

Ontology-lexica with *lemon*

John P. McCrae and Sebastian Walter

Semantic Computing Group, Bielefeld University
Session 1 of “Building the Multilingual Web of Data” Tutorial

Problem

- ▶ Ontologies have become popular.
- ▶ Use several formalisms: RDFS, OWL, F-Logic, etc.
- ▶ Ontologies do not have much linguistic information.

- ▶ What is the plural? Easy for English, not for German

Ontologies

Take a word:

“edema”

And it means something, so we put it in an ontology and give it an identifier (URI):

`http://dbpedia.org/resource/Edema`

`http://de.dbpedia.org/resource/Odem`

In fact it (already) has lots of identifiers linked on the web

`mesh:D004487` `icd10:R60.9` `umls:C0013604`

Ontologies

- ▶ We can describe the entity with *axioms*
- ▶ Relationships to entities in other ontologies
- ▶ Use reasoning to infer equivalence
- ▶ All done with the “Web Ontology Language” (OWL)
- ▶ Published by W3C in 2002; version 2 in 2008

Ontology labels

Concepts may be identified by many words

“edema” *“edemata”* *“dropsy”*

- ▶ These are all labels for the same ontology concept
- ▶ No differentiation
- ▶ Cannot say which are plural, which not

Inflection and Synonyms

We could introduce an element for each word:

`"edema"@en`

`"edemata"@en`

`"dropsy"@en`

`ll:form`

`ll:form`

`ll:form`

`lexicon:Edema`

`lexicon:Dropsy`

`ll:lexicalization`

`ll:lexicalization`

`dbpedia:Edema`

SKOS-XL

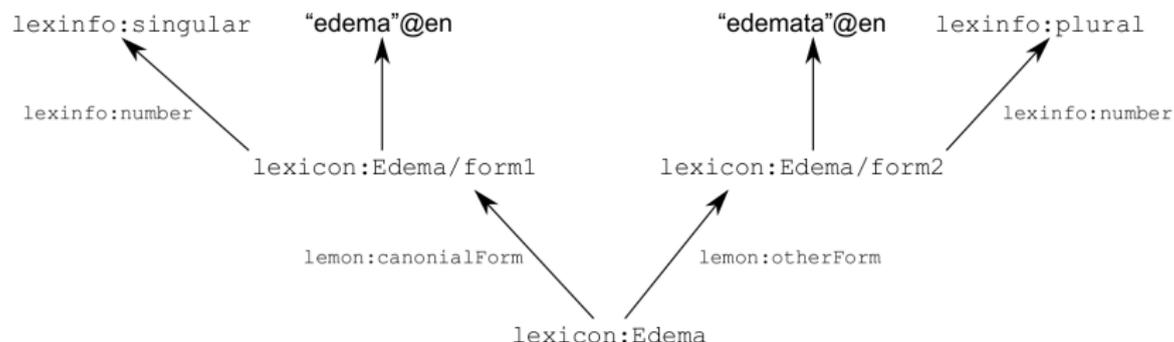
- ▶ Similar to SKOS-XL
 - ▶ eXtended Labels for the Simple Knowledge Organization System
 - ▶ W3C Recommendation since 2009
- ▶ SKOS-XL does not allow multiple forms of the same label
 - ▶ No grouping of “edema” and “edemata”
- ▶ “We [TopQuadrant] have yet to hear a use case that cannot be supported by SKOS alone” (Who need SKOS-XL? Maybe no-one. (Polikoff, 2013))

Forms

But such a distinction is only useful if we can say why:

“edema” (**singular**) *“edemata”* (**plural**)

Hence, forms are also nodes:

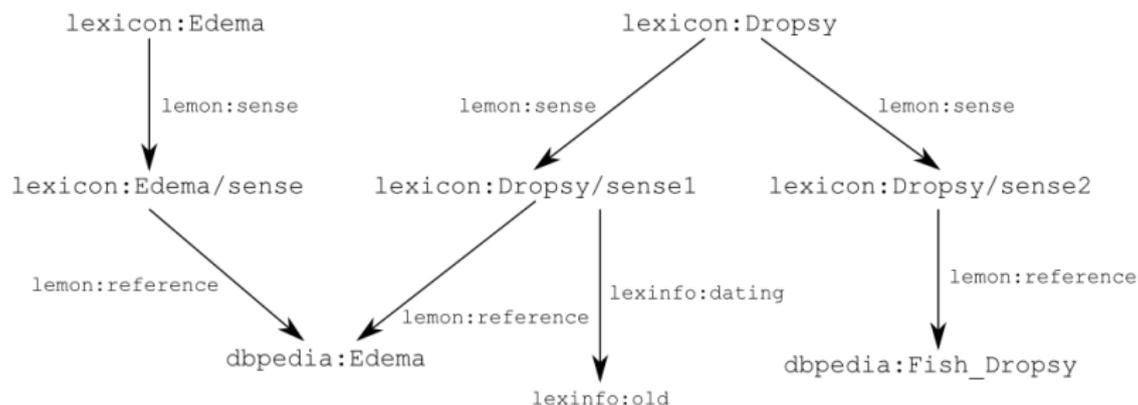


Senses

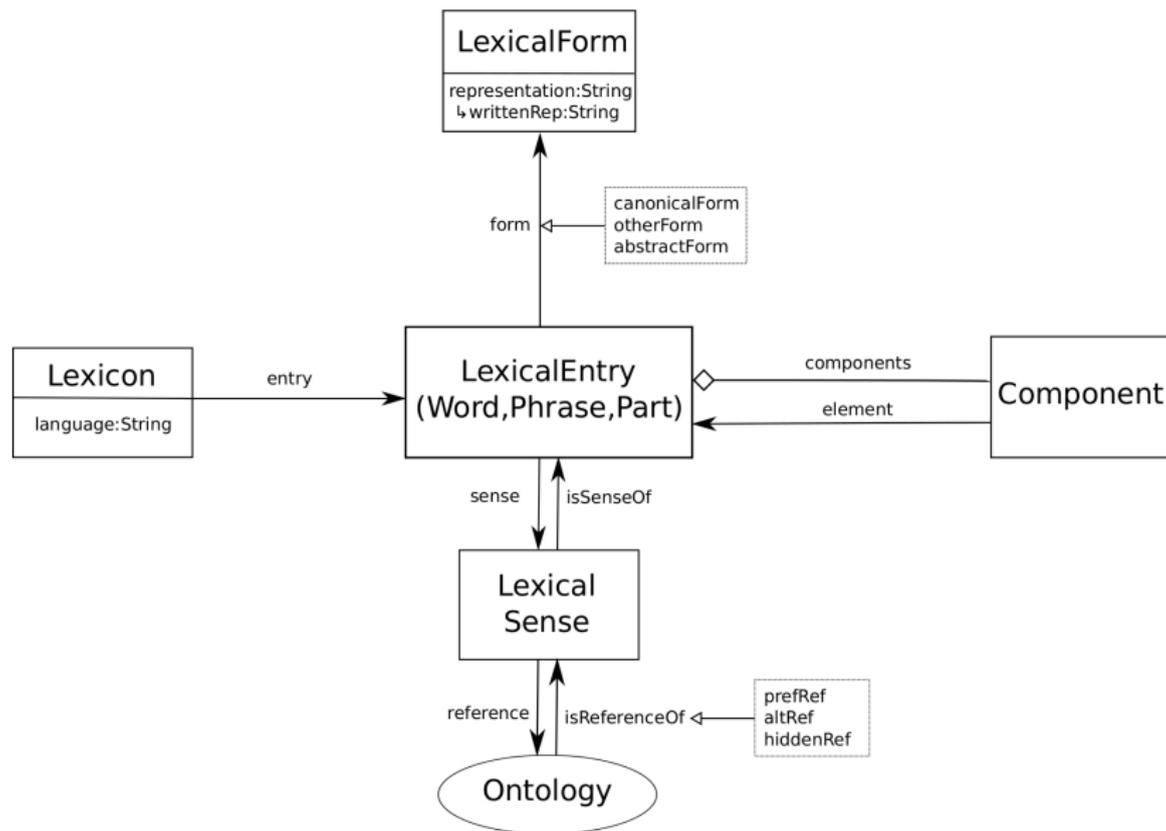
Sometimes we wish to say something about why a particular word is used

“edema” (**modern**) *“dropsy”* (**antiquated**)

Hence we introduce a sense to describe the usage of a word with a given meaning



The core of *lemon*



What is an (ontology-)lexicon?

- ▶ A lexicon is a collection of lexical information
- ▶ We do not need to define semantics within the lexicon
- ▶ “An ontology-based semantic lexicon would leave the semantics to the ontology, focusing instead on providing domain-specific terms and object descriptions in the ontology.” (Buitelaar, 2010)

Dictionaries as lexica

- ▶ In fact, a lexicon represents much of the information already found in a dictionary
- ▶ That is words, their forms and their meaning
- ▶ Must be machine-readable
- ▶ Take Wiktionary as an example

Wiktionary as a lexicon

W know - Wiktionary

Entry Discussion Citations

← Lexicon ← (Lexical) Entry

know ← (Lexical) Entry

Contents (hide)

- 1 English
 - 1.1 Etymology
 - 1.2 Pronunciation
 - 1.3 Verb
 - 1.3.1 Usage notes
 - 1.3.2 Derived terms
 - 1.3.3 Related terms
 - 1.3.4 Translations
 - 1.4 Noun
 - 1.4.1 Derived terms
 - 1.5 References
 - 1.6 Anagrams
- 2 Cornish
 - 2.1 Etymology
 - 2.2 Pronunciation
 - 2.3 Noun

English

Most common English words: before • see • over • #93 know • much • after • first

Etymology

From Middle English *knowen* from Old English *cnawan* from Proto-Germanic: **knanōnan* (“to know”) from Proto-Indo-European: **ǵnew-*, **ǵnō-* (“to know”).

cognates

Pronunciation

- (UK) IPA: /nəʊ/, SAMPA: /nəʊ/
- (US) IPA: /noʊ/, SAMPA: /noʊ/

Audio (UK) **en-us**

- *listen* · *help* · *info*
- *listen* · *help* · *info* (in some dialects or accents, but not in standard English)

Part of speech

Word Forms

Verb

(Lexical) Senses

Subcategorization

Reference

Wikidata has an article on: **know**

Done

Wiktionary as a lexicon

The image shows a screenshot of the Wiktionary website for the word "know". The browser's address bar displays "W know - Wiktionary". The page content includes the Wiktionary logo, the word "know" in large font, and a table of contents. Two orange boxes with arrows point to specific elements: "Lexicon" points to the "Entry" tab, and "(Lexical) Entry" points to the word "know".

W know - Wiktionary

a multilingual free encyclopedia

Wiktionary
[ˈwɪkʃənri] *n.*,
a wiki-based Open
Content dictionary
Witea [ˈwɪt kəri]

Entry Discussion Citations

Lexicon →

(Lexical) Entry →

know

Contents [hide]

- 1 English
 - 1.1 Etymology
 - 1.2 Pronunciation
 - 1.3 Verb

Wiktionary as a lexicon

Spanish
Euskara
فارسی
Français
Galego
한국어
Hrvatski
Ido
Italiano
עברית
Қазақша

- (US) IPA: /nɒʊ/, SAMPA: /nɒʊ/
- Audio (US) (file)
- Audio (UK) (file)
- Rhymes: -ɒʊ
- Homophones: no; now (in some dialects or accents, but not in standard English)

Verb

to know (third-person singular simple present **knows**, present participle **knowing**, simple past **knew** or **knowered** (dialect), past participle **known**, **knowen** (archaic), or **knowered** (dialect))

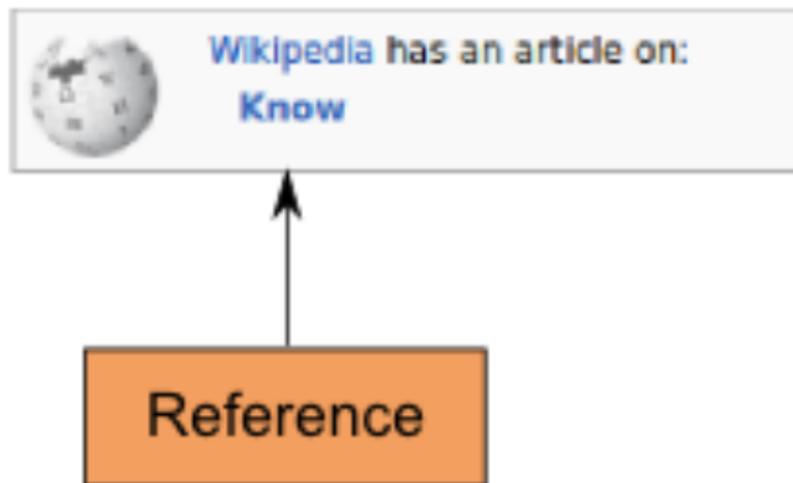
(Lexical) Senses

Subcategorization

1. (transitive) To be certain or sure about.
I know that I'm right and you're wrong.
He knew something terrible was going to happen.
2. (transitive) To be acquainted or familiar with; to have encountered.
I know your mother, but I've never met your father.
3. (transitive, also intransitive followed by **about** or, dialectally, **fram**) To have knowledge of; to have memorised information, data, or facts about.
He knows more about 1945, but he can't tell the year of 1946.

Done

Wiktionary as a lexicon



lemon's origins

- ▶ Lexical Markup Framework (ISO 24613)
 - ▶ Standard for representing lexicons
 - ▶ XML
- ▶ LexInfo, LIR
 - ▶ Represent lexical information relative to an ontology
 - ▶ OWL
- ▶ SKOS (W3C Standard)
 - ▶ Designed for Taxonomy/Vocabulary representation
 - ▶ RDF

Design goals

- ▶ RDF(S)
- ▶ Conciseness
- ▶ Not prescriptive
 - ▶ i.e., uses data categories
- ▶ Semantics by reference
 - ▶ i.e., uses ontologies
- ▶ Extensible

Why *lemon*: RDF(S)

- ▶ RDF models are labelled directed graphs
 - ▶ Allows for smarter representation
- ▶ Each entry has a URI
 - ▶ Queriable on the web using standards
 - ▶ Clear responsibility for data
- ▶ Linking possible between different lexica
 - ▶ Reuse of lexicon data
- ▶ Some induction possible (subproperties, classes etc.)

Why *lemon*: Conciseness

- ▶ Small models (i.e., fewer links, fewer kB)
- ▶ Easier to understand
- ▶ “Open-world”: Not necessary to state all facts
 - ▶ Multiple points of view

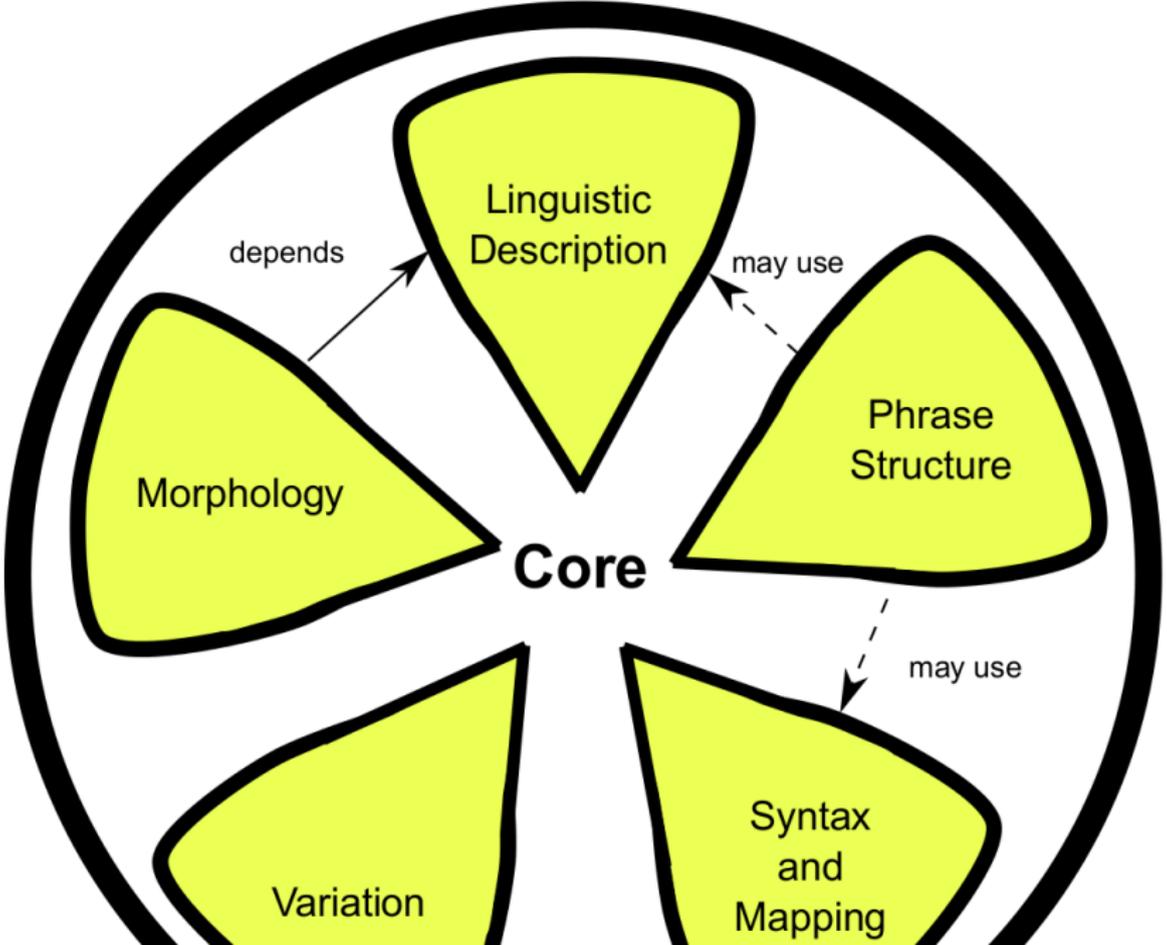
Why *lemon*: Semantics by Reference

- ▶ Meaning of a word given by reference
- ▶ Reference (generally an ontology) capable of representing more complex semantic information
- ▶ Disambiguation is performed relative to the ontology
- ▶ No (traditional) word senses
 - ▶ No clashing of word senses in cross-lingual mappings

Why *lemon*: Modular and extensible

- ▶ RDF(S) extensibility allows representation of
 - ▶ Subtle differences
 - ▶ Unexpected data categories
- ▶ Modularity
 - ▶ Different modules for different user requirements
 - ▶ New modules can be added later without affecting core

Modules



Design Patterns

Why design patterns?

- ▶ Simplify usage of *lemon*
- ▶ Factorize common patterns
- ▶ Cover 95% of entries
- ▶ Expressable as simple language

Using the compiler

`http://services.lider-project.eu/lemonpatterns`

Documentation:

`https://github.com/jmccrae/lemon.patterns`

The Design Patterns Language

- ▶ Header consisting of prefix declarations
- ▶ Any number of lexicons
 - ▶ With a URI
 - ▶ And a language
- ▶ Any number of patterns

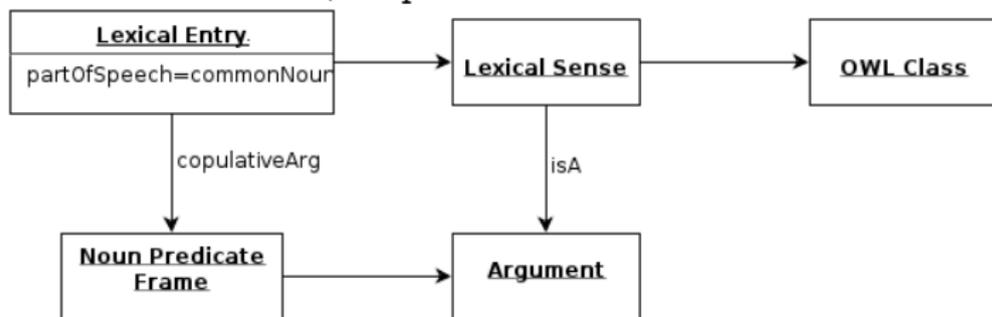
Example

```
@prefix example: <http://www.example.org/example#> .
```

```
Lexicon(<http://www.example.org/lexicon>, "en"  
  ClassNoun("test", example:Test)  
)
```

Noun Patterns

- ▶ Name, a proper noun referring to an individual
 - ▶ Name("Microsoft", <http://www.microsoft.com/>)
- ▶ Class Noun, a noun referring to a class in the ontology
 - ▶ ClassNoun("test", dbpedia:Test)



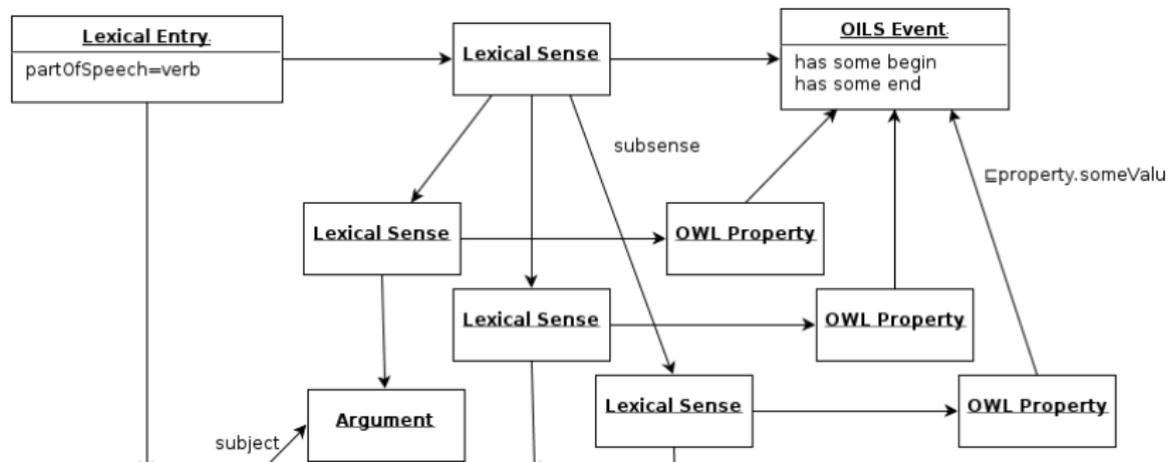
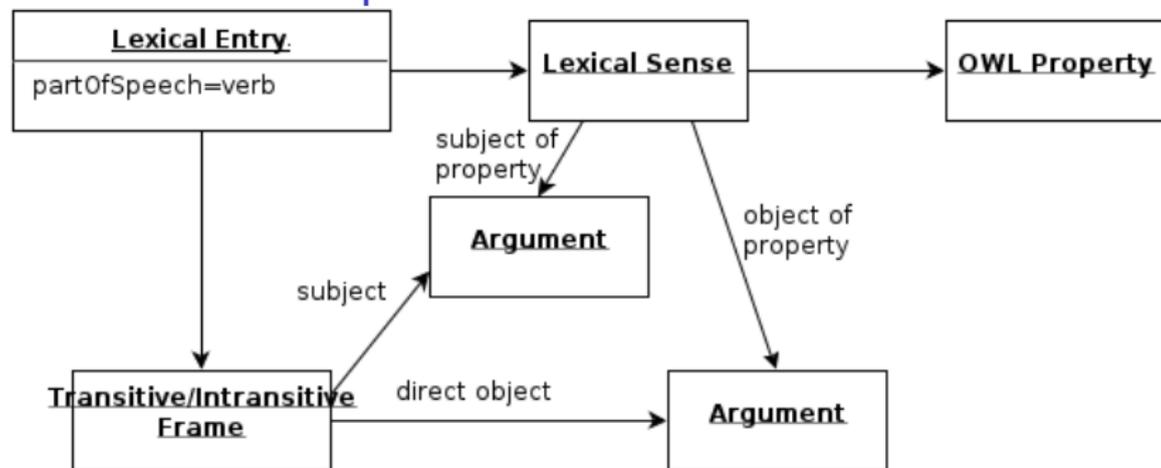
Noun Patterns

- ▶ If there is no class we may refer to a particular property value
 - ▶ `ObjectPropertyNoun("oenophile", ontology:loves, ontology:wine)`
- ▶ Relational Noun, a noun with a prepositional phrase referring to a property in the ontology
 - ▶ `RelationalNoun("father", dbpedia:father, propObj = PossessiveAdjunct)`
 - ▶ We use `PossessiveAdjunct` to allow “X is father of Y” and “X is Y’s father”

Verb Patterns

- ▶ State Verb, a verb indicating a single property in the ontology
 - ▶ `StateVerb("know", foaf:knows)`
- ▶ Consequence Verb, a verb causing a property in the ontology
 - ▶ `ConsequenceVerb("die", dbpedia:deathDate, propObj = PrepositionalObject("on"))`
- ▶ Event Verb, a verb describing an event with multiple roles

Verb Pattern Example



Adjective Patterns

- ▶ Intersective Adjectives, an adjective referring to a class
 - ▶ `IntersectiveAdjective("Belgian", ontology:Belgian)`
 - ▶ `IntersectiveObjectPropertyAdjective("Belgian", dbpedia-owl:nationality, dbpedia:Belgium)`
- ▶ Relational Adjective, like relational nouns
- ▶ Scalar Adjectives
 - ▶ `ScalarAdjective("big", [ontology:size > 5.0 for ontology:Building])`

Modifiers

- ▶ Inflectional form, e.g.,
 - ▶ `ClassNoun("Katze", onto:Cat)` with plural "Katzen"
- ▶ Gender (nouns-only)
 - ▶ `ClassNoun("Katze", onto:Cat)` feminine (*with ...*)
- ▶ Register
 - ▶ `StateVerb("knows", foaf:knows)` `neutralRegister`

Arguments

- ▶ Indicated with `propSubj` and/or `propObj`
- ▶ `Subject` (subject of a verb only!)
- ▶ `CopulativeArg` (subject for nouns and adjectives!)
- ▶ `DirectObject`
- ▶ `IndirectObject`
- ▶ `PrepositionalObject("for")` (not normally used for "of"!)
- ▶ `PossessiveAdjunct`

Multiword Expressions

- ▶ Multiword terms can be specified using the following syntax
 - ▶ `ClassNoun(["blood"/noun "vessel"/noun],
dbpedia:BloodVessel)`
- ▶ This gives a decomposition of a term

Task:

In your language model the following DBpedia properties:

- ▶ Person
- ▶ Surname
- ▶ GivenName
- ▶ profession = Historian
- ▶ Germany
- ▶ nationality = Germany
- ▶ stateOfOrigin
- ▶ relative
- ▶ predecessor
- ▶ spouse

Solution

```
@prefix dbo: <http://dbpedia.org/ontology/> .  
@prefix dbr: <http://dbpedia.org/resource/> .  
@prefix dbp: <http://dbpedia.org/property/> .
```

```
Lexicon(<>, "en",  
  ClassNoun("person", dbo:Person),  
  ClassNoun("surname", dbo:Surname),  
  ClassNoun(["given"/noun "name"/noun], dbo:GivenName),  
  ObjectPropertyNoun("historian", dbp:profession, dbr:Historian),  
  Name("Germany", dbr:Germany),  
  IntersectiveObjectPropertyAdjective("German", dbp:nationality, dbr:Germany),  
  StateVerb("come", dbp:stateOfOrigin, propObj = PrepositionalObject("from"))  
    with imperfect "came" with past participle "come",  
  RelationalAdjective("related", dbp:relative, relationalArg = PrepositionalObject("to")),  
  StateVerb("precede", dbp:predecessor, propSubj = DirectObject, propObj = Subject),  
  ConsequenceVerb("marry", dbp:spouse),  
  RelationalNoun("husband", dbp:spouse, propObj = PossessiveAdjunct)  
)
```