



Grammar Development with LFG and XLE

Miriam Butt
University of Konstanz

Last Time

- Integration of a Finite-State Morphological Analyzer
 - Sublexical Rules
 - Sublexical Entries
 - The -unknown entry
- The XLE Lexicon Lookup Model

This Time: Lesson 8

1. Verbal Complements: COMP and XCOMP
 - Finite Complements (COMP)
 - Non-finite Complements (XCOMP)
 - Control Verbs
 - Theory of Control and Complementation
2. Regression Testing (Testsuites) and Debugging

Types of Arguments

- So far we have been working with the following grammatical functions (GFs):

SUBJ, OBJ, OBJ2, OBL

- But there are also GFs corresponding to clausal arguments:

XCOMP, COMP

- COMP is generally used for finite *that*-clauses
- XCOMP is generally used for non-finite embedded clauses.

Types of Arguments

- You can read up on grammatical functions in LFG and how to test for them in the Dalrymple book, Ch. 2.

Complements: COMP

- Some verbs require finite clausal arguments:
 - *think*: Kim thinks [that the gorilla ate the bananas].
 - *say*: Kim said [that the gorilla ate the bananas].
 - *believe*: Kim believes [that the gorilla eats cake].
 - ...
- These arguments are treated as COMP in LFG at f-structure and as a CP at c-structure.
- Note that they are recursive
 - Kim thinks that Sandy said that the gorilla ate the bananas.
 - Kim thinks that Sandy said that the dog believes that the gorilla ate the bananas.

A (simplified) f-str with COMP

PRED 'believe<(^SUBJ),(^COMP)>'

SUBJ (PRED 'Kim'
NUM sg, PERS 3, CASE nom)

COMP (PRED 'eat<(^SUBJ),(^OBJ)>'

SUBJ (PRED 'gorilla'
NUM sg, PERS 3, CASE nom)

OBJ (PRED 'cake'
NUM sg, PERS 3, CASE acc)

Complements: XCOMP

- Some verbs require non-finite clausal arguments:
 - *persuade*: Kim persuaded the gorilla [to eat cake].
 - *promise*: Kim promised the gorilla [to eat cake].
 - *want*: Kim wanted [to eat cake.]
 - *want*: Kim wanted the gorilla [to eat cake.]
- These arguments are treated as XCOMP in LFG at f-structure and generally as a VP at c-structure.

Complements: XCOMP

- Note that they are recursive as well
 - Kim wanted to persuade Sandy to eat cake.
 - Kim wanted to persuade Sandy to promise the gorilla to eat cake.
 - ...
- They also interact with the COMPs
 - Kim thinks that Sandy wants to persuade Peter to believe that the gorilla ate cake.
 - Kim promised Sandy to believe that Peter persuaded the gorilla to eat cake.
 - ...

XCOMP: Functional Control

- The non-finite clauses are called XCOMPs in LFG because the embedded clause contains a open SUBJ argument that is **controlled** by an argument in the matrix clause.
- Which argument controls the embedded SUBJ is *lexically* determined.
- Who is doing the eating? – Sandy
Kim persuaded Sandy [___ to eat cake].
- Who is doing the eating? – Kim
Kim promised Sandy [___ to eat cake].

Open Functions

- XCOMP is considered to be an **open GF** because one of its arguments is not specified within the clause.

- XADJUNCT is its adjunct counterpart.

[__ Having eaten the cake], Kim ordered some tea.

- The subject of the embedded non-finite adjunct is controlled by an entity outside of the clause.

XCOMP: Functional Control

- Each control verb needs to specify in the lexicon whether it is a subject or an object control verb (*want* does both).
- Object control verb: *persuade*
Kim persuaded Sandy [___ to eat cake].
- Subject control verb: *promise*
Kim promised Sandy [___ to eat cake].

A (simplified) f-str with XCOMP – OBJ control

PRED 'persuade<(^SUBJ)(^ OBJ)(^XCOMP)>'

SUBJ {
PRED 'Kim'
NUM sg, PERS 3, CASE nom

OBJ {
PRED 'Sandy'
NUM sg, PERS 3, CASE acc

XCOMP {
PRED 'eat<(^SUBJ),(^OBJ)>'

SUBJ {

OBJ {
PRED 'cake'
NUM sg, PERS 3, CASE acc

A (simplified) f-str with XCOMP – SUBJ control

PRED 'promise<(^SUBJ)(^ OBJ)(^XCOMP)>'

SUBJ { PRED 'Kim'
NUM sg, PERS 3, CASE nom }

OBJ { PRED 'Sandy'
NUM sg, PERS 3, CASE acc }

XCOMP { PRED 'eat<(^SUBJ),(^OBJ)>'

SUBJ { }

OBJ { PRED 'cake'
NUM sg, PERS 3, CASE acc }

XCOMP: Functional Control

- The lexical specification is done by means of a **control equation**.

- Object control verb: *persuade*

$(\hat{\text{ XCOMP SUBJ}}) = (\hat{\text{ OBJ}})$

Kim persuaded Sandy [__ to eat cake].

- Subject control verb: *promise*

$(\hat{\text{ XCOMP SUBJ}}) = (\hat{\text{ SUBJ}})$

Kim promised Sandy [__ to eat cake]

Control and Complementation

- The basic theoretical concepts for understanding control and complementation in LFG were worked out in the early 1980s.
- Classic article:
 - Bresnan, Joan. 1982. Control and Complementation. In *The Mental Representation of Grammatical Relations*. The MIT Press.
 - Reprinted in: Bresnan, Joan. 1982. Control and Complementation. *Linguistic Inquiry* 11(3):343-434.

Demo

grammar7.lfg
testsuite7.lfg

**Grammar Engineering:
Testsuites and Debugging
(morphological analyzer)
COMP and XCOMP**

Practical Work

- This concludes Lesson 8.
- The practical work you should do now is detailed in Exercise 8.
- You will practice with
 - integrating verbal complements (COMP, XCOMP)
 - writing entries for control verbs
 - learn about raising verbs

