PAROLE/SIMPLE 'LexInfo' ontology and lexicons

Marta Villegas and Núria Bel
Universitat Pompeu Fabra

Abstract.

The PAROLE/SIMPLE 'LexInfo' Ontology and Lexicon are the OWL/RDF version of the PAROLE & SIMPLE lexicons (defined during the PAROLE (LE2-4017) and SIMPLE (LE4-8346) IV FP EU projects) once mapped to LexInfo model. Original PAROLE/SIMPLE lexicons contain morphological, syntactic and semantic information, organized according to a common model and to common linguistic specifications for 12 European languages. The data set we describe includes the common PAROLE/SIMPLE model mapped to LexInfo ontology and the Spanish & Catalan lexicons. All data are published in the Data Hub and are distributed under CC Attribution 3.0 Unported licence. The Spanish lexicon contains 199466 triples and 7572 lexical entries fully annotated with syntactic and semantic information. The Catalan lexicon contains 343714 triples and 20545 lexical entries annotated with syntactic information half of which are also annotated with semantic information. In this paper we briefly describe the resulting data, the mapping process and the benefits obtained.

Keywords: lexicon, ontology, open linked data, RDF, OWL, LE-PAROLE, SIMPLE, LexInfo, Lemon

1. Introduction

The PAROLE/SIMPLE 'LexInfo' Ontology is the OWL/RDF version of the PAROLE & SIMPLE model (defined during the PAROLE LE2-4017 and SIMPLE LE4-8346 projects) once mapped to LexInfo model (http://LexInfo.net).

Original PAROLE/SIMPLE lexicons contain morphological, syntactic and semantic information, organized according to a common model and to common linguistic specifications. PAROLE was the first project producing corpora and lexica in so many languages[1] and built according to the same design principles, same linguistic specifications and representation format. The model was based on EAGLES recommendations for morphosyntactic information and verb syntax (Sanfilippo *et al.* 1996[2]) and on the extended GENELEX model.

The goal of SIMPLE project was to add semantic information to the set of harmonized multifunctional lexica built for 12 European languages by the PAROLE consortium. All PAROLE/SIMPLE lexica were defined against a common model defined in the DTD. Thus all PAROLE/SIMPLE Lexica are XML files valid against the same DTD[3]. In addition, a good number of 'descriptive' elements were defined and shared by all SIMPLE lexica. Essentially, these include: (i) Template assignment: meant to guarantee coherent encoding, across sites and languages, (ii) Domain information, (iii) Semantic class information, (iv) Semantic features: distinctive features used to better specify the semantic class of a sense, and for the definition of selectional restrictions on the argu-

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

[2] Sanfilippo et al., 1996 Preliminary Recommendations on Subcategorization
http://www.ilc.cnr.it/EAGLES96/synlex/synlex.html

[3] Original PAROLE/SIMPLE lexicons were in SGML so we converted them into XML.

ments (v) Semantic Roles and (vi) Semantic Relations.

For converting all these into the LexInfo model our first task was to map the original PAROLE/SIMPLE model expressed in the PAROLE DTD into the LexInfo model (http://LexInfo.net/). Thus, the resulting PAROLE/SIMPLE Ontology imports the LexInfo Ontology and adds 'PAROLE elements' (classes and/or properties) whenever these cannot be mapped to any 'LexInfo element'.

In addition, all common descriptive elements defined by SIMPLE project were also included in the Ontology. Thus, the resulting ontology includes classes for Domain and SemanticClass and properties for Semantic Roles and Semantic Relations (defined as sub properties of the lemon:semArg property and lemon:senseRelation respectively).

2. Mapping process: from PAROLE to LexInfo

The strategy followed when mapping PAROLE/SIMPLE model to LexInfo Ontology can be summarized as follows:

(i) Elements from PAROLE/SIMPLE DTD were mapped to Classes. Whenever possible, LexInfo classes were used. Otherwise, new classes were created. For example: PAROLE *Description* elements become lemon:Frames. Note that many PAROLE/SIMPLE elements are not mapped and simply disappear in the target model. This is due to the fact that RDF allows a better modeling.

(ii) Attributes from PAROLE/SIMPLE DTD were mapped to Properties. Again, whenever possible LexInfo properties were used. For example: PAROLE MuS/@gramcat[4] becomes lexinfo:partOfSpeech.

(iii) When the PAROLE/SIMPLE DTD establishes the set of values for a given attribute, these values are mapped to the corresponding LexInfo values. For example: the PAROLE pair: [MuS/@gramcat= NOUN ; MuS/@gramsubcat=COMMON] simply translates as:

... lexinfo:partOfSpeech lexinfo:commonNoun

(iv) Parent/child relations between PAROLE/SIMPLE elements in the DTD were mapped to relevant Properties. For example: the parent/child relation between a PAROLE verbal *Construction* and its subject *InstantiatedPositionC* element become lexinfo:subject property.

(v) IDREFs pointing mechanism in the PAROLE/SIMPE DTD results in relevant property. For example: the relation between PAROLE morphological and syntactic units (MuS & SynUs) encoded in MuS/@synulist attribute is expressed by means of the lemon:synBehaviour property.

3. Some benefits: syntax/semantic linking.

LexInfo model simplifies the original PAROLE/SIMPLE model in a good number of aspects. This is partly due to the use of RDF which allows for a more compact and efficient representation. The case of syntax/semantic mappings is particularly interesting. The original PAROLE/SIMPLE data includes a complex machinery to define syntactic subcategorization frames and semantic argument structures. In the former case, we have to deal with a large set of related elements: *SynU, Description, Construction, Self, InstantiatedPositionC, PositionC, SyntagmaNT*, etc. The relation among these elements is established by means of the parent/child relation mechanism or ID/IDREF pointing mechanism. In the case of argument structure representation, things are also complex and, again, we find a good number of elements involved: *PredicativeRepresentation, Predicate, Argument, InfArg, SemanticRole*, etc.

Syntax semantic linking in the PAROLE/SIMPLE model is also complex and, in most cases, useless. LexInfo allows defining all these things in a much easier way, essentially:

- *Description*, *Construction* & *Self* elements are mapped to Lexinfo:Frame class and related to relevant entry by means of the lemon: synBehaviour property.
- *InstantiatedPositionC*, *Position* & *Syntagmas* are mapped to lemon:Argument class and related to the relevant lexinfo:Frame via a lemon:synArg relation.
- *PredicativeRepresentation* & *Predicate* are also mapped to Lexinfo:Frame
- *Argument*, *SemanticRole* & *InfArg* are mapped to lemon:Argument class and related to relevant lexinfo:Frame via lemon:semArg relation.

A simplified entry for the English verb 'write' can be found in *Figure 1*. *Figure 2* gives a partial graphical representation. There we can see that both the syntactic frame and the lexical sense point to ARG0 and ARG1 instances. In the former case, the frame

[4] We use XPath expressions when referring to original PAROLE/SIMPLE elements.

links to its arguments by means of 'subject' and 'object' properties. In the latter, case the lexical sense links to its arguments by means of 'agent' and 'patient' properties. Finally, arguments are also specified for semantic template (Human & SemioticArtifact respectively) and syntactic realization (NP in both cases).

```
## Lexical Entry
lex:write  a  lemon:LexicalEntry ;
           lexinfo:partOfSpeech  lexinfo:mainVerb ;
           lemon:synBehavior  lex:write_transitive;
           lemon:sense  lex:write_SymbolicCreation .

## Lexical Senses
lex:write_SymbolicCreation a lemon:LexicalSense ;
           parole:template  parole:SymbolicCreation ;
           parole:roleAgent  lex:write_ARG0 ;
           parole:rolePatient  lex:write_ARG1 .

## Frames
lex:write_transitive  rdf:type  owl:Thing ;
           rdf:type  lex:Transitive ;
           lexinfo:subject  lex:write_ARG0;
           lexinfo:directObject  lex:write_ARG1 .

## Argument info
lex:write_ARG0  a  lexinfo:Subject ;
           lemon:constituent lex:NP
           parole:template  parole:Human .
lex:write_ARG1  a  lexinfo:DirectObject ;
           lemon:constituent lex:NP
           parole:template  parole:SemioticArctifact .
```

Figure 1 Simplified entry for English verb write

4. Some benefits: subcategorization frames

Each original PAROLE lexicon defines the set of subcategorization frames for a particular language. Contrary to semantic descriptions, syntactic descriptions are essentially language dependent. Thus, whereas all lexicons share the same set of semantic descriptive elements (domain, semantic class, semantic relations, etc) such a homogeneity was not defined in the syntactic layer. This means that subcategorization information cannot be easily shared among the lexica.

Basically, this is due to the fact that PAROLE aimed at being a flexible model to accommodate different approaches. This is welcome but proves problematic when addressing interoperability among resources. LexInfo defines a subcategorization ontology based on the Lemon model. Lemon includes the notion of 'frame'. Frames are indicated with the 'synBehaviour' property and their arguments with the property 'synArg'. LexInfo defines subproperties of 'synArg' to represent the syntactic functions of arguments and organizes frames into subclasses. Our mapping to LexInfo implied mapping PAROLE subcategorization frames into this model (*Description* elements and their 'descendents'). The mapping process was done in two steps. First, we defined a style sheet converter that reads our PAROLE XML lexicon and for each *Description* element it generates a new 'frame'. Consequently, all newly created frames were treated as subclasses of the general Lemon Frame. Second, we collapsed some frames into one single class[5], thus simplifying the model, and organized them into the LexInfo ontology. As a result, the PAROLE ontology becomes lighter than the original model and allows queries that were otherwise impossible in the original PAROLE lexicon; for instance we can easily get all 'control' verbs; verbs with a sentential complement; verbs with an indirect object, etc.

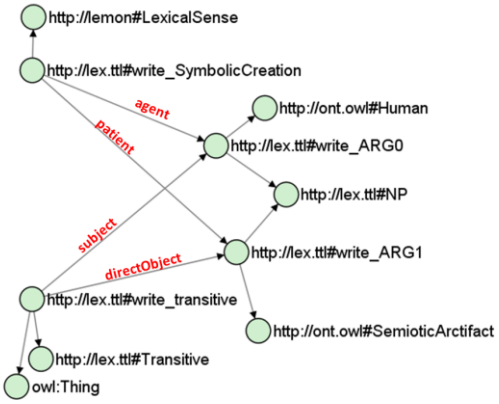

Figure 2 Simplified syn/sem linking

5. Some benefits: exploitation

The most difficult problem of the original PAROLE/SIMPLE lexica is their exploitation and management. When moving from the original PAROLE/SIMPLE model to a relational database, we end up with a complex database with a huge number of related tables[6]. Having PAROLE/SIMPLE lexicons in a database means managing lots of tables

[5] For example, the original Spanish lexicon includes 12 intransitive prepositional *Descriptions*, one for each bounded preposition. All these frames are mapped to IntransitivePP Frame as the information about preposition is encoded by means of a property attached to the PP argument.

[6] Our PAROLE/SIMPLE database included 223 tables.

and very often we need to split complex queries into several sub queries. Note, for example, that getting the senses of a given lemma is not easy and we need a complex SQL query involving up to four different tables. Similarly, a query such as "give me the lemma and the template of all senses with a negative connotation" is a real challenge in the original PAROLE/SIMPLE lexica. Such a query is quite simple in RDF as shown in *Figure 3*. The results are given in *Figure 4*.

```
SELECT ?form ?template  WHERE {
    ?entry lemon:form [ lemon:writtenRep ?form ] .
    ?entry lemon:sense ?sense.
     ?sense  parole:connotation parole:Negative .
     parole:template ?template.
     }
```

Figure 3 SPARQL sample query

Results

form	template
coco	parole:TemplLivingentity
finolis	parole:TemplHuman
asno	parole:TemplHuman
animal	parole:TemplHuman
animal	parole:TemplHuman
gentuza	parole:TemplHumanGroup
pomposidad	parole:TemplQuality
escasez	parole:TemplAmount
escondrijo	parole:TemplLocation
promiscuidad	parole:TemplQuality
cabrón	parole:TemplHuman
escabrosidad	parole:TemplQuality
tullido	parole:TemplHuman
desmemoriado	parole:TemplHuman
estrechez	parole:TemplQuality
alarmar	parole:TemplExperienceEvent
inconsistencia	parole:TemplQuality
pereza	parole:TemplPsychproperty
escepticismo	parole:TemplPsychproperty
hastiar	parole:TemplExperienceEvent
monstruo	parole:TemplHuman
monstruo	parole:TemplLivingentity
fracasado	parole:TemplHuman
drama	parole:TemplAbstractEntity

Figure 4 Results

6. The sources

The Ontology and the Spanish and Catalan lexica are distributed under CC Attribution 3.0 Unported licence. These datasets are published in the Data Hub (http://datahub.io/dataset/parole-simple-ont) and can be downloaded in XML RDF format and/or RDF Turtle format.

7. Statistics

The Spanish lexicon contains 199,466 triples and 7,572 lexical entries fully annotated with syntactic and semantic information.

The Catalan lexicon contains 343,714 triples and 20,545 lexical entries annotated with syntactic information half of which are also annotated with semantic information.

Acknowledgements

The resources reported in this paper were developed with the support of METANET4U: Enhancing the European Linguistic Infrastructure, (2011-2013), funded by UNER - Competitiveness and Innovation Framework Program, (CIP-PSP-270893).

We thank the Institut d'Estudis Catalans and the GilcUB from the University of Barcelona as the creators of the original Catalan and Spanish lexicons.

References

M.H. Antoni-Lay, Gil Francopoulo, L Zaysser. 1994. A General Model for Reusable lexicons: The Genelex Project. In *Literary and Linguistic Computing*, 9(1)847-54).

Paul Buitelaar, Philipp Cimiano, Peter Haase, Michael Sintek: Towards Linguistically Grounded Ontologies. ESWC 2009: 111-125

Philipp Cimiano, Paul Buitelaar, John McCrae., & Michael Sintek, M. (2011). LexInfo: A Declarative Model for the Lexicon-Ontology Interface. *Web Semantics: Science, Services and Agents On The World Wide Web, 9*(1).

GENELEX Consortium (1994), *Report on the Semantic Layer*, Project EUREKA GENELEX, Version 2.1, September 1994.

Gil Francopoulo, Núria Bel, Monte George, Nicoletta Calzolari, Monica Monachini, M Pet, Claudia Soria. 2007 Lexical Markup Framework: ISO standard for semantic information in NLP lexicons. GLDV (Gesellschaft für linguistische Datenverarbeitung), Tubingen.

Guimier, E., Ogonowski, A., 1998. PAROLE Report on the Syntactic Layer. http://www.ub.edu/gilcub/SIMPLE/reports/parole/parole_syn/parosyn.html

Alessandro Lenci et al 2000. SIMPLE Linguistic Specifications Deliverable D2.1. http://www.ub.edu/gilcub/SIMPLE/simple.html#Papers