

Participants:

Bettina

Max

Julia

Maria

1) Describing the meanings of stem resources

Option one is to create a new property `morp:meaning` that interconnects a stem resource with an `ontolex:LexicalSense` resource. Option two is to create an automatic connection that assigns all senses of the `ontolex:LexicalEntry` instance - to which the `ontolex:Form` instances that contain the stems belongs - to the stem resources. E.g. the instance `ex:stem_play` is a part of `ex:form_plays` and should have the same senses as `ex:lex_play`. As a result each stem resource would have exactly the same `ontolex:LexicalSense` assignments as its corresponding `ontolex:LexicalEntry` resource.

There are counter-examples where this makes wrong predictions. Consider nominal incorporation as in Inuktitut:

`qimiq` n.abs. "dog"

`qimi(q)-u-` v. "to have a dog" (verb with incorporated noun)

`qimiutuq`

`qimi(q)-u-tuq`

`dog-INC-3s.VPART`

„(he) has a dog“

However, as the `3s.vpart` morpheme also serves to mark nominalization, an alternative gloss is „dog owner“ (n.abs). If the noun is ergative, this is the only reading:

`qimiutup`

`qimi(q)-u-tu(q)-(u)p`

`dog-INC-3s.VPART-ERG`

"dog owner (erg.)"

The point is that there could be a lexical entry for `qimiutuq` as dog owner, and a morphological segmentation, but we should not infer that a dog owner is a dog nor the other way around. Meanings of stems and lexemes must be clearly distinguished at all times. In European languages, this pattern is less systematic, but we do have lexicalization processes where stem meaning and lexeme meaning are quite far apart. Think of Italian `isolamento` "insulation", via `isolare` "insulate" from `isola` "island". An island may be isolated, but not necessarily in insulation.

Since the stem is the lexical-semantic core of all grammatical forms that can be build from a lexical entry this method would save time since the senses do not need to be repeated for each stem resource. However, there might be too many exceptions applying to this and result in wrong implications. Option 1, in contrast, might result in inconsistent datasets with stem resources pointing to different senses than its corresponding lexical entry. Comments

from people working with lexical datasets sharing their experiences and practical needs for representing the meanings of stems would be very helpful here.

I would in fact recommend to provide stem-level sense information only if it differs from lexical-entry level sense information, but it must be possible to do so. Responsibility for sense consistency is with the creator, not with the data model.

morph:meaning

Definition: assigning meanings to morph resources

Domain: morph:Morph

Range: ontolex:LexicalSense

2) Describing the meanings of grammatical affixes contained in ontolex:Form resources

According to the existing OntoLex-Lemon specification an example was created that applies to the usage recommendation of reusing the lexinfo vocabulary:

```
ex:suffix_s a morph:AffixMorph ;
```

```
lexinfo:number lexinfo:plural .
```

```
ex:suffix_s a morph:AffixMorph ;
```

```
morph:grammaticalMeaning lexinfo:plural .
```

morph:grammaticalMeaning

Definition: assigning meanings to morph resources

Domain: morph:Morph

Range: none

Model “no meaning” in an elegant generic way? CC any suggestions? :)

Not sure I understand the difference correctly, but lexinfo:number has a range restriction requiring that its objects are instances of lexinfo:Number. So, if anything, A) must be rephrased as follows:

A') It states that the meaning of the suffix -s resource is the lexinfo number value of plural.

vs.

B) It states that the suffix -s carries the lexinfo number value of plural.

With this reformulation, the difference seems to be that A') entails that there is no other meaning, whereas B) does permit that. In that interpretation, A') has counterexamples where a single morpheme expresses different functions at the same time: In Italian capi "heads", the morpheme -i expresses both plural number and masculine gender.

Derivational meanings of affixes:

- a) Additional lexical meaning, e.g. diminutive, agent noun.
→ use morph:meaning property
- b) Only wordclass change with additional marking, e.g. large → largely
→ use lexinfo or morph:grammaticalMeaning?
- c) without additional marking (zero derivation), e.g. to call → the call.
→ use lexinfo or morph:grammaticalMeaning? with morph:ZeroMorph instance

Defining the subclasses of morph:Morph

RootMorph:

StemMorph:

AffixMorph:

TransfixMorph:

SimulfixMorph:

ZeroMorph:

Find and use existing definitions. Maybe also use owl:definedBy and link to element that contains preferred definition.

Currently no suffix, prefix, circumfix classes. Give usage recommendation assigning them to either ontollex:Affix or morph:AffixMorph

Check the relations for the lexinfo suffix, prefix and circumfix to make sure that they are not inferred to be lexical entries.

(--> lemon:Part → lemon:LexicalEntry)