Machine Learning approach for Predictive Maintenance in Industry 4.0

Conference Paper · July 2018

DOI: 10.1109/MESA.2018.8449150

6 authors, including:

Marina Paolanti
University of Macerata

Luca Romeo
University of Macerata

Adriano Mancini
Università Politecnica delle Marche

Emanuele Frontoni
University of Macerata

2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)

Machine Learning approach for Predictive Maintenance in Industry 4.0

Marina Paolanti, Luca Romeo
Andrea Felicetti, Adriano Mancini
Emanuele Frontoni
Department of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
Email: {m.paolanti, l.romeo}@univpm.it
a.felicetti@pm.univpm.it
{a.mancini, e.frontoni}@univpm.it

Jelena Loncarski
Department of Engineering Sciences
Division for Electricity Research
Uppsala University
Box 534, S-751 21 Uppsala, Sweden
Email: jelena.loncarski83@gmail.com

***Abstract*—Condition monitoring together with predictive maintenance of electric motors and other equipment used by the industry avoids severe economic losses resulting from unexpected motor failures and greatly improves the system reliability. This paper describes a Machine Learning architecture for Predictive Maintenance, based on Random Forest approach. The system was tested on a real industry example, by developing the data collection and data system analysis, applying the Machine Learning approach and comparing it to the simulation tool analysis. Data has been collected by various sensors, machine PLCs and communication protocols and made available to Data Analysis Tool on the Azure Cloud architecture. Preliminary results show a proper behavior of the approach on predicting different machine states with high accuracy.**

I. Introduction

Predictive maintenance (PdM) or sometimes called "on-line monitoring", "risk-based maintenance", or "condition-based maintenance", is a subject to many recent research papers with the long history behind it. It refers to the intelligent monitoring of equipment to avoid future failures. Predictive maintenance has evolved from the first method that is visual inspection to automated methods using the advanced signal processing techniques based on pattern recognition and machine learning, neural networks, fuzzy logic, etc. The automated methods provide viable solution to many industries detecting and collecting sensitive information from the equipment which are mainly motors, where human eyes or ears can cease to do so [1]. Together with integrated sensors, predictive maintenance can avoid unnecessary equipment replacement, reduce machine downtime, find the root cause the fault, and in this way save costs and improve efficiency. Predictive maintenance overlaps with the scope of preventive maintenance in terms of scheduling the maintenance activity in advance to avoid machine failures. In contrast to conventional preventive maintenance, predictive maintenance schedules activities are based on collected data from sensors and analysis algorithms [2], [3]. In process industries, induction motors make up approximately 70% of all driven electrical loads. In this regard, there has been much interest on ways to better diagnose the wellness condition of these motors [4]. The bearing failure is identified as the most frequent cause of motor failure and most common maintenance problem. Accordingly, predictive maintenance mainly focuses on two aspects: energy efficiency improvement (key of energy saving) and unscheduled downtime reduction. The algorithms developed around these two can also be generally divided into two categories:

1) energy and efficiency, there have been many evaluation methods and devices developed, as in [5],
2) system condition monitoring, including the detection of motor faults, with various fault-detection techniques developed, as for instance in [6].

Some initial concepts of intelligent predictive decision support systems have been introduced in [7]. Algorithms are essential for effective predictive maintenance. There are various kinds of techniques to be applied in various phases of PdM implementation, i.e. data processing, diagnostics, and prognostics, as given in [8]. In PdM three kinds of approach can be distinguished:

1) Data-driven approach;
2) Model-based approach;
3) Hybrid approach.

The data-driven approach is also known as the data mining approach or the machine learning approach, which uses historical data to learn a model of system behavior. Model-based approach has the ability to incorporate physical understanding of the target product, relying on the analytical model to represent the behavior of the system. Machine learning approaches are viably used in the areas where the availability of data is increasing [9], [10], [11], [12] such as the maintenance in industry sector. It is increasingly providing effective solutions, cloud-based solutions, and newly introduced algorithms [13]. Machine Learning-based PdM can be divided into the following two main classes: *Supervised*, where information on the occurrence of failures is present in the modelling dataset

978-1-5386-4643-4/18/$31.00 ©2018 IEEE

and *Unsupervised* - where logistic and/or process information is available, but no maintenance data exists [13], [14]. The availability of maintenance information mostly depends on the nature of the existing maintenance management policy. When possible, supervised solutions are preferable. From the Machine Learning perspective, depending on the output of the data set, two classes of supervised problem are possible: regression problem (if output assumes continuous values), and classification problem (if output assumes categorical values) [15].

In this paper a new PdM methodology based on PdM machine learning approach on a cutting machine is presented. PdM is a strategy viable adopted when dealing with maintenance issues given the increasing need to minimize downtime and associated costs. The methodology has been implemented in the experimental environment on the example of a real industrial group, producing accurate estimations. An example of the machine adopted in our study is depicted in figure 1.

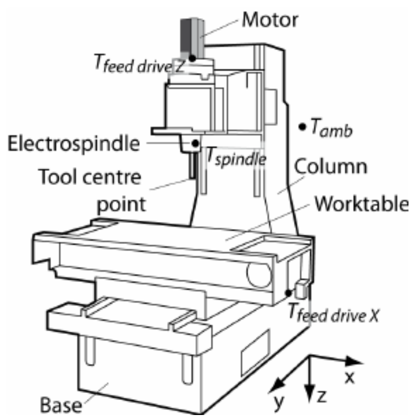

Figure 1. Example of the machine adopted in this study.

Data has been collected by various sensors, machine PLCs and communication protocols and made available to Data Analysis Tool. The proposed PdM methodology allows dynamical decision rules to be adopted for maintenance management, which is achieved by training a Random Forest approach on Azure Machine Learning Studio. Preliminary results show a proper behavior of the approach on predicting different machine states with high accuracy (95%) on a data set of 530731 data readings on 15 different machine features collected in real time from the tested cutting machine.

Paper contributions are mainly on the overall cloud architecture for Industry 4.0, on the application of ML approaches to a real data set from machines on the field, on the high level of accuracy on predicting the state of the main spindle rotor.

The paper is organized as follows: Section II gives an overview on the data analysis, including the approach used in the experimental verification to analyze the residuals. Section III gives the Machine learning approaches for PdM, explaining in deep the regression and the classification methods. Section IV gives the experimental results on the real use case where Machine Learning approach for PdM has been implemented and tested.

II. The Data Analysis

Predictive Maintenance is an important maintenance tool that is based on the possibility of estimating the future values of some quantities that characterize a system (typically a machine, a plant, or a production process) through particular mathematical models in order to identify in advance the anomalies and potential faults [16], [17]. The basic scheme of PdM is as follows:

- Measurement of physical quantities in real time.
- Estimation of measurable (or non-measurable) parameters at time $t + dt$.
- Identification of the system status considered anomalous or faulty.
- Planning of preventive and corrective activities before the system reaches the critical condition.

Examples of predictive maintenance are the following:

- Machine vibrations can signal bearing deterioration or deformation of particular mechanical parts.
- The temperature of a motor and its drown current, may indicate that friction and possible mechanical malfunction are degrading the functionality.
- The measurement of particles in a lubricant indicates the degradation of rubbing contact parts. With appropriate sensors it is possible to measure the composition of the lubricating oil and check the health of the machine.

The first phase of these processes is based on the estimation of the parameters. On the basis of PdM, the technology is capable of making reliable forecasts. If the prediction algorithms produce incorrect estimates or with too large reliability intervals, it will be difficult to identify the anomalies and make the decisions on maintenance and correction. Broadly speaking, the forecasts can be of two main types:

- Cross-Sectional Forecasting;
- Time Series Forecasting.

A. Cross-Sectional Forecasting

Cross-Sectional Forecasting is the estimation of parameters of which there exist no measurements, by using measures on other variables that have been observed. As an example, it can be possible to predict the life of an electronic component used in particular conditions, by measuring the electric current that passes through it [18].

B. Time Series Forecasting

Time Series Forecasting is referring to the estimation of parameters that change over time, being measured until time instant t and the value to be predicted is at the time instant $t + dt$. Typically, the measurements of the variable of interest can be obtained at regular intervals, and then it is possible to predict the future values. The simplest example is the prediction of the minutes of residual charge of our mobile phones, which is estimated based on the consumption that we have made, and the way we use the phone [19]. In time series, it is usual to identify:

- Trends, or a long-term increase (or decrease) in values.

- The seasonal phenomena, or the phenomena that determine changes in values over a period of time that is always repeating the same duration.
- Cyclical phenomena that cause increases and decreases in values with fluctuations that do not always have the same duration, i.e. they are not periodic.

One of the most important things to understand when analyzing data is the type of relationship between the measured quantities. To do this, graphic visualization with scattered plots can be viably used to identify how the data are dependent. The hypothesis of linear regression is that the estimating phenomenon has a linear behavior. A simple way to check if the signal we want to predict is linear, or contains other information that is not contained in our forecast model, is to analyze the residuals [20].

C. The analysis of the residuals

The residuals are the difference between the measured values of the size and the fitting values obtained with the prediction line

$$e_i = y_i - y'_i \tag{1}$$

where *ei* is the residual of the *i-th* measurement, y_i is the *i-th* measured value and $y'_i = m't_i + q'$ is the estimated value at time t_i. Coefficients m and q are calculated with the method of least squares [21]. A practical method to verify whether the calculated regression model satisfies the signal to be predicted is the autocorrelation. The autocorrelation of the residues above calculated can be calculated, counting how many of the residuals have a value included in the interval:

$$Int = \pm \frac{2}{\sqrt{N}} \tag{2}$$

where N is the number of the measurements. Usually it is considered that if $>95\%$ of the residuals are within the $\pm Int$, they are not correlated among them and the noise can be considered as white noise. This approach allows to construct algorithms that use linear regression for signal prediction, useful for predictive maintenance, and automatically verifies in real time any variations of the system under observation. It is also useful to use the test of residuals to understand if linear regression is the best choice for the predictive model we want to build.

III. Machine Learning Approaches for PdM

Figure 2 gives a scheme, showing the actions and technical activities to be undertaken to implement PdM. Machine Learning phase of the PdM can be implemented on the basis of the data collected by the I4.0 and MES machine layers (Figure 2).

When qualifying the suitability of a problem for a predictive maintenance solution (result of data analysis), three essential data sources must be found:

1) *Fault history*: Generally, error events are very rare in predictive maintenance applications. However, when compiling predictive models that estimate failures, it is necessary for the algorithm to learn the normal operating scheme, in addition to the failure scheme through the training process. Consequently, it is essential that the training data contain a sufficient number of examples in both categories.
2) *Maintenance/repair history*: An essential source of data for predictive maintenance solutions is the detailed asset of maintenance history, which contains information about replaced components, preventive maintenance tasks performed, and so on.
3) *Machine conditions*: to estimate how many days (or hours, kilometers, etc.) a machine lasts before a failure occurs, it is assumed that its health status decreases with time. It is therefore necessary that the data contain time-varying functions that acquire aging patterns or any anomaly that could cause performance reduction.

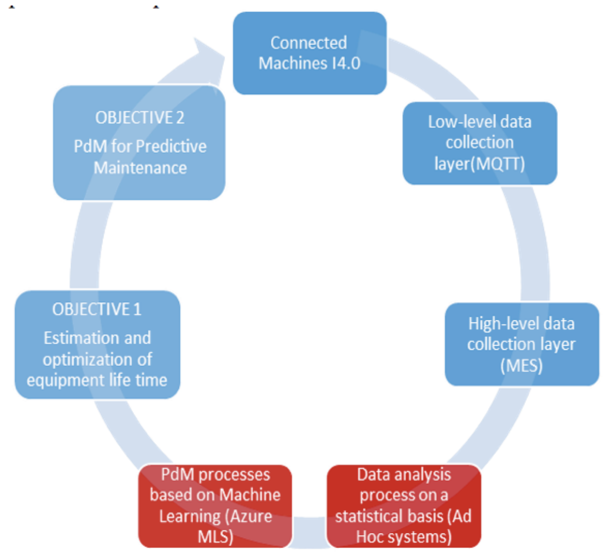

Figure 2. Scheme of the activities to be undertaken in PdM.

In order to prepare the data in the format required to create the functions to be included in the Machine Learning algorithm, it is necessary to perform some pre-processing steps. The first is to divide the duration of data collection into units of time in which each record is a unit of time for an asset. The generic data schema could be:

- *Maintenance records*: these are the records of the maintenance actions performed. The unprocessed maintenance data is usually associated with an asset ID and a timestamp with information on maintenance activities performed at that time. In the case of unprocessed data, maintenance activities must be transformed into category columns, where each category corresponds to a type of maintenance action. The base data schema for maintenance records will include columns for asset IDs, time and maintenance actions.
- *Fault records*: these are records that belong to the target of the estimate, i.e. failures or reason for the fault. They can

be specific error codes or error events defined by specific operating conditions. In some cases the data includes multiple error codes, some of which correspond to faults of interest. Not all faults are the goal of an estimate, so others are typically used to create functions that can be related to failures. The base data schema for fault records will include columns for asset ID, time and fault, or reason for failure if a pattern is available.

- *Machine conditions*: preferably real-time monitoring data related to the operating conditions of the data.
- *Machine and operator data*: these data can be combined in a scheme to identify the assets managed by a given operator together with the properties of the assets and the operator.

Table I
FEATURES DATASET.

Features	**Significance**
statoRot	Functional spindle rotor status (c)
Timestamp	Event Recorded
Machine	Running Machine
Spindle speed	Spindle rotation speed
Spindle power	Power absorbed by the spindle
Spindle position	Spindle angular position
X PositionDiff	X-axis real-to-nominal position diff
Y PositionDiff	Y-axis real-to-nominal position diff
Z PositionDiff	Z-axis real-to-nominal position diff
X Speed	X axis speed
Y Speed	Y axis speed
Z Speed	Z axis speed
X Current	Absorbed current X axis
Y Current	Absorbed current Y axis
Z Current	Absorbed current Z axis

In Table I a list of data is reported with the *statoRot* status variable that will be the machine condition variable in the result section.

The first step in modeling is the design of the functions. The idea of generating functions consists conceptually in describing and abstracting the integrity condition of a machine at a given time using historical data collected up to that instant. The design methods of the functions described below can however be used as a baseline for creating functions. In the following, delay (lag) functions need to be created from data sources including timestamps, and also static functions created from static data sources providing examples from the illustrated use cases. In the preventive maintenance the chronological data generally include timestamps that indicate the time of collection of each single data. For each asset record, a window with dimension "W" is selected, corresponding to the number of units of time for which we want to calculate the chronological aggregations. The aggregate sequencing functions are then calculated using the W periods before the date of that record. Examples of sequential aggregations are: incremental counts, averages, standard deviations, outliers based on standard deviations, cumulative sum measures, minimum and maximum window values, or trend changes, peaks and level changes using algorithms that detect anomalies, etc.

A. *Machine Learning Algorithms*

The *binary classification* is viably used for predictive maintenance, being able to estimate the probability that the equipment will fail over a future period of time. The time period is determined and based on business rules and available data. Some common time periods are the minimum downtime or the time required to perform the maintenance routines necessary to solve the problem that it could occur in that period of time. To use the binary classification, it is necessary to identify two types of examples, which are called positive and negative. Each example is a record of a unit of time for an asset that conceptually describes the operating conditions by designing the functions by using historical and other data sources. In the context of the binary classification for predictive maintenance, the positive types denote the errors, and the negative ones the normal operations. The aim is to find a model that identifies the probability that each new example may fail or work normally within the next unit of time.

Regression models in predictive maintenance are used to calculate the remaining useful life of an asset, defined as the amount of time during which the asset remains operational before the next failure occurs. As in the binary classification, each example is a record that belongs to the unit of time for an asset. In the context of regression, however, the aim is to find a model that calculates the remaining useful life of each new example as a continuous number. This time period is defined as a multiple of the unit of time.

Multi-class classification for predictive maintenance can be used to estimate two future results. The first is to assign an asset to one of the different time periods in order to allocate a time interval for the failure of each asset. The second consists in identifying the probability of failure in a future period due to one of the multiple root causes. This allows the maintenance personnel to handle the problem in advance. Another multi-class modeling technique focuses on determining the most likely root cause of a given failure. This allows to provide the suggestions for the main maintenance actions to be performed to repair a fault. By having a categorized list of root causes and associated repair actions, technicians can perform the first repair actions after failures more effectively. In predictive maintenance, similar to any other area of the solution containing time-stamp data, the typical training and test routine must consider the variable aspects over time for a better generalization of future unseen data. Many Machine Learning algorithms depend on a number of hyper parameters that can significantly modify the performance of the model. The optimal values of these hyper parameters are not automatically calculated during the training of the model, but must be specified by the data scientist. Various ways are available to find the optimal values of the hyper parameters. The most common one is the "cross-validation of k sections", which randomly subdivides the examples into "k" sections. For each set of values of the hyper parameters, the learning algorithm is executed k times. At each iteration, the examples in the current section are used as a validation set, while the rest of the examples are used as a training set. The algorithm performs training on training examples and performance metrics are calculated on validation examples. At the end of this cycle, for each set of values of the hyper parameters the values of the performance metrics k are calculated and the values of the hyper parameters with the best average performances are selected.

In the result section a Decision Forest (DF) classifier is used. The DF Classifier is an ensemble learning method used to improve classification performance. It uses bagging by combining the output of several classifiers [22]. It works by building multiple decision trees and then it votes on the common output class. The main goal of ensamble methods is that a high number of "weak learner" can be used to create a "strong learner". Generally, in each tree, a sequence of simple tests is run for each class, increasing the levels of a tree structure until a leaf node (decision) is reached. In Azure Machine Learning Studio, this kind of classifier consists of an ensemble of decision trees. Ensemble models usually provide better coverage and accuracy than single decision trees.

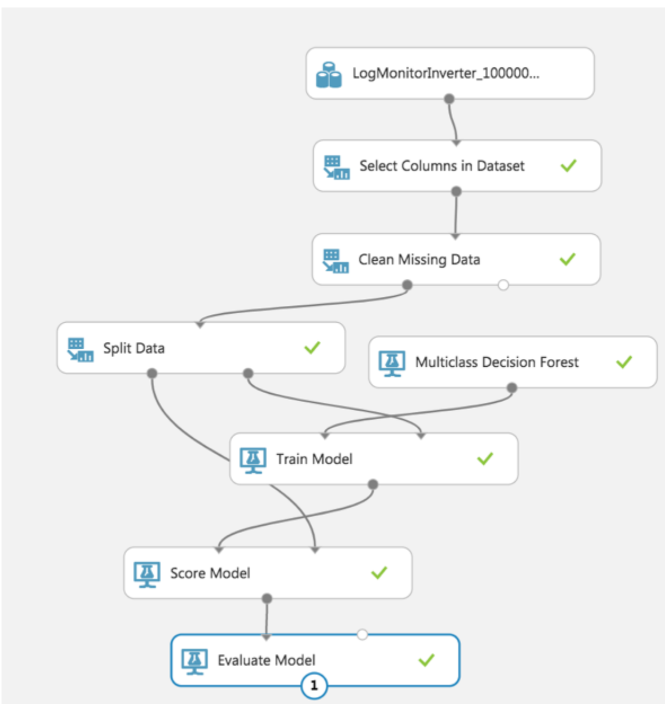

Figure 3. General schema of Classification Process on Azure Machine Learning Studio.

IV. Experimental Results

A. Use Case

The Predictive Maintenance approaches discussed above have been implemented on a real cutting machine woodworking machinery that is a machining center on wood industry. The main objectives of the proposed test is the evaluation of the proposed system for ML predictive maintenance of the cutting machine through FingerPrint analysis: analysis of drive data for axis monitoring and vibration data analysis to estimate spindle health status. The proposed feature set is reported in TableI and a data set of 530731 data readings were corrected on 15 different machine features collected in real time from the tested cutting machine.

Different sources of data are taken in consideration in the data collecting process and the following overall architecture was designed:

- *FlightRecorder*: it is an industrial PC placed inside the MCM machine tool. It is communicating with various elements inside the machine in order to collect the data necessary for the analysis.
- *IT Accelerometer*: is an accelerometer placed on the spindle head of the MCM machines that has to collect vibration data, which exposes an interface for automatic data export.

As for the architecture of the highest level will be introduced:

- *Data analysis unit*: it is a PC that hosts the data analysis software that will be implemented.
- *Simulation unit*: it is a Azure Cloud architecture based on Machine Learning Studio that hosts the ML software that will be implemented (Figure 3).

B. Architecture and data flow

The FlightRecorder system has three different data sources, as presented in Figure 4:

- *Drive data*: they are sampled in real time by the CNC and made available to the FlightRecorder through a communication protocol developed by D.Electron. This data is associated with a time reference (timestamp) by the CNC at the time of sampling.
- *Data related to I/O signals*: they are sampled in real time by the machine PLC and made available to the FlightRecorder the same way as the drive data.
- *Vibrational data*: vibrational analysis in the frequency domain is performed to allow efficient monitoring of the spindle unit. A dedicated sensor/electronic system is required, which after each episode of fingerprint or working step makes available to the FlightRecorder both raw data related to the vibration spectrum trend, and an evaluation on how much the spectrum detected is deviated from the average one obtained under good operating conditions.

All the collected data, using the proposed architecture were evaluated using the 30% of the data set as a training set and the rest for the results evaluation. Overall accuracy is reported in Table II. Results are accurate for predicting the rotor status on 4 different classes based on the proposed feature set.

V. Conclusions and Future Works

In this paper a new PdM methodology based on PdM machine learning approach on a cutting machine is presented.

Table II
CLASSIFICATION RESULTS.

Metrics	**Results**
Overall Accuracy	0.95
Average Accuracy	0.92
Micro-Averaged Precision	0.94
Macro-Averaged Precision	0.93
Micro-Averaged Recall	0.95
Micro-Averaged Recall	0.94

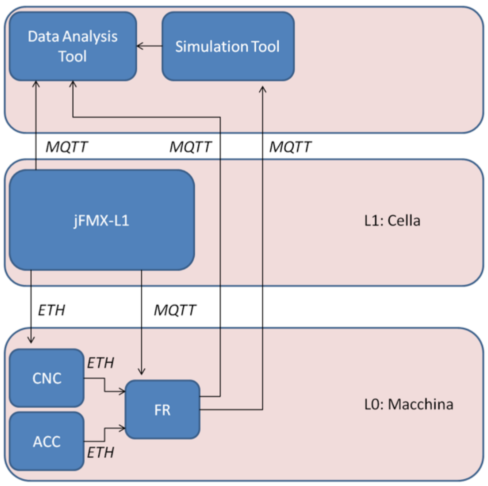

Figure 4. System Data Flow.

PdM is a strategy viable adopted when dealing with maintenance issues given the increasing need to minimize downtime and associated costs. The methodology has been implemented in the experimental environment on the example of a real industrial group, producing accurate estimations. Data has been collected by various sensors, machine PLCs and communication protocols and made available to Data Analysis Tool. The proposed PdM methodology allows dynamical decision rules to be adopted for maintenance management, which is achieved by training a Random Forest approach on Azure Machine Learning Studio. Preliminary results show a proper behavior of the approach on predicting different machine states with high accuracy (95%) on a data set of 530731 data readings on 15 different machine features collected in real time from the tested cutting machine.

Paper contributions are mainly on the overall cloud architecture for Industry 4.0, on the application of ML approaches to a real data set from machines on the field, on the high level of accuracy on predicting the state of the main spindle rotor.

Future work will go in the direction of having more robust data-set, investigating different fault scenarios, exploring a different set of features, in particular in the frequency domain.

REFERENCES

[1] H. M. Hashemian and W. C. Bean, "State-of-the-art predictive maintenance techniques," *IEEE Transactions on Instrumentation and measurement*, vol. 60, no. 10, pp. 3480–3492, 2011.

[2] S.-j. Wu, N. Gebraeel, M. A. Lawley, and Y. Yih, "A neural network integrated decision support system for condition-based optimal predictive maintenance policy," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 37, no. 2, pp. 226–236, 2007.

[3] E. Frontoni, R. Pollini, P. Russo, P. Zingaretti, and G. Cerri, "Hdomo: Smart sensor integration for an active and independent longevity of the elderly," *Sensors*, vol. 17, no. 11, p. 2610, 2017.

[4] B. Lu, D. B. Durocher, and P. Stemper, "Predictive maintenance techniques," *IEEE Industry Applications Magazine*, vol. 15, no. 6, 2009.

[5] B. Lu, T. G. Habetler, and R. G. Harley, "A survey of efficiency-estimation methods for in-service induction motors," *IEEE Transactions on Industry Applications*, vol. 42, no. 4, pp. 924–933, 2006.

[6] W. T. Thomson and M. Fenger, "Current signature analysis to detect induction motor faults," *IEEE Industry Applications Magazine*, vol. 7, no. 4, pp. 26–34, 2001.

[7] R. Yam, P. Tse, L. Li, and P. Tu, "Intelligent predictive decision support system for condition-based maintenance," *The International Journal of Advanced Manufacturing Technology*, vol. 17, no. 5, pp. 383–391, 2001.

[8] J.-H. Shin and H.-B. Jun, "On condition based maintenance policy," *Journal of Computational Design and Engineering*, vol. 2, no. 2, pp. 119–127, 2015.

[9] M. Paolanti, C. Kaiser, R. Schallner, E. Frontoni, and P. Zingaretti, "Visual and textual sentiment analysis of brand-related social media pictures using deep convolutional neural networks," in *International Conference on Image Analysis and Processing*. Springer, 2017, pp. 402–413.

[10] M. Paolanti, E. Frontoni, A. Mancini, R. Pierdicca, and P. Zingaretti, "Automatic classification for anti mixup events in advanced manufacturing system," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2015, pp. V009T07A061–V009T07A061.

[11] M. Paolanti, M. Sturari, A. Mancini, P. Zingaretti, and E. Frontoni, "Mobile robot for retail surveying and inventory using visual and textual analysis of monocular pictures based on deep learning," in *Mobile Robots (ECMR), 2017 European Conference on*. IEEE, 2017, pp. 1–6.

[12] S. Naspetti, R. Pierdicca, S. Mandolesi, M. Paolanti, E. Frontoni, and R. Zanoli, "Automatic analysis of eye-tracking data for augmented reality applications: A prospective outlook," in *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*. Springer, 2016, pp. 217–230.

[13] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi, "Machine learning for predictive maintenance: A multiple classifier approach," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 812–820, 2015.

[14] K. Wang, "Intelligent predictive maintenance (ipdm) system–industry 4.0 scenario," *WIT Transactions on Engineering Sciences*, vol. 113, pp. 259–268, 2016.

[15] G. Druck, G. Mann, and A. McCallum, "Learning from labeled features using generalized expectation criteria," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 595–602.

[16] R. K. Mobley, *An introduction to predictive maintenance*. Butterworth-Heinemann, 2002.

[17] A. Grall, L. Dieulle, C. Bérenguer, and M. Roussignol, "Continuous-time predictive-maintenance scheduling for a deteriorating system," *IEEE transactions on reliability*, vol. 51, no. 2, pp. 141–150, 2002.

[18] J. S. Armstrong, *Principles of forecasting: a handbook for researchers and practitioners*. Springer Science & Business Media, 2001, vol. 30.

[19] C. Chatfield, *Time-series forecasting*. CRC Press, 2000.

[20] F. E. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," *Omega*, vol. 29, no. 4, pp. 309–317, 2001.

[21] S. J. Haberman, "The analysis of residuals in cross-classified tables," *Biometrics*, pp. 205–220, 1973.

[22] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.