Computers in Industry 144 (2023) 103771

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.sciencedirect.com/journal/computers-in-industry

Advanced Prognosis methodology based on behavioral indicators and Chained Sequential Memory Neural Networks with a diesel engine application

Pablo Calvo-Bascones *, Miguel A. Sanz-Bobi

Department of Intelligent Systems, Comillas Pontifical University - Institute for Research in Technology, Madrid, Spain

ARTICLE INFO

Keywords:
Prognosis
Fault diagnosis
Behavior characterization
LSTM
Diesel engines

ABSTRACT

This paper presents a novel methodology in the field of Prognosis and predictive Maintenance (PdM) of industrial components. It has been designed as an inclusive PdM approach grounded on flexible strategies capable of characterizing the behavior of an industrial system regardless of its nature in terms of its physics, dynamics, or evolution in time. The proposed method includes two behavioral indicators computed through a robust method based on Behavior Patterns. These two indicators (Deviation and Similarity) provide a precise characterization of the behaviors of an industrial system. The prognosis of both indicators is carried out through three different Neural Network (NN) architectures: a Multilayer Perceptron (MLP) and two types of Long–Short Term Memory (LSTM) NNs with two different configurations. Among these configurations, this study proposes a novel LSTM architecture characterized by its Chained Sequential Memory (CSM) architecture based on Peephole Connections. The three architectures are studied and compared in detail in order to determine which one achieves better results in prognosis. The originality of this approach lies in the prognosis of behaviors by applying indicators to enhance and make more intuitive the characterization and prognosis of the state of the system. The proposed LSTM CSM architecture reduces the forecast error by around 50% in comparison to MLP and Stacked LSTM architectures. This study includes an application to a real case in which the new methodology is implemented for the prognosis of the cooling system of a power plant diesel generator. The results obtained prove the advantages and possibilities that the proposed methodology has for industrial applications.

1. Introduction

Prognostics and health management (PHM) of industrial systems has been one of the leading research areas over the last decades due to the remarkable advances in the field of the Internet of Things. Data-driven Artificial Intelligence (AI) approaches are widely used in PHM applications (Khan and Yairi, 2018). PHM can be addressed from multiple points of view depending on its application: prediction/prognosis of the remaining useful life (Stetter, 2020); reliability and failure detection (Zhao, 2021; Xu et al., 2020), component degradation assessment (Alaswad and Xiang, 2017) or normal behavior assessment (Gil et al., 2018).

Among the current trends in Industry 4.0, such as wireless connectivity, data analytics, or improvements in adaptation and flexibility, Prognosis and Predictive Maintenance (PdM) is among the applications with the highest interest in the industry (Zonta et al., 2020) due to its proven benefits in reducing downtimes owing to failures in components, maintenance costs, etc.

This paper is grounded on an inclusive methodology for prognosis based on behavioral indicators. The such inclusive methodology is conceived to be compatible with a wide range of engineering systems, the behaviors of which are determined by physical measurements. The motivation behind this type of behavior characterization is to provide a set of robust first-approach procedures that can be easily implemented and understood by non-expert technicians in the field of data analytics. Such procedures allow for a deep characterization of behaviors and deviations to be learned by the prognosis model. The proposed inclusive methodology eases its implementation in several systems through automated tuning processes (heuristic methods, e.g. Optimum number of clusters obtained through the elbow method), unsupervised learning algorithms (no prior knowledge required, e.g. Self Organizing Maps), white-box approaches (human-interpretable models, e.g.

* Correspondence to: Department of Intelligent Systems, Comillas Pontifical University - Institute for Research in Technology, Sta. Cruz de Marcenado, 26, 28015 Madrid, Spain.
E-mail address: pcalvo@comillas.edu (P. Calvo-Bascones).

https://doi.org/10.1016/j.compind.2022.103771
Received 14 March 2022; Received in revised form 19 July 2022; Accepted 27 August 2022
Available online 14 September 2022
0166-3615/© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Behavior Patterns), and indicators (more intuitive assessments). These automated unsupervised learning approaches ease their implementation into systems of different natures in comparison to other custom approaches based on physical models. The second part of this methodology focuses on the prognosis of the system, predicting the evolution of the two proposed indicators applying Long Short Term Memory (LSTM) Neural Networks (NN). The capabilities and benefits of this prognosis methodology are assessed in a real case of a diesel power generator.

Previous studies like (Venkatasubramanian, 2005) defend that any prognosis methodology has to take into account multiple aspects throughout its designing process: (1) *Quick detection and diagnosis*. The quicker the detection is, the quicker the response becomes. (2) *Isolability*, different failures must be identified separately to adapt the response according to the type of failure. (3) *Robustness* against false positive and negative alarms. (4) *Novelty identifiability*, new behaviors must be differentiated from anomalous behaviors. (5) *Classification error estimates*, deviations, and failures must be quantified and classified according to the confidence threshold defined for the diagnostic decisions. (6) *Adaptability* to new behaviors that the system might present due to changes in external inputs or external changes. (7) *Explanation facility*, the information provided must be easily understood by the operator to explain the origin and evolution to the current state. (8) *Modeling deployment complexity*: The modeling efforts must be optimized to be as minimal as possible without losing accuracy. (9) *Storage and computational requirements*, depending on the application, real-time solutions must take into account higher constraints in terms of storage and computational requirements. (10) *Inclusivity*, the procedures applied within the methodology have to be flexible enough to be compatible with systems and behaviors of different natures.

A recent anomaly detection methodology based on behavior patterns (Calvo-Bascones et al., 2021) proposes an effective method for characterizing the behavior of a system based on behavior patterns and two behavioral indicators. The use of this methodology is justified according to the previous requirements. This methodology defines the boundaries of a normal behavior through behavior patterns based on Self-Organizing Maps (SOMs) (Kohonen, 1982). Collecting the main features of a behavior into most representative clusters improves the robustness (3) of the model against outliers. It is easily adaptable, flexible(10), and scalable to new behavior conditions (6) as each new condition can be learned by the model including those samples as part of the training set of the SOM. It reduces the storage and computational requirements (9) of the anomaly detection model making it compatible with online applications because the dataset is reduced to just the most representative values. New behaviors can be easily differentiated from anomalous behaviors (4) depending on the variations registered within the behavior patterns. Anomalies and failures are easily identified and quantified (2) through two behavioral indicators computed for each input of the system independently. The detection of anomalies and diagnosis is remarkably fast (1) as they are detected as soon as the values of the indicators surpass the confident thresholds (5) that define normal behavior conditions.

The anomaly detection methodology used as a starting point of the proposed prognosis methodology requires, as it was previously mentioned, the composition of behavior patterns using SOMs. Each neuron of the SOM is made up of a centroid value, its probability density function, and its standard deviation. These last two elements are used to compute two behavioral indicators.

This paper aims to build a prognosis methodology upon a robust anomaly detection methodology with the intent to predict and assess the evolution of behaviors to their current state.

This study started this section with a brief introduction to the main requirements that a prognosis methodology must have and how the proposed methodology fulfills those requirements. Section 2 continues presenting different works related to the multiple areas covered throughout this study and its main contributions. In Section 3, the preliminary bases are described, including the presentation of the behavior characterization methodology as the starting point of the prognosis methodology. Section 4 explains the steps followed to implement the proposed prognosis methodology. Section 5 describes the main features of the NN architectures implemented as part of the prognosis stage of the methodology. Section 6 introduces the main characteristics of the variables used to describe the behavior of the diesel engine. Section 7 presents how the methodology proposed has been applied to the study of the cooling system of the engine. The results obtained are presented in Section 8 and finally, Section 9 ends with the conclusions reached from this study.

2. Related works and contributions

Nowadays, the type of maintenance applied to an industrial system can be briefly summarized in two different strategies: failure-based and condition-based. Failure-based strategies do not require any indicator or prognosis. On the other side, condition-based strategies require not only effective indicators about the behavior of a system but also their evolution in time in addition to information about maintenance tasks, etc.

The proposed methodology aims to predict how the behavior or state of a system will evolve over time and how it will affect the scheduling of the next maintenance tasks.

A wide range of AI, Machine Learning (ML), and Deep Learning (DL) approaches has been implemented in PdM applications (Zhang et al., 2019). Neural Networks (NN) are frequently implemented in fault prognosis applications. Traditional architectures such as Multilayer Perceptrons still play an important role in this field (Hou et al., 2020). Other complex architectures based on Recurrent Neural Networks (RNN) are frequently implemented due to their proven capabilities in the prediction and forecast of time series, in particular, Long Short Term Memory (LSTM) networks (Yu et al., 2019; Jalali et al., 2021; Mejia et al., 2021).

Previous works in the field of Predictive Maintenance (PdM) and condition-based assessments are built on LSTM NN. This study assesses different LSTM architectures that range from common configurations such as Stacked LSTMs (Sayah et al., 2020; Jalali et al., 2021) to other more complex LSTM architectures with state memory between LSTM cells (Yu et al., 2019) or stacked attention blocks (Su et al., 2021). This study aims to compare the level of accuracy achieved for different architectures.

Some limitations have been identified in the literature on PdM:

- The first limitation is the need for a predefined Remaining Useful Life (RUL) scenario used to train the prognosis model (Cheng et al., 2021). This type of approach presents several limitations in cases where RUL scenarios are not available. The methodology proposed in this study allows for the assessment of the system degradation based on the two indicators proposed. The RUL of the system is approximated by the temporal evolution of such indicators.
- Other studies like (Venkatesan et al., 2019) estimate the RUL of the components based on mathematical models. This type of approach presents some limitations regarding its generalization capabilities which hinder the application of the same methodology to a different system. The present study allows for the assessment of the health condition of the system based on the deviations registered in its behaviors and not in mathematical models which are specific to a particular system and might be subjected to temporal changes.
- PdM methodologies designed for systems where all the possible operating conditions are determined beforehand present limitations related to the inclusion of new operating conditions (Arunthavanathan et al., 2021). The methodology proposed in the present study allows for including additional operating conditions in the shape of behavior patterns that can be easily stacked to the previous ones.

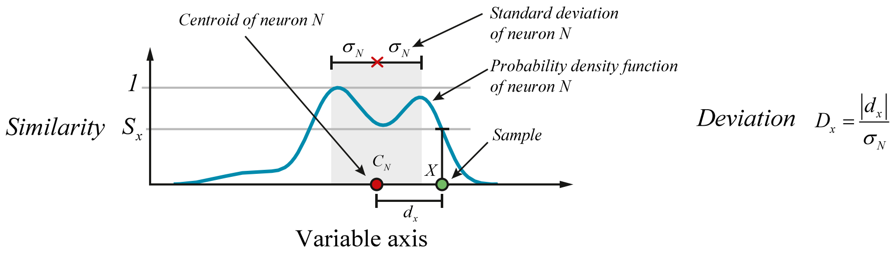

Centroid of neuron N
Standard deviation of neuron N
σ_N σ_N
Probability density function of neuron N
1
Similarity S_x
Sample
C_N
X
d_x
Variable axis
Deviation $D_x = \frac{|d_x|}{\sigma_N}$

Fig. 1. Representation of **Similarity** and **Deviation** indicators given the probability density function, the centroid, and the standard deviation of a particular neuron N.

Inspired by the previous works, the contributions of the present study can be summed up in:

- This study proposes a novel prognosis methodology based on behavioral indicators. These behavioral indicators are obtained from a previous methodology grounded on behavior patterns. The combination of both methodologies aims to achieve better prognosis and forecast capabilities.
- Three types of NNs are thoroughly compared to determine which of them performs better in a real prognosis application. A custom LSTM architecture is proposed to enhance the capabilities of this type of Recurrent NN. This type of architecture was termed LSTM with chained sequential memory (CSM LSTM).
- This study presents a real case application of the methodology. The methodology described is applied to the behavior of the cooling system of a diesel engine. This real case shows not only how this methodology succeeds in the prognosis of a real system but also how this methodology can be implemented step by step.

3. Preliminary bases. Behavior characterization.

Different strategies are commonly applied in the field of behavior characterization and anomaly detection. Xu et al. (2020), several approaches based on Belief-Rule Based (BRB) expert systems, Artificial NN, or Support Vector Machines are presented and compared. The methodology proposed in this paper is built upon an anomaly detection methodology grounded in the composition and ensuing analysis of behavior patterns (Calvo-Bascones et al., 2021). A behavior pattern is a characterization of a set of collected samples during the operation of the system. This characterization is a segmentation of a behavior into Operating Modes (OM)s. Each OM is obtained through unsupervised clustering algorithms. The centroid of each cluster determines the OM to which a sample belongs based on their Euclidean distance. This previous methodology proposes two types of indicators. The first indicator is called **Deviation** and compares the distance of a sample to the centroid of its closest neuron normalized using the standard deviation of such a neuron. The second indicator is called **Similarity** and assesses the location of a sample within the Probability Density Function (PDF) of its closest neuron. **Deviation** has an strictly positive non upper limited range of values $[0, \infty]$, on the other side, **Similarity** values are defined within the range $[0, 1]$. Fig. 1 shows how both indicators are computed given a test sample X and the PDF, the closest centroids of the SOM, and the standard deviation of its closest neuron N in one of its multiple dimensions. These two indicators aim to improve the feature extraction based on traditional Time-Domain features like the ones presented in Abid et al. (2020), which underperform under complex behavior scenarios.

The use of behavior patterns presents multiple benefits (Calvo-Bascones et al., 2021):

- This method presents high robustness against outliers. Samples that are less representative within a behavior are also less representative within the pattern.
- In those cases where the number of observations under assessment is significantly large, patterns allow for a reduction in the volume of information that has to be stored and processed in the assessment.
- Behavior patterns can be used to assess other behavior patterns or individual samples. This means that the behavior assessment can be carried out in a “pattern vs. pattern" or “pattern vs. observations" manner. This study applies the “patterns vs. observations” strategy since it presents fewer limitations for offline and online applications.

The two indicators proposed present multiple advantages in comparison to other similar methods based on Gaussian Mixture Models (GMMs) (Zhang et al., 2020). GMMs present several limitations when the observations that make up each OM present a multi-modal distribution or the boundary regions have a high degree of skewness. In those cases, gaussian modeling presents some disadvantages overcome by the joint use of the two indicators proposed.

One of the advantages of using these two indicators jointly is an effective detection of deviations in cases where a distance from the centroid is not enough to characterize an observation. Fig. 2 (Calvo-Bascones et al., 2021) shows two cases in which the joint use of these two indicators allows for a better characterization of the observation regarding its reference OM; where a normal behavior condition is defined as:

$$[S_{f,n} > S_{f,\min}] \wedge [D_{f,n} < 2.5] \tag{1}$$

Fig. 2 shows two different cases in which only one condition is fulfilled. Case a) shows an OM distribution in which the centroid ($C_{OM,f}$) is located in a region with a low density of samples. In this case, those observations that are located too close to the centroid would be considered as part of a normal behavior when only the distance to the centroid of the OM is considered. Taking into account the Similarity indicator, the normal behavior condition is not fulfilled any longer. Case b) shows an opposite situation in which there is a region located outside of the distance threshold delimited by $\pm 2.5\sigma$, but regarding its Similarity value, it is above the minimum acceptance value ($S_{f,\min}$).

4. Description of the prognosis methodology

The workflow of the methodology proposed is shown in Fig. 3. This methodology starts with the composition of reference (normal) behavior patterns (Abid et al., 2020) obtained from samples collected during normal behaviors. These samples do not have to be registered sequentially due to the fact that behavior models do not take into account the sequentiality of the samples, only the distribution of their values. This fact allows for building reference behavior models from multiple working conditions. A working condition is defined by the configuration set by the operator or controller of the system to determine its behavior. The methodology continues computing both indicators from a sequential training behavior dataset (Alaswad and Xiang, 2017). From the sequential values obtained for both indicators, a predictive model is trained as part of the Prognosis Model (Arunthavanathan

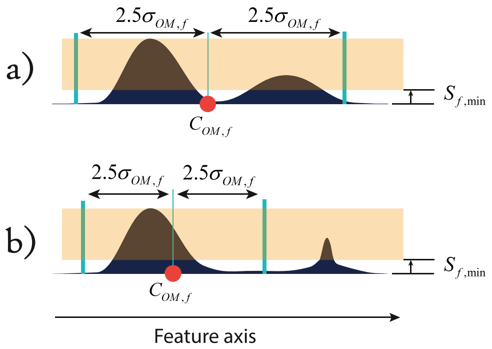


Fig. 2. Particular cases in which the joint use of distance and similarity indicators provide a better assessment of the observation regarding its reference OM.

et al., 2021). This prognosis model is based on machine learning algorithms capable of learning the temporal features and evolution of both indicators to carry out the prognosis of the system. The last step in the methodology is the application of the predicted values of both indicators (Calvo-Bascones et al., 2020) to multiple applications in the field of prognosis.

The steps followed to compute the reference behavior pattern are summarized in Algorithm 1.

Algorithm 1: Computation of the reference behavior pattern

```
Input: Reference behavior data set
Output: Reference behavior pattern
Compute the optimal number of $OM$ through a K-means and
  the elbow method;
for each $OM$ do
    compute its centroid using a SOM;
    compute its PDF ;
    compute standard deviation ;
end
```

The steps followed to train the prognosis model are summarized in Algorithm 2.

Algorithm 2: Training of the prognosis model

```
Input: Reference degradation data set, number of input
       samples $t$, number of output (forecast) samples $k$.
Output: Prognosis Model
for each observation ∈ Reference degradation data set do
    match to its closest $OM$ in the reference behavior pattern;
    compute its Similarity value;
    compute its Deviation value;
end
extract input time windows of size $t$ from the sequence of
  Similarity and Deviation values;
for each input Similarity and Deviation time window do
    extract the expected time window of size $k$ from the
      sequence of Similarity and Deviation values;
end
Start NN training process with input and output-expected
  windows;
```

The main contribution of this study is the application of behavior patterns and smart indicators to prognosis applications. This contribution is carried out in stages (Arunthavanathan et al., 2021; Calvo-Bascones et al., 2020) of Fig. 3. So far, this behavior characterization approach has been implemented only in anomaly detection applications (Wang et al., 2021), Calvo-Bascones et al. (2021). This study aims to extend the application field of such a previous methodology to the field of prognosis. This contribution is accomplished in the prognosis model through a new LSTM architecture and its application to RUL assessments, maintenance planning, or resilience analysis. The capabilities of this extension are proven in a real-case application.

The next section briefly explains the main features of the algorithms implemented in the second step of the methodology as part of the prognosis stage.

5. Predictive algorithms

Different predictive algorithms are proposed in the forecast of the indicators introduced in Section 3. The algorithms proposed range from a well-known straightforward architecture to a more complex architecture that includes all the previous ones.

5.1. Multilayer Perceptron

The Multilayer Perceptron (MLP) is one type of feed-forward neural network, which consists of an input layer, an output layer, and one or more hidden layers. MLPs are commonly applied to the study of engine behaviors, e.g. Dhahad et al. (2022) proposes an application of MLPs for prognostic the performance of the internal combustion of a diesel engine.

5.2. Long Short Term Memory neural networks

The motivation behind the architecture of LSTM networks is to deal with “long-term dependencies”. The architecture of an LSTM cell has varied throughout time from its original form (Hochreiter and Schmidhuber, 1997). These variations range from forget-gates to peephole connections (Yu et al., 2019). The standard inner layout of an LSTM cell is shown in Fig. 4
where X_t is the input vector at time t. σ and $tanh$ are activation functions. The outputs of the three sigmoid functions control the information that is kept from the previous cell state C_{t-1} and passed to the new cell C_t. h_t is the output information from the current cell.

An example of the application of LSTMs in the diagnosis of a diesel power generator can be found in Kayaalp et al. (2021). This study focuses on the analysis of the combustion efficiency applying this type of NN.

6. System description

The system under assessment corresponds to a diesel generator of a power plant made up of nine cylinders equivalent to the motor shown in Fig. 5. This two-stroke diesel engine has a power generation of 3280 kW. Prognosis applications of Diesel generator engines of this size are rarely found in the literature. Smaller engines corresponding to vehicles or small generators present lower criticality, and less marked degradation trends, allowing for a wider margin of error in the scheduling of maintenance tasks. These conditions make the proposed case study a valuable application from a technical (characterize the degradation of the system, anticipate potential failures, improve the maintenance scheduling process), economic (reduce the costs derived from inefficient maintenance planning and system break-downs), and energy-supply-planning perspectives (avoid unexpected halts and power cuts).

Each one of the nine cylinders is endowed with a set of temperature sensors that provide a detailed characterization of the current state of each cylinder. A simplified scheme of the gas–water recirculation system is presented in Fig. 6. The scope of this study is the analysis of the state of the cylinders through the temperature measurements obtained from its cooling system. The cooling system of the diesel engine presents a gas circuit, shown in gray, and a water circuit,

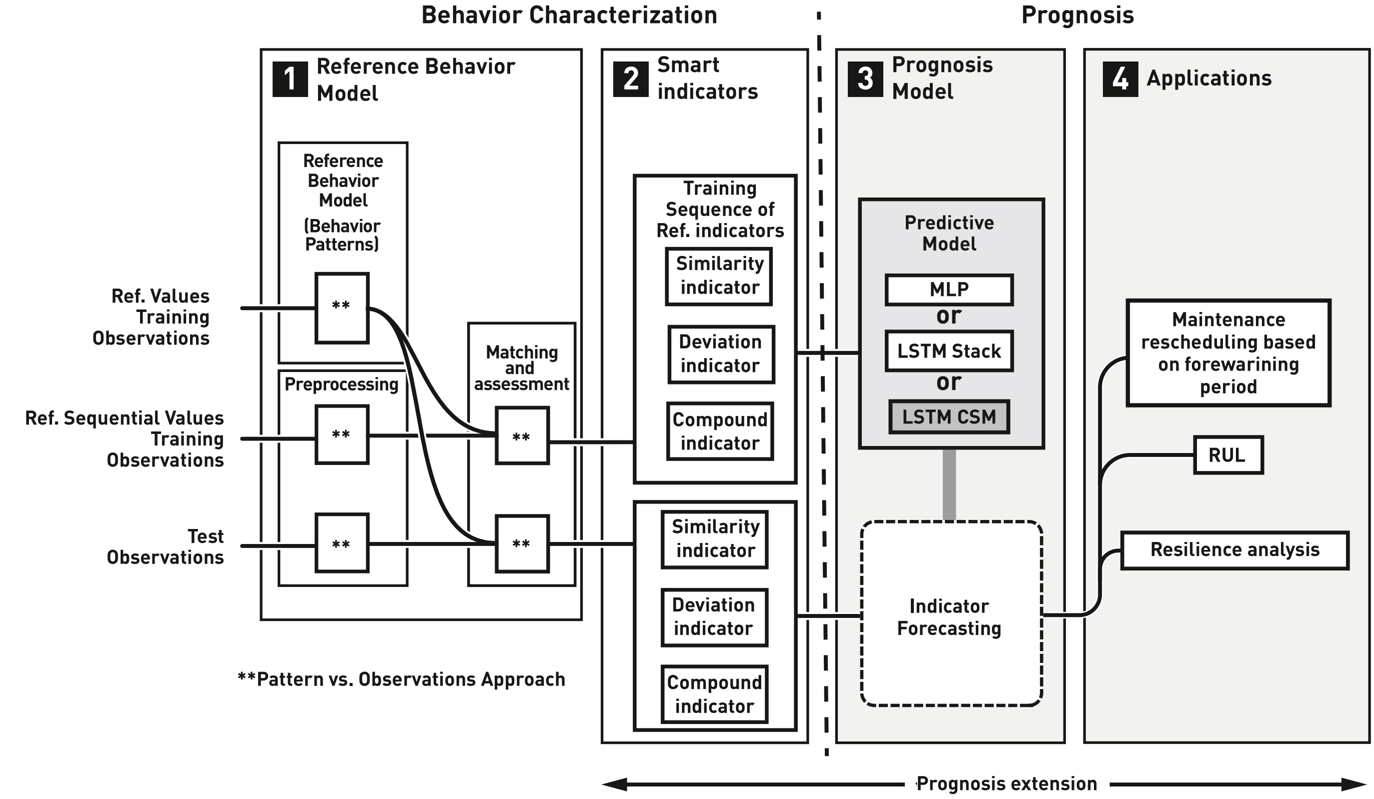

Behavior Characterization
Prognosis
1 Reference Behavior Model
2 Smart indicators
3 Prognosis Model
4 Applications
Reference Behavior Model (Behavior Patterns)
Preprocessing
Matching and assessment
Ref. Values Training Observations
Ref. Sequential Values Training Observations
Test Observations
**
Training Sequence of Ref. indicators
Similarity indicator
Deviation indicator
Compound indicator
Predictive Model
MLP
or
LSTM Stack
or
LSTM CSM
Indicator Forecasting
Maintenance rescheduling based on forewarining period
RUL
Resilience analysis
**Pattern vs. Observations Approach
Prognosis extension

Fig. 3. Structure of the methodology proposed.

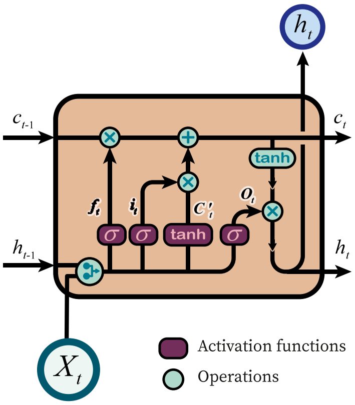


Fig. 4. Scheme of an LSTM cell.

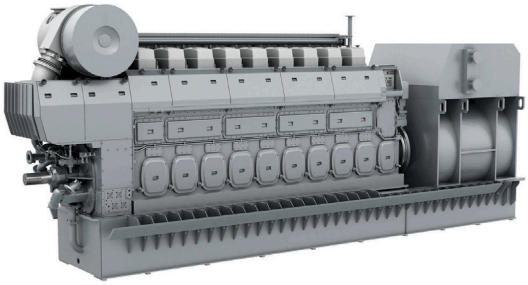


Fig. 5. Example of a diesel generator.

shown in blue. The critical part of the cooling system is located in the cylinders of the engine, where the greatest amount of thermal energy has to be evacuated. The other three main heat exchanger systems are located at the beginning of the water circuit (heat exchanger) and between the gas and water circuits, such as the inter-cooler and the Exhaust Gas Recirculation (EGR) system. These components are crucial in the cooling process of any engine. This study focuses on the analysis of degradations of the cylinders quantified through the temperatures registered in the gas at the intake and exhaust manifolds and in the coolant at the outlet of the engine block. Other systems within the gas circuit are the Turbine-Compressor and the Selective Catalytic Reduction (SCR). The analysis of these systems is out of the scope of this study. Therefore, they have not been included as part of this analysis.

The aim of this study is the assessment of the engine behavior, in particular the study of its cooling system. The cooling system is critical for the operation of the engine. The prognosis of deviations in cooling temperatures is crucial from the point of view of performance and reliability. Too low coolant temperatures might lead to underperforming behaviors for the cooling system, and too high temperatures might lead to accelerated degradation of the engine. If exhaust gas temperature is low, then the cooling system is working with good efficiency. This study is focused on the study of several temperatures registered at different locations of the engine. The state of each cylinder is characterized by its (1) sweeping air temperature, (2) exhaust gas temperature, and (3) exhaust water temperature. Taking into account also the gross power generated, the variables included as part of this study are:

- Gross power: Power generated by the engine (MW). This variable allows for determining the operation state of the engine.
- Sweeping air temperature: temperature of the air before entering the combustion chamber. (° C).
- Exhaust gas temperature: temperature of the gas leaving the combustion chamber (° C) after the diesel pressure ignition.

Sweeping air and exhaust gas temperatures are crucial to know the conditions and how the temperatures of the engine cylinders evolve.

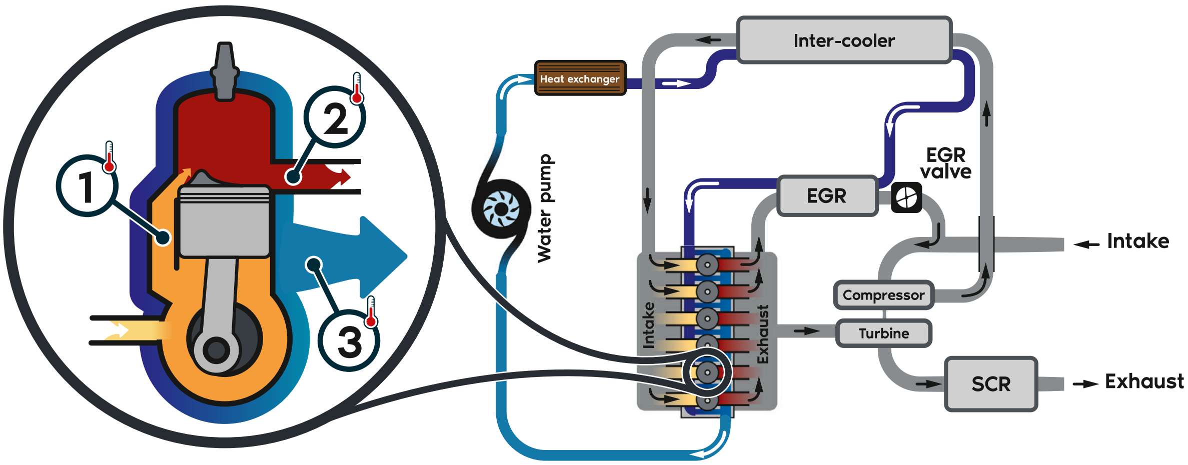


Fig. 6. Simplified scheme of the cooling system of a diesel engine.

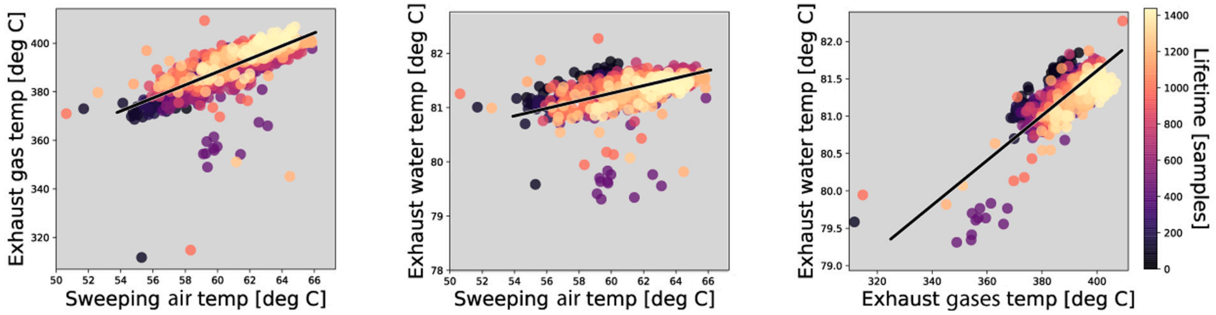


Fig. 7. Correlation between temperatures.

- Exhaust water temperature: temperature of the cooling water (° C) after surrounding the cylinder liners.

The temperature of the exhaust water provides an overview of the general state of the cooling system. Water temperatures show higher inertia, this variable shows a higher resilience against abrupt changes in temperatures of the rest of the system. Therefore, it is a good indicator to know the state of the engine regarding its cooling system.

A visual representation of the training set and their correlation lines between temperatures is shown in Fig. 7. When variables present a good correlation between them, the accuracy of the predictive model improves. The lower the dispersion along the black correlation lines, the better the correlation between variables becomes.

The dataset used in this study corresponds to a period of three years (~ 23000 samples). Temperature measurement systems in these types of generators are typically Resistance Temperature Detectors (RDTs), with an accuracy lower than 0.5 °C. There are no records of tasks of calibration or replacement of sensors; thus, the measurement devices are considered the same for the whole testing campaign.

The next section explains how the methodology proposed is applied to this system.

7. Normal behavior and prognosis models

As it was mentioned in Section 3, the methodology proposed is built upon an anomaly detection model (Normal Behavior Model) and a Prognosis Model. This section explains how both models are obtained.

7.1. Normal behavior model

The first component of the methodology is the composition of a Normal Behavior Model based on Reference Behavior Patterns (RBPs).

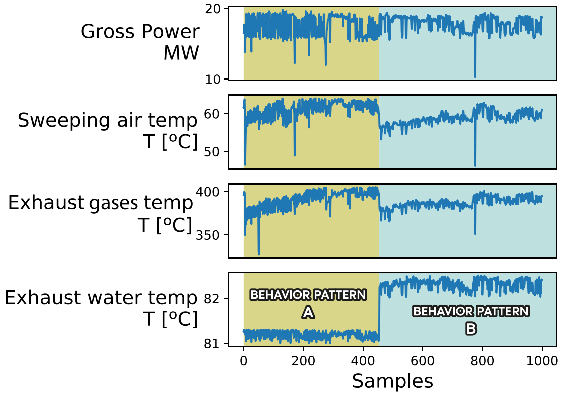


Fig. 8. Training set used to build the Reference Behavior Pattern.

An RBP is computed from normal behavior samples. A normal behavior might include several *sub-behaviors* defined by the multiple configurations that the system might present. In the case of the diesel engine, there are two different configurations to be taken into account. These two operating conditions are related to the size of the flow of the exhaust water, which has a direct effect on its temperature. In case several configurations belong to the same RBP, a behavior pattern has to be obtained for each configuration to be finally merged into a single RBP. The training dataset used to build the RBP is shown in Fig. 8. The final RBP is the combination of behavior patterns A and B.

An RBP is essentially a clustering of the main features of a behavior. Following the methodology proposed in Calvo-Bascones et al. (2021), the clustering algorithm applied is a Self Organizing Map (SOM) (Kohonen, 1982). The optimum number of clusters is determined through

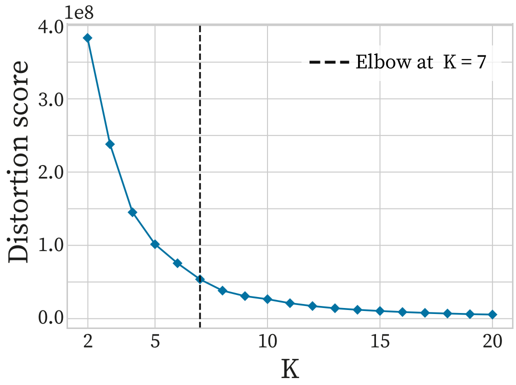


Fig. 9. Optimum number of clusters determined through the Elbow criterion applied to the distortion value obtained through K-means.

Table 1
Centroid values of the Reference Behavior Pattern.

	Centroid values			
	Power generated [kW]	Sweeping air T. [°C]	Exhaust gases T. [°C]	Exhaust water T. [°C]
A	15268.88	56.62	380.31	81.14
	18831.17	60.76	390.95	81.26
	16102.84	60.91	396.36	81.12
	18734.17	62.93	398.63	81.22
B	18028.36	63.09	399.82	82.27
	16044.75	60.43	387.90	82.16
	18016.42	57.55	383.53	82.29

Table 2
Standard deviation values of the Reference Behavior Pattern.

	Standard Deviation			
	Power generated [kW]	Sweeping air T. [°C]	Exhaust gases T. [°C]	Exhaust water T. [°C]
A	1589.38	3.71	19.21	0.053
	196.79	1.45	6.97	0.048
	440.67	1.98	10.78	0.062
	385.83	1.42	7.60	0.066
B	431.48	2.03	6.67	0.075
	755.73	2.13	9.11	0.093
	294.42	2.38	7.90	0.085

the elbow criterion applied to the distortion values obtained using a K-means algorithm (Kodinariya and Makwana, 2013). The cooperation of SOMs and K-means is widely known (Van Laerhoven, 2001). Distortion values refer to the sum of square distances between the samples of each cluster to their centroid. As it can be seen in Fig. 9, the optimum number of clusters was set to 7; divided into 4 clusters for behavior pattern A and 3 clusters for behavior pattern B. This distribution was made taking into account the higher variability that Behavior A shows regarding its sweeping air and exhaust gas temperatures.

In addition to the centroids of the clusters, it is also necessary to compute the PDFs and the standard deviations of each cluster for each variable. Both elements are required to calculate the indicators of **Similarity** and **Deviation**.

The centroid values obtained are shown in Table 1, the standard deviation values obtained for each cluster are shown in Table 2 and two examples of PDFs are shown in Fig. 10.

As seen in Table 2, cluster 1 presents significantly higher standard deviation values compared to the other clusters, not only in temperatures but, most surprisingly, in generated power. This big variation is due to the fact that this cluster comprises all the transient behaviors of the engine. Transient behaviors present on average the lowest power generated and low-temperature values.

The Normal Behavior Model is used to compute the values of **Similarity** and **Deviation** for the rest of the dataset. The prognosis of **Similarity** and **Deviation** values obtained is the task of the Prognosis Model.

7.2. Prognosis models

This model is in charge of predicting the values of **Similarity** and **Deviation**. The original dataset has a sample rate of one sample per hour. Due to the fact that variations in **Similarity** and **Deviation** are not representative enough after each hour, it is necessary to group the samples in longer periods to make such variations appreciable. Therefore, the dataset used to train, validate and test the Prognosis Model was grouped into weeks (~ 150 samples). Samples were grouped in calendar weeks; this means that each group is made up of all the samples registered from Monday to Sunday. At most, 168 samples per group.

Three types of architectures are proposed for this case study, all of them are based on the algorithms presented in Section 5. These architectures are shown in Fig. 11. The input vector X_T is made up of T sequential time distributed samples used to make the prediction of the K samples of the output vector Y_K.

The input and output of the three architectures are the same for all of them. The number of input samples is t, and the number of output samples is k. The first architecture (a) represents a concatenation of MLPs. The second architecture (b) is a basic concatenation of an LSTM layer and an MLP stacked sequentially. The third architecture (c) is a concatenation of LSTMs and MLPs with Chained Sequential Memory.

The inputs and outputs of each type of unit are:

- MLP. Each layer is essentially an MLP. Therefore each layer is a dense layer fully connected to the previous and following layers.
- Stacked LSTM. Each layer is an LSTM directly connected to a dense MLP. The inputs of each layer are:
  - The output of the previous unit which is the output of its dense layer. DENSE $\left(h_{t-1}\right)$.
  - One sample of the input vector. X_{t-1}

  The output of this unit is the output of the dense layer. DENSE $\left(h_t\right)$.
- LSTM CSM. This unit presents a similar structure as the Stacked LSTM, but the connections between units are more sophisticated. In contrast to the Stacked LSTM, the inputs of this unit are:
  - Previous LSTM layer hidden state and cell state used to initialize the current LSTM layer states. $S_{h_{t-1}}, S_{c_{t-1}}$
  - Output of the previous LSTM layer. h_{t-1}
  - Output of the previous layer as the output of its dense layer. DENSE $\left(h_t\right)$.
  - One sample of the input vector. X_{t-1}

  The outputs of each layer are:
  - LSTM hidden state and cell state used to initialize the next LSTM layer states. S_{h_t}, S_{c_t}
  - Output of the LSTM layer. h_t
  - Output of the dense layer. DENSE $\left(h_t\right)$.

The main parameters of the three architectures proposed are determined in the following section.

7.3. Determining the parameters of the prognosis models

One of the scopes of this article is determining how to implement the methodology proposed for prognosis applications. In order to reach this goal, this study compares three Neural Network architectures characterized by the type of layers of which they are made up. The

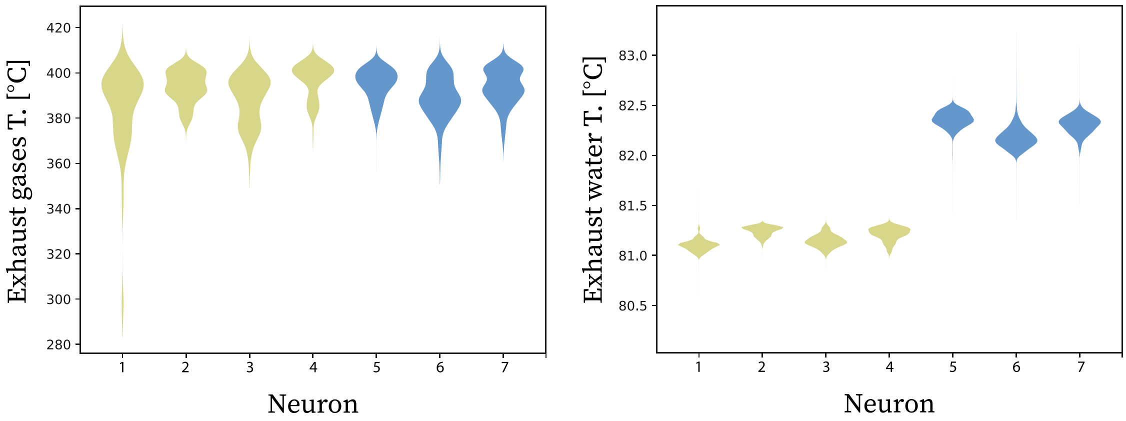


Fig. 10. Examples of Probability Density Distributions of the Reference Behavior Pattern.

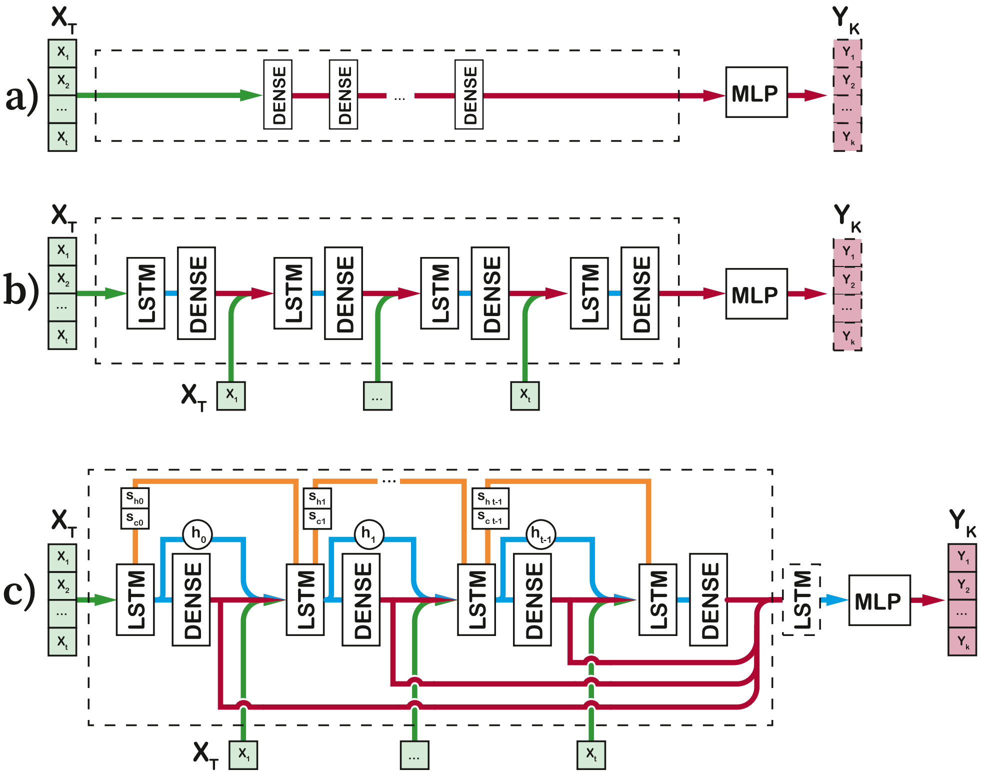


Fig. 11. Architecture of the neural networks proposed for the Prognosis Model.

layers of the Stacked LSTM and CSM LSTM require one sample of the input vector as input. Therefore the number of layers depends on the size of the input vector of the network.

Different sizes of the input vector were studied to determine the optimum size of the input vector. Too small or too big input sizes might lead to underfitting or overfitting effects in the prediction, respectively. Another aspect to take into account is that the size of the input vector also determines the size of the layers of the network. The assessment was carried out by predicting **Deviation** and **Similarity** values of Exhaust gases temperatures as the output of an CSM LSTM architecture with the configuration of Table 3:

Table 3
Network configuration for assessing the optimum size of the input vector size.

Input vector size	X	Test
Output vector size	Y	4 samples
Number of UNITS	X	Test
Hidden neurons LSTM (UNIT)	Z	7 neurons
Hidden neuron DENSE (UNIT)	Z	7 neurons
Hidden neurons LSTM (CONCAT)	Z	7 neurons
Hidden Layers MLP	(20·Z) ; (10·Z) ; Z ; Y	140 ; 70 ; 7 ; 4

The number of hidden neurons was set equal to 7 as a first reasonable approach based on previous studies carried out for the same system (Calvo-Bascones et al., 2020). This analysis only aims to assess how the size of the input vector affects the forecasting capabilities of the model; therefore, in this analysis, the size of the layers remains constant. The size of the layers of the MLP was defined as: first layer, 20 times the size of the LSTM layer; second layer, ten times the size of the LSTM layer; third layer, equal to the size of the LSTM layer; output layer, equal to the size of the output vector.

The prediction errors obtained for an input vector size equal to 4, 6, 12, and 18 are shown in Fig. 12. The training data were split into

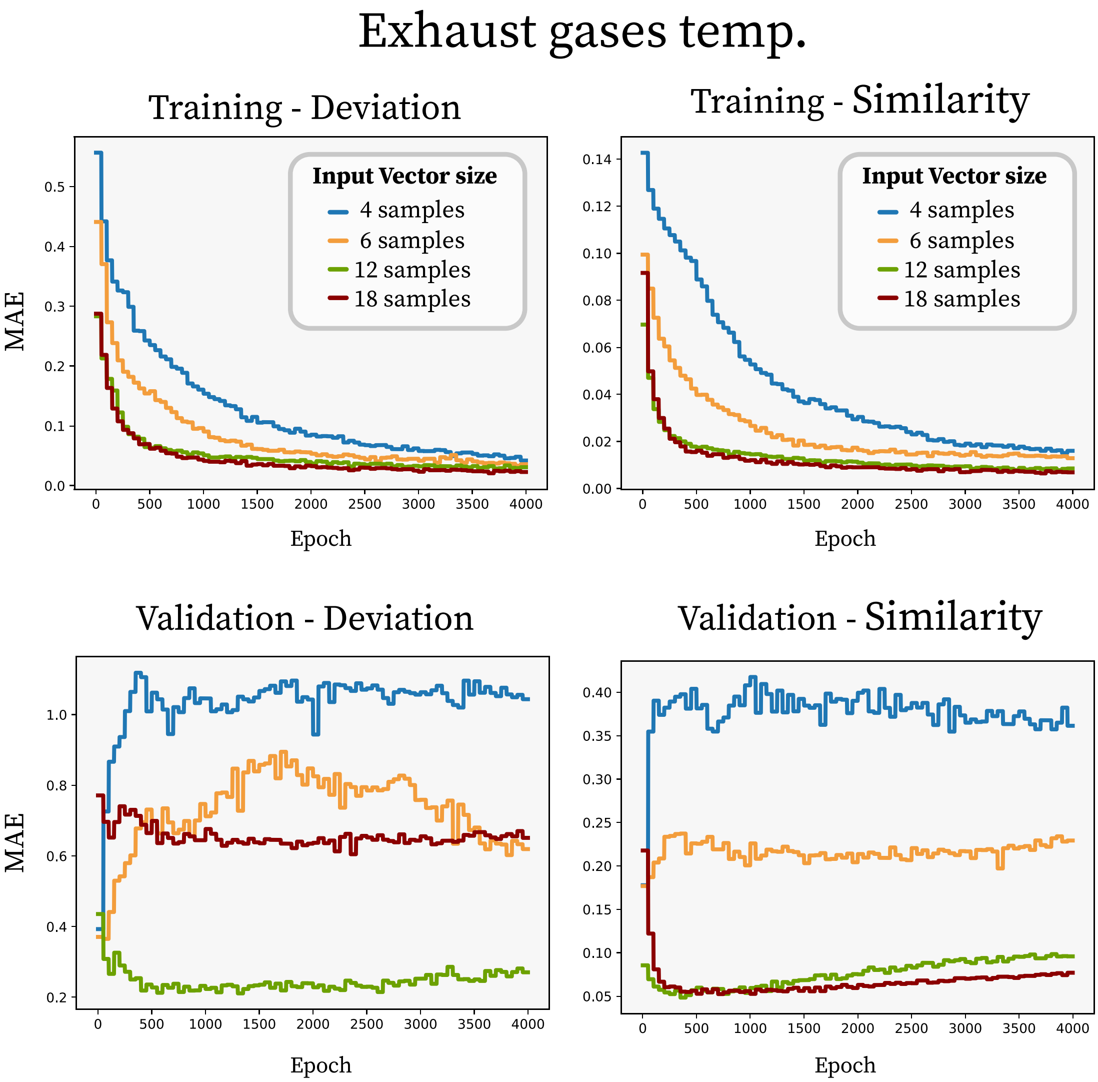


Fig. 12. Analysis of the effects that changing the input size of the model has on the prediction errors.

74 weeks for training and 18 weeks for validation. From the results obtained, the optimum number of samples is set equal to 12. This size is enough to reach a quick convergence and a low training error. Although models with 4, 6, and 18 samples reach low training errors, they present much higher validation errors in the case of **Deviation** and, in the case of **Similarity**, 18 samples show similar validation values to the ones obtained using 12 samples.

The next step is determining the number of hidden neurons. The following analysis aims to compare the accuracy for each one of three variables (sweeping air, exhaust gases, and exhaust water temperatures), taking into account the type of architecture and the size of their layers. The configuration of each architecture is shown in Table 4.

The results of this analysis are shown in Tables 5 and 6. Again, the training data were split into 74 weeks for training and 18 weeks for validation.

Fig. 13 summarizes the best accuracy values obtained for each NN architecture. The values shown correspond to the lowest MAE obtained as the average of training and validation sets according to their optimal number of neurons per layer. This figure shows how the LSTM with CSM obtains the lowest MAE in all the cases, improving the MAE up to 70% (around 50% on average) in comparison to the architecture with the second lowest MAE.

The results obtained with the optimum model for each variable are included in the next section.

Table 4
Configuration of the three architectures to assess how the number of neurons per layer affects the prediction error.

	Architecture MLP	Architecture LSTM stack	Architecture LSTM CSM
Input vector size	12 samples	12 samples	12 samples
Output vector size	4 samples	4 samples	4 samples
Number of UNITS	12 units	12 units	12 units
Hid. neur LSTM(UNIT)		Z neurons (test)	Z neurons (test)
Hid. neur DENSE(UNIT)	Z neurons (test)	Z neurons (test)	Z neurons (test)
Hid. neur LSTM (CONCAT)			Z neurons (test)
Hidden Layers MLP	(20·Z);(10·Z);Z;4	(20·Z);(10·Z);Z;4	(20·Z);(10·Z);Z;4

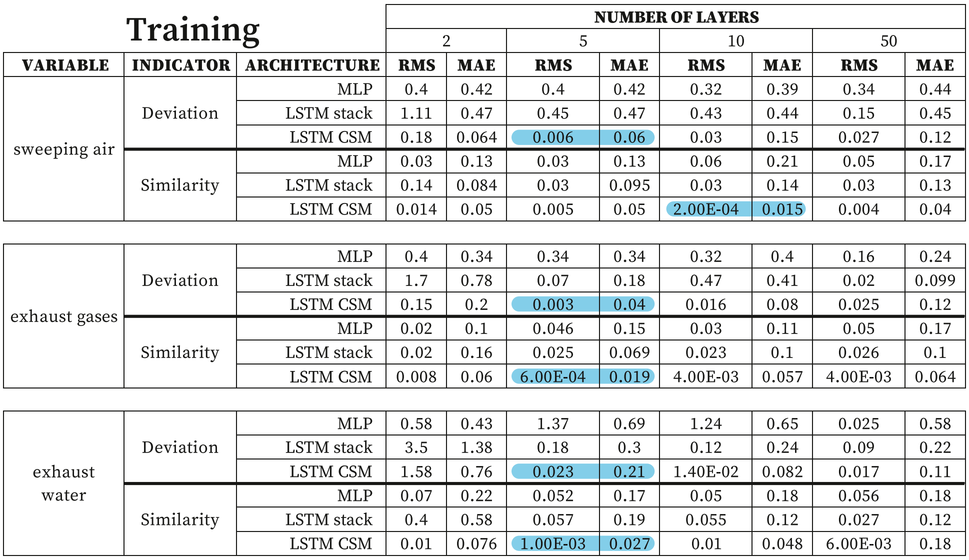
8. Results

The prognosis of **Similarity** and **Deviation** for each of the three variables are shown in Fig. 14.

The results shown correspond to the training set, validation set, and test set. It can be seen in the three variables that there is an initial period in which the predicted values have lower accuracy. This initial period corresponds to the first 12 weeks (from 2016-03 till 2016-06) that the model requires to initialize its CSM.

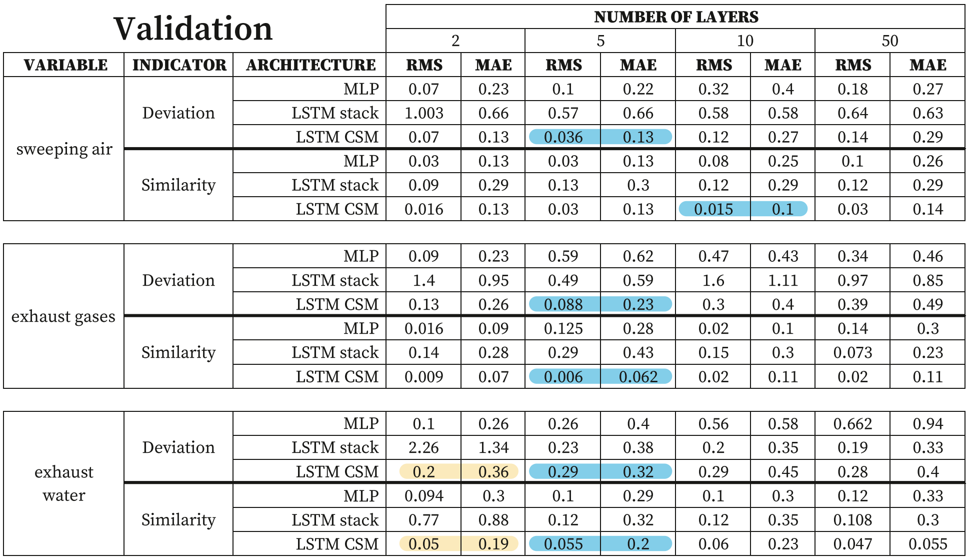
The accuracy of the six models was computed through the Mean Absolute Errors (MAE) and Root Mean Square Errors (RMSE) obtained

Table 5
Errors obtained in the training set for each NN architecture depending on the number of neurons per layer.



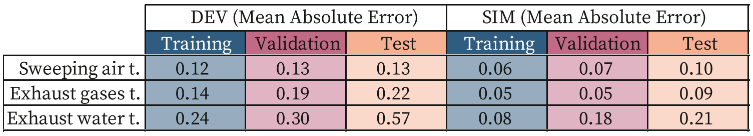
Training			NUMBER OF LAYERS							
			2		5		10		50	
VARIABLE	INDICATOR	ARCHITECTURE	RMS	MAE	RMS	MAE	RMS	MAE	RMS	MAE
sweeping air	Deviation	MLP	0.4	0.42	0.4	0.42	0.32	0.39	0.34	0.44
		LSTM stack	1.11	0.47	0.45	0.47	0.43	0.44	0.15	0.45
		LSTM CSM	0.18	0.064	0.006	0.06	0.03	0.15	0.027	0.12
	Similarity	MLP	0.03	0.13	0.03	0.13	0.06	0.21	0.05	0.17
		LSTM stack	0.14	0.084	0.03	0.095	0.03	0.14	0.03	0.13
		LSTM CSM	0.014	0.05	0.005	0.05	2.00E-04	0.015	0.004	0.04
exhaust gases	Deviation	MLP	0.4	0.34	0.34	0.34	0.32	0.4	0.16	0.24
		LSTM stack	1.7	0.78	0.07	0.18	0.47	0.41	0.02	0.099
		LSTM CSM	0.15	0.2	0.003	0.04	0.016	0.08	0.025	0.12
	Similarity	MLP	0.02	0.1	0.046	0.15	0.03	0.11	0.05	0.17
		LSTM stack	0.02	0.16	0.025	0.069	0.023	0.1	0.026	0.1
		LSTM CSM	0.008	0.06	6.00E-04	0.019	4.00E-03	0.057	4.00E-03	0.064
exhaust water	Deviation	MLP	0.58	0.43	1.37	0.69	1.24	0.65	0.025	0.58
		LSTM stack	3.5	1.38	0.18	0.3	0.12	0.24	0.09	0.22
		LSTM CSM	1.58	0.76	0.023	0.21	1.40E-02	0.082	0.017	0.11
	Similarity	MLP	0.07	0.22	0.052	0.17	0.05	0.18	0.056	0.18
		LSTM stack	0.4	0.58	0.057	0.19	0.055	0.12	0.027	0.12
		LSTM CSM	0.01	0.076	1.00E-03	0.027	0.01	0.048	6.00E-03	0.18

Table 6
Errors obtained in the validation set for each NN architecture depending on the number of neurons per layer.



Validation			NUMBER OF LAYERS							
			2		5		10		50	
VARIABLE	INDICATOR	ARCHITECTURE	RMS	MAE	RMS	MAE	RMS	MAE	RMS	MAE
sweeping air	Deviation	MLP	0.07	0.23	0.1	0.22	0.32	0.4	0.18	0.27
		LSTM stack	1.003	0.66	0.57	0.66	0.58	0.58	0.64	0.63
		LSTM CSM	0.07	0.13	0.036	0.13	0.12	0.27	0.14	0.29
	Similarity	MLP	0.03	0.13	0.03	0.13	0.08	0.25	0.1	0.26
		LSTM stack	0.09	0.29	0.13	0.3	0.12	0.29	0.12	0.29
		LSTM CSM	0.016	0.13	0.03	0.13	0.015	0.1	0.03	0.14
exhaust gases	Deviation	MLP	0.09	0.23	0.59	0.62	0.47	0.43	0.34	0.46
		LSTM stack	1.4	0.95	0.49	0.59	1.6	1.11	0.97	0.85
		LSTM CSM	0.13	0.26	0.088	0.23	0.3	0.4	0.39	0.49
	Similarity	MLP	0.016	0.09	0.125	0.28	0.02	0.1	0.14	0.3
		LSTM stack	0.14	0.28	0.29	0.43	0.15	0.3	0.073	0.23
		LSTM CSM	0.009	0.07	0.006	0.062	0.02	0.11	0.02	0.11
exhaust water	Deviation	MLP	0.1	0.26	0.26	0.4	0.56	0.58	0.662	0.94
		LSTM stack	2.26	1.34	0.23	0.38	0.2	0.35	0.19	0.33
		LSTM CSM	0.2	0.36	0.29	0.32	0.29	0.45	0.28	0.4
	Similarity	MLP	0.094	0.3	0.1	0.29	0.1	0.3	0.12	0.33
		LSTM stack	0.77	0.88	0.12	0.32	0.12	0.35	0.108	0.3
		LSTM CSM	0.05	0.19	0.055	0.2	0.06	0.23	0.047	0.055

Table 7
Accuracy of the models measured through the Mean Absolute Errors obtained for each variable and indicator.



	DEV (Mean Absolute Error)			SIM (Mean Absolute Error)		
	Training	Validation	Test	Training	Validation	Test
Sweeping air t.	0.12	0.13	0.13	0.06	0.07	0.10
Exhaust gases t.	0.14	0.19	0.22	0.05	0.05	0.09
Exhaust water t.	0.24	0.30	0.57	0.08	0.18	0.21

by comparing forecast and real values, see Table 7. Two deviations are detected, both in the exhaust section of the system. The exhaust gas temperature shows an MAE (Test) around 1.6 times its expected MAE (Training), and in the case of the exhaust water temperature, this increment reaches 2.4 times its expected MAE. These variations in the MAE reflect the existence of alterations in the behavior regarding the previous one observed within the training set.

Deviation values above 2.5 and **Similarity** values below 0.4 are considered anomalous conditions that require corrective measures to correct the behavior of the system. In most cases, anomalous behaviors are detected earlier through the **Similarity** indicator than through the **Deviation** indicator. Therefore, **Deviation** will be used for anomaly detection and **Similarity** for making behavioral predictions.

Three main applications are achieved from the method proposed: The first application is the assessment of maximum effectiveness achieved after each maintenance task. Maximum effectiveness refers to the maximum similarity value achieved after a maintenance task was carried out. This maximum value can be used as an indicator

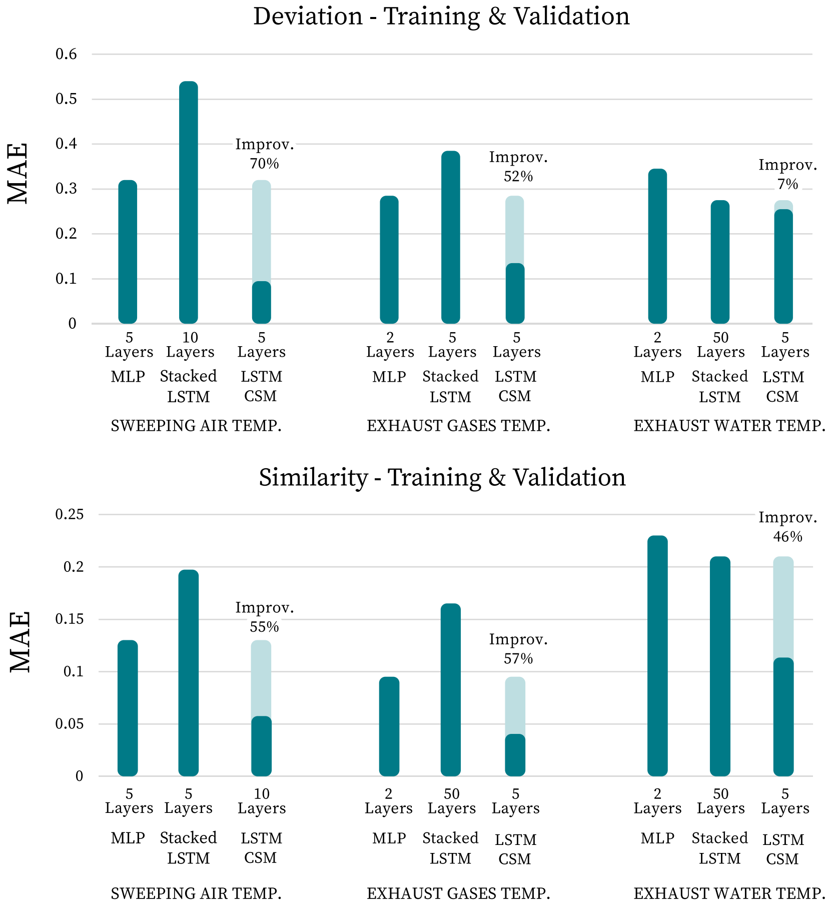


Fig. 13. Summary of the best average MAE obtained for each NN architecture with their optimal configuration.

to assess the effectiveness of a maintenance task. Once the maximum effectiveness is achieved, the evolution of the behavior can be modeled through a parametric RUL model.

The second one is the composition of a straightforward RUL model based on parametric equations allowing making predictions further than one week ahead. Thanks to the predictive model, the evolution of the behavior is simplified, easing the characterization of a behavior through a parametric model. A basic parametric model can be formulated as:

$$S'(t) = S_o - (t/T)^{1/\lambda} \quad (2)$$

where S' is the predicted similarity value, t is the number of samples (time units) after the maximum **Similarity** is reached, S_o is the value of the maximum **Similarity** reached after a maintenance task, T is the estimated number of samples that a variable with a **Similarity** value equal to 1 reaches a value equal to 0 and λ is the speed of loss of **Similarity**.

The third application focuses on raising a forewarning event when a maintenance task is required. When the predicted similarity value drops down below 0.4, a maintenance task has to be planned to avoid potential abrupt failures.

These three applications are shown in Fig. 15. The results obtained show how these three applications can be useful in the prognosis and RUL assessment of the behavior of the system.

The parametric curves of the RUL model aim to show how the behavior of a variable can be easily estimated through a simplified parametric model. In the case of the Exhaust water temperature, the first period hides a prompt deviation of the model that becomes visible during the second period. At the end of this second period, it can be seen how the predicted similarity values and real values tally. The parametric model determines from an initial state the RUL of the component. In this example, it determines when the similarity values drop from a threshold value, in this case: 0.4.

9. Conclusions

The use of behavior patterns is a powerful approach in the study of the behavior of a component. A behavior pattern is a set of clusters with the main features that define a behavior. Two indicators are used for this purpose: **Similarity** and **Deviation**. This study aims to complement an initial methodology based on anomaly detection, including a prognosis model based on the values registered from the two indicators previously mentioned. The methodology proposed aims to develop a prognosis model choosing the best model obtained from three neural network architectures. The three neural networks proposed range from a basic multilayer perceptron, a Stacked LSTM, and an LSTM with Chained Sequential Memory cells. The proposed CSM LSTM architecture presents an outstanding improvement in the forecasting capabilities of both indicators with a reduction of prediction errors up to 70% (around 50% on average) in comparison to the lowest error obtained achieved with MLP and Stacked LSTM NNs The prediction capabilities of each neural network were studied thoroughly in a real case study of a diesel generator of a power plant, determining which model achieved higher accuracy in their predictions of the two behavioral indicators. The proposed case study of a diesel generator of big power shows how the present prognosis methodology success in the characterization and forecast of degradation processes allows for better maintenance planning and optimized management of critical industrial assets. In addition to the forecast of both behavioral indicators, the results obtained from the predictive model can be used to configure

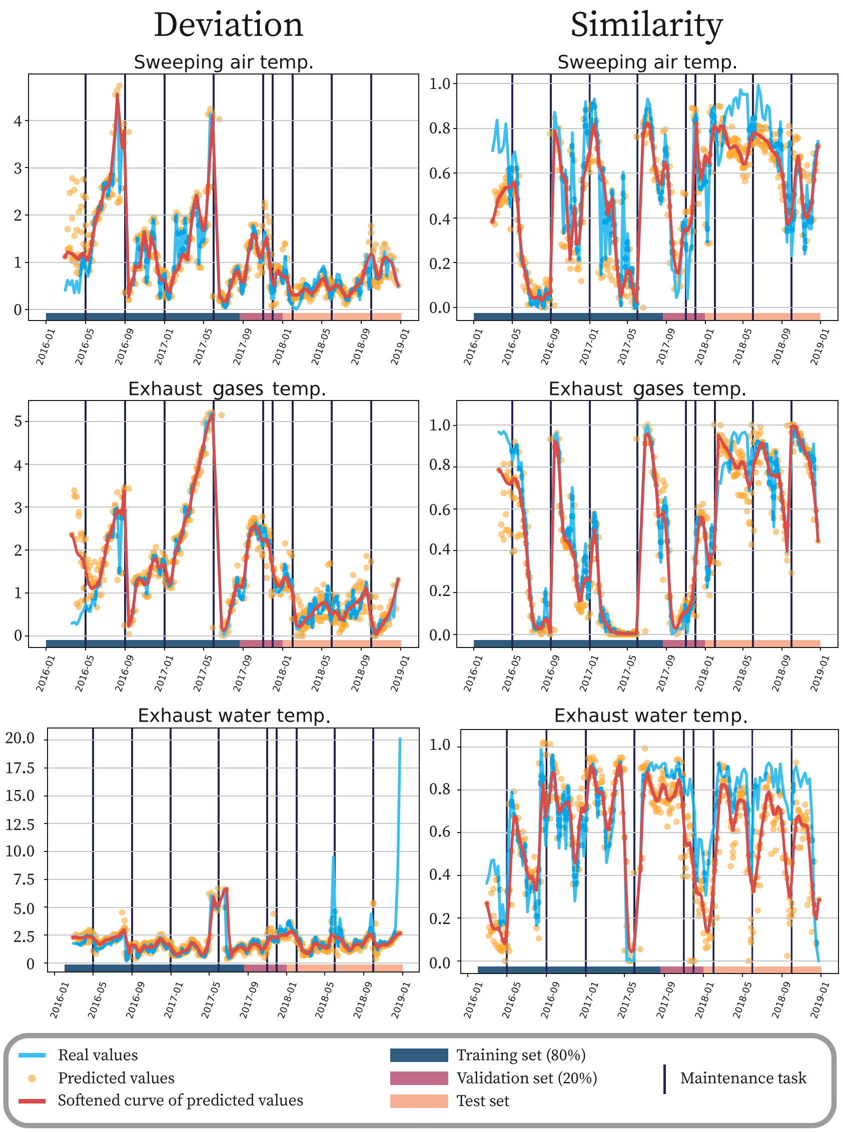


Fig. 14. Forecast values obtained for the different variables of the model.

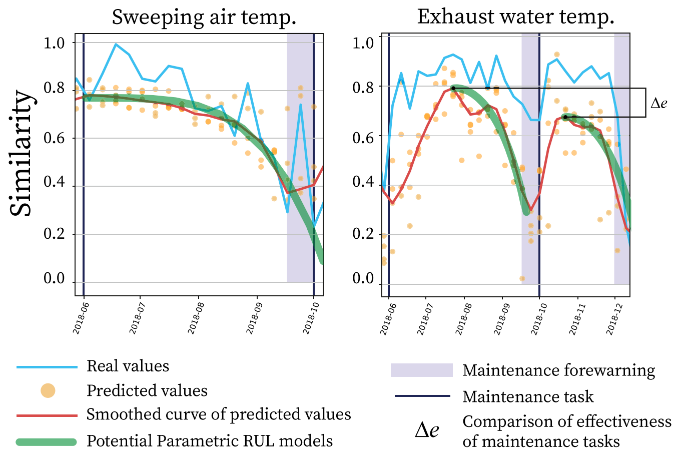


Fig. 15. Potential parametric models for assessing the RUL of the system and the effectiveness of maintenance tasks.

an additional parametric model used to carry out additional Remaining Useful Life assessments.

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.

CRediT authorship contribution statement

Pablo Calvo-Bascones: Conceptualization, Methodology, Software, Writing – original draft. **Miguel A. Sanz-Bobi:** Review, Data curation, Supervision, Funding acquisition.

References

Abid, A., Khan, M.T., Khan, M.S., 2020. Multidomain features-based GA optimized artificial immune system for bearing fault detection. IEEE Trans. Syst. Man Cybern. Syst. 50 (1), 348–359.

Alaswad, S., Xiang, Y., 2017. A review on condition-based maintenance optimization models for stochastically deteriorating system. Reliab. Eng. Syst. Saf. 157, 54–63.

Arunthavanathan, R., Khan, F., Ahmed, S., Imtiaz, S., 2021. A deep learning model for process fault prognosis. Process Saf. Environ. Prot. 154, 467–479.

Calvo-Bascones, P., Sanz-Bobi, M., Álvarez Tejedor, T., 2020. Method for condition characterization of industrial components by dynamic discovering of their pattern behaviour. In: 30th/15th European Safety and Reliability Conference and Probabilistic Safety Assessment and Management Conference - ESREL 2020 PSAM 15, Venecia, Italia, 01-05 Noviembre 2020. pp. 1–8.

Calvo-Bascones, P., Sanz-Bobi, M.A., Welte, T.M., 2021. Anomaly detection method based on the deep knowledge behind behavior patterns in industrial components. application to a hydropower plant. Comput. Ind. 125, 103376.

Cheng, Y., Wu, J., Zhu, H., Or, S.W., Shao, X., 2021. Remaining useful life prognosis based on ensemble long short-term memory neural network. IEEE Trans. Instrum. Meas. 70, 1–12.

Dhahad, H.A., Hasan, A.M., Chaichan, M.T., Kazem, H.A., 2022. Prognostic of diesel engine emissions and performance based on an intelligent technique for nanoparticle additives. Energy 238, 121855.

Gil, A., Sanz-Bobi, M.A., Rodríguez-López, M.A., 2018. Behavior anomaly indicators based on reference patterns—Application to the gearbox and electrical generator of a wind turbine. Energies 11 (1), 87.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Hou, L., Zou, J., Du, C., Zhang, J., 2020. A fault diagnosis model of marine diesel engine cylinder based on modified genetic algorithm and multilayer perceptron. Soft Comput. 24 (10), 7603–7613.

Jalali, S.M.J., Ahmadian, S., Kavousi-Fard, A., Khosravi, A., Nahavandi, S., 2021. Automated deep CNN-LSTM architecture design for solar irradiance forecasting. IEEE Trans. Syst. Man Cybern. Syst. 1–12.

Kayaalp, K., Metlek, S., Ekici, S., Şöhret, Y., 2021. Developing a model for prediction of the combustion performance and emissions of a turboprop engine using the long short-term memory method. Fuel 302, 121202.

Khan, S., Yairi, T., 2018. A review on the application of deep learning in system health management. Mech. Syst. Signal Process. 107, 241–265.

Kodinariya, T., Makwana, P., 2013. Review on determining of cluster in K-means clustering. Int. J. Adv. Res. Comput. Sci. Manage. Stud. 1, 90–95.

Kohonen, T., 1982. Self-organized formation of topologically correct feature maps. Biol. Cybernet. 43 (1), 59–69.

Mejia, J., Avelar-Sosa, L., Mederos, B., Ramírez, E.S., Díaz Roman, J.D., 2021. Prediction of time series using an analysis filter bank of LSTM units. Comput. Ind. Eng. 157, 107371.

Sayah, M., Guebli, D., Al Masry, Z., Zerhouni, N., 2020. Robustness testing framework for RUL prediction deep LSTM networks. ISA Trans. S0019057820302767.

Stetter, R., 2020. Prediction of the remaining useful life for components of automated processes. In: Stetter, R. (Ed.), Fault-Tolerant Design and Control of Automated Vehicles and Processes: Insights for the Synthesis of Intelligent Systems. In: Studies in Systems, Decision and Control, Springer International Publishing, pp. 135–175.

Su, X., Liu, H., Tao, L., Lu, C., Suo, M., 2021. An end-to-end framework for remaining useful life prediction of rolling bearing based on feature pre-extraction mechanism and deep adaptive transformer model. Comput. Ind. Eng. 161, 107531.

Van Laerhoven, K., 2001. Combining the self-organizing map and K-means clustering for on-line classification of sensor data. In: Dorffner, G., Bischof, H., Hornik, K. (Eds.), Artificial Neural Networks — ICANN 2001. In: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 464–469.

Venkatasubramanian, V., 2005. Prognostic and diagnostic monitoring of complex systems for product lifecycle management: challenges and opportunities. Comput. Chem. Eng. 29 (6), 1253–1263.

Venkatesan, S., Manickavasagam, K., Tengenkai, N., Vijayalakshmi, N., 2019. Health monitoring and prognosis of electric vehicle motor using intelligent-digital twin. IET Electr. Power Appl. 13 (9), 1328–1335.

Wang, H., Liu, X., Ma, L., Zhang, Y., 2021. Anomaly detection for hydropower turbine unit based on variational modal decomposition and deep autoencoder. 2021 International Conference on Energy Engineering and Power Systems, Energy Rep. 2021 International Conference on Energy Engineering and Power Systems, 7.938–946,

Xu, X., Yan, X., Sheng, C., Yuan, C., Xu, D., Yang, J., 2020. A belief rule-based expert system for fault diagnosis of marine diesel engines. IEEE Trans. Syst. Man Cybern. Syst. 50 (2), 656–672.

Yu, Y., Si, X., Hu, C., Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. 31 (7), 1235–1270.

Zhang, Y., Dong, Z.Y., Kong, W., Meng, K., 2020. A composite anomaly detection system for data-driven power plant condition monitoring. IEEE Trans. Ind. Inf. 16 (7), 4390–4402.

Zhang, W., Yang, D., Wang, H., 2019. Data-driven methods for predictive maintenance of industrial equipment: A survey. IEEE Syst. J. 13 (3), 2213–2227.

Zhao, B., 2021. Deep multi-scale separable convolutional network with triple attention mechanism: A novel multi-task domain adaptation method for intelligent fault diagnosis. Expert Syst. Appl. 14.

Zonta, T., da Costa, C.A., da Rosa Righi, R., de Lima, M.J., da Trindade, E.S., Li, G.P., 2020. Predictive maintenance in the industry 4.0: A systematic literature review. Comput. Ind. Eng. 150, 106889.