Big Data Summarisation and Relevance Evaluation for Anomaly Detection in Cyber Physical Systems

Conference Paper · October 2017

DOI: 10.1007/978-3-319-69462-7_28

5 authors, including:

Ada Bagozi
Università degli Studi di Brescia
20 PUBLICATIONS **149** CITATIONS
SEE PROFILE

Devis Bianchini
Università degli Studi di Brescia
44 PUBLICATIONS **278** CITATIONS
SEE PROFILE

Valeria De Antonellis
Università degli Studi di Brescia
225 PUBLICATIONS **2,778** CITATIONS
SEE PROFILE

Alessandro Marini
Università degli Studi di Brescia
7 PUBLICATIONS **102** CITATIONS
SEE PROFILE

Big Data Summarisation and Relevance Evaluation for Anomaly Detection in Cyber Physical Systems

Ada Bagozi, Devis Bianchini, Valeria De Antonellis, Alessandro Marini, Davide Ragazzi

Dept. of Information Engineering University of Brescia
Via Branze, 38 - 25123 Brescia (Italy)

Abstract. Recent advances in the smart factory created new opportunities in industrial support, specifically in anomaly detection and asset management for maintenance purposes. Data collected from machines in operation is integrated in cyberspace with advanced cockpit and dashboard visualisation tools, as well as computerised maintenance management systems (CMMS). This paves the way to collaborative environments for Cyber Physical Systems, that assist maintenance operators in making better decisions while dealing with real time data in a dynamic context of interconnected systems. To this aim, models and techniques for data representation and treatment are highly required. In this paper, we propose a state detection service for Cyber Physical Systems, able to identify anomalies based on large amounts of data incrementally collected, organized and analysed on-the-fly. The service combines in a novel way data summarisation and data relevance techniques, to focus the computation on relevant data only, as well as a multi-dimensional model, that organises summarised data according to multiple dimensions, for flexible anomaly detection according to different analysis requirements. A pilot case study in the smart factory is also described, to demonstrate the applicability of the approach.

Keywords: Industry 4.0, Cyber Physical Systems, state detection, big data, data relevance, data summarisation, data exploration

1 Introduction

In the new industrial evolution, as known as Industry 4.0, the strict interaction between physical spaces (embedded systems, sensors, mobile and wearable devices, RFID technology) and cyberspace (edge and cloud computing technologies), represented by Cyber Physical Systems (CPS), made the exchange of large quantity of data in (near) real-time between interconnected systems a reality. In the field of manufacturing, this created the opportunity of new applications, aimed to improve operation process performance, monitoring and control, anomaly detection and health assessment [9].

This trend is promoting the development of collaborative applications, with the increasing inclusion of human being in the computational ambient or cyberspace [5]. For what concerns anomaly detection, data collected from machines in operation is integrated in cyberspace with visualisation tools such as cockpits and dashboards, as well as computerised maintenance management systems (CMMS). CMMS assists maintenance operators in planning activities to prevent downtime and low performance. Visualisation tools are in charge of providing different views over data, allowing for flexible anomaly detection according to different analysis requirements. Used together in a collaborative environment, these tools pave the way in making better decisions, while dealing with real time data in a dynamic context of interconnected systems. Current anomaly detection strategies generally rely on costly models built on top of experts' experience. In this context, big data and data-driven approaches are mainly unexploited [15]. Data-intensive CPS need tools and methods to deal with huge quantity of data, collected at high rate, and efficient techniques for storing and managing it. Data value declines very quickly, making organisations' wealth more and more dependent on how efficiently they can turn collected data into actionable insights [8]. The current validity of data is a critical success factor to be considered for implementing effective anomaly detection solutions. Indeed, the real value of anomaly detection using CPS concepts with big data stands in the fact that these techniques seem to be effective to identify unknown anomalies, helping industrial analysts and operators in the resolution of possible invisible problems [11].

In this paper, we propose a state detection service for Cyber Physical Systems, able to identify anomalies based on large amounts of data incrementally collected, organized and analysed on-the-fly. The aim is to demonstrate how data summarisation and data relevance techniques, proposed in [3] as ingredients to perform exploration of real time data in a dynamic context of interconnected systems, can be used in a novel way to identify anomalies in CPS. In particular, we will show how data relevance techniques, that focus the computation and monitoring on relevant data only, and a multi-dimensional model, that organises summarised data according to multiple dimensions, can be adapted for flexible anomaly detection according to different analysis requirements. In [2] we proposed IDEAaS (Interactive Data Exploration As-a-Service), a framework where innovative services are designed to enable data exploration. Here we focus our attention on the application of the multi-dimensional model, data summarisation and relevance evaluation techniques to support anomaly detection in collaborative systems in the context of Cyber Physical Systems and Industry 4.0.

The paper is organised as follows: in Section 2 we introduce the general idea and motivations behind a big data-driven state detection service; Section 3 contains the description of the multi-dimensional model; in Section 4, data summarisation and relevance evaluation techniques are described with reference to anomaly detection; Section 5 presents the implementation details and preliminary evaluation of the state detection service using a pilot study in the smart factory; cutting-edge features of our approach, compared to the literature, are discussed in Section 6; finally, Section 7 closes the paper.

2 Big data-driven anomaly detection

Motivating collaborative scenario. Figure 1 depicts a collaborative scenario that motivates our work on anomaly detection for CPS. In the considered scenario, an Original Equipment Manufacturer (OEM) supplies a multi-spindle machine, designed to perform flexible tasks for its clients (manufacturing enterprises in sectors like automotive, aviation, water industry). As shown in figure, the machines are equipped with three identical spindles, that work independently each others on the raw material. Each spindle is mounted on a unit moved by an electric motor to perform X, Y, Z movements. The spindle rotation is impressed by another electric motor and its rotation speed is controlled by the machine control. Spindles use different tools (that are selected according to the instructions specified within the Part Program) in order to complete different steps in the manufacturing cycle. Spindle precision, working performances, as well as minimisation of tool breaks and machine downtime are critical factors. Events to be detected and avoided regard 'spindle rolling friction torque increase' and 'tool wear'. Increase of spindle rolling friction torque may happen for lack of lubrication or other mechanical wears like bearings damage. Tool wear monitoring is referred to possible tool usage optimisation in order to balance the trade-off between the number of tools used and the risk of breaking the tool during operations that may lead to long downtimes. Data collected from the physical system is saved on the cyber side (*cloud manufacturing* component) through a *data acquisition service* and is processed by a *state detection service*. CPS is represented in the cyber-space through a set of measures, that may be collected by sensors. FMEA analysis is useful to identify possible failure modes, the relative criticality and how this can be translated as collected measures. This analysis is performed in collaboration with mechanical designers and actors at business/management level, and helps to identify the set of feature spaces of interest. In our case study, for each unit, we measure the velocity of the three axes (X, Y and Z) and the electrical current absorbed by each motor, the value of rpm for the spindle, the percentage of power absorbed by the spindle motor (charge coefficient).

Spindle rolling friction torque increase and tool wear can be monitored by observing the spindle power absorption for similar rpm. If an increased power absorption is detected disregarding the tool that is used, it is possible to identify spindle rolling friction torque increase as the possible anomaly that increases the energy request to perform the manufacturing operations. If the increase in absorbed power is related only to the usage of a particular tool, this can be recognised as a symptom of a possible exceeding tool wear. Therefore, aspects such as machine components and tools, as well as time, represent multiple perspectives to perform state detection. The state detection service we describe in this paper may interact with other modules at the application level: (i) remotely, OEM is equipped with visualisation tools such as cockpits or dashboards and computerised maintenance management systems (CMMS), to assist operators in planning activities to prevent downtime and low operation performance, according to alerts coming from the state detection service; (ii) internally, the OEM client may use service outputs to plan supply chain activities (e.g., planning of

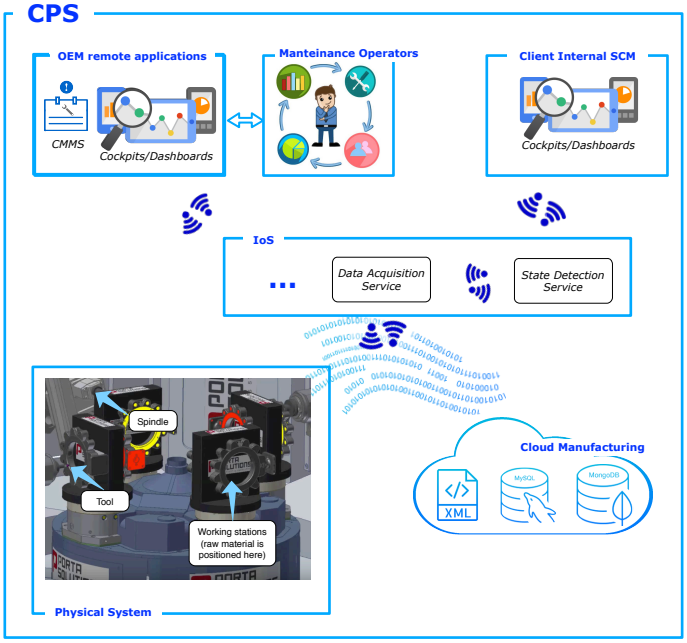

Fig. 1. A collaborative scenario for big data-driven anomaly detection of Cyber Physical Systems.

material and component orders). These collaborative needs raise a set of open issues presented in the following.

Data modeling. Collected data must be properly stored and organised on the cloud manufacturing in order to perform efficient data-driven tasks [10]. A data-driven state detection service also depends on different analysis requirements. OEM maintenance operators might be interested on spindle rolling friction torque increase, to manage maintenance activities. On the other hand, the OEM client might monitor spindle power absorption to detect tool wear events, to better plan the supply of new tools. Therefore, anomaly detection might rely on data modeling according to "facets" (e.g., categories), evenly hierarchically organised, that represent a powerful mean to enable aggregation of data according to different perspectives, in turn related to distinct observed problems and maintenance goals, although related to overlapping data. In the motivating example, the OEM client might observe spindle power absorption during the use of multiple tools, in order to detect a tool wear event. This gives the opportunity of managing different distributed activities on the same system in a cooperative way. As described in the next sections, we will combine different technologies in order to provide a multi-dimensional representation over the large amount of collected data.

Table 1. Public methods of the state detection service.

Method	Inputs	Ouputs
`GetData`	Range of timestamps, filters (required analysis dimensions)	Sends summarised data
`GetRelevantData`	/	Sends summarised data that the system recognised as relevant
`GetAlertStatus`	Current timestamp	Reports on current status of the system
`SendAlert`	/	Sends alerts on detected status

Data volume and velocity. In the considered pilot case study, we collected 140 millions of records from three machines, each one equipped with three spindles and different tools. The machines are identical.Records have been collected every 200ms. The ability of providing a compact view over the huge amount of data collected from the machine is strongly required. A data summarisation approach is recommended, where data should be observed in an aggregated way, instead of monitoring each single data record, that might be not relevant given the high level of noise in the working environment (slight variations in the measured variables). Moreover, data summarisation might have positive effects on visualisation tools. At the same time, data aggregations should be observed on the fly, given the highly dynamic nature of the application domain, and efficient computation algorithms are required to summarise data.

Data relevance. The prompt identification of anomalies by monitoring and observing collected data is one of the most important aspects to be addressed in state detection services. Data relevance evaluation techniques may help to iteratively restrict the monitoring only to the relevant measures that correspond to anomalies. This also have a positive impact on the algorithm complexity and response times, that might determine the success of an anomaly detection solution compared to the other ones. A definition of *relevance*, related to a notion of distance from an expected status, is fundamental in this case. Relevance evaluation algorithms must take into account volumes and speed of data collection phase. The dimensions of the multi-dimensional model have been considered to limit records and directions on which data summarisation and relevance detection process is applied.

Behind these open issues, heterogeneity of incoming records, as well as management of missing values have to be addressed as well, in order to prepare data for summarisation and relevance evaluation. These latter issues will be mentioned in the concluding remarks. In the following sections we will describe the models and techniques on which the state detection service relies to provide the methods described Table 1.

3 A multi-dimensional data model for state detection

The state detection service is based on a multi-dimensional model, that organises data according to different analysis dimensions, thus allowing for flexible anomaly detection according to distinct analysis requirements. The model enables the propagation of the system status over specific dimensions. Figure 2 shows the conceptual schema of the multi-dimensional model. The core concepts of the model are *features* and *measures*, defined in the following. They are organised according to *feature spaces*, *domain-specific dimensions* and *context parameters*, that constitute the overall set of dimensions on which the multi-dimensional data model is built.

Definition 1 (Feature). *A feature represents a monitored variable that can be measured. A feature F_i is described as $\langle n_{F_i}, u_{F_i} \rangle$, where n_{F_i} is the feature name, u_{F_i} represents the unit of measure. Let's denote with $F = \{F_1, F_2 \dots F_n\}$ the overall set of features.*

In the considered case study, examples of features are the velocity of the three axes X, Y and Z, the electrical current, the value of spindle rpm and the percentage of absorbed power.

Definition 2 (Measure). *We define a measure for the feature F_i as a scalar value $X_i(t)$, expressed in terms of the unit of measure u_{F_i}, taken at the time t.*

To observe different physical phenomena of a system, multiple features can be monitored together. We call such sets of features as feature spaces.

Definition 3 (Feature space). *A feature space conceptually represents a set of related features, that are jointly measured to observe a physical phenomenon. Multiple feature spaces might be observed, and the observation of a feature might be useful to monitor more than one feature space. We denote with $FS = \{FS_1, FS_2, \dots FS_m\}$ the set of feature spaces, where $FS_j \subseteq F$ and $m \leq n$. Given a feature space $FS_j = \{F_1, F_2, \dots F_h\}$, we denote with the vector $\boldsymbol{X}_j(t)$ a record of measures $\langle X_1(t), X_2(t), \dots X_h(t) \rangle$ for the features in FS_j, synchronised with respect to the timestamp t. Feature spaces can be monitored independently each others.*

In the considered case study, spindle rolling friction torque increase and tool wear can be observed by monitoring the spindle power absorption as a feature space. Feature spaces can be monitored according to different domain-specific dimensions, such as the observed machine or the tool used during manufacturing, defined as follows.

Definition 4 (Domain-specific dimension). *We denote with $\mathcal{D}$ the multi-dimensional space created by p domain-specific dimensions $\mathcal{D}_1, \dots \mathcal{D}_p$, where $\mathcal{D} = \mathcal{D}_1 \times \dots \times \mathcal{D}_p$. Dimensions can be organized in hierarchies, at different levels. We denote with $\mathcal{D}_j^i$ the i-th level in the hierarchy of j-th dimension and with $d_i \in \mathcal{D}_i$ a single instance of the dimension $\mathcal{D}_i$.*

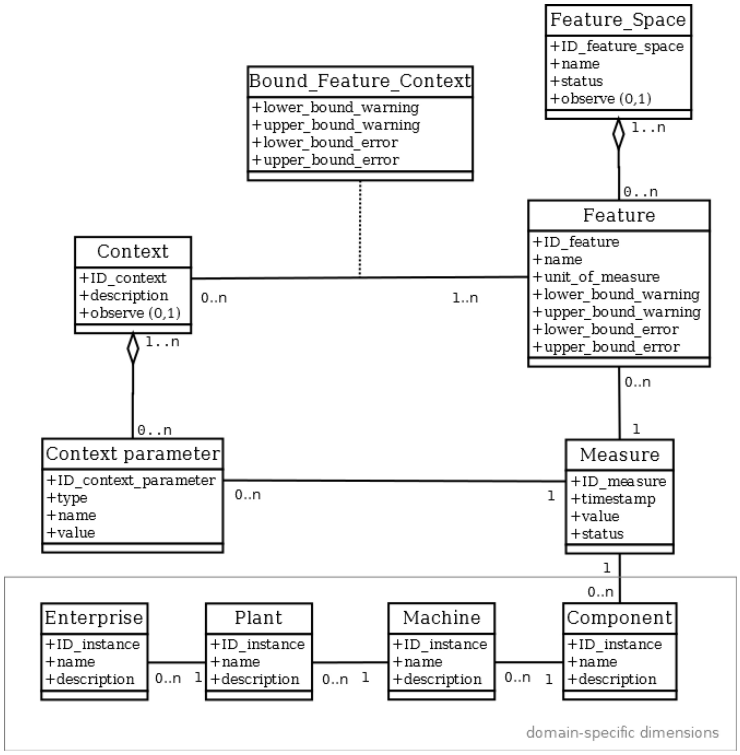

Fig. 2. Conceptual schema of the multi-dimensional model on which the state detection service relies.

For example, tools can be aggregated into tool types; in the hierarchy of the monitored physical system, components (e.g., spindles) can be aggregated into the machines they belong to, in turn organised into plants and enterprises. Furthermore, there are some characteristics that should remain constant while performing any kind of comparison between measures. For instance, spindle rolling friction torque increase or tool wear should be observed only comparing the spindle power absorption under the same working conditions of the machine, such as the *part program* that is being executed, or the *working mode* (G0, fast movement of the spindle to catch the tool, or G1, slow movement of the spindle during the manufacturing). To this aim, we introduce the concept of *context parameter*.

Definition 5 (Context parameter). *We define a context Ctx_i as a set of parameters, that identify the conditions under which the monitored system operates. Comparison between different measures makes sense only within the same context. We denote with Ctx the set of all possible contexts $\{Ctx_i\}$. A context is composed of one or more Context Parameters (e.g., part program, working mode), each one described by type, name and value.*

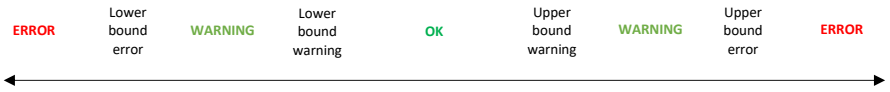

Fig. 3. System status and alert bounds.

3.1 System status and alert bounds

The goal of a state detection service is to detect anomalies and send alerts concerning the system status. We consider three different values for the *status*: (a) `ok`, when the system works normally; (b) `warning`, when the system works in anomalous conditions that may lead to breakdown or damage; (c) `error`, when the system works in unacceptable conditions or does not operate. Therefore, the warning status is used to perform an early detection of a potential deviation towards an error state. The migration of the system status from one value to the others raises an *alert* and occurs when one or more measures exceed a given bound, as shown in Figure 3. We consider four types of bounds: lower bound error, lower bound warning, upper bound warning and upper bound error. In our model, bounds are further classified as feature bounds and contextual bounds.

Definition 6 (Feature bound). *A feature bound represents a physical limit of the feature, independently from the specific context in which the monitored system is working. This bound is set as a range on the values of a feature according to the model in Figure 2.*

Definition 7 (Contextual bound). *A contextual bound represents the limit of a feature within a specific context where the feature is measured. The rationale is that, in a specific context (e.g., the working mode, the part program), when the system works normally, a feature should assume values within a specific range, that might be different by the physical limits for the same feature. These bounds are modelled as an association with attributes between the feature and the context entities in the model in Figure 2.*

These bounds set the ranges for the three different values of the status: `ok`, `warning` and `error`. Feature bounds determine the *absolute status* of a feature, while contextual bounds determine the *contextual status* of a feature. The system status (either absolute or contextual) can be propagated to the whole feature space and along the hierarchy of monitored physical system, according to the following propagation rules.

Propagation to the feature space. Consider a feature space $FS_j = \{F_1, F_2, \ldots F_h\}$, the value of the status associated to FS_j, given the status values for each feature $F_1, F_2, \ldots F_h$, is computed as follows:

- `ok`, if the status of each feature F_i, $\forall i = 1 \ldots h$, is `ok`;

- `warning`, if the status of at least one feature F_i, $\forall i = 1 \dots h$, is `warning`;
- `first level error`, if the status of at least one feature F_i, $\forall i = 1 \dots h$, is `error`;
- `second level error`, if the status of each feature F_i, $\forall i = 1 \dots h$, is `error`.

Propagation along the hierarchy of the monitored physical system. The value of a feature status for a component is propagated to the highest level of the hierarchy (machine, plant, enterprise) as follows: the status of the machine is

- `ok`, if the status for all its components is `ok`;
- `warning`, if the status of at least one component is `warning`;
- `first level error`, if the status of at least one component is `error`;
- `second level error`, if the status of all its components is `error`.

The same applies for the status of the plant (resp., enterprise), computed starting from the status of its machines (resp., plants).

Definition 8 (Multi-dimensional model). *We describe the multi-dimensional model as a set $\mathcal{V}$ of nodes. Each node $v \in \mathcal{V}$ is described as*

$$v = \langle \boldsymbol{X}_j(t), fs_j, d_1, d_2, \dots d_p, Ctx_i, \sigma_j, \sigma^c \rangle \tag{1}$$

where $\boldsymbol{X}_j(t)$ represents a record of measures taken at time t for the feature space fs_j, in the context Ctx_i, for the values $d_1, d_2, \dots d_p$ of domain-specific dimensions $\mathcal{D}_1, \dots \mathcal{D}_p$; σ_j is the status of the feature space fs_j; σ^c is the status of the component d_k at the lowest level of the monitored physical system dimension $\mathcal{D}_k$, with $k \leq p$. The status σ^c is propagated to the other levels of $\mathcal{D}_k$ using the rules presented above.

4 Data summarisation and relevance evaluation

In order to promptly detect anomalies in (near) real-time, the state detection service described in this paper relies on summarisation and relevance evaluation techniques. Details about these techniques have been presented in [3]. Here, we customise relevance evaluation techniques (Section 4.2), used in combination with the multi-dimensional model, to the anomaly detection problem.

4.1 Clustering-based data summarisation

We apply data summarisation techniques to summarise all records of measures collected during time interval Δt in the context Ctx_i and for dimensions $d_1 \in \mathcal{D}_1$, $d_2 \in \mathcal{D}_2 \dots \ d_p \in \mathcal{D}_p$, for monitoring feature space fs_j. To this aim, we denote with $\Sigma(\boldsymbol{X}_j(t), fs_j, d_1, d_2, \dots d_p, Ctx_i)$ the application of the summarisation procedure to the records of measures $\boldsymbol{X}_j(t)$ as used in Equation (1) of Definition 8. In our approach data summarisation is based on clustering-based techniques. When dealing with real time data, collected in CPS, we face with data streams,

where not all data is available since the beginning, but is collected in an incremental way. For these reasons, an incremental data-stream clustering algorithm has been developed. The clustering algorithm produces a set of clusters aimed to summarise collected measures in a time interval Δt. The clustering algorithm is performed in two steps: (i) in the first one, a variant of Clustream algorithm [1] is applied, that incrementally processes incoming data to obtain a *set of syntheses*; (ii) in the second step, X-means algorithm is applied [14] in order to cluster syntheses obtained in the previous step. X-means does not require any a-priori knowledge on the number of output clusters. Each synthesis provides a lossless summarisation of records through five elements: the number of records included into the synthesis, the vector representing the linear sum of measures, the quadratic sum of measures, the vector representing the centroid of the synthesis and the radius of the synthesis. The second step aims to cluster syntheses. Clusters give a balanced view of the observed physical phenomenon, grouping together syntheses corresponding to close data. A cluster is represented by its centroid and the set of syntheses belonging to it. Hereafter, we denote with $SC(\boldsymbol{X}_j(t), fs_j, d_1, d_2, \ldots d_p, Ctx_i)$ the set of identified clusters, that is, the output of $\Sigma(\boldsymbol{X}_j(t), fs_j, d_1, d_2, \ldots d_p, Ctx_i)$ procedure.

4.2 Relevance -based anomaly detection

Relevance-based techniques are used to detect components status over time. In literature, data relevance is defined as the *distance* from an *expected status*. The point is to define the expected status and how to compute such a distance. In our approach, the expected status corresponds to the set of clusters computed during normal working conditions for the monitored system. Let's denote with $\hat{SC}(\boldsymbol{X}_j(t), fs_j, d_1, d_2, \ldots d_p, Ctx_i)$ such cluster set. Data relevance is based on the notion of *cluster distance* between the current cluster set and $\hat{SC}(\cdot)$. This distance is defined as follows. Given a set of clusters $SC = \{C'_1, C'_2, \ldots, C'_n\}$ and the set of clusters $\hat{SC}(C_1, C_2, \ldots, C_n)$, we evaluate the distance between SC and $\hat{SC}(\cdot)$, denoted with $\Delta(SC, \hat{SC})$, by combining three factors: (i) the difference between the number of clusters, (ii) the distance between cluster centroids and (iii) the intra-cluster distances, i.e. the distance between syntheses belonging to the same cluster.

In particular, the relevance techniques allow to identify what are the clusters that changed over time. Let's denote with $\{\overline{C_i}\}$ such clusters. The distance is used to detect a *state change*. When $\Delta(SC, \hat{SC})$ exceeds a given threshold, data that is summarised in the clusters $\{\overline{C_i}\}$ is considered as relevant and, for each cluster in $\{\overline{C_i}\}$, the distance of cluster centroid from the warning and error bounds is computed. If this distance is equal or lower than the cluster radius, this means that a warning or error status has to be detected, according to the rules described in Section 3. Note that distance also helps to detect *potential* state changes. Consider for example Figure 4, that shows an example of cluster evolution over time for the smart factory case study. The figure shows how the cluster C_1 doesn't changed its position, as well as its size, from time t_n to t_{n+3}. On the other hand, cluster C_2 evolves from the wealth zone to the warning and

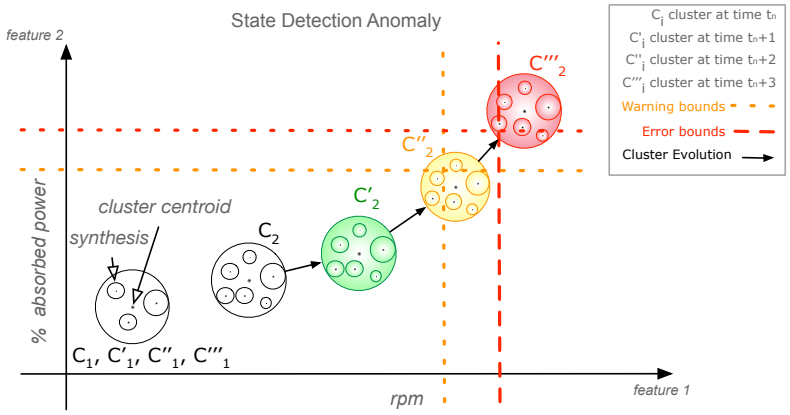

Fig. 4. Illustration of set of clusters changing over time. Distance techniques aim to detect changes in cluster set over time due to spindle rolling friction torque increase that may cause a decrease of rpm and an increase of the percentage of absorbed power.

error zones. At time t_{n+2} cluster C''_2 crosses the warning bound of rpm feature causing a warning alert, at time t_{n+3} cluster C'''_2 moves into the error zone, crossing error bounds of both the features considered. At time t_{n+1} cluster C'_2 still remains inside the wealth zone, however relevance techniques detected its change. Therefore cluster C'_2 is recognised as relevant and monitored to detect warning or error state changes. This allows for better performance of the anomaly detection algorithm, that focuses only on potential state changes. Figure 4 also shows that it is possible to identify the feature with respect to which the warning or error bound has been exceeded (e.g., among rpm and percentage of absorbed power).

5 Implementation and experimental evaluation

Figure 5 depicts a collaborative scenario, where the state detection service interacts with a cockpit to visualise the status of monitored physical system and with a CMMS to plan maintenance operations. Modules are implemented as RESTful services and use JSON as data exchange format. The figure shows the data flow across interoperating systems. Data is sent from the monitored physical system as an input for the *Data Cleaning & Normalisation* module. The input files can present different formats. In our current implementation it is composed of a set of records, where each record is associated with a timestamp, a set of measures for the considered features and a set of values for the context parameters and domain-specific dimensions. An XML configuration file (*Input Data Config*) contains all meta-data about the structure of incoming data records, after the application of data acquisition and ingestion techniques to face heterogeneity issues. The cleaned and normalised data is sent to the *Data Storage* module,

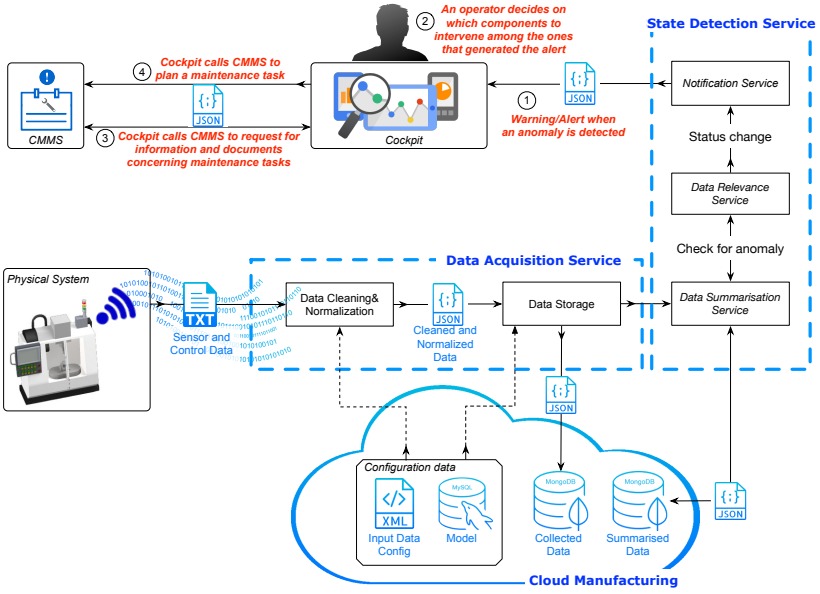

Fig. 5. Data flow in the anomaly detection collaborative scenario.

that is in charge of generating the JSON file for storing data in the *Summarised Data* collection (implemented using MongoDB). Figure 6 shows the structure of JSON documents stored within this collection. Information contained in this structure, together with involved data records, are sent as output of `getData` and `getRelevantData` methods of the state detection service.

The diagram in Figure 7 shows the interactions between the functional modules. The Data Acquisition service, that includes the cleaning and storage tasks, and the State Detection service operate in parallel. This prevents the state detection procedure from being a bottleneck for the data acquisition. The Data Acquisition service activates every Δt seconds the State Detection service, that is in charge of detecting and notifying anomalies as explained in the previous sections through the execution of *Data Summarisation*, *Data Relevance* and *Notification* services. Every Δt seconds the state detection service checks the system status using clustering and relevance evaluation techniques. If the relevance evaluation detects changes in data compared to the expected status, the service computes the new status as explained in Section 4.2. If the status is not changed with respect to the previous check, the system simply updates the data saved in MongoDB (Summarised Data collection) with the computed status value. If the status changed, the system updates the status in MongoDB, applies propagation rules to the feature space and along all the levels of the hierarchy of monitored physical system and notifies an alert message to the cockpit containing the new status, using the `SendAlert` method to report the detected anomaly. Cockpit will handle the communication with the CMMS: it requests for information and

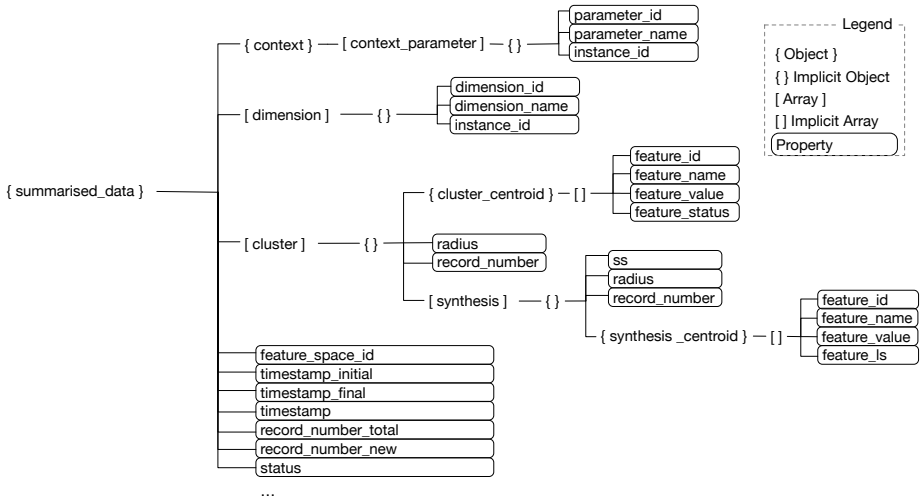

Fig. 6. The structure of the JSON document to store summarised data.

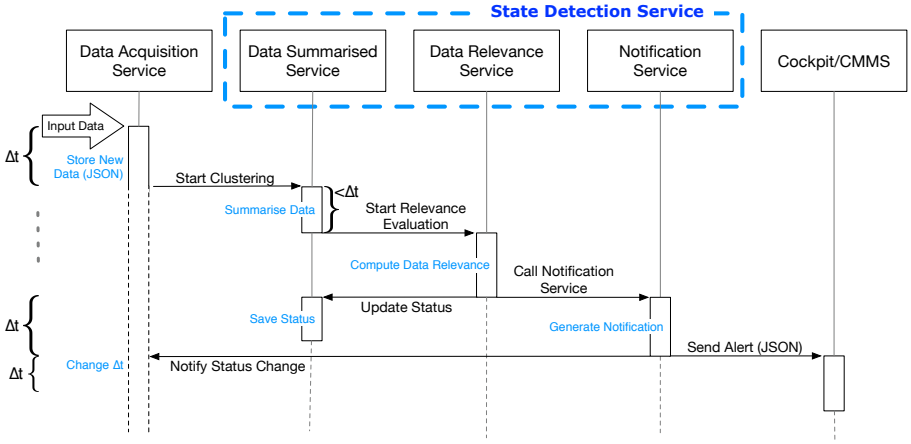

Fig. 7. Interactions between the modules of the functional architecture for the anomaly detection scenario.

documents regarding maintenance tasks and asks the CMMS to plan a maintenance task. It is worth to underline that, when the Data Relevance service detects a status change, it also calls the Data Acquisition module to modify the Δt value. In case of status changes from `ok` to `warning`, Δt decreases. The aim is to increment the frequency of controls when a potential breakdown might occur. This improves the performance of the system, at the cost of an increased frequency of status computation, as shown in the next section.

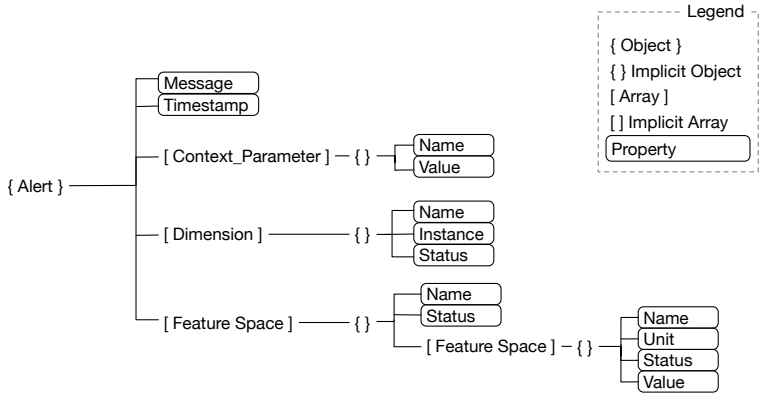

Fig. 8. Structure of JSON document as output of the `SendAlert` method.

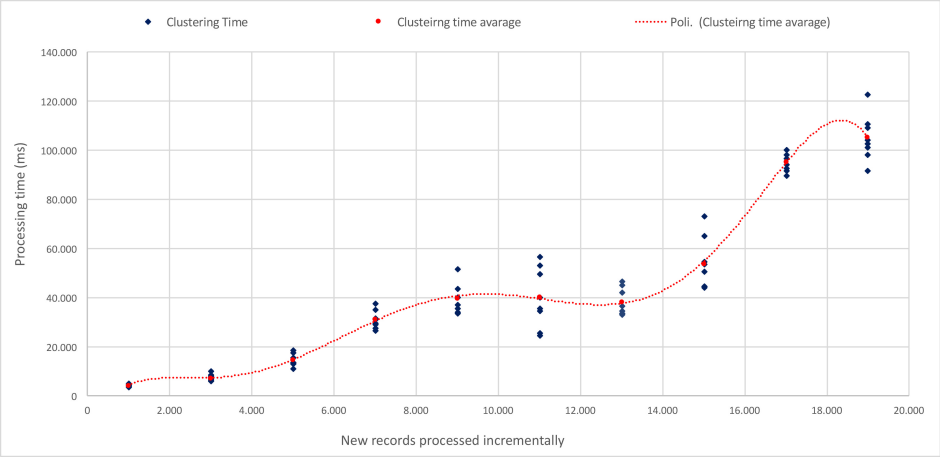

Fig. 9. Tests on the efficiency of clustering by changing the elaboration frequency ($\frac{1}{\Delta t}$).

Figure 8 shows the structure of the JSON file generated as output by the `SendAlert` method of the State Detection service, to be sent to the Cockpit. The `getAlertStatus` method returns a similar output, with the list of all current status values.

5.1 Experiments

We performed experiments on the State Detection service, in order to test its performance in terms of processing time and its effectiveness in quickly detecting anomalies. We collected 140 millions of records from the three machines considered for the case study, each one equipped with three spindles and different tools. Records have been collected every 200ms. We run experiments on

Working Days	Spindle 1		Spindle 2		Spindle 3	
	rpm decreas	energy consumption increase	rpm decreas	energy consumption increase	rpm decreas	energy consumption increase
01/08/2016 (operates normally)	/	/	/	/	/	/
02/08/2016	/	/	/	/	15%	16%
03/08/2016	20%	20%	/	/	29%	29%
04/08/2016	30%	30%	/	/	/	/
05/08/2016	40%	40%	20%	20%	/	/
06/08/2016	/	/	30%	30%	/	/
07/08/2016	/	/	20%	20%	/	/

Fig. 10. Introduced variation in collected records that simulates spindle rolling friction torque increase.

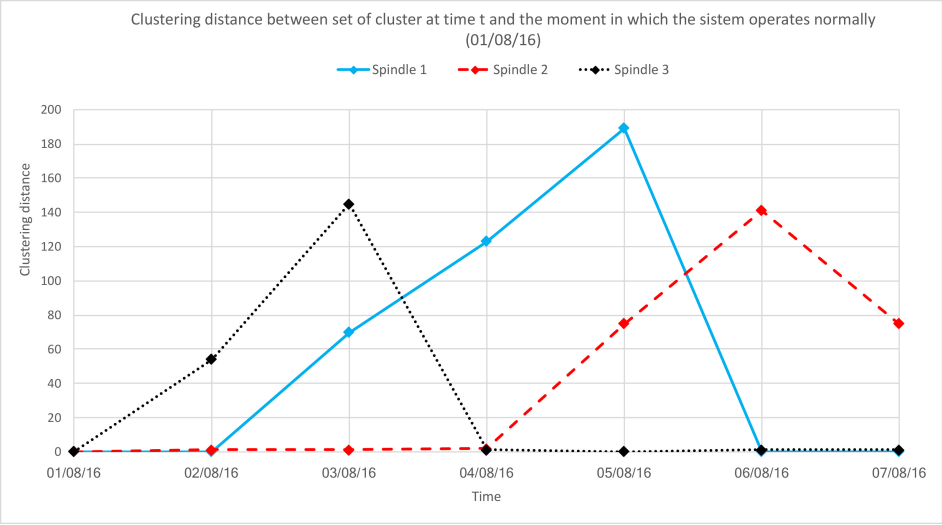

Fig. 11. Effectiveness evidences of the relevance evaluation techniques to identify changed records.

a VPS mounting Ubuntu Server 16.04, 1 vCore CPU, RAM 4GB, hard disk 20GB (SSD). Collected records of measures are saved within the *Collected Data* MongoDB as JSON documents.

We remark here that clustering and relevance evaluation are incrementally performed on slots of records on a time internal Δt. Also in the worst case, clustering, data extraction and cluster sets distance computation tasks were able to process $\sim 162 \times 10^3$ records in $\sim$170 seconds (corresponding to a processing rate of $\sim$953 records per second), thus facing an acceptable data input rate for the considered case study. These response times must be intended also when new data arrives. As expected, clustering is the most time-consuming task. Figure 9 shows further tests, where processing time for clustering is computed by varying

the number of records to which it is applied and, therefore, processing frequency $\frac{1}{\Delta t}$. As the number of new processed records every Δt seconds grows, clustering algorithm execution time presents a polynomial increment. We can conclude that the factor which impacts the most on the system performance is Δt and not the total number of processed data, thanks to the incremental approach. Therefore, when clustering processing time overtakes the data acquisition rate, a solution might be to tune the frequency $\frac{1}{\Delta t}$, as done during the interactions between services described above. Furthermore, higher frequency $\frac{1}{\Delta t}$ allows for faster detection of the state changes.

To test effectiveness of the state detection service, we introduced unexpected working states to simulate spindle rolling friction torque increase by increasing the spindle power absorption for similar rpm on a subset of collected data, as shown in Figure 10. Experimental results plotted in Figure 11 show the distance between sets of clusters calculated at time t and those calculated in case of normal working. The figure shows how the techniques proposed here allow to timely identify the unexpected situations induced in the system under observation.

6 Related Work

Several approaches in literature have been recently proposed to address anomaly detection in presence of big data collected in (near) real time. Authors in [15] present an approach for data-driven anomaly detection in manufacturing processes. They consider joint analysis of multiple variables for detecting the anomalies. They combine real-time and historical data processing in order to increase the agility for detecting and reacting to anomalies. This approach operates in two steps: (i) learning the normal behaviour of the system (based on past data), using a clustering technique (K-means algorithm) to group together close data; (ii) detecting at real-time an anomalous behaviour when new data does not belong to previously detected clusters. Similarly, the main goal of the project and the architecture presented in [7] is to facilitate the prompt detection or prediction of failures from event data. Machine learning is used to train data collected during regular execution of the manufacturing process in order to learn a probabilistic "normal model". Furthermore, authors focused on understanding causes of failures, using structured machine learning techniques, training classifiers from labeled event sequences. The architecture includes state-of-the-art distributed cloud-based big data technologies such as NoSQL databases. In [6] an approach based on in-memory big data processing is described. Anomaly detection is performed in two phases: (i) a preparation phase is used to generate a model of the system "usual state", by applying machine learning (pre-training) on stored data; (ii) an operation phase compares real-time incoming data with the "usual state" to output an anomaly index called "anomaly score" (where 0 indicates a system close to "usual state" and 1 indicates a system close to a "state different from usual").

The framework described in [13] guides maintenance decision makers on how to model the degradation process of a condition monitored device with an indi-

rectly observable multi-state degradation process (the status is a value in a range from "working perfectly" to "complete failure"). Then, it presents how to employ real-time condition monitoring data for online diagnostics and prognostics using some important measures. The framework is based on a flexible stochastic process for degradation modeling. In [16] statistical techniques based on the Tukey and Relative Entropy statistics are applied to provide online anomaly detection. The data is computed in streaming and doesn't rely on static profiling or limited sets of historical data. The statistical approach is implemented to exceed the limit of fixed thresholds. The approach involves measuring the changes in data distribution by windowing the data and using it to determine anomalies.

Compared to these approaches, we introduced the use of data relevance techniques to better focus the anomaly detection procedure, improving the overall performance of the approach. These techniques have been used on top of a complex multi-dimensional model, that adopts the concept of feature space to enable multi-parameter anomaly detection, as remarked in [15]. Multi-perspective anomaly detection for cooperative systems has been addressed in other application domains [4]. Nevertheless, in our model we also added the notion of domain-specific dimensions and context, to guide/filter the anomaly detection process. The application of data summarisation and relevance evaluation techniques on top of the multi-dimensional model constitutes a step forward compared to [12], where the notions of feature space and context have been introduced for the first time in the ambit of state detection services. Data summarisation techniques, combined with relevance evaluation, also prevent from using training datasets, that are often difficult to identify and select.

7 Concluding remarks

In this paper, we proposed a state detection service for Cyber Physical Systems working on large amounts of data incrementally collected, organized and analysed on-the-fly. The service relies on novel data summarisation and data relevance techniques: the former ones enable to deal with large amount of data in a dynamic environment, by providing a synthetic view over them; the latter ones focus the computation and monitoring on relevant data only. A multi-dimensional model, that organises summarised data according to multiple dimensions, allows for flexible anomaly detection according to different analysis requirements. The adoption of Big Data technologies for data acquisition and ingestion (e.g., Kafka or Pig), in order to face heterogeneity issues and missing records, will be investigated as future work. A case study in the smart factory is also described, where we discussed the interactions of the state detection service with: (a) a cockpit that provides different views over data; (b) a computerised maintenance management systems (CMMS), to assist maintenance operators in planning activities to prevent downtime and low operation performance. Although preliminary experiments are promising, future development will be focused on further improving the approach using technologies for streaming and parallel batch processing, such as Spark/Storm and Hadoop. Moreover, the state

detection service will be enhanced by introducing pattern recognition techniques to learn from the clusters evolution. This would in principle enable the implementation of health assessment strategies, on top of the ecosystem of models and techniques described in this paper.

References

1. C. Aggarwal, J. Han, J. Wang, and P. Yu. A framework for clustering evolving data streams. *Proc. of 29th Int. Conf. on Very Large Data Bases*, 81–92, 2003.
2. A. Bagozi, D. Bianchini, V. De Antonellis, A. Marini, and D. Ragazzi. Interactive Data Exploration As a Service for the Smart Factory. *Proc. of IEEE Int. Conference on Web Services (ICWS)*, 2017.
3. A. Bagozi, D. Bianchini, V. De Antonellis, A. Marini, and D. Ragazzi. Summarisation and Relevance Evaluation Techniques for Big Data Exploration: the Smart Factory case study. *Proc. of 29th Int. Conference on Advanced Information Systems Engineering (CAISE)*, 2017.
4. K. Böhmer and S. Rinderle-Ma. Multi-perspective Anomaly Detection in Business Process Execution Events. *Proc. of International Conference on Cooperative Information Systems (CoopIS)*, 80–98, 2016.
5. D. Gorecky, M. Schmitt, M. Loskyll, and D. Zuhlke. Human-machine interaction in the Industry 4.0 era. *IEEE Int. Conf. on Industrial Informatics*, 289–294, 2014.
6. T. Hanamori and T. Nishimura. Real-time Monitoring Solution to Detect Symptoms of System Anomalies. *FUJITSU Sci. Tech. Journal*, 52(4):23–27, 2016.
7. M. Huber, M. Voigt, and A. Ngomo. Big data architecture for the semantic analysis of complex events in manufacturing. *Proc. of GI Jahrestagung 2016*, 353–360, 2016.
8. S. Khalifa, Y. Elshater, K. Sundaravarathan, A. Bhat, P. Martin, F. Imam, D. Rope, M. Mcroberts, and C. Statchuk. The Six Pillars for Building Big Data Analytics Ecosystems. *ACM Computing Surveys*, 49(2), 2016.
9. J. Lee, H.D. Ardakani, S. Yang, and B. Bagheri. Industrial big data analytics and cyber-physical systems for future maintenance and service innovation. *Proc. of CIRP*, volume 38, 3–7, 2015.
10. J. Lee, B. Bagheri, and H. Kao. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. *Manuf. Letters*, 3:18–23, 2015.
11. J. Lee, E. Lapira, B. Bagheri, H. Kao. Recent advances and trends in predictive manufacturing systems in big data environment. *Manuf. Letters*, 1.1 (2013): 38-41.
12. A. Marini and D. Bianchini. Big Data As A Service For Monitoring Cyber-Physical Production Systems. *Proc. of 30th European Conference on Modelling and Simulation (ECMS)*, 579–586, 2016.
13. R. Moghaddass and M. J. Zuo. An integrated framework for online diagnostic and prognostic health monitoring using a multistate deterioration process. *Reliability Engineering & System Safety*, 124:92–104, 2014.
14. D. Pelleg and A. Moore. X-means: Extending K-means with Efficient Estimation of the Number of Clusters. *Proc. of 17th International Conference on Machine Learning (ICML)*, 727–734, 2000.
15. L. Stojanovic, M. Dinic, N. Stojanovic, and A. Stojadinovic. Big-data-driven anomaly detection in industry (4.0): an approach and a case study. *Proc. of IEEE International Conference on Big Data*, 1647–1652, 2016.
16. F. Wang and G. Agrawal. Effective and Efficient Sampling Methods for Deep Web Aggregation Queries. *Proc. of Conference on Extending Database Technology (EDBT)*, 425–436, 2011.