Automatically converting and enriching a computational lexicon Ontology for NLP semantic tasks*

Antonio Toral*, Monica Monachini*, Rafael Muñoz†

*Istituto di Linguistica Computazionale, Consiglio Nazionale delle Ricerche, Pisa, Italy {antonio.toral, monica.monachini}@ilc.cnr.it

†Natural Language Processing and Information Systems Group, University of Alicante, Spain rafael@dlsi.ua.es

Abstract

This paper describes the automatic transformation of a Generative Lexicon (GL) based Ontology into OWL, the Semantic Web ontology language. Furthermore, the OWL ontology is automatically enriched by means of a bottom-up procedure that extracts additional semantic information (relationships, features, predicates and quantifier restrictions) from the lexicon. The contribution of this research is two-fold. On one hand, we introduce a methodology for the formalisation of GL ontologies. On the other, we have developed automatic procedures that bring out a formalised, reasoning-capable, and semantically rich ontology, thus suitable for Natural Language Processing semantic tasks.

1. Introduction

Ontologies have gained growing interest from the Natural Language Processing (NLP) community because of their potential in the area. On one hand, some form of ontology (Hirst, 2004) is already present as a backbone in most, if not all, of the large Lexico-Semantic Resources used within the area (e.g. WordNet(Fellbaum, 1998)). On the other, maybe because of the central role they are said to play in the future Semantic Web (Berners-Lee et al., 2001), there is research in progress on applying ontologies to semantic NLP, e.g. (Pease et al., 2002).

OWL is a formal language to building ontologies proposed by the W3C to be used as a standard in Semantic Web applications (Dean and Schreiber, 2004). Its reasoning capabilities and standard nature has encouraged researchers to convert NLP resources into this language. Examples of this are the conversion of WordNet (van Assem et al., 2006) and Mesh (Soualmia et al., 2004).

The current paper deals with the conversion of a computational lexicon ontology into OWL. However, not only the original ontology is converted, but it is enriched with additional semantic relationships which are extracted from the lexicon. The aim of this research is to obtain a formal and semantically rich ontology that could be used to guide semantic NLP tasks through its reasoning capacities.

The rest of the paper is organised as follows. Section 2. introduces the computational lexicon used in this research and the modelling of its ontology in the OWL language. Next, section 3., core of the present research, covers the automatic procedure developed in order to automatically formalise and enrich the input ontology. Last, in section 4. we present some conclusions and future lines.

2. The PAROLE-SIMPLE-CLIPS ontology and its OWL modelling

PAROLE-SIMPLE-CLIPS (PSC) is a computational lexicon which has been developed in the framework of

three different projects. The first two, PAROLE (Ruimy et al., 1998) and SIMPLE (Lenci et al., 2000), were funded by the European Union and were devoted to the research and development of wide-coverage, multipurposed and harmonised computational lexicons for twelve European languages¹. While PAROLE dealt with the morphological and syntactic layers, SIMPLE, a follow-up of the first, added a semantic layer to a subset of the PAROLE data. Last, CLIPS (Ruimy et al., 2002) was an Italian national project in which the Italian lexicon developed within PAROLE and SIMPLE was enlarged and refined. The semantic layer of PSC, which is the relevant one for the current research, contains about 55,000 semantic units² organised in an ontology made up of 153 semantic types.

The main theoretical framework on which the PSC model is based is the Generative Lexicon (GL) theory (Pustejovsky, 1991). An essential feature of the GL is its ability to capture the different dimensions of word meaning. This is possible thanks to the Qualia structure, a GL core module made up of four qualia roles (formal, constitutive, agentive and telic) which express essential aspects of word meaning.

The following subsections briefly describe the elements of PSC that have been considered for the procedures treated in section 3. and how they are formalised in OWL. For a comprehensive explanation of this formalisation, refer to (Toral and Monachini, 2007), in which we analyse in detail this issue.

2.1. Semantic types

The ontology of PSC is made up of nodes called semantic types. There are two types of nodes, simple types, which are identified by only a one-dimensional aspect of meaning (formal) expressed by hyperonymic relations, and unified types, for which additional dimensions of meaning

^{*}We would like to thank Thomas Russ, Kaarel Kaljurand, Waclaw Kusnierczyk and Jane Eisenstein for their valuable help and feedback regarding our OWL doubts and questions.

¹Catalan, Danish, Dutch, English, Finnish, French, German, Greek, Italian, Portuguese, Spanish and Swedish.

²For the current research only a subset of 28,346 semantic units was used since they are specially rich in semantic relationships

(e.g. constitutive) are needed. The top nodes of the ontology are shown in figure 1. These are "Entity" and the three qualia roles that might be involved in the definition of unified types ("Constitutive", "Agentive" and "Telic"). Semantic types are modelled into OWL as ontology classes.

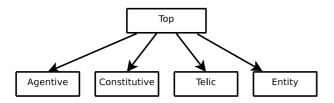


Figure 1: Top nodes of the SIMPLE ontology

2.2. Relations and features

Relations and features are the elements of PSC that allow to assign attributes to lexicon units (also called semantic units or word senses). While relations are used to link two semantic units (e.g. "usedFor" links "bisturi" to "incidere (engrave)"), features allow to link a semantic unit to a value within a closed range (e.g. "Plus_Edible" links "panino (sandwich)" to the value "yes"). Relations are modelled as object properties and features as datatype properties.

2.3. Templates

Each semantic type of the PSC ontology is associated with one template. The latter provides an interface between the lexicon and the ontology by imposing the conditions for the belonging of a lexicon's semantic unit to a semantic type. I.e. a template introduces constraints that its semantic units should satisfy and provides criteria for their well-formedness. E.g. Table 1 shows the constraints present in the template corresponding to the semantic type "Artifact_Food" of the original resource. These constraints are modelled as cardinality restrictions. Table 2 shows the mapping between the possible template contraint values and their equivalent cardinality restriction values in OWL. Table 1 states for example that the relation "Createdby" has the constraint value "RecYes", which according to table 2, will be transtaled into OWL by applying a minimum cardinality constraint equal to 1, i.e. every semantic unit should instantiate at least once this relation.

Table 1: Template for the semantic type Artifact_Food

Item	Type	constraint value
Createdby	relation	RecYes
Madeof	relation	RecNo
Objectoftheactivity	relation	RecYes
Plus_Edible	feature	Yes

2.4. Predicates

Predicates are assigned to the predicative semantic units (verbs, deverbal nouns, etc.) of the lexicon. A predicate is made up of a set of arguments, each of which is linked to a semantic role and to a selectional restriction. A

Table 2: Mapping template constraints to cardinality restrictions

Template constraint value	OWL cardinality value
Yes	min 1, max 1
RecYes	min 1
No	min 0, max 1
RecNo	min 0

selectional restriction can be a semantic type, a semantic unit or a notion, which is a cluster of restrictions combining features and semantic types. The following examples show the predicates for the word senses "guidare (drive)" and "ronzare (whirr)":

- The predicate for "guidare" contains two arguments.
 The first argument has the semantic role "Agent" and has a restriction which is a notion called "ArgHuman-HumanGroup". The second argument has the semantic role "Patient" and has a restriction which is the semantic type "Vehicle".
- The predicate for "ronzare" is made up of only one argument. This has the semantic role "Agent" and a restriction which is the semantic unit "insetto (insect)"

Predicates are modelled in OWL with functional object properties. We have created a property for each of the 15 semantic roles defined in PSC (agent, patient, kinship, beneficiary, etc). Besides, when restrictions are expressed by notions and semantic units, these need to be brought back to the correspondent semantic types. Semantic units become its correspondent semantic type (e.g. the semantic unit "insetto" is assigned the semantic type "Animal"). Regarding notions, we have established an equivalent class for each of them (e.g. the equivalent class for the notion "ArgHumanHumanGroup" is "Human_Group OR Human").

3. Procedure

This section describes the automatic transformation (see 3.1.) and semantic enrichment (see 3.2.) of the PSC ontology. Figure 2 shows an entity-relationship diagram of the PSC tables involved in these tasks. While only the three tables inside the dashed lines are employed for the transformation phase, all of them are used in the enrichment one.

In order to carry out the aforementioned tasks, we have written software that creates an OWL ontology by using the OWL API included in Jena³ (a Java framework for building Semantic Web applications). The input is provided by making queries to the original PSC database. From this database, ontology information was used for the transformation phase whereas mainly lexicon tables were queried for the enrichment step. Finally, in order to visualise and check the consistency of the created OWL ontology we have utilised the Protégé ontology editor with its

³http://jena.sourceforge.net/

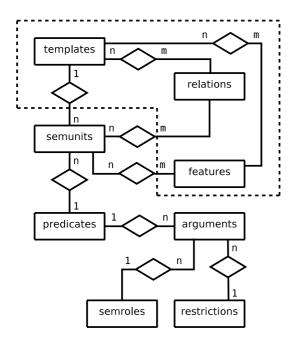


Figure 2: Entity-relationship diagram of the tables queried for the ontology transformation and its enrichment

OWL plugin⁴ (Knublauch et al., 2004) together with two OWL reasoners: FaCT++⁵ and Pellet⁶.

3.1. Transformation

The transformation of the ontology involves translating the different elements that make up the ontology from their original codification as registers of database tables into OWL-compliant expressions. Figure 2 shows the structure of the correspondant database tables (those inside the dashed lines).

Four different elements of the ontology have been identified and transformed by means of an automatic procedure. These are the ontology taxonomy, relations, features and cardinality restrictions that apply to ontology nodes on both relations and features. A detailed explanation about the translation into OWL of each of these ontology elements follows.

The taxonomy of classes is derived from the templates table. First, all the different semantic types are identified and the correspondant OWL classes are created. Next, the taxonomy is built by identifying for each class its direct ascendant and making the latter explicitly the superclass of the first. Finally, all siblings across the class taxonomy are made disjoint.

Relations are extracted from the relations table. As in the case of the templates, a taxonomy has been built for relations. The top nodes are the different relation types present in the PSC model, i.e. four types for the correspondant qualia roles (agentive, constitutive, formal and telic) and others for non-qualia relations (antonym, derivational, metaphor, mentonym, polysemy and synonym). Domain and range are both set to the top node of the ontology for non-qualia relations while for qualia relations both are set

to the ontology classes "Entity" and the class that corresponds to the specific qualia type ("Agentive", "Constitutive", "Formal" or "Telic")

Features are imported from the features and templates tables. Differently than for relations, features form a plain taxonomy, i.e. there are not sub-properties relationships. Templates information is used to establish the domain, as this is defined as the union of classes for which the feature is defined. The range is set to boolean as so far only these kind of features have been imported.

Finally, cardinality restrictions are imported from the three tables depicted in figure 2 inside the dashed lines. They are extracted in a top-down fashion so that the procedure can deal with inheritance. For each class we first check if the current restriction is already inherited from a superclass. Only if it is not, then the restriction is applied. The procedure has found 13 inconsistencies in the ontology database, i.e. restrictions inherited and explicitly encoded with different cardinality values (see table 2). Thus, the procedure has been useful also to check and improve the input resource.

3.2. Enrichment

The enrichment phase extracts from the lexicon further information not present in the original PSC ontology and, in most of the cases, automatically adds it to the OWL ontology. Different kinds of knowledge are extracted this way: quantifier restrictions, predicates and additional features and relations. Next subsections deal with the lexicon extraction and ontology enrichment regarding each of these elements.

3.2.1. Quantifier restrictions

Within an ontology, quantifier restrictions allow to establish, for a restriction applied over a property to a source class, the target class/es of this restriction. E.g. In University of Manchester's pizza ontology⁷, a restriction over the property "hasTopping" is applied to the source class "Pizza" and the target class of this restriction is "MozzarellaTopping".

There are two different types of quantifiers: existential (\exists) and universal (\forall) . An existential restriction describes the set of individuals that, for a given property, have at least one relationship with individuals that are members of the target class. On the other hand, an universal restriction describes the set of individuals that, for a given property, only have relationships with individuals that are members of the target class.

Despite of the fact that the PSC ontology does not contain the semantic types that are the target to a given restriction, this information can inderectly been extracted from the lexicon and, after some generalisation, be used to enrich the ontology.

For a given constraint over a relation that belongs to a template, we extract all the occurrences of the relation in the semantic units that belong to the template's semantic type. These are made up of a source semantic unit that

⁴http://protege.stanford.edu/overview/protege-owl.html

⁵http://owl.man.ac.uk/factplusplus/

⁶http://pellet.owldl.com/

⁷see http://www.co-ode.org/resources/tutorials/ Protege-OWLTutorial.pdf

belongs to the current semantic type and a target semantic unit. I.e. they link two semantic units. E.g. the semantic unit "bisturi (scalpel)" that belongs to the semantic type "Instrument" is linked by the relation "usedBy" to the semantic unit "chirurgo (surgeon)" that belongs to the semantic type "Profession".

For each of these occurrences, we extract the semantic type to which the target semantic unit of the relation belongs. Therefore, we obtain a list of target semantic types. Afterwards, these are generalised in this way: if in the list it is present a semantic type and one ancestor of it, then the descendant semantic type is deleted from the list. For example, there are 47 semantic units in the semantic type "Food" that instantiate the telic relation "Objectoftheactivity", out of which we obtain the target class "Relational_Act".

Regarding the quantifier type, we add an universal restriction to all the constraints while existential restrictions are only applied to that constraints of type "Yes" or "RecYes" as an existential restriction implies a minimum cardinality greater than zero. Following with the previous example, both an existential and an universal quantifier restrictions would be added for the relation "Objectoftheactivity" as its constraint value is "RecYes".

3.2.2. Predicates

Although semantic predicates are not included in PSC at the ontology level, they are defined in the lexicon (see 2.4.). The challenge consists then in establishing generic predicates for the nodes of the ontology of a predicative nature (the "Events" semantic type and its subclasses) by generalising them from the predicates present for the semantic units that belong to these semantic types. Concretely, given a semantic type and a set of predicates (those of the corresponding semantic units), we generalise the selectional restrictions that belong to each of the different predicative semantic roles to one or more semantic types.

A clear parallelism can be established between this issue and that introduced above in 3.2.1. as also here we have to generalise the target of relationships to semantic types. The difference, however, is that the previous case consisted in finding for a set of semantic units the corresponding semantic types whereas in this case not only semantic units need to be translated into semantic types but also notions (a selectional restriction can be a semantic type, semantic unit or a notion). Afterwards, as in 3.2.1. again, a quantifier restriction is introduced over each predicative semantic role relation. The target of the restriction is the semantic type/s result of the generalisation of the gathered set of semantic types.

3.2.3. Additional features and relations

Differently from predicates and quantifier restrictions, relations and features are explicitly considered in the PSC ontology. The templates, in fact, contain constraints over relations and features that the semantic units that belong to them should instantiate. These relations and features were in fact identified and established before than the lexicon population of PSC took place and used to guide this posterior process.

However, we have conducted a study regarding the relations and features present in the semantic units that belong to each of the semantic types of the ontology and have discovered that beyond the relations and features encoded in the corresponding templates, for some semantic types there are further relations and features that could be considered as type-defining if we put it on frequency terms. The following example might clarify this matter: if we take the "Instrument" semantic type, we find that the relation "Usedfor" is included in the template definition and in fact is instantiated by a high percentage of the semantic units that belong to the type (71%). Another relation, "Createdby", is instantiated by a percentage as high as 100% but however is not included in the template definition.

In a nutshell, the procedure consisted on identifying relations and features significant because of their frequency of instantiation but nevertheless not included in the templates. We consider a relation/feature significant if its frequency of instantiation is at least equal to the frequency of the least frequently instantiated relation/feature that is included in the template definition. This way, 218 additional relations and 229 features are extracted and proposed to be included in the ontology.

Anyway, automatically adding these extracted relationships to the ontology could introduce errors. In order to avoid this, we have decided to follow a very conservative approach: a relation/feature is only automatically added to the ontology when it is instantiated for all the semantic units of the semantic type. This way, 70 out of 229 features (30,56%) are automatically included in the ontology while only 1 out of 218 relations (0,45%) is automatically added to the ontology. Furthermore, we check that these relationships are not already present for the given semantic type by inheritance before adding them to the ontology. For example, the only relation that would be automatically added ("Createdby" for the semantic type "Instrument") is not added because "Instrument" already contains this relation as inherits it from its ascendant "Artifact".

4. Discussion

This paper has studied and developed an automatic procedure for the transformation of a Lexico-semantic Resource based in the GL theory into the Semantic Web ontology language. The paper has described the different tasks carried out in order to translate the original ontology into OWL and to enrich this ontology with lexicon's semantic information which has been automatically promoted to the upper ontology level.

Figure 3 shows the asserted conditions present in the node "Artifact_Food" of the output ontology. The figure presents two different areas, the upper one includes the necessary conditions, those specific of the class, whereas in the lower part we find the inherited conditions, those that the current class takes from its superclasses by inheritance. This picture allows us to compare the encoded constraints with the information present in the correspondent template of the original PSC ontology (see table 2). For each relation we can see in the resulting ontology the corresponding cardinality and quantifier restrictions, the latter including target classes extracted from the lexicon. Re-

	NECESSARY	
p1:Food	⊑	
p1:hasCreatedby only (p1:Cause_Change_of_State or p1:Cause_Constitutive_Change or p1:Change_of_State or p1:Purpose_Act) p1:hasCreatedby some (p1:Cause_Change_of_State or p1:Cause_Constitutive_Change or p1:Change_of_State or p1:Purpose_Act)		
🕝 p1:hasMadeof only (p1:Artifactual_drink or p1:Food or p1:Natural_substance or p1:	Substance_food or p1:VegetalEntity)	
🔁 p1:hasMadeof min 0	⊑	
	INHERITED	
🕝 p1:hasObjectoftheactivity only (p1:Relational_Act)	[from p1:Food] ⊑	
p1:hasObjectoftheactivity some (p1:Relational_Act)	[from p1:Food] ⊑	
p1:hasObjectoftheactivity min 1	[from p1:Food] ⊑	
p1:hasPLUS_EDIBLE min 1	[from p1:Food] 🖺	
S p1:hasPLUS_EDIBLE max 1	[from p1:Food] 🖺	
🕝 p1:hasSynonym only (p1:Entity or p1:Part)	[from p1:Entity]	
2 p1:hasSynonym min 0	[from p1:Entity] 🗖	

Figure 3: Asserted conditions for the class "Artifact_Food" in the resulting OWL ontology

garding the only feature present in the original template, "Plus_Edible", the correspondent minimum and maximum cardinality restrictions are shown in the inherited part of the figure as the direct superclass ("Food") introduces as well these constraints and thus there is no need to explicitly repeat the same information for the class "Artifact_Food".

Regarding the enrichment phase, it should be noted that through the automatic procedures developed we obtain a language independent enriched ontology from languagedependent (Italian) lexico-semantic information.

An indirect contribution of this research is that it has been useful to find inconsistencies and to enhance the quality of the computational lexicon employed. An example of the first is the discovery of incongruous cardinality restrictions in the original ontology between explicitly stated restrictions and inherited ones. On the other hand, an example of quality enhancing is the set of proposed additional relations and features which semantically enrich the ontology.

The result of the current research is a semantically rich ontology with reasoning capabilities interfaced to a lexicon. Therefore it is a valuable resource for semantic Natural Language Processing tasks. In fact, this ontology is a key element of a broader future research which is aimed at guiding automatic lexico-semantic Text Mining and Knowledge Acquisition procedures.

5. References

Berners-Lee, Tim, James Hendler, and Ora Lassila, 2001. The semantic web. *Scientific American*, 284(5):34–43.

Dean, Mike and Guus Schreiber, 2004. OWL web ontology language reference. W3C recommendation, W3C.

Fellbaum, C., 1998. WordNet: An Electronic Lexical Database (Language, Speech, and Communication). The MIT Press.

Hirst, Graeme, 2004. Ontology and the lexicon. In Steffen Staab and Rudi Studer (eds.), *Handbook on Ontologies*, International Handbooks on Information Systems. Springer, pages 209–230.

Knublauch, Holger, Ray W. Fergerson, Natalya F. Noy, and Mark A. Musen, 2004. The protege owl plugin: An open development environment for semantic web applications. In *Proceedings of the Third International Semantic Web Conference*.

Lenci, A., N. Bel, F. Busa, N. Calzolari, E. Gola, M. Monachini, A. Ogonowski, I. Peters, W. Peters, N. Ruimy, M. Villegas, and A. Zampolli, 2000. Simple: A general framework for the development of multilingual lexicons. *International Journal of Lexicography*, 13(4):249–263.

Pease, Adam, Ian Niles, and John Li, 2002. The Suggested Upper Merged Ontology: A large ontology for the semantic web and its applications. In *Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web*. Edmonton, Canada.

Pustejovsky, James, 1991. The generative lexicon. *Computational Linguistics*, 17(4):409–441.

Ruimy, N., O. Corazzari, E. Gola, A. Spanu, N. Calzolari, and A. Zampolli, 1998. The european le-parole project:
The italian syntactic lexicon. In *Proceedings of the First International Conference on Language Resources and Evaluation (LREC'98)*. Granada, Spain.

Ruimy, N., M. Monachini, R. Distante, E. Guazzini,
S. Molino, M. Ulivieri, N. Calzolari, and A. Zampolli,
2002. Clips, a multi-level italian computational lexicon:
A glimpse to data. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*. Las Palmas de Gran Canaria, Spain.

Soualmia, Lina Fatima, Christine Golbreich, and Stéfan Jacques Darmoni, 2004. Representing the mesh in owl: Towards a semi-automatic migration. In *KR-MED*.

Toral, Antonio and Monica Monachini, 2007. SIMPLE-OWL: a Generative Lexicon Ontology for NLP and the Semantic Web. In Workshop on Cooperative Construction of Linguistic Knowledge Bases (AIIA 2007).

van Assem, Mark, Aldo Gangemi, and Guus Schreiber, 2006. Conversion of WordNet to a standard RDF/OWL representation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. Genoa, Italy.