Andre Silva Gusmao

Development and Evaluation of an Industrial Condition Monitoring System based on Semantic Technologies and Machine Learning

June 2022

Master's thesis

Master's thesis

2022

Andre Silva Gusmao

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of Science and Technology

NTNU
Norwegian University of Science and Technology

Norwegian University of
Science and Technology

Development and Evaluation of an Industrial Condition Monitoring System based on Semantic Technologies and Machine Learning

Andre Silva Gusmao

Applied computer science NTNU
Submission date: June 2022
Supervisor: Ahmet Soylu
Co-supervisor: Baifan Zhou and Evgeny Kharlamov

Norwegian University of Science and Technology
Department of Computer Science

Development and Evaluation of an Industrial Condition Monitoring System based on Semantic Technologies and Machine Learning

André Silva Gusmão

Abstract

Ensuring the quality of industrial processes product and keeping the cost involved low is highly valuable and desirable among various companies. The quality of a product can be affected by different factors in the production line that often require adjustments and maintenance for the best result and performance. The process of quality monitoring often indicates the quality of the machinery and products to provide information that can be used to seek the optimal product quality and cost.

Industry 4.0 environments tend to produce big amounts of data from their processes. The data produced during these processes can be used in ML techniques to create predictive models that can be used to stipulate the quality and cost involved in the product of the production line. The prediction can be used as a quality monitoring tool for the business decision making process. Applying ML techniques in different and voluminous amounts of data to build predictive models for industrial quality monitoring is costly and presents challenges.

This paper presents a technological research for a system that aims to provide a ML-based condition monitoring solution that uses semantic technologies to address the challenges involved in the process. It presents an extensive problem analysis, system development and evaluation process of our solution.

Acknowledgments

I am grateful for every person that stood by my side and helped me achieve the competition of this thesis. Sir Isaac Newton once said “If I have seen further, it is by standing on the shoulders of giants.”, I make his words mine in a sense that I only got where I am right now because of the willingly help of humble people that guided me during this journey.

I would like to tank in special Ahmet Soylu, my main supervisor, for the wise guidance during my thesis development, directing me in the right direction for a great work. Ahmet Soylu also provided me professional and academic guidance that played a huge impact on my life, this was important and could not be omitted in this Acknowledgments

I would like to thank specially Evgeny Kharlamov and Baifan Zhou for their co-supervision, their patience and guidance during our partnership. From the moment I was a research assistant in NTNU and they proposed me to write a paper in partnership with Bosch until this final date, they were very responsive and always willing to help and guide during my writing.

I would like to provide a special tanks to all the people that took some time to participate in our evaluation process.

I would like to thank my family and friends that stood by my side during this process and also thank specially my sister, mother and father for providing me a direct support during this time.

Contents

Figures

Tables

Part I

Preamble

Chapter 1

Introduction

1.1 Keywords

Condition Monitoring, Semantic Technologies, Ontologies, Manufacturing, Technological Research, Software Evaluation

1.2 Context

The Industry 4.0 [1], also known as 4th industrial revolution is a recent direction in which companies seek a great technology-push in their processes. A recurrent feature involved in the Industry 4.0 is the use of Internet of Things (IoT) technologies [2, 3]. It is also seen an increasing in mechanization and automation of the industry processes that often comes with more digitalization and networking involved [4]. All these characteristics present in this new industrial environment have in common an increase of data generated, that could be seen in domains such as manufacturing, chemical, oil and gas [5, 6].

The reflection of this advance in the industry can be seen in new industrial machinery and production lines that are integrated with sensors, those sensors continuously gather and transmit huge amounts of data. This data is computed, observed and analysed in stations. Those stations organize the machinery and the production environment, and guarantees a via of communication from the production line to the employees.

1.3 Motivation

Because of the abundance of data produced and worked in these Industry 4.0 processes, there has been a surge in the industry and literature in knowledge extraction for industrial processes from different industrial domains such as industrial machinery, gas, fuel, chemistry, and production [7–10]. This desired knowledge extraction is often conducted with the application of Machine Learning (ML) techniques. One possible and valuable use of knowledge extracted from the process data is for conducting the *condition monitoring* of an industrial process.

The condition monitoring, along with other objectives, aims to foresee processes abnormal results, machinery involuntary inactivity, or the grade of produced goods. This monitoring uses enormous amounts of data collected from machinery sensors that will be analyzed to generate valuable insights. Errandonea et al work. [11] presents an increasing trend from the publications related to maintenance and digital twins. Digital twins can be defined as two spaces, physical and virtual, those spaces reflect each other in a way that professionals could analyze the conditions related to the life cycle of the object [12]. This technology is strongly related to condition monitoring and the Industry 4.0, which highlights one important use of the condition monitoring.

Despite this high usability and value outcome from the use of such ML techniques for condition monitoring, the creation of such machine learning algorithms and further implementation of ML models in industrial processes are time-consuming and expensive depending on the project [13]. To identify the complex and costly parts involved in the development of such ML approaches, we did observe an important machine based process conducted by Bosch and how its quality monitoring is done, from this process we could schematize a generic ML-based condition monitoring process and we could identify the challenges involved in it. We did focus our observation in the quality monitoring of one welding process from Bosch, the resistance spot welding (RSW), this process is done by a machine represented in Figure 1.1 (left) that is responsible for connecting metal components by pressing them against each other and passing a current through them [14].

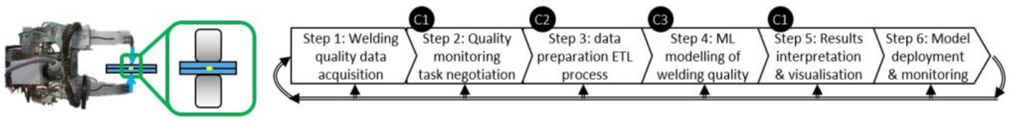

Figure 1.1: "A Bosch machine for automated welding (left) and an ML pipeline enhanced with our semantic modules for welding quality monitoring (right). ETL stands for "extract-transform-load". C stands for the challenges (further described in Chapter 1)" [15].

A proposed approach for developing a ML based welding condition monitoring at Bosch was the one presented in Figure 1.1 (right). The proposed approach workflow is used to guide the development of our proposed system solution. This workflow is depict in a more general lines so it can be adapted to different scenarios and domains for ML-based condition monitoring operations.

The proposed workflow is iterative and involves the following steps: data collection (Step 1), task negotiation, to define feasible and economic tasks (Step 2), data preparation, to integrate data from different conditions and production environments (Step 3), ML analysis (Step 4), result interpretation and model selection (Step 5), and model deployment in production (Step 6).

The main *research topic* of this thesis is that systems that applies and adapt ML techniques for condition monitoring in the industrial environment are often time costly and therefore expensive. Solving this problem, partially or completely, could allow the industry to benefit the positive aspects of applying ML for condition monitoring while avoiding the time and costs drawbacks involved.

1.4 Challenges

The main challenges inherent to the research topic are highlighted in the workflow in Figure 1.1 (right) represented in the circles. This set of critical challenges account for over 80% in the whole development time consumption of systems that applies and adapt ML techniques for condition monitoring [16, 17]. Addressing these challenges would present a contribution to the research topic. These challenges are in depth described at Chapter 4 and are briefly introduced as: *Communication*, related to the interaction between experts from different domains, *Data integration*, responsible for integrating different data, and *Generalisability*, related to the adaptability of predictive models to different scenarios.

To avoid the time consumption and complexity from those critical challenges and provide a solution that can be used in an manufacturing environment, we planned and structured the development of a ML tool for industrial data-driven condition monitoring that can overcome those challenges.

This tool is a software system based on semantics and is called *SemML*. The system extends the conventional ML workflow idea with the use of four semantic components, Ontology extender (OE), Domain knowledge annotator (DKA), Semantic enhanced ML (SEML) and Ontoloy interpreter(OI), those components are in-depth described in Chapter 5. Those components rely on ontologies, ontologies templates, and reasoning for its functions, those concepts are further described at Chapter 2.

1.5 Research questions

With the motivation and challenges from our research topic in mind we set our work based on the following hypothesis:

Hypothesis: It is possible that a ML based condition monitoring industrial system can address the main challenges inherent to it.

The thesis aims to answer the following research questions:

Research question (RQ1): How can a system be structured to handle the main challenges of industrial processes ML based condition monitoring?

Research question (RQ2): How a system resulted from RQ1 fit our industrial use case scenario and how it can be built to be operated by professionals from different domains?

Research question (RQ3): How can a solution to RQ1 be evaluated and how the evaluation process can provide artifacts for further development versions of the solution?

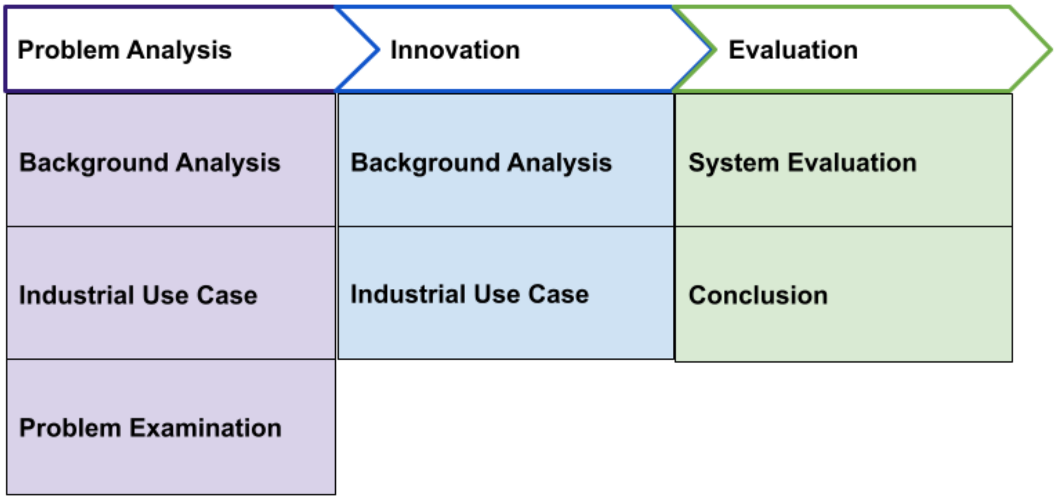

Figure 1.2: Research design adapted from the technological research from the Sintef report [18]. Steps are equal to the report material while the elements in the step were adapted to better fit our project needs.

1.6 Research design

The research conducted in this thesis, that aims to solve the presented research questions, is a technological research, as categorised in the Sintef report [18]. It is described as an iterative process "for the purpose of producing new and better artefacts", this goal is achieved from an initial hypotheses and will be used to create system artefacts that aims to answer the research questions and address the challenges inherent to the research topic. Conducting the technological research allows us to develop a software solution for RQ1, and be evaluated by different professionals under well known evaluation methods to answer RQ2 and RQ3.

The technological research described in the report consists in three main steps, problem analysis, innovation, and evaluation. We adapted a diagram based on the report where each element present in those steps are expanded in Figure 1.2. The elements in each step of our research are further described in this section.

We adopted a technological research methodology by subdividing the three main steps into the following seven elements:

1. **Problem Analysis** is the step where we document to the problem concepts and applications in the literature and industry. With these elements we can define a set of needs and characteristics that will guide the development of our system in the innovation step. The problem analysis step consists of three elements:
   a. **Background Analysis** is the element that contains a research on existing literature of the main concepts related in this thesis development. This element covered broad concepts like Condition monitoring and Semantic components while other elements in the problem analysis

step did conduct a more narrowed research like the study of specific technologies and software alternatives to specific needs in our research topic.

   b. **Industrial Use Case** element is a research on our industrial user case of welding condition monitoring conducted at Bosch, our industrial scenario.
   c. **Problem Examination** element conducts a study on characteristics and positive and week points in existing solutions in the market partially or completely related to the research topic. With this study on existing solutions in the market, along with the industrial use case and background analysis we did define a set of needs and characteristics for our solution for the research topic's challenges that can contribute to the state of the art.

2. **Innovation** is the step in which the compiled information about our solution from the problem analysis step is used to construct our solution. This steps focus on developing the concept and the idealisation of the solution. The innovation step consists of two elements:

   a. **Proposed solution** is the element where the development the concept of a solution for the problem. It is made taking into consideration the information compiled from the problem analysis step (e.g. the system is built aiming to fulfill the requirements from the problem analysis).
   b. **System development** contains the development process and description of our system solution for the problem. It takes into consideration the proposed solution to guide the development.

3. **Evaluation** is the step in which the system is evaluated as a solution to the proposed problem while via well defined evaluation tools. The step also draws a conclusion about the compiled information, the process and the evaluation of the solution to provide information for the development of future versions of the system. The evaluation step consists of three elements:

   a. **System Evaluation** element has the software solution evaluated and tested by final users according to the needs of the system via well known evaluation practices in the literature and market.
   b. **Conclusion** element aims to analyse the whole process conducted in the technological research so far, taking into consideration specially the system evaluation. Its core idea is to highlight strong and week points in our project and how it contributed to the state of the art. This element also contains the future steps desired for the continuation of the development of our solution.

1.7 Thesis outline

This thesis is structured in a way that each chapter receives the same name of one element in the technological research, as seen in Figure 1.2, and represents that element. Chapters 2,3 and 4 are the chapters relative to the Problem Analysis step and represent respectively the elements background analysis, industrial use case and problem examination. Chapters 5 and 6 are present in the Innovation step and represents the proposed solution and system development elements respectively. Chapters 6, 7 are present in the last step of our technological research, Evaluation step, and represent respectively the system evaluation and conclusion elements.

1.8 Contributions

This project is based on the continuation of the creation of SemML system that focus on the front-end development of three main semantic components further described in the thesis, the DKA, SEML and OI components. These three components were previously conceptually idealized but the design, the development process of those components and the user testing and evaluation of them was only conducted in this thesis.

While conducting the technological research to answer our research questions related to the research topics, three main artifacts produced were the main contributions of this thesis:

- The problem analysis step provided a study on the background knowledge, the study of one industrial use case scenario and an study on the characteristics of the problem and existing solutions in the market. This study can be used for the development of future versions of our system solution and work as public material for the development of new solutions related to the research topic.
- The innovation step provided a description of our conceptual solution, the development process, and system features. This system solutions descriptions can be used as material for the development of new system solutions in general, new versions of SemML or for other system solutions for the research topic.
- The evaluation step provided an evaluation process of our solution, along with an analysis over the results and processes conducted during our technological research. The evaluation step provided conclusions about our solution and the evaluation as a whole that are used to guide future development versions of the system and can work as study material for software evaluation processes.

Besides the artifacts achieved during the technological research, we can consider as the main contribution the evaluated system front-end itself. The system solution refereed in this thesis relates to the front-end components since they were projected and behave as they would in real world environments, replicating the real final experience the user would have with the system. It provides answers to the research questions related to our research topic.

1.9 Notes

SemML development started at Bosch Center for Artificial Intelligence [19] and it is still in development. This thesis is the continuation of recent SemML development published papers written in partnership with Bosch [20–22]. This thesis author is a co-author of one of the papers [20] written while he was a research assistenmt in NTNU with a research partnership with BOSCH.

This thesis is based on the presented SemML papers and has as main objective conducting a technological research process on the front-end system components responsible for tasks in steps 3, 4 and 5 from our presented ML-based condition monitoring workflow, Figure 1.1. The main artifact outcomes from this research and thesis can be used in the development of a partial solution for the research topic challenges while also providing public information for further study in the topic. The main outcome artifact contributions are the following:

- Problem analysis documentation
- Front-end system elements for steps 3, 4 and 5:
  - Data preparation tab
  - Feature processing tab
  - Modeling tab
  - Visualisation tab
- System evaluation process and results

Part II

Problem Analysis

Chapter 2

Background

2.1 Condition monitoring

Condition monitoring in the industrial environment is often defined as a set of strategies and methods for monitoring certain parameters seen in industrial machinery's condition, the monitoring is aimed to detect a substantial change that might indicate a developing malfunction [23]. There are two methods of condition monitoring in the industrial environment that are often taken into consideration [20]: *Machinery monitoring* [24] refers to the monitoring of a computational system or machinery present condition of health. The condition of health of an equipment can affect the outcome's quality of the process and can become costly for the production lines; second, there is the *Process quality monitoring* which refers to the monitoring of a manufacturing process's and its product quality. The process quality monitoring often relies on process features extracted from the machinery that play an important role while determining the quality of a process [25].

Either the machinery or the quality of the process outcome is used for condition monitoring in these highlighted methods of condition monitoring. The machinery and products of process can be analysed via Analytical approaches [21] or destructive approaches [26, 27] in processes from different domains. Bosch's welding process uses both approaches, based on data or analysing the products. Other companies like Siemens use data based monitoring of trains [28] and turbines [29] as two instances of equipment monitoring. Equipment and process monitoring at off-shore platforms and oil reservoirs at Equinor [30] are examples in the Oil and Gas business [31]. ABB and INEOS, for example, identify and process issues in the chemical or process industries for root-cause investigation [10].

Condition monitoring is becoming more data-intensive as the IoT [2, 3] and Industry 4.0 [1] technologies advance. The condition monitoring approaches that are based on data to create valuable knowledge for the company uses extracted data from sensors applied on machine learning approaches to predict or forecast one or more numerical indicators of quality. The condition knowledge extracted from the prediction can then be used for important business decision making in regards to the machinery involved in the process.

To conduct a ML based data condition monitoring, a number of processes must be completed: data must be gathered, specific tasks in various domains must be established, data must be prepared, ML models must be analyzed, the findings of the analysis must be evaluated, and finally, ML models for quality monitoring must be deployed in industry.

2.2 Data based condition monitoring and KDD

The condition monitoring aims to extract knowledge in regards to the quality of the outcome of the process or the machinery health. This quality knowledge discovery process based on sensors data can be adapted from Knowledge discovery in databases (KDD) processes. The development workflow of ML based alternatives for quality monitoring based on machinery data is presented in Figure 4.1 (right), it was an adaptation of the KDD workflows from Fayyad [32] and Mikut [33].

Fayyad work [32] describes specific data-mining techniques, challenges of knowledge discovery, and current and future research aspects in the field. During the discussion on knowledge discovery in databases, KDD, the author creates a flowchart with an overview of the steps that compose the KDD process depict in Figure 2.1 along with the outline and description of the basic steps involved in the process.

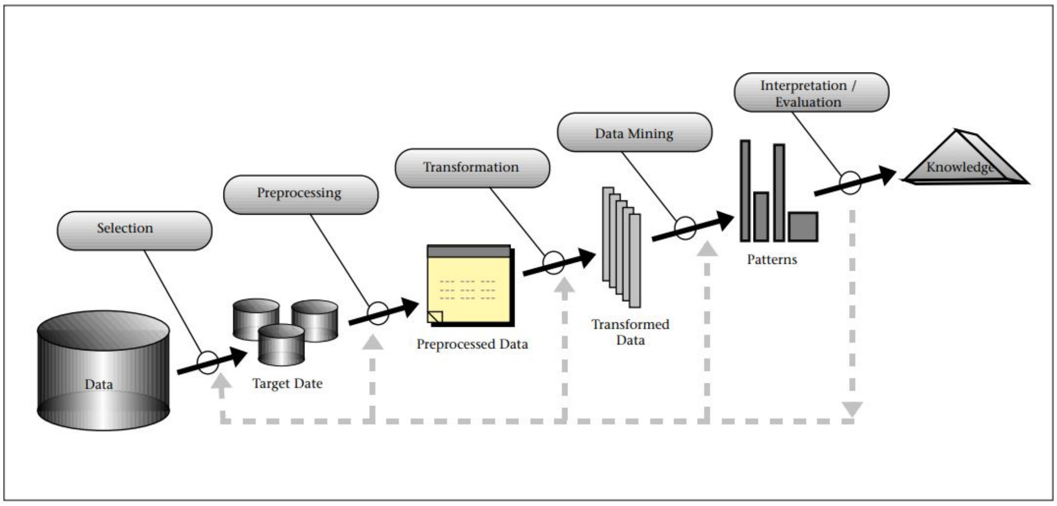

Figure 2.1: An Overview of the Steps That Compose the KDD Process Fayyad [32].

From the KDD overview presented in Figure 2.1, nine basic steps from the KDD process were outlined in Fayyad work:

1. Development and understanding of the application domain taking into consideration the goal of the KDD process and customer's viewpoint.
2. Creating a target data set.

3. Data cleaning and processing
4. Data reduction and projection.
5. Match the goals of the KDD to the specific data-mining method.
6. Exploratory analysis and model and hypothesis selection.
7. Data mining.
8. Interpreting mined patterns.
9. Acting on the discovered knowledge.

The other main reference used for the adaptation and creation of our flowchart of ML development in industrial scenario with machinery data collected for condition monitoring was Mikut work [33]. The paper proposes and discusses about: "modular and computer-based methodology to describe and compare medical problems using data mining methods".

Mikut [33] plans the design process for data mining based on the therapy planning scenario, this scenario often relies on discriminating different decisions and it is a complex and highly interactive process. The flowchart of the process is present in Figure 2.2. It is worth noting how the author highlights the communication challenge, where there is an intensive conversation and discussion from professionals from medical sector and computer science experts, the authors suggest a translation between the description said with the natural language by the domain experts to a more formal language.

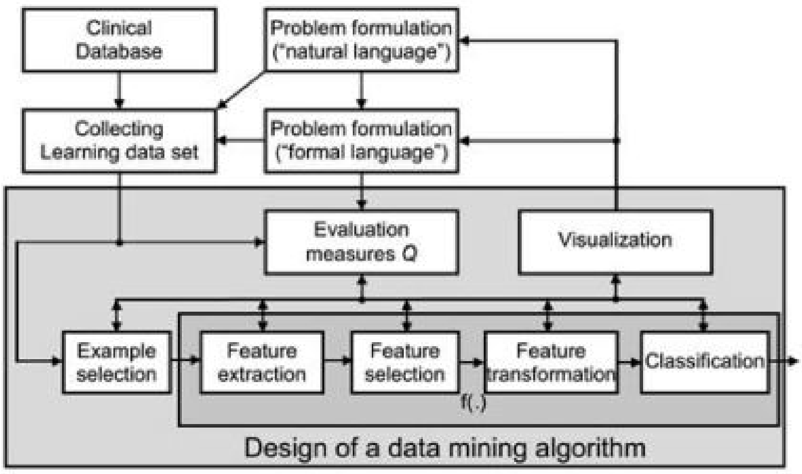

Figure 2.2: Design process for a data-mining algorithm Mikut [33].

2.2.1 Ontologies and condition monitoring

Adopting the proposed workflow for data driven ML based condition monitoring presented in Figure 1.1 inherits some challenges further discussed in Chapter 4. One studied approach to address those challenges is the use of ontologies in our process. How this semantic based approach addresses the requirements and challenges involved in the condition monitoring workflow are described in Chapter 5. The background knowledge behind this approach is described in this chapter.

Gruber describes Ontology as a "formal specification of a conceptualization" [34]. Conceptualization reefers to the modeling of real world concepts in an abstract way. Formal specification means concepts and relationships withing the conceptualization are defined by formal defined terms [35]. Formally describing a domain knowledge in a domain is known as a domain ontology [36].

Semantic technologies adoption faced uncertainty and discussion in the past and as expected in the discussion, the technology became less research centered and more adopted in real-world applications [37]. The use of such semantic technologies have seen an increase in adoption in big companies, IBM [38], Arcelor Mittal [39], Festo [40], Equinor [41], among others.

The top-level or upper-level ontologies were used in our system [42]. They consist in a general information ontology that is used as basis to describe more specific information ontologies. This element is going to be adopted as an example in our case when ML pipeline ontologies are represented by classes from an upper level ML ontology.

The ontology templates are defined as a way to describe and instantiate reappearing patterns in an ontology [43]. This presents use in our system as ontologies can be instantiated with defined desirable characteristics. The ontology reasoning derives information from one ontology that is not explicitly expressed [44]. This aspect is used in our system when raw variables from the sensors are mapped into domain ontology terms via already structured mapping and are further mapped into feature groups, even tho there is no explicit mapping of raw variable to feature group.

Chapter 3

Industrial Use Cases

This chapter is extended from the problem analysis in the technology research, represented in Figure 1.2. In the industrial use case we aim to study potential needs from the problem we aim to solve by analysing one stakeholder's industrial process. By analysing the stakeholder's processes we can have a realistic view of the inherent characteristics to our industrial use case, having a better understanding of the problem and building our solution based on the problem needs.

3.1 Bosch

The process of defining, studying, collecting and analysing the quality monitoring process adopted at Bosch welding processes was conducted in early stages and was previously documented in other papers [20, 21]. In depth information about the data is not presented because it is out of the scope of this project and it could be relevant for the company. Only publicly avaliable information is compiled and presented in this chapter.

3.1.1 Welding monitoring use case

Bosch is among the global leaders in manufacturing automotive components. The welding process is very present in automotive production lines, one standard car body can have up to 6000 welding spots [45]. Bosch welding processes is composed of various elements, machinery (Figure 1.1 left, in case of RSW), softwares, assistance, etc. Those welding solutions provided by Bosch are use in diverse plants around the world, e.g Daimler, BMW, Volkswagen, Audi, Ford. The welding process in big industries of the field, like Bosch, are automated and produce abundant data along with high computing resources behind the IoT technologies.

Making sure all RSW welding spots are in good condition and that the repair and maintenance processes are conducted correctly are essential for multiple production lines. The failure of a single spot can result in an entire car production line stop, which could cause further delays and more expanses in the repair process.

Predicting the quality of a welding spot is difficult, costly and usually makes use of destructing welded parts to evaluate the quality, this process consists in physically tearing apart welded parts to measure the welding nugget diameter. The use of non-destructive approaches, like ultrasonic wave and X-ray, face similar problems as destructive methods, they are relatively expensive, time consuming and produce imprecise outcomes [46–48].

The welding monitoring process aims to avoid damage and problems in the welding process as it also aims to keep a minimal desirable quality to the outcome of the process. The damage and repair needs from RSW can come from the wear of electrode caps that can be easily damaged because of high thermal-electric-mechanical loads and oxidation. The main repair an maintenance processes from RSW comes as after every amount of spots welded a thin layer of the cap is separated in order to restore the surface condition of the electrode cap, this is called Dressing. Another recurrent repair an maintenance process in RSW is the Cap Change, process where after the dressing process is conducted a certain amount of times and due to the removal of the layer on the cap, it becomes too short and should be replaced.

3.1.2 Bosch welding data

Currently the quality of a welding spot diameter is measured by a synthetic variable called Q-Value developed by Bosch Rexroth based on know-how and long-time experience. The variable uses variable indicators to calculate how close to ideal one welding spot diameter is. Q-Values smaller than one means a lack in the quality of the process and a values higher than one would represent a process that used more energy than the necessary for the welding, they could indicate quality deterioration or inefficiency and would require maintenance.

All the data collected from Bosch production line's different data sources for RSW was unified in a 2.74 million records and 44.61 million items from two different welding machines, WM1 and WM2. Workshops were conducted in early stages [21] with specialization professionals in order to separate the most relevant variables for the welding process conducted by these machines.After the workshops were conducted, 4 of process curves (time series features) were considered most relevant for the process: electric current (I), voltage (U), resistance (R), pulse width modulation (PWM).

During the workshops, in regards to single valued features, 188 meaningful variables were selected. They consist of five groups: Program Number are nominal numbers that represent welding configurations, count features are variables strongly related to the damage, repair and maintenance of the process, status are variables responsible for the operation or control status, process curve means is a simple collection of average values from the process curves, and finally the quality indicator that is the variable used to determine the quality of the welding outcome.

3.1.3 Problem definition

It is desirable to keep the Q-Value as close as possible to 1 during the welding operations, keeping this way the quality of the process as ensuring that minimal extra energy is used in the process. The ideal scenario would be a predictive way to measure the quality of the operations before they happen, this would allow preventive changes to being taken in case of a decrease in the process outcome quality like change machinery variable values and replace welding caps.

During the industrial use case analysis, two common welding processes conducted at Bosch were analysed, RSW and hot-staking (HS). The adoption of ontologies to describe these processes, domain ontologies, was done and they were structured in conjunction with professionals of the area. The quality monitoring of both welding process bears resemblance when the processes were semantically structured by domain specialists. This way, same ontology templates could be used to create ontologies for similar manufacturing processes [20, 21].

Since the processes share such similarities, one of the welding processes, RSW, was chosen to be the focus during this thesis development, this way all the testing and evaluation conducted with one process can be extended to the other. The ontologies used in the use case as main sources for the development of our solution are: QMM-Resistance Spot Welding Ontology that describes semantically the RWS process, QMM-ML Pipeline Catalogue that contains the four pre-built ML pipelines in our system, QMM-ML-Pipeline Ontology that describes semantically the encoding of the ML pipelines in catalogue.

Since assessing the quality of the spot welding and proceeding with the repair and maintenance process are so valuable and the current quality and maintenance assesses are often being done via the destruction of the welded part, we proposed a solution to this problem by creating a ML-based system that can predict the welding quality before the real welding process actually happen and maintenance actions could be taken beforehand. The proposed solution relies on the collected data by sensors along with the use of semantic technologies that describe processes and relevant structures of the system, like the ML pipeline.

Chapter 4

Problem Examination

This chapter is responsible for finishing the Problem Analysis step of our technological research, depicted in Figure 1.2. In this chapter we will compile the knowledge about the problem from our background analysis and industrial use case with a study on existing solutions in the literature and market. With the compiled knowledge about our problem in mind, we will set directions to be taken in order to solve the challenges inherent to the research topic, this problem will further be developed and evaluated according to these directions in next chapters.

This chapter is divided into five elements: the existing solutions, where solutions for our research topic challenges from the market and literature were previously analysed; ML development for industrial data-driven condition monitoring element presents an overview about the problem, along with its challenges and requirements; the quality attributes element where the desirable quality attributes of the system solution are presented; Evaluation of the proposed contributions element compiles the evaluation solutions used in our system evaluation process; The thesis contributions to the state of the art element compiles our main contributions.

4.1 Existing solutions

The discussion on existing solutions was conducted briefly in earlier stages of the development [20]. The uncovered points of the solutions were used to guide and motivate the development of our own solution.

The existing solutions and the points where they do not fit our solution needs is compiled as following:

- Usage of semantic technologies in data mining and knowledge discovery, Digital Twins of Manufacturing [49, 50], these solutions only partially meet our requirements further explained in this chapter.
- Ontologies for manufacturing existing solutions [51–56], do not fully serve as the communication model for our use cases and sufficiently cover our domains

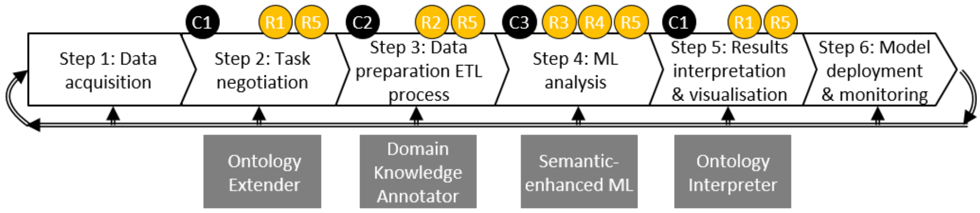

Figure 4.1: "Workflow of ML development for industrial data-driven condition monitoring with indications of *challenges* (C), *requirements* (R), and our semantic components. ETL stands for *Extract, Transform, Load*. [22]"

- Mapping-based data integration solutions [57], do not aim for reducing the ontology expert involvement into the model maintenance.
- The metadata management solutions for data lakes [58, 59], do lacks the extensibility aspect and the integration is laborious
- Existing tools for ontology extension, e.g. template-driven systems [60–62], do require considerable adjustments and could not be easily integrated with our machine learning workflow and infrastructure.

4.2 ML development for industrial data-driven condition monitoring

In this section we will discuss how our adapted workflow from knowledge discovery workflows in databases in literature describe the application of ML development for industrial condition monitoring. The second part in this section presents an in-depth description about the challenges inherent to the research topic, followed by the requirements to address those challenges.

The steps from the workflow, the critical challenges, the requirements and semantic components are represented in Figure 4.1.

4.2.1 Workflow

The workflows from Fayyad [32], Figure 2.1, and Mikut [33], Figure 2.2, were adapted for the creation of our iterative workflow of ML development for condition monitoring in industrial environments, Figure 4.1. The steps from this workflow are presented as following:

- **Step 1 Data acquisition:** This step is responsible for setting how to collect and which data is collected from the specific process. The data is often collected from software and sensors integrated to the machinery, cheaper and effective sensor alternatives have been created and along with advances in wireless communication and mobile networking that have popularized the use of sensors for condition monitoring[63, 64]. This is an iterative process in which sets of variables from sensors are constantly modified to better fit the condition monitoring of the process, this is represented by the backward arrow from step 6 to step 1. The data that is collected in this step is set by data scientists, domain experts and stakeholders, if can be from different data formats, like time series and single features.

- **Step 2 Task negotiation:** In this step experts from different domains like process experts, measurement experts, managers and stakeholders have to communicate so they can achieve a common ground knowledge about the domain process and the ML techniques elements. The understanding of all parts about these elements would guarantee a clear communication between all stakeholders, and an easier understanding and use of the ML techniques in the condition monitoring of the process from all the parts.
- **Step 3 Data preparation:** The data collected for a process can vary withing the same process. Example of the data variability in the same process is the use of different welding machines for the same welding process, different variable names, variables missing, variables collected withing a different sampling rate, etc. This heterogeneous data make the analysis process more laborious, therefore it is desired to prepare the data before, structuring it in an uniform way.
- **Step 4 ML analysis:** The ML algorithms are defined by data scientists and used as a black box solution for experts from different domains by simply selecting algorithms and configuring their parameters. It is also desirable that ML solutions for specific processes to be generalized and used in other similar processes.
- **Step 5 Results interpretation and visualisation:** In this step the results from the application of ML models on the input data should be presented in a intelligible way for professionals from different backgrounds. This extracted knowledge from the application of the model should be used in a business decision-making process, e.g. a model with high precision on quality prediction that relies on a temperature sensor can be used over other quality prediction models, it can also be the case where new and most precise temperature sensors can be bought.
- **Step 6 Model deployment and monitoring:** In this step the business decision-making are chosen and adopted. The adoption of new approaches and models should be monitored closely and be under constant analysis. The analysis of the adopted model can be used in the constant improvement process of defining new quality monitoring models, this can be done by providing knowledge back to any of the last steps.

4.2.2 Challenges

Multiple challenges were detected in all of the workflow steps. After studying our problem in the industrial environment and in the literature we have selected three critical challenges for various companies, including Bosch, that account together for over 80% in the whole development time consumption [16, 17] and our solution aims to work on these critical challenges in order to propose a solution for the research topic. These challenges are represented in the workflow from Figure 4.1 along with the steps where these challenges can be seen. The critical challenges are the following:

- ***Communication, C1***: This challenge can be seen in steps two and five and refer to the need of collaboration of specialists from different fields, all of whom have different backgrounds knowledge, which makes the communication a laborious task with high time consumption and highly susceptible to human error.
- ***Data integration, C2***: This challenge can be seen in the third step and refer to the need of human integration of data from various sources that require highly manual modification. This manual modification from different sources is time consuming and can scale according to the number of different sources the data will be extracted.
- ***Generalisability, C3***: The third challenge seen in the workflow is the system ability to generalise machine learning models from one specific dataset and application scenario to another, currently this process requires a large amount of effort. The ML models are often built in order to solve a specific problem with a specific dataset in mind, as seen in the outcome of Step 4 in our ML pipeline. Since the industrial process face relevant variability among the elements present in the process a re-work in models produced to adapt to different scenarios is highly desirable.

4.2.3 Requirements

Solving the previously presented costly challenges would promote a solution to the research topic completely or partially. A set of five system requirements were defined in order to address these challenges. Challenge C1 is addressed by requirements R1 and R5, while challenge C2 is addressed by requirements R2 and R5, and challenge C3 is addressed by requirements R3, R4 and R5, as represented in Figure 4.1. The system requirements are described as following:

- ***R1*** *Uniform communication model for various stakeholders*: This system requirement can be seen in step 2 that separates the process information that can be relevant in the creation of the condition monitoring models for the process. In this step it is desirable a common vocabulary that could allow an unambiguous, clear and machine readable communication between the experts from different domains, e.g. domain experts describing process elements that could be relevant for the condition monitoring to data scientists. Step 5 also can take advantage of this common vocabulary by facilitating the explanation of results to experts from different domains.
- ***R2*** *Uniform data format and ML vocabulary*: This requirement is present in a single step, step 3, and it refers to the data integration challenge present in this step. Step 3 requires data from different sources to be integrated to be used in the system, the possibility of storing the data in an uniform format

along with an uniformed variable names convention is this requirement.
- ***R3*** *Mechanism for generalising ML models*: This requirement present in step 3 is related to machine learning methods, where these methods developed for a specific dataset could be applyied and generalised to different datasets.
- ***R4*** *Data enrichment mechanism*: In order to facilitate the generalisability challenge, C3, present in step 4, the system requirement of data enrichment mechanism allows complementary information to be assign to the data making this integrated data connectable with other generalisable machine learning approaches.
- ***R5*** *Flexibility, extensibility, maintainability*: This system requirement is responsible for keeping the system dynamic. The possibility of adapting the system in a constant evolution represented by the arrows that go backwards in the workflow are addressed by this requirement. This adaptation should be facilitated by the system.

4.3 Quality attributes

The quality attributes, i.e. nonfunctional requirements, are responsible for assuring a desirable performance of the system [65], they are linked to behavioral properties desired in the system, like performance and usability [66]. The quality attributes analysed in this system follows the 11 obtained quality attributes by Khalili and Auer [65] in their system based on semantics. The fulfillment of these attributes are done with system features that take care of a quality attribute, e.g. the system customizability attribute can be solved with system feature of the use of ontologies that allows customizability in the use of the system.

A in depth description of how the system features fulfil the quality attributes are further described in Chapter 5. The 11 adopted quality attributes by Khalili and Auer [65] are shortly described as following:

- *QA1 Usability*, how the user experience is conducted. How it fulfil the user's expected quality of the system. Iso 9241 defines usability as effectiveness, satisfaction and efficiency in the use of the system by the user in a specified context of use [67].
- *QA2 Customizability*, how the system can be changed to fit users needs and/or preferences.
- *QA3 Generalizability*, how the system can be adapted to different scenarios and use cases.
- *QA4 Collaboration*, how the system can handle cooperation between different users.
- *QA5 Portability*, how the system can be used in different environments.
- *QA6 Accessibility*, how the system is built in order to be used in a simpler manner by as many people as possible.
- *QA7 Proactivity*, how the system should be built aiming to adapt in a proactive way, before events that require adaptation happen, instead of in a

reactive way, after events that require adaptation happen.
- *QA8 Automation*, how the system should be built aiming to reduce the need for human work. This can be seen as increasing the number of functions done by the machine exclusively instead of functions conducted entirely or partially by humans.
- *QA9 Evolvability*, how the system can adapt according to future needs in a simpler way.
- *QA10 Interoperability*, how the system can communicate and adapt to other systems.
- *QA11 Scalability*, how the system can scale in relation to number of users and load while keeping the performance in a desirable level.

4.4 Evaluation of the proposed contributions

The evaluation step in the technology research adopted aims to evaluate how the proposed solution (outcome from the innovation step) performs at assessing our research questions. In this section we will explain which evaluating approach adopted and the reason behind that choice.

Testing the usability aspect of our system is a critical aspect to be evaluated in our solution, the usability is also one of the quality attributes, ***QA1 Usability***, that are analysed to assure the desirable performance of our system. Besides that, in order for the solution to be applied in an industrial scenario as proposed by RQ2, it should present an acceptable level of usability. The system evaluation results are artifacts that can be used in further development versions of the system solution.

To evaluate the usability of the system we chose the *System Usability Scale (SUS)* [68] due to its high adoption in the literature and industry, being referenced over 1300 times in publications and articles [69]. The system consists in a set of ten questions that can be answered in an scale from one to five in which one is strongly disagree and five is strongly agree.

The workload is defined by Hart and Wickens [70] as the effort an human puts into the performance of a task. The human factors principles aims to design systems, that among other objectives, tries to reduce human error and increase productivity [71]. The practice of assessing tasks workload has been seen present in human factors journals over the years [72].

A reduced workload could represent a less prone to human error task, since it would be less demanding for the user under some situations [73]. Reducing the human error and increasing the productivity of our system is a desirable characteristic since it could result in better and safer uses of the system in the industrial environment from RQ2. The workload evaluation results is an artifact that can be used to help in future development versions of the system solution, assessing this way RQ3. A lower workload is also strongly related to the quality attribute ***QA8, Automation***.

To evaluate our workload we adopted the use of *NASA Task Load Index (NASA-TLX)* [74] due to its huge adoption in the industry and literature. The tool consists in one step where the user weights the workload factors by relevance and one step where he fills a form with the six scales for each workload factors according to his experience.

Besides the presented evaluation approaches, quantitative and qualitative variables are evaluated in order to get insights that could influence in the future builds of the system, RQ3. We specialisation area of the subjects and the time consumed in each part of the evaluation process. It is also collected how familiar the subject is to machine learning concepts.

The evaluation process is conducted by submitting the subjects to scenarios and tasks that could evaluate all the system requirements. The tasks and further details about how the process was conducted are seen in Chapters 6 and 7.

4.5 Thesis contributions to the state of the art

This thesis conducted a complete technological research [18] for the development of a technical solution capable of fitting the requirements and attributes needed for a system to address the research topic's challenges. These elements are related to the research topic and answering the research questions will provide a partial or entire solution to its challenges. The main outcome elements that contribute for the state of the art from this process are:

1. A solid compiled study for future work about our specific research topic. Our work can be used as material to guide mainly any project development linked to the research topic.
2. A version of a system solution for our research topic. The structure and the system by itself provide a partial solution to our research topic's challenges.
3. The evaluation step can provide a meticulous description of the use of evaluating methodologies for similar systems. The evaluation also provides data that can be used to conclude about how efficient our system was at answering the research questions and can be used in the development of future versions of our system or similar systems.

Part III

Innovation

Chapter 5

Proposed Solution

In this chapter we described the theory and organization of our proposed solution for the presented research topic's challenges, the development of a ML tool capable of industrial data-driven condition monitoring that fulfill our industrial use case needs and overcome the challenges present in the process. To achieve such objective we conducted a study on the characteristics of the problem. This study was conducted over publications in the literature, study of partial solutions existent in the market and our use case study, this study is presented as the outcome of the problem analysis step.

The outcome of the problem analysis is used as guidelines that did direct the way our proposed solution was structured and developed in Chapter 5 and Chapter 6 respectively. The system elements that compose our solution for the research topic are divided into system components that are seen in Figure 4.1 and are further described in this chapter.

5.1 Overview

The development of this project is a continuation of the papers [20–22], those papers did focus on the analysis and development of the solutions for requirements present in the step 2 and partially on the component for step 3, as seen in Figure 4.1. These components were the OE and part of the data mapping present in the DKA. These previously papers also conducted a evaluation of these components, and theoretical and structural study of the solution.

As a continuation of the system development we did increase the information in the problem analysis, the theoretical part and structural part from the previously published papers. We did focus on defining requirements, identifying quality attributes, structuring the implementation of the components, mapping quality features and proceeding with the user evaluation of the semantic components that were not developed previously, those components are the DKA, SEML and OI.

This chapter presents the theoretical part behind our solution, this is responsible for justifying how our system should be developed in order to fulfill the requirements and quality attributes needed and desirable for providing a solution to our research topic. The chapter is divided in the following sections:

1. **Semantic technologies**: This section contains the explanation of the adoption of semantic technologies in our solution.
2. **Semantic components**: define how semantic solutions can be used and divided into components to solve research challenges.
3. **Development process**: define the adopted development process for better fitting the system development needs and characteristics.
4. **System architecture**: presents how the system should be structured in order to fulfill its objectives.

5.2 Semantic technologies

Providing a solution to our research topic would require us to address the challenges inherent to the application and adaptation of ML techniques for industrial data-driven condition monitoring. To address these challenges one of the studied alternatives that was adopted was the use of semantic technologies as main element in our system construction.

Semantic technologies have received a lot of recent attention in the industrial environment in areas like modeling of industrial assets [75], analytic study in industrial environments [76], querying based on data from the production environment [77–79], integration [29, 30, 80], process monitoring [81], and equipment diagnostics [28]. Semantic technologies have been adopted or evaluated in a number of large high tech production companies such as Siemens [82], Festo [40], Equinor [41], and Bosch [83, 84].

Besides the recent attention, the use of semantic technologies could fit the requirements of our solution as following:

- R1, Uniform communication model for various stakeholders, could be addressed by the adoption of semantic technologies by providing a way of describing the domain processes via ontologies. These process ontologies (domain ontologies) are written in a structured way that could facilitate the communication between different stakeholders found in steps 2 and 5. E.g. the name of the welding machine can have a different name in a specific welding process, which could be confusing for the manager, if the process is described based on the core ontology that term can be associated to the core concept and the term would be clear for the general manager.
- R2, Uniform data format and ML vocabulary, could be addressed by standardizing the raw input data from different sources by mapping the input variables to the correspondent terms in the domain ontology. E.g. the temperature from a German machine could be represented in celsius with the name "tmp" while an american machine could represent it in fahrenheit with the name "temp", the mapping of the raw temperature data can be standardized in an uniform format and name, the uniform data mapped to the

domain ontology is then ready to be used in the ML modelling process.

- R3, Mechanism for generalising ML models, can be addressed by the use of semantic technologies by saving the created ML pipeline for a specific data-set and purpose as an ontology in a ML models database. E.g. a model for condition monitoring for a specific welding process that relies on a specific sensor data that is not present in another machine can be easily detected and adapted via ontologies to keep a similar structure while not including a specific ML operation that relies on that data.
- R4, Data enrichment mechanism, can be addressed by the use of semantics by assigning descriptive ML relevant information to the variables. E.g. a variable "plate angle" from a data-set can be assign to the feature group single feature, since it is a single instance of a value, and ML methods related to single features can be applied to that variable.
- R5, Flexibility, extensibility, maintainability, these factors can be addressed by the use of semantics since the system can adapt to new process ontologies and core ontologies, providing this way more flexibility and extensibility. The system can also provide an easy maintenance process due to the ease to adapt maintenance changes to the system provided by flexibility in the process ontologies, this maintenance aspect is also represented by the arrows that go backwards in the workflow from Figure 4.1 due to the constant desirable maintenance of the system.

Due to its increase in adoption in industrial environments and the possibility of fulfilling the requirements inherent to our research topic we decided to adopt the use of semantic technologies in the construction of our system. An in depth description of how the use of semantics is applied in our solution is described in the following section.

5.3 Semantic components

The construction of the system that aims to provide a solution to our research topic is guided by our problem analysis process. In this process we developed a workflow of ML development for industrial data-driven condition monitoring, we identified the challenges and requirements to cover those challenges that are inherent to the process of developing a solution to the research topic, besides that, we also defined a set of quality attributes that define desirable performance aspects of the system. The conjunction of this elements from the problem analysis are used to guide the development of our proposed solution.

Our semantic solution is divided into four system elements, here described as semantic components. These elements are responsible for performing the tasks that contains the critical challenges presented in the workflow from Figure 4.1, the steps with those challenges are Step 2 to 5. An architectural overview of our proposed solution expanded from the workflow can be seen in Figure 5.1. The four semantic components concepts of our solution are presented as following:

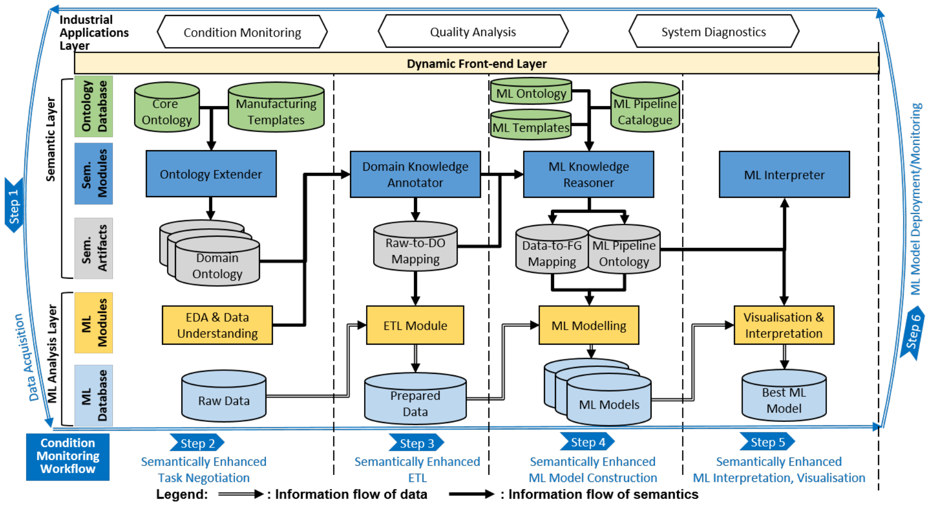

Figure 5.1: An architectural overview of our semantically enhanced ML solution for condition monitoring, where we overlay the welding quality monitoring workflow of Figure 4.1 and the use-case requirements. EDA: explorotary data analysis, Sem.: semantics, Eng.: engineered [20].

5.3.1 Ontology extender

Overview

The data acquired from the sensors in the industrial environment from the machinery, the outcome from step 1, is iterative and faces constant modification to better fit the condition monitoring of the process. Due to the constant changes of the collected data and the fact that this data comes from different sources with different nomenclature and formats, the process of working with this kind of data and extracting knowledge out of it is a costly and time consuming task.

To solve the data from different sources challenge and facilitate the process of communication of data to ML pipelines, R2, a semantic ETL pipeline was designed in our system. The ETL pipeline, as seen in Figure 5.2, consists in a three steps pipeline where the data is collected from different sources in the extract phase, it is transformed according to a set of business rules and it is then stored in a database.

The transform process used in our semantic ETL rely on domain ontologies. The second phase of our workflow aims to provide the semantic component OE, a tool that will allow insertions in our domain ontology graph database. The domain ontologies are going to be used in the DKA in step 3 by providing an upper level semantic structure that will be used when mapping the raw data to domain data and preparing it to be used in the ML modeling process in step 4.

The domain ontologies also provide a structured uniform communication model between various stakeholders, R1, where processes can be described withing a set of defined restrictions (templates) and based on upper-level ontologies (core

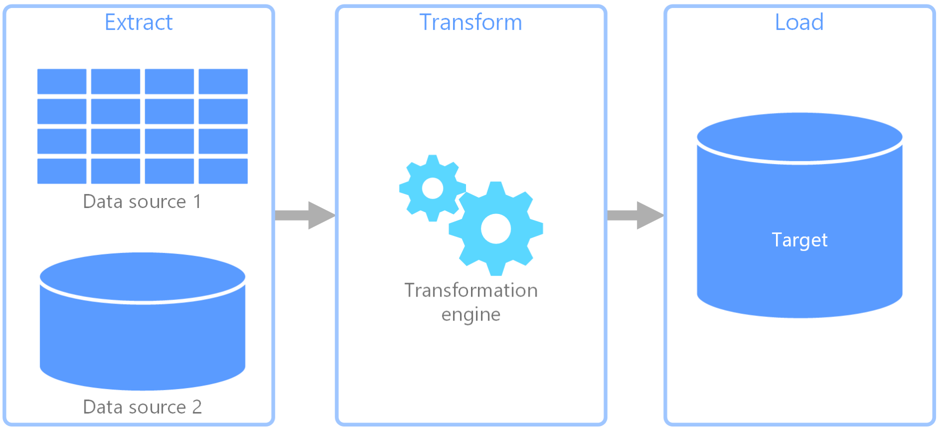

Figure 5.2: Extract, transform, and load (ETL) process description from Microsoft documentation [85]

ontologies). This would allow the association and description of domain components with elements in the core ontology that works like a common communication model, facilitating this way the requirement R1. The use of semantic technologies also facilitate R5 as discussed in section 5.2, besides that, the templates allow flexibility and extensively to new data sources added in the system.

Process

Domain experts and data scientists use the ML module of Exploratory Data Analysis (EDA) to discuss and specify the procedure of quality analysis of the specific process. EDA tool is used summarize the main characteristics of a data set and is often followed by visualization tools [86]. The better understanding of the process data provided by the EDA allows the experts involved in this step to better describe and understand the domain process.

When the experts involved in this step have a good understanding of the process and the data, the upper-level description of the general knowledge of that manufacturing process is encoded as the core ontology. The core ontology, for instance, can describe how the general welding process is conducted, while not going in depth about specific welding processes like HS or RWS.

The templates are based on the core ontologies, they contain a set of defined restrictions that will allow domain processes to be encoded based on the upper-level core ontology. E.g. an element in the specific domain process can only be encoded in the domain ontology if it reflects a specific hierarchy in the general core ontology. This would guarantee that the domain ontology created by the users would have a very good quality when inserted in the system since this ontology respects the upper-level hierarchy.

Specifying the procedure of quality analysis of the specific process, encoding core ontologies and defining templates are tasks done during the creation and

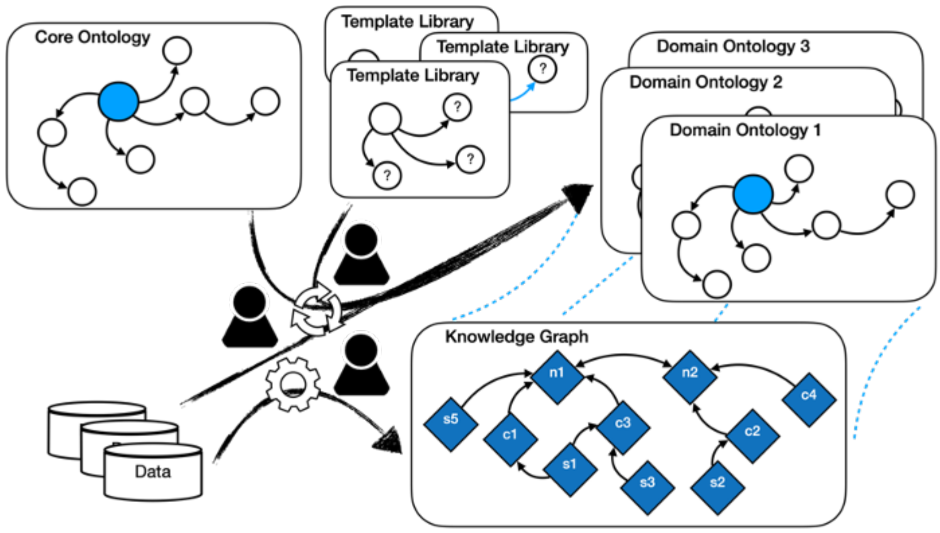

Figure 5.3: Schematic of OE process [87]

maintenance of the system. They are pre-requisites to the day-to-day use of the system.

A schematic of the day-to-day use of the OE is represented in Figure 5.3. The use of this component provides a mechanism in which the experts can encode domain specific knowledge as domain ontologies via defined templates. These templates guarantee the good encoding of the domain ontology in a way that respects the upper-level core ontology hierarchy.

5.3.2 Domain knowledge annotator

Overview

With the domain ontologies encoded from the process in step 2, the semantic enhanced ETL process can be finished in step 3. In step 3 the raw data extracted from the different sources is then mapped into terms in the domain ontologies (Raw-to-DO Mapping, DO means Domain Ontology) automatically or manually.

With the Raw-to-DO Mapping done, the transform process can be finished as the raw variables collected are prepared by the ETL module for the prepared data that will be used as input to the ML model constructor in step 4. The uniformization and preparation of the data to be used in ML model constructor in step 4 fulfils the requirements of R2.

The possibility of extending and adapting the exiting Raw-to-DO Mapping to new mappings guarantee the flexibility and extensibility of the system. These aspects along with the iterative workflow of our system allows the maintainability of the system as well, fulfilling this way the requirement R5.

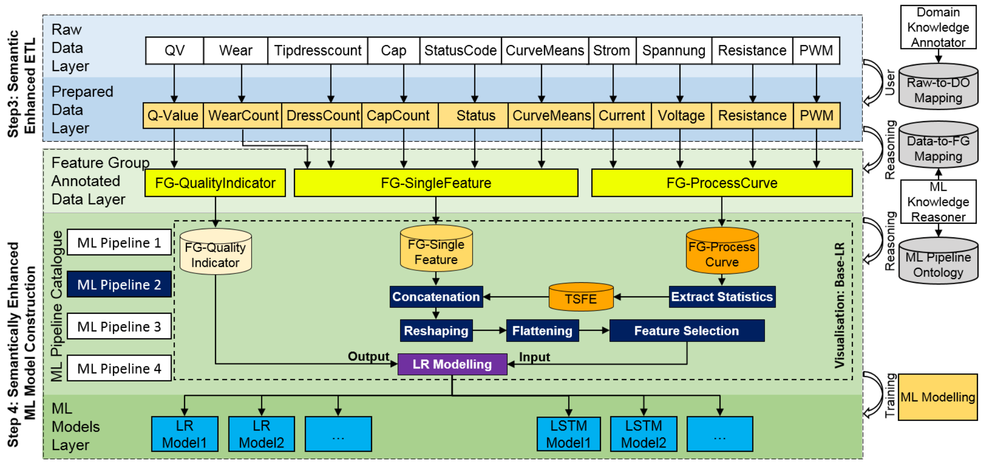

Figure 5.4: Mechanism of SEML model construction in “Static Mode”: data preparation and ML pipeline selection from a catalogue without dynamically changing the ML pipeline structures

Process

An example of the process conducted in step 3 and step 4 can be seen in Figure 5.4. The DKA provides a Raw-to-DO mapping that maps the raw variable collected in step 1 into domain names from the domain ontologies provided by the step 2. This mapping process is constructed based on previously mappings done by users, in our example the mapping of a raw variable "Wear" from a sensor that was previously mapped into the domain term "WearCount" is automatically done if there is a history of this mapping before. If a change is required or the automatic mapping could not be done, for instance if there is no history of that mapping, a manual option for adding or editing the Raw-to-Do mapping is available.

After the mapping, the data is operated in the ETL module and become prepared for being used as input for the next step. The variables identifiers are modified into unified feature names according to the Raw-to-DO mapping. E.g. "Tripdresscount", in raw data layer, is prepared and mapped into the domain ontology term "DressCount", in prepared data layer.

5.3.3 Semantic enhanced ML

Overview

The semantically enhanced ML model constructor consists in a component that relies on the prepared data layer, outcome from step 3 along with reasoning from the Raw-to-DO to input the data correctly in the ML pipeline according to the feature group of that variable. This mapping of the raw data to the domain terms to the feature groups conducted in step 4 is responsible for fulfilling mainly the requirements R4 and R5, where the enrichment of the raw data with the mapping of the feature group guarantees that the mapped data can be linked to generalisable machine learning approaches.

After the data is mapped in the correspondent feature groups, the user can opt for constructing his own ML pipeline in our dynamic interface or selecting a ML pipeline from a catalogue of ML pipelines. The pipelines are constructed based on ML templates and in a ML ontology in order to guarantee a good and proper construction of the pipelines. This process is responsible for fulfilling mainly the requirements R3 and R5, where pipelines developed for a specific scenario and dataset could be generalised or reused in different scenarios and datasets for the construction of different ML models.

After the ML pipeline is constructed the ML modeling process is conducted, where the ML model algorithm is selected and adjusted. With the model configurations set the model is trained and tested. The outcome of step 4, a trained and tested model, is submitted to the next step of the workflow.

Process

The raw data from sensors was prepared to the ML pipeline and was assigned to the domain terms in the prepared data layer in step 3. After this, the ML knowledge reasoner uses the ML Ontology to infer ML relevant information to the prepared data by mapping Raw-to-DO terms into feature groups (FG) in the Data-to-FG mapping, as seen in Figure5.1 in step 4. Mapping Data-to-FG is important to ensure a proper work with the prepared data, e.g. the methods applied in single features FG are different than the ones applied in ProcessCurves FG and this should be reflected in the prepared data used in the pipeline. This process is done automatically by semantic reasoning, an element capable of inferring logical consequences from a set of axioms [88].

This semantic component can be used in 2 different modes, static mode or dynamic mode. The process presented in Figure 5.4 shows the process conducted in step 4 by the SEML model constructor component in static mode. The dynamic mode was not represented in previous papers since those papers only evaluated elements in the OE and the DKA and the static mode was sufficient for this evaluation.

If the user opts for the dynamic mode he can edit loaded pipelines or create his own from scratch. The tool allows a dynamic free construction of the pipelines. The process of constructing pipelines that can be included in the catalogue can be seen in Figure 5.5. Despite giving freedom in the ML pipeline creation process, the tool reflects the restrictions imposed by the ML template libraries and the ML ontology. By following the templates and the ML ontology we can guarantee a proper and good construction of the ML pipeline, e.g. in the pipeline layer 2 comes before layer 3 and only correct algorithms can be selected for each feature group.

The static mode of the ML pipeline works as a black-box structure, where the user can select a pipeline from the stored catalogue but no editing of the pipeline

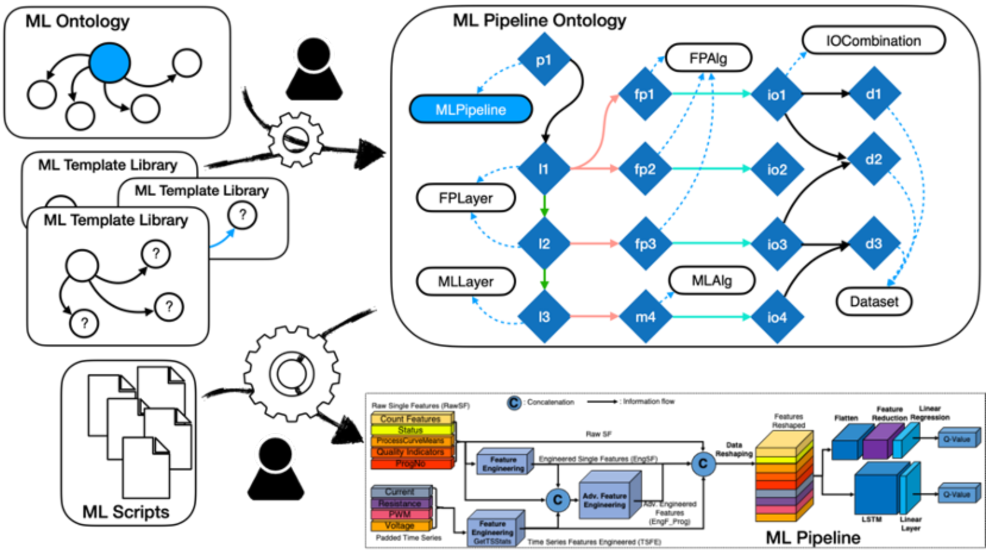

Figure 5.5: Schematics of ML pipelines creation process.

is done. This approach is necessary since the re-use and generalisation of pipelines is highly desirable in our system.

In the ML modelling the data used for the process is processed by the scripts selected from the ML pipeline. A predictive model algorithm is selected and configured among the available options and the model is trained and tested using the data inputted in the ML modelling. The results are visualized and presented in the next workflow step, OI.

5.3.4 Ontoloy interpreter

Overview

The easy visualisation and interpretation of the results from the predictive model are highly desirable in the system since they would allow a facilitating communication tool that would allow stakeholders from different backgrounds to analyse the tested model. This tool is important in the decision making of maintaining current deployed models or deploying new ones, this way, the component OI, also known as ML interpreter, addresses the requirements R1 and R5.

Process

The interpreter component will allow the plot in different models of the estimated and target values of the training and tested values with the created model. An example of the training and test values from the model can be seen in Figure 5.6 where a good predictive model plot would be closer to a line of test points going from bottom left to top right, meaning the estimation Q-value present in X axis is the closest as possible to the target Q-value present in the Y axis.

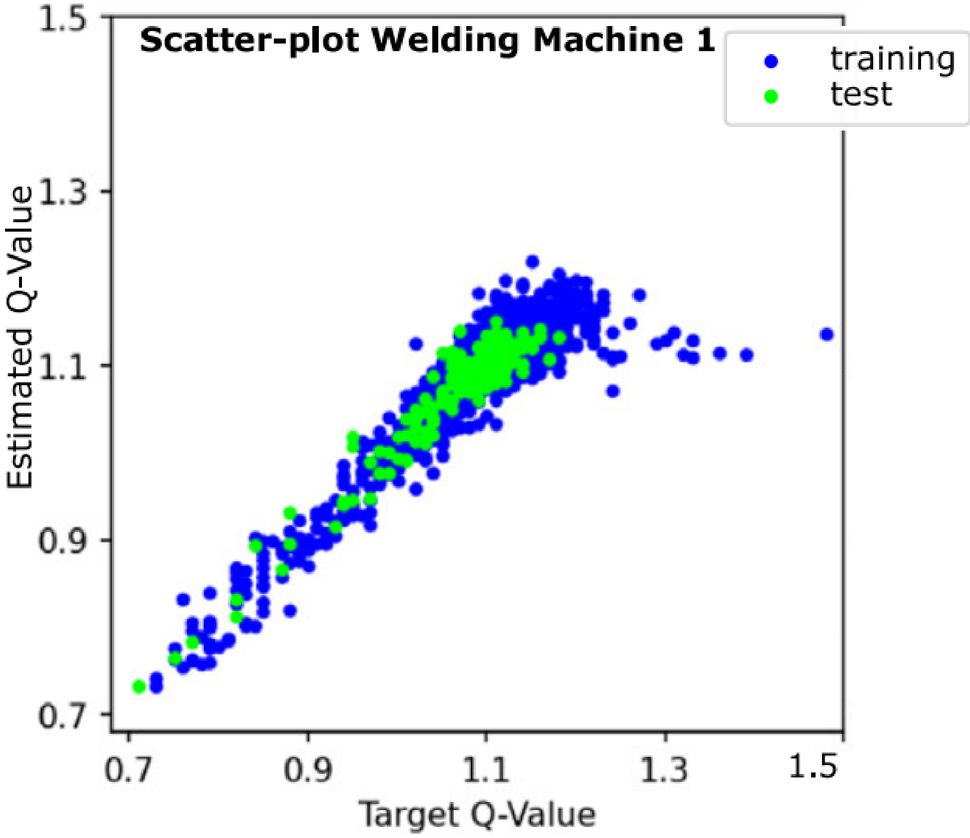

Figure 5.6: Prediction results of Advanced Linear Regression (LR) for WM1. Advanced-LR presented the best prediction results for WM1 Bosch user case in the used data set.

The interpreter component will facilitate the communication between stakeholders from different backgrounds and assist the decision making process related to the adoption of predictive models.

5.4 Development process

Our development process workflow was based on the prototype model [89], with workflow model seen in Figure 5.7. The initial requirements were collected at the end of our problem analysis step and they were used to design each component that would address these components.

The design and prototyping of the components was made with Figma tool [91] and every cycle of customer evaluation for the components was made with the product owner, that worked as a middle ground between the developer and the customers. This happened because the direct contact with the domain experts, final users, was restricted due to time constrains.

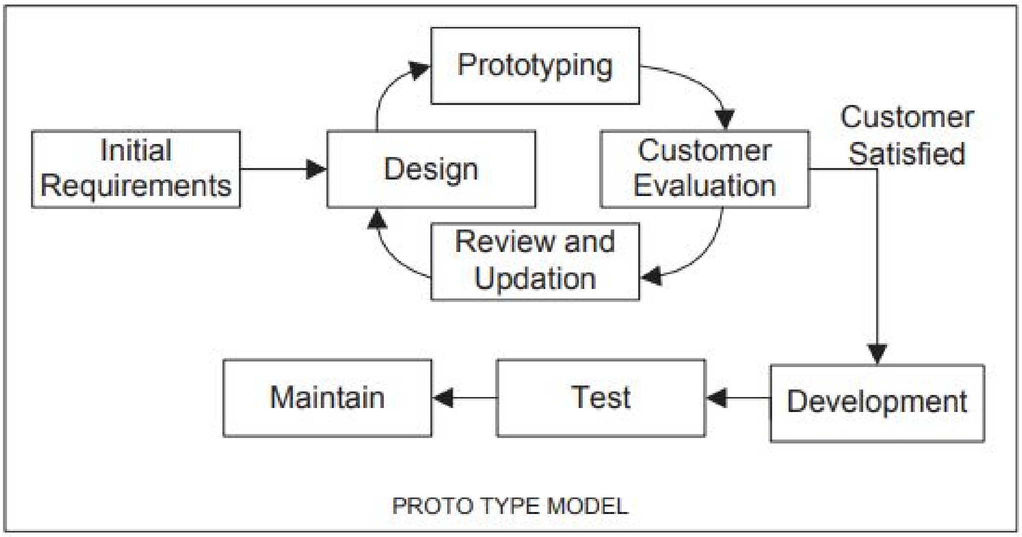

Figure 5.7: Representation of Prototype Model from Shikha et al. [90].

Before the development process we conducted a discussion on which technologies would be adopted in the system and which architecture our system would have. The study on technologies adopted is presented in section Development tools in Chapter 6 and the study on the system architecture is presented in the section system architecture in Chapter 5.

The development process framework adopted was Scrumban, represented in Figure 5.8, it was chosen due to its high adoption, 81% of the Scrum masters integrate Kanban with Scrum [92], the fact that the involved developer and scrum master were familiar with Scrumban and the lack of work estimation in the framework, which was important factor in our case since the time window for the total development was short, 6 months. For managing the kanban board We used Trello system [93].

In our case some adaptations were taken most of the time due to the small number of people involved in the team. The scrum master and the product owner roles were taken by the same person, the sprints took 1 week and we chose to not follow strictly the framework, e.g. some elements like daily meeting were not severely followed and the meetings were every one, two or three days.

Not all components were developed in depth due to time constrains. OE was already developed and evaluated in the last presented articles about SemML, a new version of the DKA was developed where functions were added, the SEML component was developed and only a simpler and mostly static version of the ontology interpreter was developed, just so during the evaluation step the users were submitted to the whole workflow.

An in depth description about the development of the system components can be seen in chapte 6. The testing of the developed changes in the system were conducted in the evaluation step in Chapter 7.

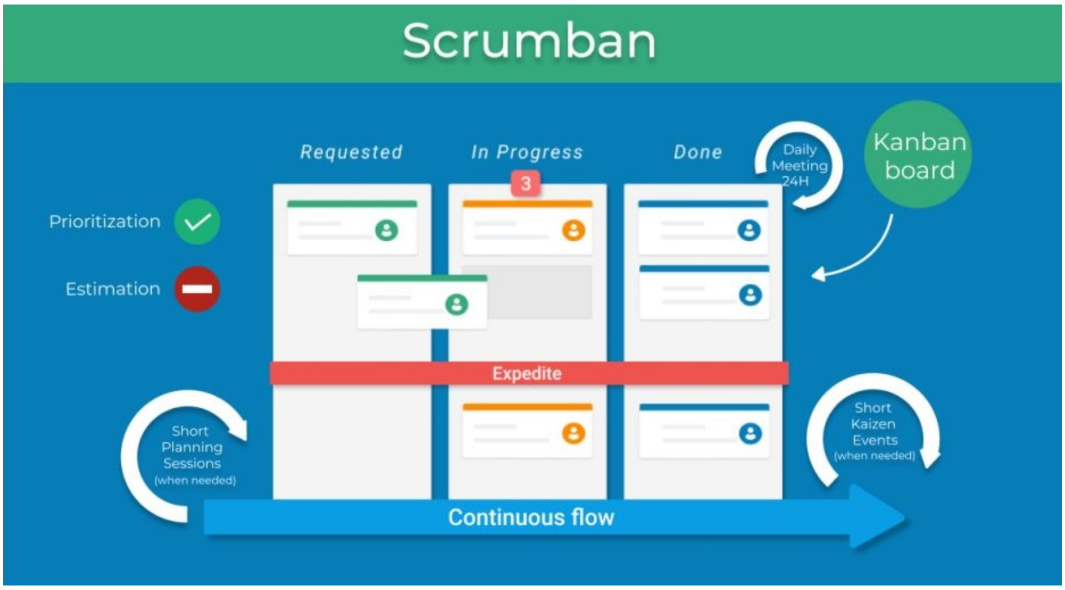

Figure 5.8: Scrumban diagram representation from Schwaber and Sutherland [92].

5.5 System architecture

5.5.1 System architecture decision making

With the design, requirements and prototypes approved by the product owner we set a diagram, Figure 5.9 in which we did establish which system architecture approaches would be adopted in SemML. The inclusion of solutions in the diagram did not follow a systematic study on existing architectural approaches in the market and literature due to the limited time involved in the development of the system and the possibility of re-adapting the system in the future, the solutions were added as the development team members conducted a short time research on possible approaches, the solutions are not necessarily excluding and might even represent different architectural aspects, e.g. development for mobile devices can also be re-used or adapted to Smartwatches, they represent in which direction the system architecture will be designed towards. The decisions were taken based on three questions:

- **Q1**: Which device will the system mainly run on?
- **Q2**: Which solution can be applied for the development of the computer software?
- **Q3**: Which web app specification better fits our development process of the system?

Q1 - Computers were selected as the main devices to run the system since they are more used in the Bosch use case environment and the industrial environment in general. Computer also allow a easier manipulation of files, which is an

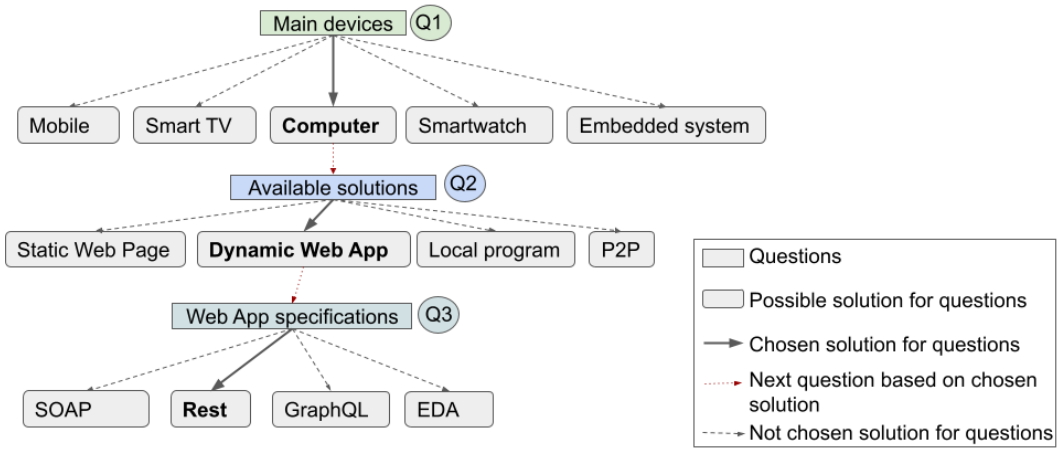

Figure 5.9: Diagram with questions and solutions related to architectural decisions in our system. Local program means programs installed in the user's machine.

recurrent activity while dealing with the process data collected.

Q2 - Dynamic web app solution alternative was chosen mainly because one desirable aspect of the system is the Scalable Machine Learning on the Cloud. This aspect consists in ML solutions and data access on a cloud scale that deals with big data, this concept is represented in the diagram in Figure 5.10.

Q3 - An analysis on the web app specifications that would guide the direction our project will go towards is presented in the following points:

Simple Object Access Protocol (SOAP) is a messaging protocol that presents a strong security aspect but requires more bandwidth, it only works with XML format and cannot make use of REST [94], which does fits perfectly to our system specifications since security is not a main aspect involved in the system and we want to keep the communication open to other API's, which is more restricted with SOAP since the only accepted format is XML and it cannot make use of REST from other systems.

GraphQL is a query language for APIs. It uses a client-driven architecture, presents no caching mechanism and has response output in JSON [95]. This option does not fits perfectly to our system specifications since dealing with JSON only is an excluding factor in the integration aspect of the system with other systems. This option also does not present caching which could cause unnecessary calls to the server and longer responses.

The event driven architecture (EDA) paradigm is oriented on events, where EDA keeps monitoring events in order to answer them. The other studied option, service oriented architecture (SOA), is based on providing services as requests come to the system, the SOA architecture is seen in the SOAP, REST and GraphQL solutions.

Following Bukhsh, Sinderen and Singh work discussion [17] on when each architecture should be used two main points were important in our decision of

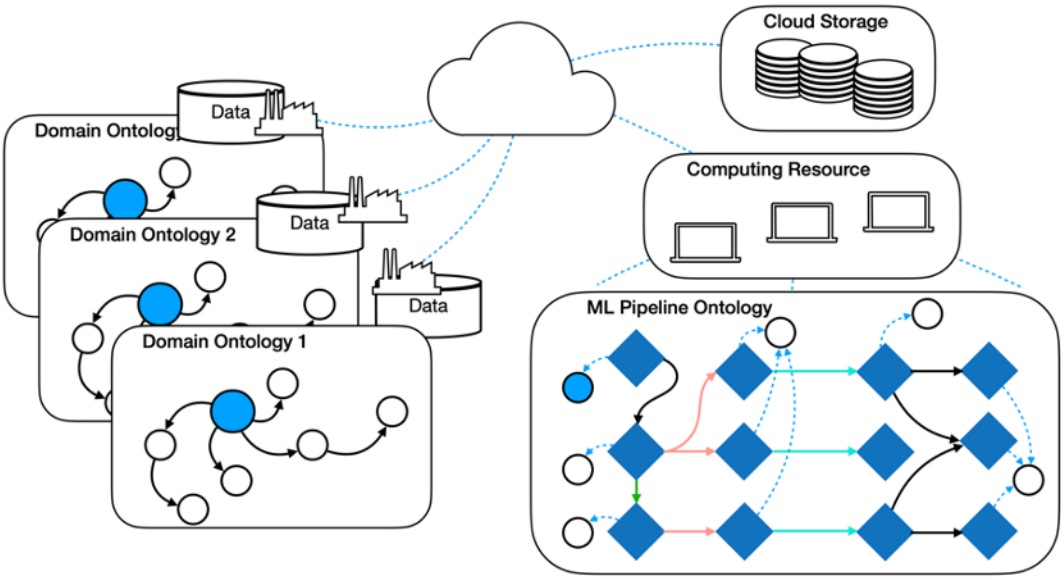

Figure 5.10: Diagram of the concept of Scalable Machine Learning on the Cloud [87].

using a SOA architecture. In SOA client's awaiting requests should be answered in timely manner, the training and test function of models in our system can take a longer response time due to its processing time nature, but we require a consistent and short response time in the other functions since the user has to have a smooth experience while going through component to component. The other point important in SOA that is present in our system is the high data integrity provided by the architecture, it is important that the data should be accurate and reliable to the proper use of the system.

Representational State Transfer (Rest) architecture is not restricted to one specific data format [94], which is highly desirable as a facilitator for future integration of the system with other systems. Rest allows caching which can save resources from the server and it has an ease to be used, saving response time and development time this way. It uses a server-driven architecture, which reflects our Scalable Machine Learning on the Cloud plan. It is also constantly viewed as a standard approach when designing APIs [95]. These factors and the ones presented about the other options put the use of a rest architecture as the most fitting option for Q3.

5.5.2 System architecture design

By the end of our architectural questionnaire diagram we concluded that the development of a web application based on Rest API solution would fit our development constrains, system characteristics and system requirements. With that architectural approach in mind we extended it into a detailed system architecture diagram seen in Figure 5.11.

In the proposed architecture a Dynamic front-end web application will work as the way the final user and admins can interact with the system functionalities and data. This structure will be available for the user as log-in credentials are filled as the user access the system through his browser in the client-side. The intercommunication between the front-end and the API is done via HTTP calls. The communication between the API and the Business logic elements is done by the API as it loads the executable program with the logic and the required arguments for the script. The database is running under a knowledge graph management system and it and can be accessed via rest calls in a port in the server-side machine.

A dynamic way of representing and interacting with the ontologies and the data is possible via this structure, e.g. users can load and edit ML Pipeline ontologies dynamically in the browser using the provided interface, those pipelines are later converted into ontologies in the SemML database. The functionalities of the semantic components are accessed through different tabs of the system, e.g. the use of the OE tab will create a structured message sent to the API that will reflect that use in the database.

The API handler is a program that constantly runs in the company server and requires authentication to be accessed. It is responsible for sending structured HTTP responses to the client-side and receiving HTTP requests from admins and users from the front-end interfaces. The requests are directed accordingly to the end-points of the front-end functionality used in the front-end itself. The API handler is also responsible for parsing the information in the requests, e.g. a change in an ontology in the OE is structured as a form in the front-end, it is parsed as a Terse RDF Triple Language (Turtle) syntax correspondent in the API.

The business logic contains the ML translator responsible for the Data-to-FG mapping via reasoning and the ML Scripts that will receive the arguments from the API.

The system databases are the SemML Database that contains the ontologies used by the semantic components and the Access database that contains information about user‘s access to the system. SemML Database is done via graph database, to avoid re-work on re-structuring the pre-exiting data into another format, and access database via relational database, since they store only management related data.

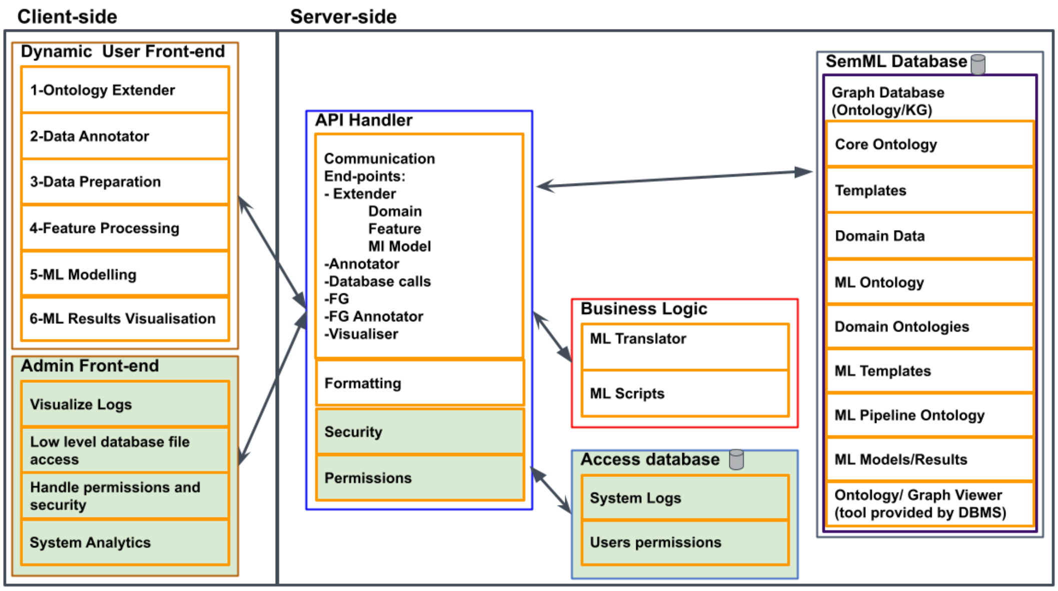

Figure 5.11: System architecture design. Elements with the green background are strongly related to management functionalities of the system. Arrows represent the direction in which system structures intercommunicate.

Chapter 6

System Development

6.1 Overview

This chapter contains the documentation of the development process of our research topic solution. The development process was conducted based on the proposed solution analysis from chapter 5. The sections of this chapter are defined as following:

1. **Development tools**: This section contains the definition of which technological tools can be used to better address the needs of our system. This covers the chosen technologies (frameworks, DBMS and languages) and how they are used in our system architecture.
2. **SemML**: This section define how the semantic components can be divided and implemented as system components.
3. **Quality features**: This section describes how system features were capable of addressing the quality attributes of the system.

6.2 Development tools

With the system architecture design defined from Chapter 5, we decided which technologies (languages, frameworks and DBMS) were going to be used in this thesis system development process. Due to time constrains, we did not conduct a systematic and extensive research on best technological tool alternatives for the development of our system. A diagram with the used technologies for each structure in our system architecture is represented in Figure 6.1.

To make our decision on development tools more selective we decided to give preference for each system structure's technologies the development team was more familiar. If the development team was between different options, the adopted technologies were selected by picking the most popular technology alternative from Stack Overflow 2021 developer survey [96]. Our tool choice for the development of the dynamic front-end was the web framework Angular, which got the 4th position, for most commonly used web framework. The choice for the API was C#

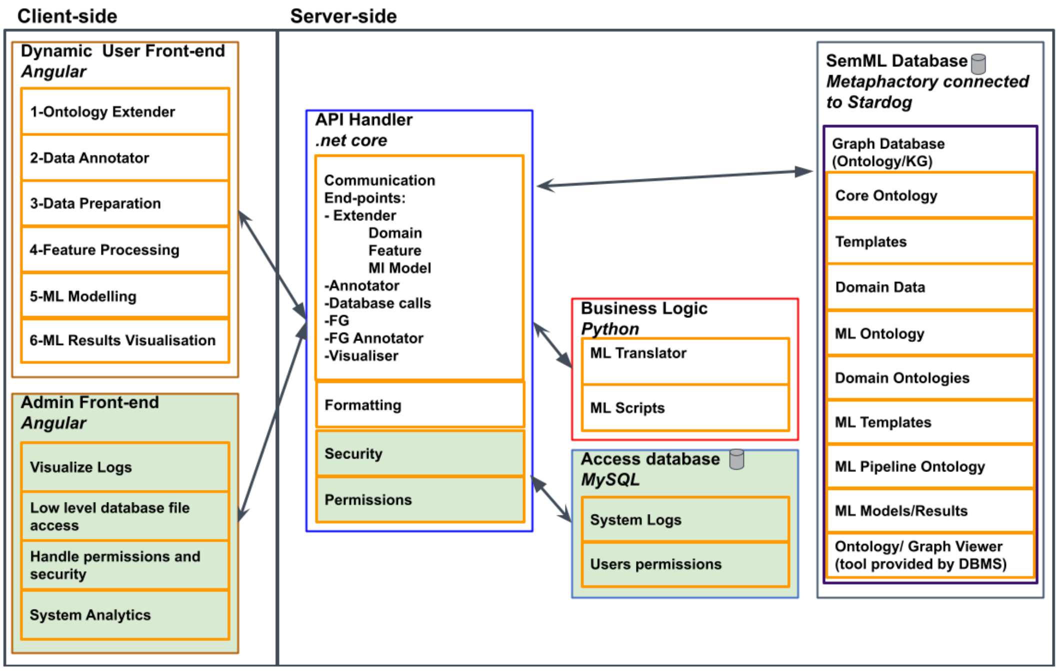

Figure 6.1: System architecture design with the development tools used or planed to be used in each structure of SemML.

with .net core framework, in the programming, scripting, and markup languages more used, C# was in the 8th position. The business logic counted on python for development which was the 3rd position as the programming, scripting, and markup languages more used.

The access database, responsible for administrative data of the system, was planned to use MySQL. SemML database was initially built using Web Ontology Language (OWL), a semantic markup language for publishing and sharing ontologies [97]. We used terse RDF Triple Language (Turtle) syntax to format the data [98]. The data was stored in Stardog software, a commercial RDF database [99]. In last versions of the system the access to SemML ontologies was made via direct access through Stardog. The data access in the database was transfered to a middle ware service from Metaphactory tool [100] during this new version of the system. This tool allows visualization and analytics of the graphs on the browser, REST APIs requests to access data in the middle ware, it deals with data from different data sources in the data access infrastructure, among other features. Using Metaphactory we keep the access to the database via REST API requests and we can use the extra features provided by the tool, a schematic of this tool architecture can be seen in Figure 6.2.

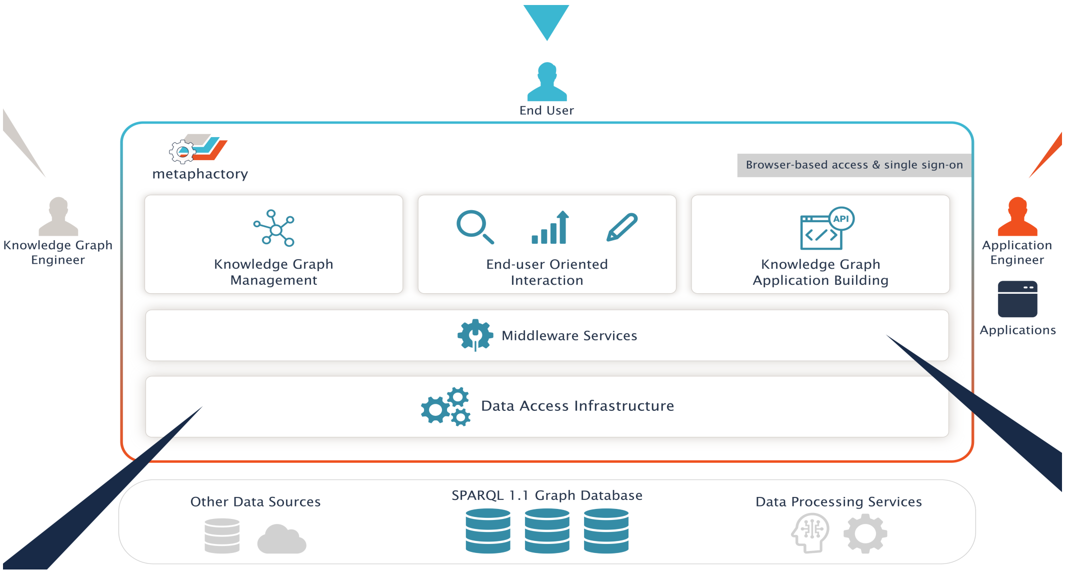

Figure 6.2: Metaphactory architecture product. The services provided by the system can be accessed via REST API requests in the middle ware and the data stored in Stardog is accessed via the data access infrastructure [100].

6.3 SemML

SemML system idea allows users to conduct the process of a ML development for industrial data-driven condition monitoring from the workflow represented in Figure 4.1. The steps 2-5 are conducted with the semantic components presented in Chapter 5 and the architectural overview of the solution is represented in Figure 5.1. The semantic component OE from step 1 and the DKA Raw-to-DO feature from step 2 were developed and evaluated in last published papers [20]. The last version of the system was tested on "Static Mode" for ML construction, a mode where the user only select pre-set pipelines without dynamically changing them and the visualisation.

In this new development version of the system we developed a Dynamic front-end for the user where the functions work based on JSON objects and are sent to the API endpoints we also developed. These objects have to be formatted accordingly in the API to interact with the SemML Database, which would be HTTP requests with SPARQL commands in them for Metaphactory, or the Business logic structure, which would be a set of arguments for the python program running.

Due to short development time and the complexity involved in system we did not develop the formatting component of the API and its communication with the SemML Database and Business logic. This way, the system is operating in a "object mode", in which the front-end can behave and dynamically change the JSON objects as it would be in the real world use of the final version of the system, but the objects are not really formatted and processed for the Business logic or interact with the SemML Database. Similar to how last versions of the system were evaluated in a "Static Mode" but kept part of the process as it would behave in the real world, this version of the system aims to evaluate the dynamic front-end structures of the system as they would behave in the final version of the solution.

The dynamic front-end was divided into tabs that deals with the semantic components processes. To better demonstrate each front-end tab in the system we are going to present each tab's demo system use and interface. The demo system use is a proposed order of tasks executed in the tabs that covers all the main functionalities of the system and reflect an example of a real world use of the system by the users, it is used to illustrate the functionalities and it was also used in the evaluation process in Chapter 7. The interface is our GUI and how the user interacts with it to proceed with the tasks from the demo system use.

The tabs division, the steps, the semantic components and tasks from the demo system use for each tab are presented in Figure 6.3 and further discussed in this chapter.

System last version GUI		Front-end Tabs				
1- Ontology extender	2- Data Mapping	3- Data Preparation	4- Feature Processing	5- Modeling	6- Visualisation	**Name**
Step 2	Step 3	Step 3	Step 4		Step 5	**Workflow Step**
Ontology extender	Domain knowledge annotator	Domain knowledge annotator	Semantic enhanced ML		Ontology Interpreter	**Semantic Component**
		3.1 - 3.6	**4.1- 4.7** **5.1- 5.4**		**6.1 - 6.4**	**Demo system use tasks**

Figure 6.3: Front-end solution diagram. Grey elements refer to the elements developed and covered by this thesis. The rows Name, Workflow step, Semantic component and Demo system use tasks represent respectively, the name of the page/tab, the step from Figure 4.1 covered by the tab, the semantic component addressed by the tab and the number identifiers of the demo use tasks.

6.3.1 Data preparation

First, the OE and Raw-to-DO mapping process is conducted in last evaluated versions of the system. After that, the front-end tab Data Preparation is accessed and allows the user to load the raw data, load the DO and automatically or manually annotate the DO to feature groups. All the process conducted in this tab provides a visualisation through tables that facilitate the comprehension of the process.

Demo system use

The demo use of the system tasks conducted in Data preparation tab are the following:

- **3.1**: The user accesses the data preparation tab which will be empty.

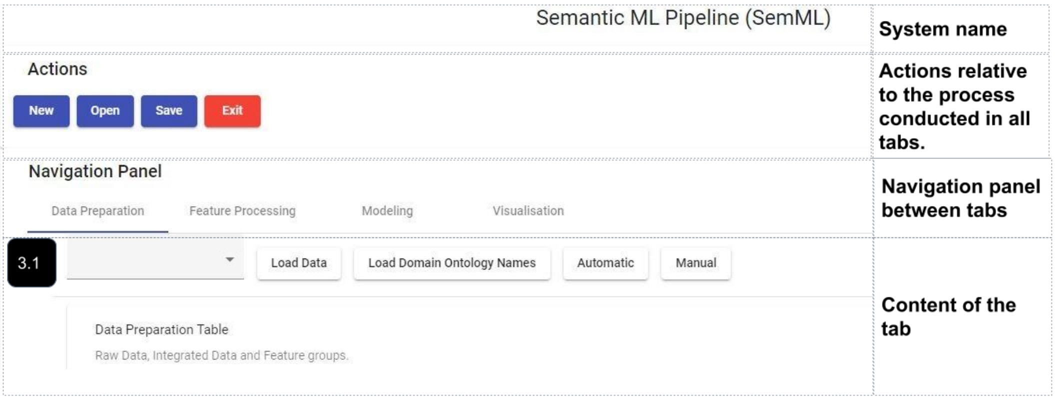

Figure 6.4: Data preparation tab GUI in initial state relative to task 3.1. Other elements of the general use of the system such as system name and actions relative to the process are presented.

- **3.2**: The user select and load the single features raw variables from local machine.
- **3.3**: The user select and load the time series raw variables from local machine.
- **3.4**: The user will select and Load the DO and map the raw feature names with DO names.
- **3.5**: The user will automatically assign the feature groups to the DO names.
- **3.6**: The user will manually change the feature group of one DO name to a different one and will change it back to the original feature group. This task is conducted only to demonstrate the possibility and the process of manually editing the feature groups.

Interface

In Figure6.4 it is seen elements in the system that are omitted in future GUI pictures due to its lack of relevance or because they are not implemented. These elements are the System name which in the future might be followed with a logo as well and the actions relative to the process in all tabs which will in the future be implemented to save the progress in the tabs while using the system. The figure also shows the initial state of the tab when opened, relative to the task 3.1.

Figure 6.5 represents the GUI as it would be after the tasks 3.2 and 3.3 were conducted. To load the raw data of each feature group the user selects the drop-down list on the left of the Load Data button and clicks the button. A navigation screen will show up in which the user can select the folder and file in the local machine that contains the raw variables. After the selection in the local machine is done the system will load them and the number of rows in the data preparation table will be updated along with the elements in the Raw Variables Names column in the table.

Figure 6.6 represents the GUI as it would be after the tasks 3.4 and 3.5 were

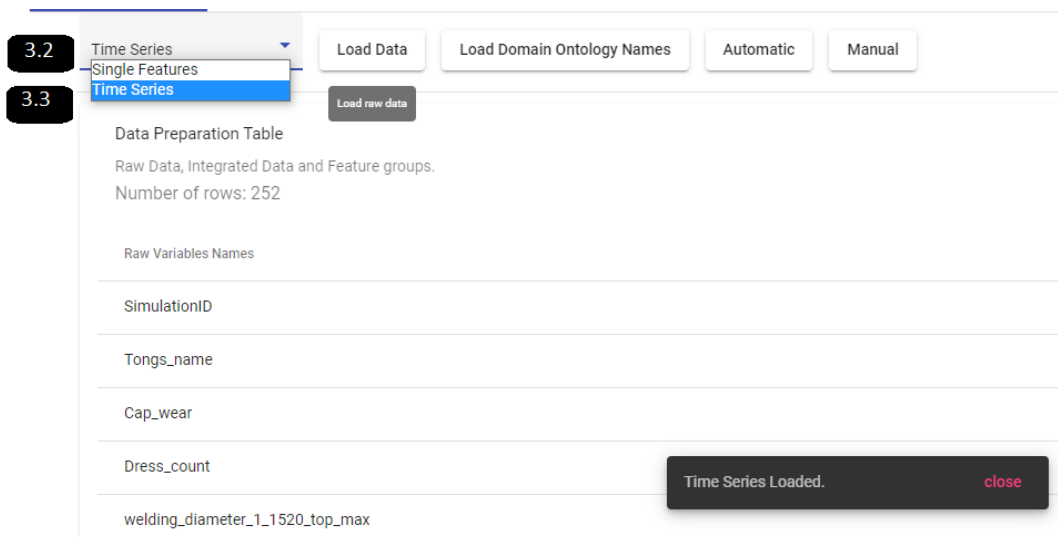

Figure 6.5: Data preparation tab GUI after raw Single Features and raw Time Series variables were selected and loaded from the local machine, relative to the tasks 3.2 and 3.3.

conducted. In task 3.4 the user will click in the Load Domain Ontology Names button and load the Domain Ontology saved in the local machine created previously at the OE. After the DO is loaded the new column Domain Feature Names is filled in the table, the elements in this column are in the same row as the annotated raw variables in the Raw Variables Names column due to the Domain knowledge annotation process conducted in the data mapping. In task 3.5 the user will click in automatic button to automatically do the Data-to-FG mapping and a new column Feature Groups is filled with the feature groups correspondent to the DO names and raw variable names.

Figure 6.7 represents the GUI for the task 3.6. The user can click in the manual button and a column with a button relative to each row is filled. When clicking to edit a feature group of a column a pop-up dialog box is created in which the user can select the feature group to annotate the DO name among a drop-down list of possible options.

6.3.2 Feature processing

After the raw data is mapped into DO terms, it is then mapped into the feature group elements, quality indicator, single features, and process curves (time series) via reasoning. All variables of each feature group are worked in conjunction, where an algorithm applied to one feature group processes all variables of that group.

The ML pipeline in this tab is made out of layers with inputs, algorithms outputs, and data processing algorithms. The inputs are composed of feature groups or outputs of previous layers algorithms. The feature processing algorithms follow

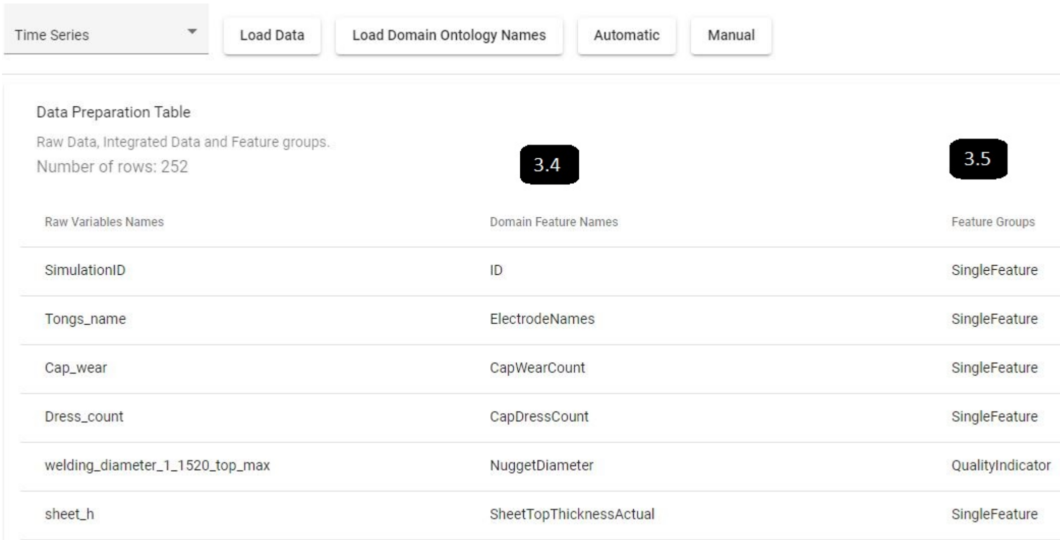

Figure 6.6: Data preparation tab GUI after DO is selected and loaded and the feature groups are automatically assigned to the DO names, relative to the tasks 3.4 and 3.5.

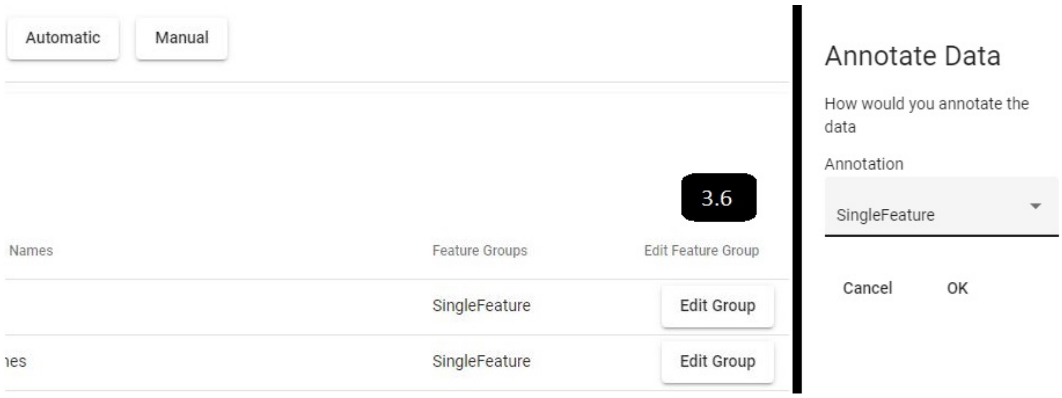

Figure 6.7: Data preparation tab GUI for manually editing the feature groups column, relative to the task 3.6. The left part shows the Edit Feature Group column with the edit group button relative to each row. The right part shows the pop-up window that allows users to proceed with the manual editing.

the restrictions imposed by the ML ontology and ML template; e.g. some feature processing methods can only be applied on time series feature groups and not in single features. This pipeline processes the data so it can be later used in the modeling process.

Demo system use

The demo use of the system tasks conducted in Feature processing tab are the following:

- **4.1**: The user accesses the data preparation tab which will contain the single features, time series, and quality indicator loaded from the data preparation tab.
- **4.2**: The user will click on load pipeline button and access the load pipeline dialog box. After visualising the available pipelines in the dialog box, the user will close the dialog box without loading any pipeline.
- **4.3**: The user will click on the add layer button and include and configure the layers in the tab to mirror the example pipeline presented to them.
- **4.4**: After the pipeline is structured in the tab as the demo pipeline, the user will add an algorithm in any layer and another algorithm in any next layer. The second algorithm input should be the output of the first one. Then, the user will delete the first algorithm, causing a input dependency problem.
- **4.5**: After the input dependency message created by task 4.4 appears the user is asked to fix it by removing the algorithm with the dependency issue. Tasks 4.4 and 4.5 serve the purpose of demonstrating how the system respects and handles the restrictions from ML ontology and ML templates.
- **4.6**: The user should save the pipeline locally as a ttl file to demonstrate how ML pipeline ontologies can be saved.
- **4.7**: The quality indicator from the feature prepared layer and the outcome of the feature processing pipeline are marked to be used in the modeling process.

Interface

The left part of Figure 6.8 represents how the feature processing tab will be displaed when first accessed, there will be a feature prepared layer column with the feature groups elements from the data preparation tab, this represent the demo task 4.1. The load pipeline button on the left opens the dialog box on the right that represents where the user can load already saved pipeline ontologies from an online catalog, or from the local machine. After opening the load pipeline dialog box only to demonstrate the feature, the user is asked to click in cancel, which represents demo task 4.2.

In Figure 6.8 the button Add layer is used to create new layers of the ML pipeline. The structure example of a feature processing algorithm is represented in Figure 6.9. Each layer can have multiple algorithms that are going to be processed

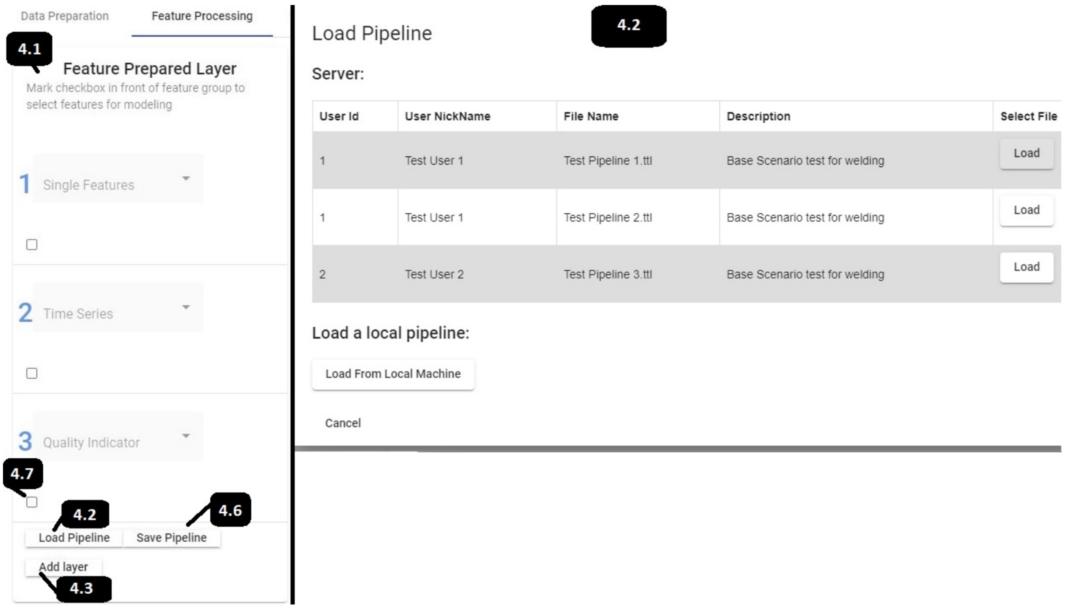

Figure 6.8: Feature processing tab GUI (left) accessed after the data preparation tasks were conducted, it includes the buttons and check boxes from demo tasks 4.2, 4.3, 4.6 and 4.7. Load pipeline dialog box (right) rendered after Load Pipeline button is clicked.

in parallel. The layers are processed sequentially and the input can be the feature prepared layer elements or the output of an algorithm from a previous layer, in the last case it is possible to have a input dependency issue if the output from a previous layer used as input for another layer is deleted.

While operating the ML pipeline, the list of possible algorithms respects the ML ontology and ML template ontologies restrictions, e.g. only time series related algorithms can be selected for inputs of group time series. Adding and configuring the layers to mirror the demo pipeline, further presented in Chapter 7, represents the demo task 4.3.

The left part of Figure 6.10 (left) represents how the system will behave if a dependency issue situation happens, similar to the one caused by tasks 4.4 and 4.5. In case of a dependency issue the input is deleted and a message informing the issue appears. After clicking on the Save Pipeline button from Figure 6.8 (left), the dialog box for saving the ML pipeline from Figure 6.10 (right) will appear and the user can save the pipeline locally as a ttl file, representing task 4.6.

Task 4.7 is conducted by marking the checkbox from the quality indicator in feature prepared layer, as represented in Figure 6.8. The task also requires marking the output of the last algorithm from the last layer, this checkbox is similar to the one represented by letter E in Figure 6.9. With both check boxes marked, the output of the ML pipeline along with the quality indicator are sent to the modeling tab.

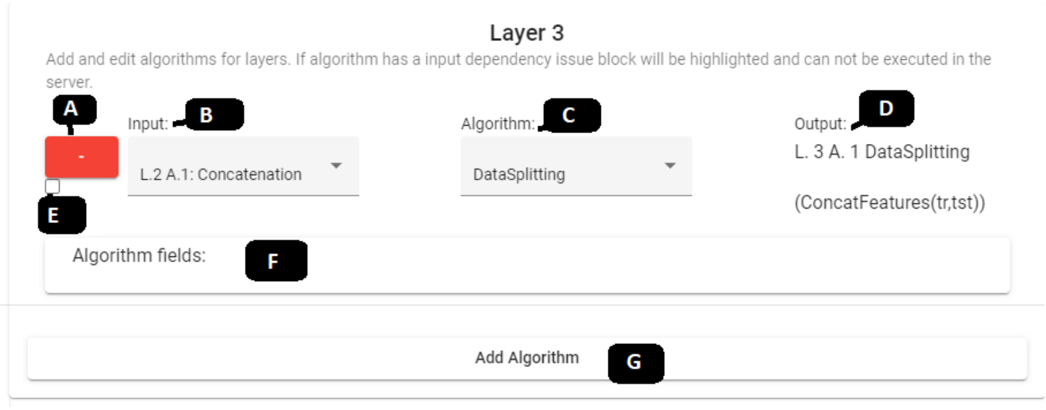

Figure 6.9: Structure example of a feature processing algorithm. A deletes the algorithm. B has the drop-down list with possible inputs. C has the drop-down list of possible algorithms. D is the output name for the selected algorithm. E is the checkbox that sends features to the modeling pipeline. F is an expandable field with parameters for the algorithms. G adds another algorithm for the layer.

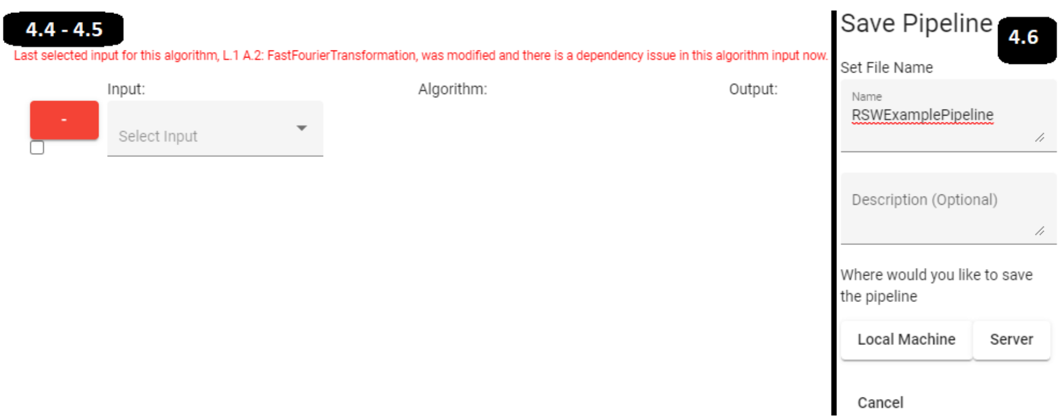

Figure 6.10: Error message example in algorithm when input dependency error occurs (left). Dialog box when saving the ML pipeline locally or in the server (right).

6.3.3 Modeling

The data that will be used as input or output of the predictive model are selected by the check boxes in feature processing tab. The modeling tab is responsible for configuring, training and testing the predictive model. The tab simulates the final version of the system as it would behave, the tab creates the object that will be sent to the API in order to be processed in the business logic layer.

Demo system use

The demo use of the system tasks conducted in Modeling tab are the following:

- **5.1**: The user accesses the modeling tab.
- **5.2**: The user selects the algorithm output from the last layer from the demo pipeline, algorithm Feature selection from layer 5.
- **5.3**: The user adds a modeling layer and selects linear regression without changing the parameters as the modeling algorithm.
- **5.4**: The user trains the model and tests it.

Interface

Figure 6.11 (left) contains the preparation of the modeling process, the part where the features are loaded and attributed to inputs or outputs in the model. The user will first access the tab represented by the task 5.1. Task 5.2 is conducted as the user selects the feature outcome from our ML pipeline as input feature and selects quality indicator feature as the output feature of our predictive model.

After the features are attributed in our modeling features layer, the user will conduct the tasks directly related to the model algorithm in Figure 6.11 (right). In task 5.3 the user adds a layer to our modeling pipeline, that layer contains the linear regression algorithm and the parameters as default. After the input and output were selected from task 5.2 and the modeling layers were structured from task 5.3, the user can conduct the training and testing of the model represented in task 5.4.

6.3.4 Visualisation

After the model is trained and tested in the modeling tab, the user can have an example of how the system will behave in the visualisation tab. The system will behave as it would in the final version but the presented scatter plot has the illustrative purpose only and comes from a simulation data set.

Demo system use

The demo use of the system tasks conducted in Modeling tab are the following:

- **6.1**: The user accesses the visualisation tab.
- **6.2**: The user sets the estimated and target Q-values to each axis.

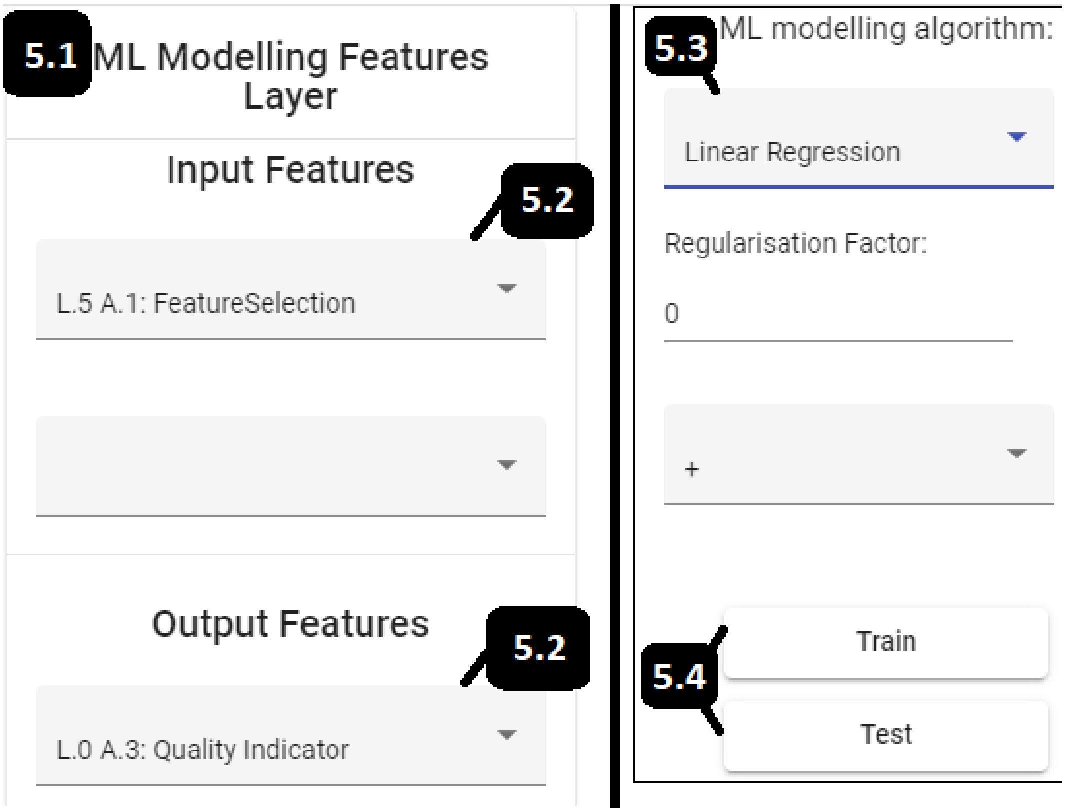

Figure 6.11: Modeling features layer with dynamic selection of input and output features for the model (left). GUI for editing, training and testing of the modeling pipeline (right).

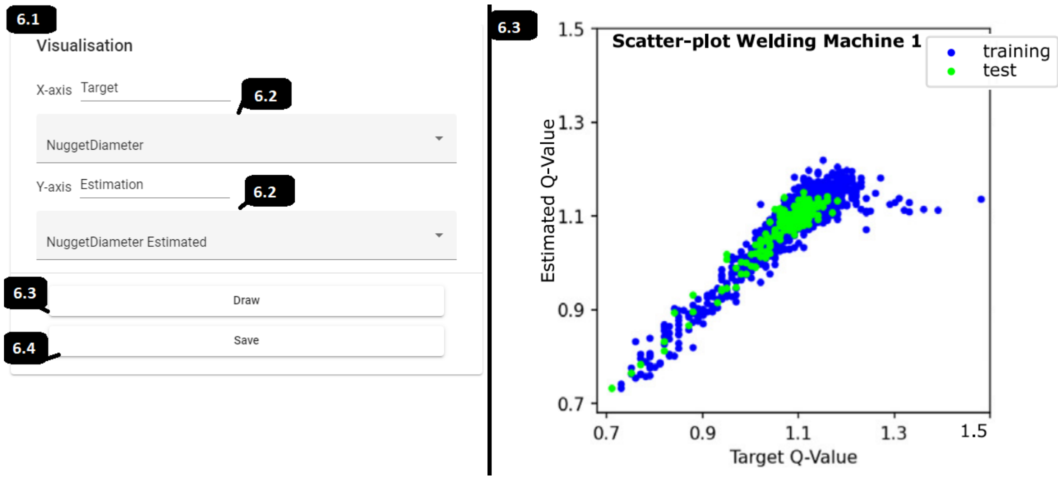

Figure 6.12: Visualisation tab GUI (left). Displayed image of the train and test values on estimated and target Q-values from a demo predictive model (right).

- **6.3**: The user plots the scatter graph of the train and test values from the model.
- **6.4**: The user will save the image of the scatter graph locally.

Interface

Figure 6.12 (left) contains the GUI of the visualisation tab from task 6.1. The axis are set by the user from a drop-down list where estimated and target quality indicator can be selected. In task 6.3 the user clicks on Draw button to plot the visualisation of the example model result, Figure 6.12 (right). Save button will allow users to save the plotted image in the local machine, as seen in task 6.4.

6.4 Quality features

To conclude the development process of our solution the assessment of the quality attributes of our system are presented in this section.

In Chapter 4 we proposed and explained the quality attributes of our system that relate to the desirable performance of the system [65]. The link between the quality attributes and the quality features responsible for assessing each quality attribute is presented in the table 6.1. Our system features that are responsible for assessing the quality attributes are labeled as quality features (QF) and are described as following:

- ***QF1***: *Domain processes and ML processes based on ontologies* is the concept in which the system is built on. This feature allows the system to be fully customized and evolve according to the needs, as long as the process is described using well designed ontologies. The domain and ML processes described as ontologies allows experts from different areas to understand and

collaborate in the system in a easier way. This feature mainly increases the ontology based system customizability, evolvability, accessibility and collaboration, representing QA2, QA4, QA6 and QA9.
The description of processes as ontologies is done by the OE. The development of ML processes ontologies is conducted in the feature processing and modeling tabs, conducted in demo tasks 4.3 and 5.3.

- ***QF2***: *Generalisability of ontologies* allows the effort used in creating an ontology for a specific scenario to be used or adapted to new ones. The goal is to ensure more efficiency to the system by reducing the effort in the tasks, this strongly relates to QA1, QA2, QA3.
The generalisability of ontologies can be seen in the OE where DO are created based on a generalisable core ontology. It is seen in data preparation demo tasks 3.4 and 3.6 where a DO can be loaded for a new data and the FG of that DO can be edited. It can also be seen while loading and editing ML pipelines ontologies in the feature processing tab by the demo taks 4.2 and 4.3.
- ***QF3***: *Test and visualisation of predictive models* guarantees the effectiveness and satisfaction of the system by providing and evaluating the predictive solution for the scenario. This is achieved as the predictive model can be trained, tested, visualized and the result of the model can be further evaluated by experts from different domains. This model can provide knowledge that can allow a proactive maintenance of the process machinery like a prediction on less quality that could occur due to dressing and would requires changes on the electrode cap. These aspects are strongly related to QA1, QA4, QA6 and QA7.
The test of predictive models can be seen in the tab Modeling by the demo tasks 5.4 where the constructed model is trained and tested. The visualisation of the predictive models can be seen in tab Visualisation represented by the task 6.3 and the adoption of the predictive model can be further evaluated.
- ***QF4***: *Custom result visualisation* allows users to customize the axes of the train and test values achieved with the predictive method on the input data-set. This customization allows users from different backgrounds to communicate and discuss the model results. This is strongly related to QA2 and QA4.
The customization of train and test results of predictive models over the input data-set can be seen in tab visualisation by the demo task 6.2.
- ***QF5***: *Automatic assign feature groups to DO names* mitigates or completely removes the manual process of mapping the raw data collected by different sources from different machinery to domain names and feature groups so they can be used in the ML pipeline. This feature is strongly related to QA3, QA8, QA9 and QA11.
The preparation for the automatic mapping is conducted in the Data Mapping where raw variable names are mapped to the terms in the DO. The

Quality attribute	Quality Feature
QA1	QF2, QF3
QA2	QF1, QF2, QF4, QF7
QA3	QF2, QF5
QA4	QF1, QF3, QF4
QA5	QF6
QA6	QF1, QF3
QA7	QF3, QF7
QA8	QF5
QA9	QF1, QF5
QA10	QF6
QA11	QF5, QF6, QF7

Table 6.1: Table containing the quality attributes of the system and the quality features responsible for assessing those quality attributes.

automatic mapping process is conducted and visualised in data preparation tab, it can be seen in the demo task 3.4 and 3.5.

- ***QF6***: *Web API based system* is the feature in our system that allows the portability of SemML as it can run in mobile device's browsers, computer browsers and even smart TV browsers. The front-end element also counts on a REST API back-end, which is based on widely used HTTP methods that allow the process of integrating with another system simpler as it also allows other system to integrate with SemML system simpler. The adoption of a rest API approach also allows the front-end functionalities to run on users computers independent of the number of users and it also allows the scalability of the back-end by allowing the insertion of new servers as the demand grows. These aspects are represented by QA5, QA10 and QA11
- ***QF7***: *Dynamic front-end* provides a satisfactory quick response to the actions that are processed on the browser and not in the servers. This response requires no back-end communication and would provide a direct and reliable response to the user. To proactively ensure that the actions of the users will follow a determined structure that will not compromise the system the front-end respects templates that restrict and correct the user inputs, e.g. methods for single features cannot be used for time series in the feature processing tab. These attributes are represented in QA1, QA7 and QA11.
  The Dynamic front-end and the templates restrictions can be seen in the whole system being highlighted in the OE and data mapping input fields, the editable table from data preparation tab, the editable ML pipeline creation in feature processing tab and the editable model pipeline in the modeling tab.

Part IV

Evaluation

Chapter 7

System Evaluation

7.1 Overview

This chapter aims to present the evaluation process of our solution along with the results found. The diagram with each step of the evaluation process is presented in Figure 7.1. Step 1 is discussed in section 7.2, step 2-4 in section 7.3 and step 5 in sections 7.4 and 7.5.

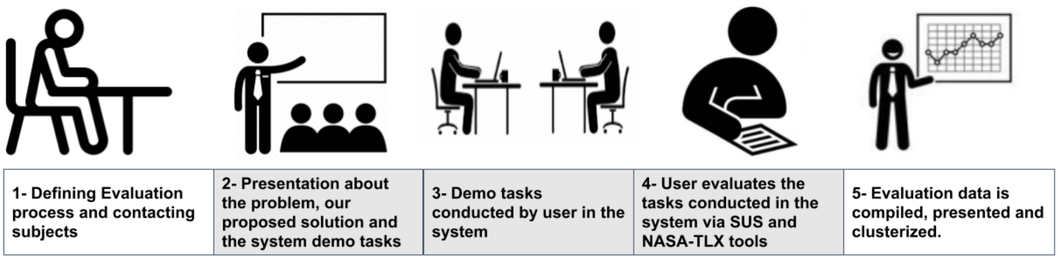

Figure 7.1: Diagram of the system evaluation process conducted. Steps are ordered. Gray steps have direct interaction with subjects and white ones have no direct interaction.

7.2 Preparation

7.2.1 Step 1

Subjects

To prepare our system evaluation process first we defined a criteria for the subjects. The subjects submitted to our evaluation process had to reflect real world final users of the system. The system is built to handle initially manufacturing condition monitoring as presented in our use case but it can be used in different scenarios. It is expected that the user are professionals from multiple field, since all kind of processes that can be structured as ontologies and could benefit from ML models for condition monitoring could be potential future users of the system.

The system can be extended to scenarios outside of manufacturing; e.g. a law firm that builds a system capable of measuring the lawsuits win rate of their lawyers by using various variables. It is possible that if the process is well designed as an ontology, the system can provide a predictive model that presents some variables as a big impact on the success rate. In this case the law firms could assign the law suits to specific attorney or do not accept the case.

The subjects used in our evaluation process had to have at least a bachelor level of education from any field. There was a total of 25 subjects, 1 from physics, 1 from public relations, 9 from computer science, 6 from medicine, 3 from law, 1 from literature, 1 from fashion design, 1 from manufacturing engineering, 1 from mechanical engineer, 1 from civil engineering.

Evaluation tools

In Chapter 4 we discussed what we aim to evaluate in this thesis, which evaluation tools alternatives we were going to adopt and why they were adopted. In our case, we want to evaluate how our proposed solution, SemML current system version running in "object mode", performs as a system solution when evaluated by the System Usability Scale (SUS) and the NASA Task Load Index (NASA-TLX).

Besides the evaluation tools some quantitative and qualitative variables are going to be collected while the subjects do the tasks and after the tasks are concluded. The collected variables were defined as exceeded information from the evaluation tools that could provide significant insights to further development steps of the system, like a general and open feedback about the use of the system, or variables could indicate possible points where improvement is desirable, like measuring the longer time taken by the user in tabs that were meant to behave quicker.

7.3 User evaluation process

7.3.1 Step 2

Scenario and data

The ideal demonstration of the system for the evaluation would require the data and demo scenario to fit a process of each subject specialisation. This would require an extensive analysis and adaptation of the data and scenario to fit each of the 10 subjects specialisation field, which is out of the scope of this project and would not be feasible in our time window.

We did opt for the RSW process in our system demo use since the data, ontologies, results visualisation and mapping is already structured from last paper [20] and the RWS process can be explained with relative ease to professionals from different fields. To make the users from different fields familiar with the RWS process and the predictive problem a presentation is conducted before the user have a contact with the system. The RWS process and its quality monitoring process can take as much time as necessary for the user to understand the topic, in our subjects sample this process took between 2 to 8 minutes.

Demo tasks

When the user is familiar with the RWS condition monitoring process and our motivation we proceed to present our system solution idea and the concepts behind it. This presentation involves explaining more technical concepts such as feature processing and ontologies.

The solution workflow diagram, from Figure 4.1, is presented and explained while the demo system tasks are explained an linked to the workflow steps.

Regarding more specific elements from our RWS in our WM1 demo use of the system, the presentation gives a brief explanation about the main variables involved in the process, such as the one that represents the nugget diameter, which is the quality indicator of this welding process. It is also presented the ML pipeline desirable to be replicated in tabs feature processing and modeling. This presentation took between 5 to 12 minutes in our subjects sample.

7.3.2 Step 3

System use

With the system demo task, the motivation and the RWS condition monitoring process clear for the user, the subject gets access to SemML system running remotely in the developer's computer. The user is asked to perform the set of demo tasks and can freely consult the presentation material to clarify any concept or check the tasks. If the user could not perform the task the task would be set as failed and the user would be asked to move to the next one, this did not happen with our subjects sample.

Despite being able to check the presentation material the developer cannot provide any further information to the subject since this could affect the way the subject would use the system under normal conditions. While the subject is using the system the developer measures the time consumed by the subject in each tab of the system.

The system runs in "object mode" and the demo tasks represent a sequential set of tasks that would cover all functionalities of the system front-end in the expected final version of the system, these elements were in depth explained in Chapter 6. The data set used in the demo is a slight adaptations of the RSW simulation scenario data from Zhou et al. work [45]. The RSW simulation scenario data was used to illustrate the use of the system under similar to normal circumstances.

The ML pipeline in Figure 7.2 contains a diagram of how the feature processing tab layers and the modeling tab layers should be structured in their pipelines by the subjects while conducting the demo tasks. In layer 1 of feature processing the time series features should be converted into single valued features to be concatenated with the single features, this is not possible due to the different representation of the features. To solve this, the GetTS Stats algorithm is used to extract 8 single-valued features out of the time series features exemplified in Figure7.3.

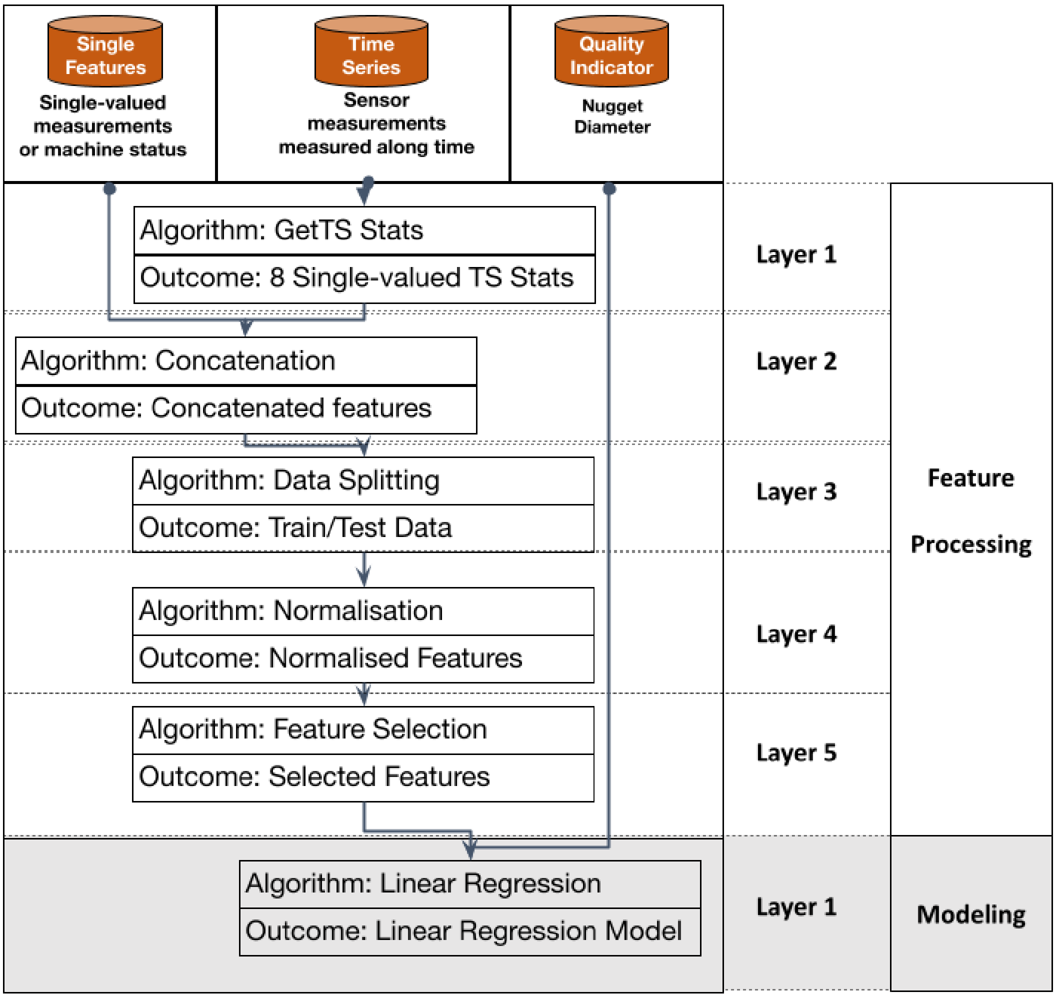

Figure 7.2: Diagram of the WM1 demo feature processing and modeling pipeline for the RSW demo process. The variables divided into their types are represented in orange. The arrows represent the data or outcome of an algorithm used in another algorithm.

In the final part of the use of the system the subject plots an prediction results image, similar to the one presented in Figure 5.6. This image is plotted for demonstration purpose of the visualisation feature, the prediction results reflects the real performance of the predictive model just created by the user that followed the demo tasks in the presented data set.

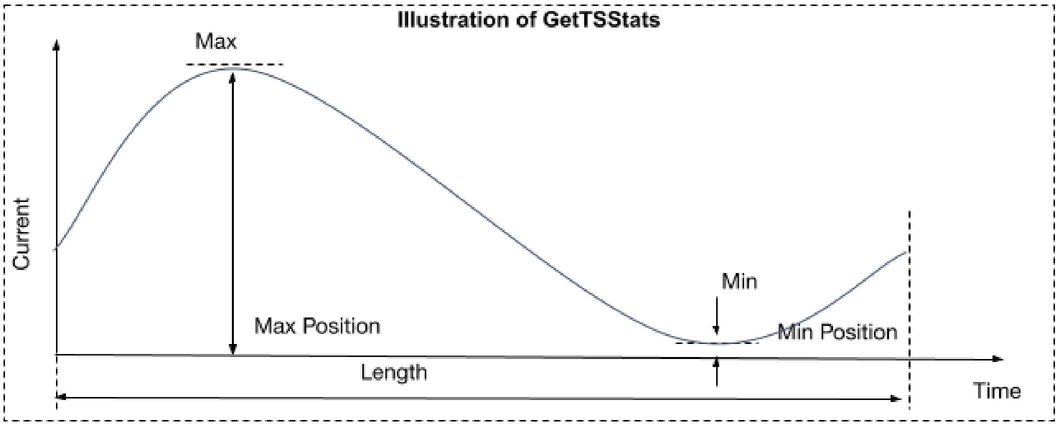

Figure 7.3: Visual representation of GetTSStats extracted single-valued variables on example current over time curve. The single-valued features extracted are the maximum and minimum registered values and the time they occurred, the total time duration of the measuring process, and the statistical values of mean, median and standard deviation.

7.3.3 Step 4

Evaluation process

In step 4 the subject is asked to conduct the NASA-TLX evaluation, the SUS evaluation and the custom questionnaire about the user experience of the system.

Evaluation tools

After all demo tasks were conducted by the subject, their workloads are evaluated in conjunction via NASA-TLX paper and pencil package [101]. First the user is introduced to the scales definitions and he conducts the weighting and rating process with a simpler familiarization task, lifting a chair in our case. After the user is familiarized with the process he is asked to evaluate task of going through SemML system.

The SUS evaluation is done on the set of demo tasks conducted in the system. Any clarification about the questions is provided if requested by the user.

Questionnaire

A custom questionnaire is conducted where the subject is asked to put in a scale from 0 to 10 how familiar he is with machine learning in general, it is also asked to put in the scale how applicable he believes the system can be in his expertise domain. The questionnaire also asks for an open feedback about the system.

Domain specialisation	**G1**	**G2**	**G3**
Developer - Computer Science	6	0	0
Data scientist - Computer Science	3	0	0
Production engineer - Production engineering	1	0	0
Astrophysicist - Physics	1	0	0
Engineer - Civil Engineering	1	0	0
Mechanical engineer - Mechanical Engineering	1	0	0
Marketing assistant - Public Relations	0	1	0
Clothes design - Fashion Designer	0	1	0
Court assistent - Law	0	2	0
Literature teacher - Letters	0	1	0
Judge - Law	0	1	0
Doctor - Medicine	0	0	4
Psychiatrist - Medicine	0	0	2
Sum of subjects	**13**	**6**	**6**

Table 7.1: Table of subjects domain specialisations by groups. The first column contains the name of the position the subject works on followed by their bachelor title. G1 relates to the field of Math and Science, G2 relates to the field of Humanities, and G3 relates to the field of Biology.

7.4 Evaluation results

7.4.1 Step 5

Overview

In this step the collected results of the evaluation process are organized and plotted in order to be further analysed in Chapter 8.

To organize our results the subjects were clustered into three groups according to their domain specialization. This division was made by asking the participants which of the proposed fields suits better his specialization domain. If the user did feel he could be better represented by another group we would include it as well, no new field inclusion occurred. The table with the grouping of the subjects can be seen in Tab 7.1.

With the subjects grouped we proceeded to present our evaluation results of the NASA-TLX tool, SUS and the time results.

NASA-TLX

Each workload factor's minimum, maximum and average weights returned in our subjects sample is divided by groups and presented in Figure 7.4. After the weighting process is conducted, the overall NASA-TLX minimum, maximum and average workload scores from our sample are grouped and presented in Figure 7.5.

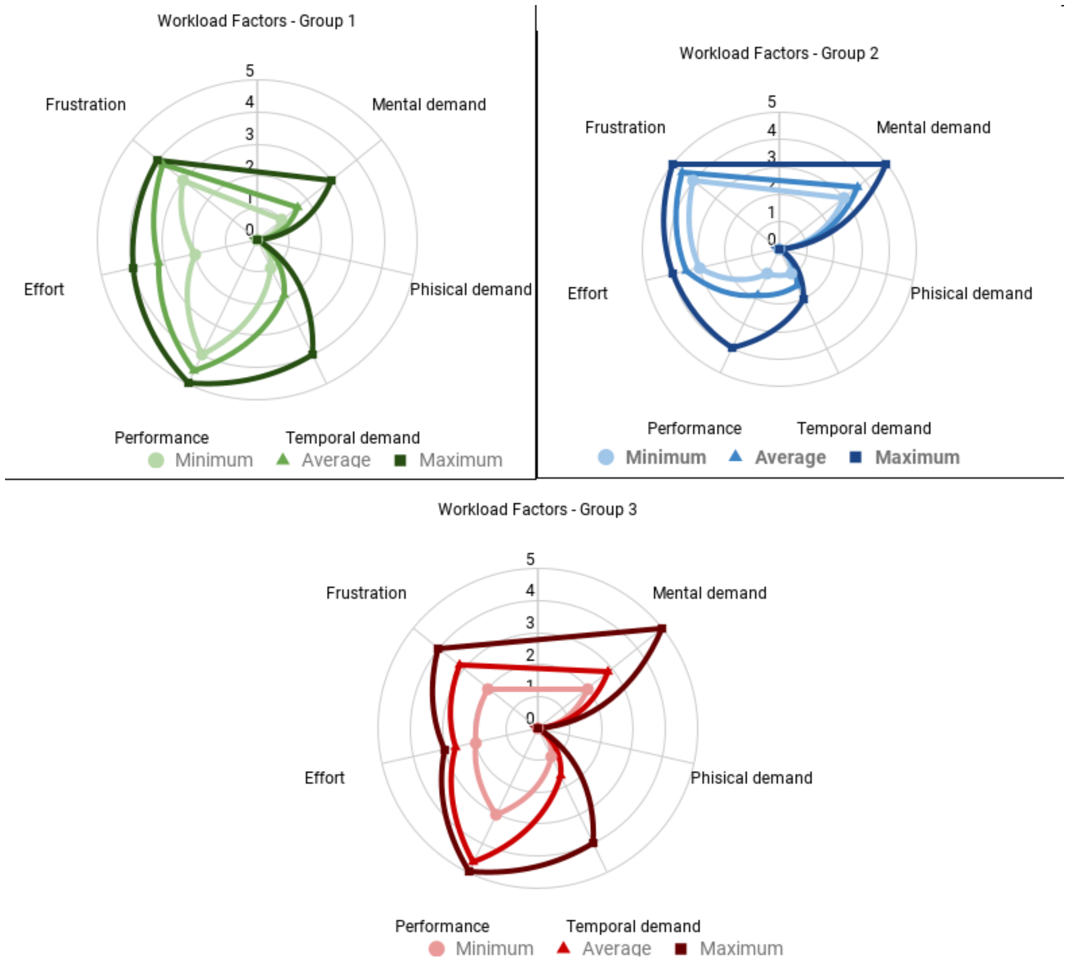

Figure 7.4: Radar chart with minimum, maximum and average values of each NASA-TLX workload factor divided by subjects groups.

SUS

The SUS minimum, maximum and average scores registered in our sample clustered by groups can be seen in Figure 7.6.

Time

The time is an approximation by how much time each subject consumed in step 2, the presentation step, and in step 3, the conduction of demo tasks. The time is clustered by groups and the time used in step 3 is divided by tab. The minimal, maximal and average time consumed by each group is presented in the chart from Figure 7.7. The overall average of times collected for presentation, data preparation, feature processing, modeling, and visualisation are respectively 9 minutes and 36 seconds, 4 minutes and 10 seconds, 11 minutes and 1 second, 1 minute

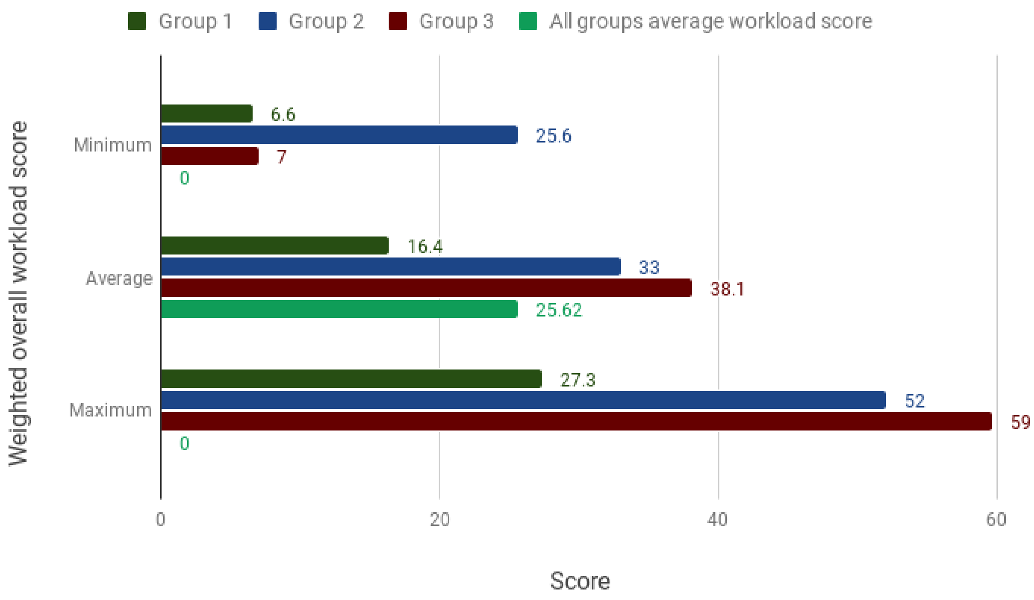

Figure 7.5: NASA-TLX weighted overall workload score bar chart. The minimum, maximum and average of these scores found in our sample are clustered by group. The average weighted score of all samples without the division by group is also presented.

and 17 seconds, and 1 minute and 23 seconds respectively.

ML familiarity

It was asked for the subjects to rate their experience with ML technologies in a scale from one to five, one representing no prior knowledge about ML technologies and five representing high familiarity with ML technologies . The results divided by group are presented in Figure 7.8

7.5 Evaluation discussion

7.5.1 NASA-TLX

The NASA-TLX process aims to evaluate the amount of workload involved in the use of the system while conducting day-to-day use tasks. The reduction of the workload is directly linked to the reduction of how the system is inclined to human errors [73]. Reducing the human errors is extremely relevant in our system since errors could be expensive and dangerous in the industrial environments it is mainly built for.

NASA-TLX can be divided into two main processes, measuring the weight of each feature that affects the workload for the subject and the weighted overall

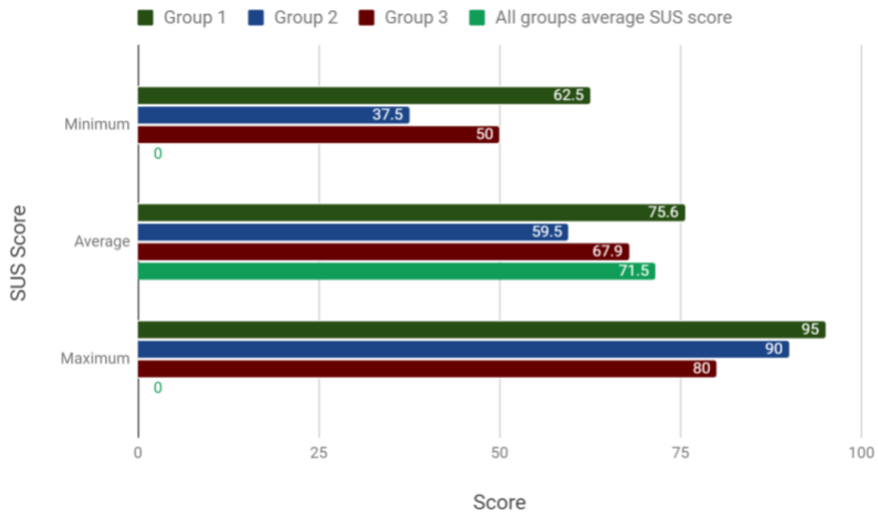

Figure 7.6: SUS score bar chart. The minimum, maximum and average of these scores found in our sample are clustered by group. The average SUS score of all samples without the division by group is also presented.

workload score measurement by the subjects. The weight measurement is the step in which the subject weights the most important factors to the experience of workload while using the system. The weighted overall workload score uses the weighted factors and the rating of each factor given by the subject to evaluate the workload of the system.

In Figure 7.4 we could notice that the average weighting of subjects in group 1 and 3 did consider similar factors as the most important to the experience of workload, while group 2 did think performance was a less relevant factor to the workload while mental demand and frustration were more relevant. More study on the subject has to be conducted and is further described in Chapter 8, but the results could reflect a new focus on further development of the system that is aimed to reduce the mental demand and frustration while elements from group 2 use the system.

The weighted overall workload score represented in Fig. 7.5 shows that in our subject sample, the average score for each group did vary a lot. If the workload is analysed according to Tab. 7.2, we conclude that the average workload for elements from group 1 would fit the system in medium workload, while the average workload for subjects in group 2 and 3 would fit into somewhat high workload category. Despite the fact that group 2 and 3 presents a higher average overall score, the all groups merged average overall score is lower than group 2 and 3 and fits in medium workload. This happens because the number of members in group 1 are unbalance in relation to members of group 2 and 3.

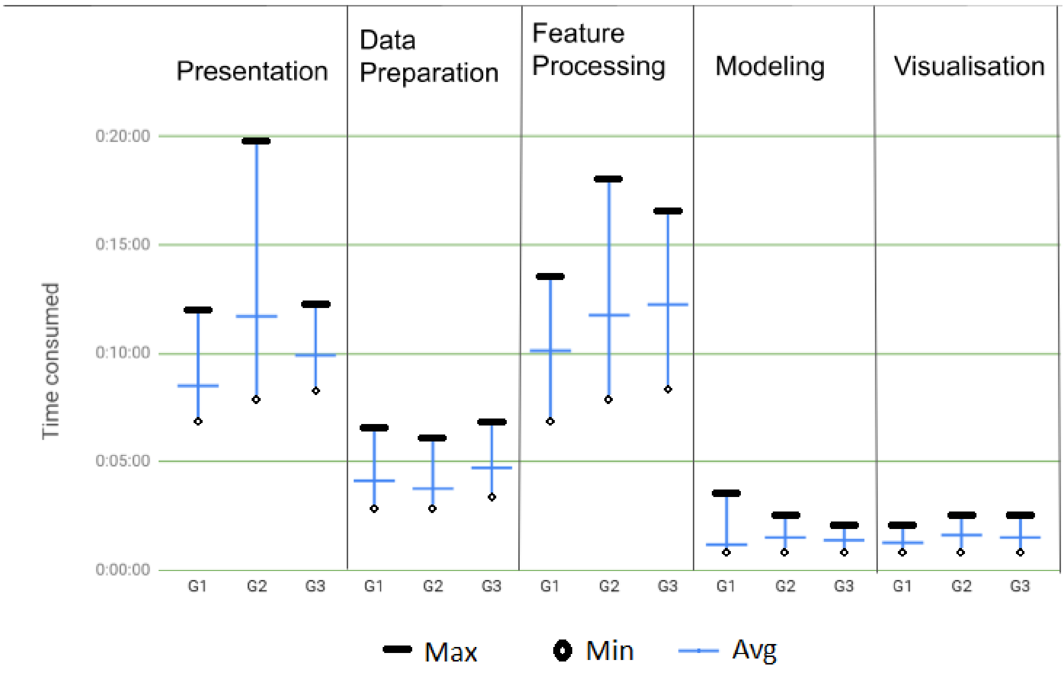

Figure 7.7: Minimum, Maximum, and Average time consumed by minute of the subjects separated in presentation and tabs. G1 stands for Group 1, G2 stands for Group 2, and G3 stands for Group 3.

7.5.2 SUS

For a system to be used in an industrial environment the usability of the system should get a presentable rating to avoid confusion, errors and facilitating from the overall experience of the potentially various daily users of the system.

We did compare our system's SUS scores collected from our subjects sample presented in Figure 7.6 and the interpretation of acceptable SUS score from Bangor et al. work [102]. The used interpretation of SUS scores consider 70 or greater a passable grade. Our average score was 71.5, slightly superior to the pass grade, with the average for subjects in group 1 reaching 75.6. In the other hand, the performance of groups 2 and 3 were bellow the pass grade, with 59.5 and 67.9

Workload	Value
Low	0-9
Medium	10-29
Somewhat high	30-49
High	50-79
Very high	80-100

Table 7.2: Table of interpretation Score of NASA TLX[101] seen in Prabaswari et al. work [101].

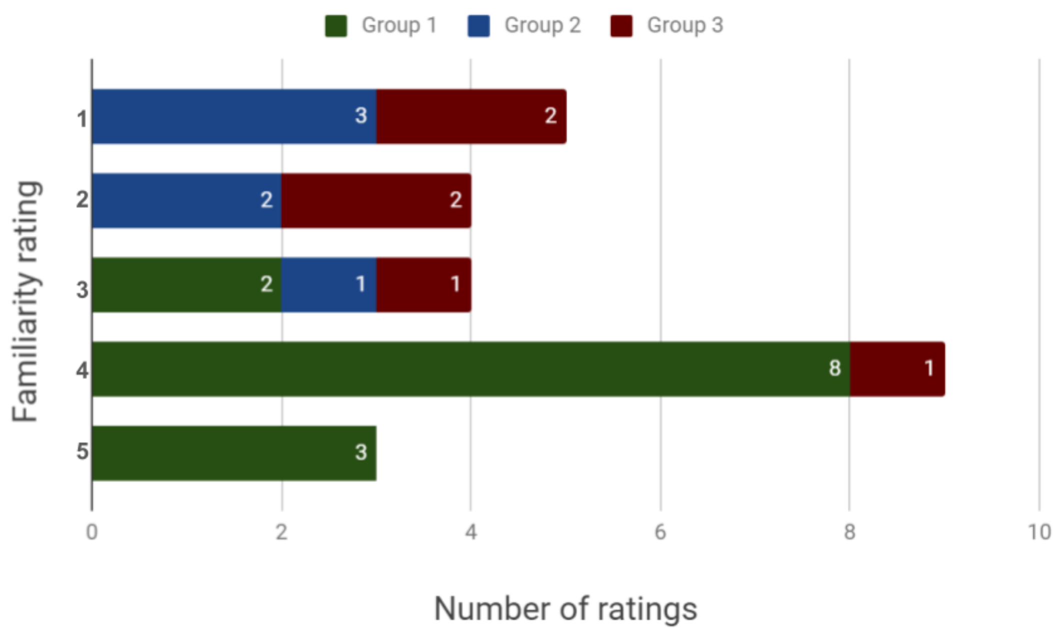

Figure 7.8: Subjects ML knowledge rating by group.

average grades respectively. This implies that, in future steps, the usability of the system could be re-worked based on a study of usability week points for groups 2 and 3.

7.5.3 Time per task

The time consumed per demo task presented in Figure 7.7 has shown that despite the average presentation time, the average times of all groups conducted in all tabs were similar, showing this way that the time used in the system is similar between all groups. Among the front-end tabs we could notice a disparity in which the tasks in feature processing were the most time consuming among the others. We conclude that the feature processing tab could be re-structured in order to consume less time in our system use.

7.5.4 ML familiarity rating

During the evaluation process it was asked for users in a scale from one to five how familiar they were with ML technologies, these results are presented in Figure 7.8. We could notice an unbalance between group 1 and groups 2 and 3 where group 1 rating is mostly concentrated in grade 4 while most of groups 2 and 3 ratings are concentrated in grades 1 and 2.

SemML is a complex system that relies heavily on ML technologies. Since ML technologies play a big role in the use of our system, the groups 2 and 3 could have their usability, workload and overall experience while using the system affected by this lack of familiarity with these technologies. We propose a re-work on the presentation in the development of future versions of the system in order to better familiarize the user subjects to the ML technologies.

Chapter 8

Conclusion

This thesis conducted a technological research on a system solution capable of providing a quality monitoring system based on ML and semantic technologies for industrial environment. The technology research main objective is to create better artifacts that can be used in future development iterations of the system solution. Our solution should address the main challenges and attributes involved in this research topic. Answering our research questions would answer how such solution was achieved and performed.

In this chapter we present the sections contributions and the future work of this thesis. The contribution presents this work's artifacts, the produced element results from our work, the contribution also contains the hypothesis and research questions conclusions, where the hypothesis is discussed and the research questions are presented and discussed. The future work section contains proposed directions for future expansion of this work.

8.1 Contributions

8.1.1 Thesis artifacts

This thesis produced a system version of our semantic based ML condition monitoring tool, SemML. This new version contains the front-end of tabs data preparation, feature processing, modeling and visualisation. These tabs behave as they would in the industrial environment and contain functionalities that assess the functionalities and challenges from steps 3, 4 and 5 from our solution workflow, Figure 4.1.

All artifacts produced in our technological research can provide material publicly available for any study on the research topic. The artifacts also can be used for the development of future versions of SemML system solution. The produced artifacts from this thesis are the following:

1. A study material about the research topic problems, industrial use case, existing solutions and background used in the development of our solution. This artifact can be seen in the problem analysis step:

   a. A background study in the literature about problem related topics and technologies.
   b. Compiled information on an industrial use case report in which the problem is analysed under a real world industrial optic.
   c. A study on the workflow of the proposed solution, along with its challenges, requirements, quality attributes and a study on evaluation processes for our solution.
2. The structuring of our system solution artifact for steps 3, 4 and 5 of our proposed workflow and a detailed description of how it address the requirements and quality attributes of the system.
3. SemML front-end solution artifact for the DKA, SEML and OI semantic components.
4. The descriptive evaluation process artifact of our system solution along with a discussion on results found.

8.1.2 Hypothesis and research questions conclusion

Our hypothesis claims that it is possible to build a system for our research topic that assess its challenges, ML techniques for condition monitoring in the industrial environment are often time costly and therefore expensive. Our hypothesis depends on our RQ1, if the answer to RQ1 describes one possible system solution that handles the challenges, our hypothesis would be correct.

RQ1 asks how a system can be structured to handle the main challenges of ML based condition monitoring of industrial processes. We had answered RQ1 by defining system requirements that would address the challenges involved in the process in Chapter 4. We did develop and explained how our semantic based system can address these system requirements in Chapter 5. We also did evaluate if the users could perform the tasks that would cover all requirements and all users could finish the proposed demo tasks in Chapter 7. These presented elements describe how a solution to RQ1 can be structured, this also validates our research hypothesis.

RQ2 asks first how the system fits the industrial use case and second how it could be built to be operated by professional from different domains. The needs for our system to be used in our industrial use case were described in Chapter 4 and how they were addressed is described in Chapter 5, these elements answer the first half of RQ2. The direct answer to the second half of RQ2 is that all users from different specialization groups could finish the demo task in our system, therefore, the way the current version of the system is structured could be used by professionals from different domains. Despite the fact the tasks were conducted and system could be used by different professionals, the performance of some specialization groups was not positive in regards to usability and workload.

RQ3 asks how the system could be evaluated and provide artifacts for further development versions. It is answered in Chapter 4 where the evaluation process is chosen, explained and justified. The artifacts from the evaluation process that are going to be used in future development of the system are the discussion of the results and the results presented in Chapter 7.

8.2 Future work

As we concluded this work some topics of future expansion were found:

- Finish the implementation of the Business logic, SemML database and API Handler. Currently these architectural components are not fully developed due to the short development time during this thesis, hence the system was running in object mode. All artifacts from this thesis could be taken into consideration when developing these new components for a better development process.
- Conduct a more extensive and balanced evaluation process. Due to time constrains the evaluation process was conducted in a small and unbalanced users sample, therefore our overall evaluation results were more inclined to group 1. This is important to provide a well functioning system to different domains, related to RQ2.
- Rework the presentation and introduction about the system and concepts. It has been seen a correlation between users familiarity rating of ML technologies and the NASA-TLX and SUS grades. Since our system is complex and rely heavily on ML technologies, it is proposed a re-structure of the presentation in order to provide a more clear explanation about ML technologies used in our system.
- Conduct new evaluation processes with new clusters. It is possible that the clustering of subjects in different groups such as level of education or years of experience could provide valuable insights such as particular groups that did not perform well. This information could provide artifacts that would guide the development of new versions of the system that could better fit all types of users.
- Rework the feature processing tab front-end. It might be interesting for new development versions of the system that the feature processing could be reworked in order to consume less time since the time consumed in the tab by the users in our sample was bigger than the time consumed in other tabs.
- Conduct a study on usability week points for groups 2 and 3. For groups 2 and 3 the average SUS score was bellow our set passing grade, this could indicate that a study on the system usability for users from those groups could be conducted to guide the development of future versions of the system that provide a better usability for those groups.

Bibliography

[1] H. Kagermann, 'Change Through Digitization – Value Creation in the Age of Industry 4.0,' in *Management of Permanent Change*, 2015.

[2] L. D. Xu, E. L. Xu and L. Li, 'Industry 4.0: State of the art and future trends,' *International Journal of Production Research*, vol. 56, no. 8, pp. 2941–2962, 2018.

[3] ITU, 'Recommendation ITU-T Y.2060: Overview of the Internet of things,' International Telecommunication Union, Tech. Rep., 2012.

[4] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld and M. Hoffmann, 'Industry 4.0,' *Business & information systems engineering*, vol. 6, no. 4, pp. 239–242, 2014.

[5] S. Chand and J. Davis, 'What is Smart Manufacturing,' *Time Magazine Wrapper*, vol. 7, pp. 28–33, 2010.

[6] T. Wuest, D. Weimer, C. Irgens and K.-D. Thoben, 'Machine Learning in Manufacturing: Advantages, Challenges, and Applications,' *Production & Manufacturing Research*, vol. 4, no. 1, pp. 23–45, 2016.

[7] B. Zhou, M. Chioua and J.-C. Schlake, 'Practical Methods for Detecting and Removing Transient Changes in Univariate Oscillatory Time Series,' *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7987–7992, 2017.

[8] D. Mikhaylov, B. Zhou, T. Kiedrowski, R. Mikut and A. F. Lasagni, 'Machine Learning Aided Phase Retrieval Algorithm for Beam Splitting With an LCoS-SLM,' in *Laser Resonators, Microresonators, and Beam Control XXI*, vol. 10904, 2019, p. 109041M.

[9] D. Mikhaylov, B. Zhou, T. Kiedrowski, R. Mikut and A.-F. Lasagni, 'High Accuracy Beam Splitting Using SLM Combined With ML Algorithms,' *Optics and Lasers in Engineering*, vol. 121, pp. 227–235, 2019.

[10] B. Zhou, M. Chioua, M. Bauer, J.-C. Schlake and N. F. Thornhill, 'Improving Root Cause Analysis by Detecting and Removing Transient Changes in Oscillatory Time Series With Application to a 1,3-Butadiene Process,' *Industrial & Engineering Chemistry Research*, vol. 58, pp. 11 234–11 250, 2019.

[11] I. Errandonea, S. Beltrán and S. Arrizabalaga, 'Digital twin for maintenance: A literature review,' *Computers in Industry*, vol. 123, p. 103 316, 2020, ISSN: 0166-3615. DOI: https://doi.org/10.1016/j.compind.2020.103316. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361520305509.

[12] R. Rosen, G. von Wichert, G. Lo and K. D. Bettenhausen, 'About the importance of autonomy and digital twins for the future of manufacturing,' *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 567–572, 2015, 15th IFAC Symposium onInformation Control Problems inManufacturing, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2015.06.141. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896315003808.

[13] Coop, *What is the cost to deploy and maintain a machine learning model?* Accessed 17. Jul 2021, phData, May 2021. [Online]. Available: https://www.phdata.io/blog/what-is-the-cost-to-deploy-and-maintain-a-machine-learning-model/.

[14] DIN, 'EN14610: Welding and Allied Processes - Definitions of Metal Welding Processes,'

[15] B. Zhou, Y. Svetashova, T. Pychynski, I. Baimuratov, A. Soylu and E. Kharlamov, 'SemFE: Facilitating ML Pipeline Development with Semantics,' in *29th ACM International Conference on Information and Knowledge Management*, ACM, 2020, pp. 3489–3492.

[16] E. Kharlamov, S. Brandt, E. Jiménez-Ruiz, Y. Kotidis, S. Lamparter, T. Mailis, C. Neuenstadt, Ö. L. Özçep, C. Pinkel, C. Svingos, D. Zheleznyakov, I. Horrocks, Y. E. Ioannidis and R. Möller, 'Ontology-Based Integration of Streaming and Static Relational Data With Optique,' in *SIGMOD*, 2016, pp. 2109–2112.

[17] E. Kharlamov, N. Solomakhina, Ö. Özcep, D. Zheleznyakov, T. Hubauer, S. Lamparter, M. Roshchin, A. Soylu and S. Watson, 'How semantic technologies can enhance data access at siemens energy,' vol. 8796, Oct. 2014, pp. 601–619, ISBN: 978-3-319-11963-2. DOI: 10.1007/978-3-319-11964-9_38.

[18] I. Solheim and K. Stølen, 'Technology research explained,' Mar. 2007.

[19] Bosch, *Bosch center for artificial intelligence home*, Accessed 17. Jan 2022, Bosch. [Online]. Available: https://www.bosch-ai.com/.

[20] B. Zhou, Y. Svetashova, A. Gusmao, A. Soylu, G. Cheng, R. Mikut, A. Waaler and E. Kharlamov, 'Semml: Facilitating development of ml models for condition monitoring with semantics,' *Journal of Web Semantics*, vol. 71, p. 100 664, 2021, ISSN: 1570-8268. DOI: https://doi.org/10.1016/j.websem.2021.100664. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570826821000391.

[21] B. Zhou, Y. Svetashova, S. Byeon, T. Pychynski, R. Mikut and E. Kharlamov, 'Predicting Quality of Automated Welding with Machine Learning and Semantics: a Bosch Case Study,' in *29th ACM International Conference on Information and Knowledge Management*, ACM, 2020, pp. 2933–2940.

[22] Y. Svetashova, B. Zhou, T. Pychynski, S. Schmidt, Y. Sure-Vetter, R. Mikut and E. Kharlamov, 'Ontology-Enhanced Machine Learning: a Bosch Use Case of Welding Quality Monitoring,' in *ISWC*, 2020, pp. 531–550.

[23] DIN, 'Maintenance-Maintenance Terminology,' *Trilingual Version EN 13306:2017*, vol. 13306, p. 2017, 2018.

[24] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang and R. X. Gao, 'Deep Learning and Its Applications to Machine Health Monitoring,' *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.

[25] R. X. Gao, X. Tang, G. Gordon and D. O. Kazmer, 'Online product quality monitoring through in-process measurement,' *CIRP Annals*, vol. 63, no. 1, pp. 493–496, 2014, ISSN: 0007-8506. DOI: `https://doi.org/10.1016/j.cirp.2014.03.041`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0007850614000444`.

[26] ISO, '14327:2004: Resistance Welding – Procedures for Determining the Weldability Lobe for Resistance Spot, Projection and Seam Welding,' International Organization for Standardization, Geneva, CH, Standard, 2004.

[27] DVS, '2902–3: Widerstandspunktschweißen von Stählen bis 3 mm Einzeldicke – Konstruktion und Berechnung,' German, Deutscher Verband für Schweißen und verwandte Verfahren e. V., Düsseldorf, DE, Standard, 2016.

[28] E. Kharlamov, G. Mehdi, O. Savkovic, G. Xiao, E. G. Kalayci and M. Roshchin, 'Semantically-Enhanced Rule-Based Diagnostics for Industrial Internet of Things: The SDRL Language and Case Study for Siemens Trains and Turbines,' *J. Web Semant.*, vol. 56, pp. 11–29, 2019.

[29] E. Kharlamov, T. Mailis, G. Mehdi, C. Neuenstadt, Ö. L. Özçep, M. Roshchin, N. Solomakhina, A. Soylu, C. Svingos, S. Brandt, M. Giese, Y. E. Ioannidis, S. Lamparter, R. Möller, Y. Kotidis and A. Waaler, 'Semantic Access to Streaming and Static Data at Siemens,' *J. Web Semant.*, vol. 44, pp. 54–74, 2017.

[30] E. Kharlamov, D. Hovland, M. G. Skjæveland, D. Bilidas, E. Jiménez-Ruiz, G. Xiao, A. Soylu, D. Lanti, M. Rezk, D. Zheleznyakov, M. Giese, H. Lie, Y. E. Ioannidis, Y. Kotidis, M. Koubarakis and A. Waaler, 'Ontology Based Data Access in Statoil,' *J. Web Semant.*, vol. 44, pp. 3–36, 2017.

[31] J. N. Desai, S. Pandian and R. K. Vij, 'Big data analytics in upstream oil and gas industries for sustainable exploration and development: A review,' *Environmental Technology and Innovation*, p. 101 186, 2020, ISSN: 2352-1864.

[32] U. Fayyad, G. Piatetsky-Shapiro and P. Smyth, 'From Data Mining to Knowledge Discovery in Databases,' *AI Magazine*, vol. 17, no. 3, pp. 37–37, 1996.

[33] R. Mikut, M. Reischl, O. Burmeister and T. Loose, 'Data Mining in Medical Time Series,' *Biomedizinische Technik*, vol. 51, 2006.

[34] T. R. Gruber, 'A translation approach to portable ontology specifications,' *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[35] M. Gruninger, 'Ontology applications and design.,' Jan. 2002.

[36] H. Kaiya and M. Saeki, 'Using domain ontology as domain knowledge for requirements elicitation,' in *14th IEEE International Requirements Engineering Conference (RE'06)*, 2006, pp. 189–198. DOI: 10.1109/RE.2006.72.

[37] V. Janev and S. Vranes, 'Semantic web technologies: Ready for adoption?' *IT Professional*, vol. 11, no. 5, pp. 8–16, 2009. DOI: 10.1109/MITP.2009.107.

[38] A. Gliozzo, O. Biran, S. Patwardhan and K. McKeown, 'Semantic technologies in ibm watson,' Aug. 2013, pp. 85–92.

[39] *Semantic systems, arcelormittal y reydesa recycling aúnan esfuerzos por la sostenibilidad*. [Online]. Available: https://aclima.eus/semantic-systems-arcelormittal-y-reydesa-recycling-aunan-esfuerzos-por-la-sostenibilidad/.

[40] S. Elmer, F. Jrad, T. Liebig, A. ul Mehdi, M. Opitz, T. Stauß and D. Weidig, 'Ontologies and reasoning to capture product complexity in automation industry,' in *ISWC (Posters & Demonstrations and Industry Tracks)*, 2017.

[41] E. Kharlamov, D. Hovland, E. Jiménez-Ruiz, D. Lanti, H. Lie, C. Pinkel, M. Rezk, M. G. Skjæveland, E. Thorstensen, G. Xiao, D. Zheleznyakov and I. Horrocks, 'Ontology based access to exploration data at statoil,' in *ISWC*, 2015, pp. 93–112.

[42] K. Stancin, P. Poščić and D. Jaksic, 'Ontologies in education – state of the art,' *Education and Information Technologies*, vol. 25, Nov. 2020. DOI: 10.1007/s10639-020-10226-z.

[43] D. P. Lupp, 'A higehr-level view of ontological modeling: Rule-based apraoches for data transformation, modeling, and maintenance,' *UiO*, vol. 25, 2019.

[44] M. Obitko, *Reasoning*, Accessed 11. Dec 2021. [Online]. Available: https://www.obitko.com/tutorials/ontologies-semantic-web/reasoning.html.

[45] B. Zhou, T. Pychynski, M. Reischl and R. Mikut, 'Comparison of Machine Learning Approaches for Time-series-based Quality Monitoring of Resistance Spot Welding (RSW),' *Archives of Data Science, Series A (Online First)*, vol. 5, no. 1, p. 13, 2018.

[46] Y. Zhang, G. Chen and Z. Lin, 'Study on Weld Quality Control of Resistance Spot Welding Using a Neuro-Fuzzy Algorithm,' in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, Springer, 2004, pp. 544–550.

[47] A. El Ouafi, R. Bélanger and J.-F. Méthot, 'An Online ANN-based Approach for Quality Estimation in Resistance Spot Welding,' in *Advanced Materials Research*, vol. 112, 2010. DOI: 10.4028/www.scientific.net/AMR.112.141.

[48] Y. Cho and S. Rhee, 'New Technology for Measuring Dynamic Resistance and Estimating Strength in Resistance Spot Welding,' *Measurement Science and Technology*, vol. 11, no. 8, 2000.

[49] P. Ristoski and H. Paulheim, 'Semantic web in data mining and knowledge discovery: A comprehensive survey,' *Journal of Web Semantics*, vol. 36, pp. 1–22, 2016.

[50] F. Jaensch, A. Csiszar, C. Scheifele and A. Verl, 'Digital Twins of Manufacturing Systems as a Base for Machine Learning,' in *2018 25th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, IEEE, 2018, pp. 1–6.

[51] S. Borgo and P. Leitão, 'The Role of Foundational Ontologies in Manufacturing Domain Applications,' in *OTM*, 2004.

[52] Z. Usman, R. I. M. Young, N. Chungoora, C. Palmer, K. Case and J. Harding, 'A Manufacturing Core Concepts Ontology for Product Lifecycle Interoperability,' in *International IFIP Working Conference on Enterprise Interoperability*, 2011, pp. 5–18.

[53] S. Lemaignan, A. Siadat, J.-Y. Dantan and A. Semenenko, 'MASON: A Proposal for an Ontology of Manufacturing Domain,' in *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, IEEE, 2006, pp. 195–200.

[54] S. Krima, R. Barbau, X. Fiorentini, R. Sudarsan and R. D. Sriram, 'OntoSTEP: OWL-DL Ontology for STEP,' NIST, Tech. Rep., 2009.

[55] X. Fiorentini, I. Gambino, V.-C. Liang, S. Rachuri, M. Mani, C. B. Nistir, C. Bock, C. M. Gutierrez and J. M. Turner, 'An Ontology for Assembly Representation,' Tech. Rep., 2007.

[56] D. Šormaz and A. Sarkar, 'SIMPM – Upper-level Ontology for Manufacturing Process Plan Network Generation,' *Robotics and Computer-Integrated Manufacturing*, vol. 55, 2019, ISSN: 07365845.

[57] E. Güzel Kalaycı, I. Grangel-González, F. Loesch, G. Xiao, A. Mehdi, E. Kharlamov and D. Calvanese, 'Semantic integration of bosch manufacturing data using virtual knowledge graphs,' Aug. 2020.

[58] R. Hai, S. Geisler and C. Quix, 'Constance: An Intelligent Data Lake System,' in *SIGMOID'16*.

[59] C. Quix, R. Hai and I. Vatov, 'GEMMS: A Generic and Extensible Metadata Management System for Data Lakes.,' in *CAiSE forum*, vol. 129, 2016.

[60] S. Jupp, T. Burdett, D. Welter, S. Sarntivijai, H. Parkinson and J. Malone, 'Webulous and the Webulous Google Add-On-A Web Service and Application for Ontology Building From Templates,' *Journal of Biomedical semantics*, vol. 7, 2016.

[61] H. Dietze, T. Z. Berardini, R. E. Foulger, D. P. Hill, J. Lomax, D. Osumi-Sutherland, P. Roncaglia and C. J. Mungall, 'Termgenie-A Web-Application for Pattern-Based Ontology Class Generation,' *Journal of Biomedical Semantics*, vol. 5, 2014.

[62] Z. Xiang, J. Zheng, Y. Lin and Y. He, 'Ontorat: Automatic Generation of New Ontology Terms, Annotations, and Axioms Based on Ontology Design Patterns,' *Journal of biomedical semantics*, vol. 6, no. 1, pp. 1–10, 2015.

[63] C. Ward, P. Weston, E. Stewart, H. Li, R. Goodall, C. Roberts, T. X. Mei, G. Charles and R. Dixon, 'Condition monitoring opportunities using vehicle-based sensors,' *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 225, Mar. 2011. DOI: `10.1243/09544097JRRT406`.

[64] V. J. Hodge, S. O'Keefe, M. Weeks and A. Moulds, 'Wireless sensor networks for condition monitoring in the railway industry: A survey,' *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1088–1106, 2015. DOI: `10.1109/TITS.2014.2366512`.

[65] A. Khalili and S. Auer, 'User interfaces for semantic authoring of textual content: A systematic literature review,' *Journal of Web Semantics*, vol. 22, pp. 1–18, 2013, ISSN: 1570-8268. DOI: `https://doi.org/10.1016/j.websem.2013.08.004`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1570826813000498`.

[66] C. Ncube, 'A requirements engineering method for cots-based systems development,' 2000.

[67] I. O. f. S. (ISO), *Usability of consumer products and products for public use — part 2: Summative test method*. [Online]. Available: `https://www.iso.org/obp/ui/#iso:std:iso:ts:20282:-2:ed-2:v1:en`.

[68] J. Brooke, '„sus-a quick and dirty usability scale (in" usability evaluation in industry", pw jordan, b thomas, i mclelland, ba weerdmeester (eds))“,' *Usability evaluation in industry (PW Jordan, B. Thomas, BA Weerdmeester, & AL McClelland (Eds.))*, 1996.

[69] *System usability scale (sus)*, Accessed 11. Dec 2021, Sep. 2013. [Online]. Available: `https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html`.

[70] S. G. Hart and C. D. Wickens, 'Workload assessment and prediction,' in Springer, 1990, pp. 257–296.

[71] C. D. Wickens, J. Lee, Y. D. Liu and S. Gordon-Becker, *Introduction to Human Factors Engineering (2nd Edition)*. USA: Prentice-Hall, Inc., 2003, ISBN: 0131837362.

[72] S. G. Hart, 'Nasa-task load index (nasa-tlx); 20 years later,' *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, pp. 904–908, 2006.

[73] J. B. Sexton and R. L. Helmreich, 'Analyzing cockpit communications: The links between language, performance, error, and workload,' *Human Performance in Extreme Environments*, vol. 5, no. 1, pp. 63–68, 2000.

[74] S. G. Hart and L. E. Staveland, 'Development of nasa-tlx (task load index): Results of empirical and theoretical research,' in *Advances in psychology*, vol. 52, Elsevier, 1988, pp. 139–183.

[75] E. Kharlamov, B. C. Grau, E. Jiménez-Ruiz, S. Lamparter, G. Mehdi, M. Ringsquandl, Y. Nenov, S. Grimm, M. Roshchin and I. Horrocks, 'Capturing Industrial Information Models With Ontologies and Constraints,' in *ISWC*, 2016.

[76] E. Kharlamov, Y. Kotidis, T. Mailis, C. Neuenstadt, C. Nikolaou, Ö. L. Özçep, C. Svingos, D. Zheleznyakov, Y. E. Ioannidis, S. Lamparter, R. Möller and A. Waaler, 'An Ontology-Mediated Analytics-Aware Approach to Support Monitoring and Diagnostics of Static and Streaming Data,' *J. Web Semant.*, vol. 56, pp. 30–55, 2019.

[77] A. Soylu, M. Giese, R. Schlatte, E. Jiménez-Ruiz, E. Kharlamov, Ö. L. Özçep, C. Neuenstadt and S. Brandt, 'Querying industrial stream-temporal data: An ontology-based visual approach,' *J. Ambient Intell. Smart Environ.*, vol. 9, no. 1, pp. 77–95, 2017.

[78] A. Soylu, E. Kharlamov, D. Zheleznyakov, E. Jiménez-Ruiz, M. Giese, M. G. Skjæveland, D. Hovland, R. Schlatte, S. Brandt, H. Lie and I. Horrocks, 'Optiquevqs: A visual query system over ontologies for industry,' *Semantic Web*, vol. 9, no. 5, pp. 627–660, 2018. DOI: `10.3233/SW-180293`. [Online]. Available: `https://doi.org/10.3233/SW-180293`.

[79] Y. Sun, L. Zhang, G. Cheng and Y. Qu, 'SPARQA: skeleton-based semantic parsing for complex questions over knowledge bases,' in *AAAI-IAAI-EAAI 2020*, 2020, pp. 8952–8959.

[80] I. Horrocks, M. Giese, E. Kharlamov and A. Waaler, 'Using Semantic Technology to Tame the Data Variety Challenge,' *IEEE Internet Comput.*, vol. 20, no. 6, pp. 62–66, 2016.

[81] M. Ringsquandl, E. Kharlamov, D. Stepanova, M. Hildebrandt, S. Lamparter, R. Lepratti, I. Horrocks and P. Kröger, 'Event-Enhanced Learning for KG Completion,' in *ESWC*, 2018.

[82] E. Kharlamov, N. Solomakhina, Ö. L. Özçep, D. Zheleznyakov, T. Hubauer, S. Lamparter, M. Roshchin, A. Soylu and S. Watson, 'ISWC,' 2014, pp. 601–619.

[83] J. Strötgen, T. Tran, A. Friedrich, D. Milchevski, F. Tomazic, A. Marusczyk, H. Adel, D. Stepanova, F. Hildebrand and E. Kharlamov, 'Towards the bosch materials science knowledge base,' in *ISWC (Posters & Demonstrations, Industry, and Outrageous Ideas)*, 2019, pp. 323–324.

[84] E. G. Kalayci, I. Grangel-González, F. Lösch, G. Xiao, A. ul Mehdi, E. Kharlamov and D. Calvanese, 'Semantic integration of bosch manufacturing data using virtual knowledge graphs,' in *ISWC*, 2020, pp. 464–481.

[85] Raunakjhawar, *Extract, transform, and load (etl) - azure architecture center*. [Online]. Available: `https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl`.

[86] I. C. Education, *What is exploratory data analysis?* [Online]. Available: `https://www.ibm.com/cloud/learn/exploratory-data-analysis`.

[87] B. Zhou, E. Kharmalov and A. Gusmao, *Sindais4: Scaling industrial ai with semantics*, Jul. 2021. [Online]. Available: `https://sirius-labs.no/sindais4-scaling-industrial-ai-with-semantics/`.

[88] S. Abburu, 'A survey on ontology reasoners and comparison,' *International Journal of Computer Applications*, vol. 57, no. 17, 2012.

[89] B. W. Boehm, 'A spiral model of software development and enhancement,' *Computer*, vol. 21, no. 5, pp. 61–72, 1988.

[90] S. Verma, 'Analysis of strengths and weakness of sdlc models,' *International Journal of Advance Research in*, 2014.

[91] D. Field, *Figma the Collaborative Interface Design Tool.* https://www.figma.com, Accessed 11. Dec 2021, Figma, Inc., 2015. [Online]. Available: `https://www.figma.com/`.

[92] K. Schwaber and J. Sutherland, 'The scrum guide,' *Scrum Alliance*, vol. 21, no. 1, 2011.

[93] *Trello.* Accessed 11. Dec 2021. [Online]. Available: `https://www.trello.com/`.

[94] A. Walker, *Soap vs. rest: Difference between web api services*, Feb. 2022. [Online]. Available: `https://www.guru99.com/comparison-between-web-services.html`.

[95] R. Ganatra, *Graphql vs. rest apis*, Nov. 2021. [Online]. Available: `https://graphcms.com/blog/graphql-vs-rest-apis`.

[96] *Stack overflow developer survey 2021*, Accessed 11. Dec 2021. [Online]. Available: `https://insights.stackoverflow.com/survey/2021`.

[97] S. Bechhofer, F. v. Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider and L. A. Stein, *Owl web ontology language reference*, Accessed 11. Dec 2021, Feb. 2004. [Online]. Available: `https://www.w3.org/TR/owl-ref/`.

[98] D. Beckett and T. Berners-Lee, *Turtle - terse rdf triple language*, Accessed 11. Dec 2021, Mar. 2011. [Online]. Available: `https://www.w3.org/TeamSubmission/turtle/`.

[99] S. Union, *High-performance graph database: Stardog*, Accessed 11. Dec 2021. [Online]. Available: `https://www.stardog.com/platform/features/high-performance-graph-database/`.

[100] *Metaphactory - your knowledge democratization platform*, Accessed 11. Dec 2021. [Online]. Available: `https://metaphacts.com/product`.

[101] M. Field, *Human systems integration division @ nasa ames - home*. [Online]. Available: `https://humansystems.arc.nasa.gov/groups/TLX/downloads/TLX_pappen_manual.pdf`.

[102] A. Bangor, P. T. Kortum and J. T. Miller, 'An empirical evaluation of the system usability scale,' *International Journal of Human–Computer Interaction*, vol. 24, no. 6, pp. 574–594, 2008. DOI: `10.1080/10447310802205776`. eprint: `https://doi.org/10.1080/10447310802205776`. [Online]. Available: `https://doi.org/10.1080/10447310802205776`.