$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \textit{Grammatik} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{语法} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{دستور} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{व्याकरण} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Organizational matters

- Please register via Moodle

Organizational matters

- Please register via Moodle
- Phone and office hours see: https://hpsg.hu-berlin.de/~stefan/

Organizational matters

- Please register via Moodle
- Phone and office hours see: https://hpsg.hu-berlin.de/~stefan/
- Complaints and suggestions:
  - in person
  - via mail
  - anonymously via the web:
    https://hpsg.hu-berlin.de/~stefan/Lehre/
- Please stick to the mail rules!
  https://hpsg.hu-berlin.de/~stefan/Lehre/mailregeln.html

Documents

- Course information:
  https://hpsg.hu-berlin.de/~stefan/Lehre/GT/

  Textbook: Müller, Stefan (2020b), *Grammatical Theory* (Textbooks in Language Science 1). Berlin: Language Science Press fourth edition. https://langsci-press.org/catalog/book/287

  A bit outdated: Müller, Stefan (2013a) *Grammatiktheorie*, (Stauffenburg Einführungen 20). Tübingen: Stauffenburg Verlag zweite Auflage. http://hpsg.hu-berlin.de/~stefan/Pub/grammatiktheorie.html

General idea in Corona times

1. Read the respective sections in the textbook.
2. Slides with spoken comments can be found in moodle.
   Please watch them before the lesson.
3. You can do 1 and 2 in your preferred order.
4. Use the online tasks to check whether you understand everything.
5. Use quick questions and exercises in the book.
6. Ask questions during the online sessions!

Leistungen

Master Linguistik, Modul 2: Theoretische Grundlagen II, 2 SWS

- Aktive Teilnahme, Vor- und Nachbereitung
- Klausur (im Modul für Linguistik)

Ideale Zeitaufteilung:

Präsenzstudium Vorlesung	25 h
Vor- und Nachbereitung	95 h (35/15 = 2 h 20 min für jede Sitzung + 60h Prüf)
Klausurvorbereitung	

Für die Veranstaltung gibt es 4 Leistungspunkte.

Recapitulation

- Linguistics 101 in the BA (4 SWS)
- Tutorial Linguistics 101

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \textit{Grammatik} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{语法} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{دستور} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{व्याकरण} \ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Motivation of (formal) syntax and basic terminology

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Reading material

- Literature: English version of the grammatical theory textbook: Müller 2020b

Reading material

- Literature: English version of the grammatical theory textbook: Müller 2020b
- There is also a German and a Chinese version.
  The fourth edition of the English book is the most recent one.

Reading material

- Literature: English version of the grammatical theory textbook: Müller 2020b
- There is also a German and a Chinese version.
  The fourth edition of the English book is the most recent one.
- For this session, please read Müller 2020b: Chapter 1.
  Topological fields are covered in Section 1.8. They are not part of the slides of this session but will be needed later on (chapter 3 and onwards).

Goals of this course

- conveyance of basic ideas about grammar

Goals of this course

- conveyance of basic ideas about grammar
- introduction to various grammatical theories and approaches

Goals of this course

- conveyance of basic ideas about grammar
- introduction to various grammatical theories and approaches
- enlightenment and attainment of supernatural powers

Ancient wisdom

[Grammar is] the gate to freedom, the medicine for the diseases of language, the purifier of all sciences; it spreads its light over them; … it is the first rung on the ladder which leads to the realization of supernatural powers and straight, royal road for those who seek freedom. (Bhartrhari, poet of sayings, died before 650 AD, from *Vakyapadiya*, found by Gabriele Knoll)

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1
- signs: form-meaning pairs (de Saussure 1916)

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1

- signs: form-meaning pairs (de Saussure 1916)
- words, word groups, sentences

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1

- signs: form-meaning pairs (de Saussure 1916)
- words, word groups, sentences

- language $\stackrel{?}{=}$ finite enummeration of word sequences

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1

- signs: form-meaning pairs (de Saussure 1916)
- words, word groups, sentences

- language $\stackrel{?}{=}$ finite enummeration of word sequences
  language is finite, if onw assumes a maximal sentence length

(1) a. This sentence goes on and on and on …

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1
- signs: form-meaning pairs (de Saussure 1916)
- words, word groups, sentences
- language $\stackrel{?}{=}$ finite enummeration of word sequences
  language is finite, if onw assumes a maximal sentence length

(1) a. This sentence goes on and on and on …
  b. [A sentence is a sentence] is a sentence.

Why syntax?

- Literature: Müller 2013b: Chapter 1 or Müller 2013a: Chapter 1

- signs: form-meaning pairs (de Saussure 1916)
- words, word groups, sentences

- language $\stackrel{?}{=}$ finite enummeration of word sequences
  language is finite, if onw assumes a maximal sentence length

  (1) a. This sentence goes on and on and on …
      b. [A sentence is a sentence] is a sentence.

  We can form enourmously many sentences.
  A restriction on complexity would be arbitrary.

- One distinguishes between competence (knowledge about what is possible) and performance (useage of this knowledge)

The Six Bullerby Children

Und wir beeilten uns, den Jungen zu erzählen, wir hätten von Anfang an gewußt, daß es nur eine Erfindung von Lasse gewesen sei. Und da sagte Lasse, die Jungen hätten gewußt, daß wir gewußt hätten, es sei nur eine Erfindung von ihm. Das war natürlich gelogen, aber vorsichtshalber sagten wir, wir hätten gewußt, die Jungen hätten gewußt, daß wir gewußt hätten, es sei nur eine Erfindung von Lasse. Und da sagten die Jungen – ja – jetzt schaffe ich es nicht mehr aufzuzählen, aber es waren so viele „gewußt", daß man ganz verwirrt davon werden konnte, wenn man es hörte. (p. 248)

We are capable of forming long, complex sentences (competence), but at some level of complexity we get confused since our brains cannot deal with the complexity anymore (performance).

Creativity

- We can form sentences we never heard before →
  There has to be structure, patterns.
  It cannot be just sequences learned by heart.

Direct evidence for syntactic structures?

- We can show that we are following rules by observing children.
  Children often use rules wrongly (or rather use their own rules).

Direct evidence for syntactic structures?

- We can show that we are following rules by observing children.
  Children often use rules wrongly (or rather use their own rules).
- Example from morphology:
  German has an unmarked Plural for some nouns: *Bagger* 'digger', *Ritter* 'knight'.
- Children apply the -*s* ending to such unmarked plurals instead:

(2) a. * die Baggers
    b. * die Ritters

Direct evidence for syntactic structures?

- We can show that we are following rules by observing children.
  Children often use rules wrongly (or rather use their own rules).
- Example from morphology:
  German has an unmarked Plural for some nouns: *Bagger* 'digger', *Ritter* 'knight'.
- Children apply the -*s* ending to such unmarked plurals instead:

(2) a. * die Baggers
    b. * die Ritters

- Side remark: We will use German examples throughout this course, since English is sooooo boring. I gloss whatever I can, but sometimes stuff would not fit onto the slide. Please refer to the textbook in such cases.

Why syntax? Computation of meaning from utterance parts

- The meaning of an utterance can be computed from the meaning of its parts.

(3) Der Mann kennt diese Frau.
the man knows this woman

Why syntax? Computation of meaning from utterance parts

- The meaning of an utterance can be computed from the meaning of its parts.

(3)

Der	Mann	kennt	diese	Frau.
the	man	knows	this	woman

- Syntax: the way parts are combined, the utterance is structured

(4) a.

Die	Frau	kennt	die	Mädchen.
the	woman	know.3SG	the	girls

‘The woman knows the girls.’

b.

Die	Frau	kennen	die	Mädchen.
the	woman	know.3PL	the	girls

‘The girls know the woman.’

Why syntax? Computation of meaning from utterance parts

- The meaning of an utterance can be computed from the meaning of its parts.

(3) Der Mann kennt diese Frau.
the man knows this woman

- Syntax: the way parts are combined, the utterance is structured

(4) a. Die Frau kennt die Mädchen.
the woman know.3SG the girls
'The woman knows the girls.'

b. Die Frau kennen die Mädchen.
the woman know.3PL the girls
'The girls know the woman.'

c. Die Frau schläft.
the woman sleep.3SG
'The woman sleeps.'

d. Die Mädchen schlafen.
the girls sleep.3PL
'The girls sleep.'

Subject-verb agreement → meaning of (4a,b) is unambiguous

Why formal?

Precisely constructed models for linguistic structure can play an important role, both negative and positive, in the process of discovery itself. By pushing a precise but inadequate formulation to an unacceptable conclusion, we can often expose the exact source of this inadequacy and, consequently, gain a deeper understanding of the linguistic data. More positively, a formalized theory may automatically provide solutions for many problems other than those for which it was explicitly designed. Obscure and intuition-bound notions can neither lead to absurd conclusions nor provide new and correct ones, and hence they fail to be useful in two important respects. I think that some of those linguists who have questioned the value of precise and technical development of linguistic theory have failed to recognize the productive potential in the method of rigorously stating a proposed theory and applying it strictly to linguistic material with no attempt to avoid unacceptable conclusions by ad hoc adjustments or loose formulation. (Chomsky 1957: 5)

As is frequently pointed out but cannot be overemphasized, an important goal of formalization in linguistics is to enable subsequent researchers to see the defects of an analysis as clearly as its merits; only then can progress be made efficiently. (Dowty 1979: 322)

- What does an analysis mean?
- Which predictions does it make?
- exclusion of alternative proposals

Grouping words

- Sentences may contain sentences containing sentences die …:

  (5) that Max thinks [that Julius knows [that Otto claims [that Karl suspects [that Richard confirms [that Friederike is laughing]]]]]

  This works like a Russian doll or like an onion.

Grouping words

- Sentences may contain sentences containing sentences die …:

(5) that Max thinks [that Julius knows [that Otto claims [that Karl suspects [that Richard confirms [that Friederike is laughing]]]]]

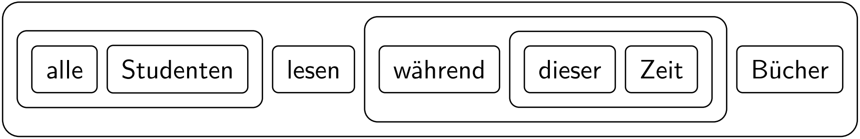
This works like a Russian doll or like an onion.

- The words in (6) can be grouped into units as well:

(6) Alle Studenten lesen während dieser Zeit Bücher.
all students read during this time books
'All the students are reading books at this time.'

Which ones?

Boxes



We put all words belonging together into a box.

Such boxes can be put into other boxes.

It is intuitively clear what belongs into a box in the example at hand, but are there tests?

Constituency

Terminology:

Word sequence An arbitrary linear sequence of words which do not necessarily need to have any syntactic or semantic relationship.

Word group, constituent, phrase One or more words forming a structural unit.

Constituency tests

Which ones do you know?

Constituency tests

Which ones do you know?

- substitution/pronominalization/question formation
- omission
- permutation
- fronting
- coordination

Constituency tests (I)

Substitution If it is possible to replace a sequence of words in a sentence with a different sequence of words and the acceptability of the sentence remains unaffected, then this constitutes evidence for the fact that each sequence of words forms a constituent.

(7) a. Er kennt [den Mann].
he knows the man
'He knows the man.'

b. Er kennt [eine Frau].
he knows a woman
'He knows a woman.'

Constituency tests (II)

Pronominalization Everything that can be replaced by a pronoun forms a constituent.

(8) a. [Der Mann] schläft.
the man sleeps
'The man is sleeping.'

b. Er schläft.
he sleeps
'He is sleeping.'

Constituency tests (III)

Question formation A sequence of words that can be elicited by a question forms a constituent.

(9) a. [Der Mann] arbeitet.
the man works
'The man is working.'

b. Wer arbeitet?
who works
'Who is working?'

Constituency tests (IV)

Permutation test If a sequence of words can be moved without adversely affecting the acceptability of the sentence in which it occurs, then this is an indication that this word sequence forms a constituent.

(10) a. dass keiner [dieses Kind] kennt
that nobody this child knows

b. dass [dieses Kind] keiner kennt
that this child nobody knows
'that nobody knows this child'

Constituency tests (V)

Fronting Fronting is a further variant of the movement test. In German declarative sentences, only a single constituent may normally precede the finite verb:

(11) a. [Alle Studenten] lesen während der vorlesungsfreien Zeit
all students read.3PL during the lecture.free time
Bücher.
books
'All students read books during the semester break.'

b. [Bücher] lesen alle Studenten während der vorlesungsfreien Zeit.
books read all students during the lecture.free time

c. * [Alle Studenten] [Bücher] lesen während der vorlesungsfreien Zeit.
all students books read during the lecture.free time

d. * [Bücher] [alle Studenten] lesen während der vorlesungsfreien Zeit.
books all students read during the lecture.free time

Constituency tests (VI)

Coordination test If two sequences of words can be conjoined then this suggests that each sequence forms a constituent.

(12) [Der Mann] und [die Frau] arbeiten.
the man and the woman work.3PL
'The man and the woman work.'

Warning

Danger!

These tests are not 100 % reliable. See Müller 2020b: Section 1.3.2 for details.

For more on the tests see also Müller 2019: Section 2.

Heads

A head determines the most important properties of a phrase.

(13) a. Träumt dieser Mann?
dreams this.NOM man
'Does this man dream?'

b. Erwartet er diesen Mann?
expects he.NOM this.ACC man
'Is he expecting this man?'

c. Hilft er diesem Mann?
helps he.NOM this.DAT man
'Is he helping this man?'

d. in diesem Haus
in this.DAT house

e. ein Mann
a.NOM man

Projection

The combination of a head with other material is called projection of the head.

Projection

The combination of a head with other material is called projection of the head.

A complete projection is a maximal projection.

Projection

The combination of a head with other material is called projection of the head.

A complete projection is a maximal projection.

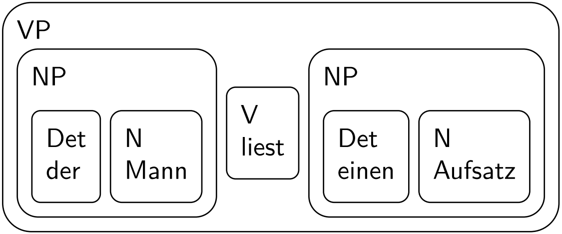
A maximal projection of a finite verb is a sentence.

Labeled boxes

Those of you who moved to a new flat know that is is good to label your boxes.

Labeled boxes

Those of you who moved to a new flat know that is is good to label your boxes.



The label on a box indicates the most important element in the box.

Boxes are replaceable

- It does not matter what exactly is in the box:

(14)

a. er
   he

b. der Mann
   the man

c.

der	Mann	aus	Stuttgart
the	man	from	Stuttgart

d.

der	Mann	aus	Stuttgart,	den	wir	kennen
the	man	from	Stuttgart	who	we	know

The only thing that matters:
all words or phrases in (14) are nominal and complete: NP.
They can be substituted for each other within bigger boxes.

Boxes are replaceable. Well, hm.

- This does not work with all NPs:

(15)
a. Der Mann liest einen Aufsatz.
the man reads an essay

b. * Die Männer liest einen Aufsatz.
the men reads an essay

c. * Des Mannes liest einen Aufsatz.
the man.GEN reads an essay

- Certain properties are important for the distribution of phrases.

More carefully labeled boxes

- VP, fin
  - NP, nom, 3, sg, mas
    - Det, nom, sg, mas
der
    - N, nom, sg, mas
Mann
  - V, fin, 3, sg
liest
  - NP, akk, 3, sg, mas
    - Det, akk, sg, mas
einen
    - N, akk, sg, mas
Aufsatz

All features that are important for the distribution of the whole phrase are projected.

Such feature are called head features.

Arguments

- Constituents are in different relations with their head.

Arguments

- Constituents are in different relations with their head.
- There are arguments and adjuncts.

Arguments

- Constituents are in different relations with their head.
- There are arguments and adjuncts.
- Certain elements are part of the meaning of a verb.
  For example in situations described by the verb *love*,
  there is a lover and a *lovee*.

(16) a. Kim loves Sandy.
     b. $love'(Kim', Sandy')$

(16b) is a logical representation of (16a).
Kim′ and *Sandy′* are logical arguments of *love′*.

Arguments

- Constituents are in different relations with their head.
- There are arguments and adjuncts.
- Certain elements are part of the meaning of a verb.
  For example in situations described by the verb *love*,
  there is a lover and a *lovee*.

  (16) a. Kim loves Sandy.
  b. $love'(Kim', Sandy')$

  (16b) is a logical representation of (16a).
  Kim′ and *Sandy′* are logical arguments of *love′*.
- Syntactic arguments usually correspond to logical arguments (more on this later).

Arguments

- Constituents are in different relations with their head.
- There are arguments and adjuncts.
- Certain elements are part of the meaning of a verb.
  For example in situations described by the verb *love*,
  there is a lover and a *lovee*.

  (16) a. Kim loves Sandy.
  b. $love'(Kim', Sandy')$

  (16b) is a logical representation of (16a).
  Kim′ and *Sandy′* are logical arguments of *love′*.
- Syntactic arguments usually correspond to logical arguments (more on this later).
- The term for such relations between head and arguments is selection or valence.

Arguments

- Constituents are in different relations with their head.
- There are arguments and adjuncts.
- Certain elements are part of the meaning of a verb.
  For example in situations described by the verb *love*,
  there is a lover and a *lovee*.

  (16) a. Kim loves Sandy.
  b. $love'(Kim', Sandy')$

  (16b) is a logical representation of (16a).
  Kim′ and *Sandy′* are logical arguments of *love′*.
- Syntactic arguments usually correspond to logical arguments (more on this later).
- The term for such relations between head and arguments is selection or valence.
- Tesnière (1959) transferred the concept of valence from chemistry to linguistics.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.
- The number of electrons on an electron shell is important for the stability of the molecule.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.
- The number of electrons on an electron shell is important for the stability of the molecule.
- If atoms combine with other atoms this can lead to completely filled electron layers, which would result into a stable compound.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.
- The number of electrons on an electron shell is important for the stability of the molecule.
- If atoms combine with other atoms this can lead to completely filled electron layers, which would result into a stable compound.
- The valency of an atom is the number of hydrogen atoms that can be combined with an atom of a certain element.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.
- The number of electrons on an electron shell is important for the stability of the molecule.
- If atoms combine with other atoms this can lead to completely filled electron layers, which would result into a stable compound.
- The valency of an atom is the number of hydrogen atoms that can be combined with an atom of a certain element.
- Oxygen has the valency 2 since it can be combined with two hydrogen atoms: H_2O.

Valency in chemistry

- Atoms can form more or less stable molecules with other atoms.
- The number of electrons on an electron shell is important for the stability of the molecule.
- If atoms combine with other atoms this can lead to completely filled electron layers, which would result into a stable compound.
- The valency of an atom is the number of hydrogen atoms that can be combined with an atom of a certain element.
- Oxygen has the valency 2 since it can be combined with two hydrogen atoms: H_2O.
- The elements can be grouped into valence classes.
  Elements with a certain valence are represented in a column in the periodice system of Mendeleev.

Valence in linguistics

- A head needs certain arguments to enter a stable compound.
- Words having the same valence (same number and type of arguments) are grouped into valence classes, since they behave alike with respect to the combinations they enter.

Combining oxygen with hydrogen and combining a verb with its arguments

Optional arguments

- Sometimes arguments may be omitted:

(17) a. I am waiting for my man.
     b. I am waiting.

Optional arguments

- Sometimes arguments may be omitted:

(17) a. I am waiting for my man.
b. I am waiting.

The prepositional object of *wait* is an optional argument.

Optional arguments

- Sometimes arguments may be omitted:

(17) a. I am waiting for my man.
b. I am waiting.

The prepositional object of *wait* is an optional argument.

- All arguments are optional in nominal environments.

(18) a. Jemand liest diese Bücher.
somebody reads these books

b. das Lesen dieser Bücher
the reading of.these books

c. das Lesen
the reading

Syntactic arguments that are not logical ones

- Syntactic arguments correspond to logical arguments in our example above:

(19) a. Kim loves Sandy.
b. $love'(Kim', Sandy')$

Syntactic arguments that are not logical ones

- Syntactic arguments correspond to logical arguments in our example above:

(19) a. Kim loves Sandy.
b. $love'(Kim', Sandy')$

- There are also arguments not contributing semantically:

(20) a. Es regnet.
it rains

b. Kim erholt sich.
Kim recreates SELF

es and *sich* are syntactic arguments, without being logical arguments.

Arguments and adjuncts

- Adjuncts do not fill a semantic role
- Adjuncts are optional
- Adjuncts can be iterated

Adjuncts do not fill a semantic role

- In a *loving* situation there is a lover and a lovee.
  since three years in (21) is of a different type:

  (21) Kim loves Sandy since three years.

  This phrase provides information about the span in which the relation between Kim and Sandy holds.

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

- Be aware! Arguments may also be optional:

(23) a. Er gibt den Armen Geld.
b. Er gibt den Armen.

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

- Be aware! Arguments may also be optional:

(23) a. Er gibt den Armen Geld.
b. Er gibt den Armen.
c. Er gibt Geld.

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

- Be aware! Arguments may also be optional:

(23) a. Er gibt den Armen Geld.
b. Er gibt den Armen.
c. Er gibt Geld.
d. Er gibt gerne.

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

- Be aware! Arguments may also be optional:

(23) a. Er gibt den Armen Geld.
b. Er gibt den Armen.
c. Er gibt Geld.
d. Er gibt gerne.
e. Du gibst. (beim Skat)

Adjuncts are optional

- Adjuncts are optional:

(22) a. Kim loves Sandy.
b. Kim loves Sandy since three years.
c. Kim loves Sandy honestly.

- Be aware! Arguments may also be optional:

(23) a. Er gibt den Armen Geld.
b. Er gibt den Armen.
c. Er gibt Geld.
d. Er gibt gerne.
e. Du gibst. (beim Skat)
f. Gib!

Adjuncts can be iterated

- Arguments can be combined with their head once:

  (24) * The man the man sleeps

  The respective slot of the head (*sleeps*) is filled.

Adjuncts can be iterated

- Arguments can be combined with their head once:

  (24) * The man the man sleeps

  The respective slot of the head (*sleeps*) is filled.

- But adjuncts are different:

  (25) A: All grey squirrels are big.
  B: No, I saw a small grey squirrel.
  A: But all small grey squirrels are ill.
  B: No, I saw a healthy small grey squirrel.
  ...

Some further examples for adjuncts

Adverbially used adjective (not all adjectives):

(26) Karl schnarcht *laut*.
Karl snores loudly

Relative clauses (not all of them):

(27) das Kind, *dem der Delphin hilft*
the child who the dolphin helps

Prepositional phrases (not all of them):

(28) a. Die Frau arbeitet *in Berlin*.
the woman works in Berlin

b. die Frau *aus Berlin*
the woman from Berlin

Various grammatical theories (I)

- Dependency Grammar (DG)
  (Tesnière 1980; 2015; Kunze 1975; Weber 1997; Heringer 1996; Eroms 2000)
- Categorial Grammar (CG)
  (Ajdukiewicz 1935; Steedman 2000)
- Phrase structure grammar (PSG)
- Transformational Grammar and its successors
  - Transformational grammar
    (Chomsky 1957; Bierwisch 1963)
  - Government & Binding
    (Chomsky 1981; von Stechow & Sternefeld 1988; Grewendorf 1988)
  - Minimalism
    (Chomsky 1995; Grewendorf 2002)

Various grammatical theories (II)

- Tree Adjoning Grammar
  (Joshi, Levy & Takahashi 1975; Joshi 1987; Kroch & Joshi 1985)
- Generalized Phrase Structure Grammar (GPSG)
  (Gazdar, Klein, Pullum & Sag 1985; Uszkoreit 1987)
- Lexical Functional Grammar (LFG)
  (Bresnan 1982a; 2001; Berman & Frank 1996; Berman 2003)
- Head-Driven Phrase Structure Grammar (HPSG)
  (Pollard & Sag 1987; 1994; Müller 1999; 2002; 2013b)
- Construction Grammar (CxG)
  (Fillmore, Kay & O'Connor 1988; Goldberg 1995; 2006; Fischer & Stefanowitsch 2006)

- We will deal with most of these in this course.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \textit{Grammatik} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{语法} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{دستور} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{व्याकरण} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Phrase structure grammars

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

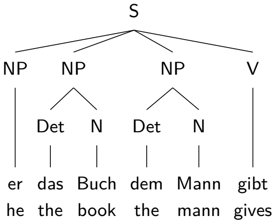
Outline

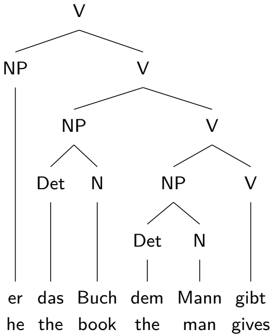
- Introduction and basic terms
- Phrase structure grammar and $\overline{\text{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoning Grammar (TAG)

Reading material

Please read Müller 2020b: Section 2.1–2.2.

Phrase structure



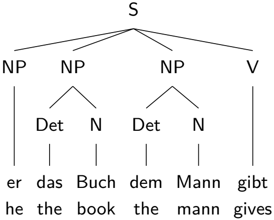


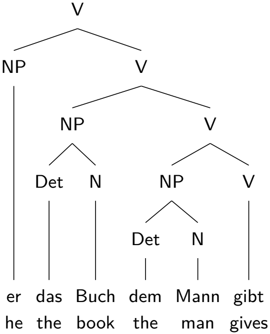
NP → Det, N
S → NP, NP, NP, V

NP → Det, N
V → NP, V

What we are after is phrase structure rules! Trees are just their visualization.

Phrase structure



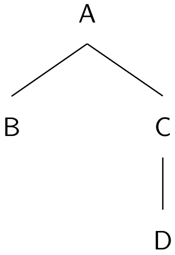


NP → Det, N
S → NP, NP, NP, V

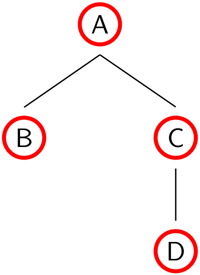
NP → Det, N
V → NP, V

What we are after is phrase structure rules! Trees are just their visualization.
Sometimes bracketed strings are used to safe space:
[$_{\text{S}}$ [$_{\text{NP}}$ er] [$_{\text{NP}}$ [$_{\text{Det}}$ das] [$_{\text{N}}$ Buch]] [$_{\text{NP}}$ [$_{\text{Det}}$ dem] [$_{\text{N}}$ Mann]] [$_{\text{V}}$ gibt]]

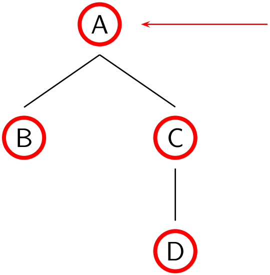
Node



Node



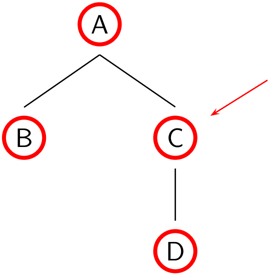
Node



branching

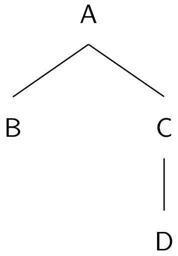
non-branching

Node



branching
non-branching

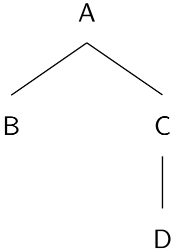
Mother, daughter and sister



A is mother of B and C
C is mother of D
B is sister of C

Relationships like in family trees

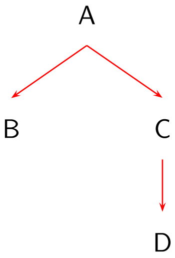
Dominance



A dominates

A dominates B if and only if A is higher in the tree and
if there is a line from A to B that exclusively goes downwards.

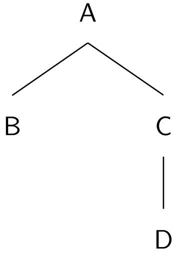
Dominance



A dominates B, C and D

A dominates B if and only if A is higher in the tree and
if there is a line from A to B that exclusively goes downwards.

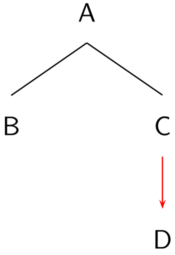
Dominance



A dominates B, C and D
C dominates

A dominates B if and only if A is higher in the tree and
if there is a line from A to B that exclusively goes downwards.

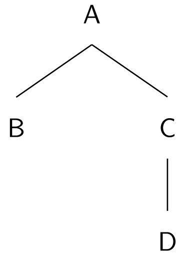
Dominance



A dominates B, C and D
C dominates D

A dominates B if and only if A is higher in the tree and
if there is a line from A to B that exclusively goes downwards.

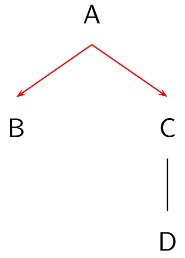
Immediate dominance



A immedeately dominates

A immedeately dominates B if and only if
A dominates B and there is no node C between A and B.

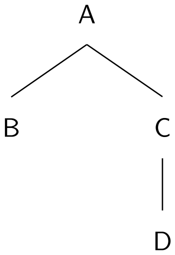
Immediate dominance



A immedeately dominates B and C

A immedeately dominates B if and only if
A dominates B and there is no node C between A and B.

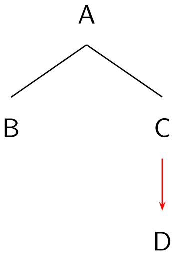
Immediate dominance



A immedeately dominates B and C
C immedeately domminates

A immedeately dominates B if and only if
A dominates B and there is no node C between A and B.

Immediate dominance



A immedeately dominates B and C
C immedeately domminates D

A immedeately dominates B if and only if
A dominates B and there is no node C between A and B.

Precedence

Precedence

A precedes B, if A is located to the left of B in a tree and none of these nodes dominates the other one.

Precedence

Precedence
: A precedes B, if A is located to the left of B in a tree and none of these nodes dominates the other one.

Immediate precedence
: A precedes B and there is no element C between A and B.

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er das Buch dem Kind gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt

Example derivation assuming flat structures

NP → Det N		NP → er		N → Buch	
S → NP NP NP V		Det → das		N → Kind	
		Det → dem		V → gibt	

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt
NP		NP	Det	Kind	gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt
NP		NP	Det	Kind	gibt
NP		NP	Det	N	gibt

Example derivation assuming flat structures

NP → Det N	NP → er	N → Buch
S → NP NP NP V	Det → das	N → Kind
	Det → dem	V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt
NP		NP	Det	Kind	gibt
NP		NP	Det	N	gibt
NP		NP		NP	gibt

Example derivation assuming flat structures

NP → Det N			NP → er		N → Buch
S → NP NP NP V			Det → das		N → Kind
			Det → dem		V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt
NP		NP	Det	Kind	gibt
NP		NP	Det	N	gibt
NP		NP		NP	gibt
NP		NP		NP	V

Example derivation assuming flat structures

NP → Det N
S → NP NP NP V

NP → er
Det → das
Det → dem

N → Buch
N → Kind
V → gibt

er	das	Buch	dem	Kind	gibt
NP	das	Buch	dem	Kind	gibt
NP	Det	Buch	dem	Kind	gibt
NP	Det	N	dem	Kind	gibt
NP		NP	dem	Kind	gibt
NP		NP	Det	Kind	gibt
NP		NP	Det	N	gibt
NP		NP		NP	gibt
NP		NP		NP	V
					S

Do try this at home!

You can actually play with such grammars.

- Go to https://swish.swi-prolog.org/.
- Click "Program".
- Enter:

```
s --> np, v, np, np.
np --> det, n.
np --> [er].
det --> [das].
det --> [dem].
n --> [buch].
n --> [kind].
v --> [gibt].
```

- Type in the following into the right lower box:
  `s([er,gibt,das,buch,dem,kind],[]).`
- If there appears a "true" in the box above this box, celebrate.

A generative grammar

- The grammar you just entered can generate sentences.

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`
- `s([X],[])` asks Prolog to come up with an X that is an "s".

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`
- `s([X],[])` asks Prolog to come up with an X that is an "s".
- `print(X),nl` prints the X and a newline and

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`
- `s([X],[])` asks Prolog to come up with an X that is an “s”.
- `print(X),nl` prints the X and a newline and
- `fail` tells Prolog that we are not happy and that it should try again.

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`
- `s([X],[])` asks Prolog to come up with an X that is an “s”.
- `print(X),nl` prints the X and a newline and
- `fail` tells Prolog that we are not happy and that it should try again.
- It keeps trying till there are no further solutions and then fails.

A generative grammar

- The grammar you just entered can generate sentences.
- You may test which sentences it generates by typing in:
  `s([X],[]),print(X),nl,fail.`
- `s([X],[])` asks Prolog to come up with an X that is an “s”.
- `print(X),nl` prints the X and a newline and
- `fail` tells Prolog that we are not happy and that it should try again.
- It keeps trying till there are no further solutions and then fails.
- Some grammars generate infinitely many Xes. So this process would never end (unless the computer runs out of memory …).

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give
(Subject verb agreement *ich*, *gibt*)

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give
(Subject verb agreement *ich*, *gibt*)

c. * er das Buch das Kind gibt
he the book the child gives

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give
(Subject verb agreement *ich*, *gibt*)

c. * er das Buch das Kind gibt
he the book the child gives
(case requirement of the verb, *gibt* requires dative)

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give
(Subject verb agreement *ich*, *gibt*)

c. * er das Buch das Kind gibt
he the book the child gives
(case requirement of the verb, *gibt* requires dative)

d. * er den Buch dem Kind gibt
he the book the child gives

Sentences described by the grammar

- The grammar is not precise enough (it *overgenerates*):
  NP → Det N
  S → NP NP NP V

(29) a. er das Buch dem Kind gibt
he the book the child gives

b. * ich das Buch dem Kind gibt
I the book the child give
(Subject verb agreement *ich*, *gibt*)

c. * er das Buch das Kind gibt
he the book the child gives
(case requirement of the verb, *gibt* requires dative)

d. * er den Buch dem Kind gibt
he the book the child gives
(determinator noun agreement *den*, *Buch*)

Subject verb agreement (I)

- Agreement in person (1, 2, 3) and number (sg, pl)

(30) a. Ich schlafe. (1, sg)
b. Du schläfst. (2, sg)
c. Er schläft. (3, sg)
d. Wir schlafen. (1, pl)
e. Ihr schlaft. (2, pl)
f. Sie schlafen. (3,pl)

- How can we express this in rules?

Subject verb agreement (II)

- We make the symbols more informative.
  Instead of S → NP NP NP V we use:

  S → NP_1_sg NP NP V_1_sg
  S → NP_2_sg NP NP V_2_sg
  S → NP_3_sg NP NP V_3_sg
  S → NP_1_pl NP NP V_1_pl
  S → NP_2_pl NP NP V_2_pl
  S → NP_3_pl NP NP V_3_pl
- six symbols for nominal phrases, six for verbs
- six rules instead of one

Case assignment by the verb

- Case must be part of the symbols used in the rules:
  S → NP_1_sg_nom NP_dat NP_acc V_1_sg_ditransitiv
  S → NP_2_sg_nom NP_dat NP_acc V_2_sg_ditransitiv
  S → NP_3_sg_nom NP_dat NP_acc V_3_sg_ditransitiv
  S → NP_1_pl_nom NP_dat NP_acc V_1_pl_ditransitiv
  S → NP_2_pl_nom NP_dat NP_acc V_2_pl_ditransitiv
  S → NP_3_pl_nom NP_dat NP_acc V_3_pl_ditransitiv
- 3 * 2 * 4 = 24 new categories for NPs in total
- 3 * 2 * x categories for V (x = number of attested valence patterns)

Determinator noun agreement

- There is agreement in gender (fem, mas, neu), number (sg, pl) and case (nom, gen, dat, acc)

(31) a. der Mann 'the man', die Frau 'the woman', das Kind 'the child' (gender)
b. das Buch 'the book', die Bücher 'the books' (number)
c. des Buches 'the.GEN book.GEN', dem Buch 'the.DAT book' (case)

Determinator noun agreement

- There is agreement in gender (fem, mas, neu), number (sg, pl) and case (nom, gen, dat, acc)

(31) a. der Mann 'the man', die Frau 'the woman', das Kind 'the child' (gender)
b. das Buch 'the book', die Bücher 'the books' (number)
c. des Buches 'the.GEN book.GEN', dem Buch 'the.DAT book' (case)

- instead of NP → Det N we have

NP_3_sg_nom → Det_fem_sg_nom N_fem_sg_nom	NP_gen → Det_fem_sg_gen N_fem_sg_gen
NP_3_sg_nom → Det_mas_sg_nom N_mas_sg_nom	NP_gen → Det_mas_sg_gen N_mas_sg_gen
NP_3_sg_nom → Det_neu_sg_nom N_neu_sg_nom	NP_gen → Det_neu_sg_gen N_neu_sg_gen
NP_3_pl_nom → Det_fem_pl_nom N_fem_pl_nom	NP_gen → Det_fem_pl_gen N_fem_pl_gen
NP_3_pl_nom → Det_mas_pl_nom N_mas_pl_nom	NP_gen → Det_mas_pl_gen N_mas_pl_gen
NP_3_pl_nom → Det_neu_pl_nom N_neu_pl_nom	NP_gen → Det_neu_pl_gen N_neu_pl_gen
… dative	… accusative

- 24 symbols for determiners, 24 symbols for nouns
- 24 rules instead of one

Problems of simple phrase structure grammars

- Gernalisations are not captured.
- neither in rules nor in category symbols
  - Where can an NP or an NP_nom be placed?
    The only question we can ask is: Where can I put an NP_3_sg_nom?
  - Commonalities between rules are not obvous.

Problems of simple phrase structure grammars

- Gernalisations are not captured.
- neither in rules nor in category symbols
  - Where can an NP or an NP_nom be placed?
    The only question we can ask is: Where can I put an NP_3_sg_nom?
  - Commonalities between rules are not obvous.
- Solution: features with values and identity of values
  Category symbol: NP feature: Per, Num, Cas, …
  We get rules like the following:
  NP(3,sg,nom) → Det(fem,sg,nom) N(fem,sg,nom)
  NP(3,sg,nom) → Det(mas,sg,nom) N(mas,sg,nom)

Features and rule schemata (I)

- Rules with specific values can be generalized to rule schemata:

  NP(3,Num,Cas) → Det(Gen,Num,Cas) N(Gen,Num,Cas)

Features and rule schemata (I)

- Rules with specific values can be generalized to rule schemata:

  NP(3,Num,Cas) → Det(Gen,Num,Cas) N(Gen,Num,Cas)
- Actual Gen, Num and Cas values do not matter
  as long as they are identical.

Features and rule schemata (I)

- Rules with specific values can be generalized to rule schemata:

  NP(3,Num,Cas) → Det(Gen,Num,Cas) N(Gen,Num,Cas)
- Actual Gen, Num and Cas values do not matter
  as long as they are identical.
- The value of the person feature (first slot in NP(3,Num,Cas))
  is fixed by the rule: 3.

Features and rule schemata (II)

- Rules with specific values can be generalized into rule schemata:

  NP(3,Num,Cas) → Det(Gen,Num,Cas) N(Gen,Num,Cas)
  S → NP(Per1,Num1,nom)
  NP(Per2,Num2,dat)
  NP(Per3,Num3,acc)
  V(Per1,Num1)

- Per1 and Num1 value of verb and subject are identical.

Features and rule schemata (II)

- Rules with specific values can be generalized into rule schemata:

  NP(3,Num,Cas) → Det(Gen,Num,Cas) N(Gen,Num,Cas)
  S → NP(Per1,Num1,nom)
  NP(Per2,Num2,dat)
  NP(Per3,Num3,acc)
  V(Per1,Num1)

- Per1 and Num1 value of verb and subject are identical.
- The values of other NPs do not matter.
  (Notation for irrelevant values: '_')

Features and rule schemata (II)

- Rules with specific values can be generalized into rule schemata:

$$
\begin{array}{lll}
\text{NP(3,Num,Cas)} & \rightarrow & \text{Det(Gen,Num,Cas) N(Gen,Num,Cas)} \\
\text{S} & \rightarrow & \text{NP(Per1,Num1,nom)} \\
 & & \text{NP(Per2,Num2,dat)} \\
 & & \text{NP(Per3,Num3,acc)} \\
 & & \text{V(Per1,Num1)}
\end{array}
$$

- Per1 and Num1 value of verb and subject are identical.
- The values of other NPs do not matter.
  (Notation for irrelevant values: '_')
- Case values of the NPs are fixed in the second rule.

Homework

1. Write a phrase structure grammar that can analyze at least the sentences in (32) but excludes the sequences in (33).

(32) a. Der Mann hilft dem Kind.
the man helps the child

b. Er gibt ihr das Buch.
he gives her the book

c. Er wartet auf ein Wunder.
he waits for a miracle

(33) a. * Der Mann hilft er.
the man helps he

b. * Er gibt ihr den Buch.
he gives her the book

The result should be one grammar for all grammatical sentences, not one for each sentence.
You may use Prolog to make sure your grammar actually works: https://swish.swi-prolog.org
See https://en.wikipedia.org/wiki/Definite_clause_grammar for the syntax of Definite Clause Grammars.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \textit{ Grammatik } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ 语法 } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ دستور } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ व्याकरण } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

$\overline{\text{X}}$ Theory

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Reading material

Please read Müller 2020b: Section 2.5.

Nominal phrases

- Until now NP → Det N, but noun phrases can be much more complex:

(34)
a. ein Buch
   a book
b. ein Buch, das wir kennen
   a book that we know
c. ein Buch aus Japan
   a book from Japan
d. ein interessantes Buch
   an interesting book
e. ein Buch aus Japan, das wir kennen
   a book from Japan that we know
f. ein interessantes Buch aus Japan
   an interesting book from Japan
g. ein interessantes Buch, das wir kennen
   an interesting book that we know
h. ein interessantes Buch aus Japan, das wir kennen
   an interesting book from Japan that we know

The additional constituents in (34) are adjuncts.

Adjectives in NPs

- Suggestion:

(35) a. NP → Det N
b. NP → Det A N

Adjectives in NPs

- Suggestion:

(35) a. NP → Det N
b. NP → Det A N

- What about (36)?

(36) alle weiteren schlagkräftigen Argumente
all further strong arguments
'all other strong arguments'

Adjectives in NPs

- Suggestion:

(35) a. NP → Det N
b. NP → Det A N

- What about (36)?

(36) alle weiteren schlagkräftigen Argumente
all further strong arguments
'all other strong arguments'

- We need a rule like (37) for (36):

(37) NP → Det A A N

Adjectives in NPs

- Suggestion:

(35) a. NP → Det N
b. NP → Det A N

- What about (36)?

(36) alle weiteren schlagkräftigen Argumente
all further strong arguments
'all other strong arguments'

- We need a rule like (37) for (36):

(37) NP → Det A A N

- But we do not want to state a limit on how many adjectives there may be:

(38) NP → Det A* N

Adjectives in NPs

- Problem: adj & noun do not form constituent in structures licensed by (39).

  (39) NP → Det A* N

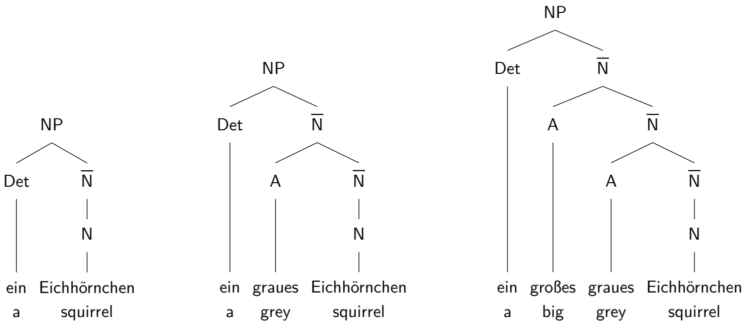
  But constituency tests suggest that A + N is a constituent:

  (40) alle [[großen Seeelefanten] und [grauen Eichhörnchen]]
  all big elephant.seals and grey squirrels
  'all the big elephant seals and grey squirrels'

Adjective + noun as constituent

- The following rule is better suited:

(41) a. NP → Det $\overline{\text{N}}$
b. $\overline{\text{N}}$ → A $\overline{\text{N}}$
c. $\overline{\text{N}}$ → N



Other adjuncts

- Other adjuncts work analogously:

(42) a. $\overline{\text{N}} \rightarrow \overline{\text{N}}$ PP

b. $\overline{\text{N}} \rightarrow \overline{\text{N}}$ relative_clause

Other adjuncts

- Other adjuncts work analogously:

(42) a. $\overline{\text{N}} \rightarrow \overline{\text{N}}$ PP

b. $\overline{\text{N}} \rightarrow \overline{\text{N}}$ relative_clause

- All given determiner-adjective-noun combinations given so far can be analyzed with these few rules.

Complements

- Until now, $\overline{\text{N}}$ consists of a single noun only, but some nouns allow arguments in addition to adjuncts.

(43)
a. der Vater von Peter
the father of Peter
'Peter's father'

b. das Bild vom Gleimtunnel
the picture of.the Gleimtunnel
'the picture of the Gleimtunnel'

c. das Kommen der Installateurin
the coming of.the plumber
'the plumber's visit'

Complements

- Until now, $\overline{\text{N}}$ consists of a single noun only,
  but some nouns allow arguments in addition to adjuncts.

(43) a. der Vater von Peter
the father of Peter
'Peter's father'

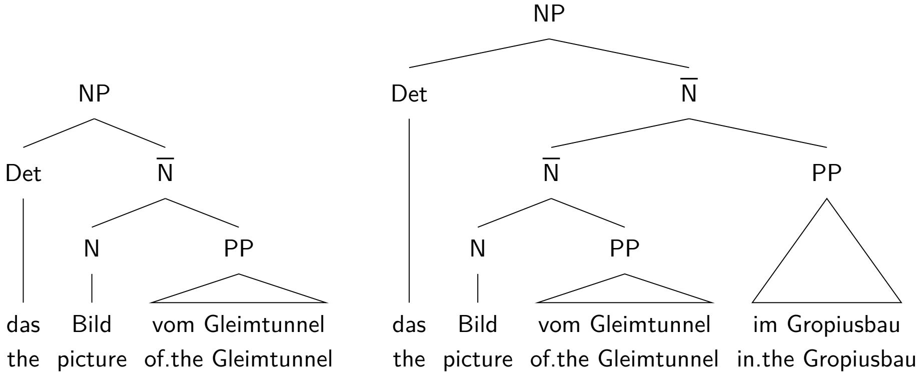
b. das Bild vom Gleimtunnel
the picture of.the Gleimtunnel
'the picture of the Gleimtunnel'

c. das Kommen der Installateurin
the coming of.the plumber
'the plumber's visit'

- Therefore:

(44) $\overline{\text{N}} \rightarrow \text{N PP}$

Complements (and adjuncts)



Missing noun (adjuncts present)

- Noun is missing but adjuncts are present:

(45) a. ein interessantes _
an interesting
'an interesting one'

b. ein neues interessantes _
a new interesting
'a new interesting one'

c. ein interessantes _ aus Japan
an interesting from Japan
'an interesting one from Japan'

d. ein interessantes _, das wir kennen
an interesting that we know
'an interesting one that we know'

Missing noun (complement present)

- noun missing, but a complement of the noun is present:

(46) a. (Nein, nicht der Vater von Klaus), der _ von Peter war gemeint.
no not the father of Klaus the of Peter was meant
'No, it wasn't the father of Klaus, but rather the one of Peter that was meant.'

b. (Nein, nicht das Bild von der Stadtautobahn), das _ vom Gleimtunnel war beeindruckend.
no not the picture of the motorway the of.the Gleimtunnel was impressive
'No, it wasn't the picture of the motorway, but rather the one of the Gleimtunnel that was impressive.'

c. (Nein, nicht das Kommen des Tischlers), das _ der Installateurin ist wichtig.
no not the coming of.the carpenter the of.the plumber is important
'No, it isn't the visit of the carpenter, but rather the visit of the plumber that is important.'

Missing noun (complement present)

- noun missing, but a complement of the noun is present:

(46) a. (Nein, nicht der Vater von Klaus), der _ von Peter war gemeint.
no not the father of Klaus the of Peter was meant
'No, it wasn't the father of Klaus, but rather the one of Peter that was meant.'

b. (Nein, nicht das Bild von der Stadtautobahn), das _ vom Gleimtunnel war beeindruckend.
no not the picture of the motorway the of.the Gleimtunnel was impressive
'No, it wasn't the picture of the motorway, but rather the one of the Gleimtunnel that was impressive.'

c. (Nein, nicht das Kommen des Tischlers), das _ der Installateurin ist wichtig.
no not the coming of.the carpenter the of.the plumber is important
'No, it isn't the visit of the carpenter, but rather the visit of the plumber that is important.'

- PSG: Epsilon production

Missing noun (complement present)

- noun missing, but a complement of the noun is present:

(46) a. (Nein, nicht der Vater von Klaus), der _ von Peter war gemeint.
no not the father of Klaus the of Peter was meant
'No, it wasn't the father of Klaus, but rather the one of Peter that was meant.'

b. (Nein, nicht das Bild von der Stadtautobahn), das _ vom Gleimtunnel war beeindruckend.
no not the picture of the motorway the of.the Gleimtunnel was impressive
'No, it wasn't the picture of the motorway, but rather the one of the Gleimtunnel that was impressive.'

c. (Nein, nicht das Kommen des Tischlers), das _ der Installateurin ist wichtig.
no not the coming of.the carpenter the of.the plumber is important
'No, it isn't the visit of the carpenter, but rather the visit of the plumber that is important.'

- PSG: Epsilon production
- Notation:

(47) a. N →

b. N → ϵ

Missing noun (complement present)

- noun missing, but a complement of the noun is present:

(46) a. (Nein, nicht der Vater von Klaus), der _ von Peter war gemeint.
no not the father of Klaus the of Peter was meant
'No, it wasn't the father of Klaus, but rather the one of Peter that was meant.'

b. (Nein, nicht das Bild von der Stadtautobahn), das _ vom Gleimtunnel war beeindruckend.
no not the picture of the motorway the of.the Gleimtunnel was impressive
'No, it wasn't the picture of the motorway, but rather the one of the Gleimtunnel that was impressive.'

c. (Nein, nicht das Kommen des Tischlers), das _ der Installateurin ist wichtig.
no not the coming of.the carpenter the of.the plumber is important
'No, it isn't the visit of the carpenter, but rather the visit of the plumber that is important.'

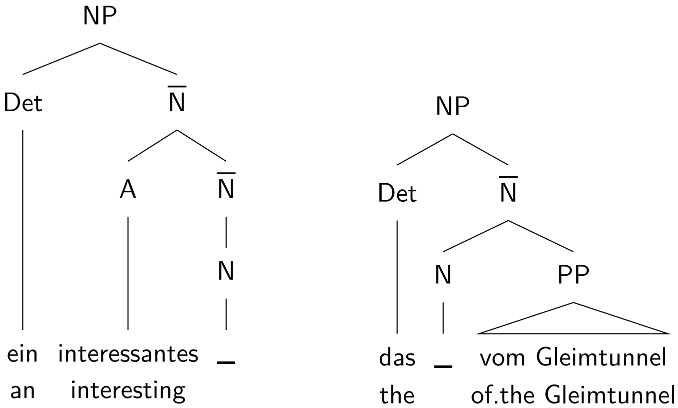
- PSG: Epsilon production
- Notation:

(47) a. N $\rightarrow$

b. N $\rightarrow \epsilon$

- Rules in (47) = empty boxes with the same label as boxes containing normal nouns.

Analysis with empty noun



Missing determiners: Plural

- Determiners can be dropped as well.
  Plural:

(48) a. Bücher
books

b. Bücher, die wir kennen
books that we know

c. interessante Bücher
interesting books

d. interessante Bücher, die wir kennen
interesting books that we know

Missing determiners: Mass nouns

- For mass nouns dropping is possible in the singular as well:

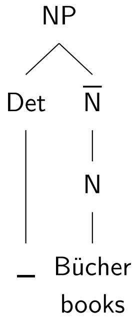
(49) a. Getreide
grain

b. Getreide, das gerade gemahlen wurde
grain that just ground was
'grain that has just been ground'

c. frisches Getreide
fresh grain

d. frisches Getreide, das gerade gemahlen wurde
fresh grain that just ground was
'fresh grain that has just been ground'

Missing determiners: The Structure



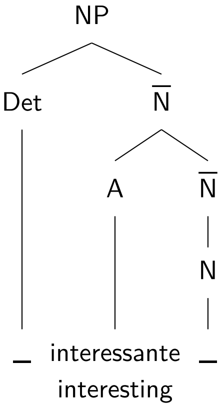
Missing determiners and missing nouns

Determiners and nouns can even be omitted simultaneously:

(50) a. Ich lese interessante.
I read interesting
'I read interesting ones.'

b. Dort drüben steht frisches, das gerade gemahlen wurde.
there over stands fresh that just ground was
'Over there is some fresh (grain) that has just been ground.'

Missing determiners and missing nouns: The structure



Adjective phrases

- Until now simple adjectives like *klug* ‘smart’ only.

Adjective phrases

- Until now simple adjectives like *klug* 'smart' only.
- But adjective phrases can be very complex:

(51) a.

der	seiner	Frau	treue	Mann
the	his.DAT	wife	faithful	man

'the man faithful to his wife'

b.

der	auf	seine	Tochter	stolze	Mann
the	on	his.ACC	daughter	proud	man

'the man proud of his daughter'

c.

der	seine	Frau	liebende	Mann
the	his.ACC	woman	loving	man

'the man who loves his wife'

d.

der	von	seiner	Frau	geliebte	Mann
the	by	his.DAT	wife	loved	man

'the man loved by his wife'

Adjective phrases

- (52) der auf seine Tochter stolze Mann
  the on his.ACC daughter proud man
  'the man proud of his daughter'
- We have to adapt the rule for attributive adjectival modifiers:

  (53) $\overline{\text{N}} \rightarrow \text{AP}\ \overline{\text{N}}$

Adjective phrases

- (52) der auf seine Tochter stolze Mann
  the on his.ACC daughter proud man
  'the man proud of his daughter'
- We have to adapt the rule for attributive adjectival modifiers:

  (53) $\overline{\text{N}} \rightarrow \text{AP}\ \overline{\text{N}}$
- Rules for AP:

  (54) a. AP → NP A
  b. AP → PP A
  c. AP → A

Prepositional phrases

- The syntax of PPs is relatively straight-forward. First attempt:

(55) PP → P NP

Prepositional phrases

- The syntax of PPs is relatively straight-forward. First attempt:

(55) PP → P NP

- But PPs can be augmented by measurement phrases (Eisenberg et al. 2005: §1300):

(56) a. [[Einen Schritt] vor dem Abgrund] blieb er stehen.
one step before the abyss remained he stand
'He stopped one step in front of the abyss.'

b. [[Kurz] nach dem Start] fiel die Klimaanlage aus.
shortly after the take.off fell the air.conditioning out
'Shortly after take off, the air conditioning stopped working.'

c. [[Schräg] hinter der Scheune] ist ein Weiher.
diagonally behind the barn is a pond
'There is a pond diagonally across from the barn.'

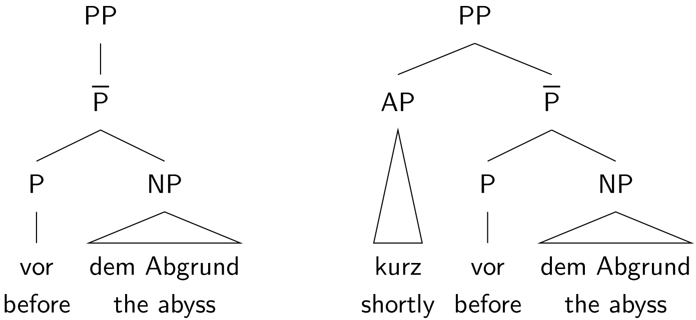
d. [[Mitten] im Urwald] stießen die Forscher auf einen alten Tempel.
middle in.the jungle stumbled the researchers on an old temple
'In the middle of the jungle, the researches came across an old temple.'

Prepositional phrases: The rules

- (57) [[Einen Schritt] vor dem Abgrund]
  one step before the abyss
  'one step in front of the abyss'

  (58) a. PP → NP $\overline{\text{P}}$
  b. PP → AP $\overline{\text{P}}$
  c. PP → $\overline{\text{P}}$
  d. $\overline{\text{P}}$ → P NP

Prepositional phrases: The structure



Generalization over rules

- head + complement = intermediate level:

(59) a. $\overline{\text{N}} \rightarrow$ N PP
b. $\overline{\text{P}} \rightarrow$ P NP

Generalization over rules

- head + complement = intermediate level:

(59) a. $\overline{\text{N}} \rightarrow$ N PP
b. $\overline{\text{P}} \rightarrow$ P NP

- intermediate level + further constituent = maximal projection

(60) a. NP $\rightarrow$ Det $\overline{\text{N}}$
b. PP $\rightarrow$ NP $\overline{\text{P}}$

Generalization over rules

- head + complement = intermediate level:

(59) a. $\overline{\text{N}} \rightarrow$ N PP
b. $\overline{\text{P}} \rightarrow$ P NP

- intermediate level + further constituent = maximal projection

(60) a. NP $\rightarrow$ Det $\overline{\text{N}}$
b. PP $\rightarrow$ NP $\overline{\text{P}}$

- parallel structures for English AP and VP as well

English adjective phrases

(61) Kim and Sandy are
  a. proud.
  b. very proud.
  c. proud of their child.
  d. very proud of their child.

English adjective phrases

(61) Kim and Sandy are
  a. proud.
  b. very proud.
  c. proud of their child.
  d. very proud of their child.

(62)
  a. AP → $\overline{\text{A}}$
  b. AP → Adv $\overline{\text{A}}$
  c. $\overline{\text{A}}$ → A PP
  d. $\overline{\text{A}}$ → A

English adjective phrases

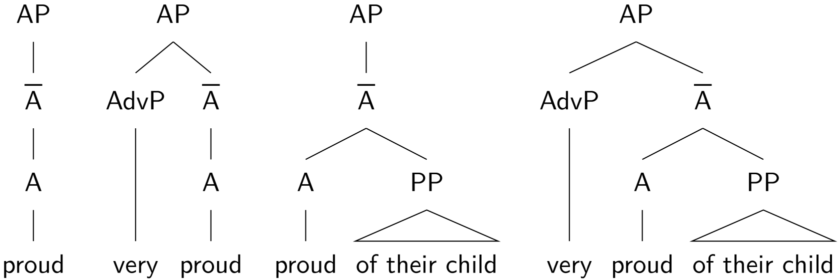
(61) Kim and Sandy are
  a. proud.
  b. very proud.
  c. proud of their child.
  d. very proud of their child.

(62) a. AP → $\overline{\text{A}}$
  b. AP → Adv $\overline{\text{A}}$
  c. $\overline{\text{A}}$ → A PP
  d. $\overline{\text{A}}$ → A

(Müller (2020b: Section 13.1.2): Does not work for German.)

English adjective phrases: The structure

(63)
a. AP $\rightarrow$ $\overline{\text{A}}$
b. AP $\rightarrow$ AdvP $\overline{\text{A}}$
c. $\overline{\text{A}}$ $\rightarrow$ A PP
d. $\overline{\text{A}}$ $\rightarrow$ A



Further abstraction

- We saw that abstraction over case and gender values is possible (variables in rule schemata).

(64) NP(3,Num,Cas) → D(Gen,Num,Cas), N(Gen,Num,Cas)

Further abstraction

- We saw that abstraction over case and gender values is possible (variables in rule schemata).

  (64) NP(3,Num,Cas) → D(Gen,Num,Cas), N(Gen,Num,Cas)

- Similarly we can abstract over the part of speech.
  Instead of AP, NP, PP, VP, we write XP.

Further abstraction

- We saw that abstraction over case and gender values is possible (variables in rule schemata).

(64) NP(3,Num,Cas) → D(Gen,Num,Cas), N(Gen,Num,Cas)

- Similarly we can abstract over the part of speech.
  Instead of AP, NP, PP, VP, we write XP.
- Instead of (65), we write (66):

(65) a. PP → $\overline{\text{P}}$
     b. AP → $\overline{\text{A}}$

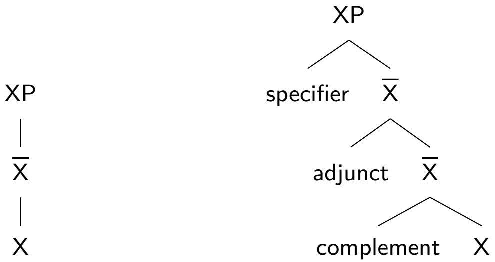
(66) XP → $\overline{\text{X}}$

$\overline{\text{X}}$ Theory: Assumptions

Phrases have at least three levels:

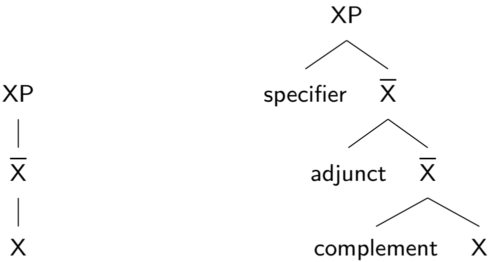
- X^0 = head
- X' = intermediate level (= $\overline{\text{X}}$, pronounced X bar; → name of the scehma)
- XP = highest node (= X'' = $\overline{\overline{\text{X}}}$), also called *maximal projection*

Minimal and maximal expansion of phrases



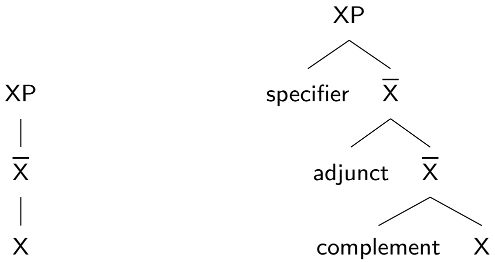
- Adjuncts are optional
  → X′ with adjunct daughter may be missing.

Minimal and maximal expansion of phrases



- Adjuncts are optional
  $\rightarrow$ X′ with adjunct daughter may be missing.
- Some categories do not have a specifier or it is optional (e.g. A).

Minimal and maximal expansion of phrases



- Adjuncts are optional
  $\rightarrow$ X′ with adjunct daughter may be missing.
- Some categories do not have a specifier or it is optional (e.g. A).
- Sometimes in addition adjunction to XP and head adjunction to X.

$\overline{\text{X}}$ Theory: Rules following Jackendoff 1977

$\overline{\text{X}}$ rule	with specific categories	example strings
$\overline{\overline{\text{X}}} \rightarrow \overline{\overline{\text{specifier}}}\ \overline{\text{X}}$	$\overline{\overline{\text{N}}} \rightarrow \overline{\overline{\text{DET}}}\ \overline{\text{N}}$	the [picture of Paris]
$\overline{\text{X}} \rightarrow \overline{\text{X}}\ \overline{\overline{\text{adjunct}}}$	$\overline{\text{N}} \rightarrow \overline{\text{N}}\ \overline{\overline{\text{REL\_CLAUSE}}}$	[picture of Paris] [that everybody knows]
$\overline{\text{X}} \rightarrow \overline{\overline{\text{adjunct}}}\ \overline{\text{X}}$	$\overline{\text{N}} \rightarrow \overline{\overline{\text{A}}}\ \overline{\text{N}}$	beautiful [picture of Paris]
$\overline{\text{X}} \rightarrow \text{X}\ \overline{\overline{\text{complement}}}{*}$	$\overline{\text{N}} \rightarrow \text{N}\ \overline{\overline{\text{P}}}$	picture [of Paris]

X stands for some arbitrary category, X is the head,
'*' stands for arbitrarily many repretitions

X may appear in any position in the right-hand side of the rule.

$$\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \textit{Grammatik}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{语法}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{دستور}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}
\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{व्याकरण}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Government & Binding

Stefan Müller
Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\mathrm{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoning Grammar (TAG)

Reading material

Müller 2020b: Section 3.1

Phrase structure grammars and natural language

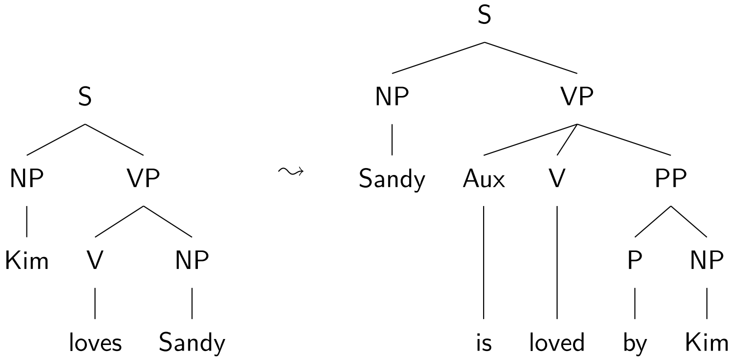
Chomsky: generlizations cannot be captured with PSGs (e.g. active/passive alternations) → transformations:

$$\begin{array}{lll} \text{NP} & \text{V} & \text{NP} \rightarrow 3\ [_{\text{AUX}}\ \text{be}]\ 2\text{en}\ [_{\text{PP}}\ [_{\text{P}}\ \text{by}]\ 1] \\ 1 & 2 & 3 \end{array}$$

(67) a. Kim loves Sandy.
b. Sandy is loved by Kim.

A tree with the sequence of symbols on the left-hand site is mapped to a tree with the sequence of symbols on the right-hand side.

Transformation of an active tree into a passive tree



$$\begin{array}{llll} \text{NP} & \text{V} & \text{NP} & \rightarrow 3\ [_{\text{AUX}}\ \text{be}]\ 2\text{en}\ [_{\text{PP}}\ [_{\text{P}}\ \text{by}]\ 1] \\ 1 & 2 & 3 & \end{array}$$

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars.
  (Chomsky Hierarchy, Type 3–0)

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars. (Chomsky Hierarchy, Type 3–0)
- What we saw so far are so called context free grammars. They are of type 2.

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars. (Chomsky Hierarchy, Type 3–0)
- What we saw so far are so called context free grammars. They are of type 2.
- Maximal level (type 0) is too powerful for human langauges.
  → Researchers wanted to be more restrictive.

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars. (Chomsky Hierarchy, Type 3–0)
- What we saw so far are so called context free grammars. They are of type 2.
- Maximal level (type 0) is too powerful for human langauges.
  → Researchers wanted to be more restrictive.
- Grammars with general transformations correspond to PSGs with type 0 complexity (Peters & Ritchie 1973).

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars. (Chomsky Hierarchy, Type 3–0)
- What we saw so far are so called context free grammars. They are of type 2.
- Maximal level (type 0) is too powerful for human langauges.
  → Researchers wanted to be more restrictive.
- Grammars with general transformations correspond to PSGs with type 0 complexity (Peters & Ritchie 1973).
- Transformations are not sufficiently restricted,
  interactions are not tractable,
  there have been problems with transformations deleting material (see Klenk 2003).

Complexity, transformations and natural languages

- There are different complexity levels for phrase structure grammars.
  (Chomsky Hierarchy, Type 3–0)
- What we saw so far are so called context free grammars. They are of type 2.
- Maximal level (type 0) is too powerful for human langauges.
  → Researchers wanted to be more restrictive.
- Grammars with general transformations correspond to PSGs with type 0 complexity (Peters & Ritchie 1973).
- Transformations are not sufficiently restricted,
  interactions are not tractable,
  there have been problems with transformations deleting material (see Klenk 2003).
- → new theoretical approaches, Government & Binding (Chomsky 1981): restrictions for the form of grammar rules, elements can be connected to the position in a tree they were coming from, general principles to restrict the power of transformations

Hypothesis regarding language acquisition: Principles & Paramaters

- Some of our linguistics knowledge is innate.
  (Not all linguists agree with this assumption! Discussion: Müller 2020b)

Hypothesis regarding language acquisition: Principles & Paramaters

- Some of our linguistics knowledge is innate.
  (Not all linguists agree with this assumption! Discussion: Müller 2020b)
- Principles all linguistic structures have to obey

Hypothesis regarding language acquisition: Principles & Paramaters

- Some of our linguistics knowledge is innate.
  (Not all linguists agree with this assumption! Discussion: Müller 2020b)
- Principles all linguistic structures have to obey
- These principles are parametrized → there is choice
  A parameter may be set differently for different languages.

Hypothesis regarding language acquisition: Principles & Paramaters

- Some of our linguistics knowledge is innate.
  (Not all linguists agree with this assumption! Discussion: Müller 2020b)
- Principles all linguistic structures have to obey
- These principles are parametrized → there is choice
  A parameter may be set differently for different languages.

  Example:
  Principle: A head is placed before or after its complements depending on the value of the parameter POSITION.

(68) a. be showing pictures of himself (English)

b. zibun -no syasin-o mise-te iru (Japanese)
   SELF of picture showing be

Deep and Surface Structure

- Chomsky claimed that simple PSGs cannot capture certain regularities.
  e.g. the relation between active and passive sentences.

Deep and Surface Structure

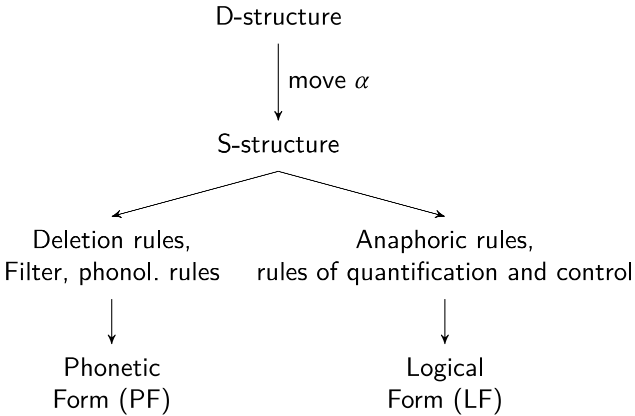
- Chomsky claimed that simple PSGs cannot capture certain regularities.
  e.g. the relation between active and passive sentences.
- Therefore he assumes an underlying structure,
  the so-called Deep Structure.

Deep and Surface Structure

- Chomsky claimed that simple PSGs cannot capture certain regularities.
  e.g. the relation between active and passive sentences.
- Therefore he assumes an underlying structure,
  the so-called Deep Structure.
- A structure can be mapped onto another structure.
  Parts may be deleted or moved to other positions in trees in such mappings.
  As a result of such transformations a new structure is derived, the so-called Surface Structure.

  Surface Structure = S Structure
  Deep Structure = D Structure

The T-model


D-structure
move α
S-structure
Deletion rules,
Filter, phonol. rules
Anaphoric rules,
rules of quantification and control
Phonetic
Form (PF)
Logical
Form (LF)

The T-model: The lexicon

- Contains a lexical entry for every word with information about:
  - morphophonological structure
  - syntactic features
  - valence frame
  - ...

  Contains list for word forms and morphemes and morphology component

The T-model: The lexicon

- Contains a lexical entry for every word with information about:
  - morphophonological structure
  - syntactic features
  - valence frame
  - …

  Contains list for word forms and morphemes and morphology component
- The lexicon is the interface between syntax and semantic interpretation of word forms.

The T-model: The lexicon

- Contains a lexical entry for every word with information about:
  - morphophonological structure
  - syntactic features
  - valence frame
  - ...

  Contains list for word forms and morphemes and morphology component
- The lexicon is the interface between syntax and semantic interpretation of word forms.
- Vocabulary is not determined by UG (not innate), just structural conditions are determined by UG. (assumption not shared by all linguists)

The T-model: The lexicon

- Contains a lexical entry for every word with information about:
  - morphophonological structure
  - syntactic features
  - valence frame
  - ...

  Contains list for word forms and morphemes and morphology component
- The lexicon is the interface between syntax and semantic interpretation of word forms.
- Vocabulary is not determined by UG (not innate),
  just structural conditions are determined by UG.
  (assumption not shared by all linguists)
- Morphosyntactic features (e.g. gender) are not pre-determined:
  Universal grammar provides a toolbox (claim not falsifiable).

The T modell: D Structure, Move-α and S Structurr (I)

- Phrase structure →
  We can describe relations between constituents.

The T modell: D Structure, Move-α and S Structurr (I)

- Phrase structure $\rightarrow$
  We can describe relations between constituents.
- A certain format for rules is given ($\overline{\text{X}}$-Schema).
  Lexicon + structures of $\overline{\text{X}}$ syntax = base for D Structure
  D Structure = syntactic representation of valence frames of particular words as determined in the lexicon.

The T-modell: D Structure, Move-α and S Structure (II)

- constituents may be appearing at different places at the surface than the one determined by the valence frame:

(69) a. [dass] der Mann dem Kind das Buch gibt
that the.NOM man the.DAT woman the.ACC book gives
'that the man gives the woman the book'

b. Gibt der Mann dem Kind das Buch?
gives the.NOM man the.DAT woman the.ACC book
'Does the man give the woman the book?'

c. Der Mann gibt dem Kind das Buch.
the.NOM man gives the.DAT woman the.ACC book
'The man gives the woman the book.'

The T-modell: D Structure, Move-α and S Structure (II)

- constituents may be appearing at different places at the surface than the one determined by the valence frame:

(69) a.

[dass]	der	Mann	dem	Kind	das	Buch	gibt
that	the.NOM	man	the.DAT	woman	the.ACC	book	gives

‘that the man gives the woman the book’

b.

Gibt	der	Mann	dem	Kind	das	Buch?
gives	the.NOM	man	the.DAT	woman	the.ACC	book

‘Does the man give the woman the book?’

c.

Der	Mann	gibt	dem	Kind	das	Buch.
the.NOM	man	gives	the.DAT	woman	the.ACC	book

‘The man gives the woman the book.’

- therefore transformational rules for reordering:
  Move α = „Move anything anywhere!“
  What exactly can be moved where and for which reason is determined by principles.

The T-modell: D Structure, Move-α and S Structure (III)

- Relations between predicates and their arguments as determined by lexical entries must be recoverable on all representational levels for semantic interpretation.

The T-modell: D Structure, Move-α and S Structure (III)

- Relations between predicates and their arguments as determined by lexical entries must be recoverable on all representational levels for semantic interpretation.
- → Starting place of moved elements is marked with traces.

(70) a. [dass] der Mann dem Kind das Buch gibt
that the man the woman the book gives
'that the man gives the woman the book'

b. Gibt$_i$ der Mann dem Kind das Buch _$_i$?
gives the man the woman the book
'Does the man give the woman the book?'

c. [Der Mann]$_j$ gibt$_i$ _$_j$ dem Kind das Buch _$_i$.
the man gives the woman the book
'The man gives the woman the book.'

Different traces are marked by indices.
Sometimes also *e* for empty element and *t* for trace.

The T-modell: D Structure, Move-α and S Structure (III)

- Relations between predicates and their arguments as determined by lexical entries must be recoverable on all representational levels for semantic interpretation.
- → Starting place of moved elements is marked with traces.

(70) a. [dass] der Mann dem Kind das Buch gibt
that the man the woman the book gives
'that the man gives the woman the book'

b. Gibt$_i$ der Mann dem Kind das Buch $\_\_ _i$?
gives the man the woman the book
'Does the man give the woman the book?'

c. [Der Mann]$_j$ gibt$_i$ $\_\_ _j$ dem Kind das Buch $\_\_ _i$.
the man gives the woman the book
'The man gives the woman the book.'

Different traces are marked by indices.
Sometimes also *e* for empty element and *t* for trace.

- S Structure is a surface-like structure but should not be equated with the structure of actual utterances.

The T-model: Phonetic Form

PF is the phonetic form of a sentence, the string of phonemes actually pronounced.

The mapping from S Structure to PF incorporates the phonological laws.

The T-model: Phonetic Form

PF is the phonetic form of a sentence, the string of phonemes actually pronounced.

The mapping from S Structure to PF incorporates the phonological laws.

Example: *wanna* contraction

(71) a. The students want to visit Paris.
b. The students wanna visit Paris.

The contratcion in (71) is licenced by the optional rule in (72):

(72) want + to → wanna

The T-model: Logical Form (I)

- Logical Form is a syntactic level mediating between S Structure and semantic interpretation of a sentence.
  anaphoric reference (binding): what can pronouns refer to?

(73) a. Peter kauft einen Tisch. Er gefällt ihm.
Peter buys a table(M) he likes him
'Peter is buying a table. He likes it/him.'

The T-model: Logical Form (I)

- Logical Form is a syntactic level mediating between S Structure and semantic interpretation of a sentence.
  anaphoric reference (binding): what can pronouns refer to?

(73) a. Peter kauft einen Tisch. Er gefällt ihm.
Peter buys a table(M) he likes him
'Peter is buying a table. He likes it/him.'

b. Peter kauft eine Tasche. Er gefällt ihm.
Peter buys a bag(F) he likes him
'Peter is buying a bag. He likes it/him.'

The T-model: Logical Form (I)

- Logical Form is a syntactic level mediating between S Structure and semantic interpretation of a sentence.
  anaphoric reference (binding): what can pronouns refer to?

(73) a. Peter kauft einen Tisch. Er gefällt ihm.
Peter buys a table(M) he likes him
'Peter is buying a table. He likes it/him.'

b. Peter kauft eine Tasche. Er gefällt ihm.
Peter buys a bag(F) he likes him
'Peter is buying a bag. He likes it/him.'

c. Peter kauft eine Tasche. Er gefällt sich.
Peter buys a bag(F) he likes himself
'Peter is buying a bag. He likes himself.'

The T-model: Logical Form (II)

- Quantification:

  (74) Every dolphin attacks a shark.

  $\forall x \exists y (dolphin(x) \rightarrow (shark(y) \wedge attack(x, y))$
  $\exists y \forall x (dolphin(x) \rightarrow (shark(y) \wedge attack(x, y))$

- Some accounts try to derive the readings via movement of quantifiers in trees (May 1985).

The T-model: Logical Form (III)

Control theory:
How is the semantic role of the subject of the infinitive filled?

(75) a. Die Professorin schlägt der Studentin vor, die Klausur noch mal zu schreiben.
the professor suggests the student PART the test once again to write
'The professor advises the student to take the test again.'

The T-model: Logical Form (III)

Control theory:
How is the semantic role of the subject of the infinitive filled?

(75) a. Die Professorin schlägt der Studentin vor, die Klausur noch mal zu schreiben.
the professor suggests the student PART the test once again to write
'The professor advises the student to take the test again.'

b. Die Professorin schlägt der Studentin vor, die Klausur nicht zu bewerten.
the professor suggests the student PART the test not to grade
'The professor suggests to the student not to grade the test.'

The T-model: Logical Form (III)

Control theory:
How is the semantic role of the subject of the infinitive filled?

(75) a. Die Professorin schlägt der Studentin vor, die Klausur noch mal zu schreiben.
the professor suggests the student PART the test once again to write
'The professor advises the student to take the test again.'

b. Die Professorin schlägt der Studentin vor, die Klausur nicht zu bewerten.
the professor suggests the student PART the test not to grade
'The professor suggests to the student not to grade the test.'

c. Die Professorin schlägt der Studentin vor, gemeinsam ins Kino zu gehen.
the professor suggests the student PART together into cinema to go
'The professor suggests to the student to go to the cinema together.'

Lexicon: Basic terminology (I)

- meaning of words → combinatoric potential with certain semantic roles (“acting person” or “affected thing”)

  Example: meaning representation of (76a) is (76b):

(76) a. Judit beats the grandmaster.

b. *beat′*(x,y)

Lexicon: Basic terminology (I)

- meaning of words → combinatoric potential with certain semantic roles ("acting person" or "affected thing")

  Example: meaning representation of (76a) is (76b):

(76) a. Judit beats the grandmaster.

  b. *beat′*(x,y)

- This is subsumed under the terms valency and selection.

  Note:

  Semantic valence may differ from syntactic valence! (see Müller 2020b: Section 1.6)

Lexicon: Basic terminology (I)

- meaning of words → combinatoric potential with certain semantic roles ("acting person" or "affected thing")

  Example: meaning representation of (76a) is (76b):

(76) a. Judit beats the grandmaster.

b. *beat′*(x,y)

- This is subsumed under the terms valency and selection.

  Note:
  Semantic valence may differ from syntactic valence! (see Müller 2020b: Section 1.6)

- Another term is subcategorization:

  beat is subcategorized for a subject and an object.

  The word *subcategorize* somehow developed its own life:
  X subcategorizes for Y is used for *X selects Y*.

Lexicon: Basic terminology (II)

- *beat* is also called the predicate
  (since *beat′* is the logical predicate).

Lexicon: Basic terminology (II)

- *beat* is also called the predicate
  (since *beat′* is the logical predicate).
- Subject and object are arguments of the predicate.

Lexicon: Basic terminology (II)

- *beat* is also called the predicate
  (since *beat′* is the logical predicate).
- Subject and object are arguments of the predicate.
- Several terms for selectional requirement (some semantic, some syntactic, some mixed): argument structure, valence frame, subcategorization frame, thematic grid and theta-grid or θ-grid

Lexicon: Basic terminology (II)

- *beat* is also called the predicate
  (since *beat′* is the logical predicate).
- Subject and object are arguments of the predicate.
- Several terms for selectional requirement (some semantic, some syntactic, some mixed): argument structure, valence frame, subcategorization frame, thematic grid and theta-grid or θ-grid
- Adjuncts modify semantic predicates.
  If semantic aspects are discussed, the term is modifier.
  Adjuncts are not listed as part of valence frames.

The Theta-Criterion

Arguments are placed into certain positions in the clause (argument positions).

Theta-Criterion (Chomsky 1981: 36):

- Each theta-role is assigned to exactly one argument position.
- Every phrase in an argument position receives exactly one theta-role.

External argument and internal arguments

- Arguments are ordered: there are higher- and lower-ranked arguments

External argument and internal arguments

- Arguments are ordered: there are higher- and lower-ranked arguments
- The highest-ranked argument of verbs and adjectives has a special status.
  It is often (and always in some languages) realized in a position outside of the verb or adjective phrase, it is called the external argument.

External argument and internal arguments

- Arguments are ordered: there are higher- and lower-ranked arguments
- The highest-ranked argument of verbs and adjectives has a special status.
  It is often (and always in some languages) realized in a position outside of the verb or adjective phrase, it is called the external argument.
- The remaining arguments occur in positions inside of the VP or AP.
  Term: internal argument or complement

External argument and internal arguments

- Arguments are ordered: there are higher- and lower-ranked arguments
- The highest-ranked argument of verbs and adjectives has a special status.
  It is often (and always in some languages) realized in a position outside of the verb or adjective phrase, it is called the external argument.
- The remaining arguments occur in positions inside of the VP or AP.
  Term: internal argument or complement
- For simple sentences: external argument = subject.

Theta roles

- There are three classes of theta-roles.

Theta roles

- There are three classes of theta-roles.
- Class 1 is usually the highest role, class 3 the lowest.

Theta roles

- There are three classes of theta-roles.
- Class 1 is usually the highest role, class 3 the lowest.
  - Class 1: agent (acting individual), the cause of an action or feeling (stimulus), holder of a certain property

Theta roles

- There are three classes of theta-roles.
- Class 1 is usually the highest role, class 3 the lowest.
  - Class 1: agent (acting individual), the cause of an action or feeling (stimulus), holder of a certain property
  - Class 2: experiencer (perceiving individual), the person profiting from something (beneficiary) (or the opposite: the person affected by some kind of damage), possessor (owner or soon-to-be owner of something, or the opposite: someone who has lost or is lacking something)

Theta roles

- There are three classes of theta-roles.
- Class 1 is usually the highest role, class 3 the lowest.
  - Class 1: agent (acting individual), the cause of an action or feeling (stimulus), holder of a certain property
  - Class 2: experiencer (perceiving individual), the person profiting from something (beneficiary) (or the opposite: the person affected by some kind of damage), possessor (owner or soon-to-be owner of something, or the opposite: someone who has lost or is lacking something)
  - Class 3: patient (affected person or thing), theme

Theta roles

- There are three classes of theta-roles.
- Class 1 is usually the highest role, class 3 the lowest.
  - Class 1: agent (acting individual), the cause of an action or feeling (stimulus), holder of a certain property
  - Class 2: experiencer (perceiving individual), the person profiting from something (beneficiary) (or the opposite: the person affected by some kind of damage), possessor (owner or soon-to-be owner of something, or the opposite: someone who has lost or is lacking something)
  - Class 3: patient (affected person or thing), theme
- Caution!
  Rather inconsistent assignment of roles by different authors. Proto-roles a la Dowty (1991) may be the only feasible way to deal with the problem.

A lexical entry (I)

Which information do we need to use a word appropriately?

Answer: The mental lexicon contains lexical entries with the specific properties of syntactic words needed to use that word grammatically.

Some of these properties are the following:

- form
- meaning (semantics)
- grammatical features: syntactic word class + morphosyntactic features
- theta-grid

A lexical entry (II)

form	*helf-* 'help'	
semantics	*helfen'*	
grammatical features	verb	
theta-grid		
theta-roles	<u>agent</u>	beneficiary
grammatical particularities		dative

Arguments are ordered according to their ranking:
the highest argument is furthest left.

In this case, the highest argument is the external argument.

The external argument is underlined.

Comment on distribution of $\overline{\text{X}}$ rules

$\overline{\text{X}}$ Theory is assumed in many other frameworks as well:

- Lexical Functional Grammar (LFG):
  Bresnan 1982a; 2001; Berman & Frank 1996; Berman 2003
- Generalized Phrase Structure Grammar (GPSG):
  Gazdar, Klein, Pullum & Sag 1985

Sometimes different categories are assuemd.
In particular so-called functional categories (e.g. INFL).

No assumptions about universality and innateness are made in most other theories.

$\overline{\text{X}}$ Theory: Heads

$\overline{\text{X}}$ Theory: Heads

Head determines the most important properties of a phrase.

(77)
a. Kim schläft.
   Kim sleeps
b. Kim mag Sandy.
   Kim likes Sandy
c. in diesem Haus
   in this house
d. ein Haus
   a house

$\overline{\text{X}}$ Theory: Lexical categories

categories are divided into lexical and functional categories
(≈ correlates roughly with the difference between open and closed word classes)

Lexical categories:

- V = verb
- N = noun
- A = adjective
- P = preposition
- Adv = adverb

$\overline{\text{X}}$ Theory: Lexical categories (cross classification)

Attempt to use binary features to cross-classify lexical categories:

	$-$ V	$+$ V
$-$ N	P = [$-$ N, $-$ V]	V = [$-$ N, $+$ V]
$+$ N	N = [$+$ N, $-$ V]	A = [$+$ N, $+$ V]

$\overline{\text{X}}$ Theory: Lexical categories (cross classification)

Attempt to use binary features to cross-classify lexical categories:

	− V	+ V
− N	P = [− N, − V]	V = [− N, + V]
+ N	N = [+ N, − V]	A = [+ N, + V]

Cross classification → simple way to refer to adjectives and verbs:
all lexical categories that are [+ V] are either verbs or adjectives.

Generalizations are possible e.g.: [+ N] categories may bear case

Note: Adverbs can be treated as prepositions not selecting an argument.

Head position dependent on the decomposed category?

Nouns and prepositions are head-initial:

(78) a. für Maria
for Maria

b. Bild von Maria
picture of Maria

Adjectives and verbs are head-final:

(79) a. dem König treu
the king loyal
'Loyal to the king'

b. der [dem Kind helfende] Mann
the the child helping man
'the man helping the child'

c. dem Mann helfen
the man help
'help the man'

Head position dependent on the decomposed category? (II)

$\rightarrow$ [+ V] $\equiv$ head-final
[− V] $\equiv$ head-initial

Head position dependent on the decomposed category? (II)

→ [+ V] ≡ head-final
[− V] ≡ head-initial

Problem: postpositions (P = [− V])

(80) a. des Geldes wegen
the money because
'because of the money'

b. die Nacht über
the night during
'during the night'

Head position dependent on the decomposed category? (II)

→ [+ V] ≡ head-final
[− V] ≡ head-initial

Problem: postpositions (P = [− V])

(80) a. des Geldes wegen
the money because
'because of the money'

b. die Nacht über
the night during
'during the night'

Assume a new feature with binary value?
But then we would get four new categories in total.
But we need only one.

So, maybe this binary encoding is not such a good idea after all.

$\overline{\text{X}}$ Theory: Functional categories

No cross-classification:

C	Complementizer (subordinating conjunctions such as *dass* 'that')
I	Finiteness (as well as Tense and Mood); also Infl in earlier work (inflection), T in more recent work (Tense)
D	Determiner (article, demonstrative)

$\overline{\text{X}}$ Theory: Assumptions

- Endocentricity:
  Every phrase has a head and every head is part of a phrase.
  more technically: every head projects to a phrase.

$\overline{\text{X}}$ Theory: Assumptions

- Endocentricity:
  Every phrase has a head and every head is part of a phrase.
  more technically: every head projects to a phrase.
- Binary branching (predominant assumption today):
  Non-terminal nodes are binary branching,
  that is, there are no teneray branching nodes or nodes with more daughters.

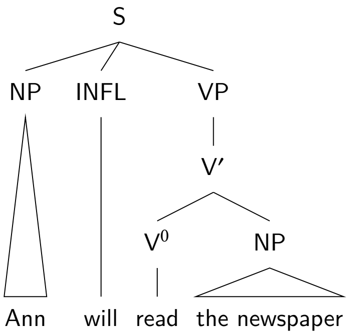
$\overline{\text{X}}$ Theory: Assumptions

- Endocentricity:
  Every phrase has a head and every head is part of a phrase.
  more technically: every head projects to a phrase.
- Binary branching (predominant assumption today):
  Non-terminal nodes are binary branching,
  that is, there are no teneray branching nodes or nodes with more daughters.
- Non-Tangling Condition:
  The branches of tree structures cannot cross.

English clause structure and $\overline{\text{X}}$ Theory

- In early work the following rules were assumed for English:

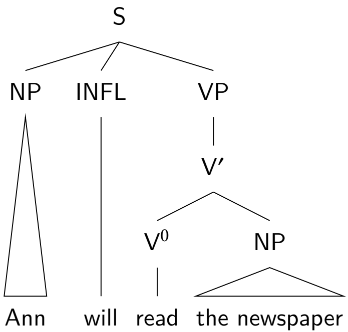
(81) a. S → NP VP
b. S → NP Infl VP



English clause structure and $\overline{\mathrm{X}}$ Theory

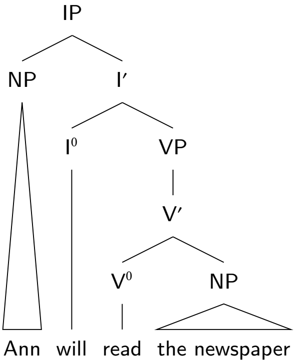
- In early work the following rules were assumed for English:

(81) a. S → NP VP
     b. S → NP Infl VP



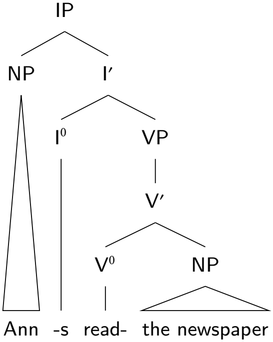
- These rules do not adhere to the $\overline{\mathrm{X}}$ schema.

The English IP and VP: Auxiliaries



- Instead of earlier approaches: INFL as head, INFL selecting a VP as complement.
- Auxiliaries are placed in I^0 (= Aux).
- Sentential adverbs may be placed between auxiliary and main verb.

The English IP and VP: Clauses without auxiliary



- Auxiliaries are placed in I^0 (= Aux).
- Position may contain the inflectional affix. The finite verb moves there.

  (Various variants of the theory …. Some assume lowering of the affix, some assume an empty I position and connection to the finite verb. For German, the best version seems to be to not assume I at all (Haider 1993; 1997).)

c-command, m-command, and government

- Case and (internal) theta roles are assigned under government.

c-command, m-command, and government

- Case and (internal) theta roles are assigned under government.
- Government is a syntactic relation in phrase structure.

c-command, m-command, and government

- Case and (internal) theta roles are assigned under government.
- Government is a syntactic relation in phrase structure.
- Government relies on m-command.
  c-command is similar to m-command and needed for Binding Theory.

c-command and m-command

Popular formulations:

- c-command: upwards and at the next possibility downward again
- m-command: upwards and downwards at any dominating node but not higher than the next XP

c-command and m-command

Popular formulations:

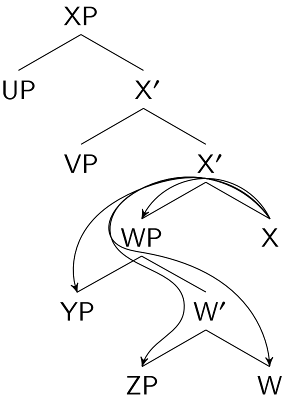
- c-command: upwards and at the next possibility downward again
- m-command: upwards and downwards at any dominating node but not higher than the next XP

Exact version:

c-command A c-commands B iff neither A dominates B nor B dominates A and the first branching node dominating A also dominates B.

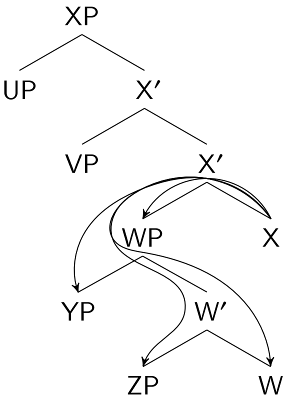
m-command A m-commands B iff neither A dominates B nor B dominates A and the first maximal projection XP dominating A also dominates B.

Examples

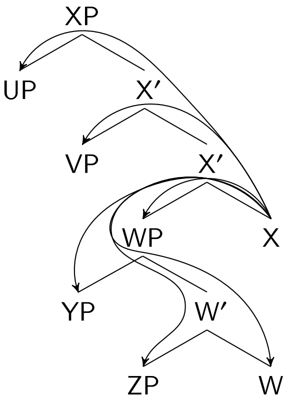


c-command

Examples



c-command



m-command

Government (definition)

Government is a structural relation between a head X^0 and a phrase YP:

Government X^0 governs YP iff a), b) and c) hold simultaneously:

a) X^0 has category V, N, A, P (= lexical cateories) or finite I.
b) X^0 m-commands YP.
c) There is no barrier between X^0 and YP.

Barrier is defined on a language-particular basis.
Simplified: maximal projections except IP.

Government (definition)

Government is a structural relation between a head X^0 and a phrase YP:

Government X^0 governs YP iff a), b) and c) hold simultaneously:

a) X^0 has category V, N, A, P (= lexical cateories) or finite I.
b) X^0 m-commands YP.
c) There is no barrier between X^0 and YP.

Barrier is defined on a language-particular basis.
Simplified: maximal projections except IP.

Clause c) makes sure that heads can assign neither case nor theta role to parts of NP or PP.

c) restricts government in depth.

Government (definition)

Government is a structural relation between a head X^0 and a phrase YP:

Government X^0 governs YP iff a), b) and c) hold simultaneously:

a) X^0 has category V, N, A, P (= lexical cateories) or finite I.
b) X^0 m-commands YP.
c) There is no barrier between X^0 and YP.

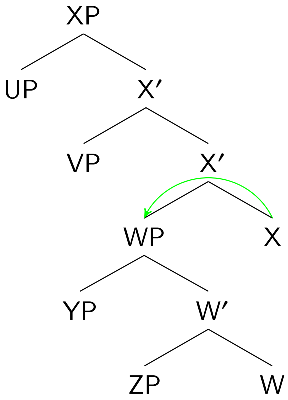
Barrier is defined on a language-particular basis.
Simplified: maximal projections except IP.

Clause c) makes sure that heads can assign neither case nor theta role to parts of NP or PP.

c) restricts government in depth.

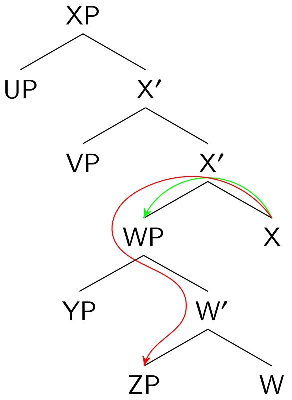
Elements inside of NPs and PPs bearing case must get it inside of the NP or PP not from outside.

Government (example)



- X can assign a theta role to WP.

Government (example)



- X can assign a theta role to WP.
- X cannot assign a theta role to ZP, since WP is a barrier, provided WP $\neq$ IP.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH}\ \langle\ \textit{Grammatik}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT}\ \langle\ \text{DET}\ \rangle \\ \text{SEM}\ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH}\ \langle\ \text{语法}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT}\ \langle\ \text{DET}\ \rangle \\ \text{SEM}\ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} \\ \text{ORTH}\ \langle\ \text{دستور}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT}\ \langle\ \text{DET}\ \rangle \\ \text{SEM}\ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH}\ \langle\ \text{व्याकरण}\ \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT}\ \langle\ \text{DET}\ \rangle \\ \text{SEM}\ \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Government & Binding: Verb position and long distance dependencies

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Reading material

Müller 2020b: Section 3.2–3.3

Excursus: The English CP/IP/VP system

- Often the grammars of languages are modeled after suggestions for English.

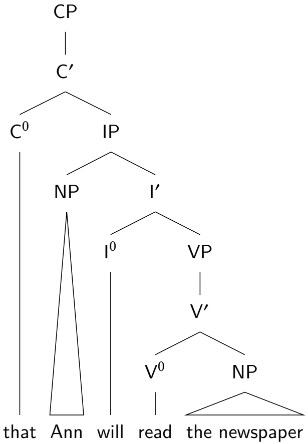
Excursus: The English CP/IP/VP system

- Often the grammars of languages are modeled after suggestions for English.
- Reasoning: Grammars are formed/limited by UG.
  We know that English has property X, hence
  all languages have property X.
  Caution: This is not a valid inference.

Excursus: The English CP/IP/VP system

- Often the grammars of languages are modeled after suggestions for English.
- Reasoning: Grammars are formed/limited by UG.
  We know that English has property X, hence
  all languages have property X.
  Caution: This is not a valid inference.
- In order to understand the particular analysis discussed here,
  we first have to look at English.

English clauses with complementizer



- The complementizer (*that*, *because*, …) requires an IP.

The English CP, IP and VP: Questions

- Ye/no questions are formed by fronting the auxiliary:

  (82) Will Ann read the newspaper?

- The auxiliary moves to the position of the complementizer.

The English CP, IP and VP: Questions

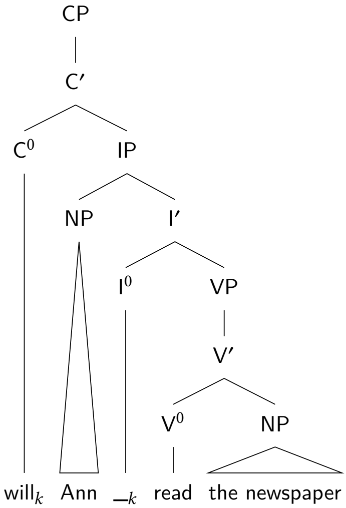
- Ye/no questions are formed by fronting the auxiliary:

(82) Will Ann read the newspaper?

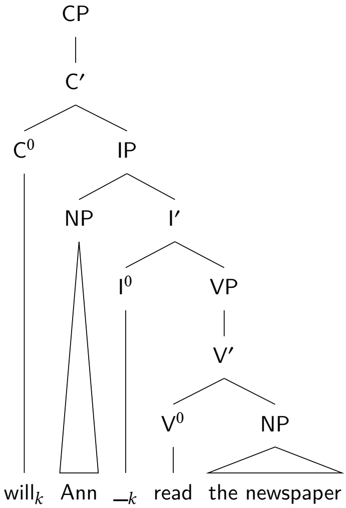
- The auxiliary moves to the position of the complementizer.
- *wh* questions are formed by additionally preposing a constituent:

(83) What will Ann read?

English CP, IP and VP: Questions



English CP, IP and VP: Questions



Topology of the German clause (I)

Before turning to the CP/IP system in grammars of German we have to sort out some terminology:

- Approaches to German constituent order often refer to topological fields.

Topology of the German clause (I)

Before turning to the CP/IP system in grammars of German we have to sort out some terminology:

- Approaches to German constituent order often refer to topological fields.
- Important works on topological fields are:
  Drach 1937, Reis 1980 and Höhle 2018d; 1986.

Topology of the German clause (I)

Before turning to the CP/IP system in grammars of German we have to sort out some terminology:

- Approaches to German constituent order often refer to topological fields.
- Important works on topological fields are:
  Drach 1937, Reis 1980 and Höhle 2018d; 1986.
- We will use Vorfeld, linke/rechte Satzklammer, Mittelfeld and Nachfeld.
  Bech 1955 introduced further fields for verbal complexes,
  but we will ignore them here.

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen **hat**.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) **Hat** Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter **hat** das Eis gegessen.
Peter has the ice.cream eaten

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter hat das Eis gegessen.
Peter has the ice.cream eaten

- verbal elements continuous in (84) only

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter hat das Eis gegessen.
Peter has the ice.cream eaten

- verbal elements continuous in (84) only
- left and right sentence bracket

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter hat das Eis gegessen.
Peter has the ice.cream eaten

- verbal elements continuous in (84) only
- left and right sentence bracket
- complementizer (*weil*, *dass*, *ob*) in left sentence bracket

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter hat das Eis gegessen.
Peter has the ice.cream eaten

- verbal elements continuous in (84) only
- left and right sentence bracket
- complementizer (*weil*, *dass*, *ob*) in left sentence bracket
- complementizer and finite verb have complementary distribution (Höhle 1997)

Verb positions and terminology

- Verb-final position

(84) Peter hat erzählt, dass er das Eis gegessen hat.
Peter has told that he the ice.cream eaten has

- Verb-initial position

(85) Hat Peter das Eis gegessen?
has Peter the ice.cream eaten

- Verb-second poisiton

(86) Peter hat das Eis gegessen.
Peter has the ice.cream eaten

- verbal elements continuous in (84) only
- left and right sentence bracket
- complementizer (*weil*, *dass*, *ob*) in left sentence bracket
- complementizer and finite verb have complementary distribution (Höhle 1997)
- region before, between and after the brackets: Vorfeld, Mittelfeld, Nachfeld

Topology of German clauses

Vorfeld	left bracket	Mittelfeld	right bracket	Nachfeld
Karl	schläft.			
Karl	hat		geschlafen.	
Karl	erkennt	Maria.		
Karl	färbt	den Mantel	um	den Maria kennt.
Karl	hat	Maria	erkannt.	
Karl	hat	Maria als sie aus dem Zug stieg sofort	erkannt.	
Karl	hat	Maria sofort	erkannt	als sie aus dem Zug stieg.
Karl	hat	Maria zu erkennen	behauptet.	
Karl	hat		behauptet	Maria zu erkennen.
	Schläft	Karl?		
	Schlaf!			
	Iß	jetzt dein Eis	auf!	
	Hat	er doch das ganze Eis alleine	gegessen.	
	weil	er das ganze Eis alleine	gegessen hat	ohne sich zu schämen.
	weil	er das ganze Eis alleine	essen können will	ohne gestört zu werden.
wer		das ganze Eis alleine	gegessen hat.	

The Rangprobe

- Fields may be empty.

(87) $\underbrace{\text{Der Delphin}}_{\text{VF}}$ $\underbrace{\text{gibt}}_{\text{LS}}$ $\underbrace{\text{dem Kind den Ball,}}_{\text{MF}}$ $\underbrace{\text{das er kennt}}_{\text{NF}}$.

The Rangprobe

- Fields may be empty.

(87) $\underbrace{\text{Der Delphin}}_{\text{VF}}$ $\underbrace{\text{gibt}}_{\text{LS}}$ $\underbrace{\text{dem Kind den Ball,}}_{\text{MF}}$ $\underbrace{\text{das er kennt}}_{\text{NF}}$.

- Test: Rangprobe (Bech 1955: 72)

(88) a. Der Delphin hat [dem Kind] den Ball gegeben, [das er kennt].
the dolphin has the child the ball given who he knows
'The dolphin has given the ball to the child who it knows.'

The Rangprobe

- Fields may be empty.

(87) $\underbrace{\text{Der Delphin}}_{\text{VF}}$ $\underbrace{\text{gibt}}_{\text{LS}}$ $\underbrace{\text{dem Kind den Ball,}}_{\text{MF}}$ $\underbrace{\text{das er kennt}}_{\text{NF}}$.

- Test: Rangprobe (Bech 1955: 72)

(88) a. Der Delphin hat [dem Kind] den Ball gegeben, [das er kennt].
the dolphin has the child the ball given who he knows
'The dolphin has given the ball to the child who it knows.'

The Rangprobe

- Fields may be empty.

(87) $\underbrace{\text{Der Delphin}}_{\text{VF}}$ $\underbrace{\text{gibt}}_{\text{LS}}$ $\underbrace{\text{dem Kind den Ball,}}_{\text{MF}}$ $\underbrace{\text{das er kennt}}_{\text{NF}}$.

- Test: Rangprobe (Bech 1955: 72)

(88) a. Der Delphin hat [dem Kind] den Ball gegeben, [das er kennt].
the dolphin has the child the ball given who he knows
'The dolphin has given the ball to the child who it knows.'

b. * Der Delphin hat [dem Kind] den Ball, [das er kennt,] gegeben.
the dolphin has the child the ball who he knows given

Replacing the finite verb by an auxiliary forces the main verb into the right sentence bracket.

The Rangprobe

- Fields may be empty.

(87) $\underbrace{\text{Der Delphin}}_{\text{VF}}$ $\underbrace{\text{gibt}}_{\text{LS}}$ $\underbrace{\text{dem Kind den Ball,}}_{\text{MF}}$ $\underbrace{\text{das er kennt}}_{\text{NF}}$.

- Test: Rangprobe (Bech 1955: 72)

(88) a. Der Delphin hat [dem Kind] den Ball gegeben, [das er kennt].
the dolphin has the child the ball given who he knows
'The dolphin has given the ball to the child who it knows.'

b. * Der Delphin hat [dem Kind] den Ball, [das er kennt,] gegeben.
the dolphin has the child the ball who he knows given

Replacing the finite verb by an auxiliary forces the main verb into the right sentence bracket.

(89) Der Delphin hat [dem Kind, das er kennt,] den Ball gegeben.
the dolphin has the child who he knows the ball given

Recursion

- Reis (1980: 82): Recursion: Vorfeld can contain other topological fields:

(90) a. Die Möglichkeit, etwas zu verändern, ist damit verschüttet für lange lange Zeit.
the possibility something to change is there.with buried for long long time
'The possibility to change something will now be gone for a long, long time.'

b. [Verschüttet für lange lange Zeit] ist damit die Möglichkeit, etwas zu verändern.
buried for long long time ist there.with the possibility something to change

Recursion

- Reis (1980: 82): Recursion: Vorfeld can contain other topological fields:

(90) a. Die Möglichkeit, etwas zu verändern, ist damit verschüttet für lange lange Zeit.
the possibility something to change is there.with buried for long long time
'The possibility to change something will now be gone for a long, long time.'

b. [Verschüttet für lange lange Zeit] ist damit die Möglichkeit, etwas zu verändern.
buried for long long time ist there.with the possibility something to change

c. Wir haben schon seit langem gewußt, daß du kommst.
we have PART since long known that you come
'We have known for a while that you are coming.'

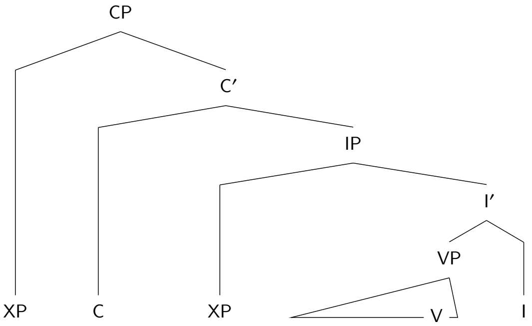
d. [Gewußt, daß du kommst,] haben wir schon seit langem.
known that you come have we PART since long

Exercise

Assign topological fields in the sentences in (91):

(91) a. Der Mann hat gewonnen, den alle kennen.
b. Sie gibt ihm das Buch, das Conny empfohlen hat.
c. Maria hat behauptet, dass das nicht stimmt.
d. Conny hat das Buch gelesen,
das Maria der Schülerin empfohlen hat,
die neu in die Klasse gekommen ist.
e. Komm!

The topological model paired with CP, IP, VP (I)



SpecCP prefield	C left SB	IP (without I, V) middle field		V , I right SB
		SpecIP subject position	phrases inside the VP	

German as SOV language

- Heads of VP and IP (V^0 and I^0) are serialized to the right of their arguments. Together they form the right sentence bracket.

German as SOV language

- Heads of VP and IP (V^0 and I^0) are serialized to the right of their arguments. Together they form the right sentence bracket.
- All other arguments and adjuncts are serialized to the left of them and form the Mittelfeld.

German as SOV language

- Heads of VP and IP (V^0 and I^0) are serialized to the right of their arguments. Together they form the right sentence bracket.
- All other arguments and adjuncts are serialized to the left of them and form the Mittelfeld.
- Typologically, German is a SOV language (basic order subject–object–verb), which is reflected at the D Structure level.
  - SOV German, …
  - SVO English, French, …
  - VSO Welsh, Arabic, …

  App. 40 % of all languages are SOV languages, app. 35 % are SVO.
- See Müller 2020a for discussion of Germanic and the classification of German.

German as SOV language

- Heads of VP and IP (V^0 and I^0) are serialized to the right of their arguments. Together they form the right sentence bracket.
- All other arguments and adjuncts are serialized to the left of them and form the Mittelfeld.
- Typologically, German is a SOV language (basic order subject–object–verb), which is reflected at the D Structure level.
  - SOV German, ...
  - SVO English, French, ...
  - VSO Welsh, Arabic, ...

  App. 40 % of all languages are SOV languages, app. 35 % are SVO.
- See Müller 2020a for discussion of Germanic and the classification of German.
- Nice result of SOV structure: The closer a constituent is related to the verb, the closer it is to the right sentence bracket, even in sentences with inital finite verb and empty right sentence bracket.

Motivation of SOV order as basic order: Particles

Bierwisch 1963: Verb particles form a close unit with the verb:

(92) a. weil sie morgen an-fängt
because she tomorrow PART-starts
'because he is starting tomorrow'

b. Sie fängt morgen an.
she starts tomorrow PART
'She is starting tomorrow.'

This unit can only be seen in verb-final structures,
which speaks for the fact that this structure reflects the base order.

Sometimes SOV is the only option

Sometimes SOV is the only option (Höhle 2018c: 370–371):

(93) a. weil sie das Stück heute ur-auf-führen
because they the play today PREF-PART-lead
'because they are performing the play for the first time today'

b. * Sie ur-auf-führen heute das Stück.
they PREF-PART-lead today the play

c. * Sie führen heute das Stück ur-auf.
they lead today the play PREF-PART

This is backformation.
Ur-auf-führung is wrongly assumed to be derived from the verb *uraufführen*.

Order in subordinated sentences

Verbs in non-finite subordinated clauses and in finite subordinated clauses introduced by a conjunction are positioned at the end (ignoring extraposition):

(94) a. Der Clown versucht, Kurt-Martin die Ware zu geben.
the clown tries Kurt-Martin the goods to give
'The clown is trying to give Kurt-Martin the goods.'

b. dass der Clown Kurt-Martin die Ware gibt
that the clown Kurt-Martin the goods gives
'that the clown gives Kurt-Martin the goods'

Order of verbs in SVO and SOV languages

Ørsnes (2009):

(95) a. dass er ihn gesehen$_3$ haben$_2$ muss$_1$ (German)
that he him seen have must

b. at han må$_1$ have$_2$ set$_3$ ham (Danish)
that he must have seen him
'that he must have seen him'

OV: embedding verbs go to the end
VO: embedding verbs go to the beginning

(ignore the Dutch for the moment …)

Scope

Netter 1992: Adverbs outscope material to their right (preference only?):

(96) a. dass er [absichtlich [nicht lacht]]
that he intentionally not laughs
'that he is intentionally not laughing'

b. dass er [nicht [absichtlich lacht]]
that he not intentionally laughs
'that he is not laughing intentionally'

Scope

Netter 1992: Adverbs outscope material to their right (preference only?):

(96) a. dass er [absichtlich [nicht lacht]]
that he intentionally not laughs
'that he is intentionally not laughing'

b. dass er [nicht [absichtlich lacht]]
that he not intentionally laughs
'that he is not laughing intentionally'

The scoping does not change if the verb is in initial position:

(97) a. Er lacht$_i$ [absichtlich [nicht $\_\_{}_i$]].
he laughs intentionally not
'He is intentionally not laughing.'

b. Er lacht$_i$ [nicht [absichtlich $\_\_{}_i$]].
he laughs not intentionally
'He is not laughing intentionally.'

C^0 – The left sentence bracket in embedded clauses

C^0 corresponds to the left sentence bracket and is filled as follows:

- In embedded sentences with subordinating conjunction the conjunction (the complementizer) is placed in C^0, as in English. The verb stays in the right sentence bracket.

(98) dass jeder diese Frau kennt
that everybody this woman knows
'that everybody knows this woman'

C^0 – The left sentence bracket in embedded clauses

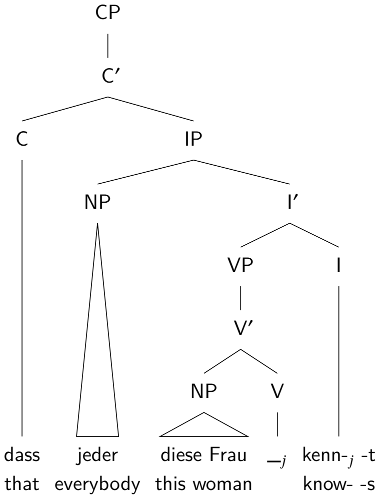
C^0 corresponds to the left sentence bracket and is filled as follows:

- In embedded sentences with subordinating conjunction
  the conjunction (the complementizer) is placed in C^0, as in English.
  The verb stays in the right sentence bracket.

  (98) dass jeder diese Frau kennt
  that everybody this woman knows
  'that everybody knows this woman'

- The verb moves from V to I.

V to I movement in embedded clauses



C^0 – The left sentence bracket in V1 and V2 clauses

- The finite verb is moved via I^0 to C^0 in verb-first and verb-second clauses: $V^0 \rightarrow I^0 \rightarrow C^0$.

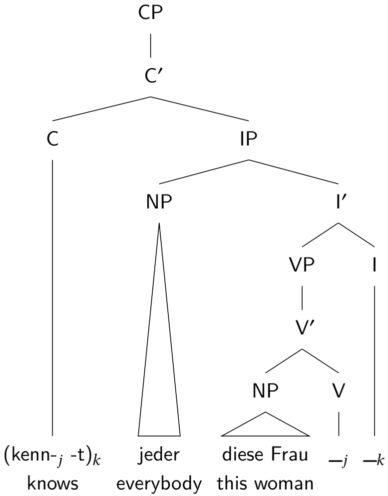
(99)

a. dass jeder diese Frau kenn- -t (verb in V^0)
that everybody this woman know- -s

b. dass jeder diese Frau $\_\_{}_i$ [kenn-$_i$ -t] (verb in I^0)
that everybody this woman know- -s

c. [Kenn-$_i$ -t]$_j$ jeder diese Frau $\_\_{}_i$ $\_\_{}_j$? (verb in C^0)
know- -s everybody this woman

V to I to C movement in V1/V2 clauses



SpecCP – The Vorfeld in declarative clauses (I)

The position SpecCP corresponds to the Vorfeld and is filled as follows:

- Declarative clauses: XP is moved to the Vorfeld.

(100) Gibt der Mann dem Kind jetzt den Mantel?
gives the.NOM man the.DAT child now the.ACC coat
'Is the man going to give the child the coat now?'

(101) a. Der Mann gibt dem Kind jetzt den Mantel.
the.NOM man gives the.DAT child now the.ACC coat
'The man is giving the child the coat now.'

SpecCP – The Vorfeld in declarative clauses (I)

The position SpecCP corresponds to the Vorfeld and is filled as follows:

- Declarative clauses: XP is moved to the Vorfeld.

(100) Gibt der Mann dem Kind jetzt den Mantel?
gives the.NOM man the.DAT child now the.ACC coat
'Is the man going to give the child the coat now?'

(101) a. Der Mann gibt dem Kind jetzt den Mantel.
the.NOM man gives the.DAT child now the.ACC coat
'The man is giving the child the coat now.'

b. Dem Kind gibt der Mann jetzt den Mantel.
the.DAT child gives the.NOM man now the.ACC coat

SpecCP – The Vorfeld in declarative clauses (I)

The position SpecCP corresponds to the Vorfeld and is filled as follows:

- Declarative clauses: XP is moved to the Vorfeld.

(100) Gibt der Mann dem Kind jetzt den Mantel?
gives the.NOM man the.DAT child now the.ACC coat
'Is the man going to give the child the coat now?'

(101) a. Der Mann gibt dem Kind jetzt den Mantel.
the.NOM man gives the.DAT child now the.ACC coat
'The man is giving the child the coat now.'

b. Dem Kind gibt der Mann jetzt den Mantel.
the.DAT child gives the.NOM man now the.ACC coat

c. Den Mantel gibt der Mann dem Kind jetzt.
the.ACC coat gives the.NOM man the.DAT child now

SpecCP – The Vorfeld in declarative clauses (I)

The position SpecCP corresponds to the Vorfeld and is filled as follows:

- Declarative clauses: XP is moved to the Vorfeld.

(100)

Gibt	der	Mann	dem	Kind	jetzt	den	Mantel?
gives	the.NOM	man	the.DAT	child	now	the.ACC	coat

'Is the man going to give the child the coat now?'

(101) a.

Der	Mann	gibt	dem	Kind	jetzt	den	Mantel.
the.NOM	man	gives	the.DAT	child	now	the.ACC	coat

'The man is giving the child the coat now.'

b.

Dem	Kind	gibt	der	Mann	jetzt	den	Mantel.
the.DAT	child	gives	the.NOM	man	now	the.ACC	coat

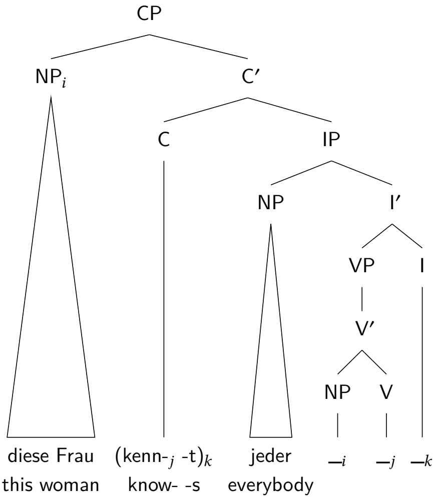
c.

Den	Mantel	gibt	der	Mann	dem	Kind	jetzt.
the.ACC	coat	gives	the.NOM	man	the.DAT	child	now

d.

Jetzt	gibt	der	Mann	dem	Kind	den	Mantel.
now	gives	the.NOM	man	the.DAT	child	the.ACC	coat

Verb movement and movement to SpecCP



SpecCP – The Vorfeld in declarative clauses (II)

- The crucial factor for deciding which phrase to move is the *information structure* of the sentence. Material connected to previously mentioned or otherwise-known information is placed further left (preferably in the prefield) and new information tends to occur to the right. Fronting to the prefield in declarative clauses is often referred to as topicalization.

SpecCP – The Vorfeld in declarative clauses (II)

- The crucial factor for deciding which phrase to move is the *information structure* of the sentence. Material connected to previously mentioned or otherwise-known information is placed further left (preferably in the prefield) and new information tends to occur to the right. Fronting to the prefield in declarative clauses is often referred to as topicalization.
- But this is rather a misnomer, since the focus (informally: the constituent being asked for) can also occur in the prefield. Expletives as well.

SpecCP – The Vorfeld in declarative clauses (II)

- The crucial factor for deciding which phrase to move is the *information structure* of the sentence. Material connected to previously mentioned or otherwise-known information is placed further left (preferably in the prefield) and new information tends to occur to the right. Fronting to the prefield in declarative clauses is often referred to as topicalization.
- But this is rather a misnomer, since the focus (informally: the constituent being asked for) can also occur in the prefield. Expletives as well.
- Caution:
  Movement to the Vorfeld does not have the same status as fronting in English!

Nonlocal dependencies

- Analysis also works for nonlocal dependencies:

(102) [Um zwei Millionen Mark]$_i$ soll er versucht haben,
around two million Deutsche.Marks should he tried have
[eine Versicherung $\_\_{}_i$ zu betrügen].[1]
an insurance.company to deceive
'He apparently tried to cheat an insurance company out of two million Deutsche Marks.'

Step-wise movement: the fronted constituent first moves to the specifier position of the phrase it originates from than to the next specifier of the next maximal projection and so on until it reaches the uppermost SpecCP position.

[1] taz, 04.05.2001, p. 20.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \textit{ Grammatik } \rangle \\ \text{SYN|CAT|SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ 语法 } \rangle \\ \text{SYN|CAT|SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ دستور } \rangle \\ \text{SYN|CAT|SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ व्याकरण } \rangle \\ \text{SYN|CAT|SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Government & Binding: Passive and local reordering

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Reading material

Müller 2020b: Section 3.4–3.5

Case and case principles

- What types of case exist?

Case and case principles

- What types of case exist?
- In which way does case depend on syntactic context?

Case and case principles

- What types of case exist?
- In which way does case depend on syntactic context?
- One way to capture case requirements is to list them in valence representations.
  If we understand the regularities, we can avoid this.
  We capture regularities and need just one lexical item for verbs like *lesen* 'read':

(103) a. Er möchte das Buch lesen.
he.NOM wants the book read

b. Ich sah ihn das Buch lesen.
I saw him.ACC the book read

The case of the subject (and the object) is determined by the principle.

Structural case: The subject

- If case depends on the syntactic environment, it is called structural case. Otherwise it is lexical case.

Structural case: The subject

- If case depends on the syntactic environment, it is called structural case. Otherwise it is lexical case.
- Subject (nominative in the active) can be realized as accusative and genitive:

(104) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

Structural case: The subject

- If case depends on the syntactic environment, it is called structural case. Otherwise it is lexical case.
- Subject (nominative in the active) can be realized as accusative and genitive:

(104) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

b. Der Mann lässt den Installateur kommen.
the man lets the.ACC plumber come
'The man is getting the plumber to come.'

Structural case: The subject

- If case depends on the syntactic environment, it is called structural case. Otherwise it is lexical case.
- Subject (nominative in the active) can be realized as accusative and genitive:

(104) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

b. Der Mann lässt den Installateur kommen.
the man lets the.ACC plumber come
'The man is getting the plumber to come.'

c. das Kommen des Installateurs
the coming of.the plumber
'the plumber's visit'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(105) a. Judit schlägt den Weltmeister.
Judit beats the.ACC world.champion
'Judit beats the world champion.'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(105) a. Judit schlägt den Weltmeister.
Judit beats the.ACC world.champion
'Judit beats the world champion.'

b. Der Weltmeister wird geschlagen.
the.NOM world.champion is beaten
'The world champion is being beaten.'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(105) a. Judit schlägt den Weltmeister.
Judit beats the.ACC world.champion
'Judit beats the world champion.'

b. Der Weltmeister wird geschlagen.
the.NOM world.champion is beaten
'The world champion is being beaten.'

c. das Schlagen des Weltmeisters
the beating of.the world.champion

Lexical case

- genitive depending on the verb is lexical case:
  The case of the genitive object does not change in passivization.

(106) a. Wir gedenken der Opfer.
we remember the.GEN victims

b. Der Opfer wird gedacht.
the.GEN victims are remembered
'The victims are being remembered.'

c. * Die Opfer wird / werden gedacht.
the.NOM victims is are remembered

Lexical case

- genitive depending on the verb is lexical case:
  The case of the genitive object does not change in passivization.

(106) a. Wir gedenken der Opfer.
we remember the.GEN victims

b. Der Opfer wird gedacht.
the.GEN victims are remembered
'The victims are being remembered.'

c. * Die Opfer wird / werden gedacht.
the.NOM victims is / are remembered

(106b) = impersonal passive, there is no subject.

Is the dative a lexical case?

- Similarly there is no change in the passive with dative objects:

(107) a. Der Mann hat ihm geholfen.
the man has him.DAT helped
'The man has helped him.'

b. Ihm wird geholfen.
him.DAT is helped
'He is being helped.'

Is the dative a lexical case?

- Similarly there is no change in the passive with dative objects:

(107) a. Der Mann hat ihm geholfen.
the man has him.DAT helped
'The man has helped him.'

b. Ihm wird geholfen.
him.DAT is helped
'He is being helped.'

- But what about (108)?

(108) a. Der Mann hat den Ball dem Jungen geschenkt.
the man has the ball the.DAT boy given

b. Der Junge bekam den Ball geschenkt.
the.NOM boy got the ball given

Dative structural or lexical?

- The status of the dative is controversial.
  Three options:

1. All datives are lexical.

Dative structural or lexical?

- The status of the dative is controversial.
  Three options:
  1. All datives are lexical.
  2. Some datives are lexical, some structural.

Dative structural or lexical?

- The status of the dative is controversial.
  Three options:
  1. All datives are lexical.
  2. Some datives are lexical, some structural.
  3. All datives are structural.

1. The dative as lexical case

- If the dative is treated as a lexical case,
  the dative has to change in the dative passive from lexical to structural.

1. The dative as lexical case

- If the dative is treated as a lexical case,
  the dative has to change in the dative passive from lexical to structural.
- Haider's examples in (109) are immediately explained (1986: 20):

(109) a. Er streichelt den Hund.
he strokes the dog

b. Der Hund wurde gestreichelt.
the dog was stroked

c. sein Streicheln des Hundes
his stroking of.the dog

1. The dative as lexical case

- If the dative is treated as a lexical case,
  the dative has to change in the dative passive from lexical to structural.
- Haider's examples in (109) are immediately explained (1986: 20):

(109)
a. Er streichelt den Hund.
   he strokes the dog
b. Der Hund wurde gestreichelt.
   the dog was stroked
c. sein Streicheln des Hundes
   his stroking of.the dog
d. Er hilft den Kindern.
   he helps the.DAT children
e. Den Kindern wurde geholfen.
   the.DAT children was helped
f. das Helfen der Kinder
   the helping of.the children
   (children agent only)
g. * sein Helfen der Kinder
   his helping of.the children

1. The dative as lexical case

- If the dative is treated as a lexical case,
  the dative has to change in the dative passive from lexical to structural.
- Haider's examples in (109) are immediately explained (1986: 20):

(109)
a. Er streichelt den Hund.
   he strokes the dog
b. Der Hund wurde gestreichelt.
   the dog was stroked
c. sein Streicheln des Hundes
   his stroking of.the dog
d. Er hilft den Kindern.
   he helps the.DAT children
e. Den Kindern wurde geholfen.
   the.DAT children was helped
f. das Helfen der Kinder
   the helping of.the children
   (children agent only)
g. * sein Helfen der Kinder
   his helping of.the children

- Dative can only be expressed prenominally:

(110) das Den-Kindern-Helfen
      the the-children-helping

All datives structural? Structural case and bivalent verbs

- If structural/lexical is the only distinction available, there is a problem with bivalent verbs:

(111) a. Er hilft ihm.
he helps him.DAT

b. Er unterstützt ihn.
he supports him.ACC

There has to be a difference between *helfen* and *unterstützen*.
Just saying the verbs require structural case, would not be enough.

All datives structural? Structural case and bivalent verbs

- If structural/lexical is the only distinction available, there is a problem with bivalent verbs:

(111) a. Er hilft ihm.
he helps him.DAT

b. Er unterstützt ihn.
he supports him.ACC

There has to be a difference between *helfen* and *unterstützen*.
Just saying the verbs require structural case, would not be enough.

- For ditransitive verbs one can derive the dative case from general principles (Nom, Dat, Acc), but this does not work for bivalent verbs.
→ Dative of *helfen* is assumed to be lexical (mixed approach).
Prediction: dative passive is not possible with two-place verbs.

Dative passive with bivalent verbs

(112) a. Er kriegte von vielen geholfen / gratuliert / applaudiert.
b. Man kriegt täglich gedankt.

Dative passive with bivalent verbs

(112) a. Er kriegte von vielen geholfen / gratuliert / applaudiert.
b. Man kriegt täglich gedankt.

Attested data:

(113) a. „Da kriege ich geholfen.“[2]
b. Heute morgen bekam ich sogar schon gratuliert.[3]
c. „Klärle“ hätte es wirklich mehr als verdient, auch mal zu einem „unrunden“ Geburtstag gratuliert zu bekommen.[4]
d. Mit dem alten Titel von Elvis Presley „I can't help falling in love“ bekam Kassier Markus Reiß zum Geburtstag gratuliert, […][5]

Hence: Haider' approach: all datives have lexical case + trick for dative passive.

[2] Frankfurter Rundschau, 26.06.1998, S. 7.
[3] Brief von Irene G. an Ernst G. vom 10.04.1943, Feldpost-Archive mkb-fp-0270
[4] Mannheimer Morgen, 28.07.1999, Lokales; „Klärle“ feiert heute Geburtstag.
[5] Mannheimer Morgen, 21.04.1999, Lokales; Motor des gesellschaftlichen Lebens.

Case assignment

- Lexical case is assigned by the verb.

Case assignment

- Lexical case is assigned by the verb.
- Verbs assign object case (accusative),
  if the object has structural case.

Case assignment

- Lexical case is assigned by the verb.
- Verbs assign object case (accusative),
  if the object has structural case.
- Finite Infl (or T in more recent versions) assigns nominative to the subject.

Case assignment

- Lexical case is assigned by the verb.
- Verbs assign object case (accusative),
  if the object has structural case.
- Finite Infl (or T in more recent versions) assigns nominative to the subject.

- Case filter: Every NP has to have case.

Case assignment

- Lexical case is assigned by the verb.
- Verbs assign object case (accusative),
  if the object has structural case.
- Finite Infl (or T in more recent versions) assigns nominative to the subject.

- Case filter: Every NP has to have case.

- Case is assigned under government, that is,
  only NPs in certain tree positions may get case.

Case and passive as movement

Assumptions regarding case and passive:

- The subject gets case from I, the other arguments get case from V.

Case and passive as movement

Assumptions regarding case and passive:

- The subject gets case from I, the other arguments get case from V.
- The passive blocks the subject (in the lexicon).

Case and passive as movement

Assumptions regarding case and passive:

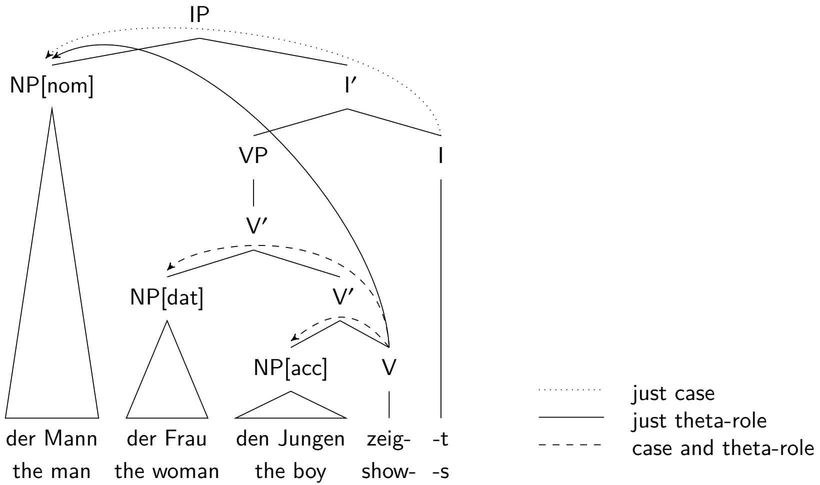
- The subject gets case from I, the other arguments get case from V.
- The passive blocks the subject (in the lexicon).
- The accusative object gets a theta role but no case.

Case and passive as movement

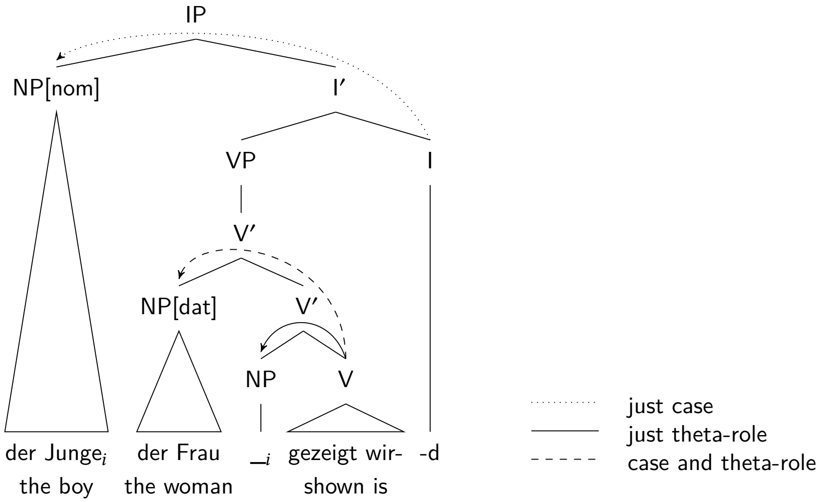
Assumptions regarding case and passive:

- The subject gets case from I, the other arguments get case from V.
- The passive blocks the subject (in the lexicon).
- The accusative object gets a theta role but no case.
- Therefore it has to move to a position where it gets case (move to SpecIP).

Case and theta role assignment in the active



Case and theta role assignment in the passive



Remarks on passive as movement analyses

- The analysis works for English: the object has to move.

(114) a. The mother gave [the girl] [a cookie].
b. [The girl] was given [a cookie] (by the mother).

Remarks on passive as movement analyses

- The analysis works for English: the object has to move.

(114) a. The mother gave [the girl] [a cookie].
b. [The girl] was given [a cookie] (by the mother).

- But this is not the case for German:

(115) a.

weil	das	Mädchen	dem	Jungen	den	Ball	schenkte
because	the.NOM	girl	the.DAT	boy	the.ACC	ball	gave

‘because the girl gave the ball to the boy’

b.

weil	dem	Jungen	der	Ball	geschenkt	wurde
because	the.DAT	boy	the.NOM	ball	given	was

‘because the ball was given to the boy’

c.

weil	der	Ball	dem	Jungen	geschenkt	wurde
because	the.NOM	ball	the.DAT	boy	given	was

(115b) is the unmarked order (Höhle 1982), not (115c).

Remarks on passive as movement analyses

- The analysis works for English: the object has to move.

(114) a. The mother gave [the girl] [a cookie].
b. [The girl] was given [a cookie] (by the mother).

- But this is not the case for German:

(115) a.

weil	das	Mädchen	dem	Jungen	den	Ball	schenkte
because	the.NOM	girl	the.DAT	boy	the.ACC	ball	gave

‘because the girl gave the ball to the boy’

b.

weil	dem	Jungen	der	Ball	geschenkt	wurde
because	the.DAT	boy	the.NOM	ball	given	was

‘because the ball was given to the boy’

c.

weil	der	Ball	dem	Jungen	geschenkt	wurde
because	the.NOM	ball	the.DAT	boy	given	was

(115b) is the unmarked order (Höhle 1982), not (115c). That is: nothing has to be moved.

- Solution: abstract movement. (empty expletive in subject position)

Remarks on passive as movement analyses

- The analysis works for English: the object has to move.

(114) a. The mother gave [the girl] [a cookie].
 b. [The girl] was given [a cookie] (by the mother).

- But this is not the case for German:

(115) a. weil das Mädchen dem Jungen den Ball schenkte
 because the.NOM girl the.DAT boy the.ACC ball gave
 'because the girl gave the ball to the boy'
 b. weil dem Jungen der Ball geschenkt wurde
 because the.DAT boy the.NOM ball given was
 'because the ball was given to the boy'
 c. weil der Ball dem Jungen geschenkt wurde
 because the.NOM ball the.DAT boy given was

(115b) is the unmarked order (Höhle 1982), not (115c). That is: nothing has to be moved.

- Solution: abstract movement. (empty expletive in subject position)
- We will learn about alternative analyses not relying on such complicated mechanisms.

Local reordering

The arguments of verbs can appear in any order in German.
So for verbs with three arguments, there are six possible orders for the arguments:

(116) a. [weil] der Mann dem Kind das Buch gibt
because the.NOM man the.DAT child the.ACC book gives
'because the man gives the book to the child'

b. [weil] der Mann das Buch dem Kind gibt
because the.NOM man the.ACC book the.DAT child gives

c. [weil] das Buch der Mann dem Kind gibt
because the.ACC book the.NOM man the.DAT child gives

d. [weil] das Buch dem Kind der Mann gibt
because the.ACC book the.DAT child the.NOM man gives

e. [weil] dem Kind der Mann das Buch gibt
because the.DAT child the.NOM man the.ACC book gives

f. [weil] dem Kind das Buch der Mann gibt
because the.DAT child the.ACC book the.NOM man gives

(116a) is the so-called unmarked order (Höhle 1982).
The number of contexts in which sentences can be used is restricted for all other sentences in (116).

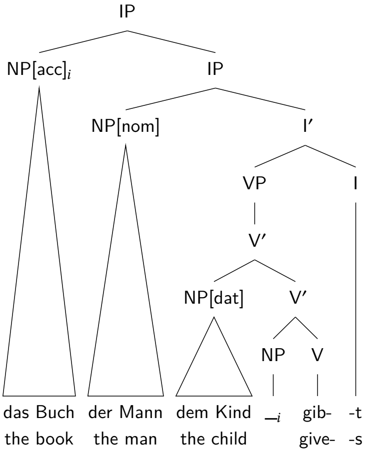
Movement or base-generation

- Two suggestions:
  - Assumption of a base order and derivation of all other orders by movement (Frey 1993).

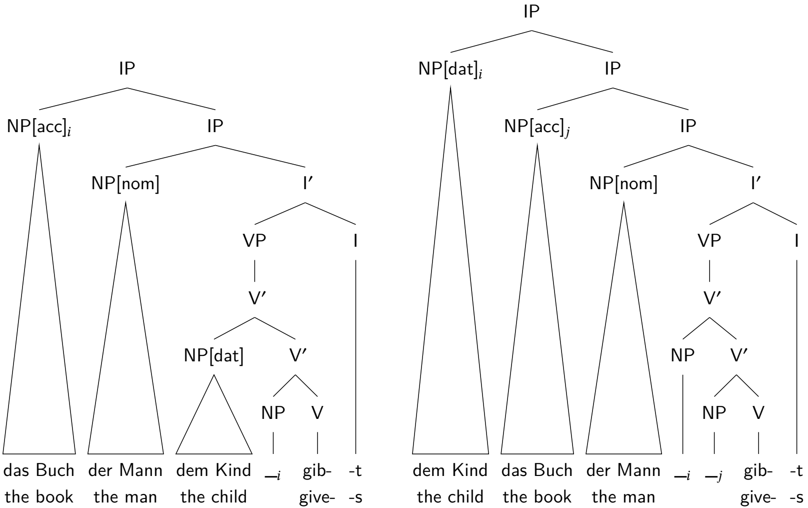
Movement or base-generation

- Two suggestions:
  - Assumption of a base order and derivation of all other orders by movement (Frey 1993).
  - Base generation: all orders are derived in the phrase structure component without movement (Fanselow 2001).

Movement


IP
NP[acc]$_i$
IP
NP[nom]
I′
VP
I
V′
NP[dat]
V′
NP
V
das Buch
the book
der Mann
the man
dem Kind
the child
$\_\_i$
gib-
give-
-t
-s

Movement


IP
NP[acc]$_i$
IP
NP[nom]
I′
VP
I
V′
NP[dat]
V′
NP
V
das Buch
the book
der Mann
the man
dem Kind
the child
—$_i$
gib-
give-
-t
-s
IP
NP[dat]$_i$
IP
NP[acc]$_j$
IP
NP[nom]
I′
VP
I
V′
NP
V′
NP
V
dem Kind
the child
das Buch
the book
der Mann
the man
—$_i$
—$_j$
gib-
give-
-t
-s

Problems of movement approaches: Quantifier scope

- Quantifier scope as motivation for movement-based approaches (Frey 1993):

(117) Es ist nicht der Fall, daß er mindestens einem Verleger fast jedes Gedicht
it is not the case that he at.least one publisher almost every poem
anbot.
offered
'It is not the case that he offered at least one publisher almost every poem.'

(117) has only one reading in which *at least one* scopes over *almost every*.

Problems of movement approaches: Quantifier scope

- Quantifier scope as motivation for movement-based approaches (Frey 1993):

(117) Es ist nicht der Fall, daß er mindestens einem Verleger fast jedes Gedicht anbot.
it is not the case that he at.least one publisher almost every poem offered
'It is not the case that he offered at least one publisher almost every poem.'

(117) has only one reading in which *at least one* scopes over *almost every*.

(118) Es ist nicht der Fall, daß er fast jedes Gedicht$_i$ mindestens einem Verleger $\__i$ anbot.
it is not the case that he almost every poem at.least one publisher offered
'It is not the case that he offered almost every poem to at least one publisher.'

(118) has two readings.
One corresponds to the surface realization and one to the reading of (117).

Quantifier scope: Movement and recreation

- Idea: Reconstruction of the moved items at D structure position.

(119) Es ist nicht der Fall, daß er fast jedes Gedicht$_i$ mindestens einem Verleger $\_{}_i$ anbot.
it is not the case that he almost every poem at.least one publisher offered
'It is not the case that he offered almost every poem to at least one publisher.'

Quantifier scope: Movement and recreation

- Idea: Reconstruction of the moved items at D structure position.

(119) Es ist nicht der Fall, daß er fast jedes Gedicht$_i$ mindestens einem Verleger $\_\_i$ anbot.
it is not the case that he almost every poem at.least one publisher offered
'It is not the case that he offered almost every poem to at least one publisher.'

- But this causes problems with two moved NPs (Kiss 2001; Fanselow 2001):

(120) Ich glaube, dass mindestens einem Verleger$_i$ fast jedes Gedicht$_j$ nur dieser Dichter $\_\_i$ $\_\_j$ angeboten hat.
I believe that at.least one publisher almost every poem only this poet offered has
'I think that only this poet offered almost every poem to at least one publisher.'

Reconstructing *mindestens einem Verleger* corresponds to a non-exiting reading. If two items are moved. Their relative scope is fixed. They cannot reconstruct independently.

Fix involving additional movements, some at PF

- Sauerland & Elbourne (2002) discuss the same problem in movement-based approaches to Japanese (in the Minimalist Program).

Fix involving additional movements, some at PF

- Sauerland & Elbourne (2002) discuss the same problem in movement-based approaches to Japanese (in the Minimalist Program).
- They suggest solving the problem by assuming additional movements some of them optionally taking place at PF without having semantic effects.

Fix involving additional movements, some at PF

- Sauerland & Elbourne (2002) discuss the same problem in movement-based approaches to Japanese (in the Minimalist Program).
- They suggest solving the problem by assuming additional movements some of them optionally taking place at PF without having semantic effects.
- The resulting analysis is highly complex and involves additional assumptions, which begs the question as how such complex systems should be acquirable.

Base generation

- Alternative: allow for the verb to combine with its arguments in any order.
  Fanselow (2001): a base generation analysis (in Minimalism)

Base generation

- Alternative: allow for the verb to combine with its arguments in any order. Fanselow (2001): a base generation analysis (in Minimalism)
- No account for (121) in IP approach, since objects are before subject:

(121)

a.	[weil]	das	Buch	der	Mann	dem	Kind	gibt
	because	the.ACC	book	the.NOM	man	the.DAT	child	gives
b.	[weil]	das	Buch	dem	Kind	der	Mann	gibt
	because	the.ACC	book	the.DAT	child	the.NOM	man	gives
c.	[weil]	dem	Kind	der	Mann	das	Buch	gibt
	because	the.DAT	child	the.NOM	man	the.ACC	book	gives
d.	[weil]	dem	Kind	das	Buch	der	Mann	gibt
	because	the.DAT	child	the.ACC	book	the.NOM	man	gives

Base generation

- Alternative: allow for the verb to combine with its arguments in any order.
  Fanselow (2001): a base generation analysis (in Minimalism)
- No account for (121) in IP approach, since objects are before subject:

(121)

a. [weil] das Buch der Mann dem Kind gibt
because the.ACC book the.NOM man the.DAT child gives

b. [weil] das Buch dem Kind der Mann gibt
because the.ACC book the.DAT child the.NOM man gives

c. [weil] dem Kind der Mann das Buch gibt
because the.DAT child the.NOM man the.ACC book gives

d. [weil] dem Kind das Buch der Mann gibt
because the.DAT child the.ACC book the.NOM man gives

- IP-less base generation approach seems to be the best option.
  (also adopted in Categorial Grammar and HPSG)

Base generation

- Alternative: allow for the verb to combine with its arguments in any order. Fanselow (2001): a base generation analysis (in Minimalism)
- No account for (121) in IP approach, since objects are before subject:

(121) a. [weil] das Buch der Mann dem Kind gibt
because the.ACC book the.NOM man the.DAT child gives

b. [weil] das Buch dem Kind der Mann gibt
because the.ACC book the.DAT child the.NOM man gives

c. [weil] dem Kind der Mann das Buch gibt
because the.DAT child the.NOM man the.ACC book gives

d. [weil] dem Kind das Buch der Mann gibt
because the.DAT child the.ACC book the.NOM man gives

- IP-less base generation approach seems to be the best option. (also adopted in Categorial Grammar and HPSG)
- Theta roles are assigned in tandem with argument selection. Not to positions.

Summary

Goals:

- Capture relations between certain structures, for example:
  - active/passive
  - verb last/verb initial/verb second position
  - almost free order of constituents in the Mittelfeld and a certain base order

  mapping from D Structure to S Structure

Summary

Goals:

- Capture relations between certain structures, for example:
  - active/passive
  - verb last/verb initial/verb second position
  - almost free order of constituents in the Mittelfeld and a certain base order

  mapping from D Structure to S Structure
- Explanation of language acquisition by
  - assumption of a general rule schema holding for all languages and all structures ($\overline{\text{X}}$ Theory)
  - general principles holding for all languages but parameterizable

Exercise

Draw the syntax trees for the fowllowing sentences:

(122) a.

dass	der	Delphin	dem	Kind	hilft
that	the.NOM	dolphin	the.DAT	child	helps

‘that the dolphin helps the child’

b.

dass	der	Delphin	den	Hai	attackiert
that	the.NOM	dolphin	the.ACC	shark	attacks

‘that the dolphin attacks the shark’

c.

dass	der	Hai	attackiert	wird
that	the.NOM	shark	attacked	is

‘that the shark is attacked’

d.

Der	Hai	wird	attackiert.
the.NOM	shark	is	attacked

‘The shark is attacked.’

e.

Der	Delphin	hilft	dem	Kind.
the	dolphin.NOM	helps	the.DAT	child

‘The dolphin is helping the child.’

$$\begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \textit{ Grammatik } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ 语法 } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ دستور } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH } \langle \text{ व्याकरण } \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT } \langle \text{ DET } \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST } \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Generalized Phrase Structure Grammar (GPSG)

Stefan Müller
Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\mathrm{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoning Grammar (TAG)

Reading material

Müller 2020b: Chapter 5 without Section 5.1.4 about semantics.

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.
- Chomsky showed PSGs to be inadequate.
  GPSG extends PSG in ways that make it possible to address Chomsky's monita:
  - categories may be complex (Harman 1963)

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.
- Chomsky showed PSGs to be inadequate.
  GPSG extends PSG in ways that make it possible to address Chomsky's monita:
  - categories may be complex (Harman 1963)
  - different treatment of local reordering

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.
- Chomsky showed PSGs to be inadequate.
  GPSG extends PSG in ways that make it possible to address Chomsky's monita:
  - categories may be complex (Harman 1963)
  - different treatment of local reordering
  - passive as metarule

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.
- Chomsky showed PSGs to be inadequate.
  GPSG extends PSG in ways that make it possible to address Chomsky's monita:
  - categories may be complex (Harman 1963)
  - different treatment of local reordering
  - passive as metarule
  - non-local dependencies as a series of local dependencies

Generalized Phrase Structure Grammar (GPSG)

- GPSG was developed as an answer to Transformational Grammar at the end of the 1970s.
- Main publication: Gazdar, Klein, Pullum & Sag (1985)
- Uszkoreit (1987) developed large GPSG fragment of German.
- Chomsky showed PSGs to be inadequate.
  GPSG extends PSG in ways that make it possible to address Chomsky's monita:
  - categories may be complex (Harman 1963)
  - different treatment of local reordering
  - passive as metarule
  - non-local dependencies as a series of local dependencies
- We will deal with each of these innovations in what follows.

General remarks on the representational format

- Categories are sets of feature value pairs.

General remarks on the representational format

- Categories are sets of feature value pairs.
- Lexical entries have a feature SUBCAT. The value is a number which says something about the kind of grammatical rules in which the word can be used.

General remarks on the representational format

- Categories are sets of feature value pairs.
- Lexical entries have a feature SUBCAT. The value is a number which says something about the kind of grammatical rules in which the word can be used.
- Examples from Uszkoreit 1987:

V2 → H[5]	(kommen ‘come’, schlafen ‘sleep’)
V2 → H[6], N2[Case Acc]	(kennen ‘know’, suchen ‘search’)
V2 → H[7], N2[Case Dat]	(helfen ‘help’, vertrauen ‘trust’)
V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben ‘give’, zeigen ‘show’)
V2 → H[9], V3[+dass]	(wissen ‘know’, glauben ‘believe’)

These rules license VPs: the combination verb & complements, but not subject.

General remarks on the representational format

- Categories are sets of feature value pairs.
- Lexical entries have a feature SUBCAT. The value is a number which says something about the kind of grammatical rules in which the word can be used.
- Examples from Uszkoreit 1987:

V2 → H[5]	(kommen 'come', schlafen 'sleep')
V2 → H[6], N2[Case Acc]	(kennen 'know', suchen 'search')
V2 → H[7], N2[Case Dat]	(helfen 'help', vertrauen 'trust')
V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben 'give', zeigen 'show')
V2 → H[9], V3[+dass]	(wissen 'know', glauben 'believe')

  These rules license VPs: the combination verb & complements, but not subject.
- The numbers following the category symbols (V or N) indicate the $\overline{\text{X}}$ level. The maximum level of a verbal projection is three rather than two.

General remarks on the representational format

- Categories are sets of feature value pairs.
- Lexical entries have a feature SUBCAT. The value is a number which says something about the kind of grammatical rules in which the word can be used.
- Examples from Uszkoreit 1987:

V2 → H[5]	(kommen 'come', schlafen 'sleep')
V2 → H[6], N2[Case Acc]	(kennen 'know', suchen 'search')
V2 → H[7], N2[Case Dat]	(helfen 'help', vertrauen 'trust')
V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben 'give', zeigen 'show')
V2 → H[9], V3[+dass]	(wissen 'know', glauben 'believe')

  These rules license VPs: the combination verb & complements, but not subject.
- The numbers following the category symbols (V or N) indicate the $\overline{\text{X}}$ level. The maximum level of a verbal projection is three rather than two.
- H stands for Head.

Principles: The Head Feature Convention

Head Feature Convention:
The mother node and the head daughter must bear the same head features unless indicated otherwise.

Metarules and ID/LP format

Two further innovations of GPSG:

- Metarules: Additional phrase structure rules are licensed via metarules.
- ID/LP format: Constraints on linearization are separated from immediate dominance.

These two tools will be discussed with respect to our set of phenomena.

Local reordering

- Arguments can appear in almost any order in the German Mittelfeld.

(123)

a.	[weil]	der	Mann	dem	Kind	das	Buch	gibt
	because	the.NOM	man	the.DAT	child	the.ACC	book	gives
	'because the man gives the book to the child'							
b.	[weil]	der	Mann	das	Buch	dem	Kind	gibt
	because	the.NOM	man	the.ACC	book	the.DAT	child	gives
c.	[weil]	das	Buch	der	Mann	dem	Kind	gibt
	because	the.ACC	book	the.NOM	man	the.DAT	child	gives
d.	[weil]	das	Buch	dem	Kind	der	Mann	gibt
	because	the.ACC	book	the.DAT	child	the.NOM	man	gives
e.	[weil]	dem	Kind	der	Mann	das	Buch	gibt
	because	the.DAT	child	the.NOM	man	the.ACC	book	gives
f.	[weil]	dem	Kind	das	Buch	der	Mann	gibt
	because	the.DAT	child	the.ACC	book	the.NOM	man	gives

Motivation for linearization rules (I)

Motivation: Permutation with phrase structure rules →
we need six phrase structure rules for ditransitive verbs in verb-final position:

(124) S → NP[nom] NP[dat] NP[acc] V
S → NP[nom] NP[acc] NP[dat] V
S → NP[acc] NP[nom] NP[dat] V
S → NP[acc] NP[dat] NP[nom] V
S → NP[dat] NP[nom] NP[acc] V
S → NP[dat] NP[acc] NP[nom] V

Motivation for linearization rules (II)

Plus six rules for verb-initial position:

(125) S → V NP[nom] NP[dat] NP[acc]
S → V NP[nom] NP[acc] NP[dat]
S → V NP[acc] NP[nom] NP[dat]
S → V NP[acc] NP[dat] NP[nom]
S → V NP[dat] NP[nom] NP[acc]
S → V NP[dat] NP[acc] NP[nom]

A generalization is missed.

Similarly for transitive verbs and other valence frames.

Abstraction from linear order: Dominance

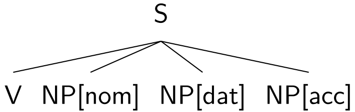
- Gazdar, Klein, Pullum & Sag (1985):
  Separation of immediate dominance = ID and linear precedence = LP.

Abstraction from linear order: Dominance

- Gazdar, Klein, Pullum & Sag (1985):
  Separation of immediate dominance = ID and linear precedence = LP.
- Dominance rules do not constrain the order of the daughters.

(126) S → V, NP[nom], NP[acc], NP[dat]

The only thing (126) says is that S dominates the other nodes:

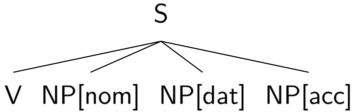


Abstraction from linear order: Dominance

- Gazdar, Klein, Pullum & Sag (1985):
  Separation of immediate dominance = ID and linear precedence = LP.
- Dominance rules do not constrain the order of the daughters.

  (126) S → V, NP[nom], NP[acc], NP[dat]

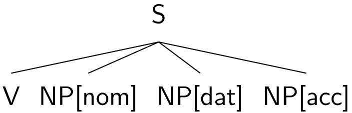
  The only thing (126) says is that S dominates the other nodes:



- Since there are no constraints on the order of the elments of the right-hand side, we need one rule rather than twelve:

Abstraction from linear order: Linear order

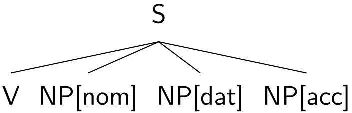
- LP rules hold for local trees, that is, trees of depth one:



→ We can say something about order of V, NP[nom], NP[dat] and NP[acc].

Abstraction from linear order: Linear order

- LP rules hold for local trees, that is, trees of depth one:



→ We can say something about order of V, NP[nom], NP[dat] and NP[acc].
An LP constraint holds for the whole grammar.
If we claim that NP[nom] precedes NP[acc],
this holds for rules for strictly transitive verbs as well as for rules for ditransitive verbs.

Getting more restrictive again

- Without restriction for the order → too much freedom

  S → V, NP[nom], NP[dat], NP[acc]

  The rule admits the following order:

  (127) * Dem Kind der Mann gibt ein Buch.
  the.DAT child the.NOM man gives the.ACC book

Getting more restrictive again

- Without restriction for the order → too much freedom

  S → V, NP[nom], NP[dat], NP[acc]

  The rule admits the following order:

  (127) * Dem Kind der Mann gibt ein Buch.
  the.DAT child the.NOM man gives the.ACC book

- Linearization rules rule out such orders.

  (128) V[+MC] < X
  X < V[−MC]

  MC stand for *main clause*.
  LP rule states: verb must be placed before all other constituents in main clauses (+MC) and after all other constituents in dependent clauses (−MC).

Passive pre-theoretically (I)

German passive theory-neutrally:

- The subject is suppressed.
- If there is an accusative object, this becomes the subject.

This holds for all verb classes forming a passive. Independent of the arity of the verb:

(129) a. weil er noch gearbeitet hat
because he.NOM still worked has
'because he has still worked'

b. weil noch gearbeitet wurde
because still worked was
'because there was still working there'

Passive pre-theoretically (I)

German passive theory-neutrally:

- The subject is suppressed.
- If there is an accusative object, this becomes the subject.

This holds for all verb classes forming a passive. Independent of the arity of the verb:

(129) a. weil er noch gearbeitet hat
because he.NOM still worked has
'because he has still worked'

b. weil noch gearbeitet wurde
because still worked was
'because there was still working there'

(130) a. weil er an Maria gedacht hat
because he.NOM on Maria thought has
'because he thought of Maria'

b. weil an Maria gedacht wurde
because on Maria thought was
'because Maria was thought of'

Passive pre-theoretically (II)

German passive theory-neutrally:

- The subject is suppressed.
- If there is an accusative object, this becomes the subject.

(131) a. weil Judit den Weltmeister geschlagen hat
because Judit.NOM the.ACC world.champion beaten has
'because Judit has beaten the world champion'

b. weil der Weltmeister geschlagen wurde
because the.NOM world.champion beaten was
'because the world champion was beaten'

Passive pre-theoretically (II)

German passive theory-neutrally:

- The subject is suppressed.
- If there is an accusative object, this becomes the subject.

(131) a. weil Judit den Weltmeister geschlagen hat
because Judit.NOM the.ACC world.champion beaten has
'because Judit has beaten the world champion'

b. weil der Weltmeister geschlagen wurde
because the.NOM world.champion beaten was
'because the world champion was beaten'

(132) a. weil er ihm den Aufsatz gegeben hat
because he.NOM him.DAT the.ACC essay given has
'because he has given him the essay'

b. weil ihm der Aufsatz gegeben wurde
because him.DAT the.NOM essay given was
'because he was given the essay'

Passive and phrase structure grammars

- One would have to write down two rules for every active/passive pair in PSG.

Passive and phrase structure grammars

- One would have to write down two rules for every active/passive pair in PSG.
- GPSG is a non-transformational theory.

Passive and phrase structure grammars

- One would have to write down two rules for every active/passive pair in PSG.
- GPSG is a non-transformational theory.
- Metarule derives passive rules from active rules.

Passive and phrase structure grammars

- One would have to write down two rules for every active/passive pair in PSG.
- GPSG is a non-transformational theory.
- Metarule derives passive rules from active rules.
- These are explained with respect to the subject introduction metarule.

Introduction of the subject via a metarule (I)

Our rules look like this:

(133)	V2 → H[7], N2[Case Dat]	(helfen 'help', vertrauen 'trust')
	V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben 'give', zeigen 'show')

Introduction of the subject via a metarule (I)

Our rules look like this:

(133)	V2 → H[7], N2[Case Dat]	(helfen 'help', vertrauen 'trust')
	V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben 'give', zeigen 'show')

The rules in (133) can be used to analyze VPs but not sentences with subject.

Introduction of the subject via a metarule (I)

Our rules look like this:

(133)	V2 → H[7], N2[Case Dat]	(helfen 'help', vertrauen 'trust')
	V2 → H[8], N2[Case Dat], N2[Case Acc]	(geben 'give', zeigen 'show')

The rules in (133) can be used to analyze VPs but not sentences with subject.

We use a metarule saying: "If there is a rule of the form 'V2 consists of something', then there is also a rule stating 'V3 consists of whatever V2 consists of + an NP in the nominative'".

Introduction of the subject via a metarule (I)

Our rules look like this:

(133) V2 → H[7], N2[Case Dat] (helfen 'help', vertrauen 'trust')
V2 → H[8], N2[Case Dat], N2[Case Acc] (geben 'give', zeigen 'show')

The rules in (133) can be used to analyze VPs but not sentences with subject.

We use a metarule saying: "If there is a rule of the form 'V2 consists of something', then there is also a rule stating 'V3 consists of whatever V2 consists of + an NP in the nominative'".

Formally:

(134) V2 → W ↦
V3 → W, N2[Case Nom]

W stands for an arbitrary number of categories (whatever).

Introduction of the subject via a metarule (II)

(135) V2 → W ↦
V3 → W, N2[Case Nom]

Introduction of the subject via a metarule (II)

(135) V2 → W ↦
V3 → W, N2[Case Nom]

This metarule takes the rules in (136) as input and produces the rules in (137):

(136) V2 → H[7], N2[Case Dat] (helfen 'help', vertrauen 'trust')
V2 → H[8], N2[Case Dat], N2[Case Acc] (geben 'give', zeigen 'show')

(137) V3 → H[7], N2[Case Dat], N2[Case Nom]
V3 → H[8], N2[Case Dat], N2[Case Acc], N2[Case Nom]

Introduction of the subject via a metarule (II)

(135) V2 → W ↦
V3 → W, N2[Case Nom]

This metarule takes the rules in (136) as input and produces the rules in (137):

(136) V2 → H[7], N2[Case Dat] (helfen 'help', vertrauen 'trust')
V2 → H[8], N2[Case Dat], N2[Case Acc] (geben 'give', zeigen 'show')

(137) V3 → H[7], N2[Case Dat], N2[Case Nom]
V3 → H[8], N2[Case Dat], N2[Case Acc], N2[Case Nom]

Subject and other arguments are on the same right-hand side of a rule and hence can be permuted, provided no LP rule is violated.

Passive as metarule

- For each active rule with subject and accusative object, a passive rule will be licensed with the subject suppressed. The relation between the rules is captured.

Passive as metarule

- For each active rule with subject and accusative object, a passive rule will be licensed with the subject suppressed. The relation between the rules is captured.
- Differences between Transformational Grammar/GB and GPSG:
  It is not the case that there are several trees that are related to each other, but rather active rules are related to passive rules.
  The active and passive rules can be used to derive two structures independently: (138b) is not derived from (138a).

(138) a. weil Judit den Weltmeister geschlagen hat
because Judit.NOM the.ACC world.champion beaten has
'because Judit has beaten the world champion'

b. weil der Weltmeister geschlagen wurde
because the.NOM world.champion beaten was
'because the world champion was beaten'

The generalization regarding active/passive alternations is captured nevertheless.

Passive in English

Gazdar, Klein, Pullum & Sag (1985) suggest the following metarule:

(139) VP → W, NP ↦
VP[PAS] → W, (PP[*by*])

This rule says that verbs selecting an object can be realized without this object in a passive VP. Optionally a *by* PP may appear in passive VPs.

(VP corresponds to V2)

Problems of the passive metarule operating on VP

1. Rule does not refer to the type of the verb (not all verbs have a passive).

Problems of the passive metarule operating on VP

1. Rule does not refer to the type of the verb (not all verbs have a passive).
2. Impersonal passive cannot be derived by suppressing an object.

(140) V2 → H[5] (*arbeiten* 'work')
 V2 → H[13], PP[*an*] (*denken* 'think')

So, if the analysis of the passive in English is not revised,
the analyses of the passive in English and German will differ.

Problems of the passive metarule operating on VP

1. Rule does not refer to the type of the verb (not all verbs have a passive).
2. Impersonal passive cannot be derived by suppressing an object.

   (140) V2 → H[5] (*arbeiten* 'work')
   V2 → H[13], PP[*an*] (*denken* 'think')

   So, if the analysis of the passive in English is not revised,
   the analyses of the passive in English and German will differ.
3. The German passive metarule could apply to rules including the subject.

Long-distance dependencies as the result of local dependencies

- Until now: verb-initial and verb-final placement of the verb:

(141) a. [dass] der Mann dem Kind das Buch gibt
that the.NOM man the.DAT child the.ACC book gives

b. Gibt der Mann dem Kind das Buch?
gives the.NOM man the.DAT child the.ACC book

Long-distance dependencies as the result of local dependencies

- Until now: verb-initial and verb-final placement of the verb:

(141) a. [dass] der Mann dem Kind das Buch gibt
that the.NOM man the.DAT child the.ACC book gives

b. Gibt der Mann dem Kind das Buch?
gives the.NOM man the.DAT child the.ACC book

- What about verb second placement:

(142) a. Der Mann gibt dem Kind das Buch.
the.NOM man gives the.DAT child the.ACC book

b. Dem Kind gibt der Mann das Buch.
the.DAT child gives the.NOM man the.ACC book

Long-distance dependencies as the result of local dependencies

- Until now: verb-initial and verb-final placement of the verb:

(141) a. [dass] der Mann dem Kind das Buch gibt
that the.NOM man the.DAT child the.ACC book gives

b. Gibt der Mann dem Kind das Buch?
gives the.NOM man the.DAT child the.ACC book

- What about verb second placement:

(142) a. Der Mann gibt dem Kind das Buch.
the.NOM man gives the.DAT child the.ACC book

b. Dem Kind gibt der Mann das Buch.
the.DAT child gives the.NOM man the.ACC book

- V2 is analyzed as a nonlocal dependency via a sequence of local dependencies.
One of the main innovations of GPSG:
transformationless analysis of nonlocal dependencies (but also Harman 1963).

Metarules for the introduction of nonlocal dependencies

We take an arbitrary category X out of the set of categories on the right-hand side of the rule and represent it on the left-hand side after a slash ('/'):

(143) $\mathrm{V3} \rightarrow \mathrm{W}, \mathrm{X} \mapsto$
$\mathrm{V3/X} \rightarrow \mathrm{W}$

Metarules for the introduction of nonlocal dependencies

We take an arbitrary category X out of the set of categories on the right-hand side of the rule and represent it on the left-hand side after a slash ('/'):

(143) V3 → W, X ↦
V3/X → W

Given the input in (144), the rule creates the rules in (145):

(144) V3 → H[8], N2[Case Dat], N2[Case Acc], N2[Case Nom]

(145) V3/N2[Case Nom] → H[8], N2[Case Dat], N2[Case Acc]
V3/N2[Case Dat] → H[8], N2[Case Acc], N2[Case Nom]
V3/N2[Case Acc] → H[8], N2[Case Dat], N2[Case Nom]

Rule for binding off nonlocal dependencies

(146) V3[+Fin] → X[+Top], V3[+MC]/X

X stands for arbitrary category marked as missing in V3 by '/'.

Rule for binding off nonlocal dependencies

(146) V3[+Fin] → X[+Top], V3[+MC]/X

X stands for arbitrary category marked as missing in V3 by '/'.

Example instantiations of the rule are given in (147):

(147) V3[+Fin] → N2[+Top, Case Nom], V3[+MC]/N2[Case Nom]
V3[+Fin] → N2[+Top, Case Dat], V3[+MC]/N2[Case Dat]
V3[+Fin] → N2[+Top, Case Acc], V3[+MC]/N2[Case Acc]

Rule for binding off nonlocal dependencies

(146) V3[+Fin] → X[+Top], V3[+MC]/X

X stands for arbitrary category marked as missing in V3 by '/'.

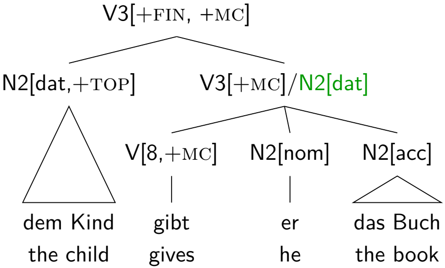
Example instantiations of the rule are given in (147):

(147) V3[+Fin] → N2[+Top, Case Nom], V3[+MC]/N2[Case Nom]
V3[+Fin] → N2[+Top, Case Dat], V3[+MC]/N2[Case Dat]
V3[+Fin] → N2[+Top, Case Acc], V3[+MC]/N2[Case Acc]

LP rule: X in (146) is serialized left of anything else (e.g. V3), since it is [+Top].

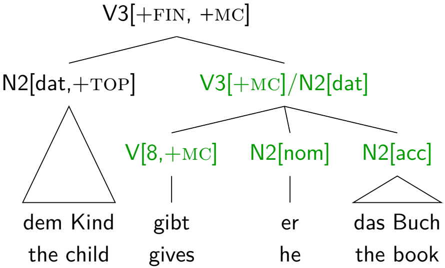
(148) [+Top] < X

An example analysis



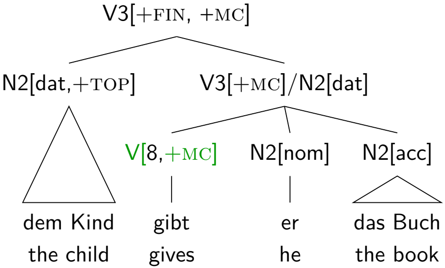
- Metarule licenses rule introducing dative object into SLASH.

An example analysis



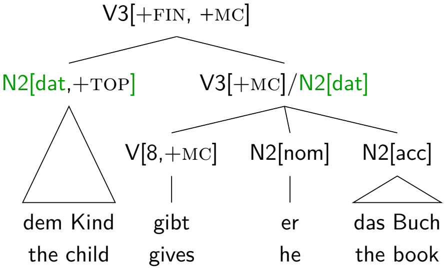
- Metarule licenses rule introducing dative object into SLASH.
- This rule is applied and licenses the subtree for *gibt er das Buch*.

An example analysis



- Metarule licenses rule introducing dative object into SLASH.
- This rule is applied and licenses the subtree for *gibt er das Buch*.
- The linearization rule orders the verb left of other constituents (V[+MC] < X).

An example analysis



- Metarule licenses rule introducing dative object into SLASH.
- This rule is applied and licenses the subtree for *gibt er das Buch*.
- The linearization rule orders the verb left of other constituents (V[+MC] < X).
- The constituent following the slash is bound off in the last step.

An example with nonlocal dependencies (I)

All NPs in (149) depend on the same verb:

(149) Dem Kind gibt er das Buch.
the.DAT child gives he.NOM the.ACC book
'He gives the child the book.'

Complicated system of linearization rules → analyze (149) with a flat structure.

[6] Scherpenisse 1986: 84.

An example with nonlocal dependencies (I)

All NPs in (149) depend on the same verb:

(149) Dem Kind gibt er das Buch.
the.DAT child gives he.NOM the.ACC book
'He gives the child the book.'

Complicated system of linearization rules → analyze (149) with a flat structure.

But this would not work for:

(150) Wen$_i$ glaubst du, daß ich _$_i$ gesehen habe?[6]
who believe you that I seen have
'Who do you think I saw?'

(150) cannot be explained by local reordering since *wen* does not depend on *glaubst* but on *gesehen* and *gesehen* is located in a different local subtree.

[6] Scherpenisse 1986: 84.

An example with nonlocal dependencies (II)

- (151) is analyzed in several steps: introduction, percolation and finally binding off of information about the long-distance dependency

(151) Wen glaubst du, daß ich gesehen habe?
who believe you that I seen have

An example with nonlocal dependencies (II)

- (151) is analyzed in several steps: introduction, percolation and finally binding off of information about the long-distance dependency

(151) Wen glaubst du, daß ich gesehen habe?
who believe you that I seen have

- *ich gesehen habe* is V3/NP[acc]
(grammar rule licensed by a metarule)

An example with nonlocal dependencies (II)

- (151) is analyzed in several steps: introduction, percolation and finally binding off of information about the long-distance dependency

(151) Wen glaubst du, daß ich gesehen habe?
who believe you that I seen have

- *ich gesehen habe* is V3/NP[acc]
(grammar rule licensed by a metarule)
- *dass ich gesehen habe* is V3/NP[acc]
(percolation of SLASH information)

An example with nonlocal dependencies (II)

- (151) is analyzed in several steps: introduction, percolation and finally binding off of information about the long-distance dependency

(151)	Wen	glaubst	du,	daß	ich	gesehen	habe?
	who	believe	you	that	I	seen	have

- *ich gesehen habe* is V3/NP[acc]
  (grammar rule licensed by a metarule)
- *dass ich gesehen habe* is V3/NP[acc]
  (percolation of SLASH information)
- *glaubst du, dass ich gesehen habe* is V3/NP[acc]
  (percolation of SLASH information)

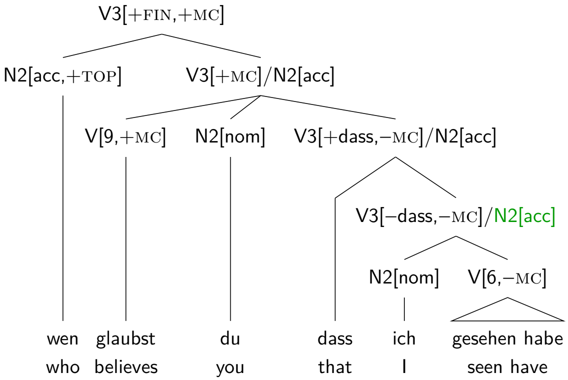
An example with nonlocal dependencies (II)

- (151) is analyzed in several steps: introduction, percolation and finally binding off of information about the long-distance dependency

(151) Wen glaubst du, daß ich gesehen habe?
who believe you that I seen have

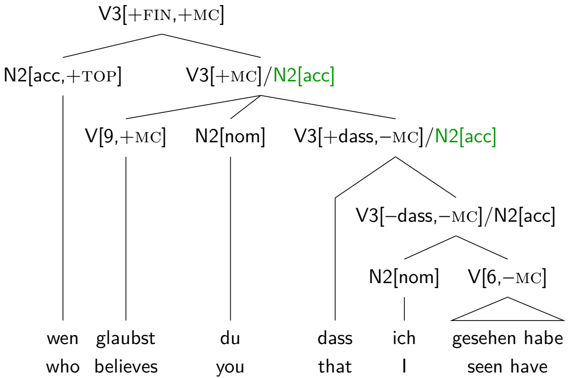
- *ich gesehen habe* is V3/NP[acc]
(grammar rule licensed by a metarule)
- *dass ich gesehen habe* is V3/NP[acc]
(percolation of SLASH information)
- *glaubst du, dass ich gesehen habe* is V3/NP[acc]
(percolation of SLASH information)
- *Wen glaubst du, dass ich gesehen habe* is V3
(binding off of SLASH information in grammar rule)

An example with nonlocal dependencies (III)



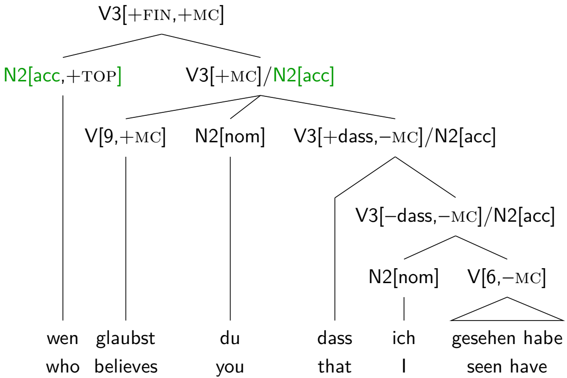
Simplifying assumption: *gesehen habe* behaves like a simplex transitive verb.

An example with nonlocal dependencies (III)



Simplifying assumption: *gesehen habe* behaves like a simplex transitive verb.

An example with nonlocal dependencies (III)



Simplifying assumption: *gesehen habe* behaves like a simplex transitive verb.

Highlights: Across the Board Extraction

- Gazdar's (1981) SLASH-based analysis can account for so-called Across the Board extraction (Ross 1967):

(152) a. The kennel which Mary made and Fido sleeps in has been stolen. (= S/NP & S/NP)

b. The kennel in which Mary keeps drugs and Fido sleeps has been stolen. (= S/PP & S/PP)

c. * The kennel (in) which Mary made and Fido sleeps has been stolen. (= S/NP & S/PP)

Conjuncts have to have the same element in SLASH and this information is percolated further and then bound off.

Highlights: Across the Board Extraction

- Gazdar's (1981) SLASH-based analysis can account for so-called Across the Board extraction (Ross 1967):

(152) a. The kennel which Mary made and Fido sleeps in has been stolen. (= S/NP & S/NP)

b. The kennel in which Mary keeps drugs and Fido sleeps has been stolen. (= S/PP & S/PP)

c. * The kennel (in) which Mary made and Fido sleeps has been stolen. (= S/NP & S/PP)

Conjuncts have to have the same element in SLASH and this information is percolated further and then bound off.

- Such sentences are a miracle for transformational analyses:
Why must two transformations move something of the same category?
How can two different things land in the same position?

Problems

- representation of valence and morphology
- partial fronting
- generative capacity

Representation of valence and morphology

- Morphology has to access valence information:

(153)	a.	lös-bar solv-able	(nominative, accusative)
	b.	vergleich-bar compar-able	(nominative, accusative, PP[mit])
	c.	* schlaf-bar sleep-able	(nominative)
	d.	* helf-bar help-able	(nominative, dative)

- Generalization: *bar* adjectives can be formed from verbs governing an accusative.

Representation of valence and morphology

- Morphology has to access valence information:

(153)	a.		lös-bar solv-able	(nominative, accusative)
	b.		vergleich-bar compar-able	(nominative, accusative, PP[mit])
	c.	*	schlaf-bar sleep-able	(nominative)
	d.	*	helf-bar help-able	(nominative, dative)

- Generalization: *bar* adjectives can be formed from verbs governing an accusative.
- This information is inaccessable in GPSG. Only valence numbers and this number does not even tell us whether there is an accusative. There may be a bunch of different rules (active/passive) with or without the accusative.

Representation of valence and morphology

- Morphology has to access valence information:

(153)	a.		lös-bar solv-able	(nominative, accusative)
	b.		vergleich-bar compar-able	(nominative, accusative, PP[mit])
	c.	*	schlaf-bar sleep-able	(nominative)
	d.	*	helf-bar help-able	(nominative, dative)

- Generalization: *bar* adjectives can be formed from verbs governing an accusative.
- This information is inaccessable in GPSG. Only valence numbers and this number does not even tell us whether there is an accusative. There may be a bunch of different rules (active/passive) with or without the accusative.
- Valence must contain detailed descriptions of arguments (CG, LFG, HPSG).

Partial fronting

German allows the fronting of (partial) VPs:

(154) a.

[Erzählen]	wird	er	seiner	Tochter	ein	Märchen	können.
tell	will	he.NOM	his.DAT	daughter	a.ACC	fairy.tale	can

‘He will be able to tell his daughter a fairy tale.’

b.

[Ein	Märchen	erzählen]	wird	er	seiner	Tochter	können.
a.ACC	fairy.tale	tell	will	he.NOM	his.ACC	daughter	can

c.

[Seiner	Tochter	ein	Märchen	erzählen]	wird	er	können.
his.DAT	daughter	a.ACC	fairy.tale	tell	will	he.NOM	can

Arguments not realized in the fronted VP have to be realized in the Mittelfeld.

Partial fronting (II)

- Arguments missing in initial position have to be realized in the Mittelfeld.
  The case in the Mittelfeld has to match the requirement of the verb in the Vorfeld:

(155) a. Verschlungen hat er es nicht.
devoured has he.NOM it.ACC not
'He did not devour it.'

b. * Verschlungen hat er nicht.
devoured has he.NOM not

c. * Verschlungen hat er ihm nicht.
devoured has he.NOM him.DAT not

Partial fronting (II)

- Arguments missing in initial position have to be realized in the Mittelfeld.
  The case in the Mittelfeld has to match the requirement of the verb in the Vorfeld:

(155) a. Verschlungen hat er es nicht.
devoured has he.NOM it.ACC not
'He did not devour it.'

b. * Verschlungen hat er nicht.
devoured has he.NOM not

c. * Verschlungen hat er ihm nicht.
devoured has he.NOM him.DAT not

- But this is impossible to do with the standard treatment of valence in GPSG.

Partial fronting (II)

- Arguments missing in initial position have to be realized in the Mittelfeld.
  The case in the Mittelfeld has to match the requirement of the verb in the Vorfeld:

(155) a. Verschlungen hat er es nicht.
devoured has he.NOM it.ACC not
'He did not devour it.'

b. * Verschlungen hat er nicht.
devoured has he.NOM not

c. * Verschlungen hat er ihm nicht.
devoured has he.NOM him.DAT not

- But this is impossible to do with the standard treatment of valence in GPSG.
- Combinations of verbs with arguments are licensed by PSG rules referring to numbers.
- But the objects can only be missing when they are realized in the Mittelfeld.
  How is this connection established?

Partial fronting (II)

- Arguments missing in initial position have to be realized in the Mittelfeld.
  The case in the Mittelfeld has to match the requirement of the verb in the Vorfeld:

(155) a. Verschlungen hat er es nicht.
devoured has he.NOM it.ACC not
'He did not devour it.'

b. * Verschlungen hat er nicht.
devoured has he.NOM not

c. * Verschlungen hat er ihm nicht.
devoured has he.NOM him.DAT not

- But this is impossible to do with the standard treatment of valence in GPSG.
- Combinations of verbs with arguments are licensed by PSG rules referring to numbers.
- But the objects can only be missing when they are realized in the Mittelfeld.
  How is this connection established?
- Nerbonne (1986) and Johnson (1986): different representation of valence.
  One similar to Categorial Grammar.

Generative capacity

- The generative capacity of GPSG corresponds to those of context free grammars.

Generative capacity

- The generative capacity of GPSG corresponds to those of context free grammars.
- Being restrictive was one of the goals of GPSG.

Generative capacity

- The generative capacity of GPSG corresponds to those of context free grammars.
- Being restrictive was one of the goals of GPSG.
- But Shieber (1985) and Culy (1985):
  there are languages that cannot be described with context free grammars.
  (see also Pullum 1986 for historical remarks)

Generative capacity

- The generative capacity of GPSG corresponds to those of context free grammars.
- Being restrictive was one of the goals of GPSG.
- But Shieber (1985) and Culy (1985):
  there are languages that cannot be described with context free grammars.
  (see also Pullum 1986 for historical remarks)
- This means that GPSG is not powerful enough to describe all languages.

Generative capacity

- The generative capacity of GPSG corresponds to those of context free grammars.
- Being restrictive was one of the goals of GPSG.
- But Shieber (1985) and Culy (1985):
  there are languages that cannot be described with context free grammars.
  (see also Pullum 1986 for historical remarks)
- This means that GPSG is not powerful enough to describe all languages.

- All mentioned problems are fixed in HPSG, the successor of GPSG.

$$\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle \textit{Grammatik} \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle \text{DET} \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle \text{语法} \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle \text{DET} \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle \text{دستور} \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle \text{DET} \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle \text{व्याकरण} \rangle \\ \text{SYN}|\text{CAT}|\text{SUBCAT} & \langle \text{DET} \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Feature descriptions, feature structures and models

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\text{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoining Grammar (TAG)

Reading material

Müller 2020b: Chapter 6

Feature descriptions and feature structures

Feature structures are used to model linguistic objects:

- attribut value structure
- feature structure

Linguistis use feature descriptions to talk about feature structures:

- attribute-value matrix (AVM)
- feature matrix

Feature descriptions and feature structures

Feature structures are used to model linguistic objects:

- attribut value structure
- feature structure

Linguistis use feature descriptions to talk about feature structures:

- attribute-value matrix (AVM)
- feature matrix

- Shieber (1986), Pollard & Sag (1987), Johnson (1988), Carpenter (1992), King (1994), Richter (2004; 2021)

An example

A feature description, describing a human being:

$$\begin{bmatrix} \textsc{firstname} & \textit{max} \\ \textsc{lastname} & \textit{meier} \\ \textsc{date-of-birth} & \textit{10.10.1985} \end{bmatrix}$$

An example

A feature description, describing a human being:

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \end{bmatrix}$$

Recursive descriptions:

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \begin{bmatrix} \text{FIRSTNAME} & \textit{peter} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.05.1960} \\ \text{FATHER} & \ldots \\ \text{MOTHER} & \ldots \end{bmatrix} \\ \text{MOTHER} & \ldots \end{bmatrix}$$

An example

A feature description, describing a human being:

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \end{bmatrix}$$

Recursive descriptions:

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \begin{bmatrix} \text{FIRSTNAME} & \textit{peter} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.05.1960} \\ \text{FATHER} & \ldots \\ \text{MOTHER} & \ldots \end{bmatrix} \\ \text{MOTHER} & \ldots \end{bmatrix}$$

Exercise: How can we represent daughters or sons of a human being?

Solution I: Features

$$\begin{bmatrix} \textsc{firstname} & \textit{max} \\ \textsc{lastname} & \textit{meier} \\ \textsc{date-of-birth} & \textit{10.10.1985} \\ \textsc{father} & \textit{...} \\ \textsc{mother} & \textit{...} \\ \textsc{daughter} & \textit{...} \end{bmatrix}$$

Solution I: Features

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTER} & \textit{...} \end{bmatrix}$$

What if we have several daughters?

Solution I: Features

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTER} & \textit{...} \end{bmatrix}$$

What if we have several daughters?

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTER-1} & \textit{...} \\ \text{DAUGHTER-2} & \textit{...} \\ \text{DAUGHTER-3} & \textit{...} \end{bmatrix}$$

Solution I: Features, a lot of features

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTER} & \textit{...} \end{bmatrix}$$

What if we have several daughters?

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTER-1} & \textit{...} \\ \text{DAUGHTER-2} & \textit{...} \\ \text{DAUGHTER-3} & \textit{...} \end{bmatrix}$$

How many features do we want to assume? Where is the limit?
What is the value of DAUGHTER-32?

Solution II: Lists

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \ldots \\ \text{MOTHER} & \ldots \\ \text{DAUGHTERS} & \left\langle \ldots,\ \ldots \right\rangle \end{bmatrix}$$

Solution II: Lists

$$\begin{bmatrix} \textsc{firstname} & \textit{max} \\ \textsc{lastname} & \textit{meier} \\ \textsc{date-of-birth} & \textit{10.10.1985} \\ \textsc{father} & ... \\ \textsc{mother} & ... \\ \textsc{daughters} & \left\langle ..., ... \right\rangle \end{bmatrix}$$

What about sons?

Solution II: Lists

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{DAUGHTERS} & \langle \textit{...}, \textit{...} \rangle \end{bmatrix}$$

What about sons?

Do we want to make this difference?
Yes, but the property is a property of the described objects:

$$\begin{bmatrix} \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{GENDER} & \textit{male} \\ \text{FATHER} & \textit{...} \\ \text{MOTHER} & \textit{...} \\ \text{CHILDREN} & \langle \textit{...}, \textit{...} \rangle \end{bmatrix}$$

Types

- Feature structures are of a certain type.
- The type is written in *italics*:

$$\begin{bmatrix} \textit{type} & \\ \mathrm{A1} & \textit{V1} \end{bmatrix}$$

Types

- Feature structures are of a certain type.
- The type is written in *italics*:

$$\begin{bmatrix} \textit{type} & \\ \mathrm{A1} & \textit{V1} \end{bmatrix}$$

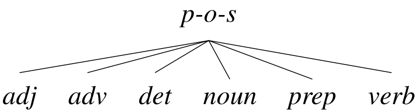
- Types specify which features have to belong to a certain feature structure.

Types

- Feature structures are of a certain type.
- The type is written in *italics*:

$$\begin{bmatrix} \textit{type} & \\ \text{A1} & \textit{V1} \end{bmatrix}$$

- Types specify which features have to belong to a certain feature structure.
- Types are organized in hierarchies.
  Example: part of speech



Feature descriptionen of type *person*

- Our example description describes objects of type *person*.

$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{firstname} \\ \text{LASTNAME} & \textit{lastname} \\ \text{DATE-OF-BIRTH} & \textit{date} \\ \text{GENDER} & \textit{gender} \\ \text{FATHER} & \textit{person} \\ \text{MOTHER} & \textit{person} \\ \text{CHILDREN} & \textit{list of person} \end{bmatrix}$$

Feature descriptionen of type *person*

- Our example description describes objects of type *person*.

person	
FIRSTNAME	*firstname*
LASTNAME	*lastname*
DATE-OF-BIRTH	*date*
GENDER	*gender*
FATHER	*person*
MOTHER	*person*
CHILDREN	*list of person*

- Properties like OPERATING VOLTAGE are irrelevant for such objects!

Feature descriptionen of type *person*

- Our example description describes objects of type *person*.

$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{firstname} \\ \text{LASTNAME} & \textit{lastname} \\ \text{DATE-OF-BIRTH} & \textit{date} \\ \text{GENDER} & \textit{gender} \\ \text{FATHER} & \textit{person} \\ \text{MOTHER} & \textit{person} \\ \text{CHILDREN} & \textit{list of person} \end{bmatrix}$$

- Properties like OPERATING VOLTAGE are irrelevant for such objects!
- Type specifies which features are relevant for such an object.

Feature descriptionen of type *person*

- Our example description describes objects of type *person*.

$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{firstname} \\ \text{LASTNAME} & \textit{lastname} \\ \text{DATE-OF-BIRTH} & \textit{date} \\ \text{GENDER} & \textit{gender} \\ \text{FATHER} & \textit{person} \\ \text{MOTHER} & \textit{person} \\ \text{CHILDREN} & \textit{list of person} \end{bmatrix}$$

- Properties like OPERATING VOLTAGE are irrelevant for such objects!
- Type specifies which features are relevant for such an object.
- We know: every human has a birthday even if we don't know the exact value.

Our example with children: One or two?

- Do we describe one or two children of Peter and Anna?

$$
\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textsc{firstname} & \textit{max} \\
\textsc{lastname} & \textit{meier} \\
\textsc{date-of-birth} & \textit{10.10.1985} \\
\textsc{father} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textsc{firstname} & \textit{peter} \\
\textsc{lastname} & \textit{meier} \\
\textsc{children} & \left\langle \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{person}} \\ \textsc{firstname} & \textit{klaus} \end{array}\right], \ldots \right\rangle
\end{array}\right] \\
\textsc{mother} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textsc{firstname} & \textit{anna} \\
\textsc{lastname} & \textit{meier} \\
\textsc{children} & \left\langle \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{person}} \\ \textsc{firstname} & \textit{klaus} \end{array}\right], \ldots \right\rangle
\end{array}\right]
\end{array}\right]
$$

Our example with children: One or two?

- Do we describe one or two children of Peter and Anna?

$$\left[\begin{array}{ll}\textit{person} & \\ \textsc{firstname} & \textit{max} \\ \textsc{lastname} & \textit{meier} \\ \textsc{date-of-birth} & \textit{10.10.1985} \\ \textsc{father} & \left[\begin{array}{ll}\textit{person} & \\ \textsc{firstname} & \textit{peter} \\ \textsc{lastname} & \textit{meier} \\ \textsc{children} & \left\langle \left[\begin{array}{ll}\textit{person} & \\ \textsc{firstname} & \textit{klaus}\end{array}\right], \dots \right\rangle\end{array}\right] \\ \textsc{mother} & \left[\begin{array}{ll}\textit{person} & \\ \textsc{firstname} & \textit{anna} \\ \textsc{lastname} & \textit{meier} \\ \textsc{children} & \left\langle \left[\begin{array}{ll}\textit{person} & \\ \textsc{firstname} & \textit{klaus}\end{array}\right], \dots \right\rangle\end{array}\right]\end{array}\right]$$

- We don't know!

Our example with children: One or two?

- Do we describe one or two children of Peter and Anna?

$$\begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{max} \\ \textsc{lastname} & \textit{meier} \\ \textsc{date-of-birth} & \textit{10.10.1985} \\ \textsc{father} & \begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{peter} \\ \textsc{lastname} & \textit{meier} \\ \textsc{children} & \left\langle \begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{klaus} \end{bmatrix}, \ldots \right\rangle \end{bmatrix} \\ \textsc{mother} & \begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{anna} \\ \textsc{lastname} & \textit{meier} \\ \textsc{children} & \left\langle \begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{klaus} \end{bmatrix}, \ldots \right\rangle \end{bmatrix} \end{bmatrix}$$

- We don't know!
- There may be two different children from previous partnerships named *Klaus*.

Our example with children: Structure sharing

- Do we describe one or two children of Peter and Anna?

$$
\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textrm{FIRSTNAME} & \textit{max} \\
\textrm{LASTNAME} & \textit{meier} \\
\textrm{DATE-OF-BIRTH} & \textit{10.10.1985} \\
\textrm{FATHER} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textrm{FIRSTNAME} & \textit{peter} \\
\textrm{LASTNAME} & \textit{meier} \\
\textrm{CHILDREN} & \left\langle \boxed{1} \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textrm{FIRSTNAME} & \textit{klaus}
\end{array}\right], \ldots \right\rangle
\end{array}\right] \\
\textrm{MOTHER} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{person}} \\
\textrm{FIRSTNAME} & \textit{anna} \\
\textrm{LASTNAME} & \textit{meier} \\
\textrm{CHILDREN} & \left\langle \boxed{1}, \ldots \right\rangle
\end{array}\right]
\end{array}\right]
$$

Our example with children: Structure sharing

- Do we describe one or two children of Peter and Anna?

$$
\begin{bmatrix}
\textit{person} & \\
\text{FIRSTNAME} & \textit{max} \\
\text{LASTNAME} & \textit{meier} \\
\text{DATE-OF-BIRTH} & \textit{10.10.1985} \\
\text{FATHER} & \begin{bmatrix}
\textit{person} & \\
\text{FIRSTNAME} & \textit{peter} \\
\text{LASTNAME} & \textit{meier} \\
\text{CHILDREN} & \left\langle \boxed{1} \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{klaus} \end{bmatrix}, \ldots \right\rangle
\end{bmatrix} \\
\text{MOTHER} & \begin{bmatrix}
\textit{person} & \\
\text{FIRSTNAME} & \textit{anna} \\
\text{LASTNAME} & \textit{meier} \\
\text{CHILDREN} & \left\langle \boxed{1}, \ldots \right\rangle
\end{bmatrix}
\end{bmatrix}
$$

- Klaus is a single child that belongs to both parents.

Our example with children: Structure sharing

- Do we describe one or two children of Peter and Anna?

$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{peter} \\ \text{LASTNAME} & \textit{meier} \\ \text{CHILDREN} & \left\langle \boxed{1} \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{klaus} \end{bmatrix}, \ldots \right\rangle \end{bmatrix} \\ \text{MOTHER} & \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{anna} \\ \text{LASTNAME} & \textit{meier} \\ \text{CHILDREN} & \left\langle \boxed{1}, \ldots \right\rangle \end{bmatrix} \end{bmatrix}$$

- Klaus is a single child that belongs to both parents.
- What about Max?

Our example with children: Cyclic descriptions

- $\boxed{2}$ is placed in front of the description and occurs within it.

$$\boxed{2}\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{max} \\ \text{LASTNAME} & \textit{meier} \\ \text{DATE-OF-BIRTH} & \textit{10.10.1985} \\ \text{FATHER} & \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{peter} \\ \text{LASTNAME} & \textit{meier} \\ \text{CHILDREN} & \left\langle \boxed{1}\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{klaus} \end{bmatrix}, \boxed{2} \right\rangle \end{bmatrix} \\ \text{MOTHER} & \begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{anna} \\ \text{LASTNAME} & \textit{meier} \\ \text{CHILDREN} & \left\langle \boxed{1}, \boxed{2} \right\rangle \end{bmatrix} \end{bmatrix}$$

Unification

- Grammatical rules & lexical items are described by feature descriptions.

Unification

- Grammatical rules & lexical items are described by feature descriptions.
- Grammatical rules contain partial descriptions of daughters,
  but not the complete information.

Unification

- Grammatical rules & lexical items are described by feature descriptions.
- Grammatical rules contain partial descriptions of daughters,
  but not the complete information.
- A specific phrase has to be compatible with the demands regarding the daughter to be able to enter the structure.

Unification

- Grammatical rules & lexical items are described by feature descriptions.
- Grammatical rules contain partial descriptions of daughters, but not the complete information.
- A specific phrase has to be compatible with the demands regarding the daughter to be able to enter the structure.
- Term for this specific kind of compatibility: unifyability

Unification

- Grammatical rules & lexical items are described by feature descriptions.
- Grammatical rules contain partial descriptions of daughters,
but not the complete information.
- A specific phrase has to be compatible with the demands regarding the daughter to be able to enter the structure.
- Term for this specific kind of compatibility: unifyability
- When two structures are unified, the result is a new structure containing all information of the two unified structures and nothing more.

Example: Detective agency

- We are searching for a blond, female person named Meier.

Example: Detective agency

- We are searching for a blond, female person named Meier.
- A possible description:

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

Example: Detective agency

- We are searching for a blond, female person named Meier.
- A possible description:

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

- If we get a search result matching the following description, we change the agency.

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{male} \\ \textsc{haircolor} & \textit{red} \end{bmatrix}$$

Example: Detective agency

- We are searching for a blond, female person named Meier.

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

a possible result:

$$\begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{katharina} \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{date-of-birth} & \textit{15.10.1965} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

Example: Detective agency

- We are searching for a blond, female person named Meier.

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

a possible result:

$$\begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{katharina} \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{date-of-birth} & \textit{15.10.1965} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

- Katharina Meier may have further properties unknown to the detective.
  Important: those he does know have to be compatible to the request.

Example: Detective agency

The unification of the request

$$\begin{bmatrix} \textit{person} & \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

with the information of the detective

$$\begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{katharina} \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{date-of-birth} & \textit{15.10.1965} \\ \textsc{haircolor} & \textit{blonde} \end{bmatrix}$$

is

$$\begin{bmatrix} \textit{person} & \\ \textsc{firstname} & \textit{katharina} \\ \textsc{lastname} & \textit{meier} \\ \textsc{gender} & \textit{female} \\ \textsc{date-of-birth} & \textit{15.10.1965} \\ \textsc{haircolor} & \textit{blond} \end{bmatrix}$$

Example: Detective agency

The unification of the request

$$\begin{bmatrix} \textit{person} & \\ \text{LASTNAME} & \textit{meier} \\ \text{GENDER} & \textit{female} \\ \text{HAIRCOLOR} & \textit{blonde} \end{bmatrix}$$

with the information of the detective

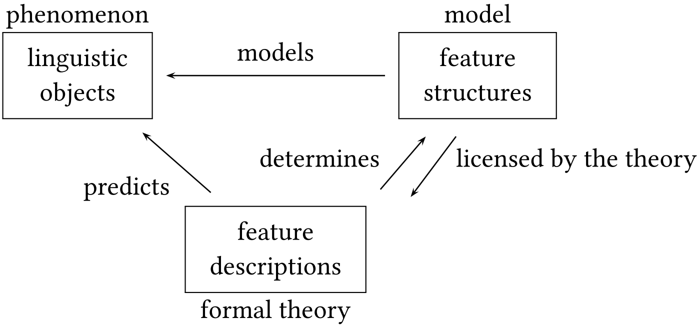
$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{katharina} \\ \text{LASTNAME} & \textit{meier} \\ \text{GENDER} & \textit{female} \\ \text{DATE-OF-BIRTH} & \textit{15.10.1965} \\ \text{HAIRCOLOR} & \textit{blonde} \end{bmatrix}$$

is but not:

$$\begin{bmatrix} \textit{person} & \\ \text{FIRSTNAME} & \textit{katharina} \\ \text{LASTNAME} & \textit{meier} \\ \text{GENDER} & \textit{female} \\ \text{DATE-OF-BIRTH} & \textit{15.10.1965} \\ \text{HAIRCOLOR} & \textit{blond} \\ \text{CHILDREN} & \langle\rangle \end{bmatrix}$$

The detective may not invent properties!
He risks his job
by providing possibly wrong information!

Phenomena, models and formal theories



Homework

1. Think about how one could describe musical instruments using feature descriptions.
2. Come up with a type hierarchy for the word classes (*det*, *comp*, *noun*, *verb*, *adj*, *prep*). Think about the ways in which one can organize the type hierachy so that one can express the generalizations that where captured by the binary features in on slide 116.
3. I motivated the introduction of lists. This may look like an extension of the formalism, but it is not as it is possible to convert the list notation into a notation which only requires feature-value pairs. Think about how one could do this.
4. (Additional exercise) The relation *append* will play a role in the introduction of HPSG. This relation serves to combine two lists to form a third. Relational constraints such as *append* do in fact constitute an expansion of the formalism. Using relational constraints, it is possible to relate any number of feature values to other values, that is, one can write programs which compute a particular value depending on other values. This poses the question as to whether one needs such powerful descriptive tools in a linguistic theory and if we do allow them, what kind of complexity we afford them. A theory which can do without relational constraints should be preferred over one that uses relational constraints (see Müller 2013b: Chapter 20 for a comparison of theories).
   For the concatenation of lists, there is a possible implementation in feature structures without recourse to relational constraints. Find out how this can be done. Give your sources and document how you went about finding the solution.

$$\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \textit{Grammatik}\ \rangle \\ \text{SYN|CAT|SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{语法}\ \rangle \\ \text{SYN|CAT|SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{دستور}\ \rangle \\ \text{SYN|CAT|SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} & \\ \text{ORTH} & \langle\ \text{व्याकरण}\ \rangle \\ \text{SYN|CAT|SUBCAT} & \langle\ \text{DET}\ \rangle \\ \text{SEM} & \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST}\ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Lexical Functional Grammar (LFG)

Stefan Müller
Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\text{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoining Grammar (TAG)

Reading material

Müller 2020b: Chapter 7 (without 7.1.5 on semantics)

Lexical Functional Grammar (LFG)

- Developed by Joan Bresnan and Ron Kaplan in the 1980s.

Lexical Functional Grammar (LFG)

- Developed by Joan Bresnan and Ron Kaplan in the 1980s.
- LFG is part of so-called West-Coast-Linguistics:
  Joan Bresnan (LFG) and Ivan Sag (HPSG) did their PhD with Chomsky
  (MIT is situated at the East Coast of the US,
  while Stanford, Palo Alto and Berkeley are in the Bay Area in California)

Lexical Functional Grammar (LFG)

- Developed by Joan Bresnan and Ron Kaplan in the 1980s.
- LFG is part of so-called West-Coast-Linguistics:
  Joan Bresnan (LFG) and Ivan Sag (HPSG) did their PhD with Chomsky
  (MIT is situated at the East Coast of the US,
  while Stanford, Palo Alto and Berkeley are in the Bay Area in California)
- LFG aims for psycholinguistical plausibility and wants to be implementable

Lexical Functional Grammar (LFG)

- Developed by Joan Bresnan and Ron Kaplan in the 1980s.
- LFG is part of so-called West-Coast-Linguistics:
  Joan Bresnan (LFG) and Ivan Sag (HPSG) did their PhD with Chomsky
  (MIT is situated at the East Coast of the US,
  while Stanford, Palo Alto and Berkeley are in the Bay Area in California)
- LFG aims for psycholinguistical plausibility and wants to be implementable
- teaching material and overview articles: Bresnan et al. 2016; Dalrymple 2006

Lexical Functional Grammar (LFG)

- Developed by Joan Bresnan and Ron Kaplan in the 1980s.
- LFG is part of so-called West-Coast-Linguistics:
  Joan Bresnan (LFG) and Ivan Sag (HPSG) did their PhD with Chomsky
  (MIT is situated at the East Coast of the US,
  while Stanford, Palo Alto and Berkeley are in the Bay Area in California)
- LFG aims for psycholinguistical plausibility and wants to be implementable
- teaching material and overview articles: Bresnan et al. 2016; Dalrymple 2006
- In-depth works on German: Berman 1996; 2003 and Cook 2001

General remarks on the representational format

- multiple levels of representation:
  - c-structure (constituent structures, licensed by PSG, $\overline{\text{X}}$ structures)

General remarks on the representational format

- multiple levels of representation:
  - c-structure (constituent structures, licensed by PSG, $\overline{\text{X}}$ structures)
  - f-structure (functional structure)

General remarks on the representational format

- multiple levels of representation:
  - c-structure (constituent structures, licensed by PSG, $\overline{\text{X}}$ structures)
  - f-structure (functional structure)
- Mappings relate c- and f-structure.

Grammatical functions and f-structure

- In LFG, grammatical functions (subject, object, …) play a very important role. They are primitives of the theory.

Grammatical functions and f-structure

- In LFG, grammatical functions (subject, object, ...) play a very important role. They are primitives of the theory.
- A sentence such as (156a) has the functional structure in (156b):

(156) a. David devoured a sandwich.

b.
$$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle\text{SUBJ,OBJ}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{spec} & \text{A} \\ \textsc{pred} & \text{‘SANDWICH’} \end{bmatrix} \end{bmatrix}$$

Grammatical functions and f-structure

- In LFG, grammatical functions (subject, object, ...) play a very important role. They are primitives of the theory.
- A sentence such as (156a) has the functional structure in (156b):

(156) a. David devoured a sandwich.

b.
$$\begin{bmatrix} \text{PRED} & \text{‘DEVOUR}\langle\text{SUBJ},\text{OBJ}\rangle\text{’} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘DAVID’} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{SPEC} & \text{A} \\ \text{PRED} & \text{‘SANDWICH’} \end{bmatrix} \end{bmatrix}$$

- All lexical items that have a meaning (e.g. nouns, verbs, adjectives) contribute a PRED feature with a corresponding value.

Grammatical functions and f-structure

- In LFG, grammatical functions (subject, object, …) play a very important role. They are primitives of the theory.
- A sentence such as (156a) has the functional structure in (156b):

(156) a. David devoured a sandwich.

b. $$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle \text{SUBJ},\text{OBJ} \rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{spec} & \text{A} \\ \textsc{pred} & \text{‘SANDWICH’} \end{bmatrix} \end{bmatrix}$$

- All lexical items that have a meaning (e.g. nouns, verbs, adjectives) contribute a PRED feature with a corresponding value.
- The grammatical functions governed by a head (government = subcategorization) are determined in the specification of PRED.

Governable grammatical functions

The respective grammatical functions are called *governable grammatical functions*.

Examples:

SUBJ: subject

Governable grammatical functions

The respective grammatical functions are called *governable grammatical functions*.

Examples:

SUBJ:	subject
OBJ:	object

Governable grammatical functions

The respective grammatical functions are called *governable grammatical functions*.

Examples:

SUBJ:	subject
OBJ:	object
COMP	sentential complement
OBJ_θ:	secondary OBJ functions that are related to a special, language specific set of grammatical roles; English has $\text{OBJ}_{\text{THEME}}$ only.

Governable grammatical functions

The respective grammatical functions are called *governable grammatical functions*.

Examples:

SUBJ:	subject
OBJ:	object
COMP	sentential complement
OBJ_θ:	secondary OBJ functions that are related to a special, language specific set of grammatical roles; English has $\text{OBJ}_{\text{THEME}}$ only.
OBL_θ:	a group of thematically restricted oblique functions, as for instance OBL_{GOAL} or $\text{OBL}_{\text{AGENT}}$. These often correspond to adpositional phrases in c-structure.

Non-governable grammatical functions

Apart from this there are non-governable grammatical functions.

Examples:

ADJ:	adjuncts
TOPIC:	the topic of an utterance
FOCUS:	the focus of an utterance

Functional descriptions

Reference to a value of the feature TENSE in the functional structure f:

(157) (f TENSE)

Functional descriptions

Reference to a value of the feature TENSE in the functional structure f:

(157) (f TENSE)

It is possible to say something about the value which this feature should have in the feature description.

(158) (f TENSE) = PAST

Functional descriptions

Reference to a value of the feature TENSE in the functional structure f:

(157) (f TENSE)

It is possible to say something about the value which this feature should have in the feature description.

(158) (f TENSE) = PAST

The value of a feature may also be a specific f-structure.
(159) ensures that the SUBJ feature in f is the f-structure g:

(159) (f SUBJ) = g

Descriptions and f-structures

(160) a. David sneezed.

b. $(f$ PRED$) =$ 'SNEEZE$\langle$SUBJ$\rangle$'
$(f$ TENSE$) =$ PAST
$(f$ SUBJ$) = g$
$(g$ PRED$) =$ 'DAVID'

Descriptions and f-structures

(160) a. David sneezed.

b. $(f\ \text{PRED}) = \text{‘SNEEZE}\langle\text{SUBJ}\rangle\text{’}$
$(f\ \text{TENSE}) = \text{PAST}$
$(f\ \text{SUBJ}) = g$
$(g\ \text{PRED}) = \text{‘DAVID’}$

The description in (160b) describes the following structure:

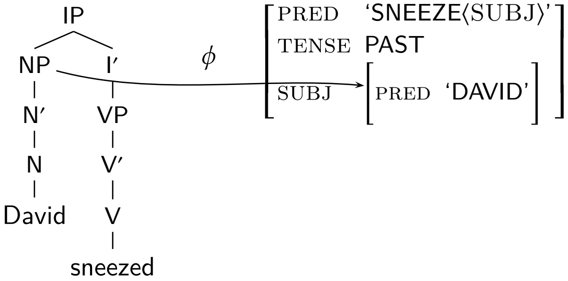
(161) $$f\text{:} \begin{bmatrix} \text{PRED} & \text{‘SNEEZE}\langle\text{SUBJ}\rangle\text{’} \\ \text{TENSE} & \text{PAST} \\ \text{SUBJ} & g\text{:} \begin{bmatrix} \text{PRED} & \text{‘DAVID’} \end{bmatrix} \end{bmatrix}$$

(160b) also describes many other structures which contain further features.
We are only interested in minimal structures containing the information provided in the description.

Mappings from c-structure to f-structure

(162) a. David sneezed.

b.



A phrase and its head always correspond to the same f-structure.
IP, I′ and I (and also VP) are mapped onto the same f-structure.

Heads and f-structure

A phrase and its head always correspond to the same f-structure:

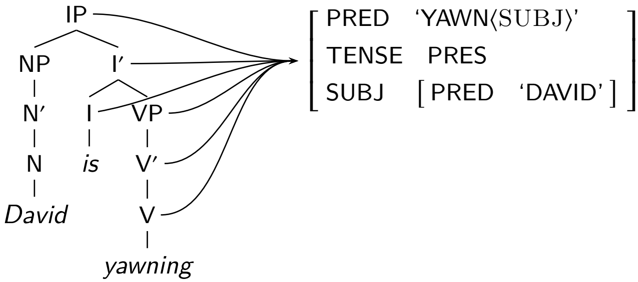
(163) $$\begin{array}{c} \text{V}' \\ | \\ \text{V} \\ | \\ \text{sneezed} \end{array} \xrightarrow{\phi} \begin{bmatrix} \text{PRED} & \text{‘SNEEZE}\langle\text{SUBJ}\rangle\text{’} \\ \text{TENSE} & \text{PAST} \end{bmatrix}$$

IP, I′, I and VP are mapped to the same f-structure

In LFG grammars of English, the CP/IP system is assumed as in GB-Theorie.
IP, I′ and I (and also VP) are mapped onto the same f-structure.

(164) a. David is yawning.

b.



Completeness

Elements required in the PRED value have to be realized.

(165) a. * David devoured.

b. $\begin{bmatrix} \text{PRED} & \text{'DEVOUR}\langle\text{SUBJ},\text{OBJ}\rangle\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'DAVID'} \end{bmatrix} \end{bmatrix}$

OBJ is missing a value in (165b), which is why (165a) is ruled out by the theory.

Coherence

All argument functions in a given f-structure have to be selected in the value of the local PRED attribut.

(166) a. * David devoured a sandwich that Peter sleeps.

b.

$$\begin{bmatrix} \text{PRED} & \text{`DEVOUR}\langle \text{SUBJ},\text{OBJ}\rangle\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`DAVID'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{SPEC} & \text{A} \\ \text{PRED} & \text{`SANDWICH'} \end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`SLEEP}\langle \text{SUBJ}\rangle\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`PETER'} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

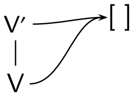
(166a) is ruled out because COMP does not appear under the arguments of *devour*.

Restrictions on the c-structure/f-structure relation

↑ : the f-structure of the immediately dominating node
↓ : f-structure of the c-structure node bearing the annotation

(167) $$\mathrm{V}' \rightarrow \underset{\substack{\uparrow = \downarrow \\ \text{f-structure of the mother} = \text{own f-structure}}}{\mathrm{V}}$$

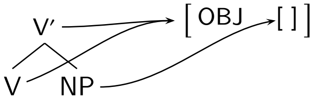
(168)



V′ rule with object

(169) $$\begin{array}{llcc} \text{V}' & \rightarrow & \text{V} & \text{NP} \\ & & \uparrow = \downarrow & (\uparrow \text{ OBJ}) = \downarrow \end{array}$$

(170)



annotation on the NP:
the OBJ value in the f-structure of the mother (↑ OBJ) is identical to the f-structure of the NP node (↓).

A lexical entry

Similarly in lexical entries:

(171) *sneezed* V $(\uparrow \text{ PRED}) = \text{‘SNEEZE}\langle\text{SUBJ}\rangle\text{’}$
$(\uparrow \text{ TENSE}) = \text{PAST}$

(172)

$$\begin{array}{c} \text{V} \\ | \\ \text{sneezed} \end{array} \longrightarrow \begin{bmatrix} \textsc{pred} & \text{‘SNEEZE}\langle\text{SUBJ}\rangle\text{’} \\ \textsc{tense} & \text{PAST} \end{bmatrix}$$

Lexical Integrity

- Bresnan & Mchombo (1995):
  Words are atoms of syntactic structure.
  Syntactic rules cannot create new words or make reference to the internal structure of words.

Lexical Integrity

- Bresnan & Mchombo (1995):
  Words are atoms of syntactic structure.
  Syntactic rules cannot create new words or make reference to the internal structure of words.
- Every terminal node (each “leaf” of the tree) is a word.

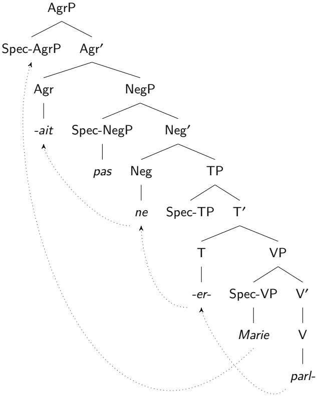
Lexical Integrity

- Bresnan & Mchombo (1995):
  Words are atoms of syntactic structure.
  Syntactic rules cannot create new words or make reference to the internal structure of words.
- Every terminal node (each "leaf" of the tree) is a word.
- This means: Pollock's (1989) analysis of (173) is excluded:

(173)	Marie	ne	parl-er-ait	pas
	Marie	NEG	speak-COND-3SG	NEG
	'Marie would not speak.'			

In Pollock's analysis, the various morphemes are in specific positions in the tree and are combined only after certain movements have been carried out.

GB analysis with morphemes as terminal symbols (Pollock 1989)



Marie	ne	parl-er-ait	pas
Marie	NEG	speak-COND-3SG	NEG

Lexical integrity and passive (I)

- observation: there are passivized adjectives which show the same morphological idiosyncrasies as the corresponding participles (Bresnan 2001: 31)

(174)
a. a well-written novel (write – written)
b. a recently given talk (give – given)
c. my broken heart (break – broken)
d. an uninhabited island (inhabit – inhabited)
e. split wood (split – split)

Lexical integrity and passive (I)

- observation: there are passivized adjectives which show the same morphological idiosyncrasies as the corresponding participles (Bresnan 2001: 31)

(174) a. a well-written novel (write – written)
 b. a recently given talk (give – given)
 c. my broken heart (break – broken)
 d. an uninhabited island (inhabit – inhabited)
 e. split wood (split – split)

- The adjectival participles have passive argument structure: the subject is suppressed and the object is what is predicated over (the noun):

(175) a. Aicke broke my heart.
 b. My heart is broken.
 c. my broken heart

(176) a. My friend is smart.
 b. my smart friend

Lexical integrity and passive (II)

- Passive participle and adjectival participle have the same form:

(177) a. Aicke broke my heart.
b. My heart was broken.
c. my broken heart

Lexical integrity and passive (II)

- Passive participle and adjectival participle have the same form:

(177) a. Aicke broke my heart.
b. My heart was broken.
c. my broken heart

- If one assumes lexical integrity,
then adjectives have to be derived in the lexicon.

Lexical integrity and passive (II)

- Passive participle and adjectival participle have the same form:

(177) a. Aicke broke my heart.
b. My heart was broken.
c. my broken heart

- If one assumes lexical integrity,
then adjectives have to be derived in the lexicon.
- If the verbal passive were not a lexical process, but rather a phrase-structural one, then the form identity would remain unexplained.

Passive as a lexical process

- Grammatical functions are primitives of the theory.
  (that is not derived from tree positions [e.g. subject = SpecIP])

Passive as a lexical process

- Grammatical functions are primitives of the theory.
  (that is not derived from tree positions [e.g. subject = SpecIP])
- Words (that is, fully inflected word forms) determine grammatical functions of their arguments.

Passive as a lexical process

- Grammatical functions are primitives of the theory.
  (that is not derived from tree positions [e.g. subject = SpecIP])
- Words (that is, fully inflected word forms) determine grammatical functions of their arguments.
- There is a hierarchy of grammatical functions.

Passive as a lexical process

- Grammatical functions are primitives of the theory.
  (that is not derived from tree positions [e.g. subject = SpecIP])
- Words (that is, fully inflected word forms) determine grammatical functions of their arguments.
- There is a hierarchy of grammatical functions.
- When participles are formed in morphology, the highest argument is suppressed.

Passive as a lexical process

- Grammatical functions are primitives of the theory.
  (that is not derived from tree positions [e.g. subject = SpecIP])
- Words (that is, fully inflected word forms) determine grammatical functions of their arguments.
- There is a hierarchy of grammatical functions.
- When participles are formed in morphology, the highest argument is suppressed.
- The next-highest argument is not realized as OBJECT but as SUBJECT.

The lexical rule

- The assignment of grammatical functions is regulated by the Lexical Mapping Theory.

The lexical rule

- The assignment of grammatical functions is regulated by the Lexical Mapping Theory.
- Earlier works (Bresnan 1982b) had an explicit formulation of the passive rule:

(178) Passive rule:
(SUBJ) $\mapsto$ ∅/(OBL)
(OBJ) $\mapsto$ (SUBJ)

This means: The subject is either not expressed at all (∅) or
as oblique Eelement (as a *von*-PP in German)

The lexical rule

- The assignment of grammatical functions is regulated by the Lexical Mapping Theory.
- Earlier works (Bresnan 1982b) had an explicit formulation of the passive rule:

(178) Passive rule:
$(\text{SUBJ}) \mapsto \emptyset/(\text{OBL})$
$(\text{OBJ}) \mapsto (\text{SUBJ})$

This means: The subject is either not expressed at all (∅) or
as oblique Eelement (as a *von*-PP in German)
If there is an accusative object, this will be realized as subject.

Verb position

- two options:
  - a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and

Verb position

- two options:
  - a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and
  - so-called extended head domains (see Berman 2003).

Verb position

- two options:
  - a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and
  - so-called extended head domains (see Berman 2003).
- Extended head domains: verb is simply omitted in the verb phrase:

(179) VP → NP* (V) (preliminary version)

All parts of the VP are optional (indicted by brackets and Kleene star).

Verb position

- two options:
  - a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and
  - so-called extended head domains (see Berman 2003).
- Extended head domains: verb is simply omitted in the verb phrase:

  (179) VP → NP* (V) (preliminary version)

  All parts of the VP are optional (indicted by brackets and Kleene star).
- As in GB analyses, the verb is in the C position.
  It contributes f-structure informtion from there.

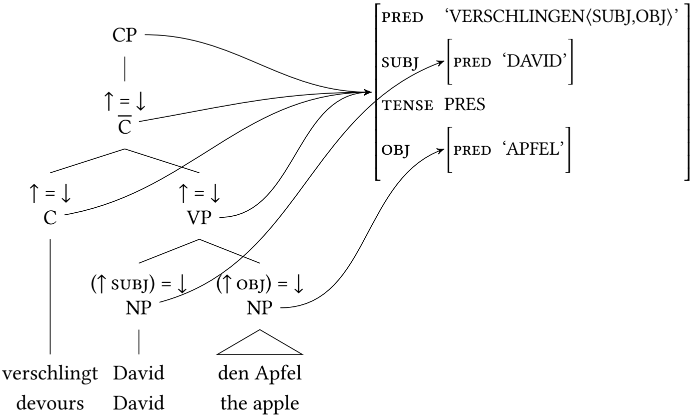
Verb position

- two options:
  - a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and
  - so-called extended head domains (see Berman 2003).
- Extended head domains: verb is simply omitted in the verb phrase:

  (179) VP → NP* (V) (preliminary version)

  All parts of the VP are optional (indicted by brackets and Kleene star).
- As in GB analyses, the verb is in the C position.
  It contributes f-structure informtion from there.
- VP without V????
  We have to make sure that all necessary items are present and nothing more: coherence and completeness.
  Where the necessary information for this comes from is not important.

An example of the verb placement analysis



Analysis adapted from Berman (2003: 41).

Local reordering

- Two options are discussed:
  - movement of arguments from a base configuration as in GB (see Choi 1999)

Local reordering

- Two options are discussed:
  - movement of arguments from a base configuration as in GB (see Choi 1999)
  - direct licensing by phrase structure rules (see Berman 1996: Section 2.1.3.1; 2003)

Local reordering as “base generateion” (I)

- Case requirements are specified in lexical items:

(180) *verschlingt* V (↑ PRED) = ‘VERSCHLINGEN⟨SUBJ,OBJ⟩’
(↑ SUBJ AGR CAS) = NOM
(↑ OBJ AGR CAS) = ACC
(↑ TENSE) = PRES

Local reordering as "base generateion" (I)

- Case requirements are specified in lexical items:

(180) *verschlingt* V $(\uparrow \text{PRED}) = \text{`VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{'}$
$(\uparrow \text{SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{TENSE}) = \text{PRES}$

- GPSG: all arguments are combined with the head in one go.

Local reordering as “base generateion” (I)

- Case requirements are specified in lexical items:

(180) *verschlingt* V
$(\uparrow \text{PRED}) = \text{`VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{'}$
$(\uparrow \text{SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{TENSE}) = \text{PRES}$

- GPSG: all arguments are combined with the head in one go.
- LFG: no argument is combined with the verb and we get a VP without anything.

(181) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{(\text{V})}$

Local reordering as “base generateion” (I)

- Case requirements are specified in lexical items:

(180) *verschlingt* V $(\uparrow \text{PRED}) = \text{‘VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{’}$
$(\uparrow \text{SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{TENSE}) = \text{PRES}$

- GPSG: all arguments are combined with the head in one go.
- LFG: no argument is combined with the verb and we get a VP without anything.

(181) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{(\text{V})}$

- Hm.

Local reordering as “base generateion” (I)

- Case requirements are specified in lexical items:

(180) *verschlingt* V $(\uparrow \text{ PRED}) = \text{‘VERSCHLINGEN}\langle \text{SUBJ,OBJ} \rangle\text{’}$
$(\uparrow \text{ SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{ OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{ TENSE}) = \text{PRES}$

- GPSG: all arguments are combined with the head in one go.
- LFG: no argument is combined with the verb and we get a VP without anything.

(181) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{\text{(V)}}$

- Hm.
- But this is just to get the recursion going.

Local reordering as “base generateion” (II)

- Case requirements are specified in lexical items:

(182) *verschlingt* V $(\uparrow \text{ PRED}) = \text{‘VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{’}$
$(\uparrow \text{ SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{ OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{ TENSE}) = \text{PRES}$

(183) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{(\text{V})}$

Local reordering as “base generateion” (II)

- Case requirements are specified in lexical items:

(182) *verschlingt* V $(\uparrow \text{ PRED}) = \text{'VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{'}$
$(\uparrow \text{ SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{ OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{ TENSE}) = \text{PRES}$

(183) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{\text{(V)}}$

- Recursive rule to add NP arguments:

(184) $\text{VP} \rightarrow \underset{(\uparrow \text{SUBJ} \mid \text{OBJ} \mid \text{OBJ}_\theta) = \downarrow}{\text{NP}} \quad \underset{\uparrow = \downarrow}{\text{VP}}$

Local reordering as “base generateion” (II)

- Case requirements are specified in lexical items:

(182) *verschlingt* V $(\uparrow \text{PRED}) = \text{‘VERSCHLINGEN}\langle\text{SUBJ,OBJ}\rangle\text{’}$
$(\uparrow \text{SUBJ AGR CAS}) = \text{NOM}$
$(\uparrow \text{OBJ AGR CAS}) = \text{ACC}$
$(\uparrow \text{TENSE}) = \text{PRES}$

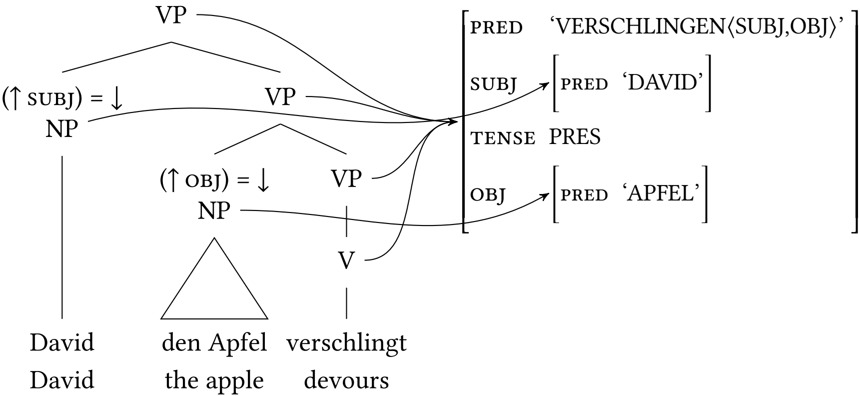
(183) $\text{VP} \rightarrow \underset{\uparrow = \downarrow}{\text{(V)}}$

- Recursive rule to add NP arguments:

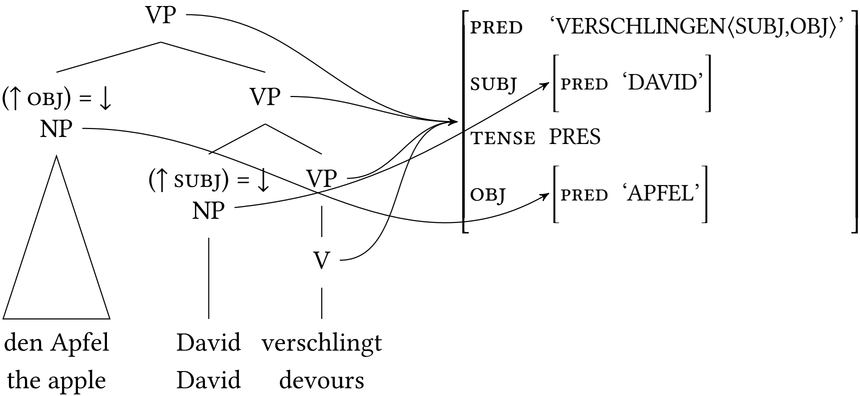
(184) $\text{VP} \rightarrow \underset{(\uparrow \text{SUBJ} \,|\text{OBJ} \,|\text{OBJ}_\theta) = \downarrow}{\text{NP}} \quad \underset{\uparrow = \downarrow}{\text{VP}}$

- similar rules for PP arguments and so on.

Binary branching with normal order (nom, acc)



Binary branching with marked order (acc, nom)



Long-distance dependencies: Discourse functions (I)

- Observation: the displaced constituent *Chris* is characterized by two functions:

  (185) Chris, we think that David saw.

  - an argument function which is normally realized in a different position: the OBJ function of *saw*

Long-distance dependencies: Discourse functions (I)

- Observation: the displaced constituent *Chris* is characterized by two functions:

  (185) Chris, we think that David saw.

  - an argument function which is normally realized in a different position: the OBJ function of *saw*
  - a certain emphasis of the information-structural status in this construction: TOPIC in the matrix clause – a discourse function

Discourse functions (II)

- grammaticalized discourse functions: TOPIC and FOCUS
  (SUBJ is a default discourse function).
  - Only grammaticalized discourse functions are represented on the level of f-structure, that is, those that are created by a fixed syntactic mechanism and that interact with the rest of the syntax.

Discourse functions (II)

- grammaticalized discourse functions: TOPIC and FOCUS
  (SUBJ is a default discourse function).
  - Only grammaticalized discourse functions are represented on the level of f-structure, that is, those that are created by a fixed syntactic mechanism and that interact with the rest of the syntax.
  - TOPIC and FOCUS are not lexically subcategorized and are therefore not subject to the completeness and coherence conditions.

Discourse functions (II)

- grammaticalized discourse functions: TOPIC and FOCUS
  (SUBJ is a default discourse function).
  - Only grammaticalized discourse functions are represented on the level of f-structure, that is, those that are created by a fixed syntactic mechanism and that interact with the rest of the syntax.
  - TOPIC and FOCUS are not lexically subcategorized and are therefore not subject to the completeness and coherence conditions.
  - TOPIC and FOCUS are identified with an f-structure that bears an argument function.

Discourse functions in f-structure

(186) a. Chris, we think that David saw.

b.
$$\begin{bmatrix} \textsc{pred} & \text{‘THINK}\langle\textsc{subj},\textsc{comp}\rangle\text{’} \\ \textsc{topic} & \begin{bmatrix} \textsc{pred} & \text{‘CHRIS’} \end{bmatrix} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘pro’} \end{bmatrix} \\ \textsc{comp} & \begin{bmatrix} \textsc{pred} & \text{‘SEE}\langle\textsc{subj},\textsc{obj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \\ \textsc{obj} & \end{bmatrix} \end{bmatrix}$$

The line means: The value of TOPIC is identical to COMP OBJ.

The constraint: $(\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj})$

Different levels of embedding (I)

(187) a. Chris, we saw.

b. $\begin{bmatrix} \text{PRED} & \text{‘SEE}\langle\text{SUBJ,OBJ}\rangle\text{’} \\ \text{TOPIC} & \begin{bmatrix} \text{PRED} & \text{‘CHRIS’} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{‘pro’} \end{bmatrix} \\ \text{OBJ} & \end{bmatrix}$

The constraint: $(\uparrow \text{TOPIC})=(\uparrow \text{OBJ})$

Different levels of embedding (II)

(188) a. Chris, we think Anna claims that David saw.

b. $$\begin{bmatrix} \text{PRED} & \text{`THINK}\langle\text{SUBJ,COMP}\rangle\text{'} \\ \text{TOPIC} & \begin{bmatrix}\text{PRED} & \text{`CHRIS'}\end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{`pro'}\end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`CLAIM}\langle\text{SUBJ,COMP}\rangle \\ \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{`ANNA'}\end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`SEE}\langle\text{SUBJ,OBJ}\rangle \\ \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{`DAVID'}\end{bmatrix} \\ \text{OBJ} & \end{bmatrix}\end{bmatrix}\end{bmatrix}$$

The constraint: $(\uparrow \text{TOPIC})=(\uparrow \text{COMP COMP OBJ})$

Functional uncertainty

- The constraints are c-structure constraints:

(189)
$$
\begin{array}{llcc}
\text{CP} & \rightarrow & \text{XP} & \text{C}' \\
 & & (\uparrow \textsc{topic}) = \downarrow & \uparrow = \downarrow \\
 & & (\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj}) &
\end{array}
$$

Functional uncertainty

- The constraints are c-structure constraints:

(189) CP $\rightarrow$ XP C′

$$\begin{array}{lll} \mathrm{CP} \rightarrow & \mathrm{XP} & \mathrm{C'} \\ & (\uparrow \textsc{topic}) = \downarrow & \uparrow{=}\downarrow \\ & (\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj}) & \end{array}$$

- But we have different levels of embedding:

(190) $(\uparrow \textsc{topic}){=}(\uparrow \textsc{obj})$
$(\uparrow \textsc{topic}){=}(\uparrow \textsc{comp obj})$
$(\uparrow \textsc{topic}){=}(\uparrow \textsc{comp comp obj})$
...

Functional uncertainty

- The constraints are c-structure constraints:

(189) $$\text{CP} \rightarrow \begin{array}{c} \text{XP} \\ (\uparrow \textsc{topic}) = \downarrow \\ (\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj}) \end{array} \quad \begin{array}{c} \text{C}' \\ \uparrow = \downarrow \\ \end{array}$$

- But we have different levels of embedding:

(190) $(\uparrow \textsc{topic}) = (\uparrow \textsc{obj})$
$(\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj})$
$(\uparrow \textsc{topic}) = (\uparrow \textsc{comp comp obj})$
...

- The generalization over these equations is:

(191) $(\uparrow \textsc{topic}) = (\uparrow \textsc{comp}^{*} \textsc{ obj})$

The Kleene star '*' stands for arbitrarily many repetitions of COMP.

Disjunctions and variables for grammatical functions

- The fronted element is not necessarily a TOPIC, FOCUS is possible as well.

Disjunctions and variables for grammatical functions

- The fronted element is not necessarily a TOPIC, FOCUS is possible as well.
- It is possible to state disjunctions:

Disjunctions and variables for grammatical functions

- The fronted element is not necessarily a TOPIC, FOCUS is possible as well.
- It is possible to state disjunctions:

(192) $(\uparrow \textsc{topic}|\textsc{focus}) = (\uparrow \textsc{comp}^{*}\ \textsc{obj})$

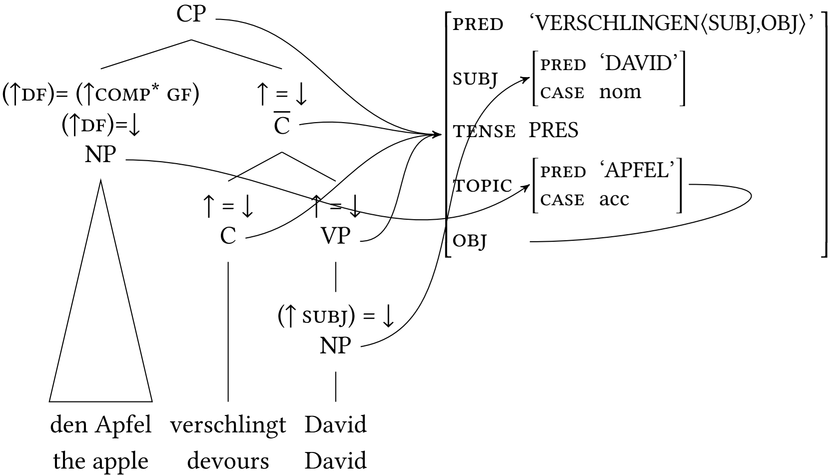
Disjunctions and variables for grammatical functions

- The fronted element is not necessarily a TOPIC, FOCUS is possible as well.
- It is possible to state disjunctions:

(192) $(\uparrow \text{TOPIC}|\text{FOCUS}) = (\uparrow \text{COMP}^{*}\ \text{OBJ})$

- TOPIC|FOCUS can be abbreviated by using the shortcut DF (discourse function).

German example



Summary

- LFG is unification-based/constraint-based and works with feature structures and PSG rules.

Summary

- LFG is unification-based/constraint-based and works with feature structures and PSG rules.
- Grammatical functions are primitives of LFG,
  they are not defined with reference to structure (as in GB)

Summary

- LFG is unification-based/constraint-based and works with feature structures and PSG rules.
- Grammatical functions are primitives of LFG,
  they are not defined with reference to structure (as in GB)
- LFG is strongly lexicalized. Valence alternations like passivization are captured in the lexicon via lexical rules.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \textit{Grammatik} \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \text{DET} \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \text{语法} \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \text{DET} \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \text{دستور} \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \text{DET} \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \text{व्याकरण} \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \text{DET} \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Categorial Grammar (CG)

Stefan Müller

Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\text{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoining Grammar (TAG)

Reading material

Müller 2020b: Chapter 8 (without 8.1.2 on semantics)

Categorial Grammar (CG)

- Categorial Grammar is the second oldest of the approaches discussed here (Ajdukiewicz 1935).
- Hotspots: Edinburgh, Uetrecht and Amsterdam
- Semanticists love CG since it syntactic combination goes hand in hand with semantic combination.
- Important articles and books:
  Steedman (1991; 2000); Steedman & Baldridge (2006)

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Representation of valence information

- complex categories replace the SUBCAT feature of GPSG

Rule	Category in the lexicon
vp → v(ditrans) np np	(vp/np)/np
vp → v(trans) np	vp/np
vp → v(np_and_pp) np pp(to)	(vp/pp)/np

Representation of valence information

- complex categories replace the SUBCAT feature of GPSG

Rule	Category in the lexicon
vp → v(ditrans) np np	(vp/np)/np
vp → v(trans) np	vp/np
vp → v(np_and_pp) np pp(to)	(vp/pp)/np

- Very few, very abstract rules:

(193) Forward application
X/Y * Y = X

Combine an X looking for a Y with a Y, where Y occurs to the right of X/Y.

Representation of valence information

- complex categories replace the SUBCAT feature of GPSG

Rule	Category in the lexicon
vp → v(ditrans) np np	(vp/np)/np
vp → v(trans) np	vp/np
vp → v(np_and_pp) np pp(to)	(vp/pp)/np

- Very few, very abstract rules:

(193) Forward application
$X/Y * Y = X$

Combine an X looking for a Y with a Y, where Y occurs to the right of X/Y.

- Valence is encoded just once, namely in the lexicon.
Until now we had two places for this:
the SUBCAT feature and the grammar rules.

Forward application

(194) Forward application
$X/Y * Y = X$

Combine an X looking for a Y with a Y, where Y occurs to the right of X/Y.

$$\frac{\textit{chased}}{vp/np} \quad \frac{\textit{Mary}}{np}$$

Forward application

(194) Forward application
X/Y * Y = X

Combine an X looking for a Y with a Y, where Y occurs to the right of X/Y.

$$\dfrac{\dfrac{chased}{vp/np} \quad \dfrac{Mary}{np}}{vp}>$$

Forward application

(194) Forward application
$X/Y * Y = X$

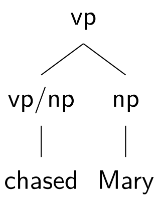
Combine an X looking for a Y with a Y, where Y occurs to the right of X/Y.

$$\dfrac{\dfrac{\textit{chased}}{vp/np} \quad \dfrac{\textit{Mary}}{np}}{vp}>$$

The category v is not needed any longer.

CG proofs vs. trees

- CG derivations may seem strange on first encounter, but you can also depict them as trees.



Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
Y * X\Y = X

Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
Y * X\Y = X

$$\frac{\textit{the}}{np/n} \quad \frac{\textit{cat}}{n} \quad \frac{\textit{chased}}{(s\backslash np)/np} \quad \frac{\textit{Mary}}{np}$$

Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
Y * X\Y = X

$$
\begin{array}{cccc}
\dfrac{\textit{the}}{np/n} & \dfrac{\textit{cat}}{n} & \dfrac{\textit{chased}}{(s\backslash np)/np} & \dfrac{\textit{Mary}}{np} \\
\multicolumn{2}{c}{\underline{\hspace{6em}}>} & & \\
\multicolumn{2}{c}{np} & &
\end{array}
$$

Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
$\mathrm{Y} * \mathrm{X}\backslash\mathrm{Y} = \mathrm{X}$

$$\dfrac{\dfrac{the}{np/n} \quad \dfrac{cat}{n}}{np}> \qquad \dfrac{\dfrac{chased}{(s\backslash np)/np} \quad \dfrac{Mary}{np}}{s\backslash np}>$$

Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
Y * X\Y = X

$$
\begin{array}{cccc}
\textit{the} & \textit{cat} & \textit{chased} & \textit{Mary} \\
\hline
np/n & n & (s\backslash np)/np & np \\
\multicolumn{2}{c}{np \;\; >} & \multicolumn{2}{c}{s\backslash np \;\; >} \\
\multicolumn{4}{c}{s \;\; <}
\end{array}
$$

Backward application

- vp can be eliminated as well: vp = s\np

(195) Backward application
Y * X\Y = X

$$\begin{array}{cccc}
the & cat & chased & Mary \\
\hline
np/n & n & (s\backslash np)/np & np \\
\hline
\multicolumn{2}{c}{np \quad >} & \multicolumn{2}{c}{s\backslash np \quad >} \\
\hline
\multicolumn{4}{c}{s \quad <}
\end{array}$$

- no explicit distinction between words and phrases:
  - intransitive verb = verb phrase = $(s\backslash np)$
  - similarly proper names = nominal phrases = np

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun
- modifiers in general: $X\backslash X$ or X/X

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun
- modifiers in general: $X\backslash X$ or X/X
- premodifier for nouns:
  noun → adj noun
  adjective $= n/n$

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun
- modifiers in general: $X\backslash X$ or X/X
- premodifier for nouns:
  noun → adj noun
  adjective $= n/n$
- postmodifier for nouns: $n\backslash n$

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun
- modifiers in general: $X\backslash X$ or X/X
- premodifier for nouns:
  noun → adj noun
  adjective $= n/n$
- postmodifier for nouns: $n\backslash n$
- vp modifier: → X $= s\backslash np$

Modification

- optional modification:
  vp → vp pp
  noun → noun pp
  arbitrarily many PPs after a VP or a noun
- modifiers in general: $X\backslash X$ or X/X
- premodifier for nouns:
  noun → adj noun
  adjective $= n/n$
- postmodifier for nouns: $n\backslash n$
- vp modifier: → X $= s\backslash np$
- vp modifier: $(s\backslash np)\backslash(s\backslash np)$.

Derivation with a Categorial Grammar

$$\frac{\textit{The}}{np/n} \quad \frac{\textit{small}}{n/n} \quad \frac{\textit{cat}}{n} \quad \frac{\textit{chased}}{(s\backslash np)/np} \quad \frac{\textit{Mary}}{np} \quad \frac{\textit{quickly}}{(s\backslash np)\backslash(s\backslash np)} \quad \frac{\textit{round}}{(s\backslash np)\backslash(s\backslash np)/np} \quad \frac{\textit{the}}{np/n} \quad \frac{\textit{garden}}{n}$$

Derivation with a Categorial Grammar

The	*small*	*cat*	*chased*	*Mary*	*quickly*	*round*	*the*	*garden*
np/n	n/n	n	$(s\backslash np)/np$	np	$(s\backslash np)\backslash(s\backslash np)$	$(s\backslash np)\backslash(s\backslash np)/np$	np/n	n
	$>$							
	n							

Derivation with a Categorial Grammar

$$
\begin{array}{ccccccccc}
\underline{\textit{The}} & \underline{\textit{small}} & \underline{\textit{cat}} & \underline{\textit{chased}} & \underline{\textit{Mary}} & \underline{\textit{quickly}} & \underline{\textit{round}} & \underline{\textit{the}} & \underline{\textit{garden}} \\
np/n & n/n & n & (s\backslash np)/np & np & (s\backslash np)\backslash(s\backslash np) & (s\backslash np)\backslash(s\backslash np)/np & np/n & n \\
 & \multicolumn{2}{c}{\overline{\quad\quad\quad\quad}>} & & & & & & \\
 & \multicolumn{2}{c}{n} & & & & & & \\
\multicolumn{3}{c}{\overline{\quad\quad\quad\quad\quad\quad\quad}>} & & & & & & \\
\multicolumn{3}{c}{np} & & & & & &
\end{array}
$$

Derivation with a Categorial Grammar

$$
\begin{array}{ccccccccc}
\underline{\textit{The}} & \underline{\textit{small}} & \underline{\textit{cat}} & \underline{\textit{chased}} & \underline{\textit{Mary}} & \underline{\textit{quickly}} & \underline{\textit{round}} & \underline{\textit{the}} & \underline{\textit{garden}} \\
np/n & n/n & n & (s\backslash np)/np & np & (s\backslash np)\backslash(s\backslash np) & (s\backslash np)\backslash(s\backslash np)/np & np/n & n \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad}>} & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad}>} & & & & \\
 & \multicolumn{2}{c}{n} & \multicolumn{2}{c}{s\backslash np} & & & & \\
\multicolumn{3}{c}{\overline{\qquad\qquad\qquad}>} & & & & & & \\
\multicolumn{3}{c}{np} & & & & & &
\end{array}
$$

Derivation with a Categorial Grammar

$$
\begin{array}{ccccccccc}
\textit{The} & \textit{small} & \textit{cat} & \textit{chased} & \textit{Mary} & \textit{quickly} & \textit{round} & \textit{the} & \textit{garden}\\
\hline
np/n & n/n & n & (s\backslash np)/np & np & (s\backslash np)\backslash(s\backslash np) & (s\backslash np)\backslash(s\backslash np)/np & np/n & n\\
\cline{2-3}\cline{4-5}
 & \multicolumn{2}{c}{n \quad >} & \multicolumn{2}{c}{s\backslash np \quad >} & & & & \\
\cline{1-3}\cline{4-6}
\multicolumn{3}{c}{np \quad >} & \multicolumn{3}{c}{s\backslash np \quad <} & & &
\end{array}
$$

Derivation with a Categorial Grammar

The	*small*	*cat*	*chased*	*Mary*	*quickly*	*round*	*the*	*garden*
np/n	n/n	n	$(s\backslash np)/np$	np	$(s\backslash np)\backslash(s\backslash np)$	$(s\backslash np)\backslash(s\backslash np)/np$	np/n	n
	n (>)		$s\backslash np$ (>)					
np (>)			$s\backslash np$ (<)					
							np (>)	

Derivation with a Categorial Grammar

The	*small*	*cat*	*chased*	*Mary*	*quickly*	*round*	*the*	*garden*
np/n	n/n	n	$(s\backslash np)/np$	np	$(s\backslash np)\backslash(s\backslash np)$	$(s\backslash np)\backslash(s\backslash np)/np$	np/n	n
	n (>)		$s\backslash np$ (>)					
np (>)			$s\backslash np$ (<)					
							np (>)	
						$(s\backslash np)\backslash(s\backslash np)$ (>)		

Derivation with a Categorial Grammar

The	*small*	*cat*	*chased*	*Mary*	*quickly*	*round*	*the*	*garden*
np/n	n/n	n	$(s\backslash np)/np$	np	$(s\backslash np)\backslash(s\backslash np)$	$(s\backslash np)\backslash(s\backslash np)/np$	np/n	n
	n >		$s\backslash np$ >					
np >			$s\backslash np$ <					
							np >	
						$(s\backslash np)\backslash(s\backslash np)$ >		
			$(s\backslash np)$ <					

Derivation with a Categorial Grammar

The	*small*	*cat*	*chased*	*Mary*	*quickly*	*round*	*the*	*garden*
np/n	n/n	n	$(s\backslash np)/np$	np	$(s\backslash np)\backslash(s\backslash np)$	$(s\backslash np)\backslash(s\backslash np)/np$	np/n	n

$n/n \;\; n \Rightarrow_{>} n$

$np/n \;\; n \Rightarrow_{>} np$

$(s\backslash np)/np \;\; np \Rightarrow_{>} s\backslash np$

$s\backslash np \;\; (s\backslash np)\backslash(s\backslash np) \Rightarrow_{<} s\backslash np$

$np/n \;\; n \Rightarrow_{>} np$

$(s\backslash np)\backslash(s\backslash np)/np \;\; np \Rightarrow_{>} (s\backslash np)\backslash(s\backslash np)$

$s\backslash np \;\; (s\backslash np)\backslash(s\backslash np) \Rightarrow_{<} (s\backslash np)$

$np \;\; (s\backslash np) \Rightarrow_{<} s$

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Verb position

- Steedman (2000: 159) for Dutch:

(196) a. verb-final *gaf* ('give'): $(s_{+SUB}\backslash np)\backslash np$
b. verb-initial *gaf* ('give'): $(s_{-SUB}/np)/np$

One item takes arguments to the left the other one to the right.

Verb position

- Steedman (2000: 159) for Dutch:

(196) a. verb-final *gaf* (‘give’): $(s_{+SUB}\backslash np)\backslash np$
b. verb-initial *gaf* (‘give’): $(s_{-SUB}/np)/np$

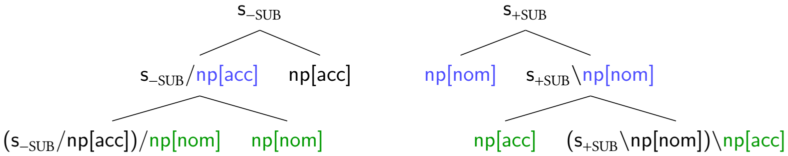
One item takes arguments to the left the other one to the right.

- Lexical items are related by lexical rule.

Comment on variable branching analysis

Note that NPs are combined in different orders:
To get normal order, one would have to assume:

(197) a. verb-final: $(\mathrm{s}_{+\mathrm{SUB}} \backslash \mathrm{np[nom]}) \backslash \mathrm{np[acc]}$
 b. verb-initial: $(\mathrm{s}_{-\mathrm{SUB}} / \mathrm{np[acc]}) / \mathrm{np[nom]}$



Two different branchings. So Müller 2005 for criticism.

Verb position with empty element

Jacobs (1991): empty element in final position
taking the arguments of the verb and the verb in initial position as arguments.

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Local reordering

- Until now: combinations either to the left or to the right.
  Combinations always in a fixed order from outside inwards.

Local reordering

- Until now: combinations either to the left or to the right.
  Combinations always in a fixed order from outside inwards.
- Steedman & Baldridge (2006) distinguish:
  - languages in which the order of combination does not matter

Local reordering

- Until now: combinations either to the left or to the right.
  Combinations always in a fixed order from outside inwards.
- Steedman & Baldridge (2006) distinguish:
  - languages in which the order of combination does not matter
  - languages in which the direction of combination does not matter

Local reordering

- Until now: combinations either to the left or to the right. Combinations always in a fixed order from outside inwards.
- Steedman & Baldridge (2006) distinguish:
  - languages in which the order of combination does not matter
  - languages in which the direction of combination does not matter

English	(s\np)/np	S(VO)
Latin	s{|np[nom], |np[acc] }	free order
Tagalog	s{/np[nom], /np[acc] }	free order, verb-initial
Japanese	s{\np[nom], \np[acc] }	free order, verb-final

Elements in brackets can be combined with s in any order.
'|' instead of '\' or '/' means that direction of combination is free.

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Passive: A lexical rule

- Lexical rule (Dowty 1978: 412; Dowty 2003: Section 3.4):

  (198) $\alpha \in (\mathrm{s}\backslash\mathrm{np})/\mathrm{np} \rightarrow \mathrm{PST\text{-}PART}(\alpha) \in \mathrm{PstP}/\mathrm{np}_{by}$

  For every (strictly) transitive verb α, there is a past participle form with the category PstP/np_{by}.
  np_{by} stands for the *by*-PP.
- example:

  (199) a. touch: (s\np)/np

  b. touched: PstP/np_{by}

Passive: An example derivation

$$
\begin{array}{ccccc}
\textit{John} & \textit{was} & \textit{touched} & \textit{by} & \textit{Mary.} \\
\hline
np & (s\backslash np)/PstP & PstP/np_{by}\ ^{\mathrm{LR}} & np_{by}/np & np \\
\cline{4-5}
 & & & \multicolumn{2}{c}{np_{by}\quad >} \\
\cline{3-5}
 & & \multicolumn{3}{c}{PstP\quad >} \\
\cline{2-5}
 & \multicolumn{4}{c}{s\backslash np\quad >} \\
\hline
\multicolumn{5}{c}{s\quad <}
\end{array}
$$

And German?

- Well, due to the possibility of reordering items, we have sets:

(200) a. lieben 'to love': $\mathrm{s}_{+\mathrm{SUB}}$ { \np[nom]$_i$, \np[acc]$_j$ }

b. geliebt 'loved': $\mathrm{s}_{\mathrm{pas}}$ { \np[nom]$_j$, \pp[von]$_i$ }

And German?

- Well, due to the possibility of reordering items, we have sets:

(200) a. lieben 'to love': $\mathrm{s}_{+\mathrm{SUB}}$ { \np[nom]$_i$, \np[acc]$_j$ }
b. geliebt 'loved': $\mathrm{s}_{\mathrm{pas}}$ { \np[nom]$_j$, \pp[von]$_i$ }

- Passive rule would be different for German and English.

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Long distance dependencies

- Steedman (1989: Section 1.2.4): analysis of long distance dependencies without movement and empty elements.

(201) a. These apples, Harry must have been eating.
      b. apples which Harry devours

Long distance dependencies

- Steedman (1989: Section 1.2.4): analysis of long distance dependencies without movement and empty elements.

(201) a. These apples, Harry must have been eating.
b. apples which Harry devours

- *Harry must have been eating* and *Harry devours* are just s/np.

Long distance dependencies

- Steedman (1989: Section 1.2.4): analysis of long distance dependencies without movement and empty elements.

(201) a. These apples, Harry must have been eating.
b. apples which Harry devours

- *Harry must have been eating* and *Harry devours* are just s/np.
- But the missing np is missing at the end of the clause. We need an extension! Type raising.

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s

b. s/(s\np) * s\np → s

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s
b. s/(s\np) * s\np → s

Type raising simply reverses the direction of selection:
a: vp is the functor and the np is the argument

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s
b. s/(s\np) * s\np → s

Type raising simply reverses the direction of selection:
a: vp is the functor and the np is the argument

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s
b. s/(s\np) * s\np → s

Type raising simply reverses the direction of selection:
a: vp is the functor and the np is the argument
b: type raised np is the functor, and the vp is the argument.

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s
b. s/(s\np) * s\np → s

Type raising simply reverses the direction of selection:
a: vp is the functor and the np is the argument
b: type raised np is the functor, and the vp is the argument.

Type Raising

The category np can be transformed into the category (s/(s\np)) by *type raising*. Combining this category with (s\np) yields the same result as combining np and (s\np) with backward application.

(202) a. np * s\np → s
b. s/(s\np) * s\np → s

Type raising simply reverses the direction of selection:
a: vp is the functor and the np is the argument
b: type raised np is the functor, and the vp is the argument.

The result is the same: s.

Forward and backward composition

- Two additional means of combination: forward and backward composition:

(203) a. Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

b. Backward composition ($<$ B)
$Y\backslash Z * X\backslash Y = X\backslash Z$

Forward and backward composition

- Two additional means of combination: forward and backward composition:

(203) a. Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

b. Backward composition ($<$ B)
$Y\backslash Z * X\backslash Y = X\backslash Z$

- Example forward composition:

(204) Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

If I find a Y, then I am a complete X.

Forward and backward composition

- Two additional means of combination: forward and backward composition:

(203) a. Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

b. Backward composition ($<$ B)
$Y\backslash Z * X\backslash Y = X\backslash Z$

- Example forward composition:

(204) Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

If I find a Y, then I am a complete X.

- I have a Y,

Forward and backward composition

- Two additional means of combination: forward and backward composition:

(203) a. Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

b. Backward composition ($<$ B)
$Y\backslash Z * X\backslash Y = X\backslash Z$

- Example forward composition:

(204) Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

If I find a Y, then I am a complete X.

- I have a Y, but a Z is missing.

Forward and backward composition

- Two additional means of combination: forward and backward composition:

(203) a. Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$
b. Backward composition ($<$ B)
$Y\backslash Z * X\backslash Y = X\backslash Z$

- Example forward composition:

(204) Forward composition ($>$ B)
$X/Y * Y/Z = X/Z$

If I find a Y, then I am a complete X.

- I have a Y, but a Z is missing.
- If I combine X/Y with Y/Z despite the missing Z,
I get something still lacking a Z.

Forward and backward composition: Passing the np on

$$\dfrac{\textit{These apples}}{np} \quad \dfrac{\dfrac{\textit{Harry}}{\quad}{>}\mathbf{T}}{s/(s\backslash np)} \quad \dfrac{\textit{must}}{(s\backslash np)/vp} \quad \dfrac{\textit{have}}{vp/vp\text{-}en} \quad \dfrac{\textit{been}}{vp\text{-}en/vp\text{-}ing} \quad \dfrac{\textit{eating}}{vp\text{-}ing/np}$$

Forward and backward composition: Passing the np on

$$
\begin{array}{cccccc}
\textit{These apples} & \textit{Harry} & \textit{must} & \textit{have} & \textit{been} & \textit{eating} \\
\hline
np & \overline{s/(s\backslash np)}^{\;>\mathbf{T}} & (s\backslash np)/vp & vp/vp\text{-}en & vp\text{-}en/vp\text{-}ing & vp\text{-}ing/np \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad s/vp \qquad\qquad}^{\;>\mathbf{B}}} & & &
\end{array}
$$

Forward and backward composition: Passing the np on

These apples	*Harry*	*must*	*have*	*been*	*eating*
np	$s/(s\backslash np)$ (>T)	$(s\backslash np)/vp$	$vp/vp\text{-}en$	$vp\text{-}en/vp\text{-}ing$	$vp\text{-}ing/np$

$$\dfrac{s/(s\backslash np) \quad (s\backslash np)/vp}{s/vp}\ {>}\mathbf{B}$$

$$\dfrac{s/vp \quad vp/vp\text{-}en}{s/vp\text{-}en}\ {>}\mathbf{B}$$

Forward and backward composition: Passing the np on

$$
\begin{array}{cccccc}
\textit{These apples} & \textit{Harry} & \textit{must} & \textit{have} & \textit{been} & \textit{eating} \\
\hline
np & \overline{s/(s\backslash np)}^{>\mathbf{T}} & (s\backslash np)/vp & vp/vp\text{-}en & vp\text{-}en/vp\text{-}ing & vp\text{-}ing/np \\
 & \multicolumn{2}{c}{\overline{\quad s/vp \quad}^{>\mathbf{B}}} & & & \\
 & \multicolumn{3}{c}{\overline{\quad s/vp\text{-}en \quad}^{>\mathbf{B}}} & & \\
 & \multicolumn{4}{c}{\overline{\quad s/vp\text{-}ing \quad}^{>\mathbf{B}}} & \\
\end{array}
$$

Forward and backward composition: Passing the np on

$$
\begin{array}{cccccc}
\textit{These apples} & \textit{Harry} & \textit{must} & \textit{have} & \textit{been} & \textit{eating} \\
\hline np & \xrightarrow{\quad}{>}\mathbf{T}\ \ s/(s\backslash np) & (s\backslash np)/vp & vp/vp\text{-}en & vp\text{-}en/vp\text{-}ing & \color{green}{vp\text{-}ing}/np \\
 & \multicolumn{2}{c}{\overline{\quad s/vp \quad}{>}\mathbf{B}} & & & \\
 & \multicolumn{3}{c}{\overline{\quad s/vp\text{-}en \quad}{>}\mathbf{B}} & & \\
 & \multicolumn{4}{c}{\overline{\quad s/\color{green}{vp\text{-}ing} \quad}{>}\mathbf{B}} & \\
 & \multicolumn{5}{c}{\overline{\quad s/np \quad}{>}\mathbf{B}}
\end{array}
$$

The top of the dependency: The topicalization rule

Steedman (1989):
rule for turning an X into a functor selecting a sentence lacking an X:

(205) Topicalization (↑):
$X \Rightarrow st/(s/X)$
where $X \in \{ np, pp, vp, ap, s' \}$

Topicalization long distance

$$
\begin{array}{cccccc}
\textit{These apples} & \textit{Harry} & \textit{must} & \textit{have} & \textit{been} & \textit{eating} \\
\overline{\quad\quad\quad}{>}\!\uparrow & \overline{\quad\quad}{>}\mathbf{T} & \overline{\quad\quad\quad} & \overline{\quad\quad\quad} & \overline{\quad\quad\quad} & \overline{\quad\quad\quad} \\
st/(s/np) & s/(s\backslash np) & (s\backslash np)/vp & vp/vp\text{-}en & vp\text{-}en/vp\text{-}ing & vp\text{-}ing/np \\
 & \multicolumn{2}{c}{\overline{\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} & & & \\
 & \multicolumn{2}{c}{s/vp} & & & \\
 & \multicolumn{3}{c}{\overline{\quad\quad\quad\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} & & \\
 & \multicolumn{3}{c}{s/vp\text{-}en} & & \\
 & \multicolumn{4}{c}{\overline{\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} & \\
 & \multicolumn{4}{c}{s/vp\text{-}ing} & \\
 & \multicolumn{5}{c}{\overline{\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} \\
 & \multicolumn{5}{c}{s/np}
\end{array}
$$

Topicalization long distance

$$
\begin{array}{cccccc}
\textit{These apples} & \textit{Harry} & \textit{must} & \textit{have} & \textit{been} & \textit{eating} \\
\hline
st/(s/np) \;{>}\uparrow & s/(s\backslash np) \;{>}\mathbf{T} & (s\backslash np)/vp & vp/vp\text{-}en & vp\text{-}en/vp\text{-}ing & vp\text{-}ing/np \\
\cline{2-3}
& \multicolumn{2}{c}{s/vp \quad {>}\mathbf{B}} & & & \\
\cline{2-4}
& \multicolumn{3}{c}{s/vp\text{-}en \quad {>}\mathbf{B}} & & \\
\cline{2-5}
& \multicolumn{4}{c}{s/vp\text{-}ing \quad {>}\mathbf{B}} & \\
\cline{2-6}
& \multicolumn{5}{c}{s/np \quad {>}\mathbf{B}} \\
\hline
\multicolumn{6}{c}{st \quad {>}}
\end{array}
$$

Topicalization across clause boundaries

$$\dfrac{\textit{Apples}}{st/(s/np)}{>}{\uparrow} \quad \dfrac{\textit{I}}{s/(s\backslash np)}{>}\mathbf{T} \quad \dfrac{\textit{believe}}{(s\backslash np)/s'} \quad \dfrac{\textit{that}}{s'/s} \quad \dfrac{\textit{Harry}}{s/(s\backslash np)}{>}\mathbf{T} \quad \dfrac{\textit{eats}}{(s\backslash np)/np}$$

Topicalization across clause boundaries

$$
\begin{array}{cccccc}
\textit{Apples} & \textit{I} & \textit{believe} & \textit{that} & \textit{Harry} & \textit{eats} \\
\overline{}{>}\uparrow & \overline{}{>}\mathbf{T} & \overline{} & \overline{} & \overline{}{>}\mathbf{T} & \overline{} \\
st/(s/np) & s/(s\backslash np) & (s\backslash np)/s' & s'/s & s/(s\backslash np) & (s\backslash np)/np \\
 & \multicolumn{2}{c}{\overline{}{>}\mathbf{B}} & & & \\
 & \multicolumn{2}{c}{s/s'} & & &
\end{array}
$$

Topicalization across clause boundaries

$$
\begin{array}{cccccc}
\textit{Apples} & \textit{I} & \textit{believe} & \textit{that} & \textit{Harry} & \textit{eats} \\
\overline{\quad\quad}{>}\uparrow & \overline{\quad\quad}{>}\mathbf{T} & \overline{\quad\quad\quad} & \overline{\quad} & \overline{\quad\quad}{>}\mathbf{T} & \overline{\quad\quad\quad} \\
st/(s/np) & s/(s\backslash np) & (s\backslash np)/s' & s'/s & s/\color{green}{(s\backslash np)} & \color{green}{(s\backslash np)}/np \\
 & \multicolumn{2}{c}{\overline{\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} & & \multicolumn{2}{c}{\overline{\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}} \\
 & \multicolumn{2}{c}{s/s'} & & \multicolumn{2}{c}{s/np}
\end{array}
$$

Topicalization across clause boundaries

$$
\begin{array}{cccccc}
\textit{Apples} & \textit{I} & \textit{believe} & \textit{that} & \textit{Harry} & \textit{eats}\\
\overline{st/(s/np)}^{>\uparrow} & \overline{s/(s\backslash np)}^{>\mathbf{T}} & \overline{(s\backslash np)/s'} & \overline{s'/s} & \overline{s/(s\backslash np)}^{>\mathbf{T}} & \overline{(s\backslash np)/np}\\
 & \multicolumn{2}{c}{\overline{\qquad s/s' \qquad}^{>\mathbf{B}}} & & \multicolumn{2}{c}{\overline{\qquad s/np \qquad}^{>\mathbf{B}}}\\
 & \multicolumn{3}{c}{\overline{\qquad s/s \qquad}^{>\mathbf{B}}} & &
\end{array}
$$

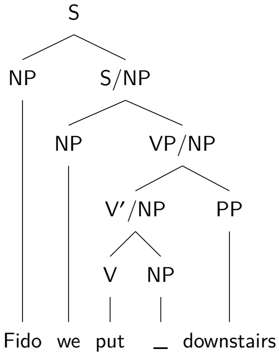
Topicalization across clause boundaries

$$
\begin{array}{cccccc}
\mathit{Apples} & \mathit{I} & \mathit{believe} & \mathit{that} & \mathit{Harry} & \mathit{eats} \\
\overline{st/(s/np)}{>}\uparrow & \overline{s/(s\backslash np)}{>}\mathbf{T} & \overline{(s\backslash np)/s'} & \overline{s'/s} & \overline{s/(s\backslash np)}{>}\mathbf{T} & \overline{(s\backslash np)/np} \\
 & \multicolumn{2}{c}{\overline{\qquad s/s' \qquad}{>}\mathbf{B}} & & \multicolumn{2}{c}{\overline{\qquad s/np \qquad}{>}\mathbf{B}} \\
 & \multicolumn{3}{c}{\overline{\qquad s/s \qquad}{>}\mathbf{B}} & & \\
 & \multicolumn{5}{c}{\overline{\qquad\qquad s/np \qquad\qquad}{>}\mathbf{B}}
\end{array}
$$

Topicalization across clause boundaries

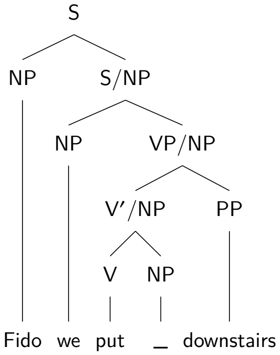
Apples	*I*	*believe*	*that*	*Harry*	*eats*
$\overline{\quad}{>}\uparrow$	$\overline{\quad}{>}\mathbf{T}$	$\overline{\quad}$	$\overline{\quad}$	$\overline{\quad}{>}\mathbf{T}$	$\overline{\quad}$
$st/(s/np)$	$s/(s\backslash np)$	$(s\backslash np)/s'$	s'/s	$s/(s\backslash np)$	$(s\backslash np)/np$
	$\overline{\quad\quad}{>}\mathbf{B}$ (I believe)			$\overline{\quad\quad}{>}\mathbf{B}$ (Harry eats)	
	s/s'			s/np	
	$\overline{\quad\quad\quad}{>}\mathbf{B}$ (I believe that)				
	s/s				
	$\overline{\quad\quad\quad\quad}{>}\mathbf{B}$ (I believe that Harry eats)				
	s/np				
$\overline{\quad\quad\quad\quad\quad}{>}$ (Apples I believe that Harry eats)					
st					

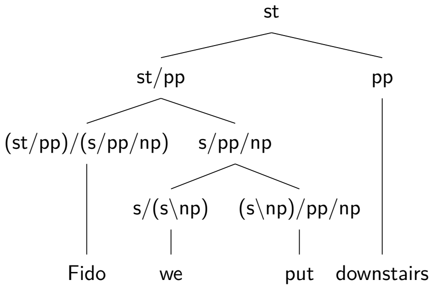
Extraction from the middle?



- Extraction from the middle is unproblematic in a GPSG-style analysis.

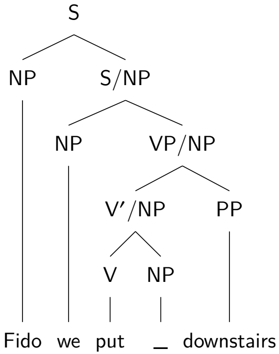
Extraction from the middle?

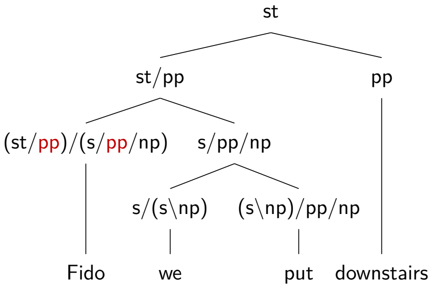




- Extraction from the middle is unproblematic in a GPSG-style analysis.
- CG would look correspond to the tree on the right.

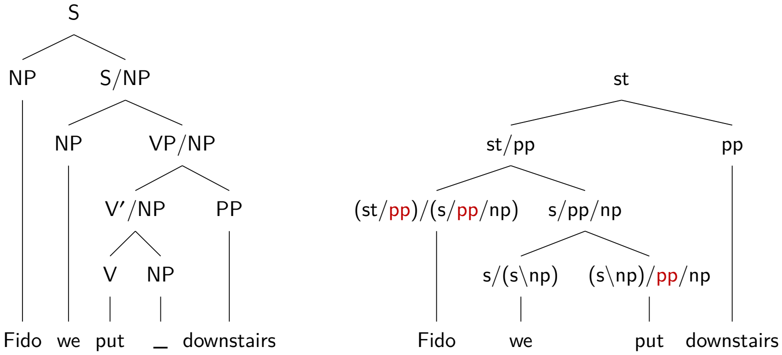
Extraction from the middle?





- Extraction from the middle is unproblematic in a GPSG-style analysis.
- CG would look correspond to the tree on the right.
- But we neither have the category for *Fido*

Extraction from the middle?



- Extraction from the middle is unproblematic in a GPSG-style analysis.
- CG would look correspond to the tree on the right.
- But we neither have the category for *Fido* nor can we combine *we* and *put*.

Additional rules

- We can combine Y with Y missing two things:

(206) Forward composition for n=2 ($>$ BB)
$X/Y * (Y/Z1)/Z2 = (X/Z1)/Z2$

Additional rules

- We can combine Y with Y missing two things:

(206) Forward composition for n=2 (> BB)
$X/Y * (Y/Z1)/Z2 = (X/Z1)/Z2$

- Topicalization turns X2 into a functor:

(207) Topicalization for n=2 (↑↑):
$X2 \Rightarrow (st/X1)/((s/X1)/X2)$
where X1 and X2 $\in$ { NP, PP, VP, AP, S′ }

Additional rules

- We can combine Y with Y missing two things:

(206) Forward composition for n=2 (> BB)
X/Y ∗ (Y/Z1)/Z2 = (X/Z1)/Z2

- Topicalization turns X2 into a functor:

(207) Topicalization for n=2 (↑↑):
X2 ⇒ (st/X1)/((s/X1)/X2)
where X1 and X2 ∈ { NP, PP, VP, AP, S′ }

The result of the combination is something that still needs the element from the right periphery of the clause (X1).

Additional rules

- We can combine Y with Y missing two things:

(206) Forward composition for n=2 (> BB)
$X/Y * (Y/Z1)/Z2 = (X/Z1)/Z2$

- Topicalization turns X2 into a functor:

(207) Topicalization for n=2 (↑↑):
$X2 \Rightarrow (st/X1)/((s/X1)/X2)$
where X1 and X2 $\in$ { NP, PP, VP, AP, S′ }

The result of the combination is something that still needs the element from the right periphery of the clause (X1).
Something with the gap (X2) at the outside is selected.

Analysis of fronting middle argument

Fido	*we*	*put*	*downstairs*
$\overline{\hphantom{(st/pp)/((s/pp)/np)}}{>}{\uparrow\uparrow}$	$\overline{\hphantom{s/(s\backslash np)}}{>}\mathbf{T}$	$\overline{\hphantom{((s\backslash np)/pp)/np}}$	$\overline{\hphantom{pp}}$
$(st/pp)/((s/pp)/np)$	$s/(s\backslash np)$	$((s\backslash np)/pp)/np$	pp

Analysis of fronting middle argument

$$
\begin{array}{cccc}
\textit{Fido} & \textit{we} & \textit{put} & \textit{downstairs} \\
\overline{(st/pp)/((s/pp)/np)}^{>\uparrow\uparrow} & \overline{s/(s\backslash np)}^{>\mathbf{T}} & \overline{((s\backslash np)/pp)/np} & \overline{pp} \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad(s/pp)/np\qquad\qquad}^{>\mathbf{BB}}} &
\end{array}
$$

Analysis of fronting middle argument

$$
\begin{array}{cccc}
\textit{Fido} & \textit{we} & \textit{put} & \textit{downstairs} \\
\overline{(st/pp)/((s/pp)/np)}{>}\!\uparrow\uparrow & \overline{s/(s\backslash np)}{>}\mathbf{T} & \overline{((s\backslash np)/pp)/np} & \overline{pp} \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad\qquad\qquad}{>}\mathbf{BB}} & \\
 & \multicolumn{2}{c}{(s/pp)/np} & \\
\multicolumn{3}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}{>}} & \\
\multicolumn{3}{c}{st/pp} &
\end{array}
$$

Analysis of fronting middle argument

Fido	*we*	*put*	*downstairs*
$\overline{}{>}\uparrow\uparrow$	$\overline{}{>}\mathbf{T}$		
$(st/pp)/((s/pp)/np)$	$s/(s\backslash np)$	$((s\backslash np)/pp)/np$	pp

$$
\dfrac{\dfrac{(st/pp)/((s/pp)/np) \quad \dfrac{s/(s\backslash np) \quad ((s\backslash np)/pp)/np}{(s/pp)/np}{>}\mathbf{BB}}{st/pp}{>} \quad pp}{st}{>}
$$

Outline

- General remarks on the representational format
- Verb position
- Local reordering (aka scrambling)
- Passive
- Long distance dependencies
- Summary and classification

Summary and classification

- lexical and phrasal approaches
- headless constructions
- relative clauses and nonlocal dependencies

Lexical and phrasal approaches

- GPSG: approaches with valence in rules have problems with
  - morphology

Lexical and phrasal approaches

- GPSG: approaches with valence in rules have problems with
  - morphology
  - partial fronting

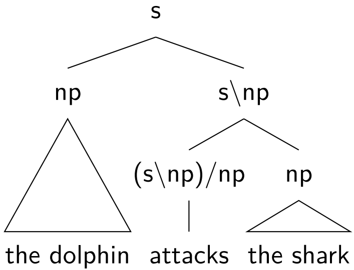
Lexical and phrasal approaches

- GPSG: approaches with valence in rules have problems with
  - morphology
  - partial fronting
- This also carries over to phrasal approaches in Construction Grammar. See Müller & Wechsler 2014 and Müller 2020b: Chapter 21 for extensive discussion.

Lexical and phrasal approaches

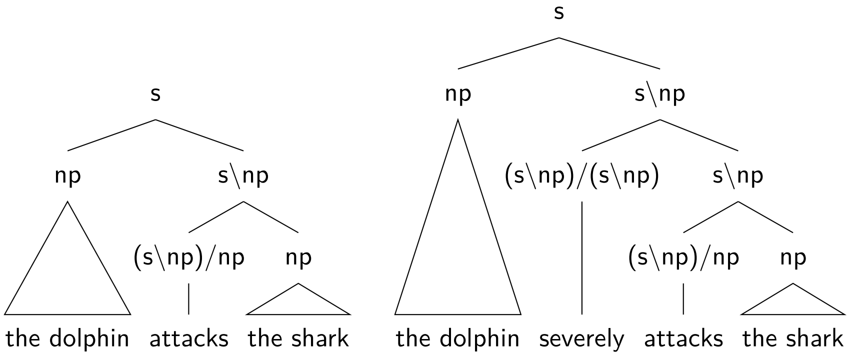
- GPSG: approaches with valence in rules have problems with
  - morphology
  - partial fronting
- This also carries over to phrasal approaches in Construction Grammar. See Müller & Wechsler 2014 and Müller 2020b: Chapter 21 for extensive discussion.
- Construction Grammarians often argue for phrasal approaches based on language acquisition, which is pattern-based, but look:

Trees are determined lexically



The pattern [Subj Verb Obj] is completely determined by (s\np)/np.
The lexicon tells the syntax what to do!

Trees are determined lexically



The pattern [Subj Verb Obj] is completely determined by (s\np)/np.
The lexicon tells the syntax what to do!

And there is room for adjuncts!

Headless constructions

- CG has very few combinatorial schemata.
  They all assume a functor and an argument.

Headless constructions

- CG has very few combinatorial schemata.
  They all assume a functor and an argument.
- But there are constructions where it is difficult/impossible to argue for a head.
  Matsuyama (2004) and Jackendoff (2008) discuss the NPN Construction:

(208) a. Student after student left the room.

b. Day after day after day went by, but I never found the courage to talk to her.[7]

[7] Bargmann (2015)

Headless constructions

- CG has very few combinatorial schemata.
  They all assume a functor and an argument.
- But there are constructions where it is difficult/impossible to argue for a head.
  Matsuyama (2004) and Jackendoff (2008) discuss the NPN Construction:

(208) a. Student after student left the room.

b. Day after day after day went by, but I never found the courage to talk to her.[7]

- This really seems to be a phrasal pattern.
  GPSG, CxG, HPSG, LFG, TAG can do this, Minimalism, CG, DG can't.
  (but see Hudson 2021 on a Word Grammar solution)

[7] Bargmann (2015)

Relative clauses and nonlocal dependencies

Steedman & Baldridge (2006: 614):

(209) the man that Manny says Anna married

Relative clauses and nonlocal dependencies

Steedman & Baldridge (2006: 614):

(209) the man that Manny says Anna married

Lexical entry for relative pronoun:

(210) (n\n)/(s/np)

If I find a sentence missing an NP to the right of me,
I can form a noun modifier (n\n) with it.

The relative pronoun is the head (functor) in this analysis.

Relative clauses and nonlocal dependencies

that	*Manny*	*says*	*Anna*	*married*
$(n\backslash n)/(s/np)$	$s/(s\backslash np)$ >T	$(s\backslash np)/s$	$s/(s\backslash np)$ >T	$(s\backslash np)/np$

Relative clauses and nonlocal dependencies

$$
\begin{array}{ccccc}
\textit{that} & \textit{Manny} & \textit{says} & \textit{Anna} & \textit{married} \\
\hline
(n\backslash n)/(s/np) & s/(s\backslash np) \; {>}\mathbf{T} & (s\backslash np)/s & s/(s\backslash np) \; {>}\mathbf{T} & (s\backslash np)/np \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad\qquad}\,{>}\mathbf{B}} & & \\
 & \multicolumn{2}{c}{s/s} & & \\
\end{array}
$$

Relative clauses and nonlocal dependencies

$$
\begin{array}{ccccc}
\textit{that} & \textit{Manny} & \textit{says} & \textit{Anna} & \textit{married} \\
\hline
(n\backslash n)/(s/np) & \overline{s/(s\backslash np)}^{>\mathbf{T}} & (s\backslash np)/s & \overline{s/(s\backslash np)}^{>\mathbf{T}} & (s\backslash np)/np \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad s/s \qquad\qquad}^{>\mathbf{B}}} & \multicolumn{2}{c}{\overline{\qquad\qquad s/np \qquad\qquad}^{>\mathbf{B}}}
\end{array}
$$

Relative clauses and nonlocal dependencies

$$
\begin{array}{ccccc}
\textit{that} & \textit{Manny} & \textit{says} & \textit{Anna} & \textit{married} \\
\hline
(n\backslash n)/(s/np) & \overline{s/(s\backslash np)}^{>T} & (s\backslash np)/s & \overline{s/(s\backslash np)}^{>T} & (s\backslash np)/np \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad}^{>B}} & \multicolumn{2}{c}{\overline{\qquad\qquad\qquad}^{>B}} \\
 & \multicolumn{2}{c}{s/s} & \multicolumn{2}{c}{s/np} \\
 & \multicolumn{4}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad}^{>B}} \\
 & \multicolumn{4}{c}{s/np}
\end{array}
$$

Relative clauses and nonlocal dependencies

$$
\begin{array}{ccccc}
\textit{that} & \textit{Manny} & \textit{says} & \textit{Anna} & \textit{married} \\
\hline
(n\backslash n)/(s/np) & \overline{s/(s\backslash np)}^{>\mathbf{T}} & (s\backslash np)/s & \overline{s/(s\backslash np)}^{>\mathbf{T}} & (s\backslash np)/np \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad s/s \qquad\qquad}^{>\mathbf{B}}} & \multicolumn{2}{c}{\overline{\qquad\qquad s/np \qquad\qquad}^{>\mathbf{B}}} \\
 & \multicolumn{4}{c}{\overline{\qquad\qquad\qquad\qquad s/np \qquad\qquad\qquad\qquad}^{>\mathbf{B}}} \\
\multicolumn{5}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad n\backslash n \qquad\qquad\qquad\qquad\qquad\qquad}^{>}}
\end{array}
$$

Remark regarding this analysis

Pollard (1988): relative pronoun = head? What about pied piping?

(211) a. Here's the minister [[in [the middle [of [whose sermon]]]] the dog barked].[8]

b. Reports [[the height of the lettering on the covers of which] the government prescribes] should be abolished.[9]

See Morrill 1995; Steedman 1997 for proposals.

[8] Pollard & Sag (1994: 212)
[9] Ross (1967: 109)

Summary

- simple combinatory rules

Summary

- simple combinatory rules
- always functor-based

Summary

- simple combinatory rules
- always functor-based
- nonlocal dependencies without empty elements but with composition
  Results in unusual constituents, but Steedman (1989) argues that they are needed for coordination.

Homework

Analyze the sentence:

(212) The children in the room laugh loudly.

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \textit{Grammatik} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{语法} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{دستور} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \textit{word} \\ \text{ORTH} \ \langle \ \text{व्याकरण} \ \rangle \\ \text{SYN|CAT|SUBCAT} \ \langle \ \text{DET} \ \rangle \\ \text{SEM} \begin{bmatrix} \text{IND} & \boxed{0} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{grammar} \\ \text{INST} \ \boxed{0} \end{bmatrix} \right\} \end{bmatrix} \end{bmatrix}$$

Grammatical theory

Head-Driven Phrase Structure Grammar (HPSG)

Stefan Müller
Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\mathrm{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoning Grammar (TAG)

Reading material

Müller 2020b: Chapter 9

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto
  (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history
- Ivan Sag was one of the developers of GPSG, Pollard worked in a version of CG.

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history
- Ivan Sag was one of the developers of GPSG, Pollard worked in a version of CG.
- HPSG is part of West-Coast linguistics (LFG, BCG).

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history
- Ivan Sag was one of the developers of GPSG, Pollard worked in a version of CG.
- HPSG is part of West-Coast linguistics (LFG, BCG).
- Hotspots: Columbus (Ohio), Buffalo, Germany, Paris, Seoul

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history
- Ivan Sag was one of the developers of GPSG, Pollard worked in a version of CG.
- HPSG is part of West-Coast linguistics (LFG, BCG).
- Hotspots: Columbus (Ohio), Buffalo, Germany, Paris, Seoul
- Teaching material and overviews:
  Müller 2013b; 2014; Levine & Meurers 2006; Müller & Machicao y Priemer 2019

Head-Driven Phrase Structure Grammar (HPSG)

- developed by Carl Pollard and Ivan Sag in the mid-80s in Stanford and in the Hewlett-Packard research laboratories in Palo Alto (Pollard & Sag 1987; 1994); see (Flickinger, Wasow & Pollard 2021) for history
- Ivan Sag was one of the developers of GPSG, Pollard worked in a version of CG.
- HPSG is part of West-Coast linguistics (LFG, BCG).
- Hotspots: Columbus (Ohio), Buffalo, Germany, Paris, Seoul
- Teaching material and overviews:
  Müller 2013b; 2014; Levine & Meurers 2006; Müller & Machicao y Priemer 2019
- 1500+ page handbook on HPSG: Müller, Abeillé, Borsley & Koenig (2021)

Outline

- General remarks on the representational format
- Passive
- Verb position
- Local reordering (aka scrambling)
- Long distance dependencies
- Summary and classification

General remarks on HPSG

- lexicalized (head-driven)

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)
- multiple inheritance

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)
- multiple inheritance
- monostratal theory
  - phonologie
  - syntax
  - semantics

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{Grammatik} \rangle \\ \text{SYNSEM}\mid\text{LOC} & \begin{bmatrix} \textit{loc} & \\ \text{CAT} & \begin{bmatrix} \textit{cat} & \\ \text{HEAD} & \begin{bmatrix} \textit{noun} \\ \text{CASE}\ \boxed{1} \end{bmatrix} \\ \text{SPR} & \left\langle \text{DET}[\text{CASE}\ \boxed{1}] \right\rangle \end{bmatrix} \\ \text{CONT} & \ldots \begin{bmatrix} \textit{grammatik} \\ \text{INST}\ \ X \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)
- multiple inheritance
- monostratal theory

- phonologie
- syntax
- semantics

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{ Grammatik } \rangle \\ \text{SYNSEM}|\text{LOC} & \begin{bmatrix} \textit{loc} & \\ \text{CAT} & \begin{bmatrix} \textit{cat} & \\ \text{HEAD} & \begin{bmatrix} \textit{noun} \\ \text{CASE} \ \boxed{1} \end{bmatrix} \\ \text{SPR} & \left\langle \text{DET}[\text{CASE} \ \boxed{1}] \right\rangle \end{bmatrix} \\ \text{CONT} & \ldots \begin{bmatrix} \textit{grammatik} \\ \text{INST} \ X \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)
- multiple inheritance
- monostratal theory

  - phonologie
  - syntax
  - semantics

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{Grammatik} \rangle \\ \text{SYNSEM}\mid\text{LOC} & \begin{bmatrix} \textit{loc} & \\ \text{CAT} & \begin{bmatrix} \textit{cat} & \\ \text{HEAD} & \begin{bmatrix} \textit{noun} & \\ \text{CASE} & \boxed{1} \end{bmatrix} \\ \text{SPR} & \langle \text{DET}[\text{CASE}\ \boxed{1}] \rangle \end{bmatrix} \\ \text{CONT} & \ldots \begin{bmatrix} \textit{grammatik} & \\ \text{INST} & X \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

General remarks on HPSG

- lexicalized (head-driven)
- sign-based (de Saussure 1916)
- typed feature structures (lexical items, phrases, principles)
- multiple inheritance
- monostratal theory
    - phonologie
    - syntax
    - semantics

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{Grammatik} \rangle \\ \text{SYNSEM|LOC} & \begin{bmatrix} \textit{loc} & \\ \text{CAT} & \begin{bmatrix} \textit{cat} & \\ \text{HEAD} & \begin{bmatrix} \textit{noun} \\ \text{CASE}\ \boxed{1} \end{bmatrix} \\ \text{SPR} & \left\langle \text{DET}[\text{CASE}\ \boxed{1}] \right\rangle \end{bmatrix} \\ \text{CONT} & \ldots \begin{bmatrix} \textit{grammatik} \\ \text{INST}\ X \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Influences

- Categorial Grammar
  (functor-argument structures, valence, argument composition)

Influences

- Categorial Grammar
  (functor-argument structures, valence, argument composition)
- GPSG
  (ID/LP format, Slash mechanism for nonlocal dependencies)

Influences

- Categorial Grammar
  (functor-argument structures, valence, argument composition)
- GPSG
  (ID/LP format, Slash mechanism for nonlocal dependencies)
- Government & Binding
  (for example analysis of verb position in German)

Influences

- Categorial Grammar
  (functor-argument structures, valence, argument composition)
- GPSG
  (ID/LP format, Slash mechanism for nonlocal dependencies)
- Government & Binding
  (for example analysis of verb position in German)
- Construction Grammar
  (increased use of inheritance hierarchies for phrasal aspects, Sag 1997; 2010; 2012)

Valence and grammar rules: PSG

- lage number of rules:

S → NP[*nom*], V	*X schläft* 'X is sleeping'
S → NP[*nom*], NP[*acc*], V	*X Y erwartet* 'X expects Y'
S → NP[*nom*], PP[*über*], V	*X über Y spricht* 'X talks about Y'
S → NP[*nom*], NP[*dat*], NP[*acc*], V	*X Y Z gibt* 'X gives Z to Y'
S → NP[*nom*], NP[*dat*], PP[*mit*], V	*X Y mit Z dient* 'X serves Y with Z'

Valence and grammar rules: PSG

- lage number of rules:
  S → NP[*nom*], V — *X schläft* 'X is sleeping'
  S → NP[*nom*], NP[*acc*], V — *X Y erwartet* 'X expects Y'
  S → NP[*nom*], PP[*über*], V — *X über Y spricht* 'X talks about Y'
  S → NP[*nom*], NP[*dat*], NP[*acc*], V — *X Y Z gibt* 'X gives Z to Y'
  S → NP[*nom*], NP[*dat*], PP[*mit*], V — *X Y mit Z dient* 'X serves Y with Z'
- Verbs have to be used with an appropriate rule.

Valence and grammar rules: HPSG

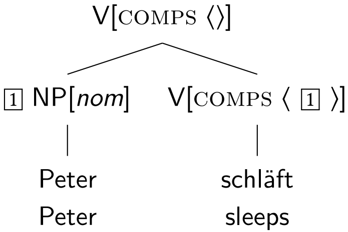
- Arguments are represented as complex categories in the lexical representation of the head (as in Categorial Grammar).

Valence and grammar rules: HPSG

- Arguments are represented as complex categories in the lexical representation of the head (as in Categorial Grammar).
- verb / COMPS

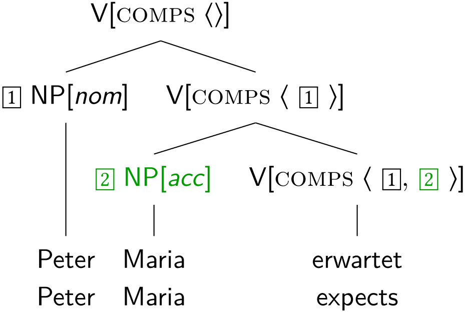
verb	COMPS
schlafen 'to sleep'	$\langle$ NP[*nom*] $\rangle$
erwarten 'to expect'	$\langle$ NP[*nom*], NP[*acc*] $\rangle$
sprechen 'to speak'	$\langle$ NP[*nom*], PP[*über*] $\rangle$
geben 'to give'	$\langle$ NP[*nom*], NP[*dat*], NP[*acc*] $\rangle$
dienen 'to serve'	$\langle$ NP[*nom*], NP[*dat*], PP[*mit*] $\rangle$

Example with valence information: Intransitive verb

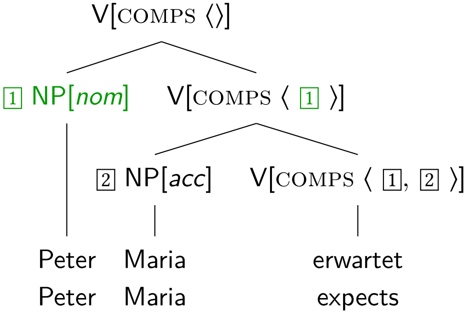


V[COMPS ⟨ ⟩] corresponds to a fully saturated phrase (VP or S)

Example with valence information: Transitive verb



Example with valence information: Transitive verb



SOV vs. SVO: Representation of subjects

- Researchers working on German assume that the subject of finite verbs behaves like the other arguments. (Pollard 1996; Eisenberg 1994: 376)
  HPSG: subjects and complements are listed in one valence list (COMPS).

SOV vs. SVO: Representation of subjects

- Researchers working on German assume that the subject of finite verbs behaves like the other arguments. (Pollard 1996; Eisenberg 1994: 376)
  HPSG: subjects and complements are listed in one valence list (COMPS).
- English: subjects are different.

SOV vs. SVO: Representation of subjects

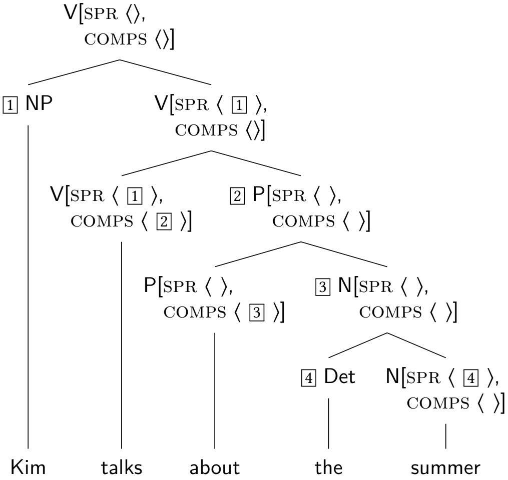
- Researchers working on German assume that the subject of finite verbs behaves like the other arguments. (Pollard 1996; Eisenberg 1994: 376)
  HPSG: subjects and complements are listed in one valence list (COMPS).
- English: subjects are different.
- ARG-ST as a underlying representation containing all arguments.

SOV vs. SVO: Representation of subjects

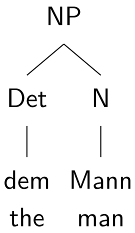
- Researchers working on German assume that the subject of finite verbs behaves like the other arguments. (Pollard 1996; Eisenberg 1994: 376)
  HPSG: subjects and complements are listed in one valence list (COMPS).
- English: subjects are different.
- ARG-ST as a underlying representation containing all arguments.
- Language dependent mapping to valence features SPR and COMPS.

verb	SPR	COMPS	ARG-ST
sleep	$\langle$ NP[*nom*] $\rangle$	$\langle\rangle$	$\langle$ NP[*nom*] $\rangle$
expect	$\langle$ NP[*nom*] $\rangle$	$\langle$ NP[*acc*] $\rangle$	$\langle$ NP[*nom*], NP[*acc*] $\rangle$
speak	$\langle$ NP[*nom*] $\rangle$	$\langle$ PP[*about*] $\rangle$	$\langle$ NP[*nom*], PP[*about*] $\rangle$
give	$\langle$ NP[*nom*] $\rangle$	$\langle$ NP[*acc*], NP[*acc*] $\rangle$	$\langle$ NP[*nom*], NP[*acc*], NP[*acc*] $\rangle$
serve	$\langle$ NP[*nom*] $\rangle$	$\langle$ NP[*acc*], PP[*with*] $\rangle$	$\langle$ NP[*nom*], NP[*acc*], PP[*with*] $\rangle$

Example analysis with SPR and COMPS



Representation of constituent structure



The tree can be represented in feature descriptions:

(213) $$\begin{bmatrix} \text{PHON} & \langle \textit{ dem Mann } \rangle \\ \text{HEAD-DTR} & \begin{bmatrix} \text{PHON} \; \langle \textit{ Mann } \rangle \end{bmatrix} \\ \text{NON-HEAD-DTRS} & \left\langle \begin{bmatrix} \text{PHON} \; \langle \textit{ dem } \rangle \end{bmatrix} \right\rangle \end{bmatrix}$$

Complete feature geometry

(214)

$$
\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{Grammatik} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{LOC} & \begin{bmatrix} \textit{local} & \\ \text{CAT} & \begin{bmatrix} \textit{category} & \\ \text{HEAD} & \begin{bmatrix}\textit{noun} & \\ \text{CASE} & \boxed{1}\end{bmatrix} \\ \text{SPR} & \langle\ \text{DET}[\text{CASE}\ \boxed{1}]\ \rangle \\ \text{COMPS} & \langle\rangle \end{bmatrix} \\ \text{CONT} & \begin{bmatrix} \textit{mrs} & \\ \text{IND} & \boxed{2}\begin{bmatrix}\text{PER} & \textit{third}\\ \text{NUM} & \textit{sg} \\ \text{GEN} & \textit{fem}\end{bmatrix} \\ \text{RELS} & \left\langle \begin{bmatrix}\textit{grammatik} & \\ \text{INST} & \boxed{2}\end{bmatrix}\right\rangle \end{bmatrix}\end{bmatrix} \\ \text{NONLOC} & \begin{bmatrix}\text{INHER|SLASH} & \langle\rangle \\ \text{TO-BIND|SLASH} & \langle\rangle\end{bmatrix}\end{bmatrix}\end{bmatrix}
$$

Information that is needed for structure sharing is grouped together.

The Head-Complement Schema (preliminary)

head-complement-phrase ⇒

$$\begin{bmatrix} \text{SYNSEM|LOC|CAT|COMPS } \boxed{1} \\ \text{HEAD-DTR|SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \langle \boxed{2} \rangle \\ \text{NON-HEAD-DTRS } \left\langle \left[\text{ SYNSEM } \boxed{2} \right] \right\rangle \end{bmatrix}$$

The Head-Complement Schema (preliminary)

head-complement-phrase ⇒

$$\begin{bmatrix} \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \boxed{1} \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \boxed{1} \oplus \langle\ \boxed{2}\ \rangle \\ \text{NON-HEAD-DTRS}\ \Big\langle \big[\ \text{SYNSEM}\ \boxed{2}\ \big] \Big\rangle \end{bmatrix}$$

(215)
$$\begin{bmatrix} \textit{head-complement-phrase} \\ \text{PHON}\ \langle\ \textit{Peter schläft}\ \rangle \\ \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \langle\rangle \\ \text{HEAD-DTR}\ \begin{bmatrix} \text{PHON}\ \langle\ \textit{schläft}\ \rangle \\ \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \langle\ \boxed{1}\ \text{NP}[\textit{nom}]\ \rangle \end{bmatrix} \\ \text{NON-HEAD-DTRS}\ \left\langle \begin{bmatrix} \text{PHON}\ \langle\ \textit{Peter}\ \rangle \\ \text{SYNSEM}\ \boxed{1} \end{bmatrix} \right\rangle \end{bmatrix}$$

Linearization rules

(216) a. Head[INITIAL +] < Complement

b. Complement < Head[INITIAL–]

Linearization rules

(216) a. Head[INITIAL +] < Complement
b. Complement < Head[INITIAL–]

Prepositions have an INITIAL value '+' and therefore have to precede arguments.

(217) a. [in [den Schrank]]
in the cupboard
b. * [[den Schrank] in]
the cupboard in

Linearization rules

(216) a. Head[INITIAL +] < Complement
b. Complement < Head[INITIAL–]

Prepositions have an INITIAL value '+' and therefore have to precede arguments.

(217) a. [in [den Schrank]]
in the cupboard
b. * [[den Schrank] in]
the cupboard in

Verbs in final position bear the value '−' and have to follow their arguments.

(218) a. dass [er [ihn umfüllt]]
that he it decants
b. * dass [er [umfüllt ihn]]
that he decants it

Head features

- Information about verb form has to be present at the top-most node of a projection:

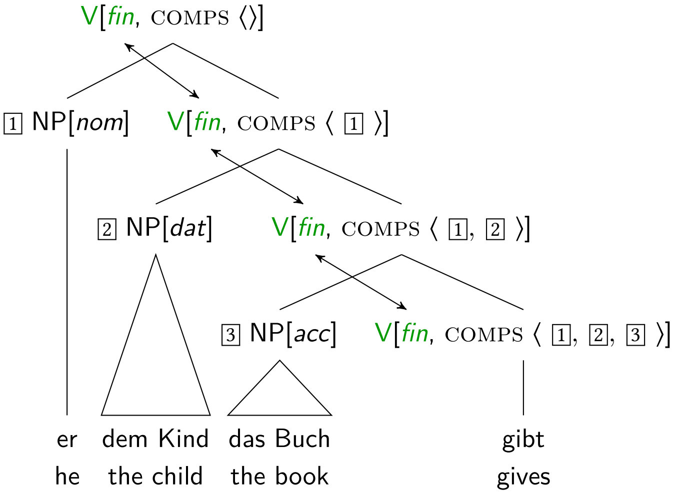
(219) a. [Dem Mann helfen] will er nicht.
the man help wants he not
'He doesn't want to help the man.'

b. [Dem Mann geholfen] hat er nicht.
the man helped has he not
'He hasn't helped the man.'

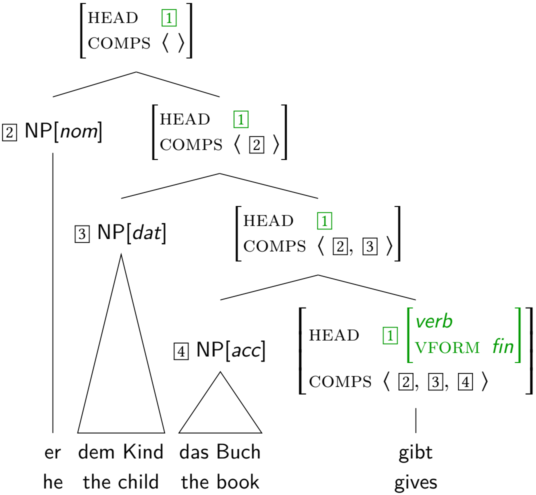
c. * [Dem Mann geholfen] will er nicht.
the man helped wants he not

d. * [Dem Mann helfen] hat er nicht.
the man help has he not

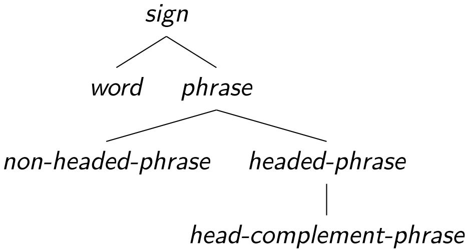
Projection of features along the head path



Structure sharing of HEAD features

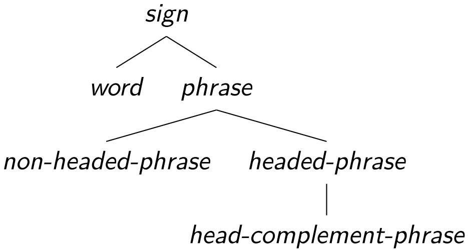


Type hierarchies and inheritance



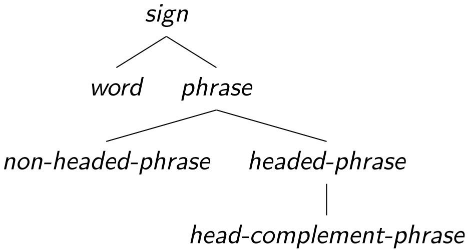
- All feature structures are typed in HPSG.

Type hierarchies and inheritance



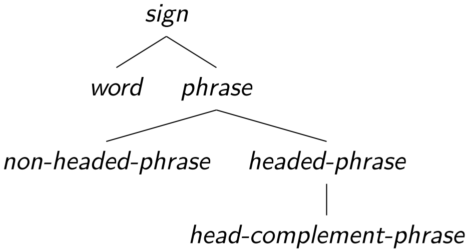
- All feature structures are typed in HPSG.
- Types are ordered in hierarchies.

Type hierarchies and inheritance



- All feature structures are typed in HPSG.
- Types are ordered in hierarchies.
- Subtypes inherit constraints from supertypes.

Type hierarchies and inheritance



- All feature structures are typed in HPSG.
- Types are ordered in hierarchies.
- Subtypes inherit constraints from supertypes.
- Example: *headed-phrase*

(220) *headed-phrase* $\Rightarrow \begin{bmatrix} \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{HEAD}\ \boxed{1} \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOC}|\text{CAT}|\text{HEAD}\ \boxed{1} \end{bmatrix}$

Inheritance of constraints

- (221) Head-Complement Schema + Head Feature Principle:

$$\begin{bmatrix} \textit{head-complement-phrase} \\ \text{SYNSEM|LOC|CAT} \begin{bmatrix} \text{COMPS } \boxed{2} \end{bmatrix} \\ \text{HEAD-DTR|SYNSEM|LOC|CAT} \begin{bmatrix} \text{COMPS } \boxed{2} \oplus \langle \boxed{3} \rangle \end{bmatrix} \\ \text{NON-HEAD-DTRS } \langle \begin{bmatrix} \text{SYNSEM } \boxed{3} \end{bmatrix} \rangle \end{bmatrix}$$

Constraints on *head-complement-phrase*

Inheritance of constraints

- (221) Head-Complement Schema + Head Feature Principle:

$$\begin{bmatrix} \textit{head-complement-phrase} \\ \text{SYNSEM}|\text{LOC}|\text{CAT} \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{COMPS} & \boxed{2} \end{bmatrix} \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOC}|\text{CAT} \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{COMPS} & \boxed{2} \oplus \langle \boxed{3} \rangle \end{bmatrix} \\ \text{NON-HEAD-DTRS} \left\langle \left[\text{SYNSEM}\ \boxed{3} \right] \right\rangle \end{bmatrix}$$

Constraints on *head-complement-phrase* and inherited constraints from *headed-phrase*

Inheritance of constraints

- (221) Head-Complement Schema + Head Feature Principle:

$$\left[\begin{array}{l} \textit{head-complement-phrase} \\ \text{SYNSEM}|\text{LOC}|\text{CAT} \left[\begin{array}{ll} \text{HEAD} & \boxed{1} \\ \text{COMPS} & \boxed{2} \end{array}\right] \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOC}|\text{CAT} \left[\begin{array}{ll} \text{HEAD} & \boxed{1} \\ \text{COMPS} & \boxed{2} \oplus \langle\ \boxed{3}\ \rangle \end{array}\right] \\ \text{NON-HEAD-DTRS} \left\langle \left[\ \text{SYNSEM}\ \boxed{3}\ \right] \right\rangle \end{array}\right]$$

  Constraints on *head-complement-phrase* and inherited constraints from *headed-phrase*

- Inheritance hierarchies are important for capturing generalizations.
  They have been used in the lexicon since Flickinger, Pollard & Wasow 1985.

Outline

- General remarks on the representational format
- Passive
- Verb position
- Local reordering (aka scrambling)
- Long distance dependencies
- Summary and classification

Passive

- HPSG follows Bresnan's argumentation that passive should be treated lexically.

Passive

- HPSG follows Bresnan's argumentation that passive should be treated lexically.
- A lexical rule takes a verb stem as input and licenses a participle form.
  The most prominent argument (the designated argument) is suppressed.

Passive

- HPSG follows Bresnan's argumentation that passive should be treated lexically.
- A lexical rule takes a verb stem as input and licenses a participle form.
  The most prominent argument (the designated argument) is suppressed.
- Since grammatical functions are not parts of the theory,
  mapping principles mapping objects onto subjects are not needed.

Passive

- HPSG follows Bresnan's argumentation that passive should be treated lexically.
- A lexical rule takes a verb stem as input and licenses a participle form.
  The most prominent argument (the designated argument) is suppressed.
- Since grammatical functions are not parts of the theory,
  mapping principles mapping objects onto subjects are not needed.
- But the change of case in passives has to be explained.

Structural and lexical case

- Case depending on the syntactic environment is called structural case. Otherwise the case is lexical case.

Structural and lexical case

- Case depending on the syntactic environment is called structural case. Otherwise the case is lexical case.
- Examples of structural case:

(222) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

Structural and lexical case

- Case depending on the syntactic environment is called structural case. Otherwise the case is lexical case.
- Examples of structural case:

(222) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

b. Der Mann lässt den Installateur kommen.
the man lets the.ACC plumber come
'The man is getting the plumber to come.'

Structural and lexical case

- Case depending on the syntactic environment is called structural case. Otherwise the case is lexical case.
- Examples of structural case:

(222) a. Der Installateur kommt.
the.NOM plumber comes
'The plumber is coming.'

b. Der Mann lässt den Installateur kommen.
the man lets the.ACC plumber come
'The man is getting the plumber to come.'

c. das Kommen des Installateurs
the coming of.the plumber
'the plumber's visit'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(223) a. Judit schlägt den Weltmeister.
Judit beats the.ACC world.champion
'Judit beats the world champion.'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(223) a. Judit schlägt den Weltmeister.
Judit beats the.ACC world.champion
'Judit beats the world champion.'

b. Der Weltmeister wird geschlagen.
the.NOM world.champion is beaten
'The world champion is being beaten.'

Structural case: The object

- Object (accusative in the active) can be realized as nominative and genitive:

(223) a.

Judit	schlägt	den	Weltmeister.
Judit	beats	the.ACC	world.champion

‘Judit beats the world champion.’

b.

Der	Weltmeister	wird	geschlagen.
the.NOM	world.champion	is	beaten

‘The world champion is being beaten.’

c.

das	Schlagen	des	Weltmeisters
the	beating	of.the	world.champion

Lexical case

- genitive depending on the verb is lexical case:
  The case of the genitive object does not change in passivization.

(224) a. Wir gedenken der Opfer.
we remember the.GEN victims

b. Der Opfer wird gedacht.
the.GEN victims are remembered
'The victims are being remembered.'

c. * Die Opfer wird / werden gedacht.
the.NOM victims is are remembered

Lexical case

- genitive depending on the verb is lexical case:
  The case of the genitive object does not change in passivization.

(224) a. Wir gedenken der Opfer.
we remember the.GEN victims

b. Der Opfer wird gedacht.
the.GEN victims are remembered
'The victims are being remembered.'

c. * Die Opfer wird / werden gedacht.
the.NOM victims is are remembered

(224b) = impersonal passive, there is no subject.

Lexical case

- genitive depending on the verb is lexical case:
  The case of the genitive object does not change in passivization.

(224) a. Wir gedenken der Opfer.
we remember the.GEN victims

b. Der Opfer wird gedacht.
the.GEN victims are remembered
'The victims are being remembered.'

c. * Die Opfer wird / werden gedacht.
the.NOM victims is are remembered

(224b) = impersonal passive, there is no subject.

- I count the dative of dative objects of verbs among the lexical cases.
  See Müller 2013b.

Valence information and the Case Principle

Case Principle (simplified)

- The first element with structural case in the argument structure list of a verb receives nominative.
- All other elements in the argument structure list of a verb with structural case receive accusative.
- In nominal environments, elements with structural case are assigned genitive.

Based on Yip, Maling & Jackendoff (1987).

Also works for Icelandic and other Germanic languages and also for Hindi.

Active

prototypical valence lists for finite verbs:

(225) a. *schläft* 'sleeps': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j\ \rangle$
b. *unterstützt* 'supports': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{str}]_k\ \rangle$
c. *hilft* 'helps': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{ldat}]_k\ \rangle$
d. *schenkt* 'gives': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{ldat}]_k,\ \mathrm{NP}[\textit{str}]_l\ \rangle$

str stands for *structural* and *ldat* for lexical dative.

Active

prototypical valence lists for finite verbs:

(225) a. *schläft* 'sleeps': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j\ \rangle$
b. *unterstützt* 'supports': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{str}]_k\ \rangle$
c. *hilft* 'helps': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{ldat}]_k\ \rangle$
d. *schenkt* 'gives': $\textsc{arg-st}\ \langle\ \mathrm{NP}[\textit{str}]_j,\ \mathrm{NP}[\textit{ldat}]_k,\ \mathrm{NP}[\textit{str}]_l\ \rangle$

str stands for *structural* and *ldat* for lexical dative.

The first element of the ARG-ST-Liste with structural case gets nominative.
All others with structural case get accusative.

Passive

(226) a. *schläft* 'sleeps': $\text{ARG-ST}\ \langle\ \text{NP}[\textit{str}]_j\ \rangle$

b. *unterstützt* 'supports': $\text{ARG-ST}\ \langle\ \text{NP}[\textit{str}]_j,\ \text{NP}[\textit{str}]_k\ \rangle$

c. *hilft* 'helps': $\text{ARG-ST}\ \langle\ \text{NP}[\textit{str}]_j,\ \text{NP}[\textit{ldat}]_k\ \rangle$

d. *schenkt* 'gives': $\text{ARG-ST}\ \langle\ \text{NP}[\textit{str}]_j,\ \text{NP}[\textit{ldat}]_k,\ \text{NP}[\textit{str}]_l\ \rangle$

Passive

(226) a. *schläft* 'sleeps': $\text{ARG-ST} \langle \text{NP}[\textit{str}]_j \rangle$
b. *unterstützt* 'supports': $\text{ARG-ST} \langle \text{NP}[\textit{str}]_j, \text{NP}[\textit{str}]_k \rangle$
c. *hilft* 'helps': $\text{ARG-ST} \langle \text{NP}[\textit{str}]_j, \text{NP}[\textit{ldat}]_k \rangle$
d. *schenkt* 'gives': $\text{ARG-ST} \langle \text{NP}[\textit{str}]_j, \text{NP}[\textit{ldat}]_k, \text{NP}[\textit{str}]_l \rangle$

Passivization results in the following ARG-ST lists:

(227) a. *geschlafen* 'slept': $\text{ARG-ST} \langle \rangle$
b. *unterstützt* 'supported': $\text{ARG-ST} \langle \text{NP}[\textit{str}]_k \rangle$
c. *geholfen* 'helped': $\text{ARG-ST} \langle \text{NP}[\textit{ldat}]_k \rangle$
d. *geschenkt* 'given': $\text{ARG-ST} \langle \text{NP}[\textit{ldat}]_k, \text{NP}[\textit{str}]_l \rangle$

Different NP in first position. If it has structural case, it gets nominative.
If the case is not structural it remains as is: lexically specified.

Outline

- General remarks on the representational format
- Passive
- Verb position
- Local reordering (aka scrambling)
- Long distance dependencies
- Summary and classification

Verb position

- Höhle (1997): Finite verbs and complementizers form a natural class:

(228) a. dass [jeder diesen Mann kennt]
that everybody this man knows
'that everybody knows this man'

b. Kennt [jeder diesen Mann _]
knows everybody this man
'Does everybody know this man?'

Verb position

- Höhle (1997): Finite verbs and complementizers form a natural class:

(228) a. dass [jeder diesen Mann kennt]
that everybody this man knows
'that everybody knows this man'

b. Kennt [jeder diesen Mann _]
knows everybody this man
'Does everybody know this man?'

- The complementizer takes a clause with verb-final verb.

Verb position

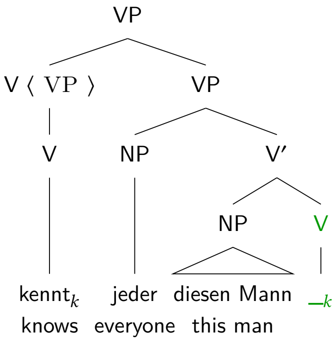
- Höhle (1997): Finite verbs and complementizers form a natural class:

(228) a. dass [jeder diesen Mann kennt]
that everybody this man knows
'that everybody knows this man'

b. Kennt [jeder diesen Mann _]
knows everybody this man
'Does everybody know this man?'

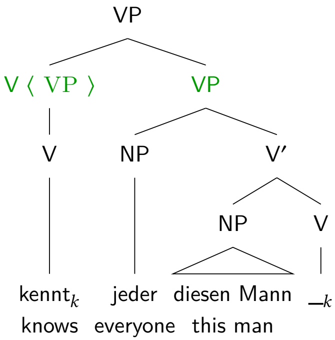
- The complementizer takes a clause with verb-final verb.
- The initial finite verb takes a verb-final clause with the verb at the end missing.

Representations and lexical rules: Verb movement



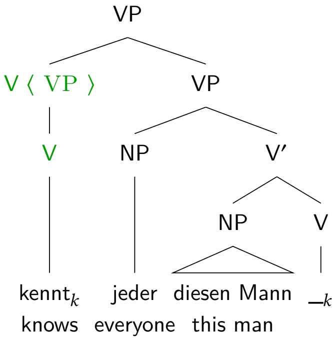
- There is a trace in verb-final position.

Representations and lexical rules: Verb movement



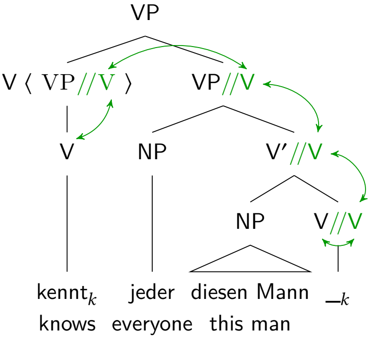
- There is a trace in verb-final position.
- The verb in initial position is a special form of the verb selecting a projection of the verb trace.

Representations and lexical rules: Verb movement



- There is a trace in verb-final position.
- The verb in initial position is a special form of the verb selecting a projection of the verb trace.
- This special lexical item is licensed by a lexical rule.

Representations and lexical rules: Verb movement



- There is a trace in verb-final position.
- The verb in initial position is a special form of the verb selecting a projection of the verb trace.
- This special lexical item is licensed by a lexical rule.
- Connection between verb and trace is done via percolation of information in the tree.

Outline

- General remarks on the representational format
- Passive
- Verb position
- Local reordering (aka scrambling)
- Long distance dependencies
- Summary and classification

Local reordering

- Arguments can appear in almost any order in the German Mittelfeld.

(229) a. [weil] der Mann dem Kind das Buch gibt
because the.NOM man the.DAT child the.ACC book gives
'because the man gives the book to the child'

b. [weil] der Mann das Buch dem Kind gibt
because the.NOM man the.ACC book the.DAT child gives

c. [weil] das Buch der Mann dem Kind gibt
because the.ACC book the.NOM man the.DAT child gives

d. [weil] das Buch dem Kind der Mann gibt
because the.ACC book the.DAT child the.NOM man gives

e. [weil] dem Kind der Mann das Buch gibt
because the.DAT child the.NOM man the.ACC book gives

f. [weil] dem Kind das Buch der Mann gibt
because the.DAT child the.ACC book the.NOM man gives

Local reordering: Two options

Two approaches:

- flat structures like in GPSG

Local reordering: Two options

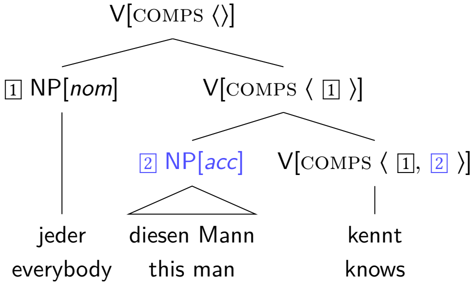
Two approaches:

- flat structures like in GPSG
- binary branching structures with arbitrary order of combination

Example: Normal order (nom, acc)

(230) a. [weil] jeder diesen Mann kennt
because everyone.NOM this.ACC man knows

b. [weil] diesen Mann jeder kennt
because this.ACC man everyone.NOM knows
'because everyone knows this man'



Example: Normal order (nom, acc)

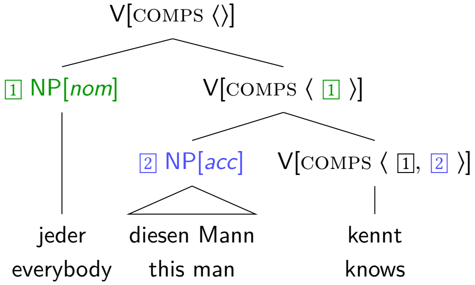
(230) a. [weil] jeder diesen Mann kennt

[weil]	jeder	diesen	Mann	kennt
because	everyone.NOM	this.ACC	man	knows

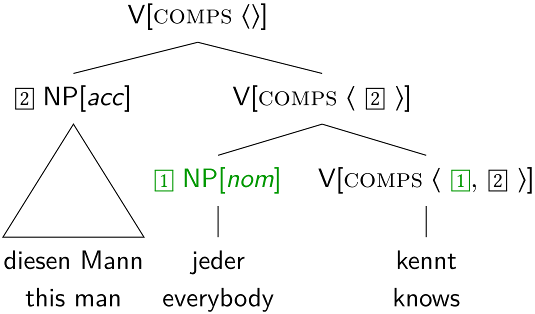
b. [weil] diesen Mann jeder kennt

[weil]	diesen	Mann	jeder	kennt
because	this.ACC	man	everyone.NOM	knows

'because everyone knows this man'

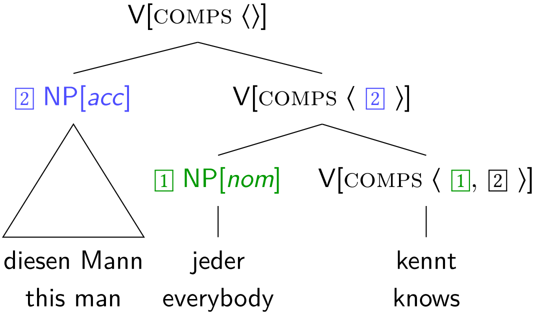


Example: Marked order (acc, nom)



Difference in order of saturation of elements in the COMPS list.

Example: Marked order (acc, nom)



Difference in order of saturation of elements in the COMPS list.

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.
- We permit to take an element from any position of the COMPS list.

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.
- We permit to take an element from any position of the COMPS list.
- We use append to split the list in three parts:
  a beginning, a one-element list, an end

head-complement-phrase ⇒

$$\begin{bmatrix} \text{SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \boxed{3} \\ \text{HEAD-DTR|SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \langle \boxed{2} \rangle \oplus \boxed{3} \\ \text{NON-HEAD-DTRS } \left\langle \left[\text{ SYNSEM } \boxed{2} \right] \right\rangle \end{bmatrix}$$

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.
- We permit to take an element from any position of the COMPS list.
- We use append to split the list in three parts:
  a beginning, a one-element list, an end

  head-complement-phrase ⇒

$$\begin{bmatrix} \text{SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \boxed{3} \\ \text{HEAD-DTR|SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \langle \boxed{2} \rangle \oplus \boxed{3} \\ \text{NON-HEAD-DTRS } \left\langle \left[\text{SYNSEM } \boxed{2} \right] \right\rangle \end{bmatrix}$$

- strict VO: We take arguments from the beginning of the list ($\boxed{1} = \langle\rangle$).

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.
- We permit to take an element from any position of the COMPS list.
- We use append to split the list in three parts:
  a beginning, a one-element list, an end

head-complement-phrase ⇒

$$\begin{bmatrix} \text{SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \boxed{3} \\ \text{HEAD-DTR|SYNSEM|LOC|CAT|COMPS } \boxed{1} \oplus \langle \boxed{2} \rangle \oplus \boxed{3} \\ \text{NON-HEAD-DTRS } \langle \left[\text{ SYNSEM } \boxed{2} \right] \rangle \end{bmatrix}$$

- strict VO: We take arguments from the beginning of the list ($\boxed{1} = \langle\rangle$).
- strict OV: We take arguments from the end of the list ($\boxed{3} = \langle\rangle$).

Generalized Head-Complement Schema

- Earlier version: An element was taken off from the end of the COMPS list.
- We permit to take an element from any position of the COMPS list.
- We use append to split the list in three parts:
  a beginning, a one-element list, an end

head-complement-phrase ⇒

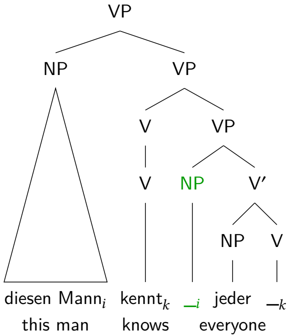
$$\begin{bmatrix} \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \boxed{1} \oplus \boxed{3} \\ \text{HEAD-DTR}|\text{SYNSEM}|\text{LOC}|\text{CAT}|\text{COMPS}\ \boxed{1} \oplus \langle\ \boxed{2}\ \rangle \oplus \boxed{3} \\ \text{NON-HEAD-DTRS}\ \langle\ [\ \text{SYNSEM}\ \boxed{2}\ ]\ \rangle \end{bmatrix}$$

- strict VO: We take arguments from the beginning of the list ($\boxed{1} = \langle\rangle$).
- strict OV: We take arguments from the end of the list ($\boxed{3} = \langle\rangle$).
- VO/OV with scrambling: We take arguments from wherever.

Outline

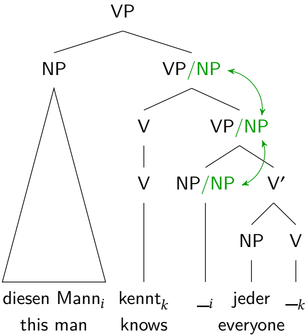
- General remarks on the representational format
- Passive
- Verb position
- Local reordering (aka scrambling)
- Long distance dependencies
- Summary and classification

Long-distance dependencies



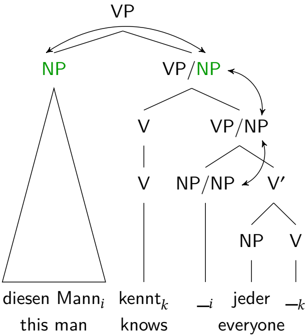
- Like verb movement: Trace in "normal" position.

Long-distance dependencies



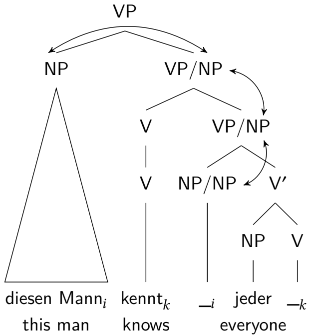
- Like verb movement: Trace in “normal” position.
- Percolation of information in the tree

Long-distance dependencies



- Like verb movement: Trace in "normal" position.
- Percolation of information in the tree
- Binding off nonlocal dependency

Long-distance dependencies



- Like verb movement: Trace in “normal” position.
- Percolation of information in the tree
- Binding off nonlocal dependency
- Constituent movement is not local, verb movement is.
  Hence, two different features are used (SLASH vs. DSL).

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.
- I would say it is a best-of:

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.
- I would say it is a best-of:
  - Linearization from GPSG,

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.
- I would say it is a best-of:
  - Linearization from GPSG,
  - valence from CG,

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.
- I would say it is a best-of:
  - Linearization from GPSG,
  - valence from CG,
  - verb placement (in German) from GB,

Summary

- Carpenter called HPSG a Frankenstein theory (Mineur 1995), since it was sewed together from so many other theories.
- I would say it is a best-of:
  - Linearization from GPSG,
  - valence from CG,
  - verb placement (in German) from GB,
  - constructional patterns from CxG, …

word
ORTH ⟨ *Grammatik* ⟩
SYN|CAT|SUBCAT ⟨ DET ⟩
SEM [IND $\boxed{0}$, RESTR { [*grammar*, INST $\boxed{0}$] }]

word
ORTH ⟨ 语法 ⟩
SYN|CAT|SUBCAT ⟨ DET ⟩
SEM [IND $\boxed{0}$, RESTR { [*grammar*, INST $\boxed{0}$] }]

word
ORTH ⟨ دستور ⟩
SYN|CAT|SUBCAT ⟨ DET ⟩
SEM [IND $\boxed{0}$, RESTR { [*grammar*, INST $\boxed{0}$] }]

word
ORTH ⟨ व्याकरण ⟩
SYN|CAT|SUBCAT ⟨ DET ⟩
SEM [IND $\boxed{0}$, RESTR { [*grammar*, INST $\boxed{0}$] }]

Grammatical theory

Tree Adjoining Grammar (TAG)

Stefan Müller
Institute for German Language and Linguistics, Syntax Lab
Sprach- und literaturwissenschaftliche Fakultät
HU Berlin

St.Mueller@hu-berlin.de

December 10, 2020

Outline

- Introduction and basic terms
- Phrase structure grammar and $\overline{\mathrm{X}}$ Theory
- Government & Binding (GB)
- Generalized Phrase Structure Grammar (GPSG)
- Feature descriptions, feature structures and models
- Lexical Functional Grammar (LFG)
- Categorial Grammar (CG)
- Head-Driven Phrase Structure Grammar (HPSG)
- Tree Adjoining Grammar (TAG)

Reading material

Müller 2020b: Chapter 12.1–12.5 (without 12.1.4 on semantics)

Tree Adjoining Grammar (TAG)

- TAG was developed by Aravind Joshi (University of Pennsylvania).

Tree Adjoining Grammar (TAG)

- TAG was developed by Aravind Joshi (University of Pennsylvania).
- Computational complexity seems to be exactly what is needed for human languages.

Tree Adjoining Grammar (TAG)

- TAG was developed by Aravind Joshi (University of Pennsylvania).
- Computational complexity seems to be exactly what is needed for human languages.
- hotspots: Paris 7 (Anne Abeillé), Columbia University in the USA (Owen Rambow) and Düsseldorf, Germany (Laura Kallmeyer)

Tree Adjoining Grammar (TAG)

- TAG was developed by Aravind Joshi (University of Pennsylvania).
- Computational complexity seems to be exactly what is needed for human languages.
- hotspots: Paris 7 (Anne Abeillé), Columbia University in the USA (Owen Rambow) and Düsseldorf, Germany (Laura Kallmeyer)
- important papers:
  Joshi, Levy & Takahashi 1975; Joshi 1987; Joshi & Schabes 1997

Tree Adjoining Grammar (TAG)

- TAG was developed by Aravind Joshi (University of Pennsylvania).
- Computational complexity seems to be exactly what is needed for human languages.
- hotspots: Paris 7 (Anne Abeillé), Columbia University in the USA (Owen Rambow) and Düsseldorf, Germany (Laura Kallmeyer)
- important papers:
  Joshi, Levy & Takahashi 1975; Joshi 1987; Joshi & Schabes 1997
- on German:
  Rambow 1994, Joshi, Becker & Rambow 2000, Gerdes 2002

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

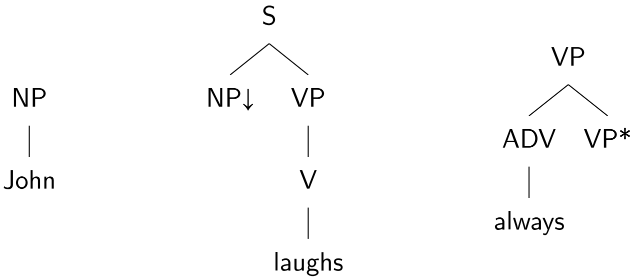
General remarks on representational format

- The basic idea is really simple:
  Every head is paired with a tree in which the head can appear.

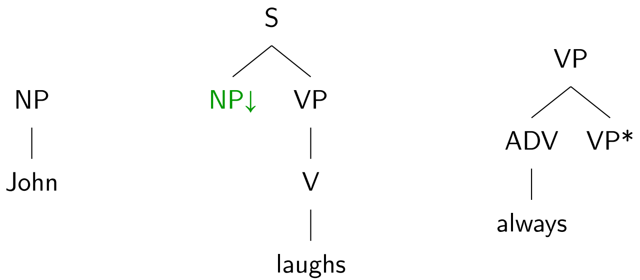
General remarks on representational format

- The basic idea is really simple:
  Every head is paired with a tree in which the head can appear.
- Such trees can be combined with other trees into more complex trees.
  There are two operations: substitution and adjunction.

Elementary Trees

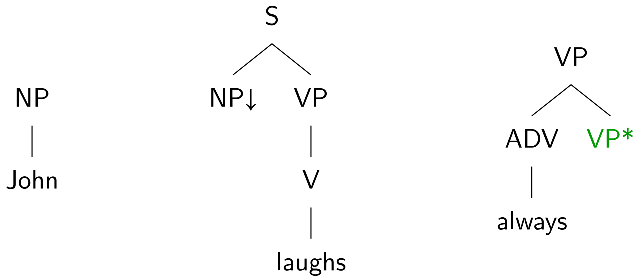


Elementary Trees



Node for inserting arguments are marked with ↓
(NP in the tree of *laughs*).

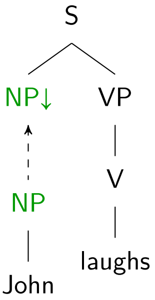
Elementary Trees



Node for inserting arguments are marked with ↓
(NP in the tree of *laughs*).

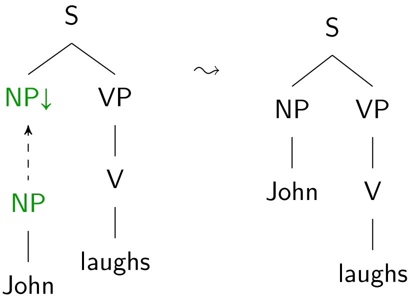
Nodes for inserting adjuncts are marked by '*' (VP in the tree of *always*).

Substitution



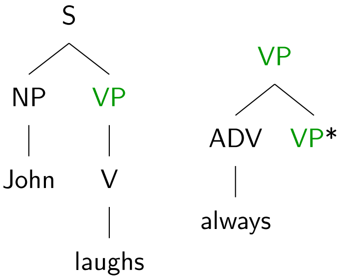
The substitution nodes have to be filled by other trees.

Substitution



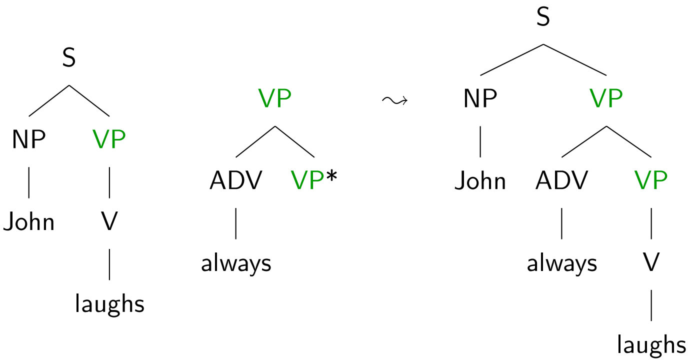
The substitution nodes have to be filled by other trees.

Adjunction



Adjunction trees may be inserted into other trees.

Adjunction



Adjunction trees may be inserted into other trees.

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

Local reordering

- Arguments can appear in almost any order in the German Mittelfeld.

(231) a. [weil] der Mann dem Kind das Buch gibt
because the.NOM man the.DAT child the.ACC book gives
'because the man gives the book to the child'

b. [weil] der Mann das Buch dem Kind gibt
because the.NOM man the.ACC book the.DAT child gives

c. [weil] das Buch der Mann dem Kind gibt
because the.ACC book the.NOM man the.DAT child gives

d. [weil] das Buch dem Kind der Mann gibt
because the.ACC book the.DAT child the.NOM man gives

e. [weil] dem Kind der Mann das Buch gibt
because the.DAT child the.NOM man the.ACC book gives

f. [weil] dem Kind das Buch der Mann gibt
because the.DAT child the.ACC book the.NOM man gives

Option one: Local reordering via lexical rules

- There is a tree family for every word.

Option one: Local reordering via lexical rules

- There is a tree family for every word.
- six trees for a ditransitive verb corresponding to the six possible orders

Option one: Local reordering via lexical rules

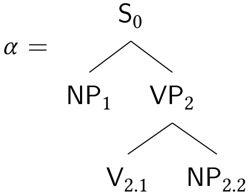
- There is a tree family for every word.
- six trees for a ditransitive verb corresponding to the six possible orders
- Trees are related via lexical rules.

Option one: Local reordering via lexical rules

- There is a tree family for every word.
- six trees for a ditransitive verb corresponding to the six possible orders
- Trees are related via lexical rules.
- This approach is parallel to the one by Uszkoreit (1986) in Categorial Grammar.

Option two: Local Domain/Linear Precedence (LD/LP)

- Joshi, Vijay-Shanker & Weir (1990): linearization rules similar to GPSG/HPSG.



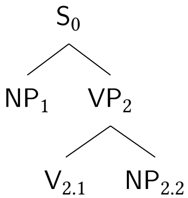
(232) $LP_1^{\alpha} = \{ 1 < 2, 2.1 < 2.2 \}$

- The LP statement in (232) orders the nodes as we need them in English.

Local Domain/Linear Precedence

- empty set of linearization constraints → anything goes.

$\alpha =$



(233) $LP^{\alpha}_{2} = \{\ \}$

(234)
a. NP_1 V NP_2
b. NP_2 V NP_1
c. NP_1 NP_2 V
d. NP_2 NP_1 V
e. V NP_1 NP_2
f. V NP_2 NP_1

- Even though we have a NP-VP structure,
NP_2 can be serialized to the left of NP_1 and NP_1 between V and NP_2.

Verbal complexes

- TAG cannot deal with reorderings when arguments depend on different heads.
- Example of the general pattern:

(235) weil es ihr jemand zu lesen versprochen hat (Haider 1990)
because it her somebody to read promised has
'because somebody promised her to read it'

Verbal complexes

- TAG cannot deal with reorderings when arguments depend on different heads.
- Example of the general pattern:

(235) weil es ihr jemand zu lesen versprochen hat (Haider 1990)
because it her somebody to read promised has
'because somebody promised her to read it'

Verbal complexes

- TAG cannot deal with reorderings when arguments depend on different heads.
- Example of the general pattern:

(235) weil es ihr jemand zu lesen versprochen hat (Haider 1990)
because it her somebody to read promised has
'because somebody promised her to read it'

Verbal complexes

- TAG cannot deal with reorderings when arguments depend on different heads.
- Example of the general pattern:

(235) weil es ihr jemand zu lesen versprochen hat (Haider 1990)
because it her somebody to read promised has
'because somebody promised her to read it'

- TAG cannot deal with sentences having a downstairs argument between the NPs from the upstairs verb:

(236) weil ihr es jemand zu lesen versprochen hat
because her it somebody to read promised has

The trees would have to be merged somehow.

Verbal complexes

- TAG cannot deal with reorderings when arguments depend on different heads.
- Example of the general pattern:

(235)

weil	es	ihr	jemand	zu	lesen	versprochen	hat	(Haider 1990)
because	it	her	somebody	to	read	promised	has	

'because somebody promised her to read it'

- TAG cannot deal with sentences having a downstairs argument between the NPs from the upstairs verb:

(236)

weil	ihr	es	jemand	zu	lesen	versprochen	hat
because	her	it	somebody	to	read	promised	has

The trees would have to be merged somehow.

- The TAG formalism has to be extended for such cases: Multi-Component TAG.

Motivation for Multi-Component TAG

Joshi, Becker & Rambow (2000): Simple LTAGs cannot account for (237b):

(237) a.

… daß	der	Detektiv	dem	Klienten	[den	Verdächtigen
that	the.NOM	detective	the.DAT	client	the.ACC	suspect

des	Verbrechens	zu	überführen]	versprach
the.GEN	crime	to	indict	promised

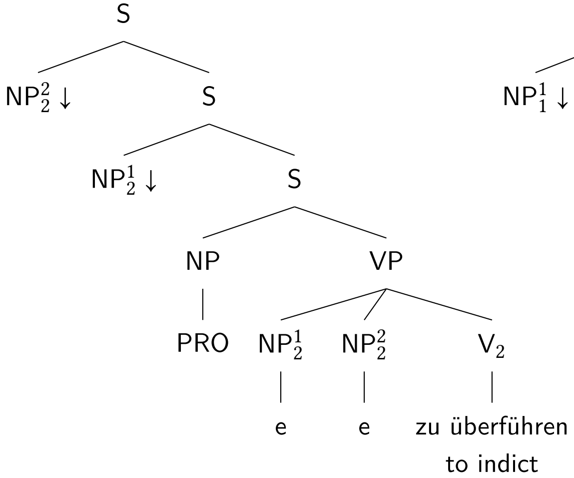
‘that the detective promised the client to indict the suspect of the crime’

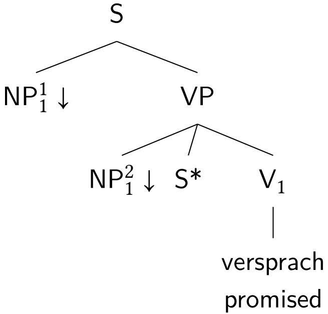
b.

… daß	des	Verbrechens$_k$	der	Detektiv	den
that	the.GEN	crime	the.NOM	detective	the.ACC

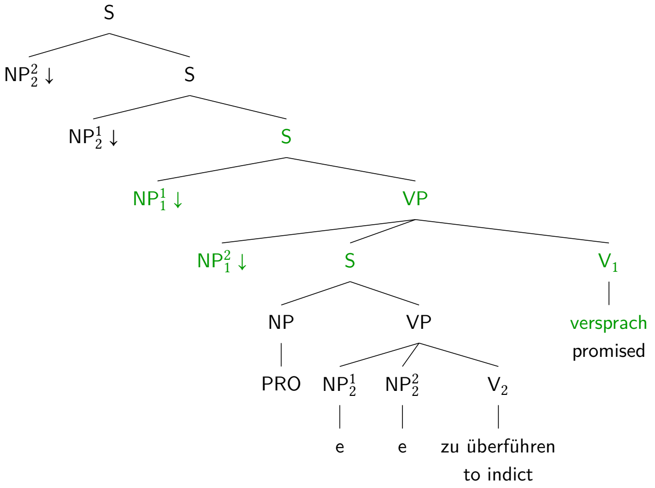
Verdächtigen$_j$	dem	Klienten	[_$_j$ _$_k$	zu	überführen]	versprach
suspect	the.DAT	client		to	indict	promised

Verbal complexes: Elementary trees with moved arguments

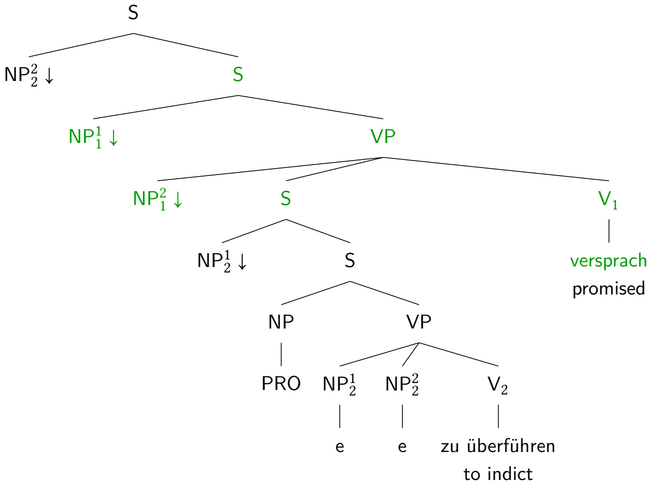




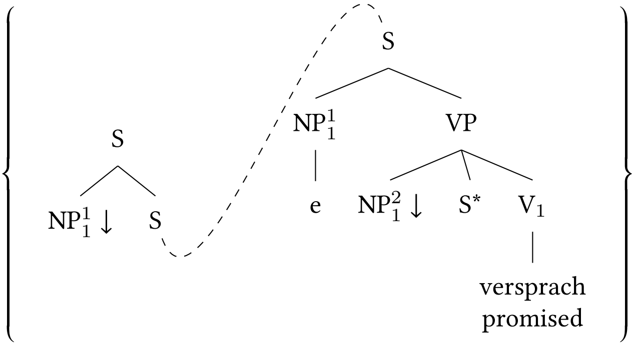
Verbal complexes: Adjunction option I



Verbal complexes: Adjunction option II

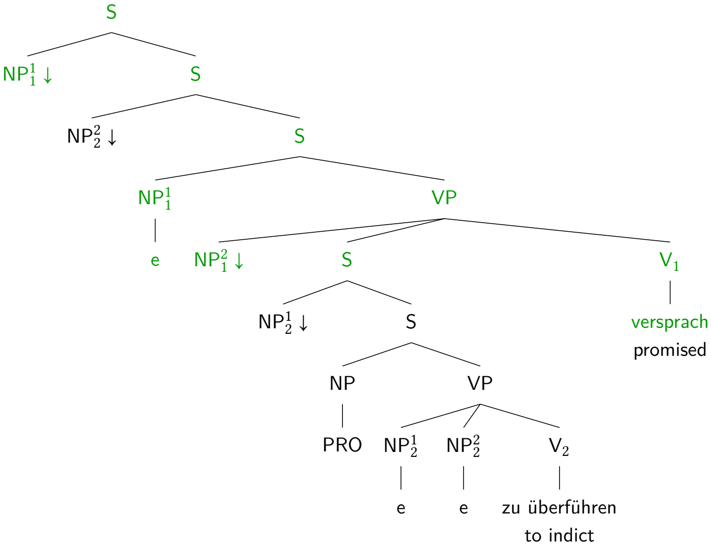


MC lexical item for *versprach* ‘promised’



dashed line: The S with the $NP^1_1 \downarrow$ sister has to dominate the other S node.

There may be other nodes in between.


S
NP$_1^1$ ↓
S
NP$_2^2$ ↓
S
NP$_1^1$
e
VP
NP$_1^2$ ↓
S
V$_1$
versprach
promised
NP$_2^1$ ↓
S
NP
PRO
VP
NP$_2^1$
e
NP$_2^2$
e
V$_2$
zu überführen
to indict

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

Verb position

- Verb position could be analyzed as in GPSG as linearization variant.

Verb position

- Verb position could be analyzed as in GPSG as linearization variant.
- Since verb position is relevant for meaning, a lexical rule-based analysis may be more appropriate:

Verb position

- Verb position could be analyzed as in GPSG as linearization variant.
- Since verb position is relevant for meaning, a lexical rule-based analysis may be more appropriate:
  - There are trees for the verb in initial position and in final position.

Verb position

- Verb position could be analyzed as in GPSG as linearization variant.
- Since verb position is relevant for meaning, a lexical rule-based analysis may be more appropriate:
  - There are trees for the verb in initial position and in final position.
  - The trees are related by lexical rules.

Verb position

- Verb position could be analyzed as in GPSG as linearization variant.
- Since verb position is relevant for meaning, a lexical rule-based analysis may be more appropriate:
  - There are trees for the verb in initial position and in final position.
  - The trees are related by lexical rules.
  - The LRs correspond to transformations in GB:
    A verb-final tree is related to a verb-initial tree.

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

Passive

- There is a family of trees for each word.

Passive

- There is a family of trees for each word.
- For each active tree there is a passive tree.

Passive

- There is a family of trees for each word.
- For each active tree there is a passive tree.
- Trees are related via lexical rules.

Passive

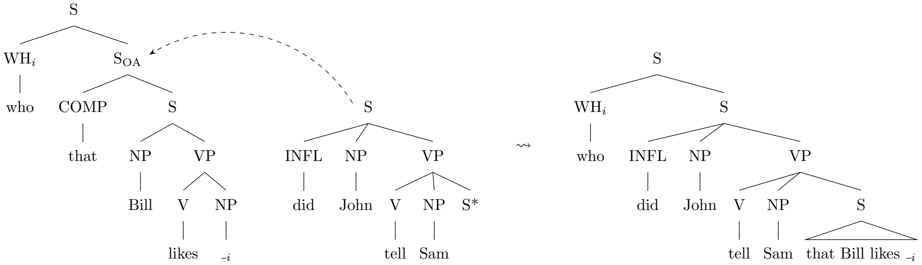
- There is a family of trees for each word.
- For each active tree there is a passive tree.
- Trees are related via lexical rules.
- These lexical rules correspond to transformations of GB mapping trees onto trees.

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

Long-distance dependencies

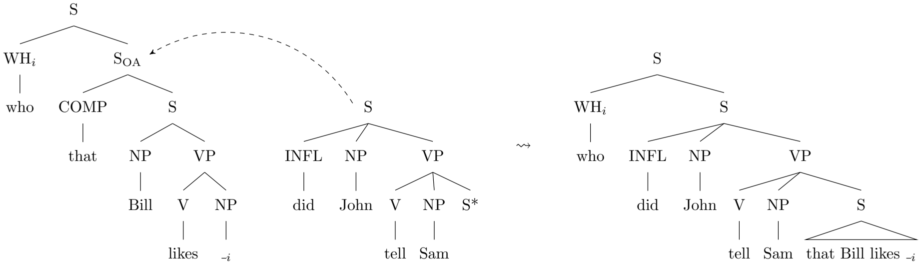
Trees are inserted into the middle of other trees:



(238) a. who$_i$ did John tell Sam that Bill likes _$_i$

Long-distance dependencies

Trees are inserted into the middle of other trees:



(238) a. who$_i$ did John tell Sam that Bill likes _$_i$

b. who$_i$ did John tell Sam that Mary said that Bill likes _$_i$

Obligatory adjunction

- The tree for *WH COMP NP likes* $\_\_{i}$ is a member of the tree family of *likes* and hence listed in the lexicon.
- Although the tree for (239) has the category S,
(239) is not a well-formed sentence in English.

(239) * who that Bill likes

Obligatory adjunction

- The tree for *WH COMP NP likes* $\__i$ is a member of the tree family of *likes* and hence listed in the lexicon.
- Although the tree for (239) has the category S,
  (239) is not a well-formed sentence in English.

  (239) * who that Bill likes

  Label OA: there has to be an obligatory adjunction at respective nodes.

Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

Feature-based TAG: FTAG

- FTAG uses AVMs to describe nodes.

Feature-based TAG: FTAG

- FTAG uses AVMs to describe nodes.
- Every node consists of two parts, a top one and a bottom one.

Feature-based TAG: FTAG

- FTAG uses AVMs to describe nodes.
- Every node consists of two parts, a top one and a bottom one.
- Exception: substitution nodes. They have just a top structure.

Feature-based TAG: FTAG

- FTAG uses AVMs to describe nodes.
- Every node consists of two parts, a top one and a bottom one.
- Exception: substitution nodes. They have just a top structure.
- The upper structure has to match the node into which it is inserted.

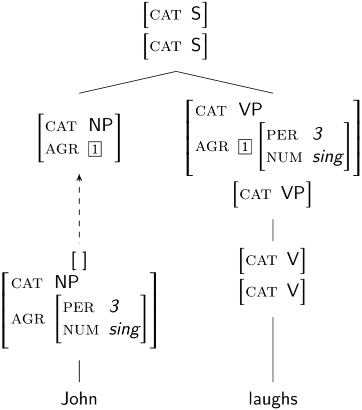
Feature-based TAG: FTAG

- FTAG uses AVMs to describe nodes.
- Every node consists of two parts, a top one and a bottom one.
- Exception: substitution nodes. They have just a top structure.
- The upper structure has to match the node into which it is inserted.
- For adjunction the upper one has to match the upper node into which it is inserted and the lower one the lower node.

Feature-based TAG: FTAG

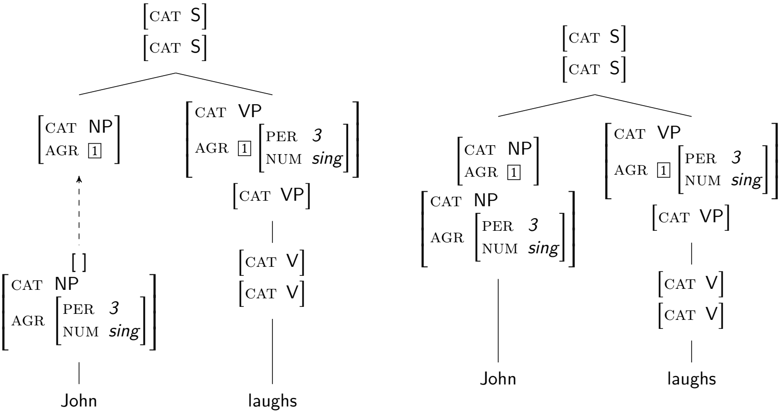
- FTAG uses AVMs to describe nodes.
- Every node consists of two parts, a top one and a bottom one.
- Exception: substitution nodes. They have just a top structure.
- The upper structure has to match the node into which it is inserted.
- For adjunction the upper one has to match the upper node into which it is inserted and the lower one the lower node.
- Pairs are kept till the end of the derivation and then a unification must be possible.

FTAG: Substitution



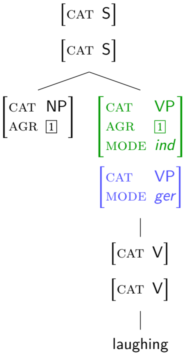
John is inserted into the substitution node

FTAG: Substitution

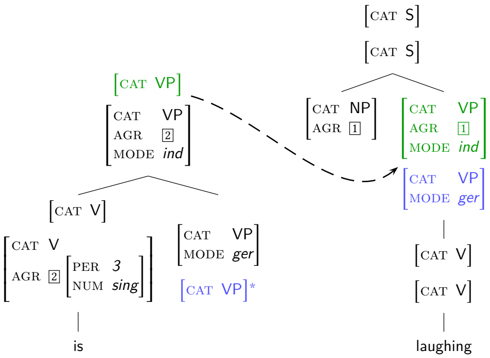


John is inserted into the substitution node and then every top structure has to match every bottom structure.

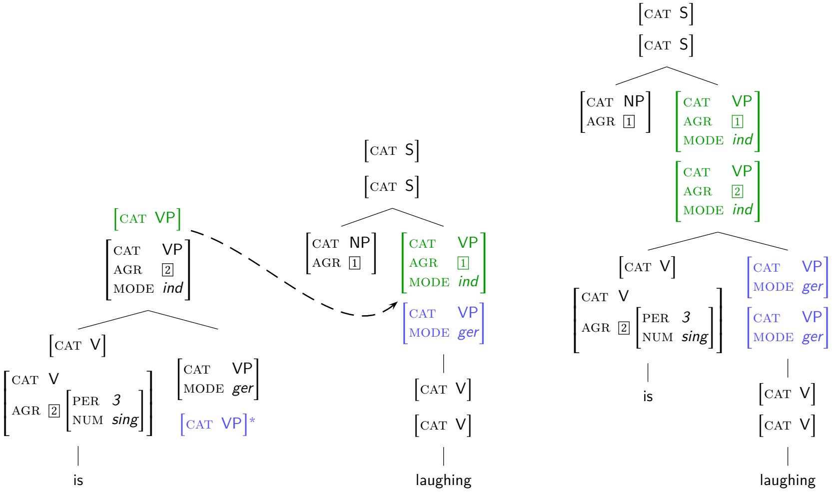
Obligatory adjunction enforced by incompatible features



Obligatory adjunction enforced by incompatible features



Obligatory adjunction enforced by incompatible features

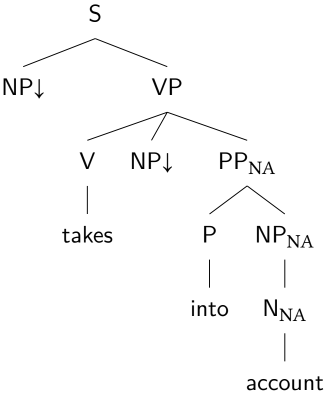


Outline

- General remarks on the representational format
- Local reordering (aka scrambling)
- Verb position
- Passive
- Long distance dependencies
- New developments and theoretical variants
- Summary and classification

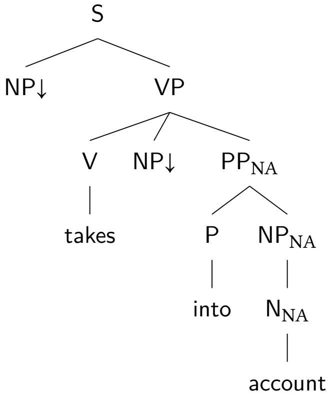
Idioms in TAG

Idioms are really simple (Abeillé & Schabes 1989):



Idioms in TAG

Idioms are really simple (Abeillé & Schabes 1989):



This is the perfect Construction Grammar (and it is lexicalized!)!

Summary

- L-TAG is really simple:
  - lexically anchored trees
  - two combination operations

Summary

- L-TAG is really simple:
  - lexically anchored trees
  - two combination operations
- recursion is filtered out of trees

Summary

- L-TAG is really simple:
  - lexically anchored trees
  - two combination operations
- recursion is filtered out of trees
- no empty elements in the lexicon but in the trees

Summary

- L-TAG is really simple:
  - lexically anchored trees
  - two combination operations
- recursion is filtered out of trees
- no empty elements in the lexicon but in the trees
- various extensions of the core formalism (multi-component, feature-based)

Abeillé, Anne & Yves Schabes. 1989. Parsing idioms in Lexicalized TAG. In Harold Somers & Mary McGee Wood (eds.), *Proceedings of the Fourth Conference of the European Chapter of the Association for Computational Linguistics*, 1–9. Manchester, England: Association for Computational Linguistics.

Ajdukiewicz, Kazimierz. 1935. Die syntaktische Konnexität. *Studia Philosophica* 1. 1–27.

Aoun, Joseph & Dominique Sportiche. 1983. On the formal theory of government. *The Linguistic Review* 2(3). 211–236.

Bargmann, Sascha. 2015. *Syntactically Flexible VP-Idioms and the N-after-N Construction*. Poster presentation at the 5th General Meeting of PARSEME, Iasi, 23–24 September 2015.

Bech, Gunnar. 1955. *Studien über das deutsche Verbum infinitum*. (Linguistische Arbeiten 139). 2. unveränderte Auflage1983. Tübingen: Max Niemeyer Verlag.

Berman, Judith. 1996. Eine LFG-Grammatik des Deutschen. In Judith Berman & Anette Frank (eds.), *Deutsche und französische Syntax im Formalismus der LFG* (Linguistische Arbeiten 344), 11–96. Tübingen: Max Niemeyer Verlag. https://doi.org/10.1515/9783110955354.

Berman, Judith. 2003. *Clausal syntax of German*. (Studies in Constraint-Based Lexicalism). Stanford, CA: CSLI Publications.

Berman, Judith & Anette Frank (eds.). 1996. *Deutsche und französische Syntax im Formalismus der LFG*. (Linguistische Arbeiten 344). Tübingen: Max Niemeyer Verlag. https://doi.org/10.1515/9783110955354.

Bierwisch, Manfred. 1963. *Grammatik des deutschen Verbs*. (studia grammatica 2). Berlin: Akademie Verlag.

Bloomfield, Leonard. 1933. *Language*. London: George Allen & Unwin.

Bresnan, Joan (ed.). 1982a. *The mental representation of grammatical relations*. (MIT Press Series on Cognitive Theory and Mental Representation). Cambridge, MA: MIT Press.

Bresnan, Joan. 1982b. The passive in lexical theory. In Joan Bresnan (ed.), *The mental representation of grammatical relations* (MIT Press Series on Cognitive Theory and Mental Representation), 3–86. Cambridge, MA: MIT Press.

Bresnan, Joan. 2001. *Lexical-Functional Syntax*. 1st edn. Oxford: Blackwell Publishers Ltd.

Bresnan, Joan, Ash Asudeh, Ida Toivonen & Stephen Wechsler. 2016. *Lexical-functional syntax*. 2nd edn. (Blackwell Textbooks in Linguistics 16). Oxford: Wiley-Blackwell. https://doi.org/10.1002/9781119105664.

Bresnan, Joan & Ronald M. Kaplan. 1982. Introduction: grammars as mental representations of language. In Joan Bresnan (ed.), *The mental representation of grammatical relations* (MIT Press Series on Cognitive Theory and Mental Representation), xvii–lii. Cambridge, MA: MIT Press.

Bresnan, Joan & Sam A. Mchombo. 1995. The Lexical Integrity Principle: Evidence from Bantu. *Natural Language & Linguistic Theory* 13. 181–254. https://doi.org/10.1007/BF00992782.

Carpenter, Bob. 1992. *The logic of typed feature structures*. (Cambridge Tracts in Theoretical Computer Science 32). Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9780511530098.

Choi, Hye-Won. 1999. *Optimizing structure in scrambling: Scrambling and information structure*. (Dissertations in Linguistics). Stanford, CA: CSLI Publications.

Chomsky, Noam. 1957. *Syntactic structures*. (Janua Linguarum / Series Minor 4). The Hague: Mouton.

Chomsky, Noam. 1970. Remarks on nominalization. In Roderick A. Jacobs & Peter S. Rosenbaum (eds.), *Readings in English Transformational Grammar*, chap. 12, 184–221. Waltham, MA/Toronto/London: Ginn & Company.

Chomsky, Noam. 1981. *Lectures on government and binding*. (Studies in Generative Grammar 9). Dordrecht: Foris Publications. https://doi.org/10.1515/9783110884166.

Chomsky, Noam. 1986. *Barriers*. (Linguistic Inquiry Monographs 13). Cambridge, MA: MIT Press.

Chomsky, Noam. 1995. *The Minimalist Program*. (Current Studies in Linguistics 28). Cambridge, MA: MIT Press.

Cook, Philippa Helen. 2001. *Coherence in German: An information structure approach*. Departments of Linguistics & German, University of Manchester dissertation.

Culy, Christopher. 1985. The complexity of the vocabulary of Bambara. *Linguistics and Philosophy* 8(3). 345–351. https://doi.org/10.1007/BF00630918.

Dalrymple, Mary. 2006. Lexical Functional Grammar. In Keith Brown (ed.), *The encyclopedia of language and linguistics*, 2nd edn., 82–94. Oxford: Elsevier Science Publisher B.V. (North-Holland).

de Saussure, Ferdinand. 1916. *Grundfragen der allgemeinen Sprachwissenschaft*. 2. Auflage 1967. Berlin: Walter de Gruyter & Co.

Dowty, David R. 1978. Governed transformations as lexical rules in a Montague Grammar. *Linguistic Inquiry* 9(3). 393–426.

Dowty, David R. 1979. *Word meaning and Montague Grammar*. (Synthese Language Library 7). Dordrecht: D. Reidel Publishing Company.

Dowty, David R. 1991. Thematic proto-roles and argument selection. *Language* 67(3). 547–619.

Dowty, David R. 2003. The dual analysis of adjuncts and complements in Categorial Grammar. In Ewald Lang, Claudia Maienborn & Cathrine Fabricius-Hansen (eds.), *Modifying adjuncts* (Interface Explorations 4), 33–66. Berlin: Mouton de Gruyter.

Drach, Erich. 1937. *Grundgedanken der deutschen Satzlehre*. 4., unveränderte Auflage 1963. Darmstadt: Wissenschaftliche Buchgesellschaft.

Eisenberg, Peter. 1994. German. In Ekkehard König & Johan van der Auwera (eds.), *The Germanic languages* (Routledge Language Family Descriptions), 349–387. London: Routledge.

Eisenberg, Peter, Jörg Peters, Peter Gallmann, Cathrine Fabricius-Hansen, Damaris Nübling, Irmhild Barz, Thomas A. Fritz & Reinhard Fiehler. 2005. *Duden: Die Grammatik*. 7th edn. Vol. 4. Mannheim, Leipzig, Wien, Zürich: Dudenverlag.

Eroms, Hans-Werner. 2000. *Syntax der deutschen Sprache*. (de Gruyter Studienbuch). Berlin: Walter de Gruyter Verlag.

Fanselow, Gisbert. 2001. Features, -roles, and free constituent order. *Linguistic Inquiry* 32(3). 405–437.

Fillmore, Charles J. 1968. The case for case. In Emmon Bach & Robert T. Harms (eds.), *Universals of linguistic theory*, 1–88. New York: Holt, Rinehart, & Winston.

Fillmore, Charles J. 1971. Plädoyer für Kasus. In Werner Abraham (ed.), *Kasustheorie* (Schwerpunkte Linguistik und Kommunikationswissenschaft 2), 1–118. Frankfurt/Main: Athenäum.

Fillmore, Charles J., Paul Kay & Mary Catherine O'Connor. 1988. Regularity and idiomaticity in grammatical constructions: The case of *let alone*. *Language* 64(3). 501–538.

Fischer, Kerstin & Anatol Stefanowitsch (eds.). 2006. *Konstruktionsgrammatik: Von der Anwendung zur Theorie*. (Stauffenburg Linguistik 40). Tübingen: Stauffenburg Verlag.

Flickinger, Daniel, Carl J. Pollard & Thomas Wasow. 1985. Structure-sharing in lexical representation. In William C. Mann (ed.), *Proceedings of the Twenty-Third Annual Meeting of the Association for Computational Linguistics*, 262–267. Chicago, IL: Association for Computational Linguistics. https://doi.org/10.3115/981210.981242.

Flickinger, Dan, Tom Wasow & Carl Pollard. 2021. The evolution of HPSG. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook* (Empirically Oriented Theoretical Morphology and Syntax). erscheint. Berlin: Language Science Press.

Frey, Werner. 1993. *Syntaktische Bedingungen für die semantische Interpretation: Über Bindung, implizite Argumente und Skopus*. (studia grammatica 35). Berlin: Akademie Verlag.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry* 12(2). 155–184.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum & Ivan A. Sag. 1985. *Generalized Phrase Structure Grammar*. Cambridge, MA: Harvard University Press.

Gerdes, Kim. 2002. *Topologie et grammaires formelles de l'allemand*. Ecole doctorale Science du langage, UFR de linguistique, Université Paris 7 dissertation.

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar approach to argument structure*. (Cognitive Theory of Language and Culture). Chicago: The University of Chicago Press.

Goldberg, Adele E. 2006. *Constructions at work: The nature of generalization in language*. (Oxford Linguistics). Oxford: Oxford University Press.

Grewendorf, Günther. 1988. *Aspekte der deutschen Syntax: Eine Rektions-Bindungs-Analyse*. (Studien zur deutschen Grammatik 33). Tübingen: original Gunter Narr Verlag jetzt Stauffenburg Verlag.

Grewendorf, Günther. 2002. *Minimalistische Syntax*. (UTB für Wissenschaft: Uni-Taschenbücher 2313). Tübingen, Basel: A. Francke Verlag GmbH.

Gruber, Jeffrey. 1965. *Studies in lexical relations*. MIT dissertation.

Haider, Hubert. 1986. Fehlende Argumente: Vom Passiv zu kohärenten Infinitiven. *Linguistische Berichte* 101. 3–33.

Haider, Hubert. 1990. Pro-bleme? In Gisbert Fanselow & Sascha W. Felix (eds.), *Strukturen und Merkmale syntaktischer Kategorien* (Studien zur deutschen Grammatik 39), 121–143. Tübingen: original Gunter Narr Verlag jetzt Stauffenburg Verlag.

Haider, Hubert. 1993. *Deutsche Syntax – generativ: Vorstudien zur Theorie einer projektiven Grammatik*. (Tübinger Beiträge zur Linguistik 325). Tübingen: Gunter Narr Verlag.

Haider, Hubert. 1997. Projective economy: On the minimal functional structure of the German clause. In Werner Abraham & Elly van Gelderen (eds.), *German: Syntactic problems—Problematic syntax* (Linguistische Arbeiten 374), 83–103. Tübingen: Max Niemeyer Verlag.

Harman, Gilbert. 1963. Generative grammars without transformation rules: A defence of phrase structure. *Language* 39. 597–616.

Heringer, Hans Jürgen. 1996. *Deutsche Syntax dependentiell*. (Stauffenburg Linguistik). Tübingen: Stauffenburg Verlag.

Höhle, Tilman N. 1982. Explikationen für „normale Betonung“ und „normale Wortstellung“. In Werner Abraham (ed.), *Satzglieder im Deutschen – Vorschläge zur syntaktischen, semantischen und pragmatischen Fundierung* (Studien zur deutschen Grammatik 15), 75–153. Republished as Höhle 2018b. Tübingen: original Gunter Narr Verlag jetzt Stauffenburg Verlag.

Höhle, Tilman N. 1986. Der Begriff „Mittelfeld“: Anmerkungen über die Theorie der topologischen Felder. In Walter Weiss, Herbert Ernst Wiegand & Marga Reis (eds.), *Akten des VII. Kongresses der Internationalen Vereinigung für germanische Sprach- und Literaturwissenschaft. Göttingen 1985. Band 3. Textlinguistik contra Stilistik? – Wortschatz und Wörterbuch – Grammatische oder pragmatische Organisation von Rede?* (Kontroversen, alte und neue 4), 329–340. Republished as Höhle 2018a. Tübingen: Max Niemeyer Verlag.

Höhle, Tilman N. 1997. Vorangestellte Verben und Komplementierer sind eine natürliche Klasse. In Christa Dürscheid, Karl Heinz Ramers & Monika Schwarz (eds.), *Sprache im Fokus: Festschrift für Heinz Vater zum 65. Geburtstag*, 107–120. Republished as Höhle 2018e. Tübingen: Max Niemeyer Verlag.

Höhle, Tilman N. 2018a. Der Begriff „Mittelfeld“: Anmerkungen über die Theorie der topologischen Felder. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur deutschen Grammatik: Gesammelte Schriften von Tilman N. Höhle* (Classics in Linguistics 5), 279–294. First published as Höhle 1986. Berlin: Language Science Press. https://doi.org/10.5281/zenodo.1145680.

Höhle, Tilman N. 2018b. Explikationen für „normale Betonung“ und „normale Wortstellung“. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur deutschen Grammatik: Gesammelte Schriften von Tilman N. Höhle* (Classics in Linguistics 5), 107–191. Berlin: Language Science Press. https://doi.org/10.5281/zenodo.1145680.

Höhle, Tilman N. 2018c. Projektionsstufen bei V-Projektionen: Bemerkungen zu F/T. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur deutschen Grammatik: Gesammelte Schriften von Tilman N. Höhle* (Classics in Linguistics 5), 369–379. First circulated in 1991. Berlin: Language Science Press. https://doi.org/10.5281/zenodo.1145680.

Höhle, Tilman N. 2018d. Topologische Felder. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur deutschen Grammatik:*

Gesammelte Schriften von Tilman N. Höhle (Classics in Linguistics 5), 7–89. First circulated as draft in 1983. Berlin: Language Science Press. https://doi.org/10.5281/zenodo.1145680.

Höhle, Tilman N. 2018e. Vorangestellte Verben und Komplementierer sind eine natürliche Klasse. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur deutschen Grammatik: Gesammelte Schriften von Tilman N. Höhle* (Classics in Linguistics 5), 417–433. First published as Höhle 1997. Berlin: Language Science Press. https://doi.org/10.5281/zenodo.1145680.

Hudson, Richard. 2021. HPSG and Dependency Grammar. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook* (Empirically Oriented Theoretical Morphology and Syntax). erscheint. Berlin: Language Science Press.

Jackendoff, Ray S. 1972. *Semantic interpretation in Generative Grammar*. Cambridge, MA: MIT Press.

Jackendoff, Ray S. 1977. *$\overline{X}$ syntax: A study of phrase structure*. Cambridge, MA: MIT Press.

Jackendoff, Ray S. 2008. Construction after Construction and its theoretical challenges. *Language* 84(1). 8–28.

Jacobs, Joachim. 1991. *Bewegung als Valenztransfer*. SFB 282: Theorie des Lexikons 1. Düsseldorf/Wuppertal: Heinrich Heine Uni/BUGH.

Johnson, Mark. 1986. A GPSG account of VP structure in German. *Linguistics* 24(5). 871–882.

Johnson, Mark. 1988. *Attribute-value logic and the theory of grammar*. (CSLI Lecture Notes 16). Stanford, CA: CSLI Publications.

Joshi, Aravind K. 1987. Introduction to Tree Adjoining Grammar. In Alexis Manaster-Ramer (ed.), *The mathematics of language*, 87–114. Amsterdam: John Benjamins Publishing Co.

Joshi, Aravind K., Tilman Becker & Owen Rambow. 2000. Complexity of scrambling: A new twist to the competence-performance distinction. In Anne Abeillé & Owen Rambow (eds.), *Tree Adjoining Grammars: formalisms, linguistic analysis and processing* (CSLI Lecture Notes 156), 167–181. Stanford, CA: CSLI Publications.

Joshi, Aravind K., Leon S. Levy & Masako Takahashi. 1975. Tree Adjunct Grammar. *Journal of Computer and System Science* 10(2). 136–163.

Joshi, Aravind K. & Yves Schabes. 1997. Tree-Adjoning Grammars. In G. Rozenberg & A. Salomaa (eds.), *Handbook of formal languages*, 69–123. Berlin: Springer Verlag.

Joshi, Aravind K., K. Vijay-Shanker & David Weir. 1990. *The Convergence of Mildly Context-Sensitive Grammar Formalisms*. Tech. rep. MS-CIS-90-01. Department of Computer & Information Science, University of Pennsylvania. https://repository.upenn.edu/cis_reports/539/ (18 August, 2020).

Kayne, Richard S. 1984. *Connectedness and binary branching*. (Studies in Generative Grammar 16). Dordrecht: Foris Publications.

King, Paul. 1994. *An Expanded Logical Formalism for Head-Driven Phrase Structure Grammar*. Arbeitspapiere des SFB 340 Nr. 59. Tübingen: Eberhard-Karls-Universität. http://www.sfs.uni-tuebingen.de/sfb/reports/berichte/59/59abs.html (18 August, 2020).

Kiss, Tibor. 2001. Configurational and relational scope determination in German. In Walt Detmar Meurers & Tibor Kiss (eds.), *Constraint-based approaches to Germanic syntax* (Studies in Constraint-Based Lexicalism 7), 141–175. Stanford, CA: CSLI Publications.

Klenk, Ursula. 2003. *Generative Syntax*. (Narr Studienbücher). Tübingen: Gunter Narr Verlag.

Kornai, András & Geoffrey K. Pullum. 1990. The X-bar Theory of phrase structure. *Language* 66(1). 24–50.

Kroch, Anthony S. & Aravind K. Joshi. 1985. *The Linguistic Relevance of Tree Adjoining Grammar*. Tech. rep. MS-CIS-85-16. University of Pennsylvania. http://repository.upenn.edu/cis_reports/671/ (18 August, 2020).

Kunze, Jürgen. 1975. *Abhängigkeitsgrammatik*. (studia grammatica 12). Berlin: Akademie Verlag.

Levine, Robert D. & Walt Detmar Meurers. 2006. Head-Driven Phrase Structure Grammar: Linguistic approach, formal foundations, and computational realization. In Keith Brown (ed.), *The encyclopedia of* *language and linguistics*, 2nd edn., 237–252. Oxford: Elsevier Science Publisher B.V. (North-Holland).

Matsuyama, Tetsuya. 2004. The N *after* N Construction: A constructional idiom. *English Linguistics* 1(11). 55–84.

May, Robert. 1985. *Logical form: Its structure and derivation*. (Linguistic Inquiry Monographs 12). Cambridge, MA: MIT Press.

Mineur, Anne-Marie. 1995. Interview with Bob Carpenter. *Ta!, the Dutch Students' Magazine for Computational Linguistics* 3(1).

Morrill, Glyn. 1995. Discontinuity in Categorial Grammar. *Linguistics and Philosophy* 18(2). 175–219.

Müller, Stefan. 1999. *Deutsche Syntax deklarativ: Head-Driven Phrase Structure Grammar für das Deutsche*. (Linguistische Arbeiten 394). Tübingen: Max Niemeyer Verlag. https://doi.org/10.1515/9783110915990.

Müller, Stefan. 2002. *Complex predicates: Verbal complexes, resultative constructions, and particle verbs in German*. (Studies in Constraint-Based Lexicalism 13). Stanford, CA: CSLI Publications.

Müller, Stefan. 2005. Zur Analyse der deutschen Satzstruktur. *Linguistische Berichte* 201. 3–39.

Müller, Stefan. 2013a. *Grammatiktheorie*. 2nd edn. (Stauffenburg Einführungen 20). Tübingen: Stauffenburg Verlag. https://hpsg.hu-berlin.de/~stefan/Pub/grammatiktheorie.html (30 June, 2019).

Müller, Stefan. 2013b. *Head-Driven Phrase Structure Grammar: Eine Einführung*. 3rd edn. (Stauffenburg Einführungen 17). Tübingen: Stauffenburg Verlag. https://hpsg.hu-berlin.de/~stefan/Pub/hpsg-lehrbuch.html (30 June, 2019).

Müller, Stefan. 2014. Artenvielfalt und Head-Driven Phrase Structure Grammar. In Jörg Hagemann & Sven Staffeldt (eds.), *Syntaxtheorien: Analysen im Vergleich* (Stauffenburg Einführungen 28), 187–233. Tübingen: Stauffenburg Verlag.

Müller, Stefan. 2019. Evaluating theories: Counting nodes and the question of constituency. *Language Under Discussion* 5(1). 52–67. https://doi.org/10.31885/lud.5.1.226.

Müller, Stefan. 2020a. *Germanic Syntax*. Ms. Humboldt Universität zu Berlin, to be submitted to Language Science Press. https://hpsg.hu-berlin.de/~stefan/Pub/germanic.html (30 June, 2019).

Müller, Stefan. 2020b. *Grammatical theory: From Transformational Grammar to constraint-based approaches*. 4th edn. (Textbooks in Language Sciences 1). Berlin: Language Science Press. https://doi.org/10.5281/zenodo.3992307.

Müller, Stefan, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.). 2021. *Head-Driven Phrase Structure Grammar: The handbook*. (Empirically Oriented Theoretical Morphology and Syntax). erscheint. Berlin: Language Science Press.

Müller, Stefan & Antonio Machicao y Priemer. 2019. Head-Driven Phrase Structure Grammar. In András Kertész, Edith Moravcsik & Csilla Rákosi (eds.), *Current approaches to syntax: A comparative handbook* (Comparative Handbooks of Linguistics 3), 317–360. Berlin: Mouton de Gruyter. https://doi.org/10.1515/9783110540253-012.

Müller, Stefan & Stephen Wechsler. 2014. Lexical approaches to argument structure. *Theoretical Linguistics* 40(1–2). 1–76. https://doi.org/10.1515/tl-2014-0001.

Muysken, Pieter. 1982. Parametrizing the notion of "head". *Journal of Linguistic Research* 2. 57–75.

Nerbonne, John. 1986. 'Phantoms' and German fronting: Poltergeist constituents? *Linguistics* 24(5). 857–870. https://doi.org/10.1515/ling.1986.24.5.857.

Netter, Klaus. 1992. On non-head non-movement: An HPSG treatment of finite verb position in German. In Günther Görz (ed.), *Konvens 92. 1. Konferenz „Verarbeitung natürlicher Sprache". Nürnberg 7.–9. Oktober 1992* (Informatik aktuell), 218–227. Berlin: Springer Verlag.

Ørsnes, Bjarne. 2009. Das Verbalfeldmodell: Ein Stellungsfeldermodell für den kontrastiven DaF-Unterricht. *Deutsch als Fremdsprache* 46(3). 143–149.

Peters, Stanley & R. W. Ritchie. 1973. On the generative power of Transformational Grammar. *Information Sciences* 6(C). 49–83. https://doi.org/10.1016/0020-0255(73)90027-3.

Pollard, Carl J. 1988. Categorial Grammar and Phrase Structure Grammar: An excursion on the syntax-semantics frontier. In Richard Oehrle, Emmon Bach & Deirdre Wheeler (eds.), *Categorial Grammars and natural language structures*, 391–415. Dordrecht: D. Reidel Publishing Company.

Pollard, Carl J. 1996. On head non-movement. In Harry Bunt & Arthur van Horck (eds.), *Discontinuous constituency* (Natural Language Processing 6), 279–305. Published version of a Ms. dated January 1990. Berlin: Mouton de Gruyter.

Pollard, Carl & Ivan A. Sag. 1987. *Information-based syntax and semantics*. (CSLI Lecture Notes 13). Stanford, CA: CSLI Publications.

Pollard, Carl & Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. (Studies in Contemporary Linguistics). Chicago: The University of Chicago Press.

Pollock, Jean-Yves. 1989. Verb movement, Universal Grammar and the structure of IP. *Linguistic Inquiry* 20(3). 365–424.

Pullum, Geoffrey K. 1985. Assuming some version of X-bar Theory. In *Papers from the 21st Annual Meeting of the Chicago Linguistic Society*, 323–353.

Pullum, Geoffrey K. 1986. Footloose and context-free. *Natural Language & Linguistic Theory* 4(3). 409–414.

Rambow, Owen. 1994. *Formal and computational aspects of natural language syntax*. University of Pennsylvania dissertation.

Reis, Marga. 1980. On justifying topological frames: 'Positional field' and the order of nonverbal constituents in German. *Documentation et Recherche en Linguistique Allemande Contemporaine. Revue de Linguistique* 22/23. 59–85.

Richter, Frank. 2004. *A mathematical formalism for linguistic theories with an application in Head-Driven Phrase Structure Grammar*. Universität Tübingen Phil. Dissertation (2000). https://publikationen.uni-tuebingen.de/xmlui/handle/10900/46230 (18 August, 2020).

Richter, Frank. 2021. Formal background. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook* (Empirically Oriented Theoretical Morphology and Syntax). erscheint. Berlin: Language Science Press.

Ross, John Robert. 1967. *Constraints on variables in syntax*. Reproduced by the Indiana University Linguistics Club and later published as Ross 1986. Cambridge, MA: MIT dissertation. http://files.eric.ed.gov/fulltext/ED016965.pdf (18 August, 2020).

Ross, John Robert. 1986. *Infinite syntax!* (Language and Being 5). Norwood, NJ: Ablex Publishing Corporation.

Sag, Ivan A. 1997. English relative clause constructions. *Journal of Linguistics* 33(2). 431–484. https://doi.org/10.1017/S002222679700652X.

Sag, Ivan A. 2010. English filler-gap constructions. *Language* 86(3). 486–545.

Sag, Ivan A. 2012. Sign-Based Construction Grammar: An informal synopsis. In Hans C. Boas & Ivan A. Sag (eds.), *Sign-Based Construction Grammar* (CSLI Lecture Notes 193), 69–202. Stanford, CA: CSLI Publications.

Sauerland, Uli & Paul Elbourne. 2002. Total reconstruction, PF movement, and derivational order. *Linguistic Inquiry* 33(2). 283–319.

Scherpenisse, Wim. 1986. *The connection between base structure and linearization restrictions in German and Dutch*. (Europäische Hochschulschriften, Reihe XXI, Linguistik 47). Frankfurt/M.: Peter Lang.

Shieber, Stuart M. 1985. Evidence against the context-freeness of natural language. *Linguistics and Philosophy* 8(3). 333–343. https://doi.org/10.1007/BF00630917.

Shieber, Stuart M. 1986. *An introduction to unification-based approaches to grammar*. (CSLI Lecture Notes 4). Stanford, CA: CSLI Publications.

Steedman, Mark. 1989. Constituency and coordination in a Combinatory Grammar. In Mark R. Baltin & Anthony S. Kroch (eds.), *Alternative conceptions of phrase structure*, 201–231. Chicago/London: The University of Chicago Press.

Steedman, Mark. 1991. Structure and intonation. *Language* 67(2). 260–296. https://doi.org/10.2307/415107.

Steedman, Mark. 1997. *Surface structure and interpretation*. (Linguistic Inquiry Monographs 30). Cambridge, MA: MIT Press.

Steedman, Mark. 2000. *The syntactic process*. (Language, Speech, and Communication). Cambridge, MA: MIT Press.

Steedman, Mark & Jason Baldridge. 2006. Combinatory Categorial Grammar. In Keith Brown (ed.), *Encyclopedia of language and linguistics*, 2nd edn., 610–621. Oxford: Elsevier.

Sternefeld, Wolfgang. 1991. *Syntaktische Grenzen*. Opladen: Westdeutscher Verlag.

Tesnière, Lucien. 1959. *Eléments de syntaxe structurale*. Paris: Librairie C. Klincksieck. Republished as *Elements of Structural Syntax*. 2015. Translated by Timothy Osborne and Sylvain Kahane. Amsterdam: John Benjamins Publishing Co., 2015.

Tesnière, Lucien. 1980. *Grundzüge der strukturalen Syntax*. Translated by Ulrich Engel. Stuttgart: Klett-Cotta.

Tesnière, Lucien. 2015. *Elements of structural syntax*. Translated by Timothy Osborne and Sylvain Kahane. Amsterdam: John Benjamins Publishing Co.

Uszkoreit, Hans. 1986. *Linear Precedence in Discontinuous Constituents: Complex Fronting in German*. Report No. CSLI-86-47. Stanford, CA: Center for the Study of Language & Information.

Uszkoreit, Hans. 1987. *Word order and constituent structure in German*. (CSLI Lecture Notes 8). Stanford, CA: CSLI Publications.

von Stechow, Arnim & Wolfgang Sternefeld. 1988. *Bausteine syntaktischen Wissens: Ein Lehrbuch der Generativen Grammatik*. Opladen/Wiesbaden: Westdeutscher Verlag.

Weber, Heinz J. 1997. *Dependenzgrammatik: Ein interaktives Arbeitsbuch*. 2nd edn. (Narr Studienbücher). Tübingen: Gunter Narr Verlag.

Yip, Moira, Joan Maling & Ray S. Jackendoff. 1987. Case in tiers. *Language* 63(2). 217–250. https://doi.org/10.2307/415655.