Translating Natural Language into Ontology

Ryan Ribeiro de Azevedo[1,2], Fred Freitas[2], Rodrigo G. C. Rocha[1,2], Daniel S. Figueredo[1], Silas Cardoso de Almeida[2] and Gabriel de França Pereira e Silva

[1] Computer Science - UAG/Federal Rural University of Pernambuco, Brazil

[2] CIn/Federal University of Pernambuco, Brazil

{rra2,fred,rgcr,gfps}@cin.ufpe.br, silas.sca@gmail.com, jdanielll@icloud.com

Abstract. In this paper, we present a natural language translator for ontologies and ensure that it is a viable solution to the automated acquisition of ontologies and complete axioms, constituting an effective solution for automating the expressive ontology building Process. The translator is based on syntactic and semantic text analysis. The viability of our approach is demonstrated through the generation of descriptions of complex axioms from concepts defined by users and glossaries found at Wikipedia. We evaluated our approach in an experiment with entry sentences enriched with hierarchy axioms, disjunction, conjunction, negation, as well as existential and universal quantification to impose restriction of properties.

Categories and Subject Descriptors: H.2 [**Database Management**]: Languages;

Keywords: Description Logic (DL), Ontology, Ontology Learning, PLN

1. INTRODUCTION

One of the subfields of Ontology that has been standing out the most along the last decade is Ontology Engineering. Its purpose is to create, represent and model knowledge domains, most of which are not trivial, such as Bioinformatics and e-business, among others. However, as pointed out by [Simperl, E and Tempich, C 2009], the task of Ontology Engineering still consumes a big amount of resources even with the exertion of principles, processes and methodologies to create ontologies, which makes it an arduous and onerous task, besides expensive [Gómez-Pérez, A et al 2004]. Thus, new technologies, methods and tools capable of dealing with the technical and economic challenges regarding the construction of ontologies have been made necessary in order to minimize the need of highly specialized personnel and manual efforts required.

As a consequence, a research line that has been increasingly important through the past two decades is the extraction of domain models from text written in natural language, using Natural Language Processing (NLP) techniques. The process of acquiring of a domain model from text and the automated creation of ontologies, for example, by means of making an analysis of a set of texts using NLP techniques is known as Ontology Learning and was first proposed by [Mädche, A. and S. Staab 2001]. Even so, as affirmed by [Zouaq, A 2011], in spite of the increasing interest and efforts taken towards the improvement of Ontology Learning methods based in NLP techniques [Völker, J *et al*., 2010] [Buitelaar, P and Cimiano, P 2008] [Cimiano, P and Völker, J 2005] [Buitelaar, P et al 2005], the notable potential of the techniques and representations available to the learning process of expressive ontologies and complex axioms has not yet been completely exploited, leaving gaps and unanswered questions that need viable and effective solutions. Among them, these stand out [Pease, A 2011][Völker, J et al 2010]:

- There is a considerable amount of tools and *frameworks* of *Ontology Learning* that have been developed aiming at the automatic or semi-automatic construction of ontologies based on structured, semi-structured or unstructured data. Nonetheless, although useful, the majority of these tools used in Ontology Learning are only capable of creating informal or unexpressive ontologies.

- Evaluating the consistency of ontologies automatically: it is necessary that the automatically created ontologies be assessed by the time of their development, minimizing the amount of errors committed by the ones involved in the development phase and verify whether or not the ontology is contradictory and free of inconsistencies.

All the questions and issues abovementioned justify the approach hereby proposed. It is based in a translator, which consists in the utilization of a hybrid method that combines syntactic and semantic text analysis both in superficial and in-depth approaches of NLP. Demonstrating that a translator for creating ontologies that formalizes and codifies knowledge in OWL DL [Horrocks, I. et al 2007] from sentences provided by users is a viable and effective solution to the process of automatic construction of expressive ontologies and complete axioms.

2. THE APPROACH AND EXAMPLE

One of the goals of this work consists in demonstrating that a translator, through the processing of sentences in natural language provided by users, is capable of creating – automatically and according to the discourse interpreted $\mathcal{ALC}$ [Horrocks, I. et al 2007] ontologies with minimal expressivity. An overview of the translator's architecture and function flow diagram are depicted in Figure 1 and described as follows.

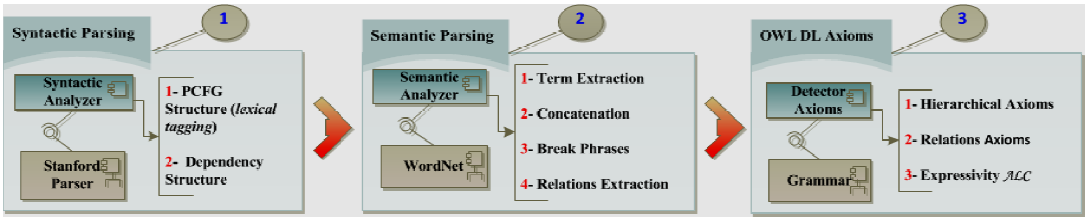

Fig. 1. Translator's architecture and function flow diagram

The architecture of our approach is composed by 3 modules: the **Syntactic Parsing Module (1)**, **Semantic Parsing Module (2)** and the **OWL DL Axioms Module (3)**. The activities executed in the respective modules and their functions are presented in the following sections.

2.1 Módulo Syntactic Parsing

The syntactic analysis of the sentences inserted by users takes place in the **Syntactic Parsing Module (1)**. Two activities are executed by this module, the lexical tagging and the dependence analysis. The results obtained by this module are shown in Fig. 2 and 3**.** We used the sentence **(S1)**: "A self-propelled vehicle is a motor vehicle or road vehicle that does not operate on rails" to illustrate the results obtained by the translator's modules.

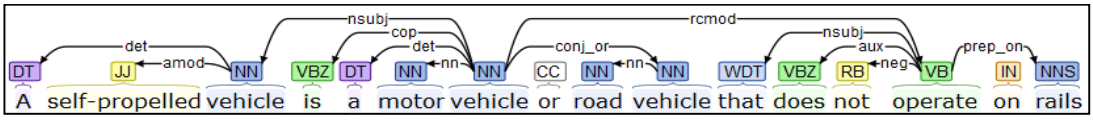

Fig. 2. Lexical tagging and dependence structure

Each word of the sentence (S1) above (Fig. 2) is grammatically classified according to their lexical categories and the dependence between them is attributed.

(**NP** (DT A) (JJ self-propelled) (NN vehicle)) | (**VP** (VBZ is) (NP (DT a) (NN motor) (NN vehicle) | (CC or) (NN road) (NN vehicle)) | (**SBAR** (**WHNP** (WDT that)) | (**VP** (VBZ does) (RB not) (**VP** (**VB** operate) (**PP** (IN on) (**NP** (NNS rails))

Fig. 3. Classification in syntagmatic or sentential categories

Syntagmatic categories are in red and, in black, the lexicon to which each category pertains (See Fig. 3).

2.2 Módulo Semantic Parsing

The results of the activities carried out by the systems of the **Syntactic Parsing Module (1)** are used by the systems of the **Semantic Parsing Module (2)**, which carries out the activities shown in Figure 4 and are detailed as follows.

Fig. 4. Activities carried out in the Semantic Parsing Module

This module initiates its activities by assessing the entry sentence and the referred result of the syntactic analysis obtained in the previous module and then starts the extraction of terms (Term Extraction) that are fit to be concepts of the ontology (Activity (1)). In this phase, terms classified as prepositions (IN), conjunctions (CC), numbers (CD), articles (IN, CC, RB, DT+PDT+WDT) and verbs (EX+MD+VB+VBD+VBG+VBN+VBP+VBZ) are discarded, and the terms classified as nouns (NN+NNS+NNP+NNPS) and adjectives (JJ+JJR+JJS) are indicated as possible concepts of the ontology, therefore, the terms extracted, who are fit to be concepts were: motor/NN, vehicle/NN, road/NN, vehicle/NN, self-propelled/JJ, vehicle/NN e rails/NNS as presented in Fig. 5.

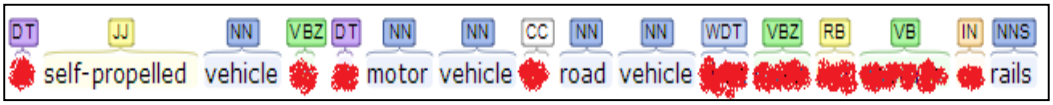

Fig. 5. Result of the extraction of terms

After the term extraction activity is done, the Activity (2), called **Concatenation** is enabled. This activity uses the results of the dependences between the terms (See Figures 2 e 3) and makes the junction of NPs composed by two or more nouns and/ or adjectives inside the analyzed sentence and which, in fact, are related. In the example sentence (*S*1), the concatenation results in the junction of the terms (self-propelled/JJ ↔ vehicle/NN), (motor/NN ↔ vehicle/NN) e (road/NN ↔ vehicle/NN) into an only term, because they are dependent of one another, resulting in just 3 terms: *self-propelled-vehicle*, *motor-vehicle* e *road-vehicle*, and no longer 6 terms, as in the initial phase of the **Term Extraction** activity.

In Activity (3), **Break Phrases,** every time terms or punctuation marks like comma (,), period (.), *and*, *or*, *that, who* or *which* (what we call sentence breakers) are found, the sentences are divided into subsentences and analyzed separately, the result for (S1) was:

A self-propelled-vehicle is a motor-vehicle **| or** road-vehicle **| that** does not operate on rails

The last activity to take place in the **Semantic Parsing Module** is Activity (4), **Relations Extraction**. The relations between the terms are verified and validated through verbs found in the sentences and patterns observed in the translator's inner grammar. The verbs are separated and the terms dependent on verbs are extracted, resulting in:

self-propelled-vehicle **is a** motor-vehicle | self-propelled-vehicle **is a** road-vehicle | self-propelled-vehicle **operate** on rails

This module detects the terms and the relations between them, both hierarchical and nonhierarchical. However, this module neither extracts disjunctions, conjunctions nor generates OWL code

corresponding to the result obtained. The activity of this module is exclusively for detecting terms, their relations and validity.

2.3 Módulo OWL DL Axioms

The function of the **OWL DL Axioms Module** is to symbolically find/learn axioms that prevent ambiguous interpretations and limit the possible interpretations of the discourse, enabling systems to verify and disregard inconsistent data. The process of discovering the axioms is the hardest part of the process of creating ontologies. Here, the axioms discovered correspond to $\mathcal{ALC}$ expressivity. The module recognizes coordinating conjunctions (*OR* and *AND*), labeled CC, indicating the union (disjunction) and intersection (conjunction) respectively for concepts and/or properties, recognizes linking verbs followed by negations, like *does not* and *is not* for negation axioms (¬), besides generating universal quantifiers (∀) and existential quantifiers (∃). It also recognizes *is* and *are* as taxonomic relations (⊑ - hierarchical). The transformations occur in four steps and make use of the results obtained by the previous modules:

Step (1): construction of taxonomic/hierarchical relations. The pattern used here is **<NPs> <VP> <NPs> where <VP> in this case is a (*is a/an*, *is* or *are*).** For all the transformations, the patterns are automatically chosen by the translator.

A self-propelled-vehicle **is a** motor-vehicle → *self-propelled-vehicle* ⊑ *motor-vehicle*
self-propelled-vehicle **is a** road-vehicle → *self-propelled-vehicle* ⊑ *road-vehicle*

Step (2): construction of nonhierarchical relations. The pattern used here is **<NPs> <VP> <NPs> where <VP>** in this case is a verb other than **(is a/an, is or are)**.

self-propelled-vehicle **operate** on rails → self-propelled-vehicle ≡ ∃*operate.rails*

Step (3): verification of conjunctions and disjunctions. The conjunctions OR and AND are verified and analyzed. They can be associated with concepts and/or properties. The pattern **<NPs> *is a/an*** or ***are* <NPs> <CC> <NPs>**, where **<CC>** is the conjunction *Or* or *And* that links two or more **<NPs>** is one of the patterns associated with union and intersections of concepts, and is chosen by the translator resulting in:

A self-propelled-vehicle is a motor-vehicle ***or*** *road-vehicle* → *self-propelled-vehicle* ⊑ *(motor-vehicle* ⊔ *road-vehicle)*

Step (4): detection of negations. The fourth analysis detects the negations, its dependences and classifies the sentence to apply the patterns. Two negations are possible: negations and disjunctions of concepts and negations of properties. Two patterns or a junction of these patterns are taken into consideration in the process of extraction of negation axioms for hierarchies: **<NPs> is not <NPs>** and the pattern **<NP>does not<VP><NP>** for negation of properties. For (S1), the following result was obtained:

self-propelled-vehicle that does not operate on rails → *self-propelled-vehicle* ⊓ ¬∃*operate.rails*

The final result, after the integration of the partial results obtained by the three modules, for (S1) in OWL 2 code, was:

(S1): *A self-propelled vehicle is a motor vehicle or road vehicle that does not operate on rails* → *self-propelled-vehicle* ≡ *(motor-vehicle* ⊔ *road-vehicle)* ⊓ ¬∃*operate.rails*

Our approach generated 4 axioms, 2 of which being hierarchical axioms, 1 being the union between concepts and 1 other of negation of properties. The approach proposed by us is effective in patterns like this and makes correct or approximately correct interpretations of what the user desires.

3. EXPERIMENTS, PRELIMINARY RESULTS AND DISCUSSION

In order to validate the translator, sentences from various knowledge domains were used. The set of data utilized in the experiments contains a total of 120 sentences and in all of them there were negation axioms and in all of them there were negation, conjunction or disjunction axioms and/or two and/or three types of axioms in the same sentence, as well as axioms with definition of terms hierarchy. We opted for sentences found in *Wikipedia* glossaries because they offered in principle a controlled language without syntactic and semantic errors, besides providing a great opportunity for automatic learning. For discussion and comparison purposes, some of the sentences analyzed were extracted from the work of [Völker, J et al 2010], which were also obtained from *Wikipedia*. Some examples of sentences having negation, union and conjunction axioms used in the experiments and the respective results generated by the translator, along with a discussion on these results are shown as follows.

Processed Sentence (1): Juvenile is an young fish or animal that has not reached sexual maturity.
Result: Juvenile ≡ (young-fish ⊔ animal) ⊓ ¬∃hasReached.Sexual-maturity

Discussion: the result of the analysis of the sentence is different from the results of the processing performed by the LExO system [Völker, J et al 2010]: Juvenile ≡ (young ⊓ (Fish ⊔ Animal) ⊓ ¬∃reached.(Sexual ⊓ Maturity). The compared system (LExO) classifies *young*, *fish* and *animal* as distinct terms, however, by the interpretation in natural language of the sentence in analysis, the word *young* is an adjective of the *fish* concept, thus, our approach classifies and represents '*young fish'* as a composite noun, that is, composing a single concept (*young-fish*). The same occurs for s*exual maturity,* being interpreted by LExO as distinct concepts when they are not, whereas in our approach these two terms are classified as a single concept in the same way as the classification of the previous concept (*young-fish*). We can also observe the creation of two axioms, one of union of concepts and one of negation of property..

Processed Sentence (2): A Vector is any agent (person, animal or microorganism) that carries and transmits an infectious pathogen
Result: Vector ≡ Agent ⊓ (∃carriesPathogen.InfectiousPathogen ⊓ ∃transmitsPathogen.InfectiousPathogen)

Discussion: the result obtained in **(2)** was also compared with resulted generated by LExO: Vector ≡ (Organism ⊓ (*carries* ⊔ ∃*transmit.Pathogen*)). The verb *to carry* was not correctly classified as an existential restriction when analyzed by LExO, whereas in our approach, the sentence was coherently classified, the existential quantifier was created and the disjunction of the relations created was performed, where *carriesPathogen* and *transmitsPathogen* are disjoint, which evidences the accurate interpretation of the sentence in natural language.

Sentença Processada (3): The Budget is a list planned expenses and is a list planned revenues.
Resultado: Budget ≡ (List-plannedexpenses ⊓ List-plannedrevenues)

Discussão: : In this sentence, the concept *budget* forms a hierarchy with the other two concepts (*list-plannedrevenues* and *list-plannedexpenses*). Besides, it generates an intersection of both terms, meaning that individuals pertaining to the concept *budget* pertain to the set of individuals of both concepts at the same time.

4. VALIDATION

The objective of planning and performing the experiments was to verify the model constructed and test it in order to answer the following research questions: is the translator capable of constructing minimal-expressivity $\mathcal{ALC}$ ontologies and represent knowledge? And, is the translator capable of identifying rules and axioms inherent to $\mathcal{ALC}$ expressivity based on the texts produced from dialogues

with users? In order to answer these questions, the following hypothesis was formulated to assess the translator's performance during the experiment:

- $H_{a,1}$: The translator constructs the ontology correctly in more than half of the cases;
- $H_{0,1}$: The translator does not construct the ontology in more than half of the cases.

Analyzing all the 120 sentences, we obtained the following results: in 75% of the sentences analyzed (90 sentences), the translator detected and created coherently the axioms, whereas in 30 sentences (25%, also including the ones the translator could not possibly solve in any way), the translator did not detect the axioms coherently and committed errors; however, in 24 out of those 30 sentences, the translator created axioms and made possible the recreation of the ontologies through the proceeding of inserting new definitions. The right unilateral binomial test was applied and the following results obtained.

Limiar → 50% | Level of significance α (5%) | p-valor = 1.886e-08

Sustention: As p-value is lower than the significance of the null hypothesis ($H_{0,1}$: The translator does not construct the ontology in more than half of the cases), it is not accepted, that is, the translator statistically constructs the ontology correctly in more than half of the cases. Using the same binomial test we confirm that this success ratio is statistically superior to 67% (0.67 < success ratio < 1; p-value=0.03636). Therefore, we conclude that the success ratio presented by our approach, 75%, is statistically higher than 67% and, in fact, significant.

5. CONCLUSIONS AND FUTURE WORKS

In this paper, we describe an approach to automatic development of expressive ontologies from definitions provided by users. The results obtained through the experiments evidence the need of automatic creation of expressive axioms, sufficient to creating ontologies with $\mathcal{ALC}$ expressivity, besides the success in the identification of rules and axioms pertaining to $\mathcal{ALC}$ expressivity. We also conclude that the translator can aid both experienced ontology engineers and developers and inexperienced users just starting to create ontologies. As future works, we include the integration of our approach with other existing approaches in the literature, the creation of a module for automatic inclusion of unprecedented patterns in the translator and one module for automatic insertion of individuals for terms of ontologies created by the translator.

REFERENCES

BUITELAAR, P and CIMIANO, P. *Ontology learning and population: Bridging the gap between text and knowledge*. In: Frontiers in Artificial Intelligence and Applications Series (Vol. 167). IOS Press, 2008.

BUITELAAR, P., CIMIANO, P. and MAGNINI, B. *Ontology learning from text: An overview*. In: Buitelaar, P., Cimiano, P., & Magnini, B.(Eds.), *Ontology learning from text: Methods, applications and evaluation* , pp. 3–12. IOS Press, 2005.

CIMIANO, P and VÖLKER, J.,. Text2Onto. *NLDB*, 227–238. 2005.

GÓMEZ-PÉREZ, A, FERNÁNDEZ-LÓPEZ, M and CORCHO, O. *Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. In: Advanced information and knowledge processing, Springer, London, 2004.

MADCHE, A. and STAAB, S. *Ontology learning for the semantic web*. In: IEEE Intelligent Systems, 16(2):72-79, 2001.

PEASE, A. *Ontology: A Practical Guide*. Published by. Articulate Software Press. USA. 2011.

SIMPERL, E and VTEMPICH, C. *Exploring the Economical Aspects of Ontology Engineering synthetic*. In: Handbook on Ontologies, International Handbooks on Information Systems. Springer, Berlin, Germany, pp. 337–358, 2009.

VÖLKER, J. *Learning Expressive Ontologies*. No. 002. In: Studies on the Semantic Web, AKA Verlag / IOS Press, ISBN: 978-3-89838-621-0. 2009.

ZOUAQ, A. *An Overview of Shallow and Deep Natural Language Processing for Ontology Learning*. Chapter 2. Pg 16-37. In: Ontology Learning and Knowledge Discovery Using the Web: Challenges and Recent Advances. IGI Global. EUA . ISBN 978-1-60960-625-1 (hardcover). 2011.

HORROCKS, I. et al. *OWL: a Description-Logic-Based Ontology Language for the Semantic Web*. In: The Description Logic Handbook: Theory, Implementation and Applications. P458-486. Cambridge University Press. 2ed. 2007