Syntax
Constituency, Projection and C-Selection

Modul 04-006-2002
Phonology – Morphology – Syntax

Institut für Linguistik

Universität Leipzig

home.uni-leipzig.de/heck

Recap

Recap Background:

- The linguistic (in particular: syntactic) knowledge of a speaker is under-determined by the input presented to it as a learner (poverty of the stimulus).
- An explanation for why the learner is nevertheless able to “acquire” language consists in assuming that they possess innate knowledge about what a possible grammar of human language may be: Universal Grammar.
- Linguistic objects (e.g. sentences) possess hierarchical structure that goes beyond the structure of linear order that is accessible in the acoustic signal.
- The theory of grammar offers a reconstruction of the linguistic knowledge of the speaker (I-language = UG + parametric settings) that defines the set of grammatical strings (observational adequacy) plus their appropriate structural descriptions (descriptional adequacy) and is compatible with observations from language acquisition (explanatory adequacy).

Constituency

The fine structure of clauses:

- Evidence from subject auxiliary inversion suggested: sentences do not just make linear strings of words but possess hidden hierarchical structure (embedded vs. non-embedded clauses).
- It turns out that this hierarchical organization of sentences is much more fine grained.
- Concretely, one can observe that some words form units together with other words, and to the exclusion of yet other words. Such units are called *constituents*.
- How do we know this? Because there are syntactic processes, such as replacement, displacement, deletion, and coordination, that make reference to these constituents.

On failed constituency tests

The one-way implication of constituency tests:

- Such syntactic operations can therefore be applied to probe into the hierarchical structure of sentences. But note that this only works in one direction.
- If a constituency test T is positive for a substring S, this is good evidence for the constituency of S.
- But if a constituency test T fails for a substring S, then one may *not* conclude that S is *not* a constituent.
- Reason: There may always be a independent constraint of grammar (unrelated to the constituency of S) that blocks application of T to S.

Constituency: replacement

Replacement:

- If a string S can be replaced by some other string S′ (e.g., a pronoun, a proper name, etc.), then S is likely to be a constituent (1-b,d).
- However, if S cannot be replaced, then this is not necessarily evidence against the constituency status of S (e.g., some strings may not be pronominalizable for some reason (2-a,b)).

(1)
  a. Frida blackmailed *the man who lived next door.*
  b. Frida blackmailed *him.*
  c. Mats believes *that he has won.*
  d. Mats believes *it.*

(2)
  a. *Frida blackmailed the man *it.*
  b. *Mats believes that *it.*

Constituency: displacement

Displacement:

- If a string S can be displaced (relative to its position in some other clause, marked by __ below), then S is likely to be a constituent (3-b,d).
- However, if S cannot be displaced, then this is not necessarily evidence against the constituency status of S (e.g., some strings may be immobile for some reason (4-b,d)).
- Note: a) Single words such as *Mats* make trivial constituents. b) The displacement in (3-c,d) simultaneously involves replacement.

(3) a. Frida blackmailed *Mats*.
  b. *Mats*, Frida blackmailed __.
  c. Mats does not eat *because he is not hungry*.
  d. *Why* doesn't Mats eat __?

(4) a. Mats believes that *he has won*.
  b. **He has won*, Mats believes that __.
  c. Frida adores *Mats'* bycicle.
  d. **Whose* does Frida adore __ bycicle?

Constituency: deletion

Deletion:

- If a string S can be deleted (the deletion site being marked by Δ below), then S is likely to be a constituent (5-a,b).
- However, if S cannot be deleted, then this is not necessarily evidence against the constituency status of S (e.g., some strings may not be deletable for some reason (6)).

(5) Frida might have blackmailed Mats, . . .

a. . . . , and Priesemut might have Δ, too.
(Δ = *blackmailed Mats*)

b. . . . , and Priesemut might Δ, too.
(Δ = *have blackmailed Mats*)

(6) *Frida wanted to blackmail Mats, . . . ,
. . . and Priesemut wanted Δ, as well.
(Δ = *to blackmail Mats*)

Constituency: coordination

Coordination:

- If a string S can be coordinated with another string S′, then S is likely to be a constituent (7-a,b).
- However, if S cannot be coordinated, then this is not necessarily evidence against the constituency status of S (e.g., sometimes S cannot be coordinated with S′ for some reason (8)).

(7) a. *Dr. Brumm* and *Pottwal* went for a hike.
b. Dr. Brumm *went hiking with Pottwal* and *got lost*.

(8) a. Dr. Brumm had *to work yesterday* and *to work today*.
b. Dr. Brumm had *worked yesterday* and *worked today*.
c. *Dr. Brumm had *to work yesterday* and *worked today*.

Note:
(8-c) is grammatical under a different reading, where *had to work yesterday* is coordinated with *worked today*.

Apparent contradictions

Note:

- In some cases, constituency tests apparently deliver contradictory results.
- (9-a) suggests that *have blackmailed* forms a constituent together with the object *Mats*, and to the exclusion of *might*.
- (9-b) suggests that *might* forms a constituent together with *have blackmailed*, and to the exclusion of the object *Nulli*.
- In order to solve the apparent contradiction one may conclude that the constituent structures underlying the strings in (9-a,b) differ in important respects.

(9) a. Frida might *have blackmailed Mats* . . .
. . . and Priesemut might Δ, too.
(Δ = *have blackmailed Mats*)

b. Frida *might have blackmailed* Mats . . .
. . . and Priesemut Δ Nulli.
(Δ = *might have blackmailed*)

A generative procedure

Task:
Define a mechanism that allows to characterize all the grammatical sentences of a given language and to assign to it appropriate structural descriptions.

Strategy:

- Syntactic structures (and the strings that are associated with them) are built up (*generated*) in a step-by-step fashion.
- The basic operation that realizes this process of structure building is the operation *Merge* in (10), as proposed in Chomsky (1993, 1995).

Aside:
The idea that the set of grammatical sentences is characterized by a generative procedure is what has lead to the coinage of the name "generative grammar". (However, for historical reasons purely representational approaches such as HPSG or LFG usually are also subsumed under this notion.)

Merge

(10) *Merge*:
Merge takes two terms α and β and combines them. The result is a new term $\gamma = \{\alpha, \beta\}$.

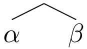
(11) *Term*:
α is a term iff a. or b. hold

a. α is a lexical item.
b. α has been generated by a previous application of Merge.

(12) Merge(α, β) $\rightarrow \{\alpha, \beta\}$ (=γ)

Comments:

- By hypothesis, Merge is binary: every constituent is made up from two elements. Whether this is true is an empirical question.
- The resulting constituent γ in (12) is defined as a set containing α and β. Alternative (by and largely equivalent) representations involve square brackets (13-a) and tree structures (13-b).

(13) a. [α β] b. 

A first example

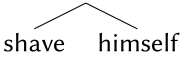
Example:

- Merge takes the two words *shave* and *himself* (out of the mental lexicon) and combines them, see (14).
- The result is a constituent *shave himself* (15) that can be affected by displacement (16-a) and deletion (16-b).

(14) Merge(shave, himself) $\rightarrow$ {shave, himself}

(15) a. [shave himself]

b.



(16) a. Dr. Brumm said that he will [shave himself] . . .
. . . and [shave himself] he will __.

b. Dr. Brumm [shaved himself], and Pottwal did Δ, too.
(Δ = *shave himself*)

Recursion

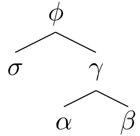
Recursion:

- Because of (11-b), Merge can apply in a recursive fashion. This means that the output of one application of Merge (a complex constituent) can become the input to another application of Merge (17).
- In this way, infinite many hierarchical structures can be generated, accounting for the creativity of language. The repeated application of Merge resulting in some constituent K is called a *derivation* of K.
- Constituents are *labeled* (here by γ, ϕ). In a bracketed structure, the label is an index on the outermost bracket pair (18-a). In a tree structure, it becomes the top node of the constituent (18-b).

(17)
a. Merge(α, β) $\rightarrow \{\alpha, \beta\}$ (= γ)
b. Merge (σ, γ) $\rightarrow \{\sigma, \{\alpha, \beta\}\}$ (= ϕ)
c. . . .

(18)
a. $[_{\phi}\ \sigma\ [_{\gamma}\ \alpha\ \beta\ ]]$

b.

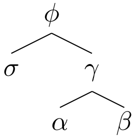


Terminology

Some useful terminology:

- α and β are called *daughters* of γ in (19); γ is the *mother* of α and β. α and β stand in a sisterhood relation. (Analog considerations apply to σ, γ, and ϕ.)
- ϕ in (19) *dominates* all nodes that are part of the tree (except for ϕ itself). Any mother is said to *immediately dominate* its daughters.
- ϕ is called the *root* node of (19); σ, α and β are *terminal* nodes, ϕ and γ are *non-terminal* nodes.

(19)



Headedness

Observation:

- The constituents in Italics in (20-a-d) all show up in the subject position (to the left of the modal *may*) of the clause. Moreover, they all contain a noun, namely *pigs*.
- (20-a-d) contrast in grammaticality with the examples in (21-a-c), where the subject position (to the left of the modal) is filled by the constituents *tomorrow*, *by Mary*, and *John slept*.

(20)
a. *Pigs* may love truffles.
b. *Those pigs* may love truffles.
c. *Those happy pigs* may love truffles.
d. *Some happy pigs that can fly* may love truffles.

(21)
a. **Tomorrow* may John work.
b. **By Mary* may have been killed John.
c. **John slept* may have.

Headedness

Interpretation:

- Assuming that the subject position of a clause must be filled by a noun, (20-a-d) suggest that the nouniness inherent to *pigs* in (20) must have become a property of the more inclusive constituent that occupies the subject position.
- Accordingly, (21-a-c) are ungrammatical because the constituents in subject position are *not* nouns (although two of them also contain a noun).

Aside:

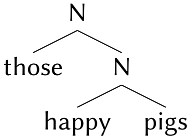
- In (21-b) *John* is linearized to the right of the predicate *killed*, which is perhaps surprising (cf. *John was killed*). However, it is not responsible for the ungrammaticality of (21-b). The justification for this will come later.

Headedness

The head:

- The label of a constituent K is not just an abstract name for K. Rather, it is a bundle of morpho-syntactic features, comprising a category feature.
- The label of K is determined by one of K's daughters, which is called the *head* of K. The head of K is also said to *project* its features onto K.
- In (22), *pigs* projects its category N onto the constituent *happy pigs*. From there, the category feature is projected further onto the root node (representing the constituent *those happy pigs*).
- By assumption, this is why all the constituents corresponding to the string in italics in (20) exhibit the property of being a noun.

(22)



Inclusiveness Condition

Inclusiveness Condition:

- The conclusion that the category feature of a complex constituent K must be projected by one of K's daughters is enforced by the assumption that the morpho-syntactic features of K must enter K via Merge of some lexical item.
- This assumption is formulated as a grammatical meta-principle, called the *Inclusiveness Condition* (Chomsky 1995).

(23) *Inclusiveness Condition*:
Morpho-syntactic features of a constituent K must have entered the derivation by Merge of a lexical item.

Headedness

Question:
Which principle of grammar determines which of the two daughters of a constituent K is the head of K?

Concretely:

- Why is it, for instance, possible for *pigs* in (24-a) to project its features onto the constituent *those pigs*, but impossible for *John* in (24-b) to project its features onto the constituent *John slept*?
- Recall that if the latter where possible, then the requirement that every subject position is filled by a noun would be fulfilled by (24-b).

(24) a. [Those pigs] may love truffles.
b. *[John slept] has.

Note:
There is a relatively recent discussion about this problem that runs under the name of “labeling” in the literature (starting with Chomsky 2013). For the moment, we stick to the discussion as presented in Adger (2003).

Θ-roles

Avant propos:
Before we can motivate the principle that determines headedness, we have to talk about predicates and their arguments.

Predicate and argument:

- Predicates denote semantic relations between “characters” that participate in the event/action/state described by the relation. Prototypical predicates are verbs.
- The characters are called *arguments*. The arguments of a predicate participate in the relation denoted by the predicate. Prototypical arguments are nouns.
- Every predicate needs a specific number of arguments in order to denote a complete proposition (become “saturated”).

Θ-roles

Types of predicates:
Predicates differ with respect to the number of arguments they require for saturation.

- Zero place predicate: *rain*
  It rained.
- One place predicate: *disappear*
  Pottwal disappeared.
- Two place predicate: *love*
  Dr. Brumm loves Pottwal.
- Three place predicate: *give*
  Dr. Brumm gives Pottwal the honey.
- Four place predicate: *bet*
  Dr. Brumm bets Pottwal a glass of honey that Dachs will lose.

Note:
The *it* showing up together with *rain* is not an argument of rain as it does not fulfill any semantic function in the event described by the predicate: There is, for instance, no-one that makes it rain.

Θ-roles

Types of Θ-roles:

- The specific function an argument fulfils in the event denoted by a predicate is determined by the *thematic role* (Θ-role) it is assigned by the predicate.
- A predicate assigns the number of Θ-roles that corresponds to its arity: Zero place predicates assign no Θ-role, one place predicates one Θ-role, etc.

Example:

- In (25), the predicate *devour* denotes an event of swallowing. The active character that participates in this event is the argument *Dr. Brumm*, which is assigned the Θ-role *agent*.
- The passive character, which simply undergoes the event, is the argument *the honey*. It is assigned the Θ-role *patient/theme*
- Other predicates may assign other Θ-roles.

(25) Dr. Brumm devours the honey.

Θ-criterion

The Θ-criterion:
Chomsky (1981) introduces the Θ-criterion in (26), the purpose of which is to account for the ungrammaticality of examples such as (26-a-c).

(26) *Θ-criterion*:

a. Every Θ-role must be realized by exactly one argument.
b. Every argument must realize exactly one Θ-role.

Comments:

- (27-a,b) both violate the Θ-criterion because there are two Θ-roles to be assigned but only one argument that can realize a Θ-Role.
- (27-c) violates the Θ-criterion because there is only one Θ-role assigned but there are two arguments, and therefore only one argument can realize a Θ-role.

(27)
a. *Dr. Brumm demonized.
b. *Shaved Dr. Brumm.
c. *Dr. Brumm slept Pottwal.

Autonomy of syntax

The autonomy of syntax:

- The Θ-criterion determines what syntactic structures should look like in order to be interpretable.
- One may speculate that this is what "drives" the application of Merge. However, it turns out that Merge is subject to purely syntactic principles and therefore cannot be conditioned by Θ-theoretic considerations (at least not fully so).
- This is expressed by the autonomy hypothesis in (28) (going back to Chomsky 1957).

(28) *Autonomy of syntax*:
Syntactic operations/principles apply independent from the semantic or phonological effects that they may have.

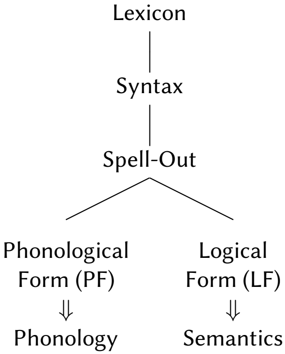
- To understand this better, it is helpful to have a look at the model of grammar that underlies the Minimalist Program.

General model

Model of grammar:

- Merge applies in the area labeled "Syntax" in (29), affecting elements taken from the lexicon (and constituents already created).
- At some point, Spell-Out applies, sending the structure generated by the syntax to the interfaces, where it is translated into formats that are readable by phonology (Phonological Form) and semantics (Logical Form).

(29)



C-Selection

Observation:
Information provided by a Θ-role θ is not sufficient to determine (in each and every case) the syntactic category of the argument that realizes θ.

Example:

- The (semantically similar) predicates *assure* and *object* both assign two Θ-roles, which may be labeled agent and theme.
- However, while both predicates may have their theme-role realized by a propositional argument (a clause, (30-b), (31-b)), only *assure* allows for its theme-role to be realized by a nominal argument.

(30) a. Pottwal assured his innocence.
b. Pottwal assured that he was innocent.

(31) a. *Pottwal objected his innocence.
b. Pottwal objected that he was innocent.

Conclusion:
Syntax must provide a mechanism to constrain the type of category of an argument that is merged with a predicate.

C-Selection

Assumptions:

- Every predicate P bears, as a lexical property, a number of c(ategory)-selection features that determine the categories of the constituents to be merged with P.
- Each θ-role of P is associated with such a c-selection feature. (But there may also be c-selection features that are not associated with Θ-roles.)
- What remains to be specified is how this feature-guided determination of Merge proceeds in detail.

Full Interpretation

Hypothesis:

- There is a meta-principle called *Full Interpretation* (Chomsky 1986, see (32)), which enforces deletion of C-selection features because they cannot be handled by the semantic interface (they are "uninterpretable").
- Deletion of c-selection features proceeds by "feature checking" (Chomsky 1995), a syntactic operation that pairs the c-selection feature of the predicate with the category feature of the argument that is merged with the predicate (33).
- By assumption, such feature checking only applies under the local syntactic configuration of sisterhood (which is why one needs feature projection).

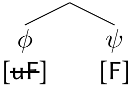
(32) *Full Interpretation*:
Syntactic objects that are send to the interfaces must not contain uninterpretable features.

Feature checking

(33) *Feature checking*:
If ϕ bears the uninterpretable feature [uF] and ψ bears its corresponding counter-part [F], then [uF] is deleted ([~~uF~~]) upon Merge(ϕ, ψ).

Consequence:
An argument of a predicate is merged within the projection of this predicate. Accordingly, also the Θ-roles of the predicate are assigned within its projection (cf. the *Head Constraint* by Jackendoff 2002).

Example:

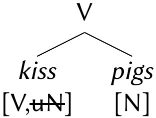
(34) Merge(ϕ[uF], ψ[F]) $\rightarrow$ 

(where, for instance, ϕ = *shave*, ψ = *himself*, F = N.)

Definition of headedness

(35) *Headedness*:
If a constituent K is the result of merging ϕ[uF] and ψ[F], then the head of K is ϕ.

Example:
The category of the constituent K = *kiss pigs* is supposed to be V because K comes about by Merge(*kiss*[uN], *pigs*[N]).

(36) Merge(*kiss*[V,uN], *pigs*[N]) $\rightarrow$ 

Definition of headedness

Prediction:
K = *kiss pigs* should have the same distribution as other, bona fide, verbal constituents, e.g., *sing*. This appears to be correct (37).

(37)
a. I want to $[_{V}$ *kiss pigs*].
b. I want to $[_{V}$ *sing*].
c. That I should $[_{V}$ *kiss pigs*] is my fondest dream.
d. That I should $[_{V}$ *sing*] is my fondest dream.
e. *$[_{V}$ *Kiss pigs*] is my happiest memory.
f. *$[_{V}$ *Sing*] is my happiest memory.

Bibliography

Adger, D. (2003). *Core Syntax.* Oxford University Press, Oxford.

Chomsky, N. (1957). *Syntactic Structures.* Mouton, The Hague.

Chomsky, N. (1981). *Lectures on Government and Binding.* Foris, Dordrecht.

Chomsky, N. (1986). *Knowledge of Language. Its Nature, Origine and Use.* Praeger, New York.

Chomsky, N. (1993). A minimalist program for linguistic theory. In Hale, K. and Keyser, S. J., editors, *The View from Building 20*, pages 1–52. MIT Press, Cambridge, Massachusetts.

Chomsky, N. (1995). *The Minimalist Program.* MIT Press, Cambridge, Massachusetts.

Chomsky, N. (2013). Problems of projection. *Lingua*, 130:33–49.

Jackendoff, R. (2002). *Foundations of Language.* Oxford University Press, Oxford.