SemML: Reusable ML for Condition Monitoring in Discrete Manufacturing

Yulia Svetashova[1,2], Baifan Zhou[1,2], Stefan Schmid[1], Tim Pychinsky[1], and Evgeny Kharlamov[3,4]

[1] Bosch Corporate Research, Robert Bosch GmbH, Germany
[2] Karlsruhe Institute of Technology, Germany
[3] Bosch Center for Artificial Intelligence, Robert Bosch GmbH, Germany
[4] University of Oslo, Norway

Abstract. Machine learning (ML) is gaining much attention for data analysis in manufacturing. Despite the success, there is still a number of challenges in widening the scope of ML adoption. The main challenges include the exhausting effort of data integration and lacking of generalisability of developed ML pipelines to diverse data variants, sources, and domain processes. In this demo we present our SemML system that addresses these challenges by enhancing machine learning with semantic technologies: by capturing domain and ML knowledge in ontologies and ontology templates and automating various ML steps using reasoning. During the demo the attendees will experience three cunningly-designed scenarios based on real industrial applications of manufacturing condition monitoring at Bosch, and witness the power of ontologies and templates in enabling reusable ML pipelines.

1 Introduction

Industry 4.0 [4] and the Internet of Things [3] behind it lead to unprecedented growth of data generated from manufacturing processes [1]. 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.

This opens new horizons for data-driven methods like Machine Learning (ML) in condition monitoring for a wide range of application scenarios, which is to assess or predict some performance indicators for machine health state, e.g. turbine life-span, or product quality, e.g. welding spot diameter. The broad practice of development and deployment of intelligent information processing technologies in discrete manufacturing is a highlighted feature in the grand trend of Industry 4.0 [12–14].

One-time development of an ML pipeline for a specific scenario can be done within a reasonably short time. However, condition monitoring requires the constant development of new ML models. On the one hand, this requires the ML pipeline to deal with a variety of data; on the other hand, the ML models have to be developed for similar processes or similar tasks. Therefore, an important challenge in the manufacturing industry is to scale ML model development and to enable reusability of already-developed ML pipelines. Direct reuse of an ML pipeline without any modification is unrealistic.

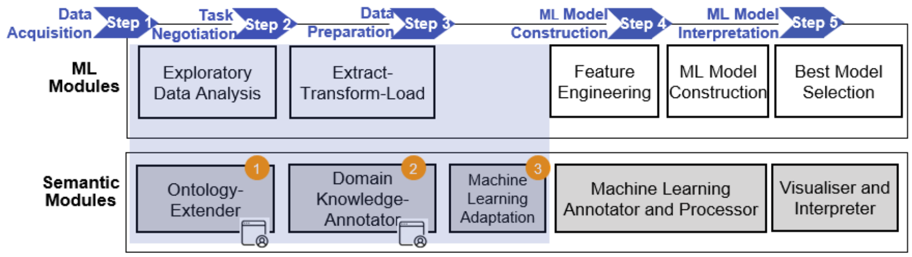

Fig. 1. An Architectural Overview of SemML

Thus, the developed ML pipelines require adaptation that should ideally be done with affordable or minimal modification.

In this work we address this challenge by relying on semantic technologies. Semantic technologies have recently gained a considerable attention in industry for a wide range of applications and automation tasks such as modelling of industrial assets [5] and industrial analytical tasks [7], integration [6, 8, 2] and querying [11] of production data, and for process monitoring [10] and equipment diagnostics [9].

Our solution uses ontologies to integrate data, to infer ML-relevant information for data of various types (time series, categorical, numerical), to perform automated feature engineering, ML model construction and ML pipeline reuse. The core component of the solution is the template-based extensibility mechanism: we employ templates to extend ontologies to new data sources and to create ontologies for new manufacturing processes. Graphical user interfaces for ontology extension and data annotation make these tasks accessible to non-ontologists. Annotated data serves as input to the configurable ontology-aware ML pipeline.

We implemented our ideas in the system called SemML. In this demonstration we will show how SemML facilitates reusability of the developed ML pipelines for quality analysis using three scenarios. First, we present how SemML allows to reuse ML models to new production lines and new data sources, e.g., laboratory data and simulations. Second, we show how SemML helps in adapting ML models to new condition monitoring tasks, e.g., estimation vs prediction, varying quality indicators. Finally, we exhibit how SemML can help in adjusting ML pipelines to new manufacturing processes.

This demo paper accompanies our accepted ISWC'20 in-use paper [12].

2 Our System SemML and Data Requirements

A typical workflow that supports the development of ML models for manufacturing consists of 5 steps shown in Figure 1: (i) data acquisition, (ii) task negotiation, (iii) data preparation, (iv) ML model construction, (v) ML model interpretation. SemML follows this workflow and provides semantic support for unambiguous process description, task negotiation, convenient data integration and configurable ML pipeline training, testing and interpretation. Thus SemML primarily focuses on Steps 2-5 of the workflow in enabling adaptability of ML pipelines. More precisely, SemML enhances traditional ML modules with the following semantic components:

- *Ontology Extender* allows experts to extend and create domain ontologies that capture manufacturing processes and ML practices by using terms from an upper-level ontology, Core Ontology, for condition monitoring in manufacturing, and by filling in Ontology Templates. The GUI of *Ontology Extender*, which can be seen in Sub-figures 1.1 and 1.2 of Figure 2, exposes templates as UI forms, and its backend transforms user input into OWL 2 ontologies.
- *Domain Knowledge Annotator* enables users to do data integration by annotating raw data with terms from domain ontologies and stores these annotations as ontology-to-data mappings. Sub-figure 2 of Figure 2 shows its browsing functionalities. Existing and newly created ontology terms become available for the annotation of data. Annotations together with the dataset form the input to the ML-related parts of SemML.
- *ML-Pipeline Adaptation Module* 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.
- *Machine Learning Annotator and Processor* enables the uniform handing 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.
- *Machine Learning Visualizer and Interpreter* uses information about the feature engineering algorithms and engineered features to facilitate the visualisation of the machine learning modelling and select the best model.

SemML is suitable for the development and reuse of ML pipelines for various datasets representing discrete manufacturing processes [**?**]. Thus, SemML requires that some specific feature types to be present in the data: (1) *performance indicators*, i.e., *machine health state* or *quality indicator*, the estimation of which is one major task of condition monitoring. (2) unique *identifiers* for each manufacturing operation (3) *single numeric features*, such as the geometrical properties of the products or equipment, (4) *single categorical features*, e.g. control mode A, B, C, (5) *time series*, continuous sensor measurements (e.g. force) with time stamps, (6) *count features*, e.g. counts of manufactured products since the last maintenance, (7) other data types like images, videos, log-files, etc. Among which, (1) is mandatory to be present, at least one of (3)-(7) needs to be present, (2) is needed to find the correspondence between different feature types.

3 Demonstration Scenarios

For the demonstration purposes, we prepared the instance of our system and two anonymised Bosch datasets from the manufacturing production lines, which represent two welding processes: resistance spot welding (RSW) and hot-staking (HS). We will start the demo with the demonstration of the Core Ontology for discrete manufacturing and the ontology templates library via the graphical user interface of the *Ontology Extender*. We then present the datasets and shortly introduce two manufacturing processes. For the resistance spot welding dataset, we will create an RSW domain ontology, then map the column names in the dataset to its terms, and execute the developed ML pipelines,

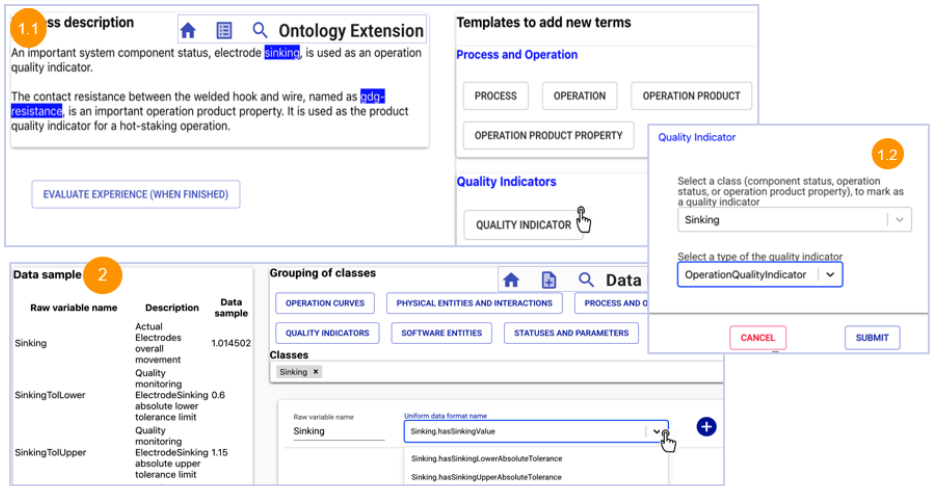

Fig. 2. Graphical User Interfaces for (1) Ontology Extension and (2) Data Annotation

which take the data and this mapping as input, and output trained ML models and predictions.

Scenario 1: Pipeline adaptation to a new production line. The typical scenario for the reuse of the developed ML pipeline is its adaptation to the new production line. Data preparation for the ML component of SemML relies on the description of the schema for each new dataset in terms of the domain ontology. The attendees will create a mapping for the RSW dataset by using the *Domain Knowledge Annotator* GUI with the RSW ontology. The mapping of the suggested dataset will require to extend the domain ontology. We formulated the typical extension requests (e.g. add a new configuration of the assembly) as tasks. For example, the initial version of the ontology will contain terms to describe a chassis part with two worksheets, the attendees will add a three-component assembly. Another adaptation use case is to extend the ML pipeline developed for the robust control system to the adaptive one (i.e. add corresponding reference parameters to the measured actual values). Thirdly, the pipeline reuse will be demonstrated for the new data sources: the simulation and laboratory datasets. In all mentioned cases, adaptation is reduced to adding new classes and properties to the RSW ontology and using them in the mappings for the new datasets.

Scenario 2: Pipeline adaptation to a new monitoring task. This scenario demonstrates the interplay between the domain knowledge acquisition and the task negotiation processes, often involving stakeholders with different backgrounds. We suggest that attendees introduce new quality indicators and show the adaptability of the pipeline on the level of the ML task. Reliable quality indicators are highly dependent on the available data. For example, simulation data in the welding domain contains such quality indicator as welding nugget diameter. This indicator is rarely present in the production data because it would mean the destruction of the welded part. The attendees will observe how the monitoring pipeline for a particular dataset handles various ML tasks.

Scenario 3: Pipeline adaptation to a new manufacturing process. In this scenario, the attendees will go through the complete cycle of data preparation for a new manufacturing process: hot-staking, and adapt the ML pipeline to an HS dataset. This will include the creation of a new domain ontology from scratch based on the compact description of the process, the mapping of data, the specification of quality indicators, and the execution of a pipeline.

References

1. Chand, S., Davis, J.: What is Smart Manufacturing. Time Magazine Wrapper **7**, 28–33 (2010)
2. 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)
3. ITU: Recommendation ITU – T Y.2060: Overview of the Internet of Things. Tech. rep., International Telecommunication Union (2012)
4. Kagermann, H.: Change Through Digitization – Value Creation in the Age of Industry 4.0. In: Management of Permanent Change (2015)
5. Kharlamov, E., Grau, B.C., Jiménez-Ruiz, E., Lamparter, S., Mehdi, G., Ringsquandl, M., Nenov, Y., Grimm, S., Roshchin, M., Horrocks, I.: Capturing Industrial Information Models With Ontologies and Constraints. In: ISWC (2016)
6. Kharlamov, E., Hovland, D., Skjæveland, M.G., Bilidas, D., Jiménez-Ruiz, E., Xiao, G., Soylu, A., Lanti, D., Rezk, M., Zheleznyakov, D., Giese, M., Lie, H., Ioannidis, Y.E., Kotidis, Y., Koubarakis, M., Waaler, A.: Ontology Based Data Access in Statoil. J. Web Semant. **44**, 3–36 (2017)
7. Kharlamov, E., Kotidis, Y., Mailis, T., Neuenstadt, C., Nikolaou, C., Özçep, Ö.L., Svingos, C., Zheleznyakov, D., Ioannidis, Y.E., Lamparter, S., Möller, R., Waaler, A.: An Ontology-Mediated Analytics-Aware Approach to Support Monitoring and Diagnostics of Static and Streaming Data. J. Web Semant. **56**, 30–55 (2019)
8. Kharlamov, E., Mailis, T., Mehdi, G., Neuenstadt, C., Özçep, Ö.L., Roshchin, M., Solomakhina, N., Soylu, A., Svingos, C., Brandt, S., Giese, M., Ioannidis, Y.E., Lamparter, S., Möller, R., Kotidis, Y., Waaler, A.: Semantic Access to Streaming and Static Data at Siemens. J. Web Semant. **44**, 54–74 (2017)
9. Kharlamov, E., Mehdi, G., Savkovic, O., Xiao, G., Kalayci, 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)
10. Ringsquandl, M., Kharlamov, E., Stepanova, D., Hildebrandt, M., Lamparter, S., Lepratti, R., Horrocks, I., Kröger, P.: Event-Enhanced Learning for KG Completion. In: ESWC (2018)
11. Soylu, A., Kharlamov, E., Zheleznyakov, D., Jiménez-Ruiz, E., Giese, M., Skjæveland, M.G., Hovland, D., Schlatte, R., Brandt, S., Lie, H., Horrocks, I.: Optiquevqs: A Visual Query System Over Ontologies for Industry. Semantic Web **9**(5), 627–660 (2018)
12. Svetashova, Y., Zhou, B., Pychynski, T., Schmidt, S., Sure-Vetter, Y., Mikut, R., Kharlamov, E.: Ontology-enhanced machine learning: a bosch use case of welding quality monitoring. In: ISWC (2020)
13. 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)
14. Zhou, B., Svetashova, Y., Pychynski, T., Baimuratov, I., Soylu, A., Kharlamov, E.: Semfe: Facilitating ml pipeline development with semantics. In: CIKM (2020)