

The Protégé Ontoling Plugin: Linguistic Enrichment of Ontologies in the Semantic Web

Maria Teresa Pazienza and Armando Stellato

AI Research Group, Dept. of Computer Science, Systems and Production
University of Rome, Tor Vergata
Via del Politecnico 1, 00133 Rome, Italy
{pazienza,stellato}@info.uniroma2.it

Abstract

This paper presents our approach in establishing a framework for semi-automatic linguistic enrichment of ontologies. This effort has been pursued through the definition and implementation of a set of general purpose API for accessing a wide variety of linguistic resources and through the development of Ontoling, a plug-in for the popular ontology development tool Protégé, which allows for browsing of different linguistic resources and for using their information to enrich the formal content of ontologies. We describe here the features and design aspects which characterize its current release

1 Introduction

Knowledge Sharing is a crucial issue in the Semantic Web: SW services expose and share knowledge content which arise from distinct languages, locales, and personal perspectives; a great effort has been spent in these years, in the form of Knowledge Representation standards and communication protocols, with the objective of acquiring semantic consensus across distributed applications. However, neither ontology mapping algorithm nor knowledge mediator agent can easily find a way through ontologies as they are organized nowadays: concepts expressed by hardly recognizable labels, lexical ambiguity represented by phenomena like synonymy and polysemy and use of different natural languages which derive from different

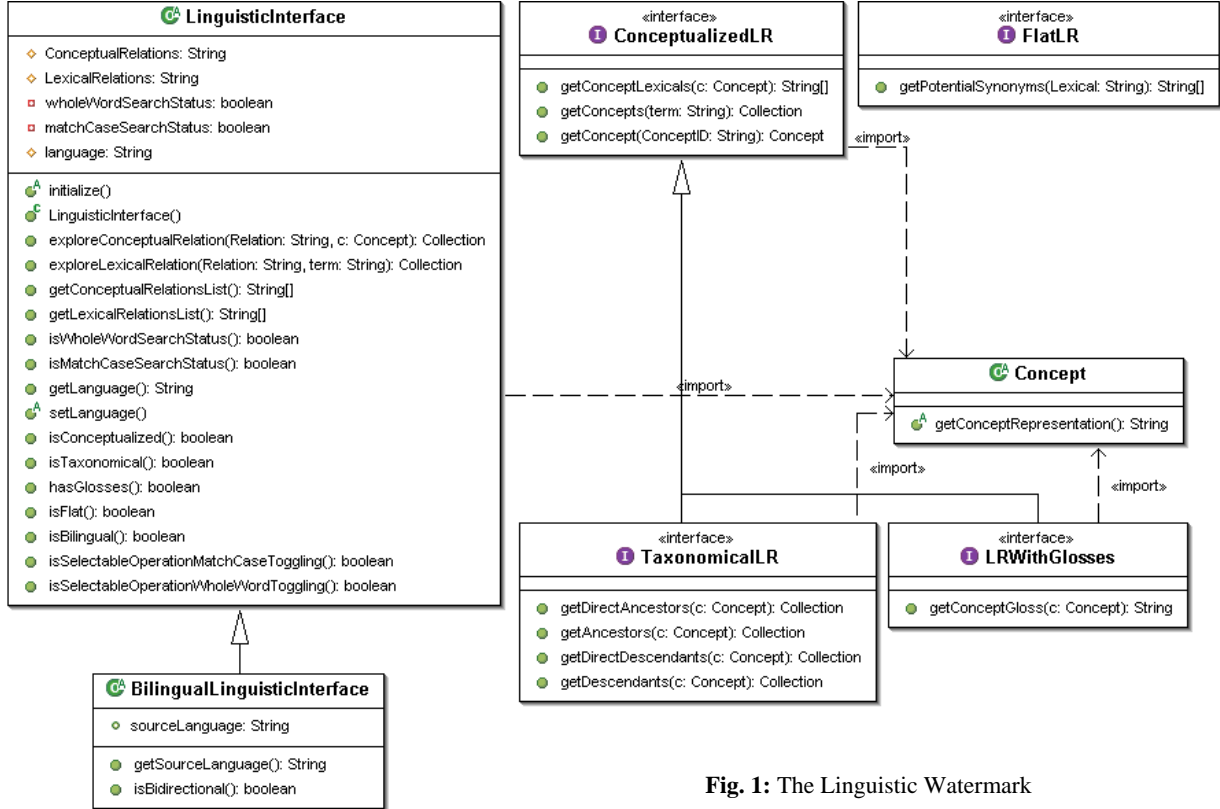


Fig. 1: The Linguistic Watermark

cultures, all together push for expressing ontological content in a linguistically motivated fashion. In this paper we introduce our work in establishing a framework for semi-automatic linguistic enrichment of ontologies, which has run through the identification of different categories of linguistic resources and in planning their exploitation to augment the linguistic expressivity of ontologies. This effort has lead to the realization of the *Linguistic Watermark*, a series of API for characterizing diverse Linguistic Resources and for providing uniform access to their content, and to the development of Ontoling, a plugin for the popular ontology editing tool Protégé (Gennari *et al.*, 2003) which exploits this content in order to linguistically enrich Semantic Web ontologies.

2 The Linguistic Watermark

We introduced the notion of Linguistic Watermark, as the series of characteristics and functionalities which distinguish a particular resource inside our framework. As we can observe from the Class Diagram in Fig. 1, we sketched a sort of ontology of linguistic resources (LRs), with the addition of operational aspects. LRs are in fact structured and described in terms of their features and how their lexical information is organized; the ontology has then been completed with query methods for accessing resource's content. We thus implemented this operational ontology as a java package on its own, providing API which can externally be imported by any application willing to exploit natural language resources like lexicons and terminologies. The core of the package is composed of an Abstract Class, named *LinguisticInterface*, which is both the locus for a formal description of a given linguistic resource and a service-provider for exposing the resource specific methods. The other abstract classes and interfaces in the package, which can be implemented or not, depending on the profile of the resource being wrapped, provide instead the signatures for known interface methods.

We have currently developed several implementations of the Linguistic Watermark. Two of them, the Wordnet Interface (Fellbaum, 1998) and the DICT (Dict site) In-

terface for Bilingual Dictionaries, being related to freely available LRs, have been made publicly available on the Ontoling site:

<http://ai-nlp.info.uniroma2.it/software/OntoLing>

3 The Protégé Ontoling Plugin

Ontoling exploits the API provided by the Linguistic Watermark (following the architecture which is depicted in Fig. 2), allowing for linguistic enrichment of ontologies inside the Protégé working environment. This objective is achieved through a series of functionalities for browsing the loaded LR:

- Search term definitions (glosses)
- Ask for synonyms
- Separate different sense of the same term
- Explore genus and differentia (where available)
- Explore resource-specific semantic relations

as well as some others for ontology editing:

- Add synonyms (or translations, for bilingual resources) as additional labels for identifying concepts
- Add glosses to concepts description (documentation)
- Use notions from linguistic resources to create new concepts (or whole new ontology taxonomy branches)

These functionalities go beyond their original intent and can be helpful for Ontologists and Knowledge Engineers in creating new ontologies or in improving/modifying existing ones. Ontoling currently supports both Protégé original model, relying on a dedicated meta-ontology for representing synonyms, translations and glosses, and the OWL Model, which offers as it is support for multiple languages and comments in the form of *rdf:labels* and *rdf:comments* meta properties.

4 Future Work

Ongoing and future work is pursuing the direction of further automatizing this process, in order to reduce human effort to a fully supervised methodology for linguistic enrichment. To this end, adaptive algorithms need to take into account the different characteristics which may be exposed through the Linguistic Watermark and provide an harmonizing view over the linguistic enrichment process.

References

[Fellbaum, 1998] C. Fellbaum.: *WordNet - An electronic lexical database*. MIT Press, 1998.

[Gennari *et al.*, 2003] J. Gennari, M. Musen, R. Ferguson, W. Grosso, M. Crubézy, H. Eriksson, N. Noy, and S. Tu. The evolution of Protégé-2000: An environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89–123, 2003.

[Dict site] www.dict.org/bin/Dict

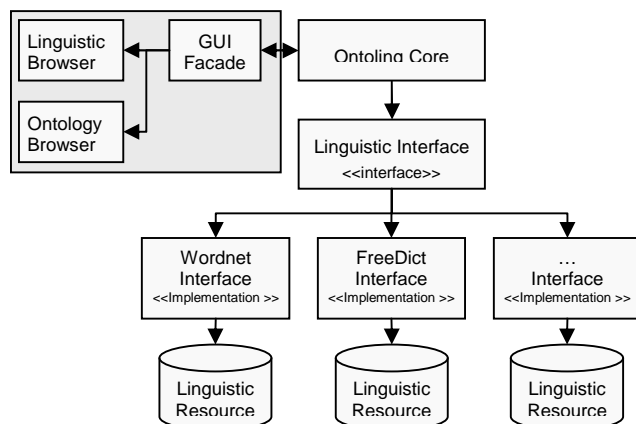


Fig. 1. Ontoling Architecture

Demo explanation

Here follows a partial extract of the Ontoling User Manual page, which can be browsed in its original form from: <http://ai-nlp.info.uniroma2.it/software/OntoLing/UserManual.html>

This is not properly a “one page” demo explanation, but being it separated from the two pages poster description (and so not intended for publication on the poster proceedings) we felt we could exceed a bit further this limit.

Accessing the Linguistic Resource

During the demo session, each user will choose the kind of resource(s) he will want to exploit, taken from:

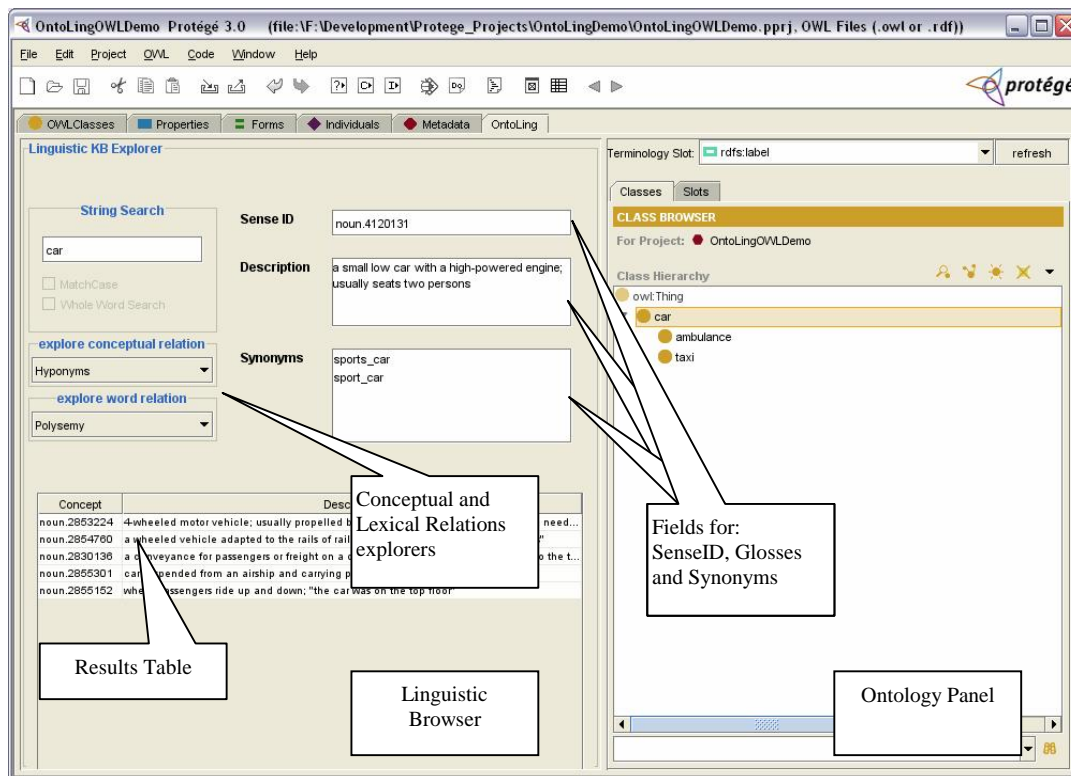
- Wordnet
- Any bilingual translation dictionary from: DICT dictionaries
- Any bilingual translation dictionary from Freelang dictionaries

First of all, the user may want to explore the Linguistic Watermark (LW) of the chosen resource, to learn how its characteristics have been mapped over the Watermark API.

Then, she will start Protégé, enable Ontoling plugin, and begin to use the application. When a LR's interface is being loaded, Ontoling parses its Watermark and expose those functionalities which are compatible with the resource's profile. As a consequence, the behavior of the whole application is dependant on the nature of the loaded resource and is thus defined at run-time. UI behaviour will be detailed to the user, showing how each resource's peculiarity is reflected into an automatically customized GUI. For example, with a [flatLR](#) (a kind of interface from the LW) resource, a search on a given term will immediately result in a list of (potential) synonyms inside a dedicated box in the GUI; instead, with a [conceptualizedLR](#) resource, a list of word senses will appear in a results table at first, then it will be browsed to access synonymical expressions related to the selected sense. Analogous adaptive approaches have been followed for many other aspects of the Linguistic Watermark (mono or bidirectional Bilingual Translators, presence of glosses, Taxonomical structures and so on...).

Users can browse the linguistic resource (LR) through the Linguistic Browser Panel (the panel on the left), using the following functionalities:

- **search** for words and terms from the loaded LR through the string search panel
- **explore** *lexical* ("explore lexical relation" Combo Box) and *semantic* ("explore conceptual relation" Combo Box) relations
- depending on the nature of the specific LR being loaded, **view full description** of *words*, *terms* and their related *senses*, including *glosses* and *synonyms*



Enrich the Ontology with terms from the linguistic resource

Frames of the ontology can be enriched with elements from the linguistic resource.

Using the context menu from the Ontology Panel (both class and slot views), it is possible to:

- **Add Terms** selected from the Linguistic Browser as additional labels for the selected frame.
- **Add Gloss** of a selected word sense as a description for the selected frame (only available on [LRWithGlosses](#) resources)
- **add IDs** of senses selected from the linguistic browser as additional labels for the frames; this is useful for people needing pointers from ontology concepts to senses from a given LR (only available on [conceptualizedLR](#) resources).

Support for OWL and Protégé Standard Models

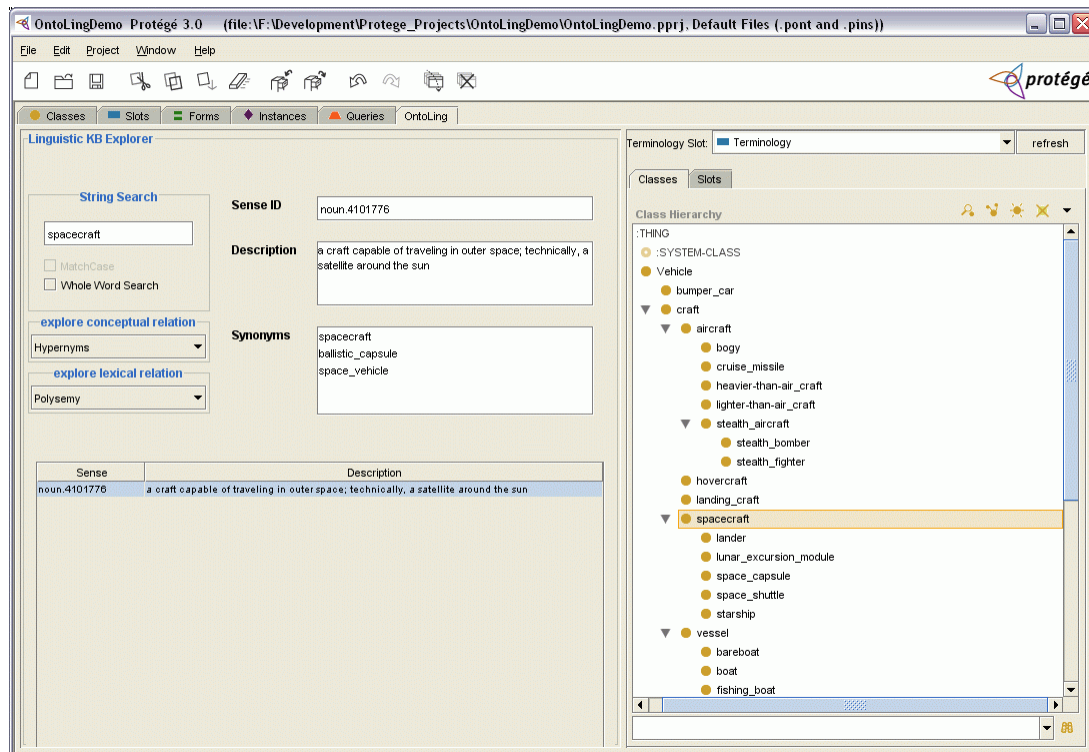
Ontoling supports linguistic enrichment of ontologies built upon both Protégé Standard Model and OWL Model. Separate examples will be shown to the user for handling with these two (partially) different models. In particular, she will learn how to build an appropriate meta-ontology for the Standard Protégé Model, to adapt existing ontologies or prepare new ones to be enriched with synonyms, glosses and different translations for the same concept.

Ontology Development through exploitation of Linguistic Resources

Other functionalities have been added to assist users in the process of ontology development:

- **Change class/slot name** using a word/term selected from the linguistic browser
- **Create a subclass** of the selected frame using as ID a word/term selected from the linguistic browser.
- **Generate SubClasses** of selected frame using all of the subsenses of the sense selected from the linguistic browser (only available on [TaxonomicalLR](#) resources).

Users will be guided in building a new ontology, with classes and slot, by capturing elements from Wordnet taxonomy.



Demo Example: creation of a taxonomy tree of ontology classes from a Wordnet subtree of synsets