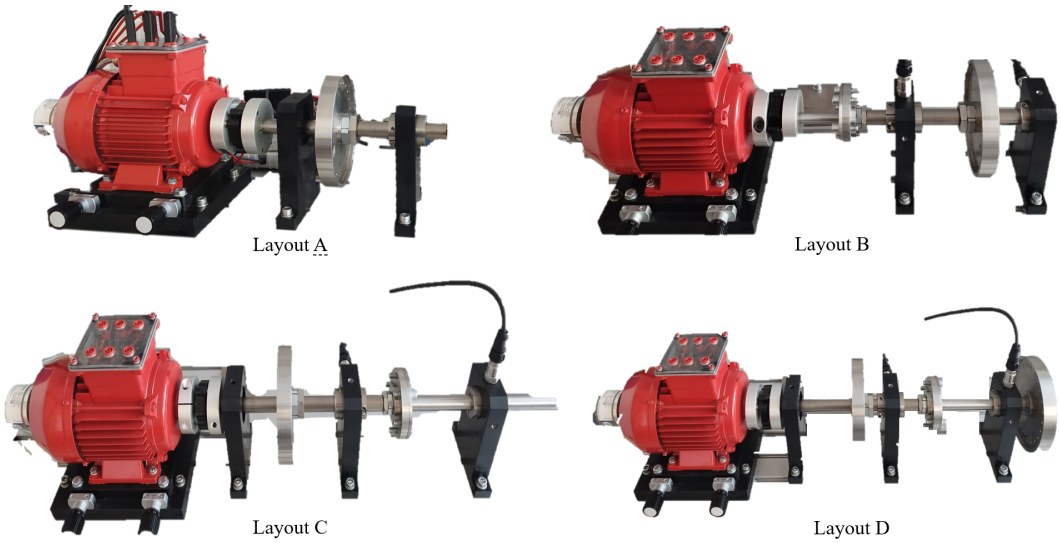

Figure 13: Different structure layouts of the test bench.

4.2. Data slicing and labeling

According to the data slicing and labeling methods presented in subsection 3.1, the monitoring vibration signals are sliced into 39342 samples with a length of 256 points. Note that the labels of fault types, fault location, and temporal vibration features are assigned for each sample, as illustrated in Tab. 4.

Table 4: Illustration of the data labeling.

Health state	**Fault type**	**Fault location(%)**	**Temporal vibration features**
H_1	(1, 0, 0)	(0, 0)	Margin factor
U_1	(0, 1, 0)	(0.175, 0)/0.355×100	Impulse factor
H_2	(1, 0, 0)	(0, 0)	Peak factor
C_1	(0, 0, 1)	(0, 0.120)/0.409×100	Wave factor
U_2	(0, 1, 0)	(0.290, 0)/0. 409×100	Kurtosis
H_3	(1, 0, 0)	(0, 0)	Skewness
C_2	(0, 0, 1)	(0, 0.355)/0.605×100	Rmse
U_3	(0, 1, 0)	(0.207, 0)/0.605×100	Variance
U_4	(0, 1, 0)	(0.110, 0)/0.605×100	Mean
U_5	(0, 1, 0)	(0.155, 0)/0.605×100	
U&C_1	(0, 1, 1)	(0.586, 0.155)/0.605×100	
U&C_2	(0, 1, 1)	(0.175, 0.55)/0.605×100	

4.3. Performance evaluation metrics

The proposed RFEMNN model not only addresses classification issues for identifying fault types but deals also with regression problems for fault localization. Thus, it is necessary to propose the appropriate metrics that allow us to evaluate the overall performance of the proposed model on both classification and regression issues (see Table 5).

Table 5: Evaluation metrics.

A	Accuracy
F	False alarm
M	Missing rate
L_U	Average positioning accuracy for unbalance faults
L_C	Average positioning accuracy for shaft crack faults
$\overline{p}$	Average positioning accuracy for all faults
T	Integrated diagnostic accuracy considering localisation and type identification

The classification metrics, i.e., accuracy (A), false alarm rate (F), and missing rate (M), are calculated by Eq. (5). In this equation, the values a_{ij} and r_i are given by Table 6, where the term a_{ij} represents the number of samples attributed to the corresponding category.

$$\begin{cases} F = \frac{a_{12}+a_{13}+a_{14}}{a_{11}+a_{12}+a_{13}+a_{14}} \\ A_i = \frac{a_{ii}}{\sum_{j=1}^{4} a_{ij}}, j = 1,2,3,4 \\ M = \frac{a_{21}+a_{31}+a_{41}}{\sum_{i=1}^{4} r_i} \end{cases} \tag{5}$$

Table 6: Confusion matrices.

	Diagnostics results				
		Healthy	**Unbalance**	**Crack**	**Un & Cra**
Real status	**Healthy**	a_{11}	a_{12}	a_{13}	a_{14}
	unbalance	a_{21}	a_{22}	a_{23}	a_{24}
	Crack	a_{31}	a_{32}	a_{33}	a_{34}
	Un&Cra	a_{41}	a_{42}	a_{43}	a_{44}
	Sum:	$r_1 = a_{11} + a_{21}+a_{31}+a_{41}$	$r_2 = a_{12} + a_{22}+a_{32}+a_{42}$	$r_3 = a_{13} + a_{23}+a_{33}+a_{43}$	$r_4 = a_{14} + a_{24}+a_{34}+a_{44}$

Regarding the performance of defect localization, the mean absolute error is used to evaluate the location error of the unbalanced defect (L_U) and the shaft cracks (L_C). The average localization accuracy of all defects is denoted by $\overline{p}$. Finally, a metric T defined in Eq. (6), which combines fault localization accuracy and fault type identification accuracy, is proposed to evaluate the total performance of the proposed model. In this equation, $L_{U_{predict}}$ and $L_{C_{predict}}$ are the diagnostics fault location measured as a percentage of the whole shaft length, while $L_{U_{real}}$ and $L_{C_{real}}$ are the real fault locations with the same measurement method.

$$\begin{cases} T = \frac{a_{11}+a_{22}+a_{33}+a_{44}}{\sum_{i=1}^{4} r_i} \times \overline{p} \\ L_U = \frac{L_{U_{predict}} - L_{U_{real}}}{\sum_{i=1}^{4} r_i} \\ L_C = \frac{L_{C_{predict}} - L_{C_{real}}}{\sum_{i=1}^{4} r_i} \\ \overline{p} = \frac{(L_{U_{predict}} - L_{U_{real}}) + (L_{C_{predict}} - L_{C_{real}})}{2\sum_{i=1}^{4} r_i} \end{cases} \tag{6}$$

4.4. Validation results

To verify the effectiveness of the proposed approach, a hierarchical 10-fold cross-validation technique is used to assess the generalization ability of the RFEMNN model on experimental datasets. As shown in Fig. 14, the dataset for each experiment is divided into 10 folds, of which 9 folds are selected in turn as the training set and the remaining folder is the test set. After iterating this process 10 times, the model performance is scored using the average value of the 10 test results.

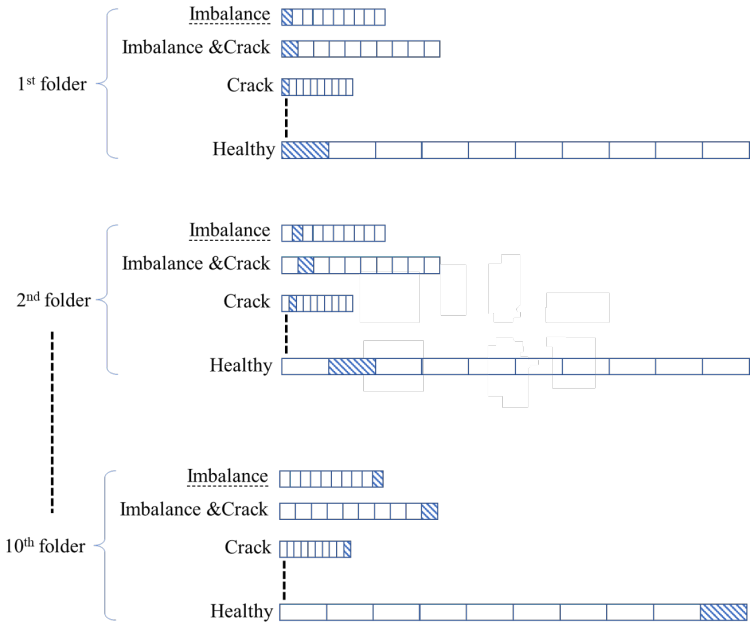

Figure 14: Illustration of 10-fold cross-validation.

Results of the proposed model

Using the proposed RFENN framework, the confusion matrices of fault classification results are presented in Table 15. Besides, the result of the fault localization is shown in Fig. 18.

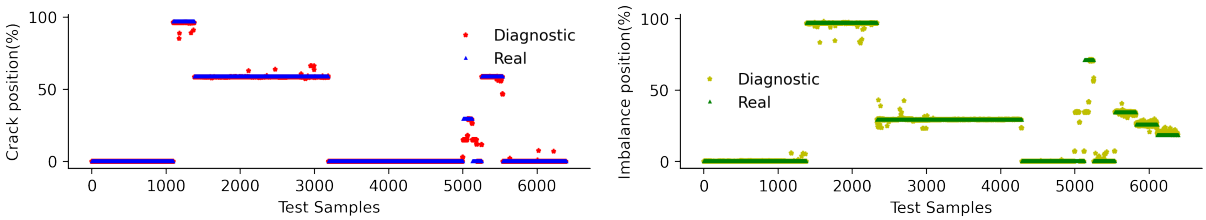

Figure 15: Average results of fault localization with 10 fold-cross validation on diagnostics fault location

Table 7: Confusion matrices of RFENN fault type identification.

		Diagnostics results			
Real status		**Healthy**	**unbalance**	**Crack**	**Im&Cra**
	Healthy	1648	167	0	0
	unbalance	100	2262	0	0
	Crack	0	0	411	0
	Im&Cra	0	0	0	1805
	Total:	1748	2429	411	1805

From Table.15 and Fig. 18, one can see that:

1. Considering different rotor structures with multiple rotor-speed-varying processes, the diagnostics results provided by RFEMNN are close to the ground truth on both aspects: fault identification and error location. Particularly, unbalanced faults, shaft cracks, and combined defects are completely distinguished by the proposed RFEMNN (Table 15).

2. All combined defects are detected and identified correctly.

3. There are still some cases of false and missing alarms. In detail, 167 healthy samples are incorrectly misrepresented as faulty ones while 100 unbalanced faulty samples are misidentified as healthy (Table 15). However, the false (F) and missing (M) rates are small enough. They are 9.2% and 1.56%, respectively.

4. For localization of unbalanced defects and shaft cracks, the results provided by RFEMNN are close to the true ones (Fig. 18). Only a small number of samples have significant prediction errors.

5. As can be seen from Fig. 18, the results of the shaft crack localization are generally better than the ones of unbalanced defects. This can be explained by the fact that the shaft cracks have greater effects on vibration signals than the unbalanced faults, because any rotating system has its initial unbalance.

4.5. Comparison of the performance of the proposed RFENN with some benchmark models

4.5.1. Description of the benchmark models

To highlight the performance of the proposed RFENN, we re-construct four state-of-the-art models used in diagnostics of unbalanced and shaft crack defects. They are: 1) Continuous wavelet transform scalogram assisted CNN (CWTS) [55], 2) Deep Residual Shrinkage Networks (DRSNs) [56], 3) Deep CNN with support vector machine as classifier (semi-DCNN) [57], and 4) Spatio-temporal fusion neural network (STFNN) [58]. Among them, STFNN is a variant of basic CNN-LSTM architecture. DRSNs build models based on residual blocks that process the raw data directly. Semi-DCNN uses a fusion of two classical architectures, CNN and SVM, and has been shown to outperform the classical one.

However, the above four models only consider fault identification issues and do not allow localization of the defects. To the best of our knowledge, no existing PHM framework in the literature has been developed for both fault identification and defect localization on rotating shafts. In general, the fault identification and localization tasks are performed by two ML algorithms separately, as shown in the papers [59, 60]. Thus, it is necessary to rebuild the existing models in the literature, which aim to localize the defects, to compare with our proposed model. Three popular benchmark models are chosen as follows: 1) ANN [60], 2) LSTM [61], and 3) Extreme learning machine (ELM) [59].

In addition to the above benchmark models, a CNN-LSTM model with the same architecture as our proposed RFEMNN but without the customized physical layers, called NO_RFEM_NN, was investigated. It is used to prove the importance of embedding physics knowledge into ML.

4.5.2. Comparison results

Eqs. 5 and 6 are used to evaluate the multi-fault diagnostics metrics of the proposed RFEMNN model and other benchmark models. The obtained results are shown in Table.8.

Table 8: Comparison of the proposed RFEMNN model with the state-of-the-art models.

Model	Input	A	F	M	Location error			T
					L_U	L_C	$\overline{p}$	
CNN[55]	Wavelet spectrum	75.21%	60.44%	0.73%	-	-	-	-
CNN[55]	Raw data	40.59%	0.6%	18.81%	-	-	-	-
DRSN[56]	Wavelet spectrum	91.48%	2.09%	1.11%	-	-	-	-
DRSN[56]	Raw data	83.07%	35.81%	2.86%	-	-	-	-
Semi-DCNN[57]	Raw data	50.80%	57.56%	9.24%	-	-	-	-
STFNN[58]	Wavelet spectrum	59.35%	58.62%	2.88%	-	-	-	-
ANN[60]	Wavelet spectrum	-	-	-	24.25%	25.90%	75.63%	-
LSTM[61]	Raw data	-	-	-	19.84%	23.66%	78.05%	-
ELM[59]	Raw data	-	-	-	23.87%	24.85%	75.64%	-
NO_RFEM_NN	Raw data	46.45%	3.74%	22.90	9.89%	17.05%	86.53%	40.19%
RFEMNN	Raw data	**97.79%**	9.20%	1.56%	**1.95%**	**4.37**%	**96.84%**	**94.70%**

From Table 8, one can find that:

1. The proposed RFEMNN model has the best diagnostics accuracy that reaches 97.79%. Although its false (F) and missing alarm (M) rates are higher than the ones of DRSNS with the input of wavelet spectrum, the proposed model does not require an additional feature engineering process, i.e., wavelet transformation. Moreover, when comparing with DRSNS using raw data, the proposed RFEMNN model provides better results.

2. By comparing the performance of the two CNN and DRSN models with different inputs, it can be found that the wavelet spectrum, which reflects the structural excitation components and the temporal response information, can improve the diagnostics performance.

3. Although NO_RFEM_NN and CNN provide low false alarm rates, their accuracy metrics and missing alarm rates are unacceptable. Hence, these false alarm rates cannot prove the performance of NO_RFEM_NN and CNN, but are only consequences of the low accuracy level. In fact, by considering the confusion matrices of NO_RFEM_NN and CNN in Appendix 3, one can see that these models misidentify most of the unbalance faults as healthy.

4. Compared with other existing models, the performance of DRSN is better under the same input conditions. Therefore, the appropriate network architecture design has a decisive impact on diagnostics performance.

5. Regarding the defect localization aspects, the proposed RFEMNN model provides the best results compared to all existing models. There is no band bias in the overall predicted data in the prediction results as shown in Fig. 19 and figures in Appendix 5. Therefore, it gives the best value of the total diagnostics performance metric (T) reaching 94.7%.

6. Without physics-embedded layers, i.e., when removing the custom layers and layer connections used to represent the rotor finite element model, the diagnostics performance of NO_RFEM_NNN falls drastically down. The overall diagnostics performance falls to 42.43% of the original. This highlights the importance of physics knowledge in RFEMNN. The PI structure is an essential support for the RFEMNN excellent performance.

5. Conclusion and future works

This paper has developed a new rotor finite element mimetic CNN-LSTM model for rotor unbalance and shaft crack detection, identification and localization. It is called RFEMNN, which allows investigating of different rotor layouts under varying rotor-speed processes. The proposed model simulates the fault diagnosis process of a rotor-based finite element model in multi-task learning by means of a customized PIML layer and its connections. A series of experimental tests verify the effectiveness of RFEMNN. Several metrics have been proposed to assess the accuracy of RFEMNN on multi-fault classification and localization issues. In detail, the scores are evaluated by using a hi-

erarchical 10-fold cross-validation on experimental data. The 10-fold averaged experimental results show that the accuracy of RFEMNN is at 97.79%, the false alarm rate is 9.20%, the fault-detection-missing rate is 1.56%, and the localization accuracy is 96.84%. Then, the total diagnostics performance reaches 94.70%.

The comparison results with existing ML models (one- or multi-dimension CNN, spatiotemporal fusion neural network, semi-DCNN, and deep Residual Shrinkage Networks in state-of-the-art) have shown the superiority of the proposed model. The RFEMNN can exceed the fault recognition performance of state-of-the-art without doing signal preprocessing.

Besides proving the effectiveness of the proposed RFEMNN model for diagnostics of rotor unbalances and shaft cracks, this study also opens up numerous promising research directions in the future:

1. The structural topology and the data flow topology of the FEM derivation process within a neural network architecture can be further explored. The combination of Neural Architecture Search methods with physical objectives should be investigated to optimize the RFEMNN structure.

2. Regarding the multi-task learning process, since the outputs of different tasks are coupled with each other, a new effective design of the multi-objective loss balance task should be developed to facilitate model convergence and thus reduce computational time.

3. The initialization of the model parameters and hyperparameters could be deeply studied and optimized under the guidance of physics knowledge. In addition, studies on the adaptation of the model in a broader sense and for a more comprehensive range of rotor structures are welcome. Thus, the proposed RFEMNN will benefit from the implementation of numerous industrial applications with different rotor configurations and more complex fault types.

References

[1] N. Rezazadeh, A. De Luca, D. Perfetto, Unbalanced, cracked, and misaligned rotating machines: A comparison between classification procedures throughout the steady-state operation, Journal of the Brazilian Society of Mechanical Sciences and Engineering 44 (10) (2022) 1–14.

[2] D. Xiao, Y. Huang, X. Zhang, H. Shi, C. Liu, Y. Li, Fault diagnosis of asynchronous motors based on lstm neural network, in: 2018 prognostics and system health management conference (PHM-Chongqing), IEEE, 2018, pp. 540–545.

[3] D. Soeffker, C. Wei, S. Wolff, M.-S. Saadawia, Detection of rotor cracks: comparison of an old model-based approach with a new signal-based approach, Nonlinear Dynamics 83 (3) (2016) 1153–1170.

[4] R. Liu, B. Yang, E. Zio, X. Chen, Artificial intelligence for fault diagnosis of rotating machinery: A review, Mechanical Systems and Signal Processing 108 (2018) 33–47.

[5] N. Teyi, S. Singh, A decadal review of various modelling and analysis of cracked rotors, Procedia Structural Integrity 39 (2022) 333–346.

[6] J. Wang, Y. Li, Z. Huang, Q. Qiao, Digital twin-driven fault diagnosis service of rotating machinery, in: Digital Twin Driven Service, Elsevier, 2022, pp. 119–138.

[7] N. Kushwaha, V. Patel, Nonlinear dynamic analysis of two-disk rotor system containing an unbalance influenced transverse crack, Nonlinear Dynamics (2022) 1–29.

[8] J. M. Reniers, G. Mulder, D. A. Howey, Review and performance comparison of mechanical-chemical degradation models for lithium-ion batteries, Journal of The Electrochemical Society 166 (14) (2019) A3189.

[9] Z. Zhang, C. Sun, B. Guo, Transfer-learning guided bayesian model updating for damage identification considering modeling uncertainty, Mechanical Systems and Signal Processing 166 (2022) 108426.

[10] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nature Reviews Physics 3 (6) (2021) 422–440.

[11] L. Pagnier, M. Chertkov, Physics-informed graphical neural network for parameter & state estimations in power systems, arXiv preprint arXiv:2102.06349 (2021).

[12] F. Perez-Sanjines, C. Peeters, T. Verstraeten, J. Antoni, A. Nowé, J. Helsen, Fleet-based early fault detection of wind turbine gearboxes using physics-informed deep learning based on cyclic spectral coherence, Mechanical Systems and Signal Processing 185 (2023) 109760.

[13] D. H. Green, A. W. Langham, R. A. Agustin, D. W. Quinn, S. B. Leeb, Physics-informed feature space evaluation for diagnostic power monitoring, IEEE Transactions on Industrial Informatics (2022).

[14] Y. Kim, K. Na, B. D. Youn, A health-adaptive time-scale representation (htsr) embedded convolutional neural network for gearbox fault diagnostics, Mechanical Systems and Signal Processing 167 (2022) 108575.

[15] J. Marcus, Quantifying uncertainty in physics-informed variational autoencoders for anomaly detection, Impact and Opportunities of Artificial Intelligence Techniques in the Steel Industry: Ongoing Applications, Perspectives and Future Trends 1338 (2021) 28.

[16] V. Ewald, R. S. Venkat, A. Asokkumar, R. Benedictus, C. Boller, R. M. Groves, Perception modelling by invariant representation of deep learning for automated structural diagnostic in aircraft maintenance: A study case using deepshm, Mechanical Systems and Signal Processing 165 (2022) 108153.

[17] H. Sun, L. Peng, J. Lin, S. Wang, W. Zhao, S. Huang, Microcrack defect quantification using a focusing high-order sh guided wave emat: the physics-informed deep neural network guwnet, IEEE Transactions on Industrial Informatics (2021).

[18] H. Sun, L. Peng, S. Huang, S. Li, Y. Long, S. Wang, W. Zhao, Development of a physics-informed doubly fed cross-residual deep neural network for high-precision magnetic flux leakage defect size estimation, IEEE Transactions on Industrial Informatics 18 (3) (2021) 1629–1640.

[19] K. Shukla, P. C. Di Leoni, J. Blackshire, D. Sparkman, G. E. Karniadakis, Physics-informed neural network for ultrasound nondestructive quantification of surface breaking cracks, Journal of Nondestructive Evaluation 39 (3) (2020) 1–20.

[20] S. Goswami, M. Yin, Y. Yu, G. Karniadakis, A physics-informed variational deeponet for predicting the crack path in brittle materials, arXiv preprint arXiv:2108.06905 (2021).

[21] S. Goswami, M. Yin, Y. Yu, G. E. Karniadakis, A physics-informed variational deeponet for predicting crack path in quasi-brittle materials, Computer Methods in Applied Mechanics and Engineering 391 (2022) 114587.

[22] B. Hou, D. Wang, Y. Chen, H. Wang, Z. Peng, K.-L. Tsui, Interpretable online updated weights: Optimized square envelope spectrum for machine condition monitoring and fault diagnosis, Mechanical Systems and Signal Processing 169 (2022) 108779.

[23] M. A. Chao, C. Kulkarni, K. Goebel, O. Fink, Hybrid deep fault detection and isolation: Combining deep neural networks and system performance models, arXiv preprint arXiv:1908.01529 (2019).

[24] E. J. Cross, S. Gibson, M. Jones, D. Pitchforth, S. Zhang, T. Rogers, Physics-informed machine learning for structural health monitoring, in: Structural Health Monitoring Based on Data Science Techniques, Springer, 2022, pp. 347–367.

[25] A. Dourado, F. A. Viana, Physics-informed neural networks for missing physics estimation in cumulative damage models: a case study in corrosion fatigue, Journal of Computing and Information Science in Engineering 20 (6) (2020).

[26] A. S. Zamzam, N. D. Sidiropoulos, Physics-aware neural networks for distribution system state estimation, IEEE Transactions on Power Systems 35 (6) (2020) 4347–4356.

[27] W. Liao, D. Yang, Y. Wang, X. Ren, Fault diagnosis of power transformers using graph convolutional network, CSEE Journal of Power and Energy Systems 7 (2) (2020) 241–249.

[28] K. Chen, J. Hu, Y. Zhang, Z. Yu, J. He, Fault location in power distribution systems via deep graph convolutional networks, IEEE Journal on

Selected Areas in Communications 38 (1) (2019) 119–131.

[29] V. N. Ioannidis, A. G. Marques, G. B. Giannakis, A recurrent graph neural network for multi-relational data, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 8157–8161.

[30] M. Sadoughi, C. Hu, Physics-based convolutional neural network for fault diagnosis of rolling element bearings, IEEE Sensors Journal 19 (11) (2019) 4181–4192.

[31] C. Jeong, Y. Yu, D. Mansour, V. Vesslinov, R. Meehan, A physics model embedded hybrid deep neural network for drillstring washout detection, in: IADC/SPE International Drilling Conference and Exhibition, OnePetro, 2020.

[32] T. Li, Z. Zhao, C. Sun, L. Cheng, X. Chen, R. Yan, R. X. Gao, Waveletkernelnet: An interpretable deep neural network for industrial intelligent diagnosis, IEEE Transactions on Systems, Man, and Cybernetics: Systems 52 (4) (2021) 2302–2312.

[33] M. Zhao, M. Kang, B. Tang, M. Pecht, Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes, IEEE Transactions on Industrial Electronics 65 (5) (2017) 4290–4300.

[34] R. T. Chen, Y. Rubanova, J. Bettencourt, D. Duvenaud, Neural ordinary differential equations, arXiv preprint arXiv:1806.07366 (2018).

[35] Z. Lai, C. Mylonas, S. Nagarajaiah, E. Chatzi, Structural identification with physics-informed neural ordinary differential equations, Journal of Sound and Vibration 508 (2021) 116196.

[36] M. Star, K. McKee, Remaining useful life estimation using neural ordinary differential equations, International Journal of Prognostics and Health Management 12 (2) (2021).

[37] Z. Zhang, C. Sun, Structural damage identification via physics-guided machine learning: a methodology integrating pattern recognition with finite element model updating, Structural Health Monitoring 20 (4) (2021) 1675–1688.

[38] M. Russell, P. Wang, Physics-informed deep learning for signal compression and reconstruction of big data in industrial condition monitoring, Mechanical Systems and Signal Processing 168 (2022) 108709.

[39] S. Shen, H. Lu, M. Sadoughi, C. Hu, V. Nemani, A. Thelen, K. Webster, M. Darr, J. Sidon, S. Kenny, A physics-informed deep learning approach for bearing fault detection, Engineering Applications of Artificial Intelligence 103 (2021) 104295.

[40] W. Li, D. Deka, Physics-informed learning for high impedance faults detection, in: 2021 IEEE Madrid PowerTech, IEEE, 2021, pp. 1–6.

[41] H. V. Tetali, K. S. Alguri, J. B. Harley, Wave physics informed dictionary learning in one dimension, in: 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, 2019, pp. 1–6.

[42] A. Odot, R. Haferssas, S. Cotin, Deepphysics: a physics aware deep learning framework for real-time simulation, arXiv preprint arXiv:2109.09491 (2021).

[43] S. Xu, H. Y. Noh, Phymdan: Physics-informed knowledge transfer between buildings for seismic damage diagnosis through adversarial learning, Mechanical Systems and Signal Processing 151 (2021) 107374.

[44] J. Willard, X. Jia, S. Xu, M. Steinbach, V. Kumar, Integrating physics-based modeling with machine learning: A survey, arXiv preprint arXiv:2003.04919 1 (1) (2020) 1–34.

[45] R. Rai, C. K. Sahu, Driven by data or derived through physics? a review of hybrid physics guided machine learning techniques with cyber-physical system (cps) focus, IEEE Access 8.

[46] Y. Chen, D. Huang, D. Zhang, J. Zeng, N. Wang, H. Zhang, J. Yan, Theory-guided hard constraint projection (hcp): A knowledge-based data-driven scientific machine learning method, Journal of Computational Physics 445 (2021) 110624.

[47] S. Cofre-Martel, E. L. Droguett, M. Modarres, Uncovering the underlying physics of degrading system behavior through a deep neural network framework: The case of remaining useful life prognosis, arXiv preprint arXiv:2006.09288 (2020).

[48] J. Sansana, M. N. Joswiak, I. Castillo, Z. Wang, R. Rendall, L. H. Chiang, M. S. Reis, Recent trends on hybrid modeling for industry 4.0, Computers & Chemical Engineering 151 (2021) 107365.

[49] J. Wang, Y. Li, R. Zhao, R. X. Gao, Physics guided neural network for machining tool wear prediction, Journal of Manufacturing Systems 57 (2020) 298–310.

[50] C. Hu, B. Yang, J. Yan, Y. Xiang, S. Zhou, F.-Z. Xuan, Damage localization in pressure vessel by guided waves based on convolution neural network approach, Journal of Pressure Vessel Technology 142 (6) (2020).

[51] S. Zhang, C. M. Li, W. Ye, Damage localization in plate-like structures using time-varying feature and one-dimensional convolutional neural network, Mechanical Systems and Signal Processing 147 (2021) 107107.

[52] S. Fu, K. Liu, Y. Xu, Y. Liu, Rolling bearing diagnosing method based on time domain analysis and adaptive fuzzy-means clustering, Shock and Vibration 2016 (2016).

[53] M. I. Friswell, J. E. Penny, S. D. Garvey, A. W. Lees, Dynamics of rotating machines, Cambridge university press, 2010.

[54] raphaeltimbo, Pt 500 machinery diagnostic system, `https://www.gunt.de/images/download/PT500_english.pdf` (2015).

[55] S. Guo, T. Yang, W. Gao, C. Zhang, A novel fault diagnosis method for rotating machinery based on a convolutional neural network, Sensors 18 (5) (2018) 1429.

[56] M. Zhao, S. Zhong, X. Fu, B. Tang, M. Pecht, Deep residual shrinkage networks for fault diagnosis, IEEE Transactions on Industrial Informatics 16 (7) (2019) 4681–4690.

[57] Y. Xue, D. Dou, J. Yang, Multi-fault diagnosis of rotating machinery based on deep convolution neural network and support vector machine, Measurement 156 (2020) 107571.

[58] Y. Pang, Q. He, G. Jiang, P. Xie, Spatio-temporal fusion neural network for multi-class fault diagnosis of wind turbines based on scada data, Renewable Energy 161 (2020) 510–524.

[59] Y. D. Mamuya, Y.-D. Lee, J.-W. Shen, M. Shafiullah, C.-C. Kuo, Application of machine learning for fault classification and location in a radial distribution grid, Applied Sciences 10 (14) (2020) 4965.

[60] A. Said, S. Hashima, M. M. Fouda, M. H. Saad, Deep learning-based fault classification and location for underground power cable of nuclear facilities, IEEE Access 10 (2022) 70126–70142.

[61] S. Belagoune, N. Bali, A. Bakdi, B. Baadji, K. Atif, Deep learning through lstm classification and regression for transmission line fault detection, diagnosis and location in large-scale multi-machine power systems, Measurement 177 (2021) 109330.

Appendix: Diagnostics results for different benchmark models

Confusion matrices for fault identification results

Convolutional neural network

Table 9: Confusion matrices of CNN (raw data as input) for fault type identification[55]

	Diagnostics results				
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	1783	11	0	21
	unbalance	1199	70	0	1094
	Crack	4	12	0	1789
	Im&Cra	0	54	0	1751
	Total:	765	2665	241	2723

Table 10: Confusion matrices of CWSCNN for fault type identification[55]

	Diagnostics results				
		Healthy	**unbalance**	**Crack**	**Im&Cra**

Real status	**Healthy**	718	1097	0	0
	unbalance	38	1452	0	873
	Crack	9	62	241	99
	Im&Cra	0	54	0	1751
	Total:	765	2665	241	2723

Deep residual shrinkage networks

Table 11: Confusion matrices of DRSNs (wavelet spectrum as input) for fault type identification[56]

	Diagnostics results				
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	1777	38	0	0
	unbalance	71	2035	52	205
	Crack	0	0	400	11
	Im&Cra	0	80	1	1724
	Total:	1848	2153	453	1940

Table 12: Confusion matrices of DRSNs (raw data as input) for fault type identification[56]

	Diagnostics results				
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	1156	650	4	5
	unbalance	170	1917	37	239
	Crack	13	5	347	46
	Im&Cra	0	253	12	1540
	Total:	1339	2825	400	1830

Convolutional neural network with SVM as the final classifier

Table 13: Fault type identification confusion matrices of CNN-SVM[57]

	Diagnostics results				
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	893	572	9	341
	unbalance	418	1079	38	828

	Crack	83	148	48	132
	Im&Cra	90	205	1	1509
	Total:	1482	2004	96	2810

Spatio-temporal fusion neural network

Table 14: Fault type identification confusion matrices of STFNN[58]

		Diagnostics results			
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	751	346	0	718
	unbalance	184	913	0	1266
	Crack	0	0	0	411
	Im&Cra	0	0	0	1805
	Total:	935	1259	0	4546

No-RFE NN (without physical knowledge)

Table 15: Fault type identification confusion matrices of STFNN

		Diagnostics results			
		Healthy	**unbalance**	**Crack**	**Im&Cra**
Real status	**Healthy**	1747	64	1	3
	unbalance	1414	307	6	636
	Crack	50	106	74	181
	Im&Cra	2	57	0	1746
	Total:	3213	534	81	2566

Fault location results

Artificial Neural Networks

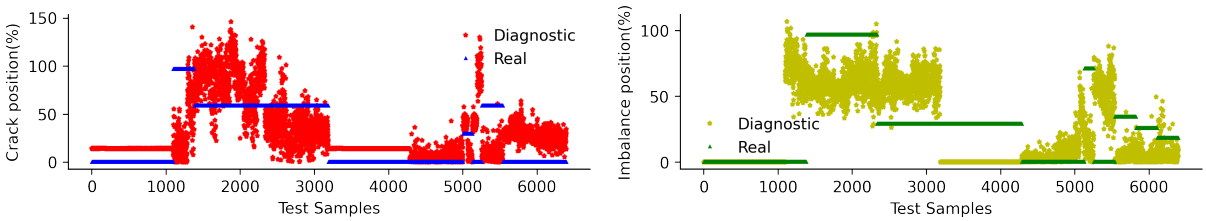

Figure 16: Average results of ANN based fault localization with 10 fold-cross validation[60]

Long and short-term memory neural networks

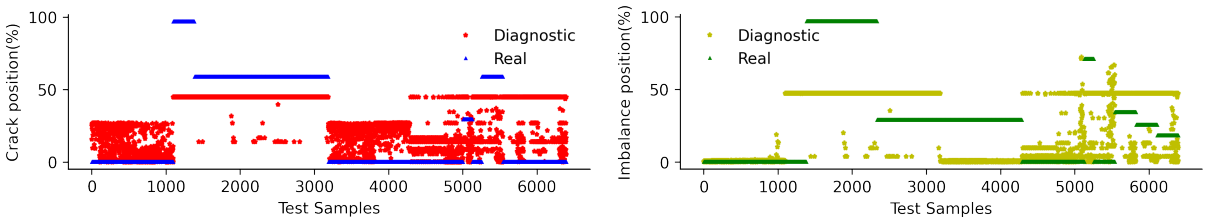

Figure 17: Average results of LSTM based fault localization with 10 fold-cross validation[61]

Extreme Learning Machine

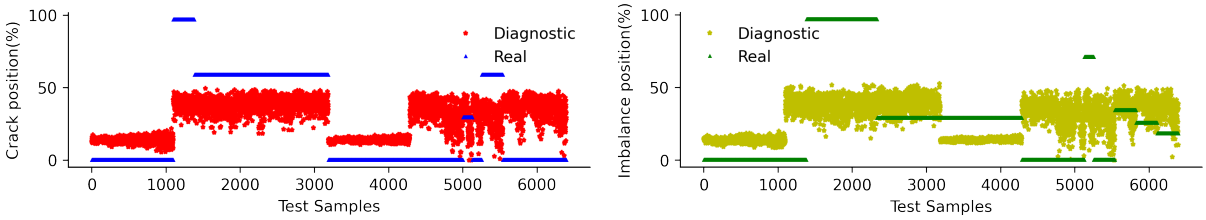

Figure 18: Average results of ELM based fault localization with 10 fold-cross validation[59]

NO_FEM_NN

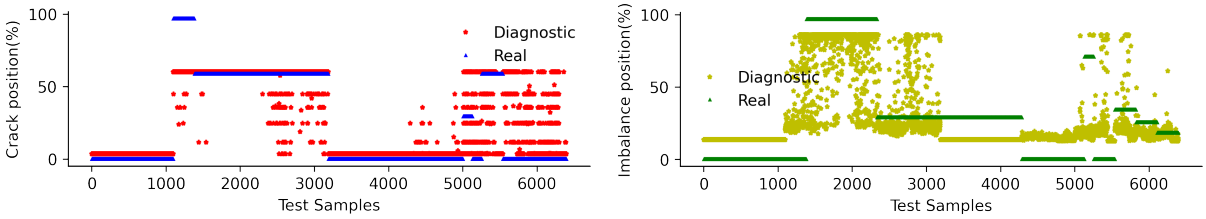

Figure 19: Average results of NO_FEM_NN based fault localization with 10 fold-cross validation