

An Open-Source Natural Language Generator for OWL Ontologies and its Use in Protégé and Second Life

Dimitrios Galanis*, George Karakatsiotis*, Gerasimos Lampouras*, Ion Androutsopoulos*⁺

*Department of Informatics, Athens University of Economics and Business, Athens, Greece

⁺Digital Curation Unit, Research Centre “Athena”, Athens, Greece

Abstract

We demonstrate an open-source natural language generation engine that produces descriptions of entities and classes in English and Greek from OWL ontologies that have been annotated with linguistic and user modeling information expressed in RDF. We also demonstrate an accompanying plug-in for the Protégé ontology editor, which can be used to create the ontology’s annotations and generate previews of the resulting texts by invoking the generation engine. The engine has been embedded in robots acting as museum tour guides in the physical world and in Second Life; here we demonstrate the latter application.

1 Introduction

NaturalOWL (Galanis and Androutsopoulos, 2007; Androutsopoulos and Galanis, 2008) is a natural language generation engine that produces descriptions of entities (e.g., items for sale, museum exhibits) and classes (e.g., types of exhibits) in English and Greek from OWL DL ontologies; the ontologies must have been annotated with linguistic and user modeling annotations expressed in RDF.¹ An accompanying plug-in for the well known Protégé ontology editor is available, which can be used to create the linguistic and user modeling annotations while editing an ontology, as well as to generate previews of the resulting texts by invoking the generation engine.²

NaturalOWL is based on ideas from ILEX (O’Donnell et al., 2001) and M-PIRO (Isard et al., 2003; Androutsopoulos et al., 2007), but it uses

¹See <http://www.w3.org/TR/owl-features/> for information on OWL and its versions. For information on RDF, consult <http://www.w3.org/RDF/>.

²M-PIRO’s authoring tool (Androutsopoulos et al., 2007), now called ELEON (Bilidas et al., 2007), can also be used; see <http://www.iit.demokritos.gr/skel/>.



Figure 1: Generating texts in Second Life.

templates instead of systemic grammars, it is publicly available as open-source software, it is written entirely in Java, and it provides native support for OWL ontologies, making it particularly useful for Semantic Web applications (Antoniou and van Harmelen, 2004).³ Well known advantages of natural language generation (Reiter and Dale, 2000) include the ability to generate texts in multiple languages from the same ontology; and the ability to tailor the semantic content and language expressions of the texts to the user type (e.g., child vs. adult) and the interaction history (e.g., by avoiding repetitions, or by comparing to previous objects).

In project XENIOS (Vogiatzis et al., 2008), NaturalOWL was embedded in a mobile robot acting as a museum guide, and in project INDIGO it is being integrated in a more advanced robotic guide that includes a multimodal dialogue manager, facial animation, and mechanisms to recognize and express emotions (Konstantopoulos et al., 2009). Here, we demonstrate a similar application, where NaturalOWL is embedded in a robotic avatar acting

³NaturalOWL comes with a GNU General Public License (GPL). The software can be downloaded from <http://nlp.cs.aueb.gr/>.

as a museum guide in Second Life (Oberlander et al., 2008), as shown in figure 1. We also demonstrate how the underlying ontology of the museum and its linguistic and user modeling annotations can be edited in Protégé.

2 NaturalOWL’s architecture

NaturalOWL adopts a typical natural language generation pipeline (Reiter and Dale, 2000). It produces texts in three stages: document planning, microplanning, and surface realization.

In document planning, the system first selects from the ontology the logical facts (OWL triples) that will be conveyed to the user, taking into account interest scores manually assigned to the facts via the annotations of the ontology, as well as a dynamically updated user model that shows what information has already been conveyed to the user. Logical facts that report similarities or differences to previously encountered entities may also be included in the output of content selection, giving rise to comparisons like the one in figure 1. The selected facts are then ordered using a manually specified partial order, which is also part of the ontology’s annotations.

In micro-planning, the system turns each selected fact into a sentence by using micro-plans, in effect patterns that leave referring expressions underspecified. Figure 2 shows a micro-plan being edited with NaturalOWL’s Protégé plug-in. The micro-plan specifies that to express a fact that involves the *made-of* property, the system should concatenate an automatically generated referring expression (e.g., name, pronoun, definite noun phrase) in nominative case for the owner of the fact (semantic subject of the triple), the verb form “is made” (or “are made”, if the subject is in plural), the preposition “of”, and then another automatically generated referring expression in accusative case for the filler of the property (semantic object). The referring expressions are generated by taking into account the context of each sentence, attempting to avoid repetitions without introducing ambiguities. Domain-independent aggregation rules are then employed to combine the resulting sentences into longer ones.

In surface realization, the final form of the text is produced; it can be marked up automatically with tags that indicate punctuation symbols, grammatical categories, the logical facts expressed by the sentences, the interest (*Int*) of each sen-

tence’s information, the degree (*Assim*) to which the information is taken to be assimilated by the user etc., as shown below. In INDIGO, comparisons are also marked up with angles that guide the robot to turn to the object(s) it compares to.

```
<Period>
  <Sentence Property=".../#type"
  Int="3" Assim="0">
    <Demonstrative ref=".../#exhibit1"
    role="owner">
      This</Demonstrative>
    <Verb>is</Verb>
    <NP ref=".../#Amphora" role="filler">
      an amphora</NP>
  </Sentence>
  <Punct>,</Punct>
  <Sentence Property=".../#subtype
  Int="3" Assim="1">
    <EmptyRef ref=".../#Amphora"
    role="owner"/>
    <NP ref=".../#Vessel" role="filler">
      a type of vessel</NP>
  </Sentence>
  <Punct>;</Punct>
  <Sentence Property=".../#paintedBy"
  Int="2" Assim="0">
    <Pronoun ref=".../#exhibit1"
    role="owner">
      it</Pronoun>
    <Verb>was painted</Verb>
    <Preposition>by</Preposition>
    <Name ref=".../#pKleo" role="filler">
      the painter of Kleophrades</Name>
  </Sentence>
  <Punct>.</Punct>
</Period>
```

2.1 Using NaturalOWL’s Protégé plug-in

NaturalOWL’s plug-in for Protégé can be used to specify all the linguistic and user modeling annotations of the ontologies that NaturalOWL requires. The annotations in effect establish a domain-dependent lexicon, whose entries are associated with classes or entities of the ontology; micro-plans, which are associated with properties of the ontology; a partial order of properties, which is used in document planning; interest scores, indicating how interesting the various facts of the ontology are to each user type; parameters that control, for example, the desired length of the generated texts. The plug-in can also be used to generate previews of the resulting texts, for different types of users, with or without comparisons, etc., as illustrated in figure 3. The resulting annotations are then saved in RDF.

2.2 Using NaturalOWL in Second Life

In Second Life, each user controls an avatar, which can, among other actions, move in the virtual world, touch objects, or communicate with other

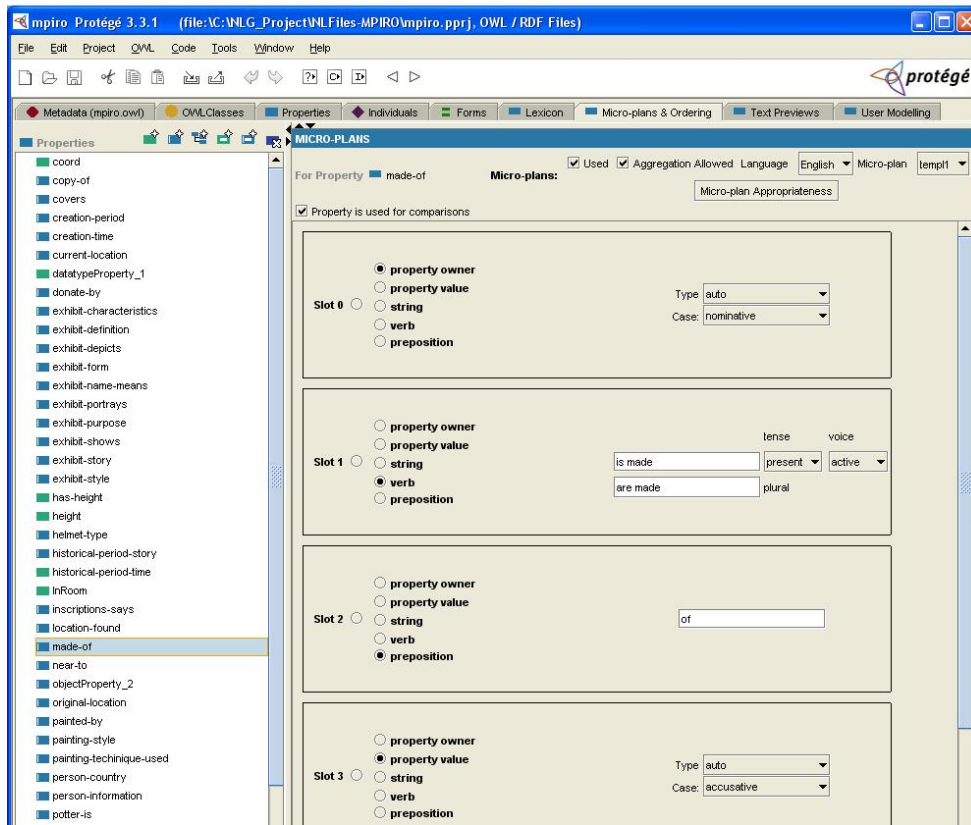


Figure 2: Specifying a micro-plan with NaturalOWL's Protégé plug-in.

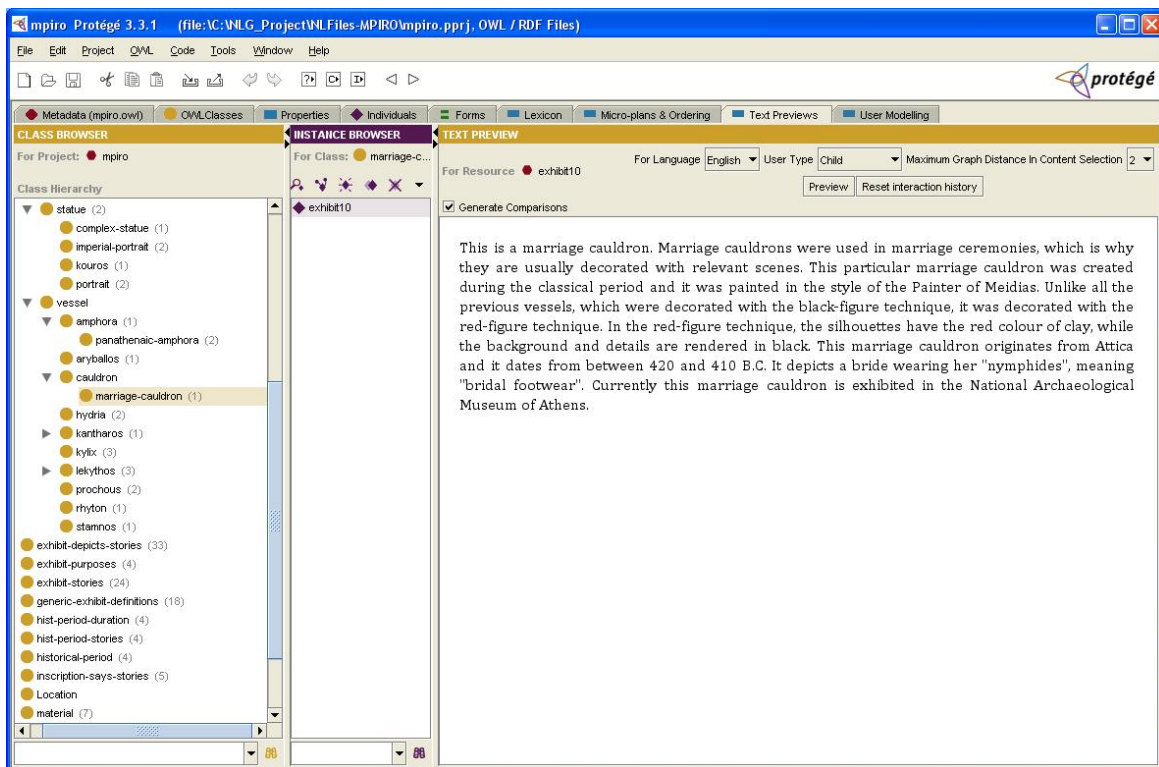


Figure 3: Generating a text preview with NaturalOWL's Protégé plug-in.

avatars; in the latter case, the user types text on the keyboard. In the Second Life application that we demonstrate, the robot is an avatar that is not controlled by a human, but by our own Second Life client software.⁴ The client software includes a navigation component, which controls the robot’s movement, and it allows the robot to “utter” texts generated by NaturalOWL, instead of expecting keyboard input. Whenever a visitor near the robot touches an exhibit, an appropriate event is sent to the robot, which then goes near the exhibit and starts describing it.⁵

3 Conclusions and further work

The demonstration presents an open-source natural language generation engine for OWL ontologies, which generates descriptions of entities and classes in English and Greek. The engine is accompanied by a Protégé plug-in, which can be used to annotate the ontologies with linguistic and user modeling information required by the generation engine. The demonstration includes an application in Second Life, where the generation engine is embedded in a robotic avatar acting as a museum guide. We are currently extending NaturalOWL to handle follow up questions about entities or classes mentioned in the generated texts.

Acknowledgments

NaturalOWL was developed in project XENIOS, which was funded by the Greek General Secretariat of Research and Technology and the European Union.⁶ NaturalOWL is now being extended in project INDIGO, which is funded by the European Union; our work in INDIGO is also supported by additional funding from the Greek General Secretariat of Research and Technology.⁷

References

I. Androutsopoulos and D. Galanis. 2008. Generating natural language descriptions from OWL ontologies: experience from the NaturalOWL system. Technical report, Department of Informatics, Athens University of Economics and Business, Greece.

I. Androutsopoulos, J. Oberlander, and V. Karkaletsis. 2007. Source authoring for multilingual generation of personalised object descriptions. *Natural Language Engineering*, 13(3):191–233.

G. Antoniou and F. van Harmelen. 2004. *A Semantic Web primer*. MIT Press.

D. Bilidas, M. Theologou, and V. Karkaletsis. 2007. Enriching OWL ontologies with linguistic and user-related annotations: the ELEON system. In *Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence*, Patras, Greece.

D. Galanis and I. Androutsopoulos. 2007. Generating multilingual descriptions from linguistically annotated OWL ontologies: the NATURALOWL system. In *Proceedings of the 11th European Workshop on Natural Language Generation*, pages 143–146, Schloss Dagstuhl, Germany.

A. Isard, J. Oberlander, I. Androutsopoulos, and C. Matheson. 2003. Speaking the users’ languages. *IEEE Intelligent Systems*, 18(1):40–45.

S. Konstantopoulos, A. Tegos, D. Bilidas, I. Androutsopoulos, G. Lampouras, P. Malakasiotis, C. Matheson, and O. Deroo. 2009. Adaptive natural-language interaction. In *Proceedings of 12th Conference of the European Chapter of the Association for Computational Linguistics (system demonstrations)*, Athens, Greece.

J. Oberlander, G. Karakatsiotis, A. Isard, and I. Androutsopoulos. 2008. Building an adaptive museum gallery in Second Life. In *Proceedings of Museums and the Web*, Montreal, Quebec, Canada.

M. O’Donnell, C. Mellish, J. Oberlander, and A. Knott. 2001. ILEX: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3):225–250.

E. Reiter and R. Dale. 2000. *Building natural language generation systems*. Cambridge University Press.

D. Vogiatzis, D. Galanis, V. Karkaletsis, I. Androutsopoulos, and C.D. Spyropoulos. 2008. A conversant robotic guide to art collections. In *Proceedings of the 2nd Workshop on Language Technology for Cultural Heritage Data, Language Resources and Evaluation Conference*, Marrakech, Morocco.

⁴Our client was built using the libsecondlife library; see <http://www.libsecondlife.org/>. More precisely, the robot is an object controlled by an invisible robotic avatar, which is in turn controlled by our client.

⁵A video showing the robotic avatar in action is available at <http://www.vimeo.com/801099>.

⁶See <http://www.ics.forth.gr/xenios/>.

⁷See <http://www.ics.forth.gr/indigo/>.