EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Syntax & Semantics WiSe 2022/2023

Lecture 7: Phrase Structure Grammar (PSG) II

17/11/2022, Christian Bentz

Overview

Section 1: Recap of Lecture 6

Symbols: Terminals

We firstly define a finite set of so-called **terminal symbols** (T). We here assume that these are words[1] in the respective language we are analyzing:

$$T = \{a, book, child, reads, the, \dots\}^2 \quad (1)$$

[1] Words are typically assumed as terminals for the analysis of natural language, but note that we could also choose morphemes, syllables, characters, etc.

[2] I here order them alphabetically, but note that the order in a set does not matter.

Symbols: Non-Terminals

Based on the definitions of constituency and parts of speech – as laid out in previous lectures – we can also define a finite set of so-called **non-terminal symbols** (NT).

We here assume that these consist of symbols for phrases (e.g. NP, VP, AP, etc.), parts of speech (N, V, A, etc.), as well as the starting symbol S.[3] We such arrive at:

$$NT = \{NP, VP, AP, \ldots N, V, A, \ldots S\} \tag{2}$$

[3] A glossary of all symbols used here is given at the end of this section.

Grammar in Formal Language Theory

A **grammar** $\mathcal{G}$ in formal language theory is then a quadruple consisting of the set of terminal symbols, non-terminal symbols, a starting symbol S, and a set of rewrite rules R:

$$\langle T, NT, S, R \rangle^4 \tag{3}$$

Jäger and Rogers (2012). Formal language theory: refining the Chomsky hierarchy.
Partee et al. (1990). Mathematical methods in linguistics.

[4] S is a "distinguished member" of NT.

Rewrite	Rule
S	
NP V NP	6
DET N V NP	7
DET N V DET N	7
DET N reads DET N	5
the N reads DET N	1
the child reads DET N	3
the child reads a N	2
the child reads a book	4

Terminals

$T = \{a, book, child, reads, the\}$

Non-Terminals

$NT = \{DET, N, NP, V\}$

R (Terminals)

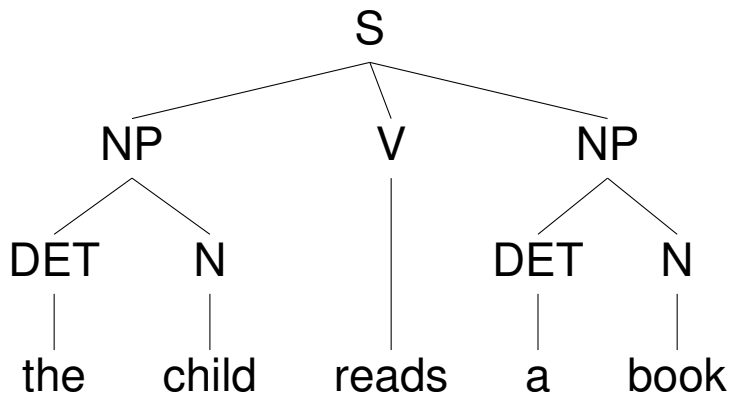
1. DET → the
2. DET → a
3. N → child
4. N → book
5. V → reads

R (Non-Terminals)

6. S → NP V NP
7. NP → DET N

Note: The horizontal line indicates the point where rules exclusively defined with non-terminals ($R(NT)$) end, and rules involving terminals ($R(T)$) start. While the order of application of non-terminal rules is often important, the order of the application of terminal rules is irrelevant.

Bracket Notation



Rewrite Notation

S
NP V NP
DET N V NP
DET N V DET N

DET N reads DET N
the N reads DET N
the child reads DET N
the child reads a N
the child reads a book

[S [NP [DET [the]][N [child]]][V [reads]][NP [DET [a]][N [book]]]][5]

[5]Note: The *Bracket Notation* is yet another equivalent way to visualize the same structure. In fact, the latex code generating this slide takes the bracket notation as input to generate the above tree. There is also an online tool at ironcreek.net/syntaxtree to generate trees based on bracket notation input.

Important Take-Home-Message

One of the most important features of PSGs is that they strongly **restrict the number of possible sentences** via *linearization constraints* in the *non-terminal rules* (inner parts of the tree). The sentences generated by the PSG above are in fact a small subset of the overall possible sentences without any linearization constraints, namely, 4 out of $5! = 120$, or around 3%.

Sentences licensed by PSG:

the child reads a book
a child reads the book
the book reads a child
a book reads the child

Possible permutations:

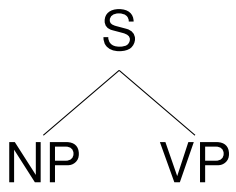
the child reads a book
**book the child reads a*
**a book the child reads*
**reads a book the child*
**child reads a book the*
etc.

Bifurcation

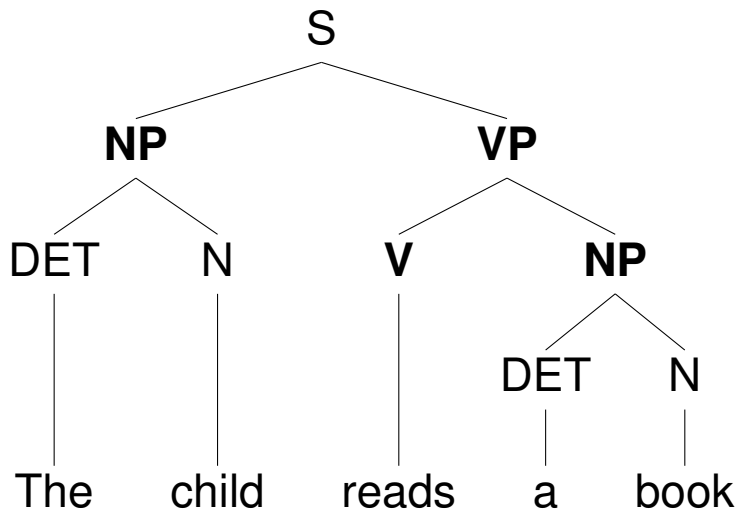
In order to restrict PSGs to a set of simpler (i.e. shorter rules), many frameworks introduce a **binarization constraint**, such that all rewrite rules have only *one symbol* on the left, and maximally *two symbols* on the right. For example,

$$S \rightarrow NP\ VP. \tag{4}$$

This yields exclusively *bifurcating* branches in the tree (except for the terminal nodes):



Tree Notation



Rewrite Notation

S
NP VP
NP V NP
DET N V NP
DET N V DET N

DET N reads DET N
the N reads DET N
the child reads DET N
the child reads a N
the child reads a book

Note: If we wanted the tree to reflect the assumption that the finite verb heads the overall sentence, then we could further introduce $S \rightarrow VP$ and then $VP \rightarrow NP\ VP$.

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Section 2: Morphological Features

Expanding the PSG: The Vocabulary

We can expand our PSG towards covering more of the grammatical sentences in actual English by simply adding terminal symbols, e.g. other two-place predicates (*sees*) and nouns (*tree, frog*).

Sentences licensed by PSG:

the child reads a book
the child sees a book
the child sees a tree
the frog sees a tree
etc.

Note: We will quickly run into the problem of semantics: *?The child reads a frog*. This is the point where *Chomsky's colourless green ideas* come into the picture. PSGs are geared towards grammatical licensing, regardless of semantics.

Expanding the PSG: Morphology

In order to also implement agreement between verbs, nouns and determiners, we have to expand the PSG by using morphological features.

License:

the child reads a book
the children read a book
a child reads the books
etc.

Do not license:

**the child read a book*
**the children reads a book*
**the child reads a books*
etc.

First Step: Expand the Terminals

Terminals

$$T = \{a, book, books, child, children, read, reads, the\}$$

Second Step: Expand the Non-Terminals

Non-Terminals

Morphological features are here given in parentheses '()', and in upper case notation according to the Leipzig Glossing Rules.

$$NT = \{DET(SG), DET(PL), N(SG), N(PL), NP(SG), NP(PL), V(SG), V(PL), VP(SG), VP(PL)\} \quad (5)$$

Third Step: Change Rewrite Rules

R (involving **terminal** symbols)

1. DET(SG) → the
2. DET(SG) → a
3. DET(PL) → the
4. N(SG) → child
5. N(SG) → book
6. N(PL) → children
7. N(PL) → books
8. V(SG) → reads
9. V(PL) → read

R (only **non-terminal** symbols)

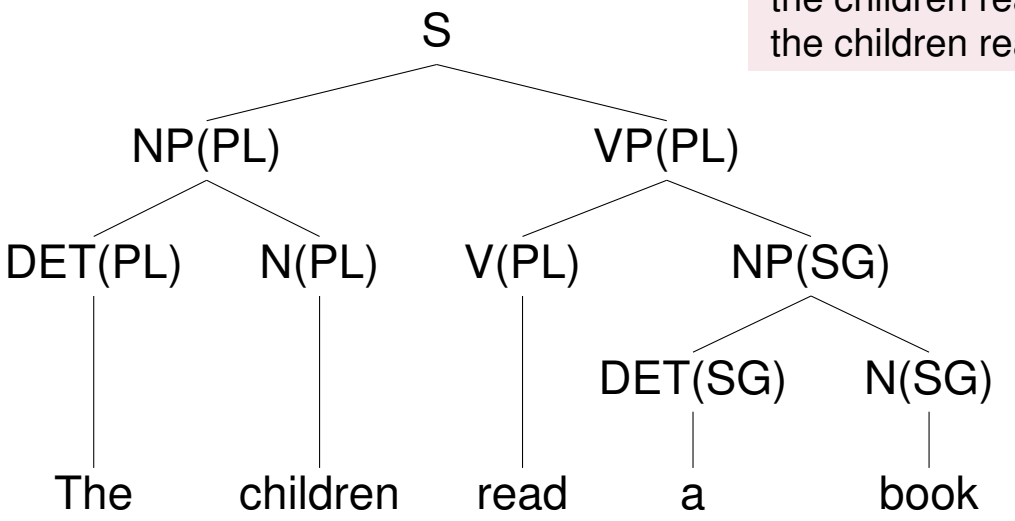
10. S → NP(SG) VP(SG)
11. S → NP(PL) VP(PL)
12. NP(SG) → DET(SG) N(SG)
13. NP(PL) → DET(PL) N(PL)
14. VP(SG) → V(SG) N(SG)
15. VP(SG) → V(SG) N(PL)
16. VP(PL) → V(PL) N(SG)
17. VP(PL) → V(PL) N(PL)

Tree Notation

Rewrite Notation

S
NP(PL) VP(PL)
NP(PL) V(PL) NP(SG)
DET(PL) N(PL) V(PL) NP(SG)
DET(PL) N(PL) V(PL) DET(SG) N(SG)

DET(PL) N(PL) read DET(SG) N(SG)
the N(PL) read DET(SG) N(SG)
the children read DET(SG) N(SG)
the children read a N(SG)
the children read a book



Problem: Complicated Agreement Systems

"The defining characteristic of gender is **agreement**: a language has a gender system only if we find different agreements ultimately dependent on nouns of different types. In other words, there must be evidence for gender outside the nouns themselves."

Corbett (2013). Number of Genders.

Russian (rus, Indo-European)

(1) Žurnal ležal na stole.
magazine.M lay.**M** on table
"The magazine lay on the table."

(2) Kniga ležal**-a** na stole.
book.F lay-**F** on table
"The book lay on the table."

Feature 30A: Number of Genders

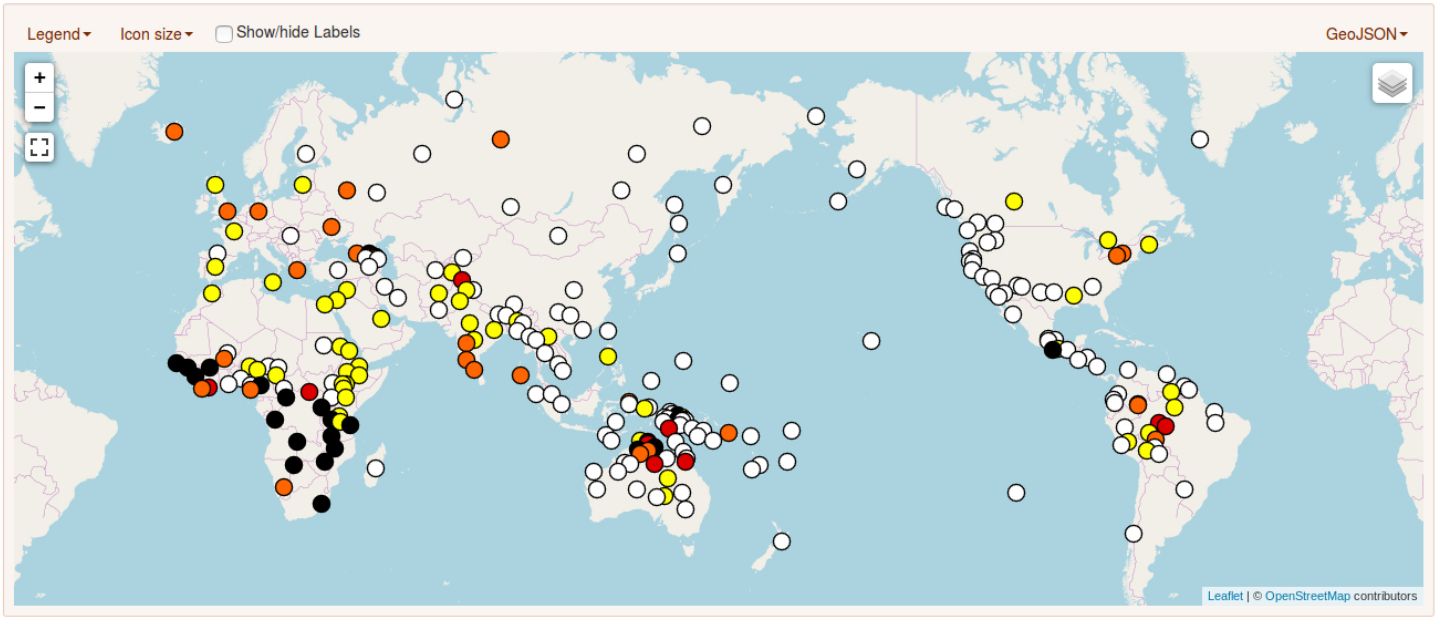
This feature is described in the text of chapter 30 Number of Genders by Greville G. Corbett cite

You may combine this feature with another one. Start typing the feature name or number in the field below.

× 30A: Number of Genders Submit

Values

○	None	145
○	Two	50
○	Three	26
○	Four	12
●	Five or more	24

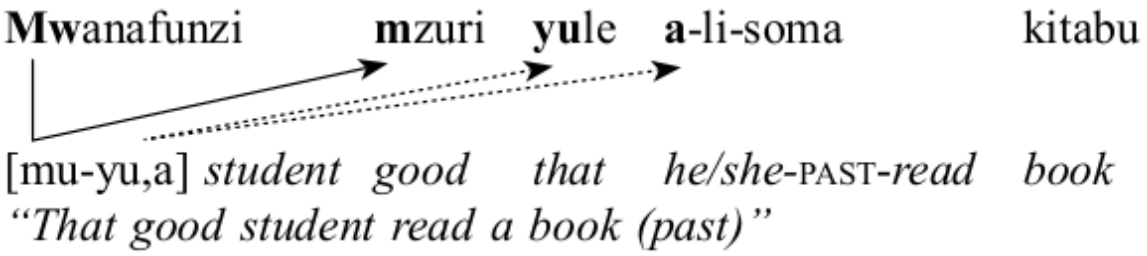


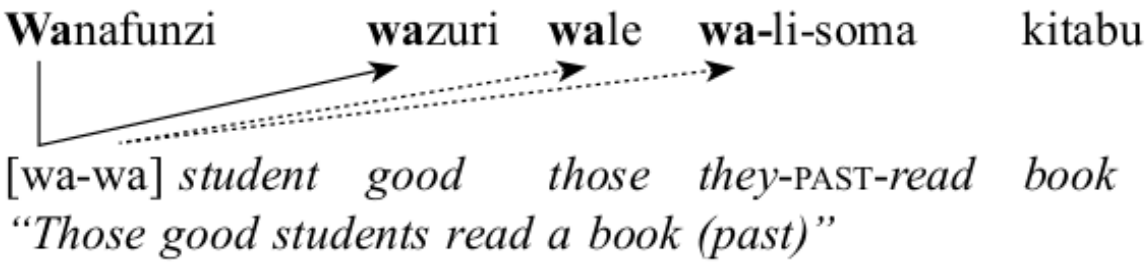
https://wals.info/feature/30A

Example: Gender in Swahili (swh, Atlantic-Congo)

"In Swahili, each noun prompts the use of certain types of agreement prefixes with adjectives (e.g. -zuri "good", -kubwa "big", -moja "one", -wili "two"), pronouns (e.g. demonstrative -le "that/those"), and verbs that depend on that noun in a given phrase or sentence."

Mpiranya (2015). Swahili Grammar and Workbook, p. 19.





Problem: Implementing Morphological Features

Given productive agreement systems for gender, number, and case, it quickly becomes a formidable task to implement morphological features into a PSG. See below the examples for the adjective *-zuri* “good” in Swahili.[6]

$A(SG, CL1) \rightarrow$ **m**zuri
$A(SG, CL2) \rightarrow$ **m**zuri
$A(SG, CL3) \rightarrow$ **ki**zuri
$A(SG, CL4) \rightarrow$ zuri
$A(SG, CL5) \rightarrow$ **n**zuri
$A(PL, CL1) \rightarrow$ **wa**zuri
$A(PL, CL2) \rightarrow$ **mi**zuri
$A(PL, CL3) \rightarrow$ **vi**zuri
$A(PL, CL4) \rightarrow$ **ma**zuri
$A(PL, CL5) \rightarrow$ **n**zuri

[6] This is based on my reading of the noun class system (CL) as defined by Mpiranya (2015), p. 22.

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Section 3: Syntactic Phenomena

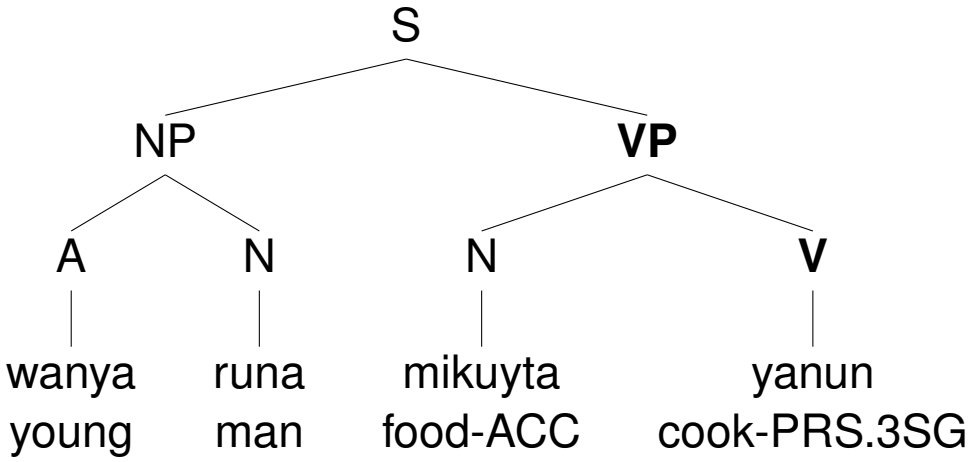
Verb Position

In some cases, the position of the verb can be handled straightforwardly by changing its position on the *left and right hand side of rules*, i.e. adapting the rules of how to combine the verb with its complements (e.g. noun phrases).

Sprach der Hase zum Igel [...]
Der Hase **sprach** zum Igel [...]
Der Hase zum Igel **sprach** [...]

Verb-final Position (Transitive)

Ayacucho Quechua (quy, Quechuan)



R (terminals)

1. A → wayna
2. N → runa
3. N → mikuyta
4. V → yanun

R (non-terminals)

5. S → **NP VP**
6. VP → **N V**
7. NP → A N

Rewrite Notation

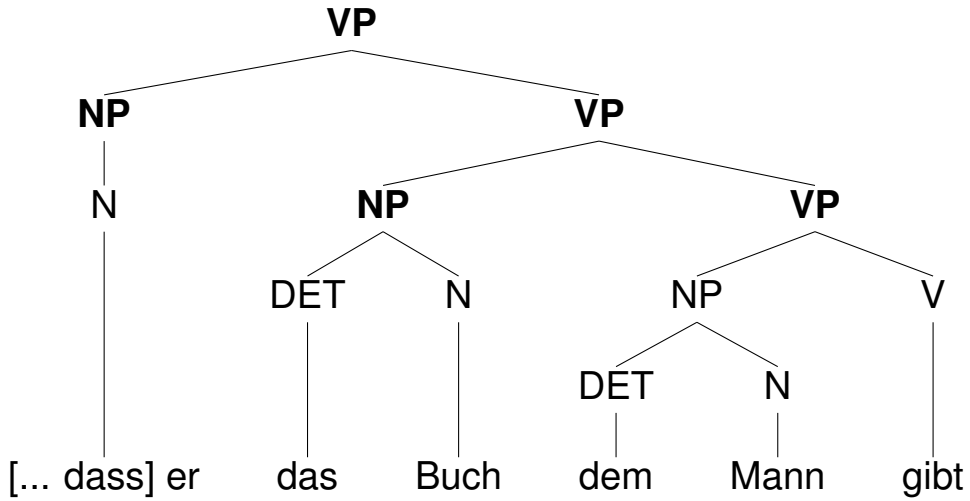
S
NP VP
NP N V
A N N V

---

wayna N N V
wayna runa N V
wayna runa mikuyta V
wayna runa mikuyta yanun

Verb-final Position (Ditransitive)

German (deu, Indo-European)



R (terminals)

1. DET → das
2. DET → dem
3. N → Buch
4. N → Mann
5. V → gibt

R (non-terminals)

6. **VP** → **NP VP**
7. VP → NP V
8. NP → DET N
9. NP → N

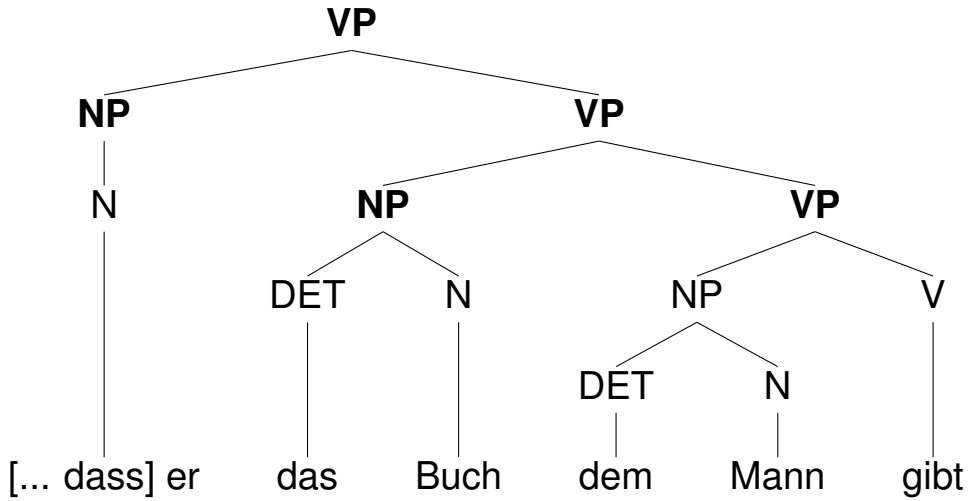
Rewrite Notation

VP
NP VP
NP NP VP
NP NP NP V
N NP NP V
N DET N NP V
N DET N DET N V

etc.

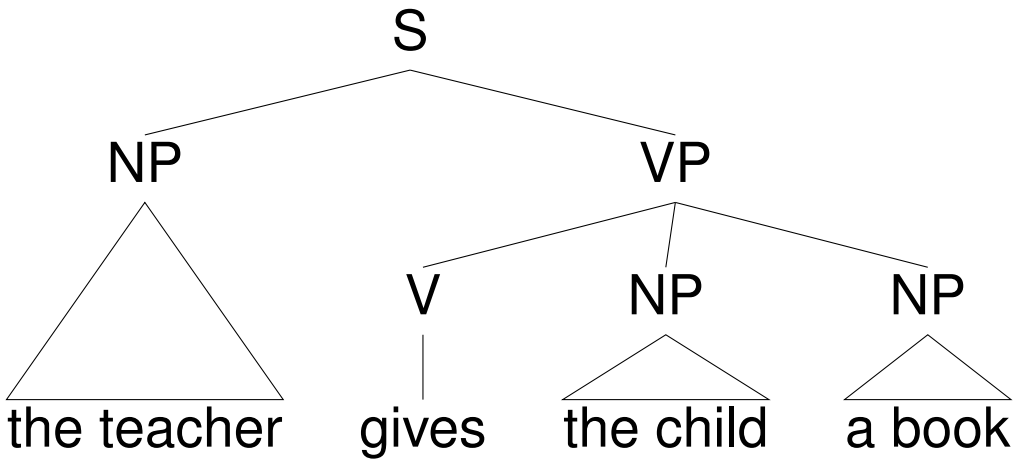
Verb-final Position (Ditransitive)

German (deu, Indo-European)



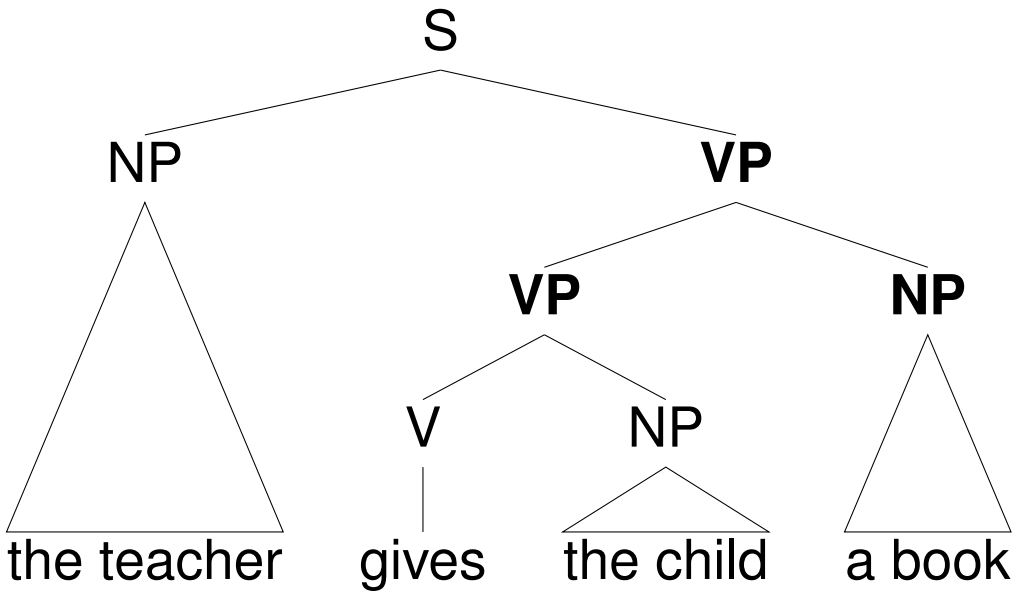
Problem: We here also have **internal unary branches** (NP $\rightarrow$ N). More importantly, we have introduced a **recursive rule** here (VP $\rightarrow$ NP VP) which will lead to the generation of ungrammatical sentences (without further constraints). To illustrate this, see the English examples below.

Verb-medial Position (Ditransitive)



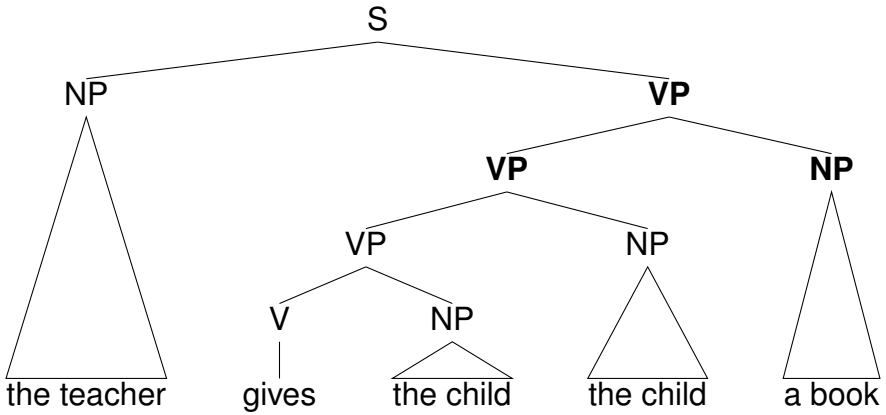
Note: If we allow for *multifurcations*, then the valency and position of the verb can be straightforwardly accounted for. Within the VP, the elements V, NP, and NP can be added and moved around in the rewrite rule.

Verb-medial Position (Ditransitive)



Note: If we only allow *bifurcations*, i.e. we apply the binarization constraint, then modelling the ditransitive construction gets more complicated. It is possible by introducing a recursive rule **VP** $\rightarrow$ **VP NP**. However, note that this set of rewrite rules will allow for the generation of ungrammatical sentences (without further constraints).

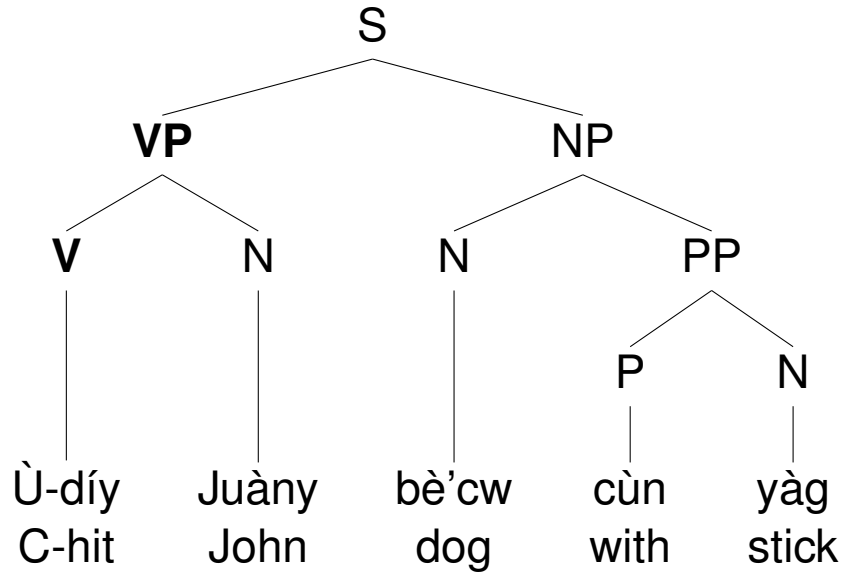
Verb-medial Position (Ditransitive)



Note: Once a recursive rule is introduced, however, we can apply it as often as we want. In order to overcome the generation of ungrammatical sentences, we would have to introduce an additional rule which limits the number of recursive applications.

Verb-initial Position

Zapotec (???, Otomanguean)



R (terminals)

1. N → yàg
2. N → bè'cw
3. N → Juàny
4. P → cùn
5. V → Ù-díy

R (non-terminals)

5. S → **VP NP**
6. VP → **V N**
7. NP → N PP
7. PP → P N

Rewrite Notation

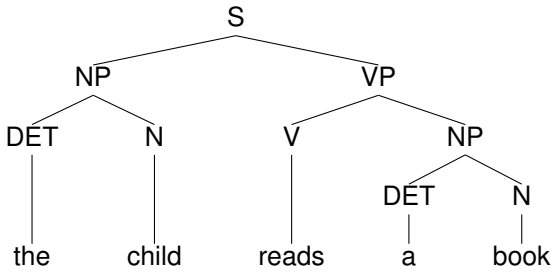
S
VP NP
V N NP
V N N PP
V N N P N

Ù-díy N N P N
Ù-díy Juàny N P N
Ù-díy Juàny bè'cw P N
Ù-díy Juàny bè'cw cùn N
Ù-díy Juàny bè'cw cùn yàg

The Passive

In a **passive construction**, the object of the corresponding *active sentence* becomes the subject.

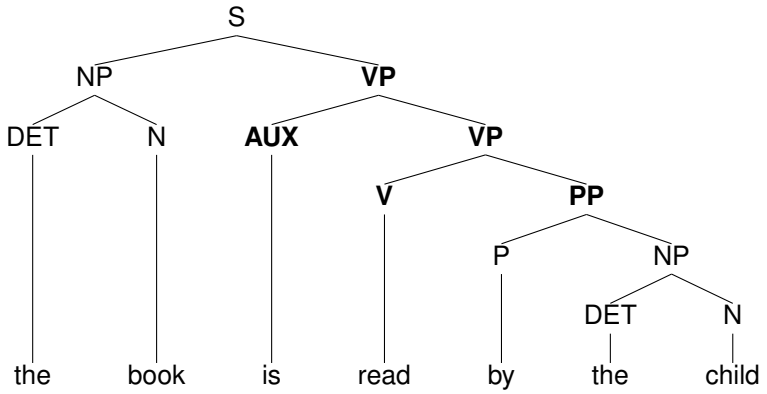
Active:



R (non-terminals)

1. S → NP VP
2. **VP → V NP**
3. NP → DET N

Passive:



R (non-terminals)

1. S → NP VP
2. **VP → AUX VP** (recursive !)
3. **VP → V PP**
4. **PP → P NP**
5. NP → DET N

Passive Transformations

Passive constructions are handled in some syntactic frameworks (e.g. Government and Binding) with the same underlying deep structure as **active constructions**. Note that this is an important deviation from traditional PSGs. In a traditional PSG you would have to formulate different phrase structure rules for active and passive sentences.

Early example of a transformational rule going back to Chomsky (1957):

$NP_1\ V_2\ NP_3 \rightarrow NP_3\ [_{AUX}\ be]\ V_2en\ [_{PP}\ [_P\ by]\ NP_1]$
John sees Mary $\rightarrow$ Mary $[_{AUX}$ is] seen $[_{PP}\ [_P$ by] John]

Müller (2019). Grammatical theory, p. 85.

Important Take-Home-Message

Introducing **recursive rules** (having the same non-terminal on both sides of the arrow $\rightarrow$, helps to stick to the binarization constraint, and hence reduces the complexity of individual rules. However, there is another problem immediately arising: the number of recursive applications will have to be limited in order to not generate ungrammatical sentences.

Section 4: Pros and Cons of PSG

Pros (Advantages)

- Implements *linearization constraints* explicitely
- Is grounded on a solid mathematical footing (automata theory)
- Can be exdended to model morphological features
- Relatively easily implementable in computational frameworks

Cons (Disadvantages)

- The assumption that all languages need phrase structure rules for their grammatical description might not be valid (e.g. free word order)
- Implementation of morphological features can be cumbersome, especially for languages with productive morphological marking (though this is also an issue for other frameworks)
- It excludes semantic aspects from questions of grammaticality
- Without further constraints, there is an infinite number of PSGs that can generate any given sentence or set of sentences. Hence, it is unclear how to choose a particular PSG

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Faculty of Philosophy
General Linguistics

Section 5: Basic Concepts in PSG

Basic Concepts in PSG

- Constituency ✓
- POS ✓
- Heads ✓
- Valency ✓[7]
- Grammatical Functions x[8]

[7] Valency is to some extent an issue that a PSG needs to deal with when formulating rewrite rules.

[8] Grammatical functions play no role for writing PSGs.

Section 6: References

References

Blevins et al. (2013). Phrase structure grammar. In: Marcel den Dikken (Ed.), *The Cambridge Handbook of Generative Syntax*. Cambridge: Cambridge University Press.

Bloomfield, Leonard (1933). *Language*. University of Chicago Press, Chicago.

Dixon, R. M. W. & Aikhenvald, A. Y. (2000). *Changing valency. Case studies in transitivity*. Cambridge: Cambridge University Press.

Greville G. Corbett (2013). Number of Genders. In: Dryer, Matthew S. & Haspelmath, Martin (eds.) The World Atlas of Language Structures Online. Leipzig: Max Planck Institute for Evolutionary Anthropology. (Available online at http://wals.info/chapter/30, Accessed on 2019-10-31.)

Harris, Zellig S (1957). Co-occurrence and transformation in linguistic structure. *Language*, 33, 283-340.

Jäger, Gerhard, and Rogers, James (2012). Formal language theory: refining the Chomsky hierarchy. *Philosophical Transactions of the Royal Society*, 367, pp. 1956-1970.

Mpiranya, Fidèle (2015). Swahili Grammar and Workbook. London/New York: Routledge.

Müller, Stefan. 2019. *Grammatical theory: From transformational grammar to constraint-based approaches. Third revised and extended edition.* ***Volume I****.* Berlin: Language Science Press.

Partee, Barbara H., Meulen, Alice ter, Wall, Robert E. (1990). *Mathematical methods in linguistics.* Studies in Linguistics and Philosophy, Vol. 30. Dordrecht/London/Boston: Kluwe Academic Publishers.

Wells, Rulon S. (1947). Immediate constituents. *Language*, 23, 81-117.

Thank You.

Contact:

Faculty of Philosophy
General Linguistics
Dr. Christian Bentz
SFS Wilhelmstraße 19-23, Room 1.24
chris@christianbentz.de
Office hours:
During term: Wednesdays 10-11am
Out of term: arrange via e-mail