Ontology-Enhanced Machine Learning: A Bosch Use Case of Welding Quality Monitoring

Yulia Svetashova[1,2(✉)], Baifan Zhou[1,2(✉)], Tim Pychynski[1], Stefan Schmidt[1], York Sure-Vetter[2], Ralf Mikut[2], and Evgeny Kharlamov[3,4]

[1] Bosch Corporate Research, Renningen, Germany
svetashova@gmail.com,
{Baifan.Zhou,Tim.Pychynski,Stefan.Schmid5}@de.bosch.com
[2] Karlsruhe Institute of Technology, Karlsruhe, Germany
{York.Sure-Vetter,Ralf.Mikut}@kit.edu
[3] Bosch Center for Artificial Intelligence, Renningen, Germany
Evgeny.Kharlamov@de.bosch.com
[4] University of Oslo, Oslo, Norway
Evgeny.Kharlamov@ifi.uio.no

Abstract. In the automotive industry, welding is a critical process of automated manufacturing and its quality monitoring is important. IoT technologies behind automated factories enable adoption of Machine Learning (ML) approaches for quality monitoring. Development of such ML models requires collaborative work of experts from different areas, including data scientists, engineers, process experts, and managers. The asymmetry of their backgrounds, the high variety and diversity of data relevant for quality monitoring pose significant challenges for ML modeling. In this work, we address these challenges by empowering ML-based quality monitoring methods with semantic technologies. We propose a system, called SemML, for ontology-enhanced ML pipeline development. It has several novel components and relies on ontologies and ontology templates for task negotiation and for data and ML feature annotation. We evaluated SemML on the Bosch use-case of electric resistance welding with very promising results.

1 Introduction

Industry 4.0 [16] and technologies of the Internet of Things (IoT) [13] behind it lead to unprecedented growth of data generated during manufacturing processes [3,35]. Indeed, modern manufacturing machines and production lines are equipped with sensors that constantly collect and send data and with control units that monitor and process these data, coordinate machines and manufacturing environment and send messages, notifications, requests. Availability of

Y. Svetashova and B. Zhou—Contributed equally to this work as first authors.

J. Z. Pan et al. (Eds.): ISWC 2020, LNCS 12507, pp. 531–550, 2020.
https://doi.org/10.1007/978-3-030-62466-8_33

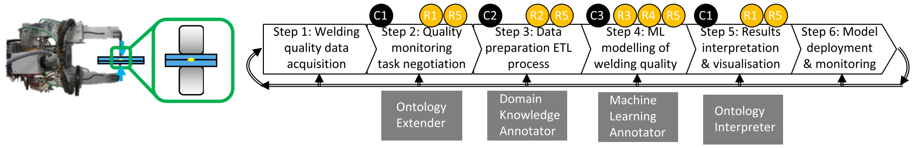

Fig. 1. A machine for automated welding (left) and an ML workflow enhanced with our semantic modules for welding quality monitoring (right). C stands for *challenges*, and R for *requirements*, see Sect. 1. ETL stands for *Extract, Transform, Load.* (Color figure online)

these voluminous data has led to a large growth of interest in data analysis for a wide range of industrial applications [25,26,41], especially the use of Machine Learning (ML) approaches for *monitoring* manufacturing processes, machines, and products by predicting machines' down-times or the quality of manufactured products [37].

Consider an example of *welding quality monitoring* at Bosch, where welding is performed with machines as shown in Fig. 1 (left) to connect pieces of metal together by pressing them and passing high current electricity through them [6]. For the purpose of developing ML approaches for welding quality monitoring, Bosch adopts the workflow slightly adjusted from [7,27] as schematically depicted in Fig. 1 (right). The workflow is iterative and includes data collection (Step 1), task negotiation, to define feasible and economic tasks (Step 2), data integration, to integrate data from different conditions and factories (Step 3), ML model development (Step 4), result interpretation and model selection (Step 5), and finally, model deployment in production (Step 6).

Development of such ML approaches is a complex and costly process where the following three challenges are of high importance for Bosch since they consume more than 80% of the overall time of development. The first challenge (C1) is *communication*: Steps 2 and 5 of welding quality monitoring require collaborative work of experts from different areas, including data scientists, engineers, process experts, and managers that have asymmetric backgrounds, which makes communication time consuming and error-prone. The second challenge (C2) is *data integration*: Step 3 requires to integrate data from dozens of sources with highly manual modification. The third challenge (C3) is *generalisability* of ML quality models: each ML model developed in Step 4 is typically tailored to a specific dataset and one welding process. Thus, reuse of this ML model for other data or processes requires a significant effort, while the reuse is highly desired, considering Bosch's wide spectrum of processes, equipment, and locations. In Fig. 1 we annotated Steps 2–5 with the challenges as C1–C3.

In this work we address these three challenges by enhancing machine learning development for quality monitoring with semantic technologies that have recently gained a considerable attention in industry for a wide range of applications and automation tasks such as modelling of industrial assets [18] and industrial analytical tasks [21], integration [11,19,20] and querying [32] of production data, and for process monitoring [29] and equipment diagnostics [22].

In particular, we developed a system, called SemML, that extends the conventional ML workflow with four semantic components that are depicted with grey boxes in Fig. 1. These components rely on ontologies, ontology templates, and reasoning. In particular, SemML exploits upper-level and concrete domain ontologies and the ML-ontology that captures machine learning tasks. The four semantic components of SemML are:

- **Ontology extender** that allows domain experts to describe domains in terms of an upper-level ontology by filling in templates. Data scientists then also use templates to annotate domain terms with quality-related information. Then, they use the ontologies they jointly developed as a "lingua franca" for task negotiation.
- **Domain knowledge annotator** that enables data integration by annotating, mapping raw data to the terms in domain ontologies with ontology-to-data mappings.
- **Machine learning annotator** that uses automated reasoning to infer ML-relevant information from ontology-to-data mappings and creates the mappings between ML ontologies and data for each raw data source.
- **Ontology interpreter** that facilitates uniform and explainable inspection of ML models and raw data.

Ontology extender and interpreter help us to address the communication challenge, domain knowledge annotator addresses the data integration challenge, and ML annotator addresses the generalisability challenge.

We evaluated SemML with a group of domain users. In particular, we conducted two experiments with data scientists, measurement experts, and domain experts from two welding processes: resistance spot welding (RSW) and hot-staking (HS). To this end, we developed a set of templates, domain ontologies, and welding quality monitoring tasks. The users were first asked to create their domain ontologies using the ontology extender, and then map the variable names in raw data to the datatype properties of their created ontologies. After each task, they answered questionnaires to provide information on subjective satisfaction. The time and accuracy of these tasks and the scores of the questionnaires were recorded, analysed, and evaluated with promising results.

In Sect. 2 we introduce the Bosch use case of electric resistance welding. Section 3 describes the architecture and functions of SemML. Section 4 reports our user study.

2 Use Case: Quality Monitoring in Electric Resistance Welding

We now discuss the Bosch welding quality monitoring use case, the corresponding ML workflow for predictive quality monitoring, and then enumerate challenges that we address with semantic technologies.

Bosch Welding Quality Monitoring. Bosch is one of the global manufacturing leaders in the automotive industry. Welding is heavily used in industry for numerous applications including car production. Indeed, a typical car body can contain up to 6000 welding spots [38] where pieces of metal are connected. Bosch welding solutions include welding equipment (Fig. 1 for RSW) software, service, development support, etc. These solutions are used in Bosch plants and many customers worldwide, e.g. Daimler, BMW, Volkswagen, Audi, Ford. Enabled by the abundant data and computing resources behind the IoT technologies, Bosch is developing ML methods as depicted in Fig. 1 to predict the welding quality of next spots, before the actual welding happens. This allows to take necessary measures beforehand, like automatic adjustment of welding parameters, to improve the expected welding quality and avoid potential quality failure.

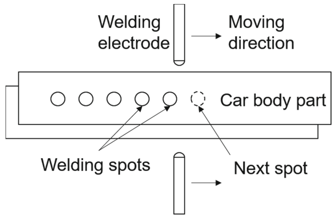

Fig. 2. Resistance spot welding

In the example illustrated in Fig. 2, the developed ML approaches should be able to predict the quality of the 6th welding spot, based on data of previous welded spots, including sensor measurements, welding configurations, past spot quality, etc. This requires acquisition of welding data from welding processes and measurements, as in *Step 1* of Fig. 1. The process, data and possible interesting tasks need to be explained to data scientists. The latter have to understand the process, the data, translate the task description from engineering languages into machine learning languages, evaluate the feasibility and cost for solving these questions. Some preliminary data analysis and visualisation are done in this step, known as Exploratory Data Analysis. Managers also have to participate in negotiating prioritising activities and goals from a view of strategic interest and make decisions of defining the task. The task negotiation is highly iterative, comprising *Step 2* of Fig. 1. This step is very time consuming and error-prone.

Then, in *Step 3* of Fig. 1 the Bosch RSW production data collected from various monitoring software in at least 4 locations and 3 original equipment manufacturers are integrated. These data may have different names for the same variables, or have some variables missing in one source but present in another, or measured with different sampling rate, etc. Besides the production data, Bosch has data collected from laboratory and simulation for process development. Extra sensors are installed in the laboratory, and the simulation data are generated with different mechanisms. Integration of all these data requires collaborative work of data scientists, data managers, process experts, and measurement experts. Thus, *Step 3* is essential but laborious and time-consuming.

After the data are integrated, the ML modelling in *Step 4* of Fig. 1 starts. It includes feature engineering which is time-consuming [1] since different datasets/domains may have different features and require different feature engineering strategies. Finally, after the heavy work of ML modelling in *Step 5* of Fig. 1, the data scientists present and visualise ML results and models, and together with other stakeholders discuss and interpret them. Managers then have to choose the best model to be deployed in *Step 6* of Fig. 1.

Use Case Requirements. Summing up, the time and effort required for ML development is heavily affected by *(C1)* the necessity of multiple iterations of communication by different stakeholders, *(C2)* the complexity of the data integration process, and *(C3)* generalisability of the developed ML models to similar processes and datasets. In order to enhance the ML workflow adopted by Bosch in a way that it addresses C1–C3 we derived the following five system requirements:

– *R1: Uniform communication model for various stakeholders*: The system should rely on a common vocabulary, with unambiguously defined relations between the terms. This vocabulary should be machine-readable and minimally controversial.
– *R2: Uniform data format and ML vocabulary*: the results of the ETL process are the input to ML modelling. Thus, the system should offer a uniform format for the data storage and a uniform naming of variables.
– *R3: Mechanism for generalising ML models*: the system should offer a mechanism for machine learning methods developed on one dataset to be reused or generalised to other datasets and manufacturing processes.
– *R4: Data enrichment mechanism*: the system should enable the enrichment of data with some task-specific information so that the integrated data can be linked to the generalisable machine learning approaches.
– *R5: Flexibility, extensibility, maintainability*: the system and its functionalities should enable accommodation of new data sources and ML tasks.

Note that the requirements R1 and R5 address the challenge C1, then R2 and R5 address C2, and R3–R5 address C3; we depict it in Fig. 1 with yellow circles.

3 SemML: Ontology-Enhanced Machine Learning Development

In this section, we present our SemML system that has a modular and multilayered architecture and illustrated in Fig. 3. In order to simplify for the reader the understanding of how SemML works, we overlay the architecture with the workflow from Fig. 1 where the steps are indicated with blue arrows. SemML has three layers: *Industry Applications Layer* where the welding monitoring, diagnostics, and analyses happen, *System Layer* that contains machine learning modules enhanced with our semantic modules, where orange circles indicate the requirements from Sect. 2 that are addressed by the corresponding modules, and *Data and Knowledge Layer* that contains ontologies, ontology templates, data, ML models and other relevant artifacts. The *Semantic Artifacts* of the latter layer serve as a bridge between the data sources and the modules in the system layer. We now discuss the layers and their interactions.

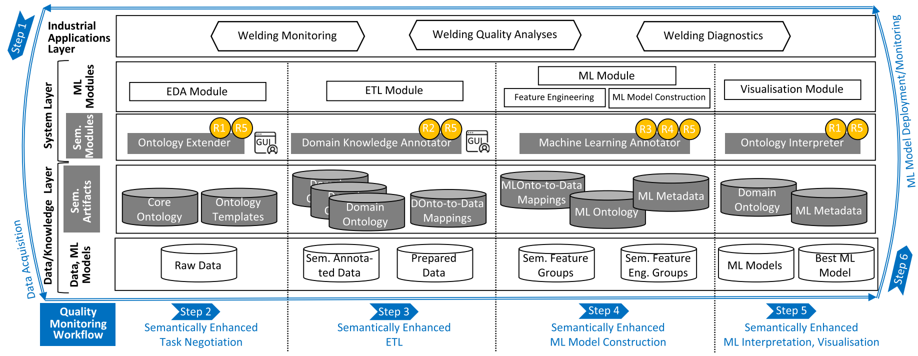

Fig. 3. An architectural overview of our semantically enhanced ML solution SemML for welding quality monitoring, where we overlay the welding quality monitoring workflow of Fig. 1 and the use-case requirements. EDA: explorotary data analysis, Sem.: semantics, Eng.: engineered. (Color figure online)

3.1 SemML Workflow

The arrows in Fig. 3 indicate the data flow from the raw sources generated by the welding monitoring and diagnostics applications through the machine learning modules back to the top layer where the developed quality models are deployed and monitored. We now walk the reader through these workflow steps.

Semantically Enhanced Task Negotiation. Once the raw data is acquired, data scientists and process experts align their backgrounds and specify the task of quality analysis. To this end, we enriched the traditional ML module for *Exploratory Data Analysis* (EDA) with the semantic *Ontology Extender*. Its graphical user interface allows experts to describe their domain in terms of an upper-level ontology, *Core Ontology* that we developed, by filling in *Ontology Templates* that we also developed. The users thus create domain ontologies that reflect the specificity of the raw data and a manufacturing process. Templates are also used by data scientists to annotate domain terms with quality-related information. Thus, the ontology extender, as well as the core ontology and templates, addresses the R1 and R5 requirements: the core ontology serves as a common communication model and the templates make the system flexible and extensible to new data sources.

Semantically Enhanced ETL. Our *Domain Knowledge Annotator* enables data integration via the mapping of the raw data to the terms in the domain ontologies. For mappings, we introduce a compact graphical user interface with browsing functionalities which is linked to the ontology extender. In case when a required term is missing, the user can switch to the ontology extender, and the newly introduced term immediately becomes available for use. The resulting domain knowledge mapping (*DOnto-to-Data Mapping*, where DOnto stands for Domain Ontology) is used by the *Extract-Transform-Load* (ETL) module to prepare the data for machine learning. Thus, the domain knowledge annotator

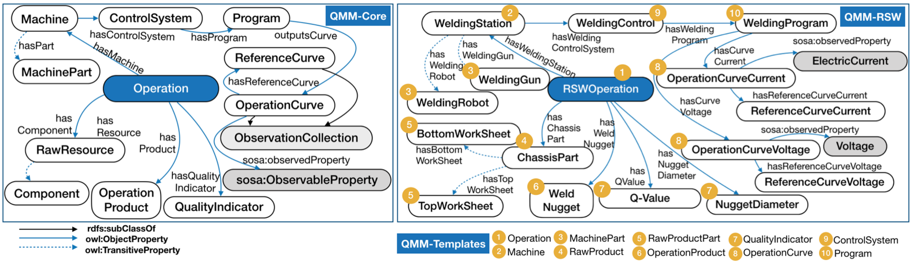

Fig. 4. `QMM-Core`, `QMM-RSW` and Templates where prefixes such as `qmm-core` are omitted.

addresses the R2 and R5 requirements: it allows to represent data in a uniform agreed format.

Semantically Enhanced ML Model Construction. The data prepared after the semantically enhanced ETL go through the *ML-Annotator* module. It relies on an ontology reasoner and the ML ontology to infer machine learning-relevant information from the DOnto-to-Data mappings. It creates the *MLOnto-to-Data Mapping* (where MLOnto stands for ML Ontology) for each raw data source. The resulting two kinds of mappings store different relationships. Indeed, consider for example a sensor measurement feature named as "CurrentAmp" that contains a series of observations of electric current values with time stamps. This feature will be mapped to the domain term "operationCurveCurrentValue" with a DOnto-to-Data mapping and to the ML term "TimeSeries" with an MLOnto-to-Data mapping. The latter indicates that this column will be treated as time-series (a special feature group) in machine learning. MLOnto-to-Data mappings enable the uniform handling of the prepared data by ML algorithms in the *Feature Engineering* module. This module performs various transformations of data categorised as feature groups and can also add new *Engineered Groups* of features. After feature engineering, several machine learning models are constructed in the *ML Model Construction* module. Information about the used feature engineering algorithms and engineered features are stored in the data layer as the *ML-Metadata*, that is, an application ontology which facilitates visualisation of the machine learning modelling. Our ML-Annotator addresses the requirements R3–R5.

Semantically Enhanced ML Interpretation and Visualisation. In order to conduct ML interpretation, data scientists discuss the ML models with other stake-holders through the *Visualisation Module*. Our *Ontology Interpreter* module facilitates a uniform and explainable inspection of ML models and raw data using ontologies, and thus, addresses the requirements R1 and R5. After the inspection, a selected ML model, and insights provided by ML analysis are deployed in the industrial applications layer.

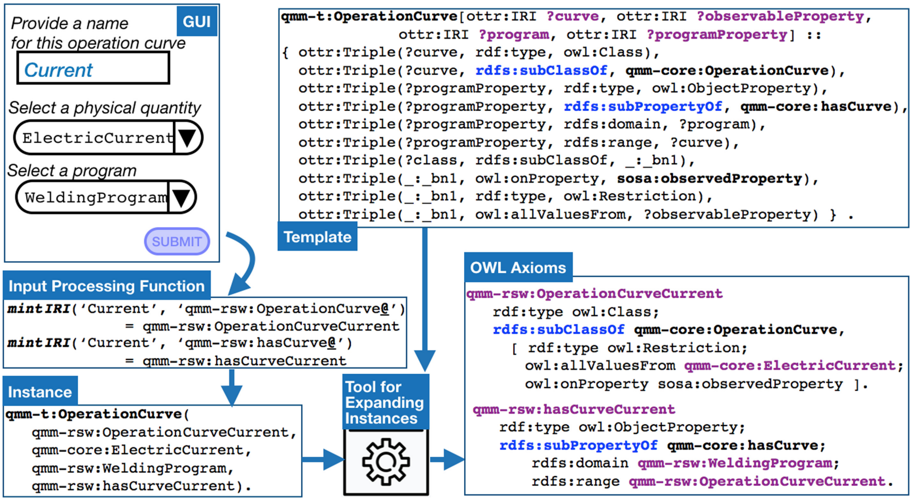

Fig. 5. Template, template instance and serialised OWL axioms.

3.2 Semantic Artifacts of SemML for Automated Welding Monitoring

We now give more details on the semantic artifacts of SemML that we developed for automated welding monitoring. In particular, we will discuss `QMM-Core`, the upper-level ontology for Quality Monitoring in Manufacturing, the library of templates, and show how they were used to construct one of the domain ontologies, `QMM-RSW`, for the manufacturing process of resistance spot welding. We then describe `QMM-ML`, the ontology for machine learning that powers the machine learning components of the system, and show how the automated reasoning and generalisability are enabled for different domains.

`QMM-Core` **ontology** is an OWL 2 ontology and can be expressed in the Description Logics $\mathcal{S}(\mathcal{D})$. With its 1170 axioms, which define 95 classes, 70 object properties and 122 datatype properties, it models the processes of discrete manufacturing with an emphasis on quality analysis. The left part of Fig. 4 displays the main classes and relations between them. This ontology has been developed through a series of workshops, taking inputs from various Bosch experts of engineering and machine learning. It can serve as one solution to reflect the consensus terminology for a common base of discussion. The ontology takes an operation-centred perspective: this orientation naturally follows from the analytical task of quality prediction described in Sect. 2. In particular, a qmm-core:Operation is performed by a qmm-core:Machine on a qmm-core:RawProduct. It results in a qmm-core:OperationProduct. Sensor observations are stored as qmm-core:OperationCurves and represent series of observation results with their corresponding timestamps. This class is our lightweight adaptation of ssn-ext:ObservationCollection from the proposed extensions to the Semantic Sensor Network Ontology [4]. We thus align

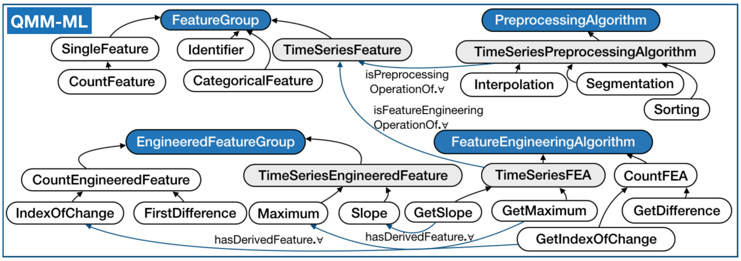

Fig. 6. A fragment of the `QMM-ML` ontology.

`QMM-Core` with the established way to model and query sensor observations – the SOSA/SSN ontology [10].

`QMM-T` **Template Library.** Our templates can be seen as parametrised ontologies and they rely on the Reasonable Ontology Templates (OTTR) framework [31]. By providing values (arguments) for each parameter and a user can create an instance of a template, which is then serialised as OWL axioms. Templates guarantee uniformity of the updates and the consistency of the updated ontology, as well as the relative simplicity of the ontology extension process. For our use cases, we created a template library `QMM-T` that relies on the classes and relationships of `QMM-Core` and has 30 templates. We also implemented a GUI that exposes the `QMM-T` to end-users. In Fig. 5 we exemplify our template instantiation process with one template qmm-t:OperationCurve.

`QMM-RSW` **Resistance Spot Welding Ontology.** By applying our templates, Bosch domain specialists created the `QMM-RSW` – ontology for the resistance spot welding process. It features 1542 axioms, which define 84 classes, 123 object properties and 246 datatype properties and can be expressed using $\mathcal{SH}(\mathcal{D})$ Description Logics. `QMM-RSW` and templates are partially shown in the right part of Fig. 4.

`QMM-ML` **Ontology** is partially depicted in Fig. 6. It has classes to categorise features as qmm-ml:FeatureGroups: time series, categorical features, identifiers, etc. It contains 62 classes, 4 object properties, 2 datatype properties as well as 210 axioms and 122 annotation assertions; it can be expressed using $\mathcal{ALH}(\mathcal{D})$ Description Logics. When `QMM-ML` is mapped to data, then MLOnto-to-Data mappings store qmm-ml:FeatureGroups for all columns in the prepared data. These mappings can be created automatically using reasoning and then modified by users. The ML module of SemML, in turn, has generic operations and algorithms with the behaviour specified on the level of qmm-ml:FeatureGroups of `QMM-ML`. Then, the ML Module retrieves the pre-processing and feature engineering algorithms for each group of features. To this end, it relies on the corresponding class definitions in the `QMM-ML`. For instance, the pre-processing algorithm for time series is defined as follows:

$$\mathsf{TimeSeriesPreprocessingAlgorithm} \sqsubseteq \forall \mathsf{isPreprocessingAlgorithmOf}.\mathsf{TimeSeries}.$$

The ML module contains the implementation for each of the algorithm's subclasses: Interpolation, Segmentation and Sorting. The feature engineering algorithm for time series is defined analogously:

$$\mathsf{TimeSeriesFeatureEngineeringAlg} \sqsubseteq \forall \mathsf{isFeatureEngineeringAlgOf.TimeSeries}.$$

Its subclasses, in turn, are related to the corresponding engineered features, and the ML module will have the implementation for all of them. For example, based on the definition: $\mathsf{GetMaximum} \sqsubseteq \forall\ \mathsf{hasDerivedFeature.Maximum}$, the Feature Engineering module of SemML will apply the implemented GetMaximum algorithm to all time series features and generate new features with the token "Maximum" in their name for all of them.

In ML terms, the way how our semantically enhanced ML module works is: $h : \mathcal{X} \xrightarrow{M} \{\{FG_1\} \cdots \{FG_N\}\} \xrightarrow{\texttt{QMM-ML}} \{\{FEG_1\} \cdots \{FEG_K\}\} \rightarrow \hat{\mathcal{QI}}$, where h is a hypothesis that maps raw input features $\mathcal{X}$ into an estimation $\hat{\mathcal{QI}}$ of a welding quality indicator $\mathcal{QI}$. This mapping has two intermediate steps: *(1)* using MLOnto-to-Data Mapping M it fetches a set of standardised Feature Groups FGs and *(2)* using `QMM-ML` it turns them into a set of Feature Engineered Groups FEGs. This makes the developed ML approaches easily extendable to similar tasks and datasets. Moreover, this enables non-ML-experts to better understand the ML approaches, and even to modify the ML approaches with minimal training of ML expertise. Note that the classical ML module starts with $\mathcal{X}$ and may develop different ad hoc feature processing strategies for different tasks and data sources to estimate $\hat{\mathcal{QI}}$, or schematically: $h : \mathcal{X} \rightarrow \hat{\mathcal{QI}}$.

3.3 Related Work

Survey [30] extensively covers the usage of semantic technologies in data mining and knowledge discovery, and in particular in the facilitation of machine learning workflows. Still, to the best of our knowledge, existent approaches and system solutions, including the recent developments of digital twins for manufacturing [14], only partially meet our requirements R1–R5. Thus we had to develop our own ontologies and templates as well as ontology-based, highly customised and configurable solution, integrated into the workflow to support quality analysis in manufacturing. The users of our system are the different experts responsible for the task of developing machine learning methods. Indeed, none of the ontologies for manufacturing (e.g., [2,8,23,24,33,34]) fully serve as the communication model for our use cases and sufficiently cover our domains. The mapping-based data integration solutions like Ontop [17] are not particularly targeted towards our aim of minimising the involvement of ontologists into the model maintenance processes. Moreover, the role of mappings in our context is not limited to the transformation of data into the RDF format. Firstly, we integrate data sources for machine learning, secondly and in line with these tools, we transform some parts of it to RDF to explore the data. In the metadata management solutions for data lakes like Constance [9] and GEMMS [28], the metadata descriptions are used to integrate the raw sources. As the mapping-based data integration solutions, these systems lack the extensibility aspect. We found that the existing tools for ontology extension, e.g. template-driven systems (Webulous [15], TermGenie [5], Ontorat [36]) required considerable adjustments (including but not limited to the development of the new graphical user interface) and could not be easily integrated with the machine learning workflow and our infrastructure. Thus, we developed our own ontology extension tooling.

4 User Study

Our user study evaluates how well SemML addresses the challenges C1 on communication by evaluating the Ontology Extender and C2 on data integration by evaluating Domain Knowledge Annotator. Evaluation for C3 is our future work. To this end, we organised a workshop with three parts. First, we organised a thirty-minutes crash course to explain the ontology `QMM-Core` and templates. Then, we conducted two experiments: Experiment 1 on *Ontology Extension*, where the users were asked to describe their domains in terms of `QMM-Core` by filling in the proposed templates, and Experiment 2 on *Data Mapping*, where the users were asked to map the variables in the raw data sources to the datatype properties in the ontologies they created. Note that our experiments do not aim at comprehensive coverage of the welding domains and data sources relevant for welding quality: in our evaluation tasks we tried to balance the coverage and the time required to accomplish them.

4.1 Design of Experiments

We give further details on experiments and participants.

Users. Two target user groups (with the roles of domain experts and data scientists) participated in the experiments with two welding processes: resistance spot welding (RSW) and hot-staking (HS). The users could choose to participate only in Experiment 1 or in both. Some of them took part in the experiments with more than one domain or role. This is the case, e.g., for users who are domain experts both for RSW and HS, and some users who are domain experts but are learning data analysis or vice versa. In total, from 14 participants 25 result instances were collected in Experiment 1, and 19 instances in Experiment 2. Before the experiments, the participants rated their domain expertise (E1), experience with semantic technologies (E2) and experience with data mapping tools (E3) on a Likert scale (1: Beginner, 2: Developing, 3: Competent, 4: Advanced, 5: Expert).

Experiment 1: Ontology Extension. The users were asked to use Ontology Extender to create their ontologies. As illustrated in Fig. 7.1.1, for each term highlighted with the blue background in the short descriptions for the welding processes on the left side, the users selected a template on the right side, and then made choices to link the created class to its dependencies (drop-down list in Fig. 7.1.2). The resulting ontology terms (classes and properties) were then visualised (Fig. 7.1.3). Note that domain experts and data scientists did their

Fig. 7. Graphical user interfaces for (1.1–1.3) ontology extension and (2) Data mapping. (Color figure online)

tasks sequentially: the former created an ontology, and then the latter inspected their ontologies and extended them with quality indicators.

Experiment 2: Data Mapping. As illustrated in Fig. 7.2, the users were asked to use Domain Knowledge Annotator to map data. For each term in the column of raw variable names on the left side, they clicked the group of classes from the right top panel, selected a class, and then chose the datatype properties where the class is a domain from a drop-down list (in the right bottom panel).

4.2 Evaluation Metrics

According to ISO 9241-11 system usability has 3 dimensions: *effectiveness*, *efficiency*, and *satisfaction* [12]. We rely on them and their correlations with user expertise.

Effectiveness shows to which extent the intended goal is achieved [12]. We use *correctness*, the percentage of successfully completed tasks, as the metric for it. We are fully aware that there is no absolute correctness for these tasks because the domains or data can be understood in different ways. This issue is however not critical in our experiments since we carefully designed the tasks so that the answers are minimally controversial across the experts. In Experiment 1, the correctness is defined as the percentage of correctly chosen templates for a given term (Template Correctness, TC), the percentage of correct choices linking the dependencies between classes (Choice Correctness, ChC), and the percentage of fully correctly created classes, for which the correct template is chosen and all dependencies are correctly specified (Final Correctness, FC). In Experiment 2, the correctness is defined as the percentage of correctly chosen classes (Class Correctness, ClC) and the percentage of correctly mapped datatype properties for each item of raw variable names (Item Correctness, IC).

Table 1. Satisfaction metrics: Questionnaires and aggregated quality dimensions.

<table>
<tr><th colspan="2">Experiment 1: Ontology Extension</th><th colspan="2">Experiment 2: Data Mapping</th><th rowspan="2">Dimension</th></tr>
<tr><th>Domain Experts</th><th>Data Scientists</th><th>Domain Experts</th><th>Data Scientists</th></tr>
<tr><td colspan="4">Q1: I felt very confident using the system</td><td rowspan="2">D1: User Friendliness</td></tr>
<tr><td colspan="4">Q2: I found the system unnecessarily complex</td></tr>
<tr><td colspan="4">Q3: I needed to learn many things before I became productive with this system</td><td rowspan="2">D2: Self-Explainability</td></tr>
<tr><td colspan="4">Q4: I needed support of a technical person to be able to use this system</td></tr>
<tr><td colspan="4">Q5: I thought there was too much inconsistency in this system</td><td>D3: Consistency</td></tr>
<tr><td colspan="4">Q6: I think the system covers most of my fundamental requirements for the *</td><td rowspan="2">D4: Completeness</td></tr>
<tr><td>* description of the process</td><td>* understanding of the process</td><td>* description of the data</td><td>* understanding of the data</td></tr>
<tr><td colspan="4">Q7: I think the system/resulting model allows me to unambiguously *</td><td rowspan="4">D5: Descriptive Power</td></tr>
<tr><td>* describe the process</td><td>* understand the process</td><td>* describe the data</td><td>* understand the data</td></tr>
<tr><td colspan="2">Q8: I think the relevant aspects of process quality are well presented in the *</td><td colspan="2">I think the mapping system saves me effort of *</td></tr>
<tr><td>* system</td><td>* system and the resulting model</td><td>* describing the data</td><td>* understanding and completing the data mapping</td></tr>
<tr><td colspan="4">Q9: I think the resulting ontology/mapping would be very easy to understand for *</td><td rowspan="4">D6: Communication Easiness</td></tr>
<tr><td>* data scientists</td><td>* domain experts</td><td>* data scientists</td><td>* domain experts</td></tr>
<tr><td colspan="4">Q10: I think the resulting ontology/mapping provides a good common base for discussion with *</td></tr>
<tr><td>* data scientists</td><td>* domain experts</td><td>* data scientists</td><td>* domain experts</td></tr>
</table>

Efficiency corresponds to "the resources (such as time or effort) needed by users to achieve their goals" [12]. We use *time spent on tasks* as the metric of efficiency.

Satisfaction was evaluated with the questionnaires after each experiment on 6 dimensions (see Table 1): [D1] *User Friendliness*: the system is easy to use; [D2] *Self-Explainability*: the system does not require extra knowledge or support; [D3] *Consistency*: the system is consistent in format, workflow, wording, etc.; [D4] *Completeness*: the system covers the domain/data to describe/understand; [D5] *Descriptive Power*: the system allows to describe the domain/data effectively, clearly; [D6] *Communication Easiness*: the system eases the communication between experts. The questions are presented in Table 1 where the first five are inspired by System Usability Scale (SUS). Questions 1–5 (and the corresponding Dimensions 1–3) target the usability of the graphical user interface. They are identical for all roles and experiments. Questions 6–10 (Dimensions 4–6) address more specific issues and differ slightly with respect to the role or the experiment. The users were asked to give scores ranging from 1 to 5 with a Likert scale (1: Strongly disagree, 2: Disagree, 3: Neither agree or disagree, 4: Agree, 5: Strongly agree). Note that Questions 2–5 are negatively formulated. Their scores are reversed in the later analysis to make the representation of the results more intuitive and consistent. E.g. if a user scores Q2 with 1, which means the user strongly disagrees that the system is complex, the corresponding score is reversed to 5, indicating the system is not complex.

Fig. 8. 1 and 2: User performance of time and correctness for RSW in 1 and for HS in 2, aggregated on template groups for both in 3. Time and correctness for data mapping in 4.

4.3 Evaluation Results and Discussion

The results of the Effectiveness and Efficiency metrics are summarised in Fig. 8, which shows the user performance on Ontology Extender (Fig. 8.1–8.3) and Domain Knowledge Annotator (Fig. 8.4).

Results for Experiment 1: Ontology Extension. Domain experts in RSW created 14 terms and those in HS created 15 terms. Data scientists created 2 terms for both processes. On average, the users needed about 50s to create a new term. Note that the description of one term adds from 4 to 25 classes and properties to the ontology (see the process exemplified in Fig. 5 of Sect. 3.2).

Some users needed extra time for the terms WeldingMachine, CapWearCount and CapDressCount (with high standard deviation shown in the figures). The potential reasons are that the users needed to understand the complex structure of

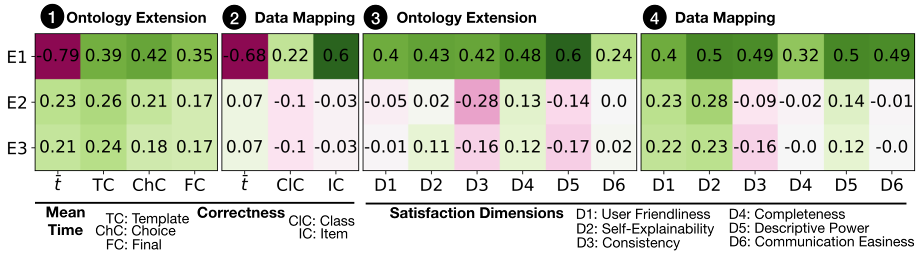

Fig. 9. Heatmaps of correlation coefficients between the usability metrics and self-accessed expertise. E1: Domain expertise, E2: Experience with semantic technologies, E3: Experience with data mapping tools.

machine and its multiple parts; for the latter two terms (both are created by the SystemComponentStatus template in Fig. 8.3) the users specified two dependencies, which is one more than the normal case of one choice. Another reason could be that the users moved to a new template group, which increased the cognitive complexity of the task and thus, the time spent on the task. In line with this tendency, we observe a gradual decline in time for subsequent terms created with the same or similar templates. For example, WeldingRobot and WeldingGun are machine parts directly following the WeldingMachine, and CapDressCount directly follows CapWearCount. This strongly supports the learnability of the system: having experience with a template increases efficiency and effectiveness.

The average correctness for applying a template is 93%, for making choices of the dependencies is 92%, and for both (final correctness) is 90%. The terms, e.g. CapWearCount, that required more time to create often have a relatively low correctness ratio. The high average correctness strongly demonstrates the usability and the error prevention potential of the system.

One of the goals of Ontology Extender was to serve as the communication platform between domain experts and data scientists. In our experimental setup, the data scientists were supposed to (1) inspect the domain ontologies created by the domain experts and (2) add the terms relevant for quality analysis. In particular, they had to add or find a term, and characterise it as a quality indicator. We separate these two parts by the vertical dashed lines in Fig. 8.1–8.3. All data scientists achieved 100% correctness with an average time of 39 s.

We now analyse the correlations between the self-reported expertise of our users and their performance. Figure 9.1 shows a strong negative correlation between domain expertise (E1) and time t and relatively strong positive correlation between E1 and the three types of correctness (template, choice and final). Not surprisingly, the users with higher domain expertise provided more correct modeling solutions and were faster than the beginners. The figure also suggests insignificant correlation between the performance of users and their experience in semantic technologies and mapping tools. This is encouraging since it suggests that the usage of our system requires no or little prior training in these disciplines and activities.

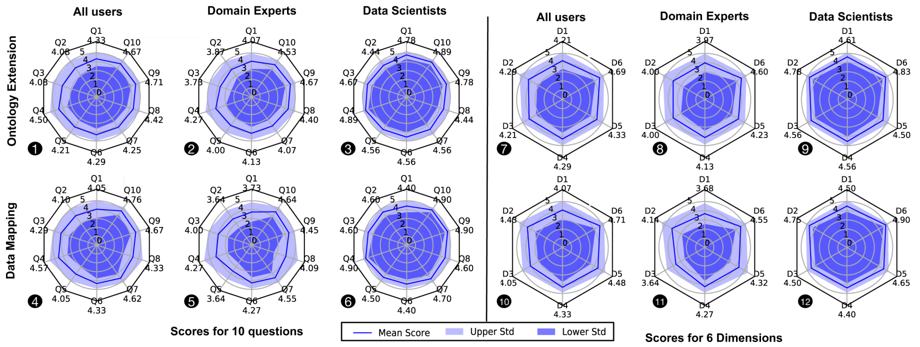

Fig. 10. Radar charts of questionnaires scores on 10 questions (1–6) and aggregated to 6 dimensions (7–13) defined in Table 1. Std is standard deviation. The blue lines indicate the mean scores, the light blue shadow the `mean + std`, and the dark blue shadow the `mean - std`. (Color figure online)

Results for Experiment 2: Data Mapping. The majority of users correctly mapped column names in the suggested files to the newly introduced terms, achieving 100% correctness (Fig. 8.4). The average time they spent for each term is about 50s. The correlations in Fig. 9.2 support the idea that domain expertise will ease the work and the other two parameters, including the self-accessed experience with mapping tools, have almost no effect. We interpret it as the evidence that the system is able to serve as a solution for both tasks – data modeling and data mapping – and does not require any prior experience with similar technologies. We complement our analysis with the results of the satisfaction questionnaires in the following section.

Satisfaction. We report the satisfaction results in the radar charts in Fig. 10 separately for data scientists and domain experts, and aggregated for all users. The charts in Fig. 10.1 and 10.4 represent the average scores for both user groups. These scores are higher than 4, which indicates a general good impression of users. The mean scores on Questions 4, 9 and 10 are very high (>4.5): The users evaluate the tool as easy to use without support of a technical person, and they think their working results will be easy to understand for other experts. This supports our vision that an ontology can serve as a good communication base.

The comparison of the scores on questions by the domain experts and data scientists (Fig. 10.2 vs. 10.3, 10.5 vs. 10.6), reveals that the data scientists evaluate the system with higher scores in average, and smaller standard deviation. This indicates that the data scientists have better and more uniform opinions on the system, while users taking the role of domain experts have more diverse opinions. One reason for that could be that the tasks for data scientists were only related to quality indicators and thus more clearly defined, while the tasks for domain experts who needed to describe complicated processes were more demanding.

In Fig. 10.7–10.12 the scores on questions are aggregated to six dimensions (see the meanings of dimensions in Table 1). This aggregation makes it easier to draw conclusions. Firstly, all six dimensions have average scores over 4, which means the users are satisfied with the system in general. The scores for D6 (communication easiness) and D5 (descriptive power) are the highest, indicating the users appreciate the ease of communication. Dimensions more related to the system usability (D1–D3) have scores around four, which means there is improvement space for the user interface.

Correlations in Fig. 9.3 and 9.4 reveal similar results as for the performance analysis: domain expertise correlates with high satisfaction scores, while the other two areas of expertise have little effect, which supports that the tool requires little prior training.

5 Conclusion, Lessons Learned, And Outlook

Conclusion. In this work we presented a Bosch use case of automated welding, challenges with ML-based welding quality monitoring, and requirements for a system to address them. To address the challenges we proposed to enhance the welding quality monitoring ML development with four semantic components: Ontology Extender, Domain Knowledge Annotator, Machine Learning Annotator, and Ontology Interpreter. We implemented the enhancement as the SemML system. We then evaluated SemML by focusing on the first two semantic modules. To this end, we conducted a user study with 14 Bosch experts. The evaluations show promising results: SemML can indeed help in addressing the challenges of communication and data integration.

Lessons Learned. First, an ontology as a formal language can be very effective to provide a lingua franca for communication between experts with different knowledge backgrounds. The process of developing the Core model was onerous, time-consuming and cognitively demanding at the initial phase. After that, we had a basis of core model and a set of templates. It revealed the development process became much easier because the developed ontologies facilitated communication. Second, the technology of templates enables non-ontologists to describe their domains and data in a machine-readable and unambiguous way. In contrast to the simple, tabular interfaces which exist for the template-based ontology construction, we needed to address the new requirements for our use case. In the use case, the users need to generalise a series of classes, while the later generated classes have dependencies on the older ones. This has two requirements: (1) the newly generated classes need to be accessible to later generated classes; (2) sequences of templates need to be applied in a particular order because the later classes presuppose the existence of their depending classes. Furthermore, the users need assistance like drop-down lists and visualisation of changes. Third, the users that are unfamiliar with the domains will need more time for some tasks and yield lower correctness. This indicates that we need to split the iterative process of task negotiation to smaller units so that different experts can digest each other's information more smoothly.

Outlook. We have an on-going study [39] that informally evaluates SemML's third semantic module and provides a user-interface [40] and we plan to further extend it to a larger scale user evaluation. Then, SemML is currently deployed in a Bosch evaluation environment, and we plan to push it into the production. This in particular requires to show the benefits of SemML with more Bosch users and in other use cases. This also requires to further improve the usability of SemML with more advanced services such as access control as well as with various ontology visualisation modules.

References

1. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**, 1798–1828 (2013)
2. Borgo, S., Leitão, P.: The role of foundational ontologies in manufacturing domain applications. In: Meersman, R., Tari, Z. (eds.) OTM 2004. LNCS, vol. 3290, pp. 670–688. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30468-5_43
3. Chand, S., Davis, J.: What is smart manufacturing. Time Mag. Wrapper **7**, 28–33 (2010)
4. Cox, S.: Extensions to the semantic sensor network ontology. W3C Working Draft (2018)
5. Dietze, H., et al.: TermGenie-a web-application for pattern-based ontology class generation. J. Biomed. Semant. **5** (2014). https://doi.org/10.1186/2041-1480-5-48
6. DIN EN 14610: Welding and allied processes - definition of metal welding processes. German Institute for Standardisation (2005)
7. Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery in databases. AI Mag. **17**(3), 37 (1996)
8. Fiorentini, X., et al.: An ontology for assembly representation. Technical report. NIST (2007)
9. Hai, R., Geisler, S., Quix, C.: Constance: an intelligent data lake system. In: SIGMOID 2016 (2016)
10. Haller, A., et al.: The SOSA/SSN ontology: a joint WEC and OGC standard specifying the semantics of sensors observations actuation and sampling. In: Semantic Web (2018)
11. Horrocks, I., Giese, M., Kharlamov, E., Waaler, A.: Using semantic technology to tame the data variety challenge. IEEE Internet Comput. **20**(6), 62–66 (2016)
12. ISO: 9241–11.3. Part II: guidance on specifying and measuring usability. ISO 9241 ergonomic requirements for office work with visual display terminals (VDTs) (1993)
13. ITU: Recommendation ITU - T Y.2060: Overview of the Internet of Things. Technical report. International Telecommunication Union (2012)
14. Jaensch, F., Csiszar, A., Scheifele, C., Verl, A.: Digital twins of manufacturing systems as a base for machine learning. In: 2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), pp. 1–6. IEEE (2018)
15. Jupp, S., Burdett, T., Welter, D., Sarntivijai, S., Parkinson, H., Malone, J.: Webulous and the Webulous Google Add-On-a web service and application for ontology building from templates. J. Biomed. Semant. **7**, 1–8 (2016)
16. Kagermann, H.: Change through digitization—value creation in the age of industry 4.0. In: Albach, H., Meffert, H., Pinkwart, A., Reichwald, R. (eds.) Management of Permanent Change, pp. 23–45. Springer, Wiesbaden (2015). https://doi.org/10.1007/978-3-658-05014-6_2

17. Kalaycı, E.G., González, I.G., Lösch, F., Xiao, G.: Semantic integration of Bosch manufacturing data using virtual knowledge graphs. In: Pan, J.Z., et al. (eds.) ISWC 2020. LNCS, vol. 12507, pp. 464–481. Springer, Cham (2020) (2020)
18. Kharlamov, E., et al.: Capturing industrial information models with ontologies and constraints. In: Groth, P., et al. (eds.) ISWC 2016. LNCS, vol. 9982, pp. 325–343. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46547-0_30
19. Kharlamov, E., et al.: Ontology based data access in Statoil. J. Web Semant. **44**, 3–36 (2017)
20. Kharlamov, E., et al.: Semantic access to streaming and static data at Siemens. J. Web Semant. **44**, 54–74 (2017)
21. Kharlamov, E., et al.: An ontology-mediated analytics-aware approach to support monitoring and diagnostics of static and streaming data. J. Web Semant. **56**, 30–55 (2019)
22. Kharlamov, E., Mehdi, G., Savković, O., Xiao, G., Kalaycı, E.G., Roshchin, M.: Semantically-enhanced rule-based diagnostics for industrial Internet of Things: the SDRL language and case study for Siemens trains and turbines. J. Web Semant. **56**, 11–29 (2019)
23. Krima, S., Barbau, R., Fiorentini, X., Sudarsan, R., Sriram, R.D.: OntoSTEP: OWL-DL ontology for STEP. Technical report. NIST (2009)
24. Lemaignan, S., Siadat, A., Dantan, J.Y., Semenenko, A.: MASON: a proposal for an ontology of manufacturing domain. In: IEEE DIS (2006)
25. Mikhaylov, D., Zhou, B., Kiedrowski, T., Mikut, R., Lasagni, A.F.: High accuracy beam splitting using SLM combined with ML algorithms. Opt. Lasers Eng. **121**, 227–235 (2019)
26. Mikhaylov, D., Zhou, B., Kiedrowski, T., Mikut, R., Lasagni, A.F.: Machine learning aided phase retrieval algorithm for beam splitting with an LCoS-SLM. In: Laser Resonators, Microresonators, and Beam Control XXI, vol. 10904, p. 109041M (2019)
27. Mikut, R., Reischl, M., Burmeister, O., Loose, T.: Data mining in medical time series. Biomed. Tech. **51**, 288–293 (2006)
28. Quix, C., Hai, R., Vatov, I.: GEMMS: a generic and extensible metadata management system for data lakes. In: CAiSE Forum (2016)
29. Ringsquandl, M., et al.: Event-enhanced learning for KG completion. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 541–559. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_35
30. Ristoski, P., Paulheim, H.: Semantic web in data mining and knowledge discovery: a comprehensive survey. J. Web Semant. **36**, 1–22 (2016)
31. Skjæveland, M.G., Lupp, D.P., Karlsen, L.H., Forssell, H.: Practical ontology pattern instantiation, discovery, and maintenance with reasonable ontology templates. In: Vrandečić, D., et al. (eds.) ISWC 2018. LNCS, vol. 11136, pp. 477–494. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00671-6_28
32. Soylu, A., et al.: OptiqueVQS: a visual query system over ontologies for industry. Semant. Web **9**(5), 627–660 (2018)
33. Usman, Z., Young, R.I.M., Chungoora, N., Palmer, C., Case, K., Harding, J.: A manufacturing core concepts ontology for product lifecycle interoperability. In: van Sinderen, M., Johnson, P. (eds.) IWEI 2011. LNBIP, vol. 76, pp. 5–18. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19680-5_3
34. Šormaz, D., Sarkar, A.: SIMPM - upper-level ontology for manufacturing process plan network generation. Robot. Comput. Integr. Manuf. **55**, 183–198 (2019)
35. Wuest, T., Weimer, D., Irgens, C., Thoben, K.D.: Machine learning in manufacturing: advantages, challenges, and applications. Prod. Manuf. Res. **4**, 23–45 (2016)

36. Xiang, Z., Zheng, J., Lin, Y., He, Y.: Ontorat: automatic generation of new ontology terms, annotations, and axioms based on ontology design patterns. J. Biomed. Semant. **6** (2015). https://doi.org/10.1186/2041-1480-6-4
37. Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., Gao, R.X.: DL and its applications to machine health monitoring. MS&SP **115**, 213–237 (2019)
38. Zhou, B., Pychynski, T., Reischl, M., Mikut, R.: Comparison of machine learning approaches for time-series-based quality monitoring of resistance spot welding (RSW). Arch. Data Sci. Ser. A **5**(1), 13 (2018). (Online first)
39. Zhou, B., Svetashova, Y., Byeon, S., Pychynski, T., Mikut, R., Kharlamov, E.: Predicting quality of automated welding with machine learning and semantics: a Bosch case study. In: CIKM (2020)
40. Zhou, B., Svetashova, Y., Pychynski, T., Kharlamov, E.: SemFE: facilitating ML pipeline development with semantics. In: CIKM (2020)
41. Zhou, B., Chioua, M., Bauer, M., Schlake, J.C., Thornhill, N.F.: Improving root cause analysis by detecting and removing transient changes in oscillatory time series with application to a 1, 3-butadiene process. Ind. Eng. Chem. Res. **58**, 11234–11250 (2019)