Theorien der Morphologie 8

Modul 006-1006: Grammatiktheorie, SoSe 2019

Di, 11:15–12:45, HSG, HS 20

Network Morphology

Gereon Müller (Universität Leipzig)

1. Network Morphology

Lit.:

Corbett & Fraser 1993, Baerman, Brown & Corbett (2005), Brown & Hippisley (2012)

Basic assumptions:

- Morphology is lexeme-based. (The rules of morphology are functions from lexical items into sets of inflected forms.)
- Morphology is inferential-realizational. (Inflection markers are not lexical entries.)
- Morphology relies on default inheritance. (DATR is used.)
- Morphological rules involve underspecification relative to the fully specified morphosyntactic paradigm.
- Generlized referral:
  1. One feature specification may refer to another feature specification for its realization.
  2. Referrals may be underspecified.
  3. Extensions of the referring specification will be realized by extensions of the referred-to specification.

A case study: A DATR fragment for Russian

(1) *Inflection class I, Sg.: masc*

	I		
	zavod$_m$ ('factory')	*student*$_m$ ('student')	*žitel*$_m$ ('inhabitant')
nom/sg	zavod-Ø	student-Ø	žitel'-Ø
acc/sg	zavod-Ø	student-a	žitel-ja
dat/sg	zavod-u	student-u	žitel-ju
gen/sg	zavod-a	student-a	žitel-ja
inst/sg	zavod-om	student-om	žitel-em
loc/sg	zavod-e	student-e	žitel-e

(2) *Inflection class II, Sg.: fem, masc*

	II			
	komnat$_f$ ('room')	*učitel'nic*$_f$ ('teacher')	*nedel'*$_f$ ('week')	*mužčin*$_m$ ('man')
nom/sg	komnat-a	učitel'nic-a	nedel-ja	mužčin-a
acc/sg	komnat-u	učitel'nic-u	nedel-ju	mužčin-u
dat/sg	komnat-e	učitel'nic-e	nedel-e	mužčin-e
gen/sg	komnat-y	učitel'nic-y	nedel-i	mužčin-y
inst/sg	komnat-oj(u)	učitel'nic-ej(u)	nedel-ej(u)	mužčin-oj(u)
loc/sg	komnat-e	učitel'nic-e	nedel-e	mužčin-e

(3) *Inflection class III, Sg.: fem*

	III		
	tetrad'$_f$ ('notebook')	*myš'*$_f$ ('mouse')	*doč'*$_f$ ('daughter')
nom/sg	tetrad'-Ø	myš'-Ø	doč'-Ø
acc/sg	tetrad'-Ø	myš'-Ø	doč'-Ø
dat/sg	tetrad-i	myš-i	doč-er-i
gen/sg	tetrad-i	myš-i	doč-er-i
inst/sg	tetrad'-ju	myš'-ju	doč-er'-ju
loc/sg	tetrad-i	myš-i	doč-er-i

(4) *Inflection class IV, Sg.: neut*

	IV			
	mest$_n$ ('place')	*jablok*$_n$ ('apple')	*suščestv*$_n$ ('creature')	*pol'*$_n$ ('field')
nom/sg	mest-o	jablok-o	suščestv-o	pol-e
acc/sg	mest-o	jablok-o	suščestv-o	pol-e
dat/sg	mest-u	jablok-u	suščestv-u	pol-ju
gen/sg	mest-a	jablok-a	suščestv-a	pol-ja
inst/sg	mest-om	jablok-om	suščestv-om	pol-em
loc/sg	mest-e	jablok-e	suščestv-e	pol-e

(5) *Inflection classes I-IV in the plural*

	I	II	III	IV
	zavod$_m$ ('factory')	*komnat*$_f$ ('room')	*tetrad'*$_f$ ('notebook')	*mest*$_n$ ('place')
nom/pl	zavod-y	komnat-y	tetrad-i	mest-a
acc/pl	zavod-y	komnat-y	tetrad-i	mest-a
dat/pl	zavod-am	komnat-am	tetrad-jam	mest-am
gen/pl	zavod-ov	komnat-Ø	tetrad-ej	mest-Ø
inst/pl	zavod-ami	komnat-ami	tetrad-jami	mest-ami
loc/pl	zavod-ax	komnat-ax	tetrad-jax	mest-ax

```
Mor-nominal:
 < >           == Mor-word
 <mor sg acc>  == Accusative:<sg “<syn gender>” “<syn animacy>”>
 <mor pl acc>  == Accusative:<pl “ <syn animacy>”>
 <mor pl nom>  == “<stem pl nom>” i “<stress pl nom>”
 <mor pl gen>  == “<mor pl loc>”
 <mor pl dat>  == “<stem pl>” “<mor theme-vowel>” “<stress pl>” m
 <mor pl inst> == “<stem pl>” “<mor theme-vowel>” “<stress pl>” m′i
 <mor pl loc>  == “<stem pl>” “<mor theme-vowel>” “<stress pl>” x.

Mor-noun:
 < >               == Mor-nominal
 <mor sg dat>      == “<mor sg loc>”
 <mor sg loc>      == “<stem sg>” e “<stress sg>”
 <mor pl gen>      == Mgp:<“mor stem hardness>” pl gen>
 <mor theme-vowel> == a.

N-0:
 < >           == Mor-noun
 <mor sg gen>  == “<stem sg gen>” a “<stress sg>”
 <mor sg dat>  == “<stem sg>” u “<stress sg>”
 <mor sg inst> == “<stem sg>” om “<stress sg>”.

N-I:
 < >                   == N-0
 <mor formal gender> == m
 <mor pl nom>        == Nom-pl:<“<index>”>
 <mor hard pl gen>   == “<stem pl>” ov “<stress pl>”.

N-IV:
 < >                   == N-0
 <mor formal gender> == n
 <mor sg nom>        == “<stem sg nom>” o “<stress sg>”
 <mor pl nom>        == “<stem pl>” a “<stress pl nom>”
 <mor pl gen>        == Gen-pl:<“<stem pl final>”>
```



```
N-II:
 < >                   == Mor-noun
 <mor formal gender> == f
 <mor sg nom>        == “<stem sg nom>” a “<stress sg>”
 <mor sg acc>        == “<stem sg>” u “<stress sg acc>”
 <mor sg gen>        == “<stem sg>” i “<stress sg>”
 <mor sg inst>       == “<stem sg inst>” “<mor vowel sg>” “<stress sg>” j′ (′u′)′
 <mor pl gen>        == Stemstress: <”<mor stem hardness>” “<stress pl>”>.

N-III:
 < >                   == Mor-noun
 <mor stem hardness> == soft
 <mor sg gen>        == N-II
 <mor sg loc>        == “<mor sg gen>”
 <mor sg inst>       == “<stem sg inst>” ju
 <mor formal gender> == f.
```

2. DATR: A Language for Lexical Knowledge Representation

2.1. Basic Notions

Lit.:
Evans & Gazdar (1996), Hippisley (2007)

Core feature of DATR: Default inheritance

(6) *Default Inheritance* (Hippisley (2007), after Evans & Gazdar):
In addition to conventional inference, DATR has a nonmonotonic notion of inference by default: Each definitional sentence about some node/path combination implicitly determines additional sentences about all the extensions to the path at that node for which no more specific definitional sentences exists in the theory.

(7) *Ontology*:

a. *Node*:
There is a a network of nodes. A node is a collection of closely related information. (Nodes begin with capital letters.)

b. *Path/Value pairs*:
Each node has associated with it a set of path/value pairs.

c. *Path*:
Sequence of *atoms* (atoms are primitives). Paths are enclosed in angle brackets: < >.

d. *Attributes*:
   Atoms in paths.
e. *Value*:
   Sequence of *atoms*.
f. *DATR descriptions*:
   Consist of *sentences*.
g. *Sentences*:
   Consist of *statements*.
h. *Statements*:
   Statements are equations consisting of node, path, and value information.
   (i) *Extensional statements*:
   Low-level facts about the knowledge representation system, using the *equality operator* =, derivable from:
   (ii) *Definitional statements*:
   Higher-level generalization expressed by the *equality operator* ==.

I think that an extensional statement describes some existing object whereas a definitional statement determines the conditions under which some object can exist.

A case study: English verb inflection.

2.2. Analysis A

```
Word1:
 <syn cat>  = verb
 <syn type> = main
 <syn form> = present participle
 <mor form> = love ing.
```

Note:
Values can be atomic or consist of sequences of atoms.

```
Word2:
 <syn cat>  = verb
 <syn type> = main
 <syn form> = passive participle
 <mor form> = love ed.
```

Problem:
This looks like an adequate description of the facts, but it does not capture *generalizations*, and it *duplicates* information.

Assumption:
Extensional statements (=) as in Word1 and Word2 can be derived from more general definitional statements (==).

2.3. Analysis B

```
Word1:
 <syn cat>  == verb
 <syn type> == main
 <syn form> == present participle
 <mor form> == love ing.
Word2:
 <syn cat>  == verb
 <syn type> == main
 <syn form> == passive participle
 <mor form> == love ed.
```

Problem:
Not a lot is gained. If some value is defined for a path with a definitional statement, the corresponding extensional statement also holds. (In one case, an object is described, in the other case, a possible object is described.) But now the door is open: Generalization can take place.

2.4. Analysis C

Note:
Now there can be other ways of describing values, rather than just simply specifying them: Values can be captured by *descriptors* that bring about *inheritance* of properties.

```
Verb:
 <syn cat>  == verb
 <syn type> == main.
Word1:
 <syn cat>  == Verb:<syn cat>
 <syn type> == Verb:<syn type>
 <syn form> == present participle
 <mor form> == love ing.
Word2:
 <syn cat>  == Verb:<syn cat>
 <syn type> == Verb:<syn type>
 <syn form> == passive participle
 <mor form> == love ed.
```

Note:
The right-hand side of the `<syn cat>` statement is not a direct value specification, but an inheritance descriptor: The value associated with `<syn cat>` at `Word1` is the same as the value associated with `<syn cat>` at `Verb`.

Problem:
Now the analysis is actually longer than the original one. However, it can be improved in two ways: first, as regards notation ("a syntactic trick"); second, by integrating defaults.

2.5. Analysis D

Notational simplification:
If the path on the right-hand side is the same as the path on the left-hand side, it can be omitted.

```
Verb:
 <syn cat>  == verb
 <syn type> == main.
Word1:
 <syn cat>  == Verb
 <syn type> == Verb
 <syn form> == present participle
 <mor form> == love ing.
Word2:
 <syn cat>  == Verb
 <syn type> == Verb
 <syn form> == passive participle
 <mor form> == love ed.
```

Note:
In the same way, notation can be simplified if the path on the right-hand side is different but the node is identical.

```
Come:
 <mor root> == come
 <mor past participle> == Come:<mor root>.
```

This becomes:

```
Come:
 <mor root> == come
 <mor past participle> == <mor root>.
```


2.6. Analysis E

Defaults:
Paths are sequences of attributes. Path *extension*: P_1 extends P_2 iff the attributes of P_1 occur in the same order at the left-hand end of P_2.

(8) *Two cases*:
For some path P_1 not defined at node N, there are two possible scenarios:

a. P_1 is not the extension of some path defined at N: There is no definition for P_1 at N.
b. P_1 is the extension of some path defined at N: P_1 extends some P_2 which is defined at N. Here, P_1 assumes a definition by default.
(If P_1 extends several paths defined at N, it takes its definition from the most specific – the longest – of the paths that it extends.)

```
Verb:
 <syn cat>  == verb
 <syn type> == main.
Word1:
 <syn>      == Verb
 <syn form> == present participle
 <mor form> == love ing.
Word2:
 <syn>      == Verb
 <syn form> == passive participle
 <mor form> == love ed.
```

Note:
Extensions of `<syn form>` obtain their definitions from `<syn form>` rather than from `<syn>`; here the default inheritance is overriden by the more specific path.

References

Baerman, Matthew, Dunstan Brown & Greville Corbett (2005): *The Syntax-Morphology Interface. A Study of Syncretism*. Cambridge University Press, Cambridge.

Brown, Dunstan & Andrew Hippisley (2012): *Network Morphology*. Cambridge University Press, Cambridge.

Corbett, Greville & Norman Fraser (1993): Network Morphology: A DATR Account of Russian Nominal Inflection, *Journal of Linguistics* 29, 113–142.

Evans, Roger & Gerald Gazdar (1996): DATR: A language for lexical knowledge representation, *Computational Linguistics* 22, 167–216.

Hippisley, Andrew (2007): Lexical Knowledge Representation and DATR. Ms., University of Surrey.