A semantic-driven approach for data analytics to support Prognostics and Health Management

Polenghi A., Cattaneo L., Macchi M.

Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133, Milano

Abstract: Today data analytics is vital for companies willing to extrapolate information from their assets to support asset-related decisions. Information is relevant but not enough to exploit the potentials hidden in domain-related knowledge. The focus of this paper is predictive maintenance, herein knowledge is relevant to support the design of a Prognostics and Health Management (PHM) process to achieve a reliable decision-making. In this scope, the paper builds on the presumption that data analytics can be empowered by semantic data modelling to conceptualise and formalize data before the application of any kind of advanced algorithm implementing a data-driven approach. Thus, this research aims at proposing a semantic data model that guides the data analytics by revealing data characteristics and inter-relationships and guarantees completeness to finally support the PHM process. A data-driven approach, joining semantic data modelling and analytics, is proven through examples taken from the controlled environment of the Industry 4.0 Laboratory of the School of Management of Politecnico di Milano.

Keywords: data analytics, semantic data model, PHM, maintenance

1.Introduction

Today data analytics (DA) is vital for all the companies willing to extrapolate more and more information from their physical assets (plants, machines, and equipment) in the production systems. In fact, the extensive installation of sensors and, in general, condition monitoring systems are increasing the data sources at hands to improve the asset-related decision-making support. This phenomenon is influencing different domains. Production management (Cheng *et al.*, 2018) and maintenance management (Lee *et al.*, 2015) are two areas in which DA fosters benefits; their intersection is also fertile for DA (Ji and Wang, 2017). Specifically, maintenance is looking towards DA tools to extrapolate information about the assets the company owns, such as AHI (Asset Health Indicator) and RUL (Remaining Useful Life) (Lee, Jin and Bagheri, 2017).

To obtain these results, the data collected by companies could be processed by statistical techniques (such as e.g. traditional reliability analysis techniques like Weibull analysis, or degradation modelling as Gamma process, Markovian-based models, Covariate-based hazard models etc.) and more advanced AI-based techniques (as machine learning and deep learning) (Si *et al.*, 2011; Lei *et al.*, 2018). This may be exploited in condition-based maintenance (CBM), which is a promising application of DA to monitor and control the asset degradation (Fumagalli *et al.*, 2019). This may help also to evolve towards a predictive maintenance strategy, anticipating failures by comparing normal conditions with forecasted asset performance (Márquez, Del Castillo and Fernández, 2020).

However, DA per se cannot solve all maintenance-related issues especially if domain-related knowledge is missing or partially used. When addressing CBM and predictive maintenance, domain-related knowledge appears to be a cornerstone (Nuñez and Borsato, 2018). Thus, to exploit the data available by the increased number of sources, it is fundamental to enable their interpretation based on their semantics (Kiritsis, 2013). This promises to favour a step forward towards knowledge synthesis into a semantic data model useful for the effective building of CBM/predictive maintenance strategy based on DA. Hence, this research work aims at proposing a semantic data model to guide DA to monitor and control current asset state and predict its future degradation.

More specifically, the overarching goal of the work is to develop an integrated semantic data modelling and DA to support the design of the PHM process implementing the CBM/predictive maintenance strategy within advanced systems; the final objective is to optimise the maintenance management based on proper decision-making powered by PHM (Guillén *et al.*, 2016). The integration of semantic data modelling and DA is practically explored in a Proof of Concept (PoC) run within the controlled environment of Industry 4.0 Laboratory (I4.0Lab, industry40lab.org) of the School of Management of Politecnico di Milano.

The document is structured as follows. Section 2 explores the literature summarising the main contributions related to semantic data modelling for CBM/predictive maintenance; Section 3 describes the proposed semantic data model while Section 4 presents the application in the I4.0Lab; at the end, Section 5 draws some conclusions and envisions future research works.

2. Literature review

A preliminary search of the extant scientific literature is performed to assess the use of semantic data modelling for CBM and predictive maintenance. The literature search is set according to the research protocol implying:

Scopus as database, limitation to English-written documents, both journal and conference papers, keywords as (*data model**) AND ((*condition*) OR (*predictive*)) AND (*maintenance*). It is worth noting the word *semantic* is not added to the keywords to not narrow the document search. Overall, the searching process returns 122 documents with an ever-increasing trend, described by a mean yearly rate of +6 documents, with a peak of +14 and +17 documents in 2018 and 2019, respectively. Conference papers govern the statistics about types of publication, with a significant 61,5% of the total amount.

The literature review is conducted on 78 out of 122 documents, by limiting the search to those documents addressing engineering topics, excluding medicine, agriculture and so on. Among the 78 documents, 3 are duplicated, 10 are not accessible, and 33 are out of scope. The documents labelled as out of scope include all works that do not address data modelling (e.g. papers generally reviewing the literature without any reference to data modelling) or that give data models a meaning different with respect to the intended focus of the research (for example, simulation models, Bayesian approach to cope with uncertainty, or statistical algorithm for prediction). At the end, 32 papers undergo a full text reading to extrapolate relevant variables for the analysis, looking at both data modelling objectives and languages (see Figure 1 and Figure 2, respectively).

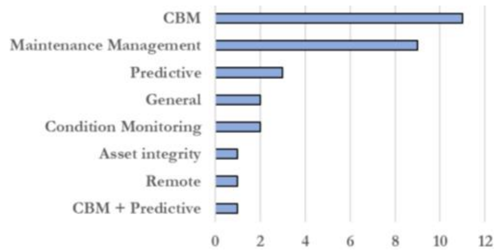

Figure 1: Objective of the analysed documents

The documents mainly deal with CBM or look for a more general improvement of the maintenance management software platform, especially relating to interoperability (Hassanain, Froese and Vanier, 2000; Campos *et al.*, 2010). It should be noted that the total number of documents in Figure 1 is greater than the ones in Figure 2. This happens because most documents deal with data model for maintenance, but do not actually report the developed model in the paper, so it is hard to trace back to the used language.

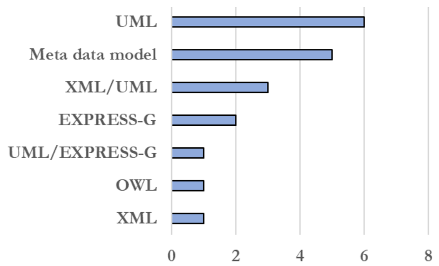

Figure 2: Use of data modelling language

Based on the evidence from Figure 2, UML (Unified Modelling Language) is the most widespread modelling language due to its flexibility in adoption and ease of understanding. It is the de-facto standard formalism for object-oriented modelling (Negri *et al.*, 2016). Also, EXPRESS-G is used as the graphical representation of EXPRESS semantic language used in the ISO 15926-2 (West, 2011), in which a data model for reliability data exchange along the lifecycle is developed. In addition to UML and EXPRESS-G, other formalisms are adopted for semantic data modelling. The reader interested in an overview of those languages can refer to (Negri *et al.*, 2016).

2.1 Data model definition

The literature review shows how the term data model in maintenance comes with different meanings. Thus, it is important to properly define the perspective the authors are taking while dealing with this topic. To this end, Table 1 summarises the most relevant definitions of data model as retrieved in scientific literature selected for this paper.

Table 1: Relevant definitions of data models

Data model definition	Reference
"application-specific implementation-independent representation of the data that will be handled by the prospective application"	(Keet, 2018)
"classes of component objects with their own relationships and attributes"	(Macchi *et al.*, 2016)
"depiction of the relationships that exist among specific values of data"	(Luisi, 2014)
"representation of data in terms of named sets of objects, named sets of values, named sets of relationships, and constraints over these object, value, and relationship sets"	(Embley, 2009)
"method of organizing data that reflects the basic meaning of data items and the relationships among them"	(Gartner, 2019)

In line with the definitions, other evidences from literature are remarking the role of "semantic data model" as a data model with semantic-enriched relationships that could be further exploited in ontology for reasoning capabilities (Negri *et al.*, 2017)[1]. Furthermore, it is observed that data models are extending their native domain of application, i.e. database modelling, to other domains, like PLM (Product Lifecycle Management) and ALM (Asset Lifecycle Management), where a remarkable role for decision-making is emergent (Polenghi *et al.*, 2019).

2.2 Concluding remarks

The results of the literature review show interesting evidences on the adoption of semantic data modelling for CBM and predictive maintenance. The main overarching goals of their use could be summarised as follows:

- to establish a stable connection between several transducers and to streamline the collected data

[1] In the reminder, "data model" and "semantic data model" are used interchangeably.

towards de-/centralised databases to foster a reliable asset monitoring;

- to ensure interoperability between different IT software systems and their modules so as to establish an integrated information system; an important example in this regard is OSA-CBM, see also (ISO 13374-1, 2003) and mimosa.org;
- to bridge and integrate business processes as design and operation of the asset or asset system;
- to guarantee data quality, insuring availability of data and information while preventing errors by protecting and restricting from manual inputs.

Notwithstanding the wide scope of work, Figure 1 shows that, even if well-adopted for CBM-related purposes, data modelling for predictive maintenance is not so developed in scientific research. Therefore, the current work aims at focusing predictive maintenance application.

3. Proposed semantic data model

The aim of this section is to propose a semantic data model that support the PHM process by guiding the DA for predictive maintenance. The formalism adopted is the UML due to its widespread adoption. Specifically, the formalism follows the guidelines from the OMG (Object Management Group, omg.org). To holistically represent the data model, the framework proposed by (Polenghi *et al.*, 2019) is used, which is composed by five blocks. Being the framework defined for any AM-related decisions, it should be fitted for the PHM: this alignment is done by relating the framework's blocks with the PHM process steps proposed by (Guillén *et al.*, 2016) and levels in ISO 13374 (L1-L6), as summarised in Figure 3.

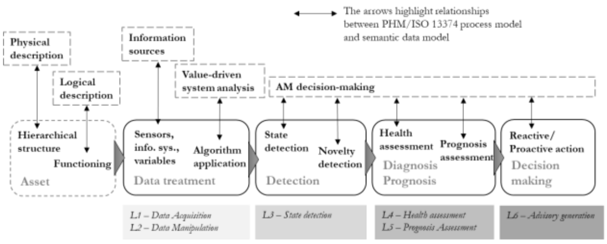

Figure 3: Alignment of semantic data model, PHM process step and ISO 13374 levels (adapt from (Guillén *et al.*, 2016))

According with the schematic representation of Figure 3, the asset initially requires the data model for its physical and logical description, which is needed to prepare further actions in regard to the data sources for the data treatment process step; thereafter, the "core" steps of the PHM process are enabled by the data model in its value-driven system analysis block to finally support decision-making. The semantic data model is illustrated in the next subsections. For a deeper view on model details, including its structure based on the five blocks, it is consultable and downloadable at the following GitHub link.

3.1 Physical description block

The physical description block collects all classes and relationships related to the physical part of the production plant. This is mainly composed by a series of classes connected by a composition relationship and it enables to express the hierarchical structure of the asset, within an asset system. Hence, the top class is the **Asset_system**, which is composed by different **Asset** classes representing the physical assets, like machines or equipment (e.g. milling machines or pumps). The **Asset** is composed by many **Maintainable_item**, like the power supply part or the unit in which a tool is installed (example in Figure 4).

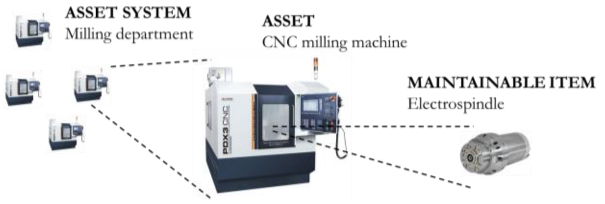

Figure 4: Asset system, asset and maintainable item classes

Sensor is also a class belonging to this block. It represents a sensor installed on a specific **Maintainable_item** to monitor a specific variable (**Sensor_maintainable_item**) or a generic-purpose sensor when it measures a variable not directly related to the asset (**Sensor_generic**) such as e.g. the temperature of the department in which the asset is installed. The importance to distinguish these two inherited classes of **Sensor** is explained in subsection 3.3.

3.2 Logical description block

The logical description block collects all classes and relationships that describe the asset functioning. Indeed, the **Asset** must perform a specific **Asset_function** as it is designed; if it cannot perform the function due to a breakdown, a **Functional_failure** happens. For example, the asset is no longer able to cut metal. This is named as **Functional_failure** and comes from a failure of one of the **Maintainable_item**, having its own **Failure_mode**, for example cutting tool-head wear. Definition of these terms may be found in the ISO 13372.

Moreover, the **Asset** has its own **Asset_health_state**, i.e. healthy, degraded, abnormal/faulty, which is influenced by the **Asset_working_state** (see Figure 5). This state is described at the beginning by data available from already installed sensors or from other elements of automation, as PLCs (Programmable Logic Controller); however, the working states are updated once the **State_detection** is established, as further specified in subsection 3.5.

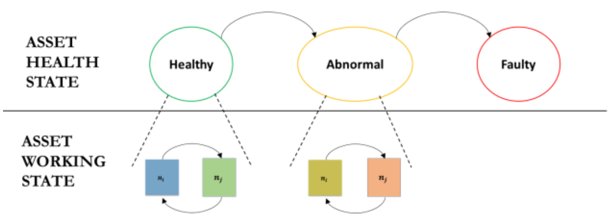

Figure 5: Asset_health_state and Asset_working_state

3.3 Information sources block

The information sources block collects all classes that represent a data or information. A first source is the **Process_requirement**, which is a set of requirements the asset must fulfil to respect the production demand. Due to the changes in requirements (as changes of production mix), the asset must work differently from time to time; therefore, the **Asset_working_regime** is useful to be sensitive to these changes. This class is composed (through aggregation relationships) by a set of **Regime_variable** as inputs to the asset specific for **Asset_working_regime**. Then, the **Regime_variable** are usually provided to the asset, through CNC (Computer Numeric Control) or information systems such as a MES (Manufacturing Execution System), and represent parameters for its functioning. For example, the spindle must rotate at 2500 rpm as a regime variable.

Data could be also sourced from the **Sensor**. If sourced from **Sensor_maintainable_item**, the variable is named as an **Operation_variable**; in case of **Sensor_generic**, it is an **Environment_variable**. **Operation_variable** is a variable measured on the asset, and specifically on a maintainable item. This variable could be thought as the result of working in a specific regime: the milling machine must have a spindle rotating speed of 2500 rpm and so the vibration of the spindle is measured, as a result of the operation. **Environment_variable** is a variable measured not directly on the asset: this variable could affect the functioning of one or more maintainable item and the corresponding asset. E.g., the temperature measured in a room is not (strictly) the result of the functioning of the asset. However, the temperature value may influence the precision of micro-operation, as the positioning of the o-ring to seal a component to prevent fluid leakage. In the model, **Environment_variable** and **Regime_variable** are represented as inheritance of the **Context_variable**. Figure 6 is a conceptual summary of the relationships of the three natures of variables discussed so far.

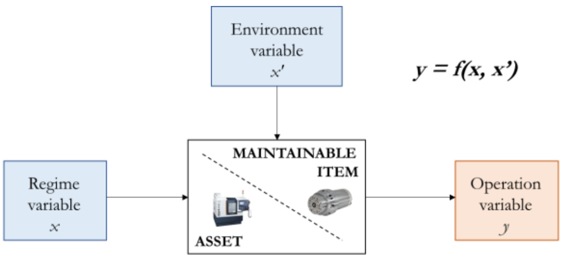

Figure 6: Regime, environment and operation variables

Through this formalisation, it is possible to improve the overall DA approach: context variables allow to describe the time-varying working conditions of the asset (Wang *et al.*, 2019). In this way, the output represented by the operation variable may be recognised, through *f*, aligned with effort required to the asset or unexpected.

Therefore, the semantic data model helps in providing the contextualisation of the variables by labelling them before the application of any algorithm: any identified variable could help unblinding the kind of needed statistical/ML algorithms, by providing a first interpretation of the data to be analysed. Also, the semantic data model guarantees making evident all data classes and related attributes for further analysis; this leads to data quality, with concern to data completeness (Tam and Kwan, 2019).

3.4 Value-driven system analysis block

This block collects the analyses and relative results that support the asset-related decision-making. The analyses rely upon the identification of the needed variables of different natures illustrated in subsection 3.3; this provides the starting point for further development by means of the potentialities due to several algorithms.

The first class is the **Algorithm_feature_generation** that allows to generate one or more **Feature** from the original **Operation_variable** and **Context_variable**. The correct generation of features could be previously requiring a proper **Algorithm_preprocessing**, which may be needed for different reasons: there could be outliers or missing values in the current dataset (Zhu *et al.*, 2018) (thus the need for an **Algorithm_preprocessing_data_cleaning**), or the variables are not yet fused (thus the need for an **Algorithm_preprocessing_data_fusion**). With the term "fusion" we intend a methodology that allows to integrate (fuse) variables from different sensors and different points in time to enable both automated and human analysis or decision-making (Khaleghi *et al.*, 2013).

Once the whole set of features is available, one or more **Significant_feature** must be extrapolated since some of the generated features could be correlated. This could be done through **Algorithm_dimensionality_reduction**, such as PCA (Principal Components Analysis), that could be useful to reduce the feature space, highlighting only those features that are significant as they better describe the variability of the functioning process under analysis. A relevant inheritance of **Significant_feature** is the class of **Health_indicator**. In practice, a **Health_indicator** is a **Significant_feature**, but the purposes are different: a significant feature is for the **State_detection_analysis**, while the health indicator is a significant feature relevant for the **Prognosis_analysis**. These analyses rely on more **Algorithm** that could be of different types, formalized by inheritances, that is: **Machine_learning_algorithm** type, **Statistical_algorithm** type, **Physics_algorithm** type (Lei *et al.*, 2018). The application of the **Algorithm** requires to evaluate its performance in **Evaluation_algorithm**.

3.5 AM decision-making block

The value-driven system analysis builds the ground for implementing the decision-making process in the AM decision-making block, fitted in this work for PHM with concern to prognosis assessment. Therefore, the **State_dection_analysis** supports the **State_detection** to describe the **Asset_working_state**; it also supports the **Novelty_detection** (Pimentel *et al.*, 2014), to recognize the novelty leading to define the **Health_state_division** which finally allows to describe the **Asset_health_state** enabling to discern healthy, abnormal and faulty. On this basis, as decision, a **Reactive_action** could come out to restore the current functioning of **Maintainable_item** or **Asset**. While supporting the **Health_state_division**, the **Prognosis_analysis** also provides the computation of the **Predicted_health_indicator** (an **Health_indicator** projected in the future (Lei *et al.*, 2018)); besides, it could evaluate the **Remaining_useful_life**, i.e. a measure of the remaining time the asset has before failing (Si *et al.*, 2011). Based on the prognosis outcomes, as decision, a **Proactive_action** could be done on the **Maintainable_item** or **Asset**.

4.Proof of Concept in the I4.0Lab

The I4.0Lab is a laboratory of the School of Management of Politecnico di Milano that allows to face industry-like problems related to automation, software architecture and management. It is a controlled environment in which fictitious operations are performed according to a two-option production routing. For more information refer to (Cimino, Negri and Fumagalli, 2019). The I4.0Lab is used to develop and improve the semantic data model in Section 3, by instantiating its classes to support DA.

Within the Lab, the drilling station represents one of the most critical asset and therefore it is monitored through different low-cost sensors (Cattaneo and Macchi, 2019). For the sake of shortness, only some of the instances of the classes within the physical and logical description block are summarised in Table 2.

Table 2: Some instances of semantic data model for physical and logical description blocks

Class	Attribute	Instance
Asset_ system	assetSystemID	1
	assetSystemName	I4.0Lab
	assetSystemType	Assembly sys.
Asset	assetID	1.3
	assetName	Drilling stat.
	assetType	Automated
	assetClass	Primary
	assetManufacturerName	Festo
Main-tainable_ item	maintainableItemID	1.3.1
	maintainableItemName	Drilling unit
	maintainableItemType	Primary
Sensor_ main-tainable_ item	sensorID	ws32sensor6
	sensorName	Vibration sens.
	sensorType	Accelerom.
	sensorManufacturerName	Raspberry-Pi
	sensorSamplingFreq	200 Hz
	sensorFullScale	-

There are no generic sensors installed, so they are not reported in the above Table 2.

Table 3 describes some instances of **Failure_mode** class.

Table 3: Instances of Failure_mode class

Class attribute	Instance
failureModeID	f5
failureModeName	tool breakage
failureModeEffect	low hole quality
failureModeID	f8
failureModeName	tool asymmetry
failureModeEffect	low hole quality
failureModeID	f12
failureModeName	handling Z axis
failureModeEffect	stop of tool movement

The function (**Asset_function**) performed is to realise two or four holes on the cover of the product. The **Functional_failure** is the inability to perform the holes.

The information sources block implies the classes mainly related to the outputs of sensors (**Operation_variable** and **Environment_variable**) or to the inputs to the MES (Manufacturing Execution System) that describes the **Asset_working_regime** as a set of parameters carried by the **Regime_variable**. Examples of **Operation_variable** are reported in Table 4, related to vibration measured on the drilling unit: their values are saved in an open-source document database, named MongoDB (mongodb.com).

Table 4: Instances of Operation_variable class

Class attribute	Instance
operationVariableID	1.3.1.x
operationVariableName	x vibration
operationVariableUnitMeasure	Hz
operationVariableType	waveform
operationVariableValue	*not reported here*
operationVariableID	1.3.1.y
operationVariableName	y vibration
operationVariableUnitMeasure	Hz
operationVariableType	waveform
operationVariableValue	*not reported here*
operationVariableID	1.3.1.z
operationVariableName	z vibration
operationVariableUnitMeasure	Hz
operationVariableType	waveform
operationVariableValue	*not reported here*

The regime variables are hidden behind the user interface of the MES, while, as anticipated, no environment variable is available as relative sensors are missing.

Now, having fixed physical and logical description, as well as the information sources, the DA can start. To this end, it is important to specify that the dataset is built for drilling station running at **Asset_working_regime** <1, two holes on the left side of the cover>. First, an activity of **Algorithm_preprocessing_data_fusion** is required: in our PoC, in fact we face time alignment problems. The availability of raw values of the **Operation_variable**(s) allows then the generation of different features (**Algorithm_feature_generation**): Skewness, Kurtosis, and mean absolute deviation, among many others. These represent instances of the **Feature** class in the semantic data model.

The PCA (**Algorithm_dimensionality_reduction**) is then applied to retain only one or more **Significant_features**. For the drilling station, the RMS (Root Mean Square) is the most significant feature, able to describe most of the variability (for the failure mode experimented in the case).

In decision-making, the **State_detection_analysis** is then performed by means of a **Statistical_algorithm**, namely Statistical Process Control (Bersimis, Psarakis and Panaretos, 2007): the **State_detection** allows to describe the **Asset_working_state** through the RMS only. At this step, the **Novelty_detection** may be implemented using different techniques. Different trials are performed and all show that there are two health states for the failure mode experimented in the case (**Health_state_division**), based on the RMS value. Therefore, the SPC is used, with a threshold RMS^{Up} set to be $\mu + 3\sigma$ (Fumagalli *et al.*, 2019). If the monitored RMS goes above this limit, a novelty is present, and the asset is going towards an abnormal/faulty state (see Figure 7). **Novelty_detection** may then bring to a **Reactive_action** on the maintainable item or asset.

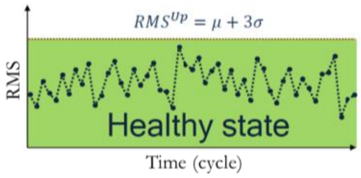

Figure 7: Control chart for RMS for state detection

The **Prognosis_analysis** uses **Algorithm** to forecast a **Predicted_health_indicator**, i.e. the RMS for the drilling station, and the **Remaining_useful_life**. In so doing, it contributes to enrich the **Health_state_division**. More specifically, the RMS as the **Predicted_health_indicator** is modelled through the Exponential Degradation Model (Gebraeel *et al.*, 2005).

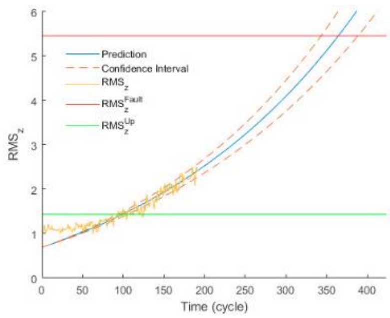

Figure 8: RMS prediction (Cattaneo and Macchi, 2019)

Remaining_useful_life and **Health_state_prediction** allow to take a decision, that is **Proactive_action**, on the maintainable item or asset to prevent future stops.

5.Conclusions

This research work aims at proposing a semantic data model that guides DA to support the PHM process. The model is developed in UML as main formalism adopted in the scientific literature. Data models are in fact used for several objectives in maintenance, but mainly for CBM. Predictive maintenance has room for improvement and this paper is demonstrating a proof. Herein, data modelling helps in reaching goals mainly related to data contextualisation and completeness to support DA. The distinction between operation, environment and regime variables is a core part of the model as it provides a contextualisation to the available data. Data are interpreted before starting the entire DA path, with ML, statistical and physics algorithms and, in the authors' experience, this empowers the application of suitable algorithms for prediction, also coping with the challenge of time-varying operating conditions. Also, the formalisation of all the data needed helps in preparing and checking data quality, namely supporting the data completeness. Overall, the semantic data model provides background semantics to DA for PHM. In so doing, the model could guide DA to monitor current asset state and predict future degradation within the PHM process.

The main limitation of the built semantic data model is its "static" nature. The model must be instantiated manually; moreover, the asset or maintainable item states (health states and working states) must be defined a priori and the results from the DA do not directly influence the model. But it would be interesting that the model is able to update its classes integrating the new knowledge every time this emerges from DA. This barrier could be broken through the application of ontology to enable an automated (machine-readable) semantic-driven predictive maintenance. The extension of the concept of the sematic data modelling towards ontology engineering may in fact be useful to introduce reasoning capabilities and auto-update potentialities, supporting DA more proactively geared to domain-related knowledge.

References

Bersimis, S., Psarakis, S. and Panaretos, J. (2007) 'Multivariate statistical process control charts: an overview', Quality and Reliability Engineering International. Wiley Online Library, 23(5), pp. 517–543.

Campos, M. A. L. et al. (2010) 'UML model for integration between RCM and CBM in an e-Maintenance architecture', in IFAC Proceedings Volumes, pp. 110–115. doi: 10.3182/20100701-2-PT-4012.00020.

Cattaneo, L. and Macchi, M. (2019) 'A Digital Twin Proof of Concept to Support Machine Prognostics with Low Availability of Run-To-Failure Data', IFAC-PapersOnLine. Elsevier Ltd, 52(10), pp. 37–42. doi: 10.1016/j.ifacol.2019.10.016.

Cheng, Y. et al. (2018) 'Data and knowledge mining with big data towards smart production', Journal of Industrial Information Integration. Elsevier, 9, pp. 1–13.

Cimino, C., Negri, E. and Fumagalli, L. (2019) 'Review of digital twin applications in manufacturing', Computers in Industry. Elsevier B.V., 113, pp. 103–130. doi: 10.1016/j.compind.2019.103130.

Embley, D. W. (2009) 'Semantic Data Model BT - Encyclopedia of Database Systems', in LIU, L. and ÖZSU, M. T. (eds). Boston, MA: Springer US, pp. 2559–2561. doi: 10.1007/978-0-387-39940-9_105.

Fumagalli, L. et al. (2019) 'Data-driven CBM tool for risk-informed decision-making in an electric arc furnace', International Journal of Advanced Manufacturing

Technology. Springer, pp. 1–14. doi: 10.1007/s00170-019-04189-w.

Gartner (2019) Gartner IT Glossary. Available at: https://www.gartner.com/it-glossary/ (Accessed: 23 March 2019).

Gebraeel, N. Z. et al. (2005) 'Residual-life distributions from component degradation signals: A Bayesian approach', IiE Transactions. Taylor & Francis, 37(6), pp. 543–557.

Guillén, A. J. et al. (2016) 'On the role of Prognostics and Health Management in advanced maintenance systems', Production Planning and Control. Taylor & Francis, 27(12), pp. 991–1004. doi: 10.1080/09537287.2016.1171920.

Hassanain, M. A., Froese, T. M. and Vanier, D. J. (2000) 'IFC-based data model for integrated maintenance management', in Computing in Civil and Building Engineering, pp. 796–803.

ISO 13374-1 (2003) 'Condition Monitoring and Diagnostics of Machines — Data Processing, Communication and Presentation— Part 1: General guidelines', British Standards Institution, 3, p. 31.

Ji, W. and Wang, L. (2017) 'Big data analytics based fault prediction for shop floor scheduling', Journal of Manufacturing Systems. Elsevier, 43, pp. 187–194.

Keet, C. M. (2018) An introduction to ontology engineering. University of Cape Town.

Khaleghi, B. et al. (2013) 'Multisensor data fusion : A review of the state-of-the-art', Information Fusion. Elsevier B.V., 14(1), pp. 28–44. doi: 10.1016/j.inffus.2011.08.001.

Kiritsis, D. (2013) 'Semantic technologies for engineering asset life cycle management', International Journal of Production Research. Taylor & Francis, 51(23–24), pp. 7345–7371. doi: 10.1080/00207543.2012.761364.

Lee, J. et al. (2015) 'Industrial Big Data Analytics and Cyber-physical Systems for Future Maintenance & Service Innovation', Procedia CIRP. Elsevier B.V., 38, pp. 3–7. doi: 10.1016/j.procir.2015.08.026.

Lee, J., Jin, C. and Bagheri, B. (2017) 'Cyber physical systems for predictive production systems', Production Engineering. Springer, 11(2), pp. 155–165.

Lei, Y. et al. (2018) 'Machinery health prognostics: A systematic review from data acquisition to RUL prediction', Mechanical Systems and Signal Processing. Elsevier, 104, pp. 799–834.

Luisi, J. V (2014) 'Part IV - Information Architecture', in Luisi, J. V. B. T.-P. E. A. (ed.) Strategies to Transform Information Systems in the Era of Big Data. Boston: Morgan Kaufmann, pp. 189–261. doi: 10.1016/B978-0-12-800205-6.00004-4.

Macchi, M. et al. (2016) 'Economic and environmental impact assessment through system dynamics of technology-enhanced maintenance services', International Journal of Industrial and Systems Engineering, 23(1), pp. 36–56. doi: 10.1504/IJISE.2016.075806.

Márquez, A. C., Del Castillo, A. C. and Fernández, J. F. G. (2020) 'Integrating artificial intelligent techniques and continuous time simulation modelling. Practical predictive analytics for energy efficiency and failure detection', Computers in Industry. Elsevier, 115, p. 103164.

Negri, E. et al. (2016) 'Requirements and languages for the semantic representation of manufacturing systems', Computers in Industry, 81, pp. 55–66. doi: 10.1016/j.compind.2015.10.009.

Negri, E. et al. (2017) 'Modelling internal logistics systems through ontologies', Computers in Industry, 88, pp. 19–34. doi: 10.1016/j.compind.2017.03.004.

Nuñez, D. L. and Borsato, M. (2018) 'OntoProg: An ontology-based model for implementing Prognostics Health Management in mechanical machines', Advanced Engineering Informatics. Elsevier, 38(October), pp. 746–759. doi: 10.1016/j.aei.2018.10.006.

Pimentel, M. A. F. et al. (2014) 'A review of novelty detection', Signal Processing. Elsevier, 99, pp. 215–249. doi: 10.1016/j.sigpro.2013.12.026.

Polenghi, A. et al. (2019) 'Conceptual framework for a data model to support Asset Management decision-making process', in Ameri, F. et al. (eds) Advances in Production Management Systems. Production Management for the Factory of the Future. APMS 2019. IFIP Advances in Information and Communication Technology. Springer, Cham, pp. 283–290. doi: 10.1007/978-3-030-30000-5_36.

Si, X.-S. et al. (2011) 'Remaining useful life estimation–a review on the statistical data driven approaches', European Journal of Operational Research. Elsevier, 213(1), pp. 1–14.

Tam, A. and Kwan, I. (2019) 'Data Quality in Asset Management—Creating and Maintaining a Foundation for Data Analytics', in Asset Intelligence through Integration and Interoperability and Contemporary Vibration Engineering Technologies. Springer, pp. 567–574.

Wang, Hong et al. (2019) 'Early fault detection of wind turbines based on operational condition clustering and optimized deep belief network modeling', Energies. Multidisciplinary Digital Publishing Institute, 12(6), p. 984.

West, M. (2011) Developing high quality data models. Elsevier.

Zhu, J. et al. (2018) 'Review and big data perspectives on robust data mining approaches for industrial process modeling with outliers and missing data', Annual Reviews in Control. Elsevier, 46, pp. 107–133.