

Dbnary: Wiktionary as a Lemon Based RDF Multilingual Lexical Ressource

Gilles Sérasset

UJF-Grenoble 1, Laboratoire d'Informatique de Grenoble
GETALP Team, BP 53, 38051 Grenoble cedex 9, France
`gilles.serasset@imag.fr`

Abstract

This short paper presents our effort to extract an RDF Multilingual Lexical Resource from the wiktionary data. This lexical resource is structured using the LEMON Model.

The extracted data is registered at <http://thedatahub.org/dataset/dbnary>. **Keywords:** Wiktionary, Multilingual Lexical Resource, Lexical Networks, LEMON, RDF.

1 Introduction

The GETALP (Study group for speech and language translation/processing) team of the LIG (Laboratoire d'Informatique de Grenoble) is in need for multilingual lexical resources that should include language correspondences (translations) and word sense definitions. In this regard, the set data included in the different wiktionary language edition is a precious mine.

Alas, many inconsistencies, errors, difference in usage do exist in the different wiktionary language edition. Hence, we decided to provide an effort to extract precious data from this source and provide it to the community a Linked Data. After a first version that used an RDF version of the LMF model and described in Sérasset [2012], we decided to adapt our extractors to LEMON model.

2 Extracting data from wiktionary

Errors and incoherence are inherent to a contributive resource like wiktionary. Some language editions (like French and English) have many moderators that do limit the number of incoherence among entries of the same language. Moreover, such languages, which contain the most data, use many *templates* that simplify the extraction process. For instance, the translation section of the French dictionary usually uses a template to identify each individual translation.

This is not true anymore with less developed wiktionary language editions. For instance, in the Finnish edition, some translations are introduced by a template giving the language (e.g. {fr} precedes French translation) and others are introduced by the string "ranska" which is the Finnish translation for "French". In this case the translator needs to know the Finnish translation of all language

names to cope with the second case and avoid losing almost half of the available translation data.

Our extractor is written in java and is open-source (LGPL licensed, available at <http://dbnary.forge.imag.fr>).

3 Extracted Data as a LEMON Lexical Resource

3.1 Extracted Entries

The main goal of our efforts is not to extensively reflect wiktionary data, but to create a lexical resource that is structured as a set of monolingual dictionaries + bilingual translation information. Such data is already useful for several application, but it is merely a starting point for a future multilingual lexical database.

The monolingual data is always extracted from its own wiktionary lexical edition. For instance, the French lexical data is extracted from French language edition (the data available on <http://fr.wiktionary.org>). Hence, we completely disregard the French data that may be found in other language editions.

We also filtered out some part of speeches in order to produce a result that is closer to existing monolingual dictionaries. For instance, in French, we disregard abstract entries that are prefixes, suffixes or flexions (e.g.: we do not extract data concerning *in-* or *-al* that are prefixes/suffixes and have a dedicated page in French language Edition).

Our work did focus only on the lexical data. Hence, we do not provide any reference to any ontology.

3.2 LEMON and non-LEMON modeled Extracted Data

All the extracted data could not be structured using LEMON only model. For instance, LEMON does not contain anything to represent translations between languages as it assumes that such a translation will be handled by the ontology description. Moreover, LEMON assumes that all data is well-formed and fully specified. As an example, synonymy relation is a property linking a *Lexical Sense* to another *Lexical Sense*. While this is correct to assume as a *principle*, this does not account for the huge amount of legacy data that is available in dictionaries and lexical databases.

In order to cope with this legacy data, we introduced several classes and properties that are not LEMON entities. However, when a piece of data is representable as a LEMON entity, then it is done so.

3.3 Example of an extracted lexical entry

The dbnary extracted data contains the following information:

Lexical Entries: an instance of `lemon:LexicalEntry` corresponds more or less to a "part of speech" section in a wiktionary page. This means that it is defined by an unique canonical written form, a part of speech and a number (in case of homonymy). When wiktionary data allows for it, we try to distinguish between `lemon:Word` and `lemon:Phrase` that are defined as specific lexical entries.

Lexical Forms: lexical entries are connected, through the `lemon:canonicalForm` property to a lexical form that gathers a written form and a pronunciation (when available). They may also be connected to alternative spelling through `lemon:lexicalVariant` property.

Lexical Senses: an instance of `lemon:LexicalSense` correspond to one definition in the wiktionary page. It is the target of the `lemon:sense` property of its containing Lexical Entry. Each lexical sense is associated with a `dbpedia:senseNumber` property (that contains the rank at which the definition appeared in the wiktionary page) and a `lemon:definition` property.

Part Of Speech part of speech properties are available in the wiktionary data in 2 distinct properties that are attached to lexical entries:

- `dbnary:partOfSpeech` is a data property whose value is a string that contains the part of speech *as it was defined in wiktionary*
- `lexinfo:partOfSpeech` is a standard property that is bound to isocat data categories and which value is a correct isocat data category. This property is only available when the mapping between wiktionary part of speech and isocat part of speech is known.

Vocable: the main unit of data in wiktionary is a *wiktionary page* that may contain several lexical entries. Many lexical data is represented as links to a *page*. Most of the time, there is not enough data to know to which lexical entry (or lexical sense) these links point to. Hence if we want to keep these *underspecified* relations, we need to define units that represent wiktionary pages. This is the role of the `dbnary:Vocable` class. Instances of this class are related to their lexical entries through the `dbnary:refersTo` property.

Nyms: most wiktionary language edition do provide "nym" relations (mainly synonym, antonym, hypernym, hyponym, meronym and holonym). As we already mentioned this legacy data is not representable using LEMON model, unless we know for sure the source and target lexical sense of the relation. In order to cope with this legacy data, 6 new "nym" properties (in `dbnary` name space). Additionally, we defined a class called `dbnary:LexicalEntity` that is defined as the union of LEMON lexical entries and lexical senses. The "nym" properties domain and range are lexical entities.

Most of these properties do link a *lexical entry* to a *vocable*, as there is not enough information in wiktionary to promote this relation to a full class *sense to sense* relation. Some of these properties are however promoted to a *Lexical Sense to Vocable* relation when the lexical entry is unambiguous (contains only one sense).

Translations: As there is no way to represent bilingual translation relation in LEMON, we introduced the `dbnary:Equivalent` class that collects translation information contained in wiktionary. This class admits several properties:

- `dbnary:isTranslationOf` relates the equivalent to its source *lexical entry*. In this extraction process, we decided not to relate the equivalent object to its source *lexical sense*. The reason is that, when some information is available to distinguish between lexical senses, it is mainly targeted to a human audience and there is no simple and reliable process to link to the correct sense. Instead, we rather keep all the available disambiguation information for a later specialized processing.
- `dbnary:targetLanguage` is a data property whose type is a string containing the target language code as defined in ISO639-3 standard.
- `dbnary:writtenForm` gives the written form of the translation in the target language. Here, we decided not to relate to a vocable as some translations are not to be defined as lexical entries in the target language.
- `dbnary:glose` is a string property that contains any available information used to dentate the lexical sense of the source of the equivalent.
- `dbnary:usage` is a string property that contains any available information concerning this equivalent object. It usually gives additional information on the target entry.

3.4 Size of the involved data

At the time of writing, the extracted data from the most up to date dump files are the following:

| Nodes in graphs | | | | | | |
|-----------------|---------|---------------------|--------|---------|---------|------------|
| | English | French | German | Finnish | Italian | Portuguese |
| Lexical entries | 478764 | 260647 ^a | 101867 | 30478 | 24030 | 31105 |
| Vocables | 458317 | 270048 | 166567 | 30946 | 29591 | 32784 |
| Lexical Senses | 386030 | 341720 | 88780 | 38713 | 33731 | 55331 |
| Equivalents | 942425 | 406947 | 438379 | 101733 | 56883 | 49029 |

^aamong which 231522 words and 24434 phrases.

Table 1: Size of the extracted lexical networks.

As the extraction is performed each time a wiktory dump is available, this numbers are constantly evolving, as the wiktory data is evolving and as the extractor itself maybe improved.

4 Conclusion and Perspectives

The current paper shows preliminary results on an open source tool to extract a LEMON based lexical network from different wiktory language editions. Such a work is interesting for many users that will be able to use the extracted data in their own NLP system. Moreover, as the extracted resource uses the Resource Description Framework (RDF) standard and the LEMON model, the extracted data is also directly usable for researchers on the Semantic Web, where

it could be used to ease the ontology alignment systems when terms in different languages are used to describe ontologies of a domain.

Our next objectives are to better generalize the treatments of the current extractors, so that it will be easier to create extractors for other languages. We are currently working on the Russian and we welcome all initiative aiming at the addition of new language to this open-source tool.

5 Acknowledgements

The work presented in this paper was conducted in the Videosense project, funded by the French National Research Agency (ANR) under its CONTINT 2009 programme (grant ANR-09-CORD-026).

References

Gilles Sérasset. Dbmary: Wiktionary as a LMF based Multilingual RDF network. In *Language Resources and Evaluation Conference, LREC 2012*, Istanbul, 2012. URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/387_Paper.pdf.