Semi-Automated Ontology Generation Process from Industrial Product Data Standards

Alvaro L. Fraga, Marcela Vegetti, Horacio P. Leone
INGAR – Instituto de Desarrollo y Diseño (3000), Avellaneda 3657, Santa Fe, República Argentina.
{alvarofraga, mvegetti, hleone}@santafe-conicet.gov.ar

Abstract. Ontology development has become an important research area for manufacture industries. Ontologies are one of the most popular methods to achieve semantic interoperability between information systems. In previous works, an ontology network that reuses ontological and non-ontological resources have been proposed in order to reach semantic interoperability. However, processing non-ontological resources to build an ontology is a great time-consuming task. Therefore, this work presents a framework and a prototype tool to support the reuse of the non-ontological resources involved in the development of the ontology network.

Keywords: Ontologies, Product Lifecycle Process, Natural Process Language, Semantic Interoperability

1 Introduction

Product Lifecycle Management (PLM) is an approach including a set of methods, models, and IT tools for managing product information, engineering processes and applications along the different phases of the product lifecycle [1]. In the lifespan of a product, many areas of the manufacturing organization are involved, each one having its own applications, processes and data models. Therefore, the PLM approach has to deal with the integration and interoperability of the different information silos belonging to the lifecycle of a product. The problem of interoperability gets worse when in the lifecycle of product areas from different organizations are involved. The word interoperability can be defined as the ability of two or more systems to cooperate despite different execution environment, language and interfaces [1]. There are three main barriers to overcome before reaching interoperability [2]:

- Conceptual: Concerned with the syntactic and semantic differences of information to be exchanged.
- Technological: Refers to the incompatibility of information technologies
- Organizational: Refers to the definition of responsibility, authority, and structure.

This paper focuses on the conceptual barrier, particularly semantic interoperability between standards of industrial product data, which are commonly used for information representation in tools and applications supporting PLM in manufacturing industries.

Standards are a great choice to solve interoperability problems. Those published by the Technical Committee 184 Subcommittee 4 from the International Standard Organization (ISO) can be highlighted [3] in manufacturing domain. The mentioned committee offers capabilities to describe and manage industrial product data throughout the life of the product [4]. However, as one tries to implement a set of standards from the ISO TC184/SC4, problems appear in different areas where the terminology is less rigorously defined. In [5] the results of an analysis of the vocabulary defined by a set of standards from the mentioned standard has been presented. Table 1 summarizes the main issues that were identified in [5]. These issues can lead to potential interoperability problems.

Table 1. Some problems found between standards

Terms involved	Problems found
Resource	Terms with more than one definition between ISO 15531-1[6], ISO 10303-239[7], ISO 10303-232[8].
Process, Resource, Product	Resource and Product are defined as a result of a Process in ISO 10303-239 [7] and ISO 15926-11[9], but Process is only defined as a product producer in ISO 10303-49[10].
Resource, Product, Product Information	Resource, Product, and Product Information have the same definition along ISO 10303-232[8], ISO 13584-102[11], ISO 10303-1[12].

One of the most popular and widespread approach to achieve a successful information exchange and semantic interoperability is the definition and use of ontologies [13]. Therefore, in order to solve the main semantic problems among standard terms belonging to ISO TC184/SC4, an ontology network has been proposed. A description of the ontology network three level hierarchical organization can be found in [14]. The lowest level is compounded by ontologies representing the vocabulary defined in specific standards in the scope of ISO TC184/SC4. The development of such ontology has been made from scratch, analyzing the standard documentation and defining concepts and relations in the ontology editor. This activity represents a great time-consuming task. For this reason and in order to speed up the development process of the lowest level ontologies of the proposed network, this article presents a methodology for obtaining an OWL ontology from the documentation of the standards and a prototype application that supports it. The methodology is based on natural language processing (NLP) techniques to extract concepts, relationships, and properties from the specific documentation.

This paper is structured as follows. Section 2 presents a brief review and background of ontology learning using natural language processing techniques and related works. Section 3 introduces briefly the three levels of the ontology network and then describes the proposed methodology for creating ontologies to be added to the network and a tool that implements it. Finally, in Section 4 conclusions and future work are drawn.

2 Related Works

Ontology learning is defined as the set of methods and techniques for building, enriching or adapting an ontology in a semi-automated way [15]. Techniques from various fields are used e.g. statistics, machine learning, NLP, and information retrieval. This technique helps ontology engineers to build ontologies and can be applied to unstructured, semi-structured and fully structured data to support semi-automatic and cooperative ontology engineering [16]–[18].

NLP techniques include chunking, tagging, parsing, wrapping, structured and dependency analysis. Furthermore, Lexico-syntactic pattern (LSP) uses predefined rules to extract the conceptual links between entities. For example, Hearst Pattern [19] is used to find hyponym and hypernym relations. Other authors, extract taxonomic relationships using patterns [20] to build ontologies. In [21]–[23], authors use syntactic patterns to extract relations between concepts to perform ontology learning from text.

Dahab et al. [21] presents TextOntoEx, a tool for building ontologies from natural language texts, using a pattern-based approach. However, TextOntoEx was applied in agricultural domain.

Jiang and Tan [22] developed Concept-Relation-Concept Tuple-based Ontology Learning (CRCTOL). This system employs a combination of statistical and LSP methods to generate ontologies. Authors also shows its performance against other tools like Text-to-Onto [24] and its successor Text2Onto [20]. However, CRCTOL was designed for building domain ontology from scratch and not enriching an existing ontology.

One of the main activities for this kind of strategy is the named entity recognition (NER). Named entity (NE) is a term designated to those words that are invariant or rigid designators that point at referential entities e.g. dates, proper nouns, formulae [25]. In an ontological point of view, these NE act as ontological objects in documents and are used to find what a document studies [26].

NER is known as the set of strategies and techniques to identify NE in texts. NER often makes use of named entity dictionaries and specific types of documents. Furthermore, named entity tagging could have ambiguities and variations, so NER involves matching rules, i.e. regular expressions or rules that can be set by the NLP library or toolkit.

Other works that use Natural Language Processing techniques and tools to build ontologies, can be mentioned. Wauer et al. [27] present a generic architecture to integrate the existing sources of information focusing on product lifecycle management at a large company. The mentioned proposal uses the Unstructured Information Management Architecture (UIMA) [28] as a blueprint for extracting information. However, differently to our proposal, which can be apply to different type of manufacturing enterprise, the scope of Wauer et al. proposal is constrained to a specific industry and a system used in it.

Kasneci et al. [29] describe the YAGO architecture and its approach for extracting information to build an accurate knowledge base of common facts, which can be interpreted as a collection of RDF triples. To build the knowledge base, the core extractor gathers knowledge integrating information from different sources, performs rule-based information extraction over the sources and validates the extracted facts with WordNet taxonomy to derive the knowledge base.

As it was mentioned in previous paragraphs there are works that can build ontologies from heterogeneous sources applying NLP techniques. However, such proposals only focus on the ontology construction and do not consider its mapping to other existent ontologies. So, they cannot be used to support the ontology generation process proposes in [5].

In the present work, the semi-automatic NER strategy mentioned above is used for constructing ontologies to simplify the building task of the lowest level of the ontology network proposed in [5]. The following section gives more details of the applied strategy and its implementation in a prototype application.

3 Ontology Network Lowest Level Generation Process

This section introduces the methodology that supports a semi-automatic generation process of ontologies that formalize standard documentation and their addition into the lowest level of the ontology network proposed in [5]. First, a brief description of the structure of the mentioned network is presented. Then, the proposed methodology and a tool supporting it are described.

3.1 Standards Interoperability Ontology Network

The proposed ontology network is based on a three-layer architecture, which is shown in Fig. 1. The top level of the network specifies the most representative entities related to product management in manufacturing domains. The most prominent terms belonging to the standards involved in the study are Product, Process, Resource, and Enterprise. Among these terms a set of binary relations are identified: a product is produced by a process, a resource is used by a process, the resource is held by an Enterprise, and a process is involved in an Enterprise. This level serves to achieve an abstraction that is flexible and useful for systems involved in different areas of a company because it ensures interoperability without the need to go into details.

The middle level of the architecture of the proposed network is composed of four ontology modules. Each module is designed to refine the definition of the terms defined at the top level. The terms belonging to each module are linked among them and with the terms defined at the upper level using binary relations.

Below the layer mentioned in the previous paragraph, a mediation component is proposed. This component specifies SWRL (Semantic Web Rule Language) [30] rules to map concepts belonging to intermediate level ontologies and terms from the lowest level ontologies.

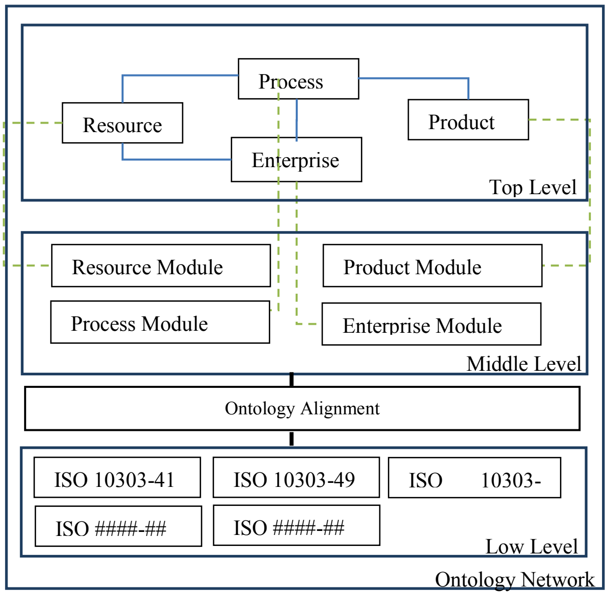


Fig. 1. Three Level Architecture Ontology Network Proposal

The aim of the lowest level of the network is to contain the ontologies that formalize the documentation of the standards from the ISO TC184/SC4. In order to a specific standard become part of this level it is necessary: i) to create an OWL (Ontology Web Language) ontology and ii) to specify alignment rules between terms in the new ontology and concepts already defined in ontologies already included into the ontology network. In the next section, a methodology supporting these two tasks is introduced.

3.2 Semi-Automatic Generation Process

Adding a new component into the lowest level of the Standards Interoperability Ontology Network described in section 3.1 requires the creation of ontologies that formalize the new added standard and its alignment with ontologies of the network. This section proposes a methodology that guides the semi-automatic process of ontology creation from specific standard documentation and its integration into the mentioned ontology network. Fig. 2 illustrates the activities defined by the proposed methodology.

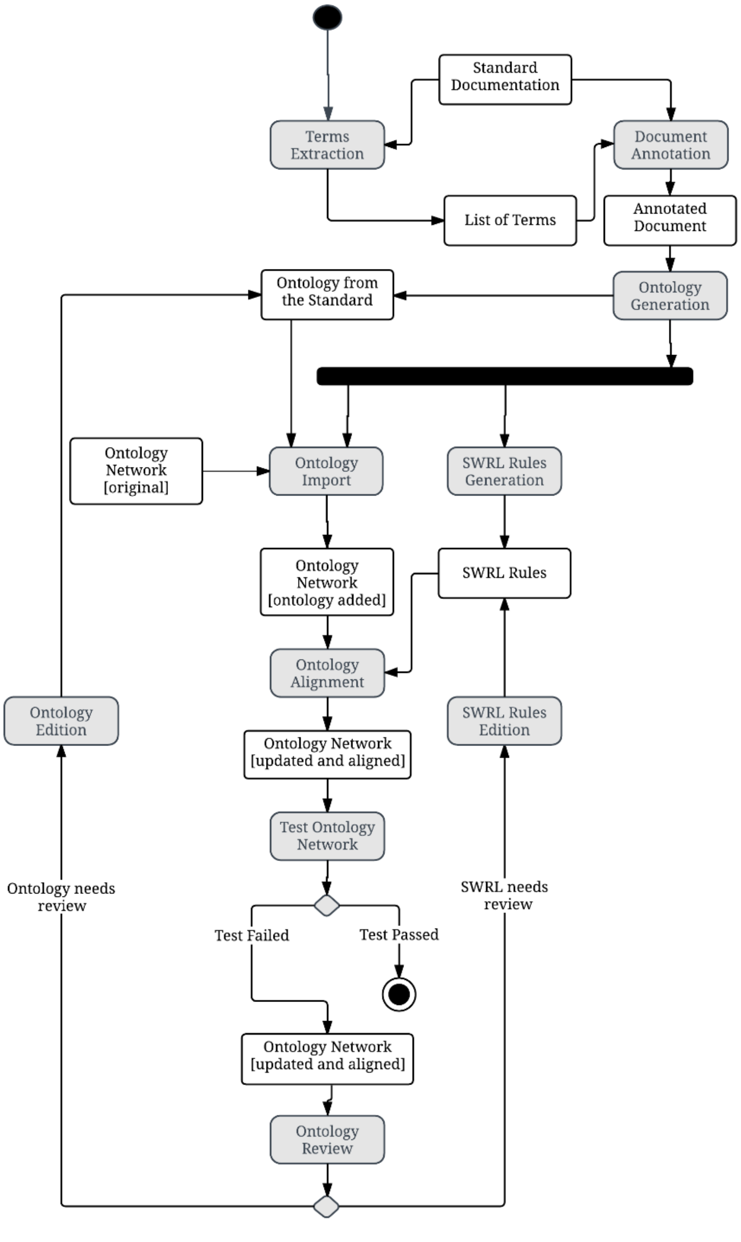


Fig.2. Activity Diagram of the Proposed Methodology

The main input of the proposed generation process is the documentation of the standard from which the ontology will be created. This documentation has to be written in portable document format (PDF) and express definitions of relevant terms through natural language or EXPRESS code blocks. In particular, for ISO 10303 standards.

The generation process first action is called Term Extraction. This task requires user interaction in order to to create a list of terms that the user identified as important from the standard documentation. In this activity, the user writes a plain text file containing one per line of the identified terms. This text file works as a seed to the tool for rapidly recognize terms that the user needs from the standard.

The text file with the terms and the standard documentation are the input of the Document Annotation action, which is an automatic process. This second activity highlights the terms, their definitions and relationships in the standard based on the list of terms and delivers to the next step the annotated standard document. The annotated standard document is the standard document that has been modified by the addition of tags that assign a category to words. These tags are identified by a unique name and were created through the definition of rules. Some of the created tags are: *object_property*, *data_property*, *type*, *hierarchy*, *entity*, *entity_domain*, *entity_range,* and *EXPRESS code*. The addition of tags into the standard documentation helps the transformation process in the identification of the kind of transformation that is required for each marked word.

The annotated document is the input of the Ontology Generation action, which involves the extraction and interpretation of the marked part of the annotated document to get as result a preliminary automatic generated ontology of the standard implemented in OWL.

In general, standard documentation is expressed using text in natural language. Hence, tagging is essential to get a good translation of the standard into the ontology. But there are some standards, especially those from the family of ISO 10303, specify concepts using EXPRESS code blocks. In order to have this different type of representation, the process separates the annotated text and classifies it in EXPRESS code or text in natural language. The Ontology Generation task handles both kinds of content in a different way. On one hand, the content in natural language is interpreted through the tags and the OWL code is generated. On the other hand, the EXPRESS code content is handled using EXPRESS to OWL strategies. Pauwels and Terkaj [31] exposed several strategies from different authors to take over a variety of EXPRESS fragments code to generate OWL code. Unfortunately, most of the strategies proposed in [31] lack of an implementation. So, authors had to implement some of the Pauwels and Terkaj strategies in order to deal with this activity. As a result of the Ontology Generation activity an operational ontology, defined in OWL file format is obtained.

Ontology Import is the third activity of the proposed methodology. Its aim is to add the OWL ontology, which is obtained in the previous task, as a module at the lowest level of the ontology network. The newly generated ontology needs to be interoperable with the intermediate level of the network. Therefore, the SWRL Rules Generation action has been triggered in order to define the rules to specify the concept mappings that are required. A file contained the new rules is the output of this task.

The Ontology Alignment step adds the SWRL rules to the ontology network. At the end of this task, the ontology network updated with the new module and its alignment rules is ready to be tested.

The next step is to test the ontology network out with the competency questions, represented as SPARQL queries. If the user is not satisfied with the results of the queries, the network should be reviewed to see if the problems are in the OWL ontology produced by Ontology Generation task and/or in the file generated by the SWRL Rules Generation action.

Ontology Edition and SWRL rules edition activities are executed to solve problems that occur in the OWL ontology or in the alignment rules, respectively. Once the errors are fixed, the import, alignment, and test activities are performed again. These activities loop until the user is satisfied with the answer to the competency questions.

In this subsection, authors presented the methodology, how the user has to interact with the process in many tasks, the inputs, and outputs of each activity. The next subsection describes an overview of the prototype implementation, through the presentation of some key examples and the technologies that were used.

3.3 Prototype Implementation

To support the methodology explained in the previous subsection a prototype application implementing Document Annotation and Ontology Generation activities is also proposed. This first prototype is a tool that has been developed using the programming language Java, with the Java Virtual Machine version 1.8.0_101 and UIMA Ruta Workbench version 2.6.0.

The architecture of this tool, which is shown in Fig. 3, consists of two main components. The first one is responsible for the tagging and information extraction provided by UIMA Ruta Engine and UIMA Ruta Core. The other component is a developed library containing a set of implemented strategies to handle the tagged document and build the operational ontology.

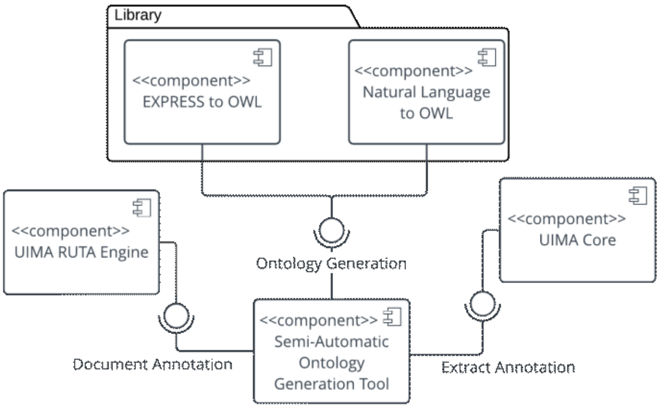


Fig. 3. Proposed Tool Component Diagram

The UIMA Ruta Engine interprets and executes Ruta script files among artifacts, such as documents. The Ruta script files are written using the UIMA Ruta language, which is an imperative rule language extended with scripting elements. These rules define patterns of annotations with conditions where the action of a rule is applied to the matched pattern. The Document Annotation action made use of a UIMA Ruta script file to define the LSP that annotate the terms and all elements related to them and get the annotated document. The UIMA Core library provides the necessary methods to be able to read and extract the annotation from the document.

The other main component is a library that supports the interpretation and process of the document annotations. The mentioned library, which has been developed by the authors, implements the above-mentioned EXPRESS to OWL conversion strategies and techniques for interpreting natural language text annotations to OWL.

As an example of the application of the implemented strategies Fig. 4 illustrates the translation of two terms: *process_or_process_relationship* and *action_method_with_associated_documents*, these two terms were defined within ISO 10303-49 page 7. The upper part of the mentioned figure presents the extracted EXPRESS code block that is used to define the first term in the standard. At the bottom, the definition of the other term is expressed in natural language, as it is specified in the standard. The *process_or_process_relationship* concept represents items that identify a process, a relationship between processes, or a relationship between actions or potential actions that affect a process. The other term refers to documents that define another term called *action_method*, which defines a potential means of satisfying an action.

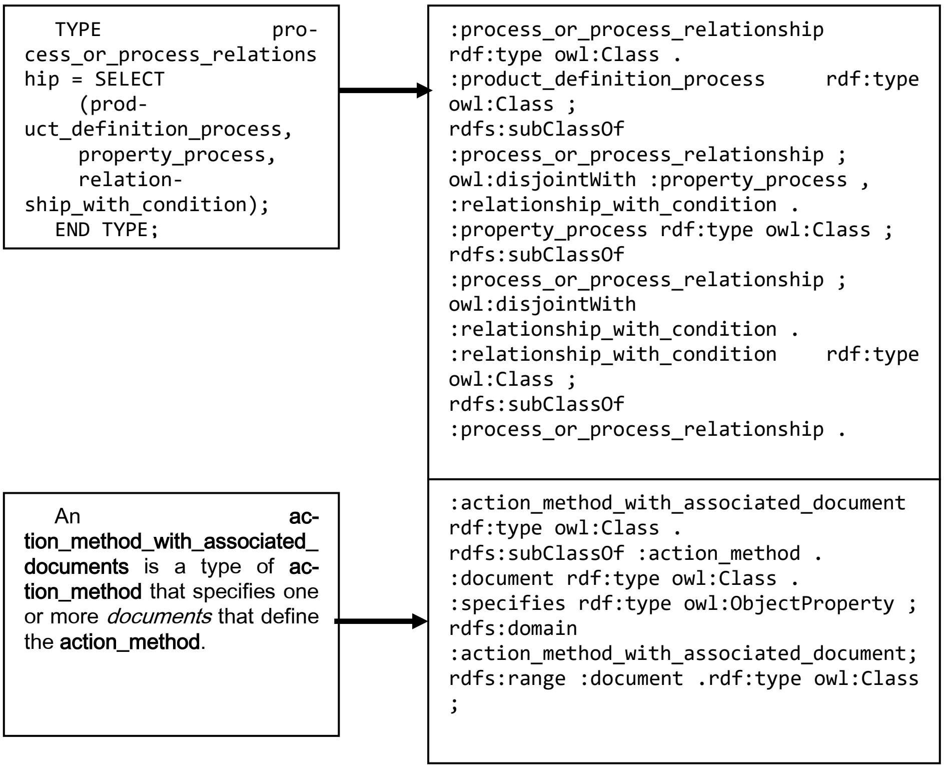
In order to obtain the transformation of the *process_or_process_relationship* term the steps mentioned below has been followed:

I) Once the marked code is obtained from the annotated document of the standard, the tool identifies it an EXPRESS fragment code.

II) The tool processes the fragment and distinguishes the definition of an entity type and the word SELECT with multiple terms.

III) It transforms this fragment into an OWL definition of the entity called `process_or_process_relationship, which` has two subclasses identified as: `product_definition_process` and `rocess_or_process_relationship`. The OWL definition also specifies that these subclasses are disjoint.

The translation process of the term *action_method_with_associated_documents* involves the identification of the tags that the Ontology Annotation activity has added to the standard document. Table 2, illustrates the tags that correspond to the definition of the last-mentioned term.



```
  TYPE              pro-
cess_or_process_relations
hip = SELECT
     (prod-
uct_definition_process,
     property_process,
     relation-
ship_with_condition);
  END TYPE;
```

```
:process_or_process_relationship
rdf:type owl:Class .
:product_definition_process     rdf:type
owl:Class ;
rdfs:subClassOf
:process_or_process_relationship ;
owl:disjointWith :property_process ,
:relationship_with_condition .
:property_process rdf:type owl:Class ;
rdfs:subClassOf
:process_or_process_relationship ;
owl:disjointWith
:relationship_with_condition .
:relationship_with_condition    rdf:type
owl:Class ;
rdfs:subClassOf
:process_or_process_relationship .
```

An **action_method_with_associated_documents** is a type of **action_method** that specifies one or more *documents* that define the **action_method**.

```
:action_method_with_associated_document
rdf:type owl:Class .
rdfs:subClassOf :action_method .
:document rdf:type owl:Class .
:specifies rdf:type owl:ObjectProperty ;
rdfs:domain
:action_method_with_associated_document;
rdfs:range :document .rdf:type owl:Class
;
```

Fig.4. Example two terms from the ISO 10303-49 equivalent codification in OWL

Analyzing the tags in the annotated document, the proposed tool applies a set of rules to transform the tagged concepts into OWL concepts. In particular, for the example shown in Fig. 4, the tool recognizes:

I) The tag *hierarchy* and look for the preceding and succeeding words that are tagged as an *entity* to create the subclass relation.

II) The word tagged as *object_property* and the two words tagged as *entity_domain* and *entity_range* to create the object property specifies and add *ac-*

tion_method_with_associated_document and *document* as domain and range of this object property.

Table 2. Example of tags

Annotated Text	**Tags**
action_method_with_associated_documents	Entity entity_domain
Document	Entity entity_range
action_method	Entity
is a type of	Hierarchy
Specifies	object_property

An example of SWRL rules implemented to align the ontology of ISO 10303-49 is shown in Table 3. The rule whose name is *R_Process* specifies that if X is an individual in the population of the concept *procces_property_schema_product_definition_process*, then X is a *process*. The *R_Resource* rule specifies that if X is *a process_property_schema_action_resource_requirement* and is related to Y through the *operation* object property, then X is a *process_material* and is related to Y through the *usedBy* object property. Finally, the *R_Product* rule points out that if X is a *procces_property_schema_prodyct_association* and is related to Y by the term *process*, and X is a *product* and is associated with Y using the term *producedBy*.

Table 3. SWRL Rules for terms in ISO 10303-49 ontology and Intermediate Level

Name	**Rule**
R_Process	process_property_schema_product_definition_process(?x) => process(?x)
R_Resource	process_property_schema_action_resource_requirement(?x)^operation(?x,?y)=>usedBy(?x,?y)^process_material(?x)
R_Product	process_property_schema_product_association(?x)^process(?x,?y)=>product(?x)^producedBy(?x,?y)

The results were promising as the ontology of the ISO 10303-49 were built by the tool, but limited since not all the standards included in ISO TC184/SC4 have this content configuration and different ones are needed to get results that verify the usefulness of this tool.

4 Conclusions and Future Work

This work presents a methodology that simplifies the task of constructing ontologies of the lower level of the Standard Interoperability Ontology Network. The proposal avoids the construction of the ontology from scratch by reusing the documentation of the standard. A prototype application that partially supports such methodology is also introduced in this article. Such application implements NLP techniques and authors own implementation of the EXPRESS to OWL translation strategies [31] in order to recognize entities and their relationships from the standard documentation files and used them to build an OWL ontology capable of being imported into the network. This tool helps to speed up the ontology building task for documentation of the standard from ISO TC184/SC4.

For future work, the authors will improve the tool to get an automatic information extraction methodology for standards. Leaving aside the steps of selecting the terms and creating the text file and SWRL rules. Also, the authors want to add to the tool a user interface to insert the additional terms that the user would want to search within the standard and enable the ability to see the outcome OWL code and edit them before the tool generates the file.

References

[1] P. Wegner, "Interoperability," *ACM Comput. Surv.*, vol. 28, no. 1, pp. 285–287, 1996.

[2] "European Interoperability Framework." [Online]. Available: http://www.urenio.org/e-innovation/stratinc/files/library/ict/15.ICT_standards.pdf. [Accessed: 10-May-2016].

[3] A. F. Cutting-Decelle, R. I. M. Young, J. J. Michel, R. Grangel, J. Le Cardinal, and J. P. Bourey, "ISO 15531 MANDATE: A Product-process-resource based Approach for Managing Modularity in Production Management," *Concurr. Eng.*, vol. 15, no. 2, pp. 217–235, 2007.

[4] "ISO TC184/SC4." [Online]. Available: http://web.archive.org/web/20071112123812/http://www.tc184-sc4.org/About_TC184-SC4/About_SC4_Standards/. [Accessed: 10-Apr-2017].

[5] A. L. Fraga and H. P. Leone, "Una Arquitectura de Niveles Basada en Ontologías para Lograr la Interoperabilidad entre Estándares," in *45º Jornadas Argentinas de Informática*, 2016, no. 3000.

[6] "ISO 15531-1:2004 - Industrial automation systems and integration -- Industrial manufacturing management data -- Part 1: General overview." [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=28144. [Accessed: 09-May-2016].

[7] "ISO 10303-239:2005 - Industrial automation systems and integration -- Product data representation and exchange -- Part 239: Application protocol: Product life cycle support." [Online]. Available: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=38310. [Accessed: 09-May-2016].

[8] "ISO 10303-232:2002 - Industrial automation systems and integration -- Product data representation and exchange -- Part 232: Application protocol: Technical data packaging core information and exchange." [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=27831. [Accessed: 09-May-2016].

[9] "ISO/TS 15926-11:2015 - Industrial automation systems and integration -- Integration of life-cycle data for process plants including oil and gas production facilities -- Part 11: Methodology for simplified industrial usage of reference data." [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=57859. [Accessed: 09-May-2016].

[10] "ISO 10303-49:1998 - Industrial automation systems and integration -- Product data representation and exchange -- Part 49: Integrated generic resources: Process structure and properties." [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=25101. [Accessed: 09-May-2016].

[11] "ISO 13584-102:2006 - Industrial automation systems and integration -- Parts library -- Part 102: View exchange protocol by ISO 10303 conforming specification." [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=43987. [Accessed: 09-May-2016].

[12] "ISO 10303-1:1994 - Industrial automation systems and integration -- Product data representation and exchange -- Part 1: Overview and fundamental principles." [Online]. Available: http://www.iso.org/iso/catalogue_detail?csnumber=20579. [Accessed: 09-May-2016].

[13] S. Mostefai and A. Bouras, "What ontologies for PLM: A critical analysis," *2006 IEEE Int. Technol. Manag. Conf. ICE 2006*, vol. 3, no. 1, 2016.

[14] A. L. Fraga, M. Vegetti, and H. P. Leone, "Interoperabilidad Semántica entre Estándares de Datos Industriales para Productos," in *4to Congreso Nacional de Ingeniería Informática / Sistemas de Información*, 2016.

[15] A. Al-Arfaj and A. Al-Salman, "Ontology Construction from Text: Challenges and Trends," *Int. J. Artif. Intell. Expert Syst.*, no. 62, pp. 2015–15, 2015.

[16] A. Zouaq, D. Gasevic, and M. Hatala, "Towards open ontology learning and filtering," *Inf. Syst.*, vol. 36, no. 7, pp. 1064–1081, 2011.

[17] P. Cimiano, J. Völker, and R. Studer, "Ontologies on Demand? - A Description of the State-of-the-Art, Applications, Challenges and Trends for Ontology Learning from Text," *Information, Wiss. und Prax.*, vol. 57, no. 6–7, pp. 315–320, 2006.

[18] L. Zhou, "Ontology learning: State of the art and open issues," *Inf. Technol. Manag.*, vol. 8, no. 3, pp. 241–252, 2007.

[19] M. A. Heart, "Automatic Acquisition of Hyponyms from Large Text Corpora Lexico-Syntactic for Hyponymy Patterns," *Int. Conf. Comput. Linguist.*, pp. 23–28, 1992.

[20] J. Cimiano, Philipp and Völker, "Text2Onto: A Framework for Ontology Learning and Data-Driven Change Discovery," *Nat. Lang. Process. Inf. Syst.*, pp. 227–238, 2005.

[21] M. Y. Dahab, H. A. Hassan, and A. Rafea, "TextOntoEx: Automatic ontology construction from natural English text," *Expert Syst. Appl.*, vol. 34, no. 2, pp. 1474–1480, 2008.

[22] X. Jiang and A.-H. Tan, "CRCTOL: A Semantic-Based Domain Ontology Learning System," *J. Am. Soc. Inf. Sci. Technol.*, vol. 61, no. 1, pp. 150–168, 2010.

[23] D. Sánchez, A. Moreno, and L. Del Vasto-Terrientes, "Learning relation axioms from text: An automatic Web-based approach," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5792–5805, 2012.

[24] A. Maedche and S. Staab, "The text-to-onto ontology learning environment," in *Software Demonstration at ICCS-2000-Eight International Conference on Conceptual Structures*, 2000, vol. 38.

[25] S. A. Kripke, "Naming and Necessity," in *Semantics of Natural Language*, D. Davidson and G. Harman, Eds. Dordrecht: Springer Netherlands, 1972, pp. 253–355.

[26] N. Guarino, D. Oberle, and S. Staab, "What is an Ontology?," *Decis. Support Syst.*, pp. 1–17, 2009.

[27] M. Wauer, D. Schuster, and J. Meinecke, "Aletheia: An architecture for semantic federation of product information from structured and unstructured sources," *Proc. 12th Int. Conf. Inf. Integr. Web-based Appl. Serv. - iiWAS '10*, no. January, p. 325, 2010.

[28] D. Ferrucci and A. Lally, "UIMA: An Architectural Approach to Unstructured

Information Processing in the Corporate Research Environment," *Nat. Lang. Eng.*, vol. 10, no. 3–4, pp. 327–348, 2004.

[29] G. Kasneci, M. Ramanath, F. Suchanek, and G. Weikum, "The YAGO-NAGA Approach to Knowledge Discovery," *Acm Sigmod*, vol. 37, no. 4, p. 7, 2008.

[30] "SWRL: A Semantic Web Rule Language Combining OWL and RuleML." [Online]. Available: https://www.w3.org/Submission/SWRL/. [Accessed: 21-Apr-2017].

[31] P. Pauwels and W. Terkaj, "EXPRESS to OWL for construction industry: Towards a recommendable and usable ifcOWL ontology," *Autom. Constr.*, vol. 63, pp. 100–133, 2016.