LexOWL: A Bridge from LexGrid to OWL

Cui Tao PhD Jyotishman Pathak PhD Harold R. Solbrig Christopher G. Chute MD DrPH
Mayo Clinic College of Medicine, Rochester, MN

Abstract

The Lexical Grid project is an on-going community driven initiative that provides a common terminology model to represent multiple vocabulary and ontology sources as well as a scalable and robust API for accessing such information. In order to add more powerful functionalities to the existing infrastructure and align LexGrid more closely with various Semantic Web technologies, we introduce the LexOWL project for representing the ontologies modeled within the LexGrid environment in OWL (Web Ontology Language). The crux of this effort is to create a "bridge" that functionally connects the LexBIG (a LexGrid API) and the OWL API (an interface that implements OWL) seamlessly. In this paper, we discuss the key aspects of designing and implementing the LexOWL bridge. We compared LexOWL with other OWL converting tools and conclude that LexOWL provides an OWL mapping and converting tool with well-defined interoperability for information in the biomedical domain.

Introduction

The Lexical Grid project (LexGrid) [2, 12] coordinated by the Mayo Clinic Division of Biomedical Statistics and Informatics provides support for a distributed network of lexical resources such as terminologies and ontologies via standards-based tools, storage formats, and access mechanisms. The LexGrid system supports formats such as HL7 RIM, OBO, OWL/Protégé frame, UMLS RRF, and LexGrid XML. It models ontology information including versioning, provenances, entities, associations, and instances. LexGrid loads ontologies and terminologies from different sources, maps the information into the LexGrid model, and stores them in a backend database. Information modeled by LexGrid can be accessed through LexBIG, an interface that implements the LexGrid model, on top of which standard tools and services can be built.

A valuable augmentation to LexGrid is the adoption of Semantic Web technologies. The recent emergence of the Semantic Web and the Web Ontology Language (OWL) [4] is fostering a new level of interoperability. The biomedical informatics community greatly benefit by applying OWL's combination of formal semantics, rich expressiveness and shared software base to biomedical and clinical terminologies. The LexOWL project provides a round trip between LexGrid and OWL. In this paper, we focus on the direction from LexBIG to OWL. Through LexOWL, information modeled in LexGrid can be represented in OWL. Hence, tools and services that have been developed by the Semantic Web community can be directly applied to the biomedical and clinical domain. To name a few, we can use Protégé, which is a widely-used OWL ontology authoring tool, to browse and edit the information modeled in LexGrid. We can apply different reasoning tools to medical and clinical terminologies, to check consistency or to infer new knowledge. We can use OWL ontology modularity tools to integrate or extract ontology modules as well as use OWL ontology mapping tools to map ontologies. The biomedical terminology community has been actively seeking connections to OWL. OBO2OWL [1], OBOInOWL [9], Protégé OBO to OWL Tab [10], and Protégé 4 OBO loader provide mappings and conversions from OBO to OWL. The conversion from UMLS Semantic Network to OWL has been studied in [6, 8]. The NCI Thesaurus to OWL DL conversion is discussed in [11]. The International Healthcare Terminology Standards Development Organization also released a Perl converter for converting from SNOMED CT to OWL in recent SNOMED CT releases. LexOWL augments all these efforts by providing LexGrid a converter to OWL. Compared to the other tools, LexOWL has an inherit advantage in that, it can convert all the ontologies and terminologies from different sources modeled by LexGrid without individual mappers and converters. As an immediate benefit, LexOWL provides a well-defined interoperability across these sources since all the different resources are modeled by LexGrid.

We make the following contributions in this paper.

- LexOWL functionally converts LexGrid to OWL through an API bridge and represents the information modeled in LexGrid in the OWL API representation. By doing so, we can leverage the services and tools developed for OWL and the Semantic Web directly.

- LexOWL provides an OWL converter with relatively well-defined interoperability for different biomedical terminologies and ontologies.

- LexOWL provides a dynamic interface between LexGrid and Protégé so that Protégé can use LexGrid as its backend database, which could be a valuable addition to Protégé 4.

The rest of the paper is structured as fellow. We begin with an overview of the LexOWL system in Section 2. In Section 3, we discuss how LexOWL maps LexGrid components to OWL. In Section 4, we compare the OWL ontologies exported by LexOWL to those converted by the existing tools. Finally, in Section 5 we summarize and consider future work.

LexOWL System Overview

Figure 1 shows the LexOWL system overview. The core component of LexOWL is the LexOWLManager. It manages both the LexBIG service through which we can access the LexBIG API, and the OWL Ontology manager through which we can access the OWL API. On the left hand side of the system overview, the LexGrid system loads ontologies in different formats from different sources, translates them to LexGrid representation as well as saves the knowledge to a relational database. Through the LexBIG API, LexOWL can access the ontologies loaded in the database. On the right hand side of the system overview, through the OWL API, LexOWL re-represents the information in the LexGrid database virtually to the OWL API Ontology representation, which can the be used directly by Protégé 4 and other Semantic Web tools.

Thus, in essence, LexOWL maps LexGrid to OWL on the API level. It is not just a tool that maps and converts from one format to another. In addition to that, it generates a "bridge" between the two APIs. The "bridge" accesses information from the LexBIG API and translates it to the OWL API's representations. The benefit of an API "bridge" is that even if the backend representations for ontologies change, the "bridge" still performs the same way and an update is not necessary.

We also defined the LexGrid to OWL mapping and a lexgrid2owl meta-ontology [3], based on which LexOWL can re-represent a selected LexGrid ontology to the OWL API representation. In the next section, we discuss how LexOWL maps LexGrid to OWL.
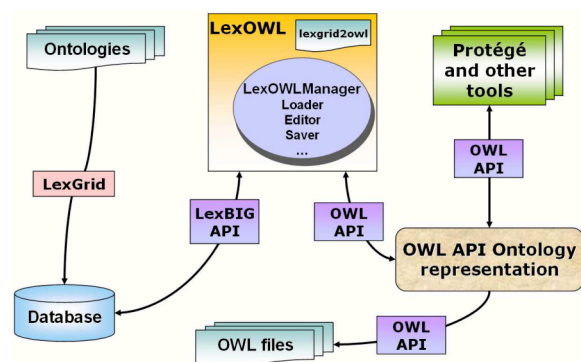

Figure 1 LexOWL System Overview.

LexGrid to OWL Mapping

LexOWL first maps the general ontology information. This includes information about the ontology itself such as name, version, and copyright. For some information, we can find equivalent representations in OWL (e.g., codingScheme to *owl:ontology*, localName to *rdfs:label*, and representsVersion to *owl:versionInfo*). For some

information, we can find equivalent representations in standard name spaces such as dublin core (e.g., formalName to *dc:title* and copyright to *dc:rights*). We used the lexgrid2owl meta-ontology to represent the rest information (e.g, we define ApproxNumConcepts and isNative as two annotation properties in the meta-ontology).

LexOWL maps each LexGrid concept1[1] to an OWL class. A concept in the LexGrid model can have properties such as a concept code, descriptions, presentations, definitions, and sources. LexOWL uses the concept code as the OWL class name and assign concept descriptions to a set of *rdfs:label*. In the lexgrid2owl meta-ontology, we define three OWL classes, Presentation, Definition, and Source, to represent the presentations, definitions, and sources in the LexGrid concept properties. We also defined annotation properties: *hasPresentation*, *hasDefinition*, and *hasSource* in the meta-ontology, to represent the relationships between concepts and such properties. Figure 2(a) shows a sample OBO term and Figure 2(b) shows its LexGrid representation. Figure 2(c) shows how LexOWL represents this concept and its properties in OWL. LexOWL creates an OWL class for the Concept Code "TAIR:0000055" and assign the Entity Description "pollen development" as a *rdfs:label*. The class has three annotation properties, one *hasDefinition* and two *hasPresentation*s, which link to "definition21" (an instance of the Definition Class), "presentation37", and "presentation38" (two instances of the Presentation Class) respectively. In addition, "definition21" has an annotation property *hasSource*, which links to "source21". Each of these instances also has annotation properties that represent contents such as synonyms and definitions from the source document.

LexGrid also has a special kind of concepts— anonymous concepts—which it uses to represent the anonymous classes in OWL. LexOWL parses each anonymous class and translates it back to OWL based on concept properties. Figure 3 shows an example. The upper part shows the LexGrid representation. The concept "A38" is the anonymous concept which is equivalent to the concept "Father". LexOWL can translate it back to OWL as the lower part of Figure 3 shows, which is identical to the original OWL representation.
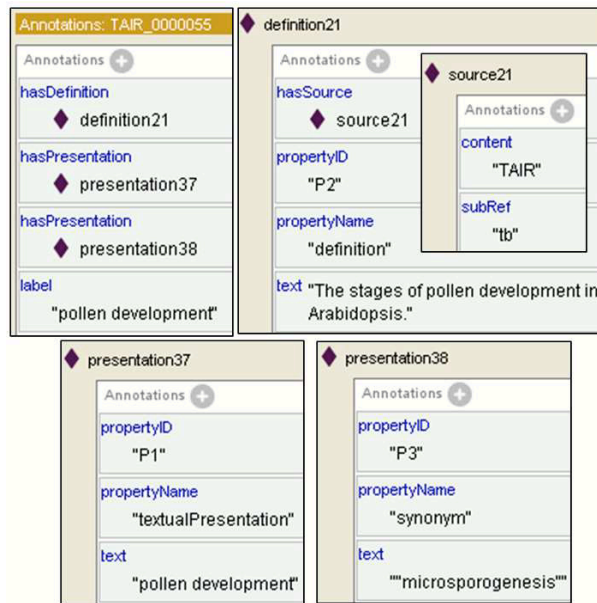
```
[Term]
id: TAIR:0000055
name: pollen development
def: "The stages of pollen development in Arabidopsis." [TAIR:tb]
synonym: "microsporogenesis" []
is_a: TAIR:0000022 ! body part developmental stages
```

(a) A Sample OBO Term

---

[1] A "concept" represents a "kind" or "universal" entity in the LexGrid 2008 model. Here we still use "concept" to be compatible with LexGrid 2008. We are upgrading both LexGrid and LexOWL to avoid using this confusing label.

(b) LexGrid Representation for the Sample Term

(c) LexOWL Representation for the Sample Term

Figure 2: An Example for Entity Mapping

An association in the LexGrid model establishes a relation between two LexGrid entities. LexOWL classifies the LexGrid associations into two types: pre-defined associations and other associations. A pre-defined association can be directly mapped to an OWL element. For example, the associations "subClassOf" (OWL), "CHD" (ICD 10), and "is a" (OBO) are all mapped to *owl*:*subClassOf*. The association "hasSubtype" (UMLS) is mapped as an inverse of OWL element subClassOf. The associations "equivalentClass" (OWL) and "same as" (UMLS) are mapped to *owl*:*equivalentClass*. For detailed information about the pre-defined-association mapping, please see [3].
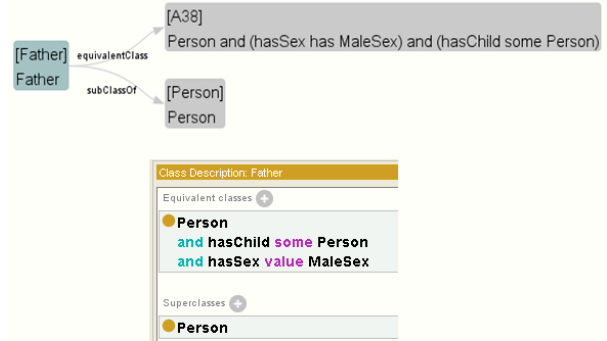

Figure 3: An Example of Anonymous Concept

Evaluation and Discussion

We tested LexOWL using different ontologies from various sources: OWL, OBO, UMLS Semantic Network, and WHO ICD10. We used Protégé Prompt [5] to compare the OWL ontologies generated by LexOWL and by other tools. We also sampled concepts and associations in each test ontology and compared them with the original source and checked whether all the related information are represent properly. The details of the results are listed below.

We tested on 5 OWL files. We chose these 5 ontologies carefully so that they cover most of the OWL Lite syntax introduced in [4]. We compared the OWL ontologies generated by LexOWL with the original ontologies. Each pair of ontologies is semantically equivalent to each other.

We also tested on 10 OBO files. For each OBO file, we compared the OWL ontology translated by LexOWL with those converted by OBO2OWL [1], Protégé 3.3.1 OBO to OWL Tab [10], and Protégé 4.0 OBO loader. All the four tools mapped OBO terms to OWL classes, OBO "isa" to OWL *subClassOf*, and used OWL *someValuesFrom* to represent relationships two classes. Semantically, the corresponding ontologies from all the 4 converters are identical. However, each converter defined its own annotation properties and used different annotation properties to represent the same OBO information. OBO2OWL and Protégé 4.0 OBO loader have relatively simple and straightforward conversions where they used the OBO labels directly as the OWL annotation property names. Protégé OBO to OWL Tab and LexOWL processed information in a lower granularity (e.g., the "def" in Figure 2(a) is parsed and the source information is annotated separately.)

We used LexOWL to export UMLS Semantic Network loaded in LexGrid to an OWL file and compared it with the one converted by Jimenez-Ruiz [6]. LexOWL uses the UIs as the OWL class names versus Jimenez-Ruiz uses the actual names. Hierarchically, these two ontologies are identical. Jimenez-Ruiz introduced some annotation properties that are specific for the UMLS Semantic Network where LexOWL used lexgrid2owl

meta-ontology to represent all the information. For example, Jimenez-Ruiz mapped SRDEF to *rdfs:comment*, whereas LexOWL mapped it to *lexgrid2owl :Definition*, which can bring better interoperability since definitions of terms from other sources are also mapped to *lexgrid2owl:Definition*. Jimenez-Ruiz used *owl:allValuesFrom* to represent relationships between two classes and LexOWL used *owl:someValuesFrom* since this is the default restriction LexOWL uses for representing relationships between classes[2].

We also used LexOWL to export ICD10 WHO second edition loaded in LexGrid to an OWL file and compared it with the OWL file converted by Cardillo, et al. [7]. Hierarchically, these two ontologies are identical. The ontology converted by [7] only covered hierarchical information, however. Information such as exclusions and inclusions are ignored whereas LexOWL considered them as OWL *ObjectProperties*, thereby preserving the semantics.

In summary, the test results show that LexOWL can convert information modeled in LexGrid to OWL successfully. LexOWL uses a single meta-ontology for all different sources where other tools use different meta-ontologies even for the same format. Hence, the ontologies converted by LexOWL has better Interoperability that will bring benefits in ontology mapping, integration and reasoning in the future.

Concluding Remarks and Future Work

We introduced LexOWL, a system that functionally connects LexGrid to OWL through a bridge over the LexBIG and the OWL APIs. LexOWL can represent information modeled in LexGrid in the OWL API representation, so that tools and services that are developed for OWL can be applied to the biomedical terminologies and ontologies. LexOWL also provides a LexGrid-to-OWL converter with a well-defined interoperability for information from different sources and in different formats.

As for the future work, several directions remain to be pursued. First, we would like to investigate performance of LexOWL with large-sized ontologies such as SNOMED CT, the Gene Ontology, and ICD10. We would like to add the editing and saving function as Figure 1 shows, so that we not only can browse, but also edit information represent in LexGrid using Protégé. Finally, LexOWL serves as a foundational pillar for ontology reasoning and inference. Our next step is to explore toward that direction on biomedical and clinical information.

---

[2] How to represent the semantic relationships between classes in a more precise way is a problem we are investigating when mapping information to LexGrid and is out of the scope of this paper.

References

1. OBO2OWL: Lossless transformation between OBO and OWL. http://www.cs.utexas.edu/~hamid/research/obo2owl.cgi, 2008.
2. LexGrid: The Lexical Grid. https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexGrid, 2009.
3. LexGrid to owl Mapping documentations. https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexGrid_Documentation, 2009.
4. OWL Web Ontology Language Reference. http://www.w3.org/ TR/owl-ref/, 2009.
5. The Protégé Prompt Tab. http://protege.stanford.edu/plugins/prompt/prompt.html, 2009.
6. The UMLS Semantic Network in OWL. http://krono.act.uji.es/people/Ernesto/UMLS_SN_OWL, 2009.
7. E. Cardillo, C. Eccher, L. Serafini, and A. Tamilin. Logical analysis of mappings between medical classification systems. In *Proceedings of the 13th international conference on Artificial Intelligence*, pp 311–321, Sep 2008.
8. D. Fensel, K. Sycara, and J. Mylopoulos. Representing the UMLS semantic network using OWL. In *Proceedings of the Second International Semantic Web Conference*, pp 1–16, Sanibel Island, Florida, Oct 2003.
9. C. Golbreic, M. Horridge, I. Horrocks, B. Motik, and R. Shearer. OBO and OWL: Leveraging semantic web technologies for the life sciences. In *Proceedings of the 6th International Semantic Web Conference*, pp 169–182, Busan, Korea, Nov 2007.
10. D. Moreira and M.A. Musen. OBO to OWL: a protégé OWL tab to read/save OBO ontologies. *Bioinformatics*, 23(14):1868–1870, 2007.
11. N. Noy, S. de Coronado, H. Solbrig, G. Fragoso, F. Hartel, and M. Musen, Representing the NCI Thesaurus in OWL DL: Modeling tools help modeling languages. *Journal of Applied Ontology*, 3(3):173–190, 2008.
12. J. Pathak, H. Solbrig, J. Buntrock, T. Johnson, and C. Chute. LexGrid: A framework for representing, storing, and querying biomedical terminologies from simple to sublime. *Journal of the American Medical Informatics Association*, 16(9), 2009.