



*Review*

## **Machine fault detection methods based on machine learning algorithms:**

### **A review**

**Giuseppe Ciaburro\***

Department of Architecture and Industrial Design, Università degli Studi della Campania Luigi Vanvitelli, Borgo San Lorenzo - 81031 Aversa (Ce), Italy

\* **Correspondence:** Email: [giuseppe.ciaburro@unicampania.it](mailto:giuseppe.ciaburro@unicampania.it); Tel: +390818122530; Fax: +390818122530.

**Abstract:** Preventive identification of mechanical parts failures has always played a crucial role in machine maintenance. Over time, as the processing cycles are repeated, the machinery in the production system is subject to wear with a consequent loss of technical efficiency compared to optimal conditions. These conditions can, in some cases, lead to the breakage of the elements with consequent stoppage of the production process pending the replacement of the element. This situation entails a large loss of turnover on the part of the company. For this reason, it is crucial to be able to predict failures in advance to try to replace the element before its wear can cause a reduction in machine performance. Several systems have recently been developed for the preventive faults detection that use a combination of low-cost sensors and algorithms based on machine learning. In this work the different methodologies for the identification of the most common mechanical failures are examined and the most widely applied algorithms based on machine learning are analyzed: Support Vector Machine (SVM) solutions, Artificial Neural Network (ANN) algorithms, Convolutional Neural Network (CNN) model, Recurrent Neural Network (RNN) applications, and Deep Generative Systems. These topics have been described in detail and the works most appreciated by the scientific community have been reviewed to highlight the strengths in identifying faults and to outline the directions for future challenges.

**Keywords:** machine fault diagnosis; machine learning; deep learning; classification; regression; review

---

## 1. Introduction

Maintenance and related activities have always played a role of primary importance within a production context. Over time, as the processing cycles are repeated, the machinery in the production system is subject to wear with a consequent loss of technical efficiency compared to optimal conditions [1,2].

Maintenance, therefore, is of crucial importance in the industrial context, both to guarantee the continuity of processes and to ensure the safety of operators. This means ensuring maximum reliability and availability of the systems, supporting the minimum cost, and planning the necessary activities of both a technical and organizational nature, through the practical execution of the interventions. A maintenance policy is therefore essential for any production plant and, if implemented appropriately, can lead to the achievement of various objectives [3,4].

Among these, it should be remembered that effective maintenance gives us an increase in plant productivity through the drastic reduction of machine downtime. Furthermore, a minimization of the costs related to the maintenance of the plant is achieved through an effective identification of the spare parts, determining the correct management of the warehouse. Finally, as already mentioned, effective maintenance makes us safe from any accidents at work, guaranteeing the necessary safety of the operators working on the production line [5].

Recently, industrial automation processes have seen an ever-increasing use of IIoT (Industrial Internet of Things) technologies [6]. With this acronym (IIoT) we refer to the connection between smart objects and smart grids. Smart objects can perform a series of activities - identification, localization, status diagnosis, data acquisition, processing, implementation, and communication - while smart grids are open, standard, and multifunctional [7]. This new way of conceiving the industrial process is part of the so-called Industry 4.0 according to which digital technologies such as IoT (Internet of Things) devices, but also sensors, cloud, machine learning, collaborative robotics, 3D printing, can increase the efficiency and the value of production by stimulating interconnection and cooperation between all resources [8]. Through their connection to the plants and instruments present in the supply chain, these technologies give the opportunity to process a huge amount of data in real time, contributing to the process's optimization, reducing waste of resources and errors, increasing business competitiveness [9]. The opportunities provided by these tools have led to the renewal and evolution of the industrial maintenance approach in many companies, which has taken on an increasingly complex and central role in the production context. In this way, we witnessed the transition from a preventive maintenance policy to a predictive one. The preventive approach is often disconnected from a temporal point of view to the actual conditions of the production plants, as it is unable to provide efficient management of the interventions, lacking a historical memory that is the basis of the decisions. Conversely, with a predictive approach, supported by the necessary sensors and data analysis methodologies, it can have a lot of constantly updated information, which acts as a support to optimize the times and resources [10].

In this innovative context that has made available a large amount of data, the need to adopt data analysis methodologies that allow us to extract knowledge has become evident. When we deal with huge volumes of heterogeneous data, it is not easy to identify characteristics that can make the phenomenon we are observing more explanatory, at least it is not so for the human eye. In this regard, modern technologies based on artificial intelligence come to the rescue, which use algorithms based on machine learning to extract knowledge from data [11]. The information underlying the knowledge

is extracted from the data which are explored and analyzed with techniques called Data Mining in search of recurring patterns or to discover hidden causal associations or relationships [12]. Machine learning overturns the traditional paradigm that draws an output from the input data and uses an algorithm that explains how to use it. In new systems, however, knowledge is an inductive process: the input is the data and, possibly, a first example of the expected output, so it will be the machine that will learn the algorithm to follow to obtain the same result [13].

For the extraction of patterns, it is necessary to follow a process divided into various phases, called the knowledge discovery process, which initially starts with the selection of the data to be analyzed from the various available sources, such as data taken from databases [14]. Subsequently, a phase of fundamental importance for the final accuracy of the extracted patterns is passed, the preprocessing phase. In fact, this phase allows to improve the quality of the data, for example, by resolving eventually inconsistencies, by removing any anomalous data not useful for the analysis, so called because they differ from the rest of the data within the dataset, and by solving any conflicts present between the data after the integration of the different sources. This pre-processing phase is followed by a transformation phase that allows us to prepare the data for the next phase of Data Mining, thanks to which, as previously mentioned, it is possible to extract the patterns from the data by applying specific algorithms. Finally, after a phase of interpretation and evaluation of the extracted patterns, we move on to the final phase in which the final data of the analyzes are presented [15].

In this work we will analyze and describe the most common methods based on Machine Learning for machine fault diagnosis. The paper is structured as follows: Section 2 describes in detail the most common failures affecting machines, analyzing their characteristics and peculiarities that make mathematical modeling extremely complex. Section 3 analyzes the most popular machine learning-based fault diagnosis methods. Finally, Section 4 summarizes the results obtained in applying these methods to real cases, highlighting their merits, and listing their limits.

## **2. Maintenance and fault diagnostics**

### *2.1. Types of maintenance*

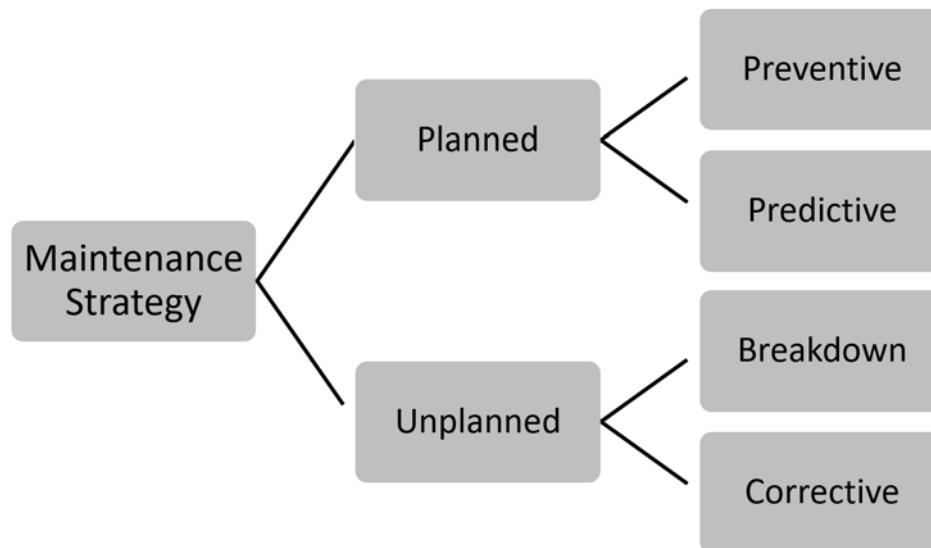
The purpose of a maintenance process is to preserve and keep industrial machinery in a state of full efficiency. But this definition must not diminish the role of an effective maintenance activity which cannot be understood as a simple corrective action. Maintenance is not just the activity that is carried out in the event of a fault that leads to the blocking of production in the event of a fault [16].

There are several approaches that can be adopted in the management of a maintenance process (Figure 1): Corrective maintenance is generated in response to an event whose effect is to prevent, at different levels of severity, the continuation of an activity, or the interruption of a service, or the degradation of the service itself which therefore can no longer be provided with an adequate level of safety or efficiency [17].

To prevent corrective maintenance activities, other maintenance methods can be adopted, aimed at a programming-based system, which tries to avoid the occurrence of the fault, foreseeing and correcting it before it occurs. These scheduled interventions are the basis of preventive maintenance, which offers the advantage of limiting the total time of the intervention which results in a system block [18]. Through experience and the improvement of maintenance activities, a more efficient way to carry out maintenance has been defined. This is how condition-based maintenance is defined, which

represents an advance over preventative maintenance [19].

This improvement is found in the useful life of the component: The concept underlying condition-based maintenance is precisely that of being able to fully exploit the potential of the component [20]. To adopt this methodology, it is necessary to identify warning symptoms.



**Figure 1.** Type of maintenance: In this scheme the different maintenance strategies are classified, distinguishing between those that are planned and those that are instead linked to events.

The warning symptoms are signs that usually the machine or system communicates before the actual failure occurs and can be visual, audible, or even olfactory. These signals are recognizable because they are not part of the normal activity of the machinery, they are completely extraneous signals, and which often cause alarm. The detection of an anomaly condition occurs when some physical parameter of the machine is not compliant with normal operation. Typical examples are the increase in noise, vibrations, or temperature. It can be implemented either by man, for example by maintenance technicians with appropriate inspections or by expert operators who notice changes during use, or by means of special sensors, which continuously monitor the system parameters [21].

Predictive maintenance is a further specialization of condition-based maintenance. It is performed following a prediction derived from repeated analysis or from known characteristics and from the display of significant parameters relating to the degradation of the component [22]. According to this approach, the data relating to the functioning and conditions of the various components are recorded and saved in a history, to be then used to build a trend of the overall behavior. The information thus obtained is exploited to predict the evolution of the degradation level, and then plan a related maintenance activity [23]. The main advantage with respect to condition maintenance lies precisely in the part of analyzing the trend and building an evolution model of the state based on the experience deduced from the past analysis, which allows to be able to estimate the residual useful lifetime of the component after having detected a deviation from normal operation, when it is still in its first phase. An effective predictive system greatly improves and optimizes the availability of machinery and the time spent in production, reducing the number of maintenance interventions and their cost [24].



Furthermore, it also has positive effects on the quality of the product. By connecting sensors to a machine, we can detect its operating parameters, such as the measurement of vibrations or the operating temperature, and thus profile its activity in optimal conditions. A variation in these parameters will indicate the increase in the degradation of the machinery components: Using appropriate mathematical models it will be possible to predict the time of failure. Often, however, these variations are made explicit to human sensory capacities too late, when the damage has already been done, making the ability to predict the failure useless. In this sense, the algorithms based on Machine Learning can help us which, on the other hand, are able to identify anomalies long before they become perceptible to humans [25].

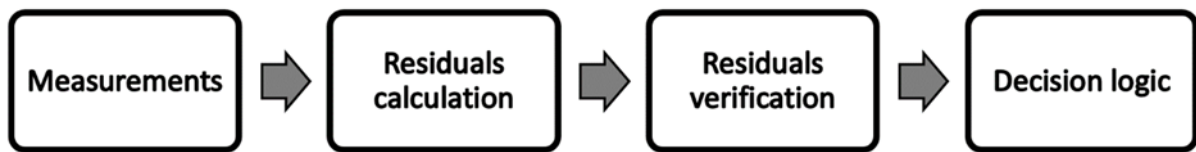
## 2.2. Fault diagnostics

A physical system, in its life cycle, can be subject to failures or malfunctions that can compromise its normal operation. It is therefore necessary to introduce a fault diagnosis system within a plant capable of preventing critical interruptions: This is called a fault diagnosis system and can identify the possible presence of a malfunction within the monitored system [26]. The search for the fault is one of the most important and qualifying maintenance intervention phases and it is necessary to act in a systematic and deterministic way. To carry out a complete search for the fault, it is necessary to analyze all the possible causes that may have determined it.

Failure is the condition of non-preservation of the desired state of operation, or the cessation of an entity's ability to perform the required function. It is deduced that the failure is an event, a passage from one state of good operation to another that does not respect the expected performance: The failure is a state, a stationary situation, in which the entity is unable to operate. A failure can occur due to a breakage of a component, involving the hardware structure of the object, or due to a software error, or a human error. A system can always have a functional defect when, while working regularly, it is called to do something for which it was not designed or is exposed to transitory conditions that cause the momentary failure [27].

Depending on the technology involved in the event, the failures of a machine can be divided into mechanical, electrical, and IT. Mechanical failures generally involve breaking or permanent deformation of mechanical parts. The causes of mechanical failures can be many, among the most frequent are the following: corrosion, material fatigue, thermal shock, external mechanical loads higher than those foreseen by the designers. Electrical failures generally involve insulation failure and can be caused by overcurrent, overvoltage, unsuitable environmental conditions. Finally, in the computer electronic field, failures can concern both hardware and software during the execution of a program. Furthermore, faults can be permanent if once they appear they persist over time, not permanent if they occur in an unstable and repeated manner over time, and transients if they appear only in conjunction with particular and temporary environmental conditions. A fundamental distinction concerns the nature of the failure: In systematic failures there is a correlation of a deterministic nature to a certain cause. In other words, a precise origin can be identified for systematic failures. A failure of this type is usually caused by human errors in the design, production, installation, or from incorrect use. This type of error can only be eliminated by changing the design or production process or conditions of use. Non-systematic failures, on the other hand, also occur if a component or system has been correctly designed, built, and is correctly used in accordance with the manufacturer's specifications [28]. To search for the fault, we start with the identification of the symptom of malfunction and continue with the search for

the cause that generated it (Figure 2).



**Figure 2.** Block diagram of a typical fault detection and isolation procedure.

This occurs with a process of progressive elimination of the possible causes, until the one that caused the problem is identified. This way of proceeding requires that the maintenance technician know what the possible failures of the equipment are and therefore can verify the relationship between the failure and the cause that produced it. Upstream of the process there is therefore a specific knowledge of the device that requires the observation of its operation for an adequate time through the measurement procedures of the characteristic parameters [29].

### 3. Machine Learning-based methods for machine fault diagnosis

Classical methods based on an approach that uses models or prior knowledge of the phenomenon are suitable for general supervision of processes. However, the situation becomes more complex if the process changes rapidly, as in the case of dynamic systems.

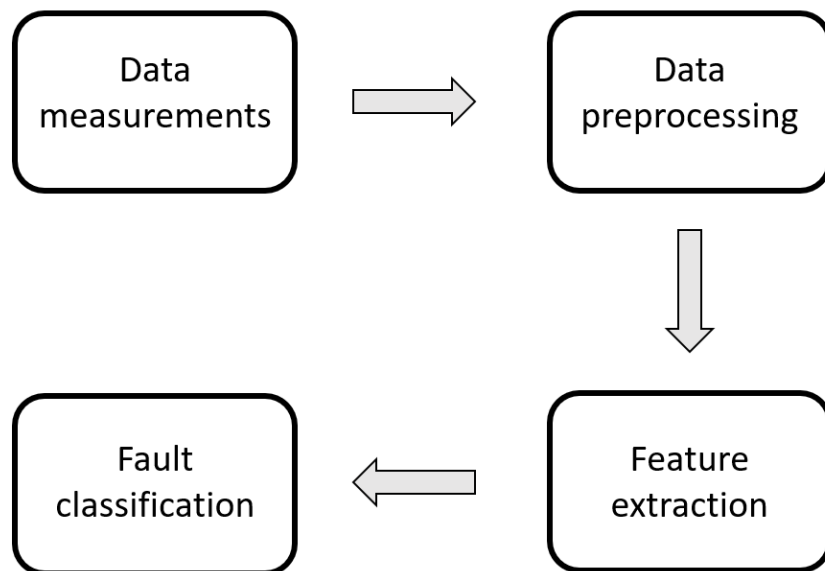
Also, in the presence of closed loops, changes in the process are covered by control actions and cannot be detected by the output signals if the manipulated process inputs remain in the normal range. Therefore, feedback systems hinder the early detection of process errors. Advanced methods of supervision and fault diagnosis are therefore required, which ensure, for example, the timely detection of small faults with sudden temporal behavior, as well as the diagnosis of faults in the actuator, process components or sensors [30].

The methods based on Machine Learning provide for the identification and selection of functions and the classification of faults, allow a systematic approach to Fault Diagnosis, and can be used in automated and unattended environments (Figure 3). These types of solutions are increasingly used in many industrial sectors to maximize equipment uptime and minimize both maintenance and operating costs. But the types of algorithms available are many and varied, in the following subsections we will introduce the technologies most widely used by the scientific community to tackle different types of applications [31].

#### 3.1. Support Vector Machine (SVM) solutions

SVMs solve the learning problem starting from a training set of experimental data whose characteristic parameters are known. The goal will therefore be to build a system that learns from already correctly classified data and that thanks to them is able to build a classification function capable

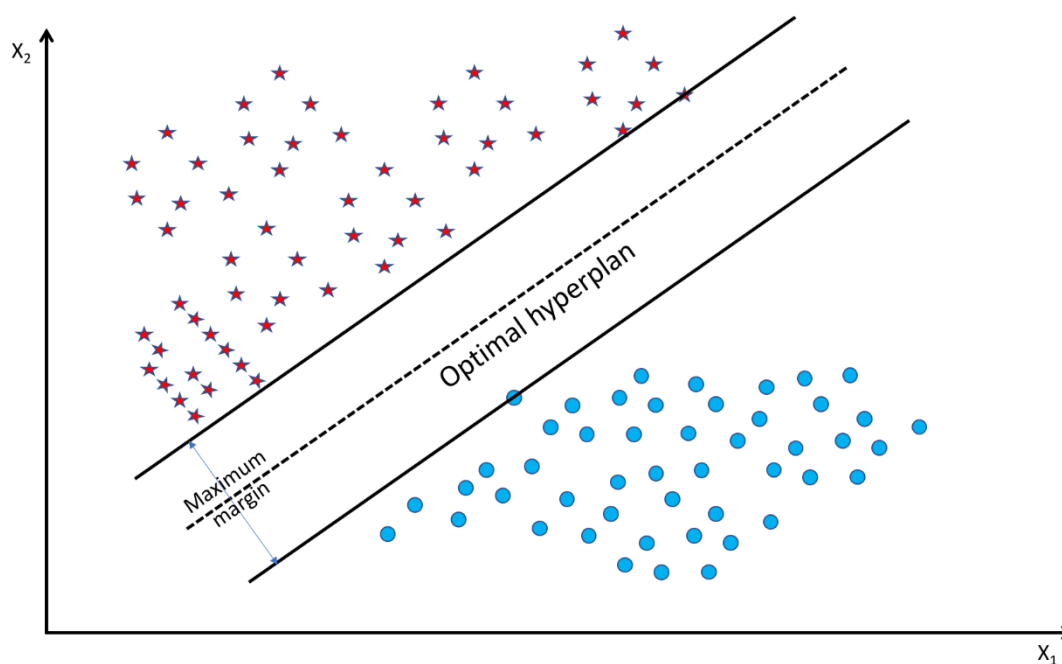
of cataloging data even outside this set [32]. The main characteristic of SVMs is that they allow to reach high performances in practical applications based on simple ideas: they are rather simple to analyze mathematically but they allow to analyze complex models. The algorithm that allows to train them can be traced back to a quadratic programming problem with linear constraints: It finds application in very different fields, among which the most common are the recognition of patterns, the cataloging of texts and the identification of faces in images [33].



**Figure 3.** Scheme of a typical fault diagnosis process.

Classification, understood as the assignment of a pattern to a specific class already known a priori, is a topic of extraordinary importance for solving real-world problems. It can be used in very different fields, even for problems that at first sight might seem of another kind. The first case to be analyzed is the one in which the training set samples are linearly separable [34]. The method to be used to solve the problem will be to determine a hyperplane that separates the data classes, so that the points can be divided between two half-spaces. In most practical cases it must also be considered that there may be errors in the experimental data. If there are points that result in the wrong half plane, in fact, an exact linear classification is almost impossible and approximation techniques are needed that try to minimize the number of errors. To use the classification through hyperplanes, even for data that would need to separate non-linear functions, we can apply the technique of the feature spaces (Figure 4). This method, which is the basis of the SVM theory, consists in mapping the initial data in a space of higher dimension: The data is mapped in a space in which it becomes linearly separable and in which it will be possible to find a hyperplane that separates them [35]. To do this, the input data is multiplied scalarly between them, and to make this calculation simple, which in large spaces becomes very complicated, a function called kernel is used that directly returns the scalar product of the images. To make it possible to generalize the problem, even in the non-linear case, in which the kernel functions will be used, a Lagrangian formulation is required [36], thanks to which the data will appear only in the form of a scalar product.

Widodo et al. [37] used SVMs for monitoring and diagnosing machine failures. The authors made a comparison between this technology and those based on other algorithms that exploit Machine Learning, showing that SVMs return high precision on the generalization of the problem. Fei et al. [38] used SVMs for power transformer fault diagnosis. The authors used information on the content of the characteristic gases of the transformer, then identified the optimal parameters of the classifier using a genetic algorithm, and finally used SVMs to solve the problem with small samplings, non-linearities and large sizes.



**Figure 4.** Support Vector Machine (SVM) based classification.

Wu et al. [39] applied SVMs for bearing failure diagnosis. The authors first subjected the vibration signal to a feature extraction procedure based on the multiscale permutation entropy (MPE) [40], and then proceeded to classify the conditions for the extracted characteristics. Tang et al. [41] diagnosed failures of a wind turbine transmission system using SVMs. The authors used non-stationary vibration signals from wind turbine transmission systems as inputs, extracting essential features. This high-dimensional information was subsequently reduced by applying a manifold learning algorithm. This information was finally sent to a Shannon wavelet SVM classifier.

Wang et al. [42] used semi-supervised mapping to diagnose rolling bearing failures in wind turbines. The authors first extracted the characteristics from the multi-scale rolling bearing vibration signals and subsequently, the low-dimensional characteristics were sent as input to the SVM-based classifier for pattern recognition. Yao et al. [43] exploited SVMs for fault diagnosis for electric vehicles powered by lithium batteries. To ensure the robustness of the method, the authors adopted an algorithm based on a grid search method to optimize the kernel function parameter and the penalty factor. Zhao et al. [44] applied a robust vector machine for least squares support for aircraft engine failure diagnosis. The implemented algorithms have been tested on both regression and classification problems.

Machine Learning-based algorithms make extensive use of optimization methods to search for the optimal solution. A widely used optimization procedure is the one called particle swarm

optimization (PSO) [45]. At each iteration, the algorithm identifies a new candidate for the best in the search space, based on a specific quality measure called fitness. Van et al. [46] used SVMs in combination with a particle swarm optimization and a least squares procedure. These algorithms have been implemented for the elaboration of a classifier for the diagnosis of the bearing failures of a rotating machine. A similar procedure was implemented by Li et al. [47] for diagnosing machinery failures in high voltage circuit breakers. Again Fan et al. [48] exploited particle swarm optimization in combination with SVMs for rolling bearing failure detection.

Mirakhorli [49] developed an SVM-based classifier for diagnosing faults in a distillation column. Gao et al. [50] used SVMs to diagnose the mechanical failures of an on-load tap-changer. Huang et al. [51] used SVMs with a modified gray wolf optimization for transformer fault diagnosis. The optimization is performed on the penalty factor and on the kernel parameter. Zhang et al. [52] used a combination of vector support machine (SVM) and genetic algorithms (GA) for transformer fault diagnosis. The methodology also makes it possible to evaluate the conditions of the transformer oil bath insulation, providing an accurate tool for predicting the operating conditions. Liu et al. [53] applied SVMs for the prediction and diagnosis of energy consumption of public buildings in China. 11 input parameters were selected, including historical data on energy consumption, climatic factors, and time cycle factors, monitoring the energy consumption to produce air conditioning in the city of Wuhan, in the period from June to September. Ibrahim et al. [54] used SVM-based algorithms to diagnose failures of satellite subsystems using related telemetry parameters. Telemetry data was collected by the Egiptosat-1 satellite, launched into Earth orbit in April 2007 and which lost communication with the ground station in 2010. Zhao et al. [55] used SVMs for turboshaft engine failure detection. The method assigns a weight to each sample through a specific weight calculation method to improve the robustness of the algorithm.

Guo et al. [56] exploited SVMs for monitoring and diagnostics of an industrial chemical process. The authors transformed a multi-class problem into multiple binary classification problems, thus training N models, each with the task of distinguishing one class from all the others and obtaining a correct diagnosis overall. Poyhonen et al. [57] used SVMs to solve an induction motor rotor failure diagnosis problem. The authors used vibration signals collected from real motors under different health conditions, with a vibration sampling rate of 40 KHz. Three different feature extraction techniques are then proposed, which are used as input for the SVM model. das Chagas Moura et al. [58] used SVMs to solve a regression problem for predicting the remaining life of turbochargers of diesel engines and the remaining travel distance of automobile engines. Chen et al. [59] exploited SVMs for detecting equipment failures in a thermal power plant. This work integrates a size reduction scheme to analyze turbine failures in thermal power plants. He et al. [60] used SVMs to identify weld quality using machine current, voltage, and speed data as inputs: The authors used SVM models in sequence.

Yan et al. [61] used a semi-supervised SVM-based algorithm for ventilation air conditioning (HVAC) failure detection that only captures a few defective training samples. Yin et al. [62] verified the performance of an SVM-based model for process monitoring in complicated industrial processes. The authors showed that these algorithms are particularly advantageous in generalization performance and in the case of small input datasets.

Islam et al. [63] collected acoustic emission signals for the diagnosis of bearing defects by adopting a model based on SVMs. First, they perform an extraction of the high-dimensional fault characteristics to train the classifier, which are composed of the statistical descriptors in the frequency domain and the complex analysis of the envelope spectrum. Monteiro et al. [64] developed an SVM-

based decision model for fault diagnosis on automotive vehicle transmission gearboxes. Yang et al. [65] applied an SVM-based algorithm for bearing failure diagnosis. To overcome the limits to the model recognition capacity due to a not very good function of the kernel and its parameters, the authors have introduced an Ant Lion optimization [66]. You et al. [67] have applied SVMs for the diagnosis of failures of rotating machines. The vibration signal from the machinery functioning was subjected to a characteristic extraction procedure, returning the time spectrum of the dyadic wavelet energy and the power spectrum of the coefficients of the maximum wavelet energy level. Kumar et al. [68] used SVMs for automatic defect detection from the centrifugal pump's vibration signal. The raw signal measured on the pump is subjected to a characteristic extraction procedure in the time-frequency domain, then a genetic algorithm is applied to identify the optimal parameters of the SVM-based model. Chen et al. [69] fault-diagnosed the Loader Gearbox using SVM-based algorithms. The authors measured the emitted noise using sound intensity probes and extracted the characteristics through the independent component analysis (ICA) technique [70]. Finally, they sent the correlation coefficient between the independent components and the source data as input to the classifier. Wenyi et al. [71] developed a wind turbine failure diagnosis model using SVMs. Vibration signals from the rotating parts of wind turbines were used as inputs, the diagonal spectrum was extracted from them. Djeziri et al. [72] has developed an automatic system for identifying the presence of a pollutant in a gas mixture using an intelligent sensor based on temporal based SVM.

### 3.2. Artificial Neural Network (ANN) algorithms

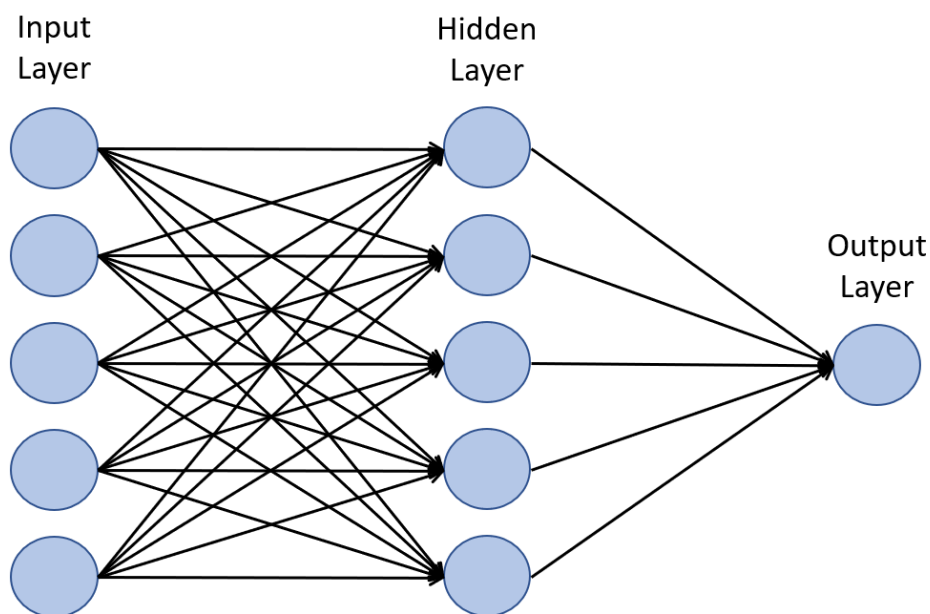
Artificial Neural Networks (ANNs) are mathematical models that are used to solve engineering problems of Artificial Intelligence. They are mathematical-informatic computation models whose functioning is like that of biological neural networks, in fact, they are models made up of interconnections of information. ANNs are inspired by the functioning of the animal brain, defining the central body as a mathematical model, called a node, characterized by an activation function, a threshold value and possibly a bias.

Each node receives as input a set of signals from the previous units: These signals reach the neuron after being weighed, their combination, after having algebraically summed the bias if present, becomes the variable of the activation function ( $f$ ), determining the activation, or non-activation, of the neuron [73].

A neural network is therefore a set of nodes arranged in layers connected to each other by weights. The first layer is called the input layer, the last is the output layer, while the intermediate ones are defined as hidden layers, and are not accessible from the outside as all the characteristics of the complete network are stored in the matrices that define the weights [74]. The type of network determines the type of connections that are present between the nodes of different layers and between those of the same layer. The typical architecture of an ANN provides a feedforward configuration, in which each node is connected to those of the previous layer, from which it receives inputs, and to those of the next layer, to which it provides output [75] (Figure 5).

The choice of the activation function determines the substantial difference with the equivalent of the biological neuron: In the latter the sum of the incoming impulses is transmitted directly to the axons if the threshold is exceeded, essentially behaving like a linear regression model, approximating the distribution of data with a straight line [76]. The use of a non-linear function allows, however, to have a better representation of the signals, without considering that sometimes a linear regression is not

usable. The most used activation functions are step, sigmoid, rectified linear unit (ReLU), hyperbolic, and logistic [77].



**Figure 5.** Artificial Neural Network architecture.

For neural networks, both training and error evaluation are fundamental: For this reason, the data are divided into training sets, validation sets and possibly a testing set. The first group is used for neural network training and contains the correct inputs and outputs, as supervised training is required: It usually consists of about 70–80% of the total data [78]. The second is used for validation, that is, to evaluate the accuracy of the neural network by calculating the prediction error that is chosen to use. The third allows you to test the network, so it consists in the real use of the designed network [79].

The training phase is crucial in the development of the model as it is in this phase that the system learns the characteristics of the system to be able to simulate them later. This ability is acquired by updating the connection weights, with an optimization procedure [80]. At each iteration, the system compares the output obtained with the target provided in the training set and evaluates the error made. Based on this value, it proceeds to update the weights until the iterative procedure converges [81].

Feedforward networks are those with the simplest architecture, being composed of an input layer, one or more hidden layers and an output layer. Each neuron has its input parameters from the previous layer and no cross connections are possible between nodes of the same layer or cycles in which the output is sent to previous layers. The information flow, therefore, proceeds in only one direction and the output of each cycle is determined only by the current input. Being a very simple type of network, it is by far the most used [82].

There are many scientific works that have adopted the ANNs for the elaboration of models for the diagnosis and detection of faults. Zhang et al. [83] investigated the problem of faults diagnosing in oil-filled power transformers. The model detects the gas dissolved in the oil and identifies the possible failure. Hoskins et al. [84] applied ANNs to identify failures in complex chemical plants. Ali et al. [85] used vibrational signals to diagnose rolling bearing failures using an ANN-based model. The feature

extraction was performed using an algorithm based on the energetic entropy of decomposition in an empirical mode. Sorsa et al. [86] detected the acoustic signals of a continuous stirred tank reactor with heat exchanger and developed an ANN-based fault classification model. Saravanan et al. [87] used ANNs for fault diagnosis of a mechanical gearbox. The discrete wavelet transform (DWT) is evaluated for the extraction of the characteristics demonstrating its validity in representing all possible types of transients in the vibration signals generated by faults in a reducer. Chine et al. [88] have adopted ANNs for the diagnosis of faults for photovoltaic systems. The authors evaluated several attributes through simulation models and compared these values with those measured in the field. They then labeled the fault conditions and submitted the input to the ANN classifier. Li et al. [89] have studied the problem of fault diagnosis of rolling bearings of an electric motor with the use of ANNs. The authors extracted the characteristics from the vibration signals of the bearings in the time/frequency domain and subjected the results to an ANN classifier.

Samanta et al. [90] compared three ANN-based models for bearing fault diagnosis. Multilayer perceptron (MLP), radial basis function network (RBF) and probabilistic neural network (PNN) were treated by the authors using time domain vibration signals as input signals. Han et al. [91] proposed a method of diagnosing induction motor failures using ANNs. The stator current signals were detected and subjected to discrete wavelet transformation (DWT), subsequently a genetic algorithm was applied for the optimization of the characteristic parameters of the ANNs, finally the ANN was trained and tested. Wang et al. [92] developed a classification model based on partially linearized neural networks (PNNs) for the diagnosis of failures of a rolling element bearing. The vibration signals were detected in the frequency domain and used as input of the classification model. Hashim et al. [93] have proposed a method of diagnosing positive ignition engine combustion failures. The method uses the detected vibration signals, performs a wavelet packet transformation for the extraction of the characteristics, operates an optimization process for the selection of the wavelet denoising, and finally uses the latter for classification through an ANN.

Iannace et al. [94] used ANNs to diagnose failure in the blades of an unmanned aerial vehicle (UAV). The acoustic signals produced by the blades of the UAV were detected in an anechoic chamber, and subsequently the frequency components of the signals were extracted, finally the records were labeled. This data was sent as input to an ANN classifier for the detection of fault conditions. Kordestani et al. [95] applied ANNs for the diagnosis of faults in the multifunctional spoiler (MFS) of a jet aircraft. The model correctly classified three types of faults: zero bias current, actuator leakage coefficient, and internal leakage faults. The features extraction was performed with the discrete wavelet transform (DWT). Shi et al. [96] developed a refrigerant charge failure diagnosis system of a variable refrigerant flow (VRF) system. The model uses the ReliefF algorithm for feature selection and processes and optimizes the ANN with the Bayesian regularization algorithm. Xu et al. [97] used vibration signals to diagnose failures of rotating machines, developing a model based on neural networks and fuzzy systems. For the extraction of the characteristics, the authors applied the principle of function selection of the wavelet transform and the principle of the soft threshold of the denoising of wavelet packets. Viveros-Wacher et al. [98] proposed a method of diagnosing failures in an ANN-based CMOS RF negative feedback amplifier. Heo et al. [99] studied the problem of fault detection in process systems engineering with the use of ANNs. The authors developed a classification model for fault detection problems, and subsequently trained the neural networks to perform fault detection. Furthermore, the effects of two hyperparameters such as the number of hidden layers and the number of neurons in the last hidden layer, and the increase in data on the performance of neural networks

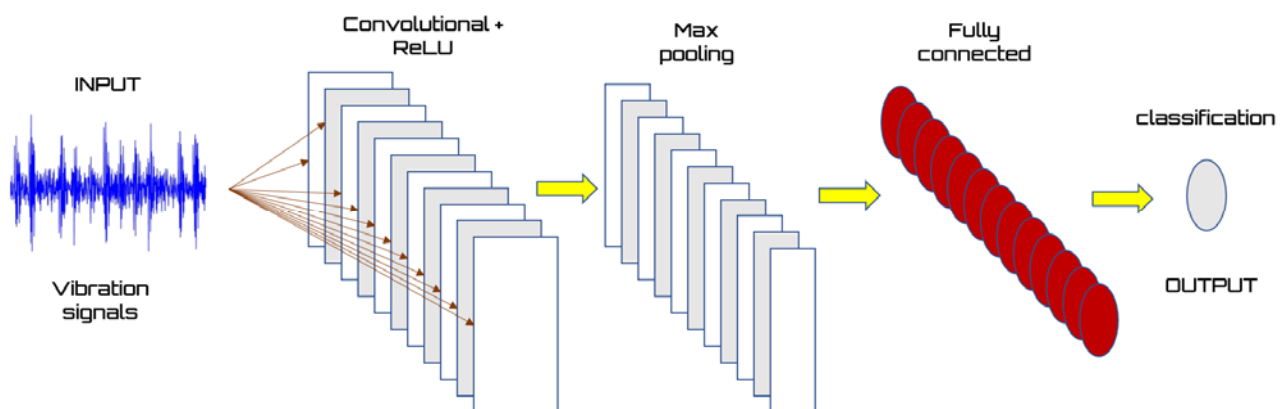


were investigated. Agrawal et al. [100] compared the results of ANN-based and SVM-based models for diagnosing bearing failures. The vibration signals of the bearings were detected on an experimental test bench, and the wavelets are used for the extraction of the characteristics, selecting them according to the criteria of maximum energy and minimum entropy. The authors concluded that the SVM-based model is the most accurate of all the classification algorithms considered followed by ANN showing 98% accuracy.

### 3.3. Convolutional Neural Network (CNN) model

Deep Learning is a branch of Machine Learning that is based on the use of algorithms whose purpose is the modeling of high-level abstractions on data [101]. It is part of a family of techniques aimed at learning methods for representing data. In Deep Learning, specialized learning algorithms are developed in the automatic extrapolation of features in a data set, to be used later for the training of machine learning systems. The result is relevant because without these techniques the features would have to be produced and evaluated manually and prior to training. The key concept on which deep learning is based is to subject the input data to numerous levels of cascade processing, the result of which is the emergence of these features [102].

In the field of neural networks this concept has been put into practice with the addition of numerous hidden levels of neurons. Like classical neural networks, deep neural networks can model complex relationships between input and output data. Among the most successful applications we find computer vision, with tasks that include classification, image regression and object detection. In object detection, for example, a deep neural network can generate a layered representation of objects in which each object is identified by a set of characteristics in the form of visual primitives, that is, edges, oriented lines, textures, and patterns recurring [103].



**Figure 6.** Typical Convolutional Neural Network Architecture for a fault diagnosis system.

Convolutional neural networks (CNNs) represent a type of neural network in which the connection pattern between neurons is inspired by the structure of the visual cortex in the animal world. The single neurons present in this part of the brain respond to certain stimuli in a restricted region of observation, called the receptive field. The receptive fields of different neurons are partially overlapped so that they cover the entire visual field as a whole [104]. The response of a single neuron to stimuli

that take place in its receptive field can be mathematically approximated by a convolution operation. CNNs are designed to recognize visual patterns directly in pixelated images and require little or no preprocessing. They can recognize extremely variable patterns, such as freehand writing and images representing the real world. Typically, a CNN consists of several alternating levels of convolution and subsampling or pooling, followed by one or more fully connected final levels in the case of classification, or by several up-sampling levels in the case of regression. In the latter case we speak of a fully convolutional neural network (FCN) [105].

In a typical convolutional neural network architecture, we can find the following layers (Figure 6):

- Input Level: Provides the data that needs to be analyzed.
- Convolutional level: it aims to identify patterns, there are more than one and each of them focuses on the search for essential characteristics present in the initial dataset.
- ReLu level (Rectified Linear Units): the objective is to cancel negative values obtained in previous levels.
- Pool level: allows you to identify if the study feature is present in the previous level.
- Fully connected level: connects all the neurons of the previous level in order to establish the various identifying classes according to a certain probability. Each class represents a possible final answer.

Typically, CNNs use a higher number of hyperparameters than classic neural networks. Among those that differentiate them from the latter we find for example the number of filters: Since the spatial dimensions of the feature maps decrease going deeper into the network, the levels close to the input level will tend to have a reduced number of filters while you near the output will have more filters [106]. To try to equalize the number of filters along the entire network, we usually try to keep the product between the number of feature maps and the number of spatial positions that are considered in the input constant between all levels. By doing this, the information deriving from the input is preserved throughout the network [107].

The shape of the filters represents another hyperparameter, it usually varies from network to network and is chosen based on the characteristics of the dataset used. The goal is to find the right compromise between granularity and detail to create abstractions of the right scale for a particular dataset. Furthermore, the shape of the filters used in max pooling represents a parameter that depends on the specific dataset used. High resolution images may need large filters to appropriately reduce the size of the inputs, while for low resolution images too large rectangles may lead to too small representations in the most advanced stages of the network, with consequent loss of information. Typically,  $2 \times 2$  size rectangles are used [108].

As with classic neural networks, even for CNNs it is possible to use the classic regularization techniques to combat overfitting. Furthermore, it is possible to make use of the so-called data augmentation technique. This technique consists in making small random changes in the inputs, such as rotations, translations, cropping and other image processing operations, with the aim of increasing the effective number of examples and consequently counteracting overfitting [109].

CNNs represent one of the latest evolutions of Machine Learning and the fault diagnosis sector immediately understood the usefulness of this tool in dealing with such problems. Wen et al. [110] applied CNNs to develop a method of diagnosing mechanical component failures. The LeNet-5 architecture [111] was used by converting the vibrational signals acquired into two-dimensional (2-D) images, in this way the effect of the extraction of the preventive characteristics is eliminated. The authors tested the model on three datasets: motor bearing dataset, self-priming centrifugal pump

dataset, and axial piston hydraulic pump datasets. Wu et al. [112] developed a CNN-based model for diagnosing faults in chemical processes. For the extraction of features in the spatial and temporal domains, the authors exploited convolutional layers, pooling layers, dropouts, and FC layers. The Tennessee Eastman (TE) reference process was used for performance verification. Zhang et al. [113] proposed a methodology for diagnosing bearing failures in noisy environments based on the use of CNNs. Raw acoustic signals were used as input without any pre-denoising method. The model demonstrated strong domain adaptability with returning high accuracy under different workloads. Jing et al. [114] for the monitoring of the conditions of the reducer of a mechanical system they exploited the abstraction capacity of CNN. The authors demonstrated that learning the features offered by a CNN architecture provides better results than hand-crafted feature extraction. Chen et al. [115] used CNNs for fault identification in a gearbox. Gearbox vibration signals were detected and preprocessed using statistical measurements from the time domain signal such as standard deviation, asymmetry, and kurtosis. In the frequency domain, the spectrum obtained with FFT is divided into multiple bands, and the mean square value (RMS) is calculated for each so that the energy retains its shape at the peaks of the spectrum. The authors tested the model with 20 test cases with different combinations of condition models, where each test case includes 12 combinations of different underlying condition models. Guo et al. [116] used an algorithm based on hierarchical adaptive deep CNN for determining the severity of bearing failures. The vibrational signals were acquired on a test bench and sent to CNN which returned a good recognition of the failure pattern and a good evaluation of the failure size. Janssens et al. [117] detected failures of rotating machines using a CNN-based algorithm. The authors detected the vibrational signals of different types of bearing defects, such as outer race failures and lubrication degradation, while also adding healthy bearing signals and rotor imbalance signals to these defects. The performance of the model was compared with those returned by a model based on a random forest classifier, from the comparison the CNNs obtained greater accuracy in the classification of faults. Zhang et al. [118] developed a CNN-based fault diagnosis system. The authors sent the raw vibration signals to a CNN that uses large kernels in the first convolutional layer to extract features and suppress high-frequency noise. The small convolutional kernels in the previous layers are used for multilayer nonlinear mapping. Adaptive Batch Normalization (AdaBN) [119] was used to improve the adaptability of the model. Ince et al. [120] proposed CNNs for early detection of engine failures. The authors applied a 1-D CNN with an inherent adaptive design that merges the features extraction and classification steps into a single tool. The raw vibrational signals were detected and sent to the fault detection system in real time.

Zhang et al. [121] studied bearing failure diagnosis using CNNs. To overcome the critical issues related to the use of these methodologies, the authors extracted an input image from the vibrational data through the application of the theory of the short-lived Fourier transform. In addition, they exploited the Scaled Exponential Linear Unit (SELU) activation function to avoid the deactivation of an excessive number of nodes during the training process, and finally applied hierarchical smoothing for better results. Azamfar et al. [122] performed a motor current signature analysis using CNNs to diagnose gearbox failures. The authors detected the current signals through multiple sensors and sent those raw signals directly to CNN without doing any manual feature extraction. The method was validated using motor current data measured on a test bench equipped with industrial gearboxes in various health conditions and with different working speeds. Zhou et al. [123] used CNNs on an unbalanced dataset of rotating machinery failures. The authors adopted a non-linear self-regressive neural network (NARNN) to expand the small number of failure records available in the dataset.

Subsequently, the detected one-dimensional vibration signals are processed with the continuous wavelet transform to convert them into two-dimensional time-frequency images. Finally, a CNN-based classification model is developed to automatically learn characteristics and obtain fault identification. Zhang et al. [124] adopted an augmented CNN for bearing failure diagnosis. In the application of Machine Learning-based algorithms, the crucial component for the success of modeling is the quality and quantity of the samples. A reduced or unbalanced dataset on a class is unlikely to return good classification performance. To overcome these problems, the authors added a multiscale feature extraction unit to the deep neural network layers to extract features at different time scales without adding convolution layers. This technological solution reduces the depth of the network while still providing a good classification capacity, and the simplicity in its architecture reduces any overfitting problems. Yongbo et al. [125] used infrared thermal imaging (IRT) to diagnose failures of rotating machinery by applying CNNs. The authors first detected IRT images of rotating machines by predicting the different operating conditions including failures. Subsequently, they developed a CNN to extract the characteristics of the faults; the obtained characteristics are inserted in a Softmax regression classifier (SR). Chen et al. [126] developed a bearing failure diagnosis model based on the combination of Cyclic Spectral Coherence (CSCoh) [127] and CNN. The Cyclic Spectral Coherence is exploited to extract the discriminating characteristics of the bearing health states in different operating conditions, starting from the vibration signals. The data obtained, after group normalization (GN), are subjected to a classifier based on CNN. Zhou et al. [128] proposed a gas turbine failure diagnosis methodology that leverages CNNs. The authors note that there is a strong coupling between gas path failures and sensor failures, when both failures occur simultaneously then it becomes difficult to correctly identify the nature of the failure. In this work, a method based on a CNN optimized by Extreme Gradient Boosting (XGBoost) [129] is developed to make the effects of sequencing on the diagnostic accuracy of the network interpretable.

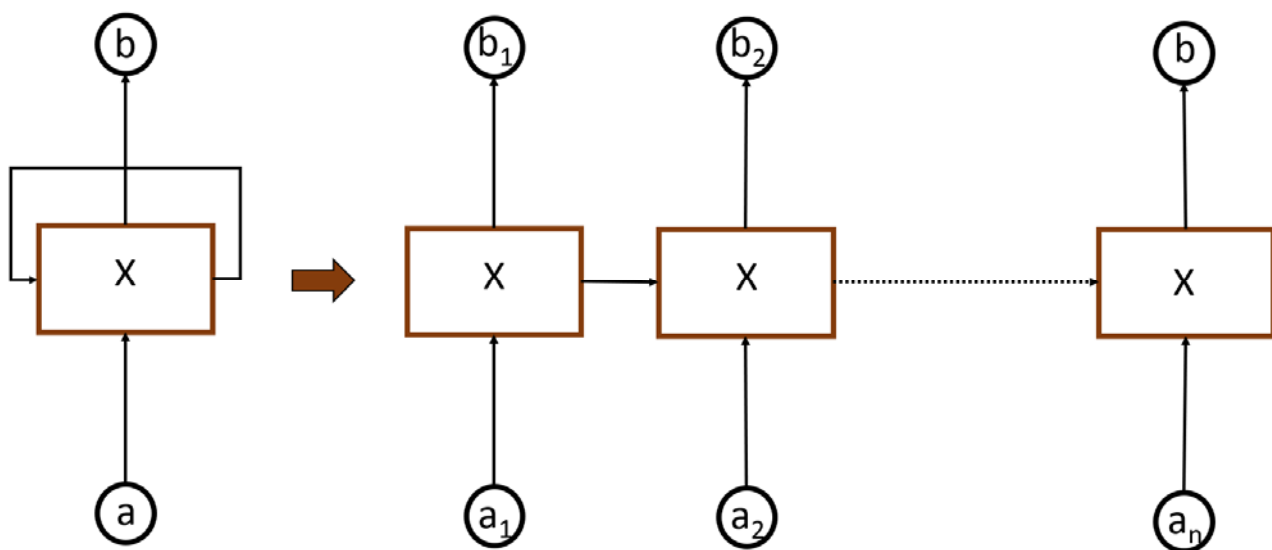
Li et al. [130] applied CNNs to develop a fault diagnosis system. The method is structured with a fusion layer in the frequency domain and a feature extractor. The first layer uses convolution operations to filter signals at different frequency bands and combine them into new input signals. These signals are sent to the feature extractor to extract features and perform domain adaptation. Chen et al. [131] diagnose rolling bearing failures with CNNs. The authors detected the vibration signals at the rolling bearing. Raw signals are divided into training, validation, and test sets. The training set is sent as input to a one-dimensional CNN. After validation, the test set is sent to the trained model for fault detection. Liu et al. [132] developed a rotating machinery failure diagnosis technique using CNNs. The authors transform the vibration signal acquired through the decomposition of the wavelet packet into an energy spectrum matrix containing information relating to faults. The training of the model is performed with dynamic adaptation to adaptively extract the robust characteristics from the spectrum matrix. Hoang et al. [133] applied CNNs for bearing failure diagnosis. The authors demonstrated that a CNN could extract discriminating features automatically with greater efficacy than that returned by multiple sensors connected in parallel.

### *3.4. Recurrent Neural Network (RNN) applications*

Recurrent Neural Networks (RNN) are neural networks specialized in processing sequential data. This type of network is therefore optimized for tasks related to the recognition of defects in machines. The sequential input data is analyzed one at a time following the order of the sequence of discrete

times [134]. At the base of the RNNs architectures there is the sharing of parameters in different parts of the model. This property makes it possible to extend and apply the model to examples of different forms of data by increasing the generalization capabilities of the network on these parameters. During the input processing phase, the RNNs keep a state vector in their hidden layers that implicitly contains information about the history of all the elements of the past of the sequence, that is, of the previous instants of time. Considering the output of the hidden layers at different times of the sequence as the output of different neurons of a deep multilayer neural network, it becomes easy to apply backward propagation to train the network. However, although RNNs are powerful dynamic systems, the training phase often turns out to be problematic because the gradient obtained with the backward propagation either increases or decreases at each discrete time, so after many instants of time it can either become too large or become little appreciable.

Figure 7 shows a typical training process of an RNN with an indication of the typical recursive structure. In the Figure 7, to the left of the arrow we used a cyclical representation of sequential processing, while to the right of the arrow the same sequence is deployed along all the processing processes iterated during sequential processing: This procedure is called network unfolding. The clustered hidden units take input from the neurons of the previous phase, so the network can map an input sequence formed by the input elements into a sequence of output elements, where each element depends on all input data at instants prior to the current one. The same parameters are reused at each subsequent stage [135].



**Figure 7.** Recurrent Neural Network architecture with the unfolded scheme.

Many other architectures are possible: An example can be obtained by including a variant in the network that causes it to generate an output sequence that is used as an input for subsequent instants. The backward propagation algorithm is applied directly to the computational graph obtained by deploying the sequential branch of the network, which in this situation can be considered as a multilayer network of which each layer represents a single cycle of the sequence, having shared weights [136].

Despite the main purposes of recurrent networks in long-term learning, theoretical and empirical

evidence shows that it is difficult to learn by storing so much information for very long-time sequences. In fact, the web often tends to focus on recent information. Data learned in very previous instants of time could generate errors during training. The solution to this problem is to increase the size of the network by adding explicit memory [137].

One type of networks that implement this is long short-term memory networks (LSTM). These networks use special hidden layers formed by units that specialize in remembering inputs for very large time intervals [138]. A special unit called a memory cell act as an accumulator, as if the neuron in the network were equipped with a permeable membrane (gate). It has connections on itself to the next time with a unit weight, so it can copy the real value of the state by accumulating external signals. This auto-connection is linked to a unit instructed to decide when to clear the memory contents. The concatenated structure of LSTMs allows instead to have a single layer with several interacting neurons according to a particular pattern. The state cell undergoes few linear operations allowing the information to travel unaltered. The network can remove or add information to the state cell by means of the gate structure. These structures are composed of a neural layer with a sigmoid activation function and a point multiplication operation. The output values of this state, between 0 and 1, quantify how much information from the input must be allowed to flow into the network. A value of 0 means that nothing must pass while a value of 1 indicates the total passage of information. The mechanism of this gate-like cell that opens and closes explains its name gate and justifies the use of the sigmoid function to control the flow of input information [139].

An LSTM has three gates to protect and control the state cell. The first, called the forget gate layer, decides what information you want to put into the network flow. The second, called input gate layer, decides which values must be updated, immediately after a hyperbolic tangent layer creates a vector having as elements the new candidate values to add to the state. The last gate finally decides which part of the state vector must be returned in output. LSTMs are more effective than traditional RNNs in many applications, especially when they have many layers for each instant of time [140].

Given the sequential nature of the vibrational and sound signals of the machines, this type of algorithm has been widely used by the scientific community to address the problems associated with the diagnosis of faults in machines. Jiang et al. [141] have adopted RNNs for fault diagnosis for rolling bearings. The authors used the frequency spectrum sequences of the vibrational signals as input to reduce the data size and ensure good robustness. The recurring hidden layer is adopted to automatically extract features from the input spectrum sequences, and an adaptive learning rate is applied in the training process to improve model performance. De Bruin et al. [142] proposed a system for diagnosing faults in railway circuits with the use of LSTMs. The authors detected signals from multiple railway tracks in a specific geographic area and sought to diagnose failures as a function of spatial and temporal dependencies. They showed that the LSTM network can learn these dependencies directly from the data. Yang et al. [143] applied LSTMs for diagnosing wind turbine transmission failures. The authors exploited the spatial and temporal dependencies of the measurement signals detected by multiple sensors in rotating machines, to detect the different types of faults and proceed with their classification. Talebi et al. [144] developed an RNN-based fault detection and isolation (FDI) system. The authors applied this model to the data collected by the attitude control subsystem of satellites in low earth orbit. Faults related to both actuators and sensors were considered. Zhang et al. [145] used RNNs to study a fault diagnosis system in chemical industrial processes. Data-based fault detection and diagnosis (FDD) methods are particularly suitable for this type of problem even if they require a great deal of computational effort given the amount of information to be handled. The methods based on the RNN,

can extract the characteristics from the raw data for time series data. The authors adopted a bidirectional RNN to improve the number of features extracted and thus improve the performance of the system. An et al. [146] exploited a dataset containing vibration signals from bearings of rotating machines with varying speeds and loads over time to test an LSTM-based fault detection model. To begin, the data is segmented, then the classification labels are transferred to the LSTM, and finally the probability of occurrence of the failure is detected by the output network.

Rotating machines are once again studied by Liu et al. [147] who applied LSTMs to detect failures. The authors measured the vibration signals and subsequently segmented them to shorten the length of the timeline. Furthermore, they addressed the problem of the large number of parameters and calculations required by an LSTM, exploiting a cellular structure with a forgetful gate. Liang et al. [148] addressed the problem of diagnosing faults in the bogie of a high-speed train with the adoption of a model based on a recurrent convolutional neural network. The authors detected the cart's vibrational signals and then used convolutional layers to filter the characteristics of those signals. These characteristics are sent to the recurring layers with a simple recurring cell, recording performances superior to a CNN and to models based on the learning of the ensemble. The same problem was subsequently addressed by Huang et al. [149] exploiting a model based on an LSTM. The authors used SIMPACK simulation software to generate fault data. These data were subsequently exploited to train and test the network showing a good ability to learn the spatial and temporal correlation of fault characteristics in vibration signals, without data preprocessing and prior knowledge.

Shahnazari et al. [150] applied RNNs for fault detection and isolation of a heating, ventilation, and air conditioning (HVAC) system. The authors developed predictive models by exploiting the plant data to incorporate them into the filters of the diagnosis system. The method was tested using simulation data on a test bench, and real data. The same author then applied RNN more generally for fault diagnosis for non-linear systems [151]. Guo et al. [152] instead applied RNNs to predict the remaining useful life of the bearings. The difficulties related to this problem are to be referred to the different contribution of the characteristics and the difficulties in identifying a threshold value. The authors extracted six similarity characteristics from the vibration signals and correlated them with eight time-frequency characteristics. Then they selected the most sensitive ones and sent them to an RNN. A similar work but this time related to the aeronautical field was carried out by Yuan et al. [153]. In this case, a model based on LSTMs was adopted and tested with vibrational signals from aircraft turbofan engines supplied by NASA.

Wu et al. [154] once again dealt with the problem of bearing diagnosis using an LSTM. In this work, LSTMs are used to generate ancillary datasets, so using a small amount of labeled data you can achieve more effective and robust fault diagnosis performance than other methods. Yin et al. [155] modeled the operating conditions of a wind turbine gearbox with the aid of an LSTM. The authors used Cosine Loss to reduce the signal strength product and improve the accuracy of the diagnosis. The characteristics of the energy sequence and the entropy of the wavelet energy were extracted from the vibration signals and sent to the LSTM for fault diagnosis. Xia et al. [156] estimated the useful life of the machines by developing a forecasting model based on LSTMs. The authors detected the sequential vibrational data with the use of different sensors and then merged them and sent them as input to the model without operating any feature extraction. The long-term memory levels of LSTMs are exploited to extract temporal characteristics from sequential data and keep track of them. The model training process is performed by adopting the dropout technique and decreasing learning rate. Wang et al. [157] studied an automatic fault diagnosis system for use on Internet Data Center chillers. The system adopts

a hybrid approach using a 1-Dimensional Convolutional Neural Network (1D-CNN) and a Gated Recurrent Unit (GRU). First the time series sequences for the refrigeration system are detected, which are sent to the convolutional layer which extracts the local characteristics, then the GRU intervenes which, with its memory, can extract the global characteristics.

### 3.5. Deep generative systems

Generative models have the purpose of learning a certain distribution defined on a set of data belonging to some space. The model analyzes a training dataset from an unknown distribution and learns to represent an estimate of that distribution in some way. The result is a probability distribution that can be explicitly estimated or used implicitly only to generate new examples [158]. Generative models search for joint probabilities, creating points where a given input characteristic and a desired output exist simultaneously. Generative models then estimate probabilities and likelihood, modeling the data presented to them and distinguishing between classes based on these probabilities. Having learned a probability distribution, the model builds on this probability distribution to generate new data instances. By learning the distribution of data, it is therefore possible to build new ones with characteristics like those of the originals. To do this we can see our examples as if they belonged to a distribution and our goal is to learn another distribution sufficiently like the starting one [159].

- Generative models can be grouped into two types:
- Generative Adversarial Networks (GAN)
- Autoencoders

The basic idea of opposing generative networks (GANs) is to establish a non-cooperative game between two players, one player being called a generator and the other a discriminator [160]. The generator creates samples from an estimate of the distribution of the training data, while the discriminator obtains samples from both the generator and the training set and must be able to distinguish where each of these come from, or in other words determine whether they are true or false [161]. So, the goal of the generator is to induce the discriminator to classify the generated images as true. As the game progresses, the generator learns to produce more and more real samples and the discriminator on its part learns to better recognize the data generated from the real data: All this with the aim that at the end of the competition the generated data are indistinguishable from actual data. To know the distribution of the generator on the training data, which have an unknown distribution, and a priori probability distribution is defined on the input variables, which are generated randomly [162].

Autoencoders are a type of neural network used to obtain a compressed representation of a data with many dimensions, such as an image. The structure includes a neural network called encoder, which has the task of compressing the input into a small-sized vector  $z$ , and a neural network called decoder, which taken as input  $z$  tries to reconstruct the starting image [163]. A cost function that evaluates the difference between the starting image and the reconstructed image allows the network to learn and reconstruct increasingly similar images. If the activation function that regulates the activation of the neurons of the hidden level is linear, and the criterion of mean square error is used to train the network, then the  $n$  hidden units learn to project the input in the interval of the first  $n$  main components of the data. If, on the other hand, the non-linearity characteristic is somehow conferred on the hidden level, the auto-encoder acquires the ability to capture multi-modal aspects regarding the distribution of the input [164].

The decoder could be a useful tool for generating content, eliminating the encoder. However, the



space of latent variables produced by an autoencoder consists of a series of scattered points without a precise structure. By randomly sampling the latent space, it would be unlikely to obtain a vector of variables that corresponds to a reasonable encoding of an input data, precluding the possibility of generating realistic content. A solution to this difficulty is offered by the Variational Autoencoder (VAE), whose basic structure remains the same as an autoencoder, with the only difference that the encoder no longer generates a vector of latent variables, but, for each variable, a mean  $\mu$  and a variance  $\Sigma$ . From the normal distribution with mean  $\mu$  and variance  $\Sigma$  the  $z$  is then sampled and taken as input by the decoder. This procedure allows to define, for each record of the training set, not only a single point in the latent space, but a point and its surroundings. However, the problem of the structure of latent space has not been solved. We still have spots that, although they offer more coverage, are still scattered around. The structuring is obtained by adding the Kullback-Leibler divergence between the distribution produced by the encoder and the normal distribution with mean 0 and variance 1 to the cost function of the model. In this distribution and from which it is then possible to sample in the future [165].

Finally, there is a third model called adversarial autoencoder (AAE), which deals with generative models produced by the union of VAE and GAN. What differentiates this structure from a VAE is the fact that what drives the distribution learned from the encoder towards that learned from the decoder is an opposing network [166]. The VAE encoder is considered as the generator of a GAN for which a discriminator can be used that tries to distinguish data belonging to the encoder from those coming from decoders. The training of the opposing network and the autoencoder takes place jointly, using stochastic gradient descent. Two phases are performed on each minibatch [167]:

- the reconstruction phase, in which the encoder and decoder try to minimize the input reconstruction error.
- the regularization phase, in which the discriminator parameters are modified to identify the data generated by the encoder from those belonging to the decoder.

A generative model represents a design choice suitable for learning any type of data distribution using unsupervised learning. To do this, we can use the power of neural networks to learn a function that can approximate the true distribution to allow us to predict the distribution of the model. What we get is not an exact copy of the original distribution but an approximation of it that can remind us of the essential characteristics. The usefulness of such a representation for identifying faults is intuitive. In fact, a fault represents a deviation from the normal operating conditions of the machine, therefore a model that can capture the essential information of the operation of the machine will also be able to identify these conditions of deviation. Liu et al. [168] used GANs to diagnose rolling bearing failures. The authors showed how convenient it is to use an unsupervised methodology for fault diagnosis while negating the operational cost of data labeling. The mixed time-frequency characteristics of the vibrational signal were first extracted, and the GAN showed its great ability to group the data. Shao et al. [169] leveraged GANs for data augmentation. The authors detected the vibration signals of an induction motor through sensors and generated one-dimensional raw data using a GAN. Zhang et al. [170] developed a GAN-based fault diagnosis system. The noise distributions and data on temporal vibrations of real machinery were first collected: The dataset obtained is unbalanced due to the difficulty of collecting fault data compared to those relating to normal operating conditions. To balance the data content, a GAN-based model is used to explicitly produce failure data. Wang et al. [171] studied a planetary gearbox failure diagnosis model based on GANs. The model uses both Generative Adversarial Networks and a Stacked Denoising Autoencoder (SDAE) [172]. The vibration signals from the planetary gearbox are sent to a GAN generator which creates new samples with similar

distribution to the original samples. Subsequently, these data are transformed by the SDAE discriminator to automatically extract the fault characteristics and discriminate their authenticity and fault categories. Li et al. [173] applied GANs to enrich the dataset of vibrational data measured on rotating machines to identify fault conditions. The problem of unbalanced data was also addressed with the help of GAN by Wang et al. [174] who studied the classification of mechanical failures. A similar thing was performed by Xie et al. [175] who have exploited the GAN as part of the work process of industrial machines. The authors developed an algorithm that combines GANs with CNNs to simulate the original distribution from minority classes and generate new data to solve the imbalance problem. Zhong et al. [176] exploited GANs for fault diagnosis of air handling units for residential buildings.

Zhao et al. [177] exploited a VAE model to generate additional vibration signals using the hidden variables sampled from the Gaussian distribution. These signals are mixed with the original signals and used for training a classifier for fault identification. An et al. [178] propose a method of detection of anomalies based on VAE. The method exploits the probability of reconstruction of a variational autoencoder which measures the variability of the distribution of variables. The results of this work tell us that the method allows to derive the reconstruction of the data to analyze the underlying cause of the anomaly. San Martin et al. [179] applied VAE for the diagnosis of ball bearing element failures. An unsupervised VAE is applied resulting in reduced dimensionality and automatic coding capability. To do this, the latent representations provided by the variational self-encoders are compared with those of the principal component analysis. Kawachi et al. [180] used VAEs for the detection of invisible anomalies. The approach provides for the discrimination between the normal distribution and that relating to the anomaly using the relationship between a set and the complementary set: An unsupervised VAE is transformed into a supervised VAE. Park et al. [181] detected multimodal anomalies of a robot-assisted feeding system by combining VAE and LSTM. The model combines the signals and reconstructs their expected distribution by introducing a variation based on previous progress, thanks to the use of the LSTM. Lee et al. [182] monitored the thin-film transistor liquid crystal visualization process using the VAE, while Wang et al. [183] exploited EVAs for monitoring non-linear processes. Ping et al. [184] applied VAE in the prognostics and health management of rolling bearings. The authors developed an asymmetric feature extraction system based on VAE logarithmic distribution algorithms. Wu et al. [185] applied an AAE-based algorithm for machine anomaly detection. The authors studied this technology to automatically identify the low-dimensional collector embedded in the high-dimensional space of the starting signal.

#### **4. Summary and discussion**

In the previous sections we have analyzed the approaches for fault diagnosis based on Machine Learning most used by the scientific community. Fault detection requires in-depth knowledge of the system which is often not available. The monitoring of the environment with the use of the most modern sensors does not ensure the return of a complete representation of the phenomenon. An efficient fault detection system must therefore manage this lack of information and add missing data through the different techniques available. The diagnostic system is then designed in discrete time and includes a term for the compensation of the effect of non-modeled dynamics and disturbances. The compensation term, in turn, is calculated based on the dynamics of the manipulator and the state estimation error. Furthermore, the modeling of the machine must give us a system capable of

generalizing. The main goal of machine learning is to obtain an algorithm that is performing in the classification of new inputs and not only in the classification of the set of examples used during the learning phase [186]. The ability to perform well on inputs not observed during training returns the generalization capacity of the system. The factors that determine how well a machine learning algorithm is are its ability to obtain a low training error and to keep the difference between training and test error small. These two factors correspond to the two most common problems associated with Machine Learning: overfitting and underfitting. Underfitting occurs when the model is unable to obtain a sufficiently small training error. Overfitting, on the other hand, occurs when the gap between training and test errors is too large [187,188].

One aspect to consider when choosing a method for identifying failures is the optimization procedure. The optimization techniques of the classification models play an important role in the convergence of the models to avoid local optimal. The presence of local minima of the error function constitutes a difficulty to be taken into account in the design of an optimization algorithm for training machine learning based algorithms. The study of the theoretical properties of the error function in relation to local minima has been the subject of various works [189,190]. The calculation experience shows that in many applications the greatest difficulties in the training process are due to the presence of plateaus in the error function rather than that of local minima. Furthermore, the training of an algorithm is not necessarily aimed at identifying a global minimum point of the associated optimization problem but has the purpose of determining a vector of parameters corresponding to a sufficiently low value of the error committed on the samples. of the training set, so that the algorithm has an adequate generalization capacity. For these reasons, the study of specific global optimization algorithms has not been one of the most relevant aspects in the literature. In general, global optimization algorithms are divided into stochastic methods [191] and deterministic methods [192]. The online version of the backpropagation algorithm represents one of the most widely used stochastic methods. Hybrid training strategies have also been proposed, consisting of a stochastic phase followed by one in which a standard optimization method is applied. The reasons for such strategies are, on the one hand, to avoid local minima through the stochastic phase, and on the other hand to accelerate the convergence towards the desired minimum with the second phase.

**Table 1.** ML algorithms features for fault diagnosis.

Model	Strengths	Weaknesses
Support Vector Machine (SVM)	High accuracy, low storage	Slow Computing for big data, noise sensitive
Artificial Neural Network (ANN)	High accuracy, fault tolerance	Physical meaning, high computing cost
Convolutional Neural Network (CNN)	Feature extraction free, efficient for big data	High computing cost, long time training
Recurrent Neural Network (RNN)	Robust to input size, short memory problem free	Computing complexity, higher memory required
Deep Generative Systems (DGM)	Complex structure detection, generalized training	Harder to train, physical meaning

There are many algorithms based on Machine Learning and each family of algorithms has specific characteristics that govern their use in every context. Table 1 summarizes the strengths and weaknesses of each family of algorithms. The different potentials make us understand how the choice of the algorithm depends on the characteristics of the system that needs to be modeled.

From Table 1 the complexity of the model is linked to the characteristics of the input to be processed. Systems with significant input dimensions require more complex modeling tools with an increase in the computational cost. However, this does not tell us that necessarily the most complex choice is the most suitable for the solution of the problem; in fact, it often happens that the performances of the algorithms are different according to different inputs.

Table 2 compares the performances returned by the models adopted by the scientific community for the identification of faults. To facilitate comparison, ranges of values have been reported as declared by the authors in their respective papers. The Accuracy metric was adopted for the performance evaluation. Accuracy measures how close the forecast is to the current value; it is usually indicated as a percentage.

**Table 2.** Performance of different Machine Learning-based models for fault diagnosis.

Model	Min Accuracy (%)	Max Accuracy (%)
Support Vector Machine (SVM)	71.5 [44]	98.5 [38]
Artificial Neural Network (ANN)	85.6 [90]	99.4 [99]
Convolutional Neural Network (CNN)	97.4 [120]	99.8 [110]
Recurrent Neural Network (RNN)	98.5 [155]	99.7 [142]
Deep Generative Systems (DGM)	86.3 [175]	99.8 [171]

Analyzing the table, we can see that the accuracy returned by the models for identifying faults has comparable values. This confirms that the evaluation metric, referable to works available in the literature, is not the right tool to guide the researcher in choosing the most appropriate algorithm for identifying a specific fault. This choice can be made only after verifying how the different algorithms adapt to the available data, providing the system with an adequate generalization capacity. However, a comparison of the results in Table 2 tells some things: The algorithms based on CNN and RNN seem to return results with greater accuracy, this can be justified by the ability to automatically extract the features. This greater ability to extract knowledge is paid for in computational costs which become more expensive.

The review of a substantial number of contributions, which have received the approval of the scientific community in terms of citations, has clearly outlined what the hotspots are and where the future challenges awaiting experts in the sector are headed.

- Machine learning-based methods require reliable data and correct labeling. In the case of fault diagnosis, this availability of information strongly depends on the sensors used for data collection and the labeling process. To improve the performance of these methodologies, it is therefore necessary to invest in the quality of sensors and in the labeling procedure, which requires substantial economic resources. On the other hand, the increasingly widespread availability of low-cost sensors that can be connected to the data network imposes a cost-benefit balance between systems that, using cutting-edge technologies, require excessive costs and systems that, based on economic sensors, return less precise results.

- Automated fault detection systems are highly dependent on feature selection, feature extraction, and data collection. Deep learning has been shown to yield excellent results in the automatic selection of features: This frees the researcher from the onerous task of identifying those features that best highlight the presence of anomalies in the behavior of the machine. On the other hand, the computational cost required by these algorithms, even if partially offset by the increase in performance offered by modern hardware platforms, represents a parameter to be evaluated in the choice of survey methodology.
- Most of the automatic fault identification systems we have analyzed treat the case as a supervised classification problem. The fault diagnosis process could instead be approached as a clustering problem. However, most current studies tend to address the problem by devising a pattern recognition system. In the future, research should be concentrated on developing suitable clustering. The availability of low-cost sensors that can be connected to each other through wireless networks according to modern IoT technologies offers the scientific community the opportunity to develop automatic fault detection that is increasingly within everyone's reach. These systems, therefore, will not only be developed for the detection of faults in the industrial environment but can be extended to local realities up to the needs of the individual user within their own home. Systems of this type can be offered together with home automation systems, offering new functions aimed at improving home security.

## 5. Conclusions

In this review we have analyzed the Machine Learning based methods most widely used by the scientific community for diagnosing machine failures. For each type of methodology, we first provided the background necessary to understand the method and then we analyzed the most representative works that have yielded these techniques to identify failures in the industrial machine. The work carried out highlighted the strong use of these methods which confirms the extreme usefulness of these techniques in identifying failures in scenarios heavily contaminated by residual noise. The automatic extraction of knowledge today represents a valid tool for identifying faults: Technicians who must manage the maintenance of an industrial process must necessarily use these methods for a correct forecast of the mechanical parts to be replaced.

## Conflict of interest

The author declare there is no conflict of interest.

## References

1. A. Muller, A. C. Marquez, B. Iung, On the concept of e-maintenance: Review and current research, *Reliab. Eng. Syst. Saf.*, **93** (2008), 1165–1187. <https://doi.org/10.1016/j.ress.2007.08.006>
2. K. Gandhi, A. H. Ng, Machine maintenance decision support system: a systematic literature review, in *Advances in Manufacturing Technology XXXII: Proceedings of the 16th International Conference on Manufacturing Research, incorporating the 33rd National Conference on Manufacturing Research*, September 11–13, University of Skövde, IOS Press, Sweden, **8** (2018), 349.

3. A. Garg, S. G. Deshmukh, Maintenance management: literature review and directions, *J. Qual. Maint. Eng.*, **12** (2006), 205–238. <https://doi.org/10.1108/13552510610685075>
4. D. Sherwin, A review of overall models for maintenance management, *J. Qual. Maint. Eng.*, **6** (2000), 138–164. <https://doi.org/10.1108/13552510010341171>
5. K. C. Ng, G. G. G. Goh, U. C. Eze, Critical success factors of total productive maintenance implementation: a review, in *2011 IEEE international conference on industrial engineering and engineering management*, IEEE, Singapore, 269–273. <https://doi.org/10.1109/IEEM.2011.6117920>
6. E. Sisinni, A. Saifullah, S. Han, U. Jennehag, M. Gidlund, Industrial internet of things: Challenges, opportunities, and directions, *IEEE Trans. Ind. Inf.*, **14** (2018), 4724–4734. <https://doi.org/10.1109/TII.2018.2852491>
7. H. Boyes, B. Hallaq, J. Cunningham, T. Watson, The industrial internet of things (IIoT): An analysis framework, *Comput. Ind.*, **101** (2018), 1–12. <https://doi.org/10.1016/j.compind.2018.04.015>
8. J. Wan, S. Tang, Z. Shu, D. Li, S. Wang, M. Imran, et al., Software-defined industrial internet of things in the context of industry 4.0, *IEEE Sens. J.*, **16** (2016), 7373–7380. <https://doi.org/10.1109/JSEN.2016.2565621>
9. Y. Liao, E. D. F. R. Loures, F. Deschamps, Industrial Internet of Things: A systematic literature review and insights, *IEEE Internet Things J.*, **5** (2018), 4515–4525. <https://doi.org/10.1109/JIOT.2018.2834151>
10. M. Hartmann, B. Halecker, Management of innovation in the industrial internet of things, in *The International Society for Professional Innovation Management ISPIM Conference Proceedings*, 2015.
11. M. Mohri, A. Rostamizadeh, A. Talwalkar, *Foundations of Machine Learning*, MIT press, 2018.
12. C. Sammut, G. I. Webb, *Encyclopedia of Machine Learning*, Springer Science & Business Media, 2011.
13. G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, et al., Machine learning and the physical sciences, *Rev. Mod. Phys.*, **91** (2019), 045002. <https://doi.org/10.1103/RevModPhys.91.045002>
14. M. Du, N. Liu, X. Hu, Techniques for interpretable machine learning, *Commun. ACM*, **63** (2019), 68–77. <https://doi.org/10.1145/3359786>
15. H. Sahli, An introduction to machine learning, in *TORUS I-Toward an Open Resource Using Services: Cloud Computing for Environmental Data*, (2020), 61–74. <https://doi.org/10.1002/9781119720492.ch7>
16. R. H. P. M. Arts, G. M. Knapp, L. Mann, Some aspects of measuring maintenance performance in the process industry, *J. Qual. Maint. Eng.*, **4** (1998) 6–11. <https://doi.org/10.1108/13552519810201520>
17. C. Stenström, P. Norrbin, A. Parida, U. Kumar, Preventive and corrective maintenance-cost comparison and cost-benefit analysis, *Struct. Infrastruct. Eng.*, **12** (2016), 603–617. <https://doi.org/10.1080/15732479.2015.1032983>
18. H. P. Bahrick, L. K. Hall, Preventive and corrective maintenance of access to knowledge, *Appl. Cognit. Psychol.*, **5** (1991), 1–18. <https://doi.org/10.1002/acp.2350050102>
19. J. Shin, H. Jun, On condition based maintenance policy, *J. Comput. Des. Eng.*, **2** (2015), 119–127. <https://doi.org/10.1016/j.jcde.2014.12.006>

20. R. Ahmad, S. Kamaruddin, An overview of time-based and condition-based maintenance in industrial application, *Comput. Ind. Eng.*, **63** (2012), 135–149. <https://doi.org/10.1016/j.cie.2012.02.002>
21. J. H. Williams, A. Davies, P. R. Drake, *Condition-Based Maintenance and Machine Diagnostics*, Springer Science & Business Media, 1994.
22. R. K. Mobley, *An Introduction to Predictive Maintenance*, 2nd edition, Elsevier, 2002. <https://doi.org/10.1016/B978-0-7506-7531-4.X5000-3>
23. C. Scheffer, P. Girdhar, *Practical Machinery Vibration Analysis and Predictive Maintenance*, Elsevier, 2004.
24. K. Efthymiou, N. Papakostas, D. Mourtzis, G. Chryssolouris, On a predictive maintenance platform for production systems, *Procedia CIRP*, **3** (2012), 221–226. <https://doi.org/10.1016/j.procir.2012.07.039>
25. G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, A. Beghi, Machine learning for predictive maintenance: A multiple classifier approach, *IEEE Trans. Ind. Inf.*, **11** (2014), 812–820. <https://doi.org/10.1109/TII.2014.2349359>
26. R. Isermann, *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, Springer Science & Business Media, 2005.
27. Z. Gao, C. Cecati, S. X. Ding, A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches, *IEEE Trans. Ind. Electron.*, **62** (2015), 3757–3767. <https://doi.org/10.1109/TIE.2015.2417501>
28. S. Leonhardt, M. Ayoubi, Methods of fault diagnosis, *Control Eng. Pract.*, **5** (1997), 683–692. [https://doi.org/10.1016/S0967-0661\(97\)00050-6](https://doi.org/10.1016/S0967-0661(97)00050-6)
29. R. J. Patton, P. M. Frank, R. N. Clark, *Issues of Fault Diagnosis for Dynamic Systems*, Springer Science & Business Media, 2013.
30. M. I. Jordan, T. M. Mitchell, Machine learning: Trends, perspectives, and prospects, *Science*, **349** (2015), 255–260. <https://doi.org/10.1126/science.aaa8415>
31. U. S. Shanthamallu, A. Spanias, C. Tepedelenlioglu, M. Stanley, A brief survey of machine learning methods and their sensor and IoT applications, in *2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, IEEE, (2017), 1–8. <https://doi.org/10.1109/IISA.2017.8316459>
32. D. A. Pisner, D. M. Schnyer, Support vector machine, in *Machine Learning*, Academic Press, (2020), 101–121. <https://doi.org/10.1016/B978-0-12-815739-8.00006-7>
33. W. S. Noble, What is a support vector machine, *Nat. Biotechnol.*, **24** (2006), 1565–1567. <https://doi.org/10.1038/nbt1206-1565>
34. L. Wang, *Support Vector Machines: Theory and Applications*, Springer Science & Business Media, 2005. <https://doi.org/10.1007/b95439>
35. S. I. Amari, S. Wu, Improving support vector machine classifiers by modifying kernel functions, *Neural Networks*, **12** (1999), 783–789. [https://doi.org/10.1016/S0893-6080\(99\)00032-5](https://doi.org/10.1016/S0893-6080(99)00032-5)
36. O. L. Mangasarian, D. R. Musicant, Lagrangian support vector machines, *J. Mach. Learn. Res.*, **1** (2001), 161–177.
37. A. Widodo, B. S. Yang, Support vector machine in machine condition monitoring and fault diagnosis, *Mech. Syst. Sig. Process.*, **21** (2007), 2560–2574. <https://doi.org/10.1016/j.ymsp.2006.12.007>

38. S. W. Fei, X. B. Zhang, Fault diagnosis of power transformer based on support vector machine with genetic algorithm, *Expert Syst. Appl.*, **36** (2009), 11352–11357. <https://doi.org/10.1016/j.eswa.2009.03.022>
39. S. D. Wu, P. H. Wu, C. W. Wu, J. J. Ding, C. C. Wang, Bearing fault diagnosis based on multiscale permutation entropy and support vector machine, *Entropy*, **14** (2012), 1343–1356. <https://doi.org/10.3390/e14081343>
40. W. Aziz, M. Arif, Multiscale permutation entropy of physiological time series, in *2005 Pakistan Section Multitopic Conference*, IEEE, (2005), 1–6. <https://doi.org/10.1109/INMIC.2005.334494>
41. B. Tang, T. Song, F. Li, L. Deng, Fault diagnosis for a wind turbine transmission system based on manifold learning and Shannon wavelet support vector machine, *Renewable Energy*, **62** (2014), 1–9. <https://doi.org/10.1016/j.renene.2013.06.025>
42. Z. Wang, L. Yao, Y. Cai, J. Zhang, Mahalanobis semi-supervised mapping and beetle antennae search based support vector machine for wind turbine rolling bearings fault diagnosis, *Renewable Energy*, **155** (2020), 1312–1327. <https://doi.org/10.1016/j.renene.2020.04.041>
43. L. Yao, Z. Fang, Y. Xiao, J. Hou, Z. Fu, An intelligent fault diagnosis method for lithium battery systems based on grid search support vector machine, *Energy*, **214** (2021), 118866. <https://doi.org/10.1016/j.energy.2020.118866>
44. Y. P. Zhao, J. J. Wang, X. Y. Li, G. J. Peng, Z. Yang, Extended least squares support vector machine with applications to fault diagnosis of aircraft engine, *ISA Trans.*, **97** (2020), 189–201. <https://doi.org/10.1016/j.isatra.2019.08.036>
45. F. Marini, B. Walczak, Particle swarm optimization (PSO). A tutorial, *Chemom. Intell. Lab. Syst.*, **149** (2015), 153–165. <https://doi.org/10.1016/j.chemolab.2015.08.020>
46. M. Van, D. T. Hoang, H. J. Kang, Bearing fault diagnosis using a particle swarm optimization-least squares wavelet support vector machine classifier, *Sensors*, **20** (2020), 3422. <https://doi.org/10.3390/s20123422>
47. X. Li, S. Wu, X. Li, H. Yuan, D. Zhao, Particle swarm optimization-support vector machine model for machinery fault diagnoses in high-voltage circuit breakers, *Chin. J. Mech. Eng.*, **33** (2020), 1–10. <https://doi.org/10.1186/s10033-019-0428-5>
48. Y. Fan, C. Zhang, Y. Xue, J. Wang, F. Gu, A bearing fault diagnosis using a support vector machine optimised by the self-regulating particle swarm, *Shock Vib.*, **2020** (2020). <https://doi.org/10.1155/2020/9096852>
49. E. Mirakhorli, Fault diagnosis in a distillation column using a support vector machine based classifier, *Int. J. Smart Electr. Eng.*, **8** (2020), 105–113.
50. S. Gao, C. Zhou, Z. Zhang, J. Geng, R. He, Q. Yin, C. Xing, Mechanical fault diagnosis of an on-load tap changer by applying cuckoo search algorithm-based fuzzy weighted least squares support vector machine, *Math. Probl. Eng.*, **2020** (2020). <https://doi.org/10.1155/2020/3432409>
51. X. Huang, X. Huang, B. Wang, Z. Xie, Fault diagnosis of transformer based on modified grey wolf optimization algorithm and support vector machine, *IEEEJ Trans. Electr. Electron. Eng.*, **15** (2020), 409–417. <https://doi.org/10.1002/tee.23069>
52. Y. Zhang, J. Li, X. Fan, J. Liu, H. Zhang, Moisture prediction of transformer oil-immersed polymer insulation by applying a support vector machine combined with a genetic algorithm, *Polymers*, **12** (2020), 1579. <https://doi.org/10.3390/polym12071579>



53. Y. Liu, H. Chen, L. Zhang, X. Wu, X. J. Wang, Energy consumption prediction and diagnosis of public buildings based on support vector machine learning: A case study in China, *J. Cleaner Prod.*, **272** (2020), 122542. <https://doi.org/10.1016/j.jclepro.2020.122542>
54. S. K. Ibrahim, A. Ahmed, M. A. E. Zeidan, I. E. Ziedan, Machine learning techniques for satellite fault diagnosis, *Ain Shams Eng. J.*, **11** (2020), 45–56. <https://doi.org/10.1016/j.asej.2019.08.006>
55. Y. P. Zhao, G. Huang, Q. K. Hu, B. Li, An improved weighted one class support vector machine for turboshaft engine fault detection, *Eng. Appl. Artif. Intell.*, **94** (2020), 103796. <https://doi.org/10.1016/j.engappai.2020.103796>
56. M. Guo, L. Xie, S. Q. Wang, J. M. Zhang, Research on an integrated ICA-SVM based framework for fault diagnosis, in *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme-System Security and Assurance (Cat. No. 03CH37483)*, IEEE, **3** (2003), 2710–2715. <https://doi.org/10.1109/ICSMC.2003.1244294>
57. S. Poyhonen, P. Jover, H. Hyotyniemi, Signal processing of vibrations for condition monitoring of an induction motor, in *First International Symposium on Control, Communications and Signal Processing*, IEEE, Tunisia, (2004), 499–502. <https://doi.org/10.1109/ISCCSP.2004.1296338>
58. M. C. Moura, E. Zio, I. D. Lins, E. Droguett, Failure and reliability prediction by support vector machines regression of time series data, *Reliab. Eng. Syst. Saf.*, **96** (2011), 1527–1534. <https://doi.org/10.1016/j.res.2011.06.006>
59. K. Y. Chen, L. S. Chen, M. C. Chen, C. L. Lee, Using SVM based method for equipment fault detection in a thermal power plant, *Comput. Ind.*, **62** (2011), 42–50. <https://doi.org/10.1016/j.compind.2010.05.013>
60. K. He, X. Li, A quantitative estimation technique for welding quality using local mean decomposition and support vector machine, *J. Intell. Manuf.*, **27** (2016), 525–533. <https://doi.org/10.1007/s10845-014-0885-8>
61. K. Yan, C. Zhong, Z. Ji, J. Huang, Semi-supervised learning for early detection and diagnosis of various air handling unit faults, *Energy Build.*, **181** (2018), 75–83. <https://doi.org/10.1016/j.enbuild.2018.10.016>
62. Z. Yin, J. Hou, Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes, *Neurocomputing*, **174** (2016), 643–650. <https://doi.org/10.1016/j.neucom.2015.09.081>
63. M. M. Islam, J. M. Kim, Reliable multiple combined fault diagnosis of bearings using heterogeneous feature models and multiclass support vector Machines, *Reliab. Eng. Syst. Saf.*, **184** (2019), 55–66. <https://doi.org/10.1016/j.res.2018.02.012>
64. R. P. Monteiro, M. Cerrada, D. R. Cabrera, R. V. Sánchez, C. J. Bastos-Filho, Using a support vector machine based decision stage to improve the fault diagnosis on gearboxes, *Comput. Intell. Neurosci.*, **2019** (2019). <https://doi.org/10.1155/2019/1383752>
65. D. Yang, J. Miao, F. Zhang, J. Tao, G. Wang, Y. Shen, Bearing fault diagnosis using a support vector machine optimized by an improved ant lion optimizer, *Shock Vib.*, **2019** (2019). <https://doi.org/10.1155/2019/9303676>
66. S. Mirjalili, The ant lion optimizer, *Adv. Eng. Software*, **83** (2015), 80–98. <https://doi.org/10.1016/j.advengsoft.2015.01.010>
67. L. You, W. Fan, Z. Li, Y. Liang, M. Fang, J. Wang, A fault diagnosis model for rotating machinery using VWC and MSFLA-SVM based on vibration signal analysis, *Shock Vib.*, **2019** (2019). <https://doi.org/10.1155/2019/1908485>

68. A. Kumar, R. Kumar, Time-frequency analysis and support vector machine in automatic detection of defect from vibration signal of centrifugal pump, *Measurement*, **108** (2017), 119–133. <https://doi.org/10.1016/j.measurement.2017.04.041>
69. Z. Chen, F. Zhao, J. Zhou, P. Huang, X. Zhang, Fault diagnosis of loader gearbox based on an Ica and SVM algorithm, *Int. J. Environ. Res. Public Health*, **16** (2019), 4868. <https://doi.org/10.3390/ijerph16234868>
70. T. W. Lee, Independent component analysis, in *Independent Component Analysis*, Springer, Boston, (1998), 27–66. [https://doi.org/10.1007/978-1-4757-2851-4\\_2](https://doi.org/10.1007/978-1-4757-2851-4_2)
71. W. Liu, Z. Wang, J. Han, G. Wang, Wind turbine fault diagnosis method based on diagonal spectrum and clustering binary tree SVM, *Renewable Energy*, **50** (2013), 1–6. <https://doi.org/10.1016/j.renene.2012.06.013>
72. M. A. Djeziri, O. Djedidi, N. Morati, J. L. Seguin, M. Bendahan, T. Contaret, A temporal-based SVM approach for the detection and identification of pollutant gases in a gas mixture, *Appl. Intell.*, **52** (2022), 6065–6078. <https://doi.org/10.1007/s10489-021-02761-0>
73. G. Ciaburro, G. Iannace, J. Passaro, A. Bifulco, D. Marano, M. Guida, et al., Artificial neural network-based models for predicting the sound absorption coefficient of electrospun poly (vinyl pyrrolidone)/silica composite, *Appl. Acoust.*, **169** (2020), 107472. <https://doi.org/10.1016/j.apacoust.2020.107472>
74. S. Agatonovic-Kustrin, R. Beresford, Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research, *J. Pharm. Biomed. Anal.*, **22** (2000), 717–727. [https://doi.org/10.1016/S0731-7085\(99\)00272-1](https://doi.org/10.1016/S0731-7085(99)00272-1)
75. G. Ciaburro, G. Iannace, M. Ali, A. Alabdulkarem, A. Nuhait, An artificial neural network approach to modelling absorbent asphalts acoustic properties, *J. King Saud Univ. Eng. Sci.*, **33** (2021), 213–220. <https://doi.org/10.1016/j.jksues.2020.07.002>
76. J. Misra, I. Saha, Artificial neural networks in hardware: A survey of two decades of progress, *Neurocomputing*, **74** (2010), 239–255. <https://doi.org/10.1016/j.neucom.2010.03.021>
77. Z. Zhang, K. Friedrich, Artificial neural networks applied to polymer composites: a review, *Compos. Sci. Technol.*, **63** (2003), 2029–2044. [https://doi.org/10.1016/S0266-3538\(03\)00106-4](https://doi.org/10.1016/S0266-3538(03)00106-4)
78. G. Iannace, G. Ciaburro, A. Trematerra, Modelling sound absorption properties of broom fibers using artificial neural networks, *Appl. Acoust.*, **163** (2020), 107239. <https://doi.org/10.1016/j.apacoust.2020.107239>
79. K. P. Singh, A. Basant, A. Malik, G. Jain, Artificial neural network modeling of the river water quality—a case study, *Ecol. Modell.*, **220** (2009), 888–895. <https://doi.org/10.1016/j.ecolmodel.2009.01.004>
80. H. Zhu, X. Li, Q. Sun, L. Nie, J. Yao, G. Zhao, A power prediction method for photovoltaic power plant based on wavelet decomposition and artificial neural networks, *Energies*, **9** (2015), 1–15. <https://doi.org/10.3390/en9010011>
81. V. P. Romero, L. Maffei, G. Brambilla, G. Ciaburro, Modelling the soundscape quality of urban waterfronts by artificial neural networks, *Appl. Acoust.*, **111** (2016), 121–128. <https://doi.org/10.1016/j.apacoust.2016.04.019>
82. S. Fabio, D. N. Giovanni, P. Mariano, Airborne sound insulation prediction of masonry walls using artificial neural networks, *Build. Acoust.*, **28** (2021), 391–409. <https://doi.org/10.1177/1351010X21994462>

83. Y. Zhang, X. Ding, Y. Liu, P. J. Griffin, An artificial neural network approach to transformer fault diagnosis, *IEEE Trans. Power Delivery*, **11** (1996), 1836–1841. <https://doi.org/10.1109/61.544265>
84. J. C. Hoskins, K. M. Kaliyur, D. M. Himmelblau, Fault diagnosis in complex chemical plants using artificial neural networks, *AIChE J.*, **37** (1991), 137–141. <https://doi.org/10.1002/aic.690370112>
85. J. B. Ali, N. Fnaiech, L. Saidi, B. Chebel-Morello, F. Fnaiech, Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals, *Appl. Acoust.*, **89** (2015), 16–27. <https://doi.org/10.1016/j.apacoust.2014.08.016>
86. T. Sorsa, H. N. Koivo, Application of artificial neural networks in process fault diagnosis, *Automatica*, **29** (1993), 843–849. [https://doi.org/10.1016/0005-1098\(93\)90090-G](https://doi.org/10.1016/0005-1098(93)90090-G)
87. N. Saravanan, K. I. Ramachandran, Incipient gear box fault diagnosis using discrete wavelet transform (DWT) for feature extraction and classification using artificial neural network (ANN), *Expert Syst. Appl.*, **37** (2010), 4168–4181. <https://doi.org/10.1016/j.eswa.2009.11.006>
88. W. Chine, A. Mellit, V. Lughi, A. Malek, G. Sulligoi, A. M. Pavan, A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks, *Renewable Energy*, **90** (2016), 501–512. <https://doi.org/10.1016/j.renene.2016.01.036>
89. B. Li, M. Y. Chow, Y. Tipsuwan, J. C. Hung, Neural-network-based motor rolling bearing fault diagnosis, *IEEE Trans. Ind. Electron.*, **47** (2000), 1060–1069. <https://doi.org/10.1109/41.873214>
90. B. Samanta, K. R. Al-Balushi, S. A. Al-Araimi, Artificial neural networks and genetic algorithm for bearing fault detection, *Soft Comput.*, **10** (2006), 264–271. <https://doi.org/10.1007/s00500-005-0481-0>
91. T. Han, B. S. Yang, W. H. Choi, J. S. Kim, Fault diagnosis system of induction motors based on neural network and genetic algorithm using stator current signals, *Int. J. Rotating Mach.*, **2006** (2006). <https://doi.org/10.1155/IJRM/2006/61690>
92. H. Wang, P. Chen, Intelligent diagnosis method for rolling element bearing faults using possibility theory and neural network, *Comput. Ind. Eng.*, **60** (2011), 511–518. <https://doi.org/10.1016/j.cie.2010.12.004>
93. M. A. Hashim, M. H. Nasef, A. E. Kabeel, N. M. Ghazaly, Combustion fault detection technique of spark ignition engine based on wavelet packet transform and artificial neural network, *Alexandria Eng. J.*, **59** (2020), 3687–3697. <https://doi.org/10.1016/j.aej.2020.06.023>
94. G. Iannace, G. Ciaburro, A. Trematerra, Fault diagnosis for UAV blades using artificial neural network, *Robotics*, **8** (2019), 59. <https://doi.org/10.3390/robotics8030059>
95. M. Kordestani, M. F. Samadi, M. Saif, K. Khorasani, A new fault diagnosis of multifunctional spoiler system using integrated artificial neural network and discrete wavelet transform methods, *IEEE Sens. J.*, **18** (2018), 4990–5001. <https://doi.org/10.1109/JSEN.2018.2829345>
96. S. Shi, G. Li, H. Chen, J. Liu, Y. Hu, L. Xing, et al., Refrigerant charge fault diagnosis in the VRF system using Bayesian artificial neural network combined with ReliefF filter, *Appl. Therm. Eng.*, **112** (2017), 698–706. <https://doi.org/10.1016/j.applthermaleng.2016.10.043>
97. X. Xu, D. Cao, Y. Zhou, J. Gao, Application of neural network algorithm in fault diagnosis of mechanical intelligence, *Mech. Syst. Sig. Process.*, **141** (2020), 106625. <https://doi.org/10.1016/j.ymsp.2020.106625>

98. A. Viveros-Wacher, J. E. Rayas-Sánchez, Analog fault identification in RF circuits using artificial neural networks and constrained parameter extraction, in *2018 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, IEEE, (2018), 1–3. <https://doi.org/10.1109/NEMO.2018.8503117>
99. S. Heo, J. H. Lee, Fault detection and classification using artificial neural networks, *IFAC-PapersOnLine*, **51** (2018), 470–475. <https://doi.org/10.1016/j.ifacol.2018.09.380>
100. P. Agrawal, P. Jayaswal, Diagnosis and classifications of bearing faults using artificial neural network and support vector machine, *J. Inst. Eng. (India): Ser. C*, **101** (2020), 61–72. <https://doi.org/10.1007/s40032-019-00519-9>
101. Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, et al., Handwritten digit recognition with a back-propagation network, in *Advances in Neural Information Processing Systems*, (1990), 396–404.
102. T. Chen, Y. Sun, T. H. Li, A semi-parametric estimation method for the quantile spectrum with an application to earthquake classification using convolutional neural network, *Comput. Stat. Data Anal.*, **154** (2021), 107069. <https://doi.org/10.1016/j.csda.2020.107069>
103. F. Perla, R. Richman, S. Scognamiglio, M. V. Wüthrich, Time-series forecasting of mortality rates using deep learning, *Scand. Actuarial J.*, **2021** (2021), 1–27. <https://doi.org/10.1080/03461238.2020.1867232>
104. G. Ciaburro, G. Iannace, V. Puyana-Romero, A. Trematerra, A comparison between numerical simulation models for the prediction of acoustic behavior of giant reeds shredded, *Appl. Sci.*, **10** (2020), 6881. <https://doi.org/10.3390/app10196881>
105. C. Yildiz, H. Acikgoz, D. Korkmaz, U. Budak, An improved residual-based convolutional neural network for very short-term wind power forecasting, *Energy Convers. Manage.*, **228** (2021), 113731. <https://doi.org/10.1016/j.enconman.2020.113731>
106. G. Ciaburro, Sound event detection in underground parking garage using convolutional neural network, *Big Data Cognit. Comput.*, **4** (2020), 20. <https://doi.org/10.3390/bdcc4030020>
107. R. Ye, Q. Dai, Implementing transfer learning across different datasets for time series forecasting, *Pattern Recognit.*, **109** (2021), 107617. <https://doi.org/10.1016/j.patcog.2020.107617>
108. J. Han, L. Shi, Q. Yang, K. Huang, Y. Zha, J. Yu, Real-time detection of rice phenology through convolutional neural network using handheld camera images, *Precis. Agric.*, **22** (2021), 154–178. <https://doi.org/10.1016/j.patcog.2020.107617>
109. G. Ciaburro, G. Iannace, Improving smart cities safety using sound events detection based on deep neural network algorithms, *Informatics*, **7** (2020), 23. <https://doi.org/10.3390/informatics7030023>
110. L. Wen, X. Li, L. Gao, Y. Zhang, A new convolutional neural network-based data-driven fault diagnosis method, *IEEE Trans. Ind. Electron.*, **65** (2017), 5990–5998. <https://doi.org/10.1109/TIE.2017.2774777>
111. Y. LeCun, LeNet-5, Convolutional Neural Networks, 2015, Available from: <http://yann.lecun.com/exdb/lenet/>, Accessed date: 28 April 2022.
112. H. Wu, J. Zhao, Deep convolutional neural network model based chemical process fault diagnosis, *Comput. Chem. Eng.*, **115** (2018), 185–197. <https://doi.org/10.1016/j.compchemeng.2018.04.009>
113. W. Zhang, C. Li, G. Peng, Y. Chen, Z. Zhang, A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load, *Mech. Syst. Sig. Process.*, **100** (2018), 439–453. <https://doi.org/10.1016/j.ymsp.2017.06.022>

114. L. Jing, M. Zhao, P. Li, X. Xu, A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox, *Measurement*, **111** (2017), 1–10. <https://doi.org/10.1016/j.measurement.2017.07.017>
115. Z. Chen, C. Li, R. V. Sanchez, Gearbox fault identification and classification with convolutional neural networks, *Shock Vib.*, **2015** (2015). <https://doi.org/10.1155/2015/390134>
116. X. Guo, L. Chen, C. Shen, Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis, *Measurement*, **93** (2016), 490–502. <https://doi.org/10.1016/j.measurement.2016.07.054>
117. O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, et al., Convolutional neural network based fault detection for rotating machinery, *J. Sound Vib.*, **377** (2016), 331–345. <https://doi.org/10.1016/j.jsv.2016.05.027>
118. W. Zhang, G. Peng, C. Li, Y. Chen, Z. Zhang, A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals, *Sensors*, **17** (2017), 425. <https://doi.org/10.3390/s17020425>
119. Y. Li, N. Wang, J. Shi, X. Hou, J. Liu, Adaptive batch normalization for practical domain adaptation, *Pattern Recognit.*, **80** (2018), 109–117. <https://doi.org/10.1016/j.patcog.2018.03.005>
120. T. Ince, S. Kiranyaz, L. Eren, M. Askar, M. Gabbouj, Real-time motor fault detection by 1-D convolutional neural networks, *IEEE Trans. Ind. Electron.*, **63** (2016), 7067–7075. <https://doi.org/10.1109/TIE.2016.2582729>
121. Y. Zhang, K. Xing, R. Bai, D. Sun, Z. Meng, An enhanced convolutional neural network for bearing fault diagnosis based on time-frequency image, *Measurement*, **157** (2020), 107667. <https://doi.org/10.1016/j.measurement.2020.107667>
122. M. Azamfar, J. Singh, I. Bravo-Imaz, J. Lee., Multisensor data fusion for gearbox fault diagnosis using 2-D convolutional neural network and motor current signature analysis, *Mech. Syst. Sig. Process.*, **144** (2020), 106861. <https://doi.org/10.1016/j.ymsp.2020.106861>
123. Q. Zhou, Y. Li, Y. Tian, L. Jiang, A novel method based on nonlinear auto-regression neural network and convolutional neural network for imbalanced fault diagnosis of rotating machinery, *Measurement*, **161** (2020), 107880. <https://doi.org/10.1016/j.measurement.2020.107880>
124. K. Zhang, J. Chen, T. Zhang, Z. Zhou, A compact convolutional neural network augmented with multiscale feature extraction of acquired monitoring data for mechanical intelligent fault diagnosis, *J. Manuf. Syst.*, **55** (2020), 273–284. <https://doi.org/10.1016/j.jmsy.2020.04.016>
125. Y. Li, X. Du, F. Wan, X. Wang, H. Yu, Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging, *Chin. J. Aeronaut.*, **33** (2020), 427–438. <https://doi.org/10.1016/j.cja.2019.08.014>
126. Z. Chen, A. Mauricio, W. Li, K. Gryllias, A deep learning method for bearing fault diagnosis based on cyclic spectral coherence and convolutional neural networks, *Mech. Syst. Sig. Process.*, **140** (2020), 106683. <https://doi.org/10.1016/j.ymsp.2020.106683>
127. J. Antoni, Cyclic spectral analysis in practice, *Mech. Syst. Sig. Process.*, **21** (2007), 597–630. <https://doi.org/10.1016/j.ymsp.2006.08.007>
128. D. Zhou, Q. Yao, H. Wu, S. Ma, H. Zhang, Fault diagnosis of gas turbine based on partly interpretable convolutional neural networks, *Energy*, **200** (2020), 117467. <https://doi.org/10.1016/j.energy.2020.117467>
129. T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, Xgboost: extreme gradient boosting, *R package version 0.4-2*, **1** (2015), 1–4.

130. X. Li, J. Zheng, M. Li, W. Ma, Y. Hu, Frequency-domain fusing convolutional neural network: A unified architecture improving effect of domain adaptation for fault diagnosis, *Sensors*, **21** (2021), 450. <https://doi.org/10.3390/s21020450>
131. C. C. Chen, Z. Liu, G. Yang, C. C. Wu, Q. Ye, An improved fault diagnosis using 1D-convolutional neural network model, *electronics*, **10** (2021), 59. <https://doi.org/10.3390/electronics10010059>
132. Y. Liu, Y. Yang, T. Feng, Y. Sun, X. Zhang, Research on rotating machinery fault diagnosis method based on energy spectrum matrix and adaptive convolutional neural network, *Processes*, **9** (2021), 69. <https://doi.org/10.3390/pr9010069>
133. D. T. Hoang, X. T. Tran, M. Van, H. J. Kang, A deep neural network-based feature fusion for bearing fault diagnosis, *Sensors*, **21** (2021), 244. <https://doi.org/10.3390/s21010244>
134. T. Mikolov, M. Karafiát, L. Burget, J. Černocký, S. Khudanpur, Recurrent neural network based language model, in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.
135. K. Gregor, I. Danihelka, A. Graves, D. Rezende, D. Wierstra, Draw: A recurrent neural network for image generation, in *International Conference on Machine Learning (PMLR)*, **37** (2015), 1462–1471.
136. T. Mikolov, G. Zweig, Context dependent recurrent neural network language model, in *2012 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, (2012), 234–239. <https://doi.org/10.1109/SLT.2012.6424228>
137. G. Ciaburro, Time series data analysis using deep learning methods for smart cities monitoring, in *Big Data Intelligence for Smart Applications*, Springer, Cham, (2022), 93–116. [https://doi.org/10.1007/978-3-030-87954-9\\_4](https://doi.org/10.1007/978-3-030-87954-9_4)
138. H. Sak, A. W. Senior, F. Beaufays, Long short-term memory recurrent neural network architectures for large scale acoustic modeling, *Interspeech*, (2014), 338–342. <https://doi.org/10.21437/Interspeech.2014-80>
139. J. Kim, J. Kim, H. L. T. Thu, H. Kim, Long short term memory recurrent neural network classifier for intrusion detection, in *2016 International Conference on Platform Technology and Service (PlatCon)*, IEEE, (2016), 1–5. <https://doi.org/10.1109/PlatCon.2016.7456805>
140. Y. Tian, L. Pan, Predicting short-term traffic flow by long short-term memory recurrent neural network, in *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, IEEE, (2015), 153–158. <https://doi.org/10.1109/SmartCity.2015.63>
141. H. Jiang, X. Li, H. Shao, K. Zhao, Intelligent fault diagnosis of rolling bearings using an improved deep recurrent neural network, *Meas. Sci. Technol.*, **29** (2018), 065107. <https://doi.org/10.1088/1361-6501/aab945>
142. T. De Bruin, K. Verbert, R. Babuška, Railway track circuit fault diagnosis using recurrent neural networks, *IEEE Trans. Neural Networks Learn. Syst.*, **28** (2016), 523–533. <https://doi.org/10.1109/TNNLS.2016.2551940>
143. R. Yang, M. Huang, Q. Lu, M. Zhong, Rotating machinery fault diagnosis using long-short-term memory recurrent neural network, *IFAC-PapersOnLine*, **51** (2018), 228–232. <https://doi.org/10.1016/j.ifacol.2018.09.582>
144. H. A. Talebi, K. Khorasani, S. Tafazoli, A recurrent neural-network-based sensor and actuator fault detection and isolation for nonlinear systems with application to the satellite's attitude control subsystem, *IEEE Trans. Neural Networks*, **20** (2008), 45–60. <https://doi.org/10.1109/TNN.2008.2004373>

145. S. Zhang, K. Bi, T. Qiu, Bidirectional recurrent neural network-based chemical process fault diagnosis, *Ind. Eng. Chem. Res.*, **59** (2019), 824–834. <https://doi.org/10.1021/acs.iecr.9b05885>
146. Z. An, S. Li, J. Wang, X. Jiang, A novel bearing intelligent fault diagnosis framework under time-varying working conditions using recurrent neural network, *ISA Trans.*, **100** (2020), 155–170. <https://doi.org/10.1016/j.isatra.2019.11.010>
147. W. Liu, P. Guo, L. Ye, A low-delay lightweight recurrent neural network (LLRNN) for rotating machinery fault diagnosis, *Sensors*, **19** (2019), 3109. <https://doi.org/10.3390/s19143109>
148. K. Liang, N. Qin, D. Huang, Y. Fu, Convolutional recurrent neural network for fault diagnosis of high-speed train bogie, *Complexity*, **2018** (2018). <https://doi.org/10.1155/2018/4501952>
149. D. Huang, Y. Fu, N. Qin, S. Gao, Fault diagnosis of high-speed train bogie based on LSTM neural network, *Sci. Chin. Inf. Sci.*, **64** (2021), 1–3. <https://doi.org/10.1007/s11432-018-9543-8>
150. H. Shahnazari, P. Mhaskar, J. M. House, T. I. Salsbury, Modeling and fault diagnosis design for HVAC systems using recurrent neural networks, *Comput. Chem. Eng.*, **126** (2019), 189–203. <https://doi.org/10.1016/j.compchemeng.2019.04.011>
151. H. Shahnazari, Fault diagnosis of nonlinear systems using recurrent neural networks, *Chem. Eng. Res. Des.*, **153** (2020), 233–245. <https://doi.org/10.1016/j.cherd.2019.09.026>
152. L. Guo, N. Li, F. Jia, Y. Lei, J. Lin, A recurrent neural network based health indicator for remaining useful life prediction of bearings, *Neurocomputing*, **240** (2017), 98–109. <https://doi.org/10.1016/j.neucom.2017.02.045>
153. M. Yuan, Y. Wu, L. Lin, Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network, in *2016 IEEE international conference on aircraft utility systems (AUS)*, IEEE, (2016), 135–140. <https://doi.org/10.1109/AUS.2016.7748035>
154. Z. Wu, H. Jiang, K. Zhao, X. Li, An adaptive deep transfer learning method for bearing fault diagnosis, *Measurement*, **151** (2020), 107227. <https://doi.org/10.1016/j.measurement.2019.107227>
155. A. Yin, Y. Yan, Z. Zhang, C. Li, R. V. Sánchez, Fault diagnosis of wind turbine gearbox based on the optimized LSTM neural network with cosine loss, *Sensors*, **20** (2020), 2339. <https://doi.org/10.3390/s20082339>
156. M. Xia, X. Zheng, M. Imran, M. Shoaib, Data-driven prognosis method using hybrid deep recurrent neural network, *Appl. Soft Comput.*, **93** (2020), 106351. <https://doi.org/10.1016/j.asoc.2020.106351>
157. Z. Wang, Y. Dong, W. Liu, Z. Ma, A novel fault diagnosis approach for chillers based on 1-D convolutional neural network and gated recurrent unit, *Sensors*, **20** (2020), 2458. <https://doi.org/10.3390/s20092458>
158. R. Salakhutdinov, Learning deep generative models, *Annu. Rev. Stat. Appl.*, **2** (2015), 361–385. <https://doi.org/10.1146/annurev-statistics-010814-020120>
159. A. Gupta, A. Agarwal, P. Singh, P. Rai, A deep generative framework for paraphrase generation, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **32** (2018). <https://doi.org/10.1609/aaai.v32i1.11956>
160. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al., Generative adversarial networks, 2014, preprint, arXiv:1406.2661.
161. L. Metz, B. Poole, D. Pfau, J. Sohl-Dickstein, Unrolled generative adversarial networks, 2016, preprint, arXiv:1611.02163.

162. G. Ciaburro, Security systems for smart cities based on acoustic sensors and machine learning applications, in *Machine Intelligence and Data Analytics for Sustainable Future Smart Cities*, Springer, Cham, (2021), 369–393. [https://doi.org/10.1007/978-3-030-72065-0\\_20](https://doi.org/10.1007/978-3-030-72065-0_20)
163. X. Hou, L. Shen, K. Sun, G. Qiu, Deep feature consistent variational autoencoder, in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, (2017), 1133–1141. <https://doi.org/10.1109/WACV.2017.131>
164. M. J. Kusner, B. Paige, J. M. Hernández-Lobato, Grammar variational autoencoder, in *International Conference on Machine Learning (PMLR)*, **70** (2017), 1945–1954.
165. Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, et al., Variational autoencoder for deep learning of images, labels and captions, 2016, preprint, arXiv:1609.08976.
166. A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, B. Frey, Adversarial autoencoders, 2015, preprint, arXiv:1511.05644.
167. Z. Zhang, Y. Song, H. Qi, Age progression/regression by conditional adversarial autoencoder, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2017), 5810–5818. <https://doi.org/10.1109/CVPR.2017.463>
168. H. Liu, J. Zhou, Y. Xu, Y. Zheng, X. Peng, W. Jiang, Unsupervised fault diagnosis of rolling bearings using a deep neural network based on generative adversarial networks, *Neurocomputing*, **315** (2018), 412–424. <https://doi.org/10.1016/j.neucom.2018.07.034>
169. S. Shao, P. Wang, R. Yan, Generative adversarial networks for data augmentation in machine fault diagnosis, *Comput. Ind.*, **106** (2019), 85–93. <https://doi.org/10.1016/j.compind.2019.01.001>
170. W. Zhang, X. Li, X. D. Jia, H. Ma, Z. Luo, X. Li, Machinery fault diagnosis with imbalanced data using deep generative adversarial networks, *Measurement*, **152** (2020), 107377. <https://doi.org/10.1016/j.measurement.2019.107377>
171. Z. Wang, J. Wang, Y. Wang, An intelligent diagnosis scheme based on generative adversarial learning deep neural networks and its application to planetary gearbox fault pattern recognition, *Neurocomputing*, **310** (2018), 213–222. <https://doi.org/10.1016/j.neucom.2018.05.024>
172. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P. A. Manzagol, L. Bottou, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *J. Mach. Learn. Res.*, **11** (2010), 3371–3408.
173. Q. Li, L. Chen, C. Shen, B. Yang, Z. Zhu, Enhanced generative adversarial networks for fault diagnosis of rotating machinery with imbalanced data, *Meas. Sci. Technol.*, **30** (2019), 115005. <https://doi.org/10.1088/1361-6501/ab3072>
174. J. Wang, S. Li, B. Han, Z. An, H. Bao, S. Ji, Generalization of deep neural networks for imbalanced fault classification of machinery using generative adversarial networks, *IEEE Access*, **7** (2019), 111168–111180. <https://doi.org/10.1109/ACCESS.2019.2924003>
175. Y. Xie, T. Zhang, Imbalanced learning for fault diagnosis problem of rotating machinery based on generative adversarial networks, in *2018 37th Chinese Control Conference (CCC)*, IEEE, (2018), 6017–6022. <https://doi.org/10.23919/ChiCC.2018.8483334>
176. C. Zhong, K. Yan, Y. Dai, N. Jin, B. Lou, Energy efficiency solutions for buildings: Automated fault diagnosis of air handling units using generative adversarial networks, *Energies*, **12** (2019), 527. <https://doi.org/10.3390/en12030527>
177. D. Zhao, S. Liu, D. Gu, X. Sun, L. Wang, Y. Wei, et al., Enhanced data-driven fault diagnosis for machines with small and unbalanced data based on variational auto-encoder, *Meas. Sci. Technol.*, **31** (2019), 035004. <https://doi.org/10.1088/1361-6501/ab55f8>



178. J. An, S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, *Spec. Lect. IE*, **2** (2015), 1–18.
179. G. San Martin, E. López Droguett, V. Meruane, M. das Chagas Moura, Deep variational autoencoders: A promising tool for dimensionality reduction and ball bearing elements fault diagnosis, *Struct. Health Monit.*, **18** (2019), 1092–1128. <https://doi.org/10.1177/1475921718788299>
180. Y. Kawachi, Y. Koizumi, N. Harada, Complementary set variational autoencoder for supervised anomaly detection, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, (2018), 2366–2370. <https://doi.org/10.1109/ICASSP.2018.8462181>
181. D. Park, Y. Hoshi, C. C. Kemp, A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder, *IEEE Rob. Autom. Lett.*, **3** (2018), 1544–1551. <https://doi.org/10.1109/LRA.2018.2801475>
182. S. Lee, M. Kwak, K. L. Tsui, S. B. Kim, Process monitoring using variational autoencoder for high-dimensional nonlinear processes, *Eng. Appl. Artif. Intell.*, **83** (2019), 13–27. <https://doi.org/10.1016/j.engappai.2019.04.013>
183. K. Wang, M. G. Forbes, B. Gopaluni, J. Chen, Z. Song, Systematic development of a new variational autoencoder model based on uncertain data for monitoring nonlinear processes, *IEEE Access*, **7** (2019), 22554–22565. <https://doi.org/10.1109/ACCESS.2019.2894764>
184. G. Ping, J. Chen, T. Pan, J. Pan, Degradation feature extraction using multi-source monitoring data via logarithmic normal distribution based variational auto-encoder, *Comput. Ind.*, **109** (2019), 72–82. <https://doi.org/10.1016/j.compind.2019.04.013>
185. J. Wu, Z. Zhao, C. Sun, R. Yan, X. Chen, Fault-attention generative probabilistic adversarial autoencoder for machine anomaly detection, *IEEE Trans. Ind. Inf.*, **16** (2020), 7479–7488. <https://doi.org/10.1109/TII.2020.2976752>
186. G. Ciaburro, An ensemble classifier approach for thyroid disease diagnosis using the AdaBoostM algorithm, in *Machine Learning, Big Data, and IoT for Medical Informatics*, Academic Press, (2021), 365–387. <https://doi.org/10.1016/B978-0-12-821777-1.00002-1>
187. Z. Gao, C. Cecati, S. X. Ding, A survey of fault diagnosis and fault-tolerant techniques—Part I: fault diagnosis with model-based and signal-based approaches, *IEEE Trans. Ind. Electron.*, **62** (2015), 3757–3767. <https://doi.org/10.1109/TIE.2015.2417501>
188. M. Djeziri, O. Djedidi, S. Benmoussa, M. Bendahan, J. L. Seguin, Failure prognosis based on relevant measurements identification and data-driven trend-modeling: Application to a fuel cell system, *Processes*, **9** (2021), 328. <https://doi.org/10.3390/pr9020328>
189. M. Aliramezani, C. R. Koch, M. Shahbakhti, Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions, *Prog. Energy Combust. Sci.*, **88** (2022), 100967. <https://doi.org/10.1016/j.pecs.2021.100967>
190. D. Passos, P. Mishra, A tutorial on automatic hyperparameter tuning of deep spectral modelling for regression and classification tasks, *Chemom. Intell. Lab. Syst.*, **233** (2022), 104520. <https://doi.org/10.1016/j.chemolab.2022.104520>
191. A. Zakaria, F. B. Ismail, M. H. Lipu, M. A. Hannan, Uncertainty models for stochastic optimization in renewable energy applications, *Renewable Energy*, **145** (2020), 1543–1571. <https://doi.org/10.1016/j.renene.2019.07.081>

---

192. M. H. Lin, J. F. Tsai, C. S. Yu, A review of deterministic optimization methods in engineering and management, *Math. Probl. Eng.*, **2012** (2012). <https://doi.org/10.1155/2012/756023>



AIMS Press

©2022 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)