

Session 4: Ontology-lexica with lemon

John McCrae

CITEC, University of Bielefeld



NUI Galway
OÉ Gaillimh



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH



Universität Bielefeld



Ontology
Engineering
Group



The core

Beyond the core

Modeling examples

- Linguistic Description

- Variation

- Phrase structure

- Syntax and (Argument) Mapping

- Morphology

Generation of lexica

- Overview of Methodology

- Introduction to lemon-Editor

The core

Beyond the core

Modeling examples

- Linguistic Description

- Variation

- Phrase structure

- Syntax and (Argument) Mapping

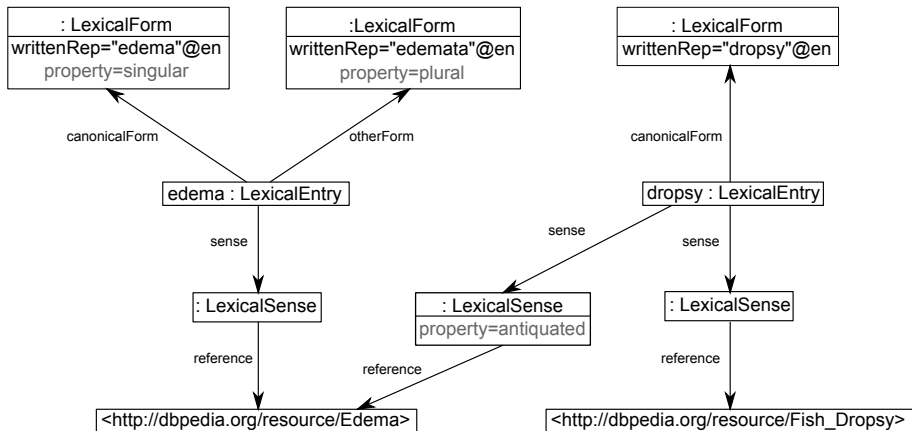
- Morphology

Generation of lexica

- Overview of Methodology

- Introduction to lemon-Editor

1. Ontology entity
2. Lexical sense
3. Lexical entry
4. Lexical form
5. (Written) Representation



- ▶ Form preferences
 - ▶ **Canonical:** The dictionary form of the word, e.g., “pescar”
 - ▶ **Other:** An inflected form, e.g., “pesco”, “pescas”, “pesca”, ...
 - ▶ **Abstract:** A partial form, root or stem, e.g., “pesqu-”
- ▶ Lexical preference
 - ▶ **Preferred:** The standard term for a concept, e.g., “tuberculosis”
 - ▶ **Alternative:** Another valid form, e.g., “TB”
 - ▶ **Hidden:** A deprecated version of the term, e.g., “consumption”

The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

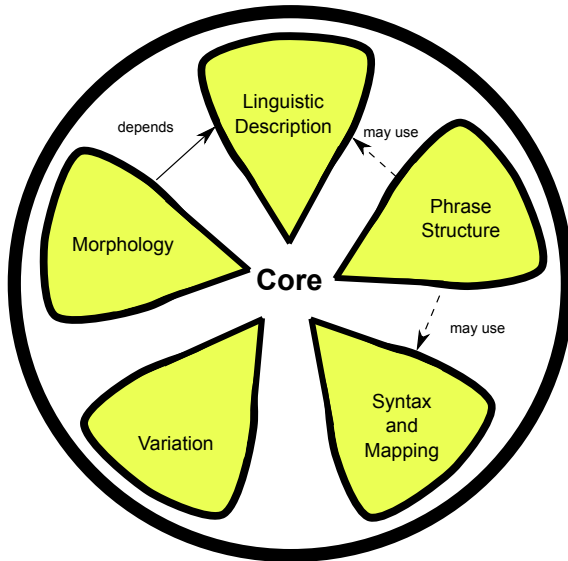
Morphology

Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

- ▶ Linguistic Description
 - ▶ Assigning words, forms and senses to semantic categories
- ▶ Variation
 - ▶ Describing links between entries
- ▶ Phrase structure
 - ▶ Linguistic analysis of multiple word labels
- ▶ Syntax and (Argument) Mapping
 - ▶ Describes how syntactic “frames” are bound to ontologies
- ▶ Morphology
 - ▶ Reusable descriptions of inflectional and agglutinative variation



Verb

love (third-person singular simple present **loves**, present participle **loving**, simple past and past participle **loved**)

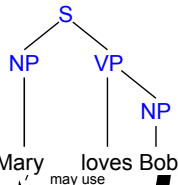
person	singular	
	first	second
Indicative	lo	tu
present	amo	ami
imperfect	amavo	amavi
past historic	amai	amasti
future	amerò	amerai

strong affection

- Afrikaans: liefde
- Albanian: dashuri ^(sq) f
- Arabic: حب ^(ar) (Hubb) m, محبة ^(ar) (maḥābba) f
- Aramaic: ܡܚܒܐ
- Armenian: սեր ^(hy) (ser)
- Azeri: sevgi ^(az), eşq ^(az), eşq ^(az)
- Bashkir: мәхәббәт (mәxәbbәt)
- Belarusian: любіць ^(be) (ljubóŭ) f, каханьня ^(be) (kaxánnja) n
- Bosnian: ljubav ^(bs) f
- Breton: karantez ^(br) f
- Bulgarian: любов ^(bg) (ljubóv) f, обича ^(bg) (obič) f
- Catalan: amor ^(ca) m
- Cebuano: gugma
- Cherokee: ᎠᎵᏍᏉᏚᎩ ^(chr) (tsigeyui)
- Chinese:

Mandarin: 爱 ^(cmn), 爱 ^(cmn) (ài), 爱情 ^(cmn), 爱情 ^(cmn) (àiqing)

Core



Mary loves Bob

⇒ $\lambda x. \text{Mary}(x) \quad \lambda x \lambda y. \text{Love}(x, y) \quad \lambda x. \text{Bob}(x)$
 ⊢ $\lambda x \lambda y. \text{Love}(\text{Mary}(x), \text{Bob}(y))$

The core

Beyond the core

Modeling examples

- Linguistic Description

- Variation

- Phrase structure

- Syntax and (Argument) Mapping

- Morphology

Generation of lexica

- Overview of Methodology

- Introduction to lemon-Editor

The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

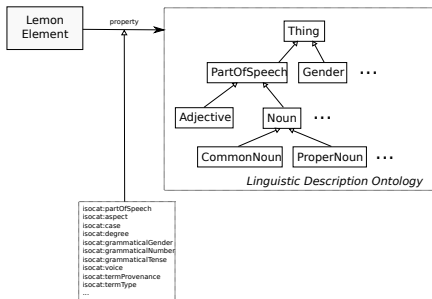
Morphology

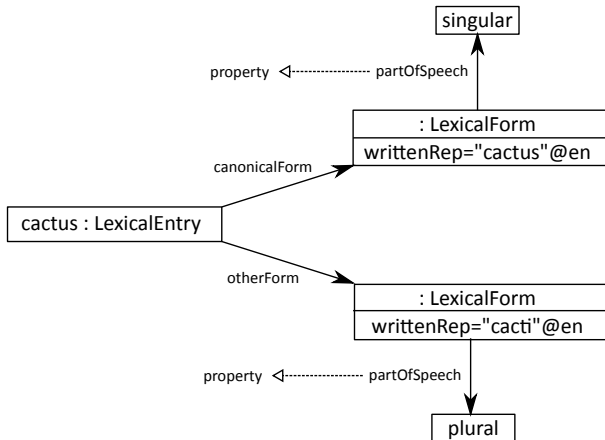
Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

- ▶ Any element in the lexicon may have properties
- ▶ Properties may be stated as subproperties of **lemon**'s property
- ▶ **lemon** does not have any such properties or values. A separate ontology is required, e.g.,
 - ▶ **GOLD**: General ontology for linguistic description
 - ▶ **ISOcat**: The ISO data category registry.
 - ▶ **LexInfo**: Monnet project collection of **lemon**-aligned categories (partially derived from ISOcat)





The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

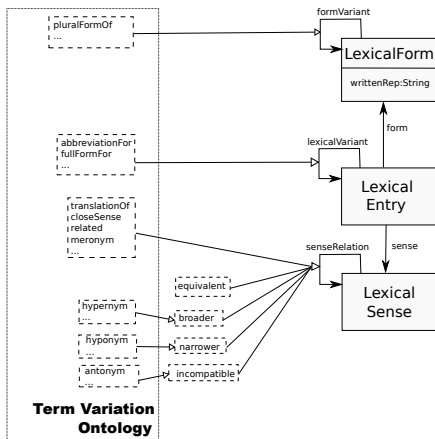
Morphology

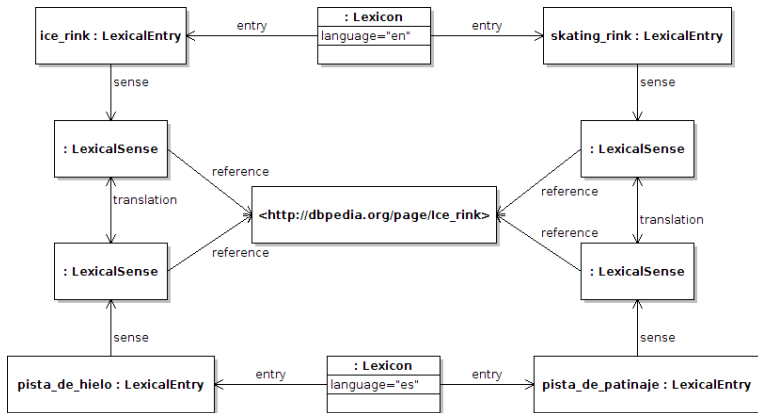
Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

- Forms, entries and senses may be related.
- Again, few **lemon** properties, mostly use external ontology
- Links typed as subproperties of `formVariant`, `lexicalVariant`, `senseRelation`
- Sense Relation does have subproperties `equivalent`, `broader`, `narrower`, `incompatible`
- Sense Relation can be used to model translation





The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

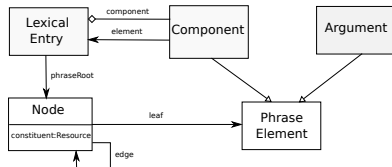
Morphology

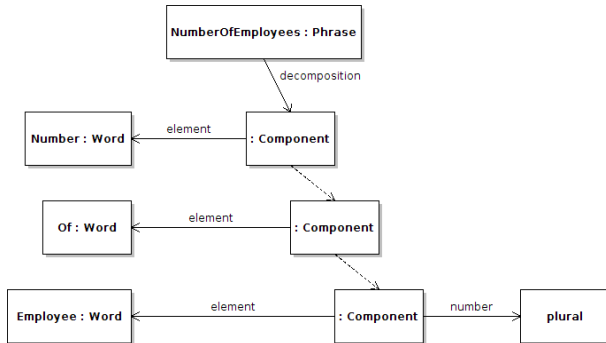
Generation of lexica

Overview of Methodology

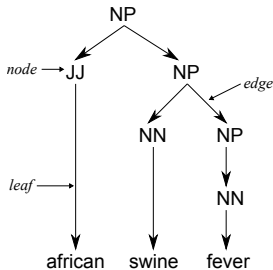
Introduction to lemon-Editor

- ▶ Entries marked as *Word*, *Phrase* or *Part* (of word)
- ▶ Decomposed into sub-entries
 - ▶ Phrase → Words
 - ▶ Word → Words and/or Parts
- ▶ Implemented by RDF list
 - ▶ Ordered
- ▶ Components may be marked to show necessary form properties

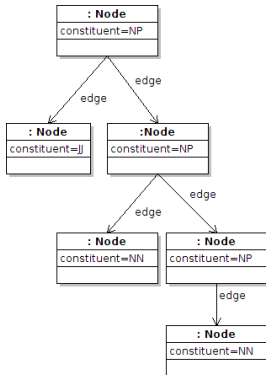




- ▶ We may also wish to describe the phrase structure (parse tree) of a term
- ▶ e.g.,

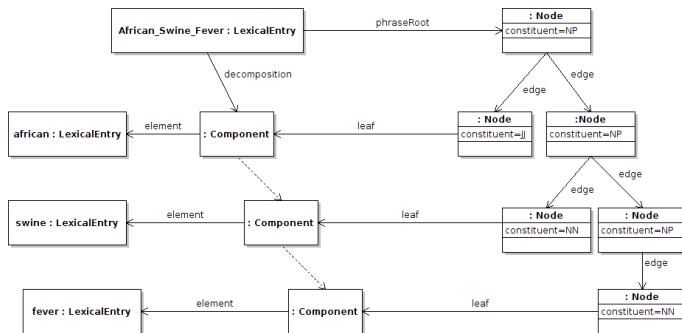


- ▶ Phrase structure is represented easily in RDF:



- ▶ Not ordered!

- ▶ Linked to the components and lexical entries by leaf and phraseRoot arcs



The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

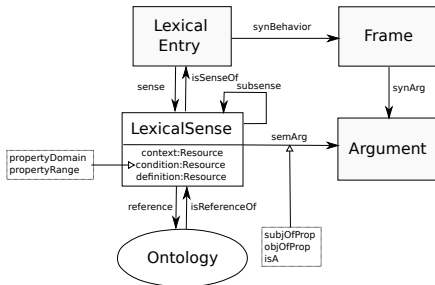
Morphology

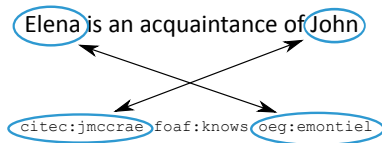
Generation of lexica

Overview of Methodology

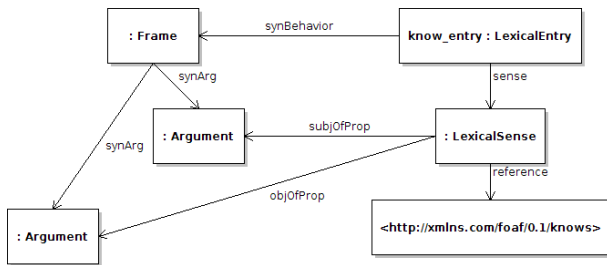
Introduction to lemon-Editor

- ▶ A verb (e.g., “know”) is always used in a sentence
 - ▶ “John knows Elena”
- ▶ Similarly foaf:knows is only used in a triple
 - ▶ citec:jmccrae
foaf:knows
oeg:emontiel .
- ▶ It is necessary to state how these correspond.





- ▶ Linguistically we define each word as having a *subcategorization frame*
 - ▶ e.g., "X knows Y"
- ▶ Each RDF property has two arguments
 - ▶ *Subject*
 - ▶ *Object*
- ▶ We need to state the correspondence of syntactic arguments and semantic arguments



The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

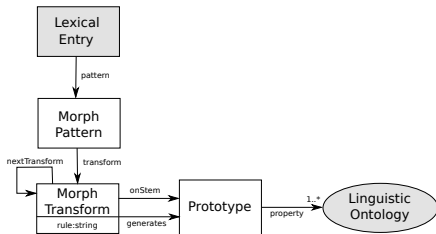
Morphology

Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

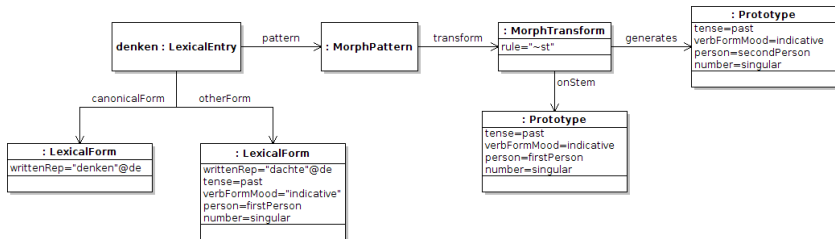
- ▶ Forms represented as new resources
- ▶ OK for English, not practical for languages with many regular forms
 - ▶ e.g., “love”, “loves”, “loved”
 - ▶ e.g., “amo”, “amas”, “amat”, “amamus”, “amatis”, “amant”, “amabam”, “amabas”, ...
- ▶ Represent forms by a generic pattern
- ▶ Defined by *Perl-like regular expression*



- ▶ German Mixed Verbs,
 - ▶ e.g., “denken” (“dachte”, “hat gedacht”)
 - ▶ “to think” (“thought”, “has thought”)
- ▶ Conjugation of past indicative (preterite) forms:

Ich (I)	Du (you, s.)	Er (he)	Wir (we)	Ihr (you, pl.)	Sie (they)
dachte	dachtest	dachte	dachten	dachtet	dachten

- ▶ Rule is as follows:
 - ▶ Generate form (past,indicative,secondPerson,singular)
 - ▶ By taking form (past,indicative,firstPerson,singular)
 - ▶ And apply rule “~st”
- ▶ The set of such rules is the *pattern* for German Mixed Verbs



The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

Morphology

Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

Morphology

Generation of lexica

Overview of Methodology

Introduction to lemon-Editor

- ▶ Standard NLP systems are used
 - ▶ Tokenizer
 - ▶ Stemmer
 - ▶ Part-of-speech tagger
 - ▶ Parser
 - ▶ Subcategorizer
- ▶ Output is mapped to appropriate **lemon** structure using LexInfo categories, e.g.,
 - ▶ NNS (*noun plural*, Penn Tagset) tag mapped to LexInfo noun, plural
 - ▶ Form added as otherForm
 - ▶ noun Added to lexical entry
 - ▶ plural Added to form
- ▶ For more details see my talk as part of the main conference (Wednesday Afternoon)

- ▶ Input is ontology
- ▶ Labels are extracted by looking for one of the following:
 - ▶ RDFS label
 - ▶ SKOS prefLabel
 - ▶ SKOS-XL prefLabel literalForm chain
 - ▶ A URI specified by the user
- ▶ Otherwise the label is deduced from the URI itself
 - ▶ e.g., `http://dbpedia.org/page/Crete` → “Crete”

The core

Beyond the core

Modeling examples

Linguistic Description

Variation

Phrase structure

Syntax and (Argument) Mapping

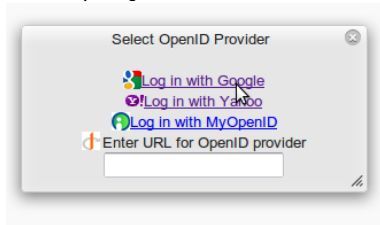
Morphology

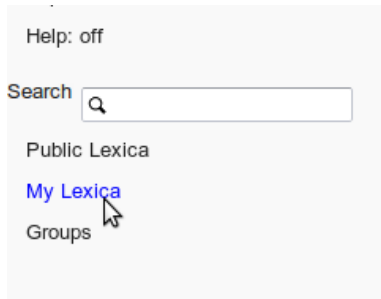
Generation of lexica

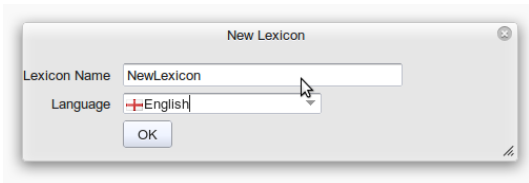
Overview of Methodology

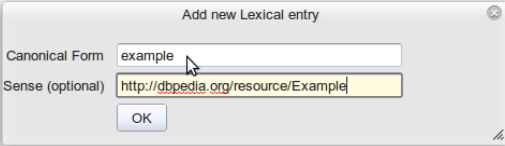
Introduction to lemon-Editor

`http://monnetproject.der1.ie/Lemon-Editor`









Add new Lexical entry

Canonical Form

Sense (optional)

OK

entry/example

Properties

+

Forms

Canonical

example

+

x

Sense

<http://dbpedia.org/page/Example>

+

x

Home

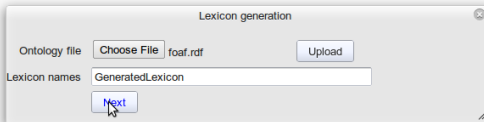
Generate

Import

Help: off

lemon source is a
To view the set o

Generate a new Lexicon using automatic values



The screenshot shows a dialog box titled "Lexicon generation". It contains two main sections. The first section, labeled "Ontology file", has a "Choose File" button followed by the text "foaf.rdf" and an "Upload" button. The second section, labeled "Lexicon names", has a text input field containing "GeneratedLexicon". Below this field is a "Next" button, which is highlighted by a mouse cursor. The dialog box has a standard close button (X) in the top right corner and a small icon in the bottom right corner.

