Natural Language Processing
>> Syntax <<
>> from transformation to unification <<

winter / fall 2011/2012
41.4268

Prof. Dr. Bettina Harriehausen-Mühlbauer
Univ. of Applied Sciences, Darmstadt, Germany
https://www.fbi.h-da.de/organisation/personen/harriehausen-muehlbauer-bettina.html

Bettina.Harriehausen@h-da.de

content

1 What is syntax ?

2 Grammar theories and formalisms

- Dependency Grammar
- Transformational Grammar
- Phrase Structure Grammar
- Case Grammar
- Unification Based Grammar

Syntax - introduction

What is Syntax?

Syntax deals with:

- the analysis of NLP input on sentence level
- the generation of NLP output on sentence level
- structural descriptions on sentence level, mostly
  - in form of PS-(phrase structure) trees and/or
  - unification-based formalisms
- structural rules on sentence level (this can vaguely be compared to how „grammar“ of a language is traditionally taught)

Syntax - introduction

Acronyms used in structural descriptions of natural language („vocabulary“) = the auxiliary dictionary for the node descriptions:

S	=sentence/clause
N	=(a single) noun
NP	=noun phrase
V	=verb
VP	=verb phrase
AUX	=auxiliary verb
AJ/ADJ	=adjective
ADJP	=adjective phrase
ADV	=adverb
ADVP	=adverb phrase
DET	=determiner
CONJ	=conjunction
COMP	=complementizer
PRO	=pro-constituent
PUNC	=punctuation

Syntax - introduction

Examples for the node names:

S	=sentence/clause	„Does the dog chase the cat?“
N	=(a single) noun	„dog“
NP	=noun phrase	„the old dog“
V	=verb	„chase“
VP	=verb phrase	„chase the cat“
AUX	=auxiliary verb	„does“
AJ/ADJ	=adjective	„old“
ADJP	=adjective phrase	„old and gray“
ADV	=adverb	„happily“
ADVP	=adverb phrase	„once upon a time“
DET	=determiner	„the“

Syntax - introduction

Examples for the node names:

CONJ	=	conjunction	„and“
COMP	=	complementizer	„what“
PRO	=	pro-constituent	„he“
PUNC	=	punctuation	„?“

Grammar theories and formalisms

1. **Dependency Grammar**
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Genealogy of grammar theories and formalisms (and their influence on AI)

1. Dependency grammar (Tesnière 1953, 1959):
   The finite verb is the focal point of the sentential analysis, i.e. the ***valency*** of the verb determines the structure of the sentence.

   e.g. transitive verbs: valency of 2: Bob loves Mary.

   intransitive verbs: valency of 1. The cat sleeps.

Grammar theories and formalisms

Genealogy of grammar theories and formalisms (and their influence on AI)

2. Generative transformational grammar (TG) -> X-bar -> GB (Chomsky 1957, 1959)

The attempt to construct a formal model of the language competence of an ideal speaker-hearer:

*„I understand a generative grammar to simply be a rule system, which assigns **structural descriptions** to sentences in an explicit and well defined manner“ (Chomsky 1965)*

= formalization of natural language

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Genealogy of grammar theories and formalisms (and their influence on AI)

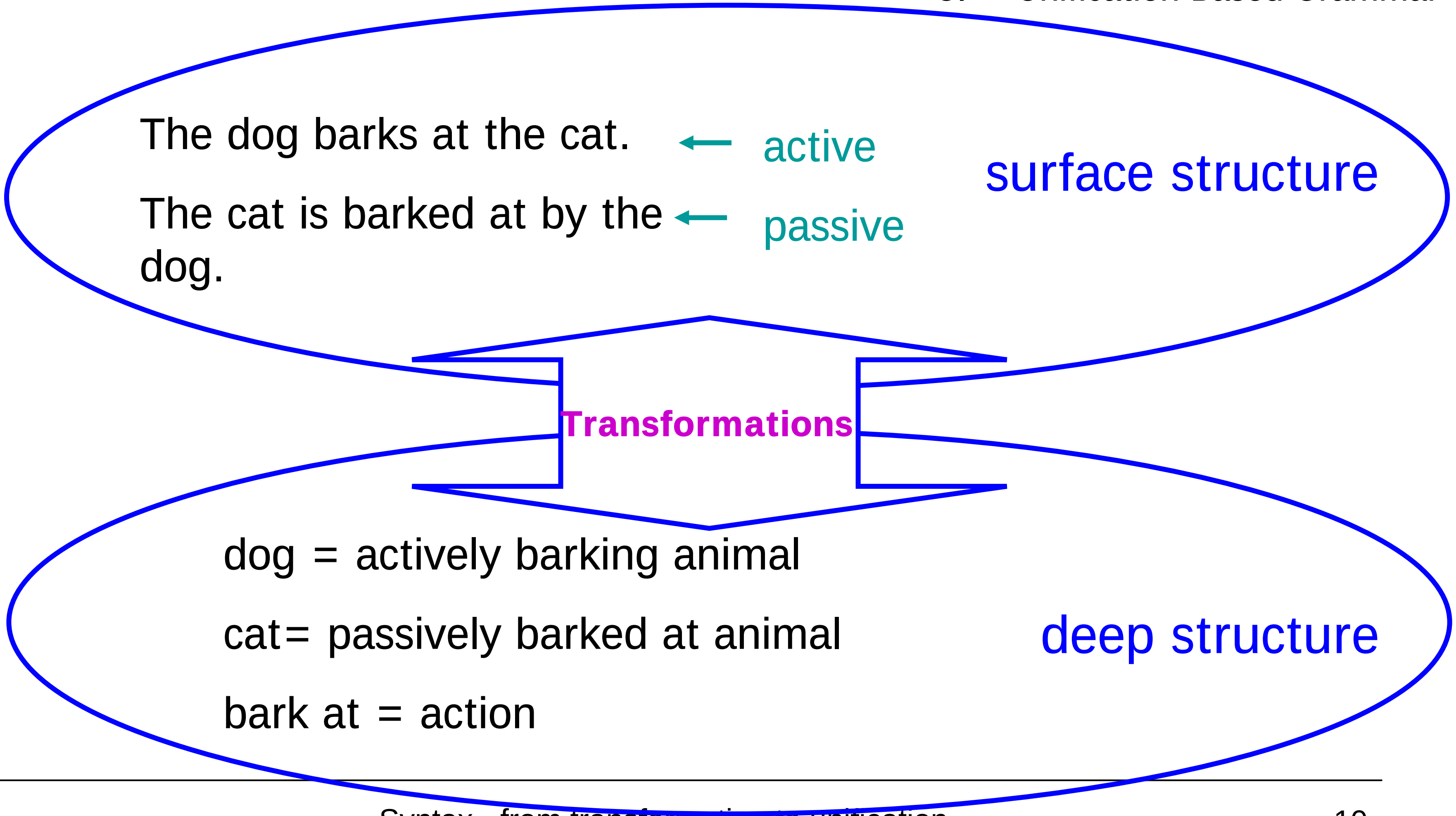
deep structure <-> surface structure

↑

transformation rules

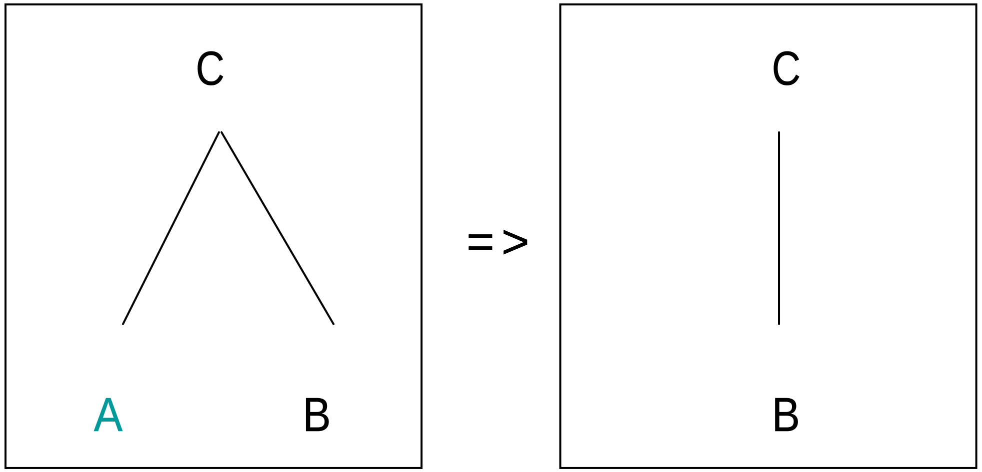
Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar



Examples for transformations

Deletion: A + B => B

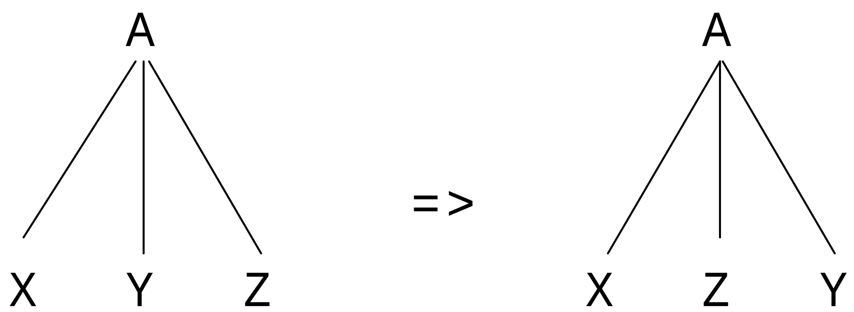


Example: He didn‘t know that he should read the book.

He didn‘t know --- he should read the book.

Examples for transformations

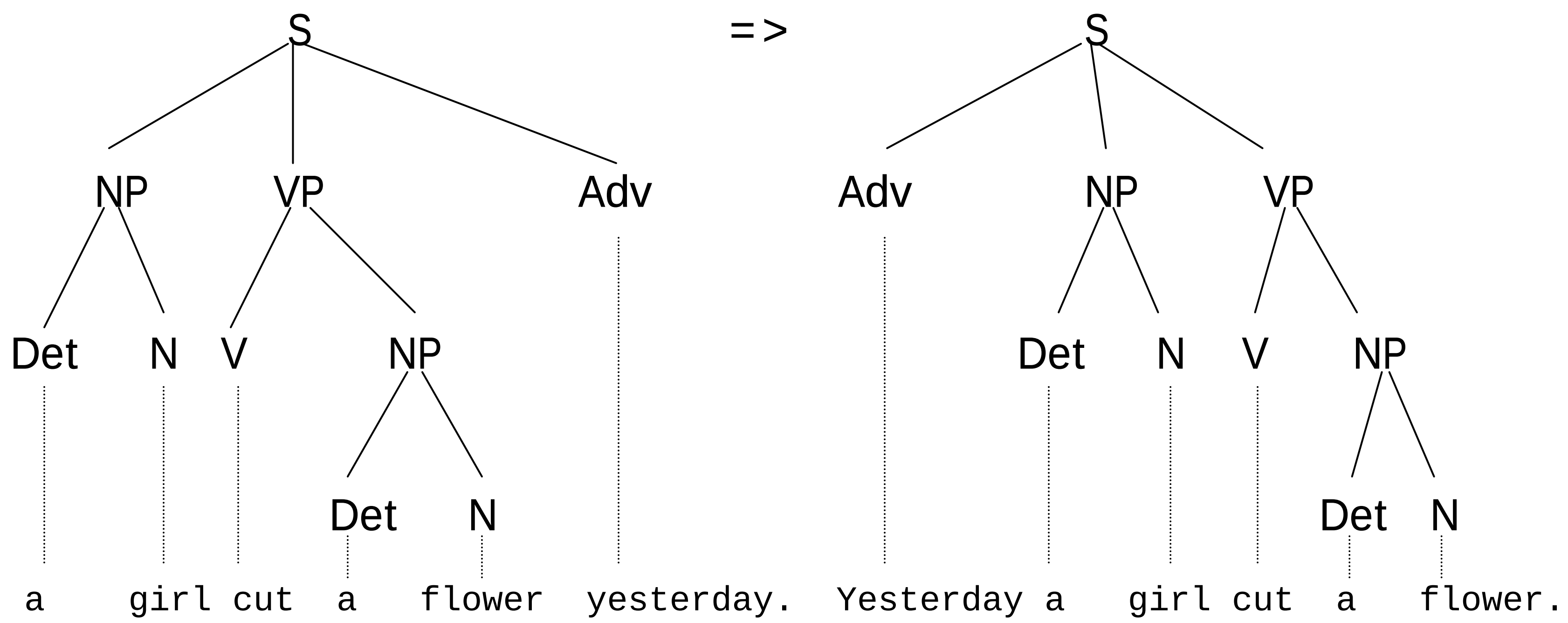
Permutation: X + Y + Z => X + Z + Y



Examples for transformations

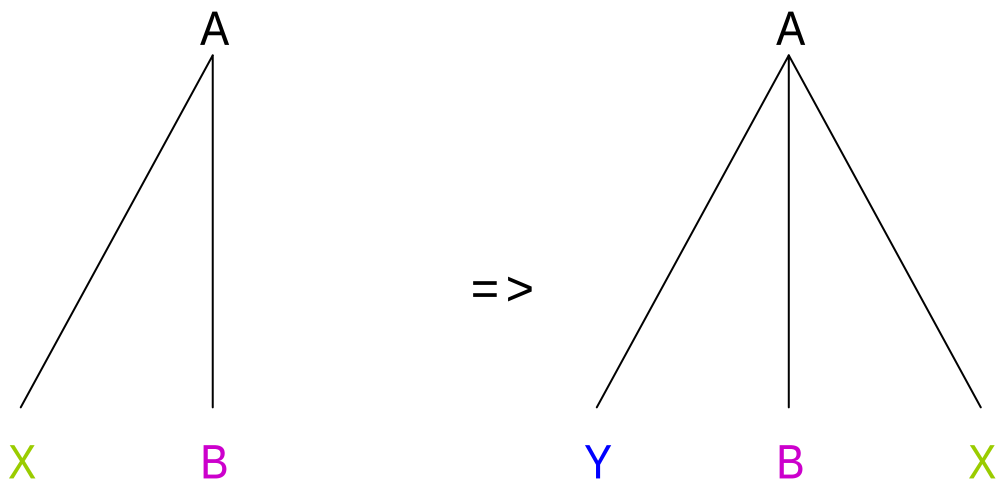
Sample of a permutation rule:

NP + VP + ADV => ADV + NP + VP



Examples for transformations

Substitution / Replacement: X + B => Y + B + X



Example: (in combination with permutation)

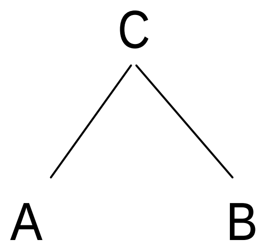
That it is raining is too bad.

It is too bad that it is raining.

Examples for transformations

Adjunction: B => A + B

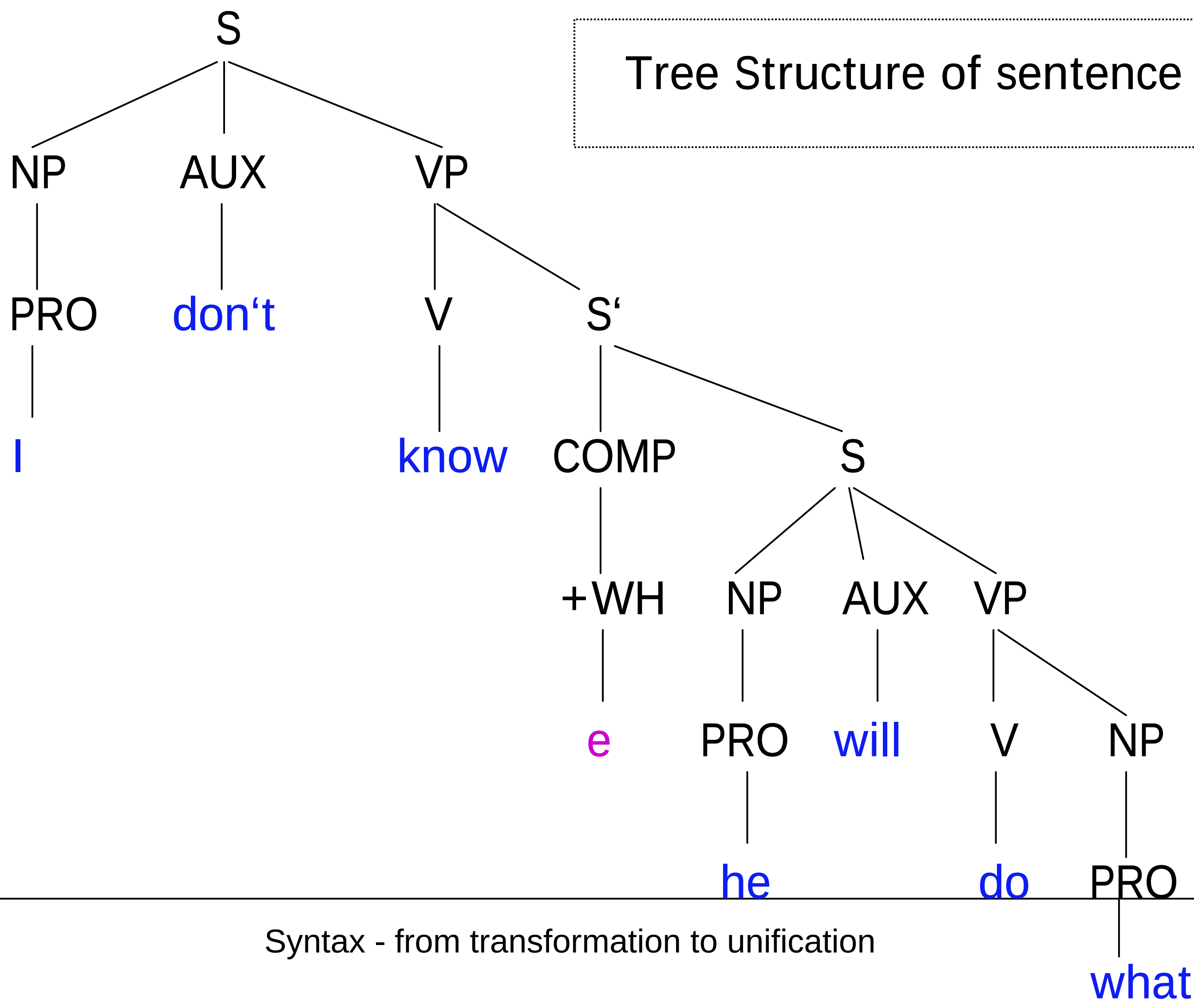
initial structure
before adjunction



sister-adjoin A
to the left of B

Examples for transformations

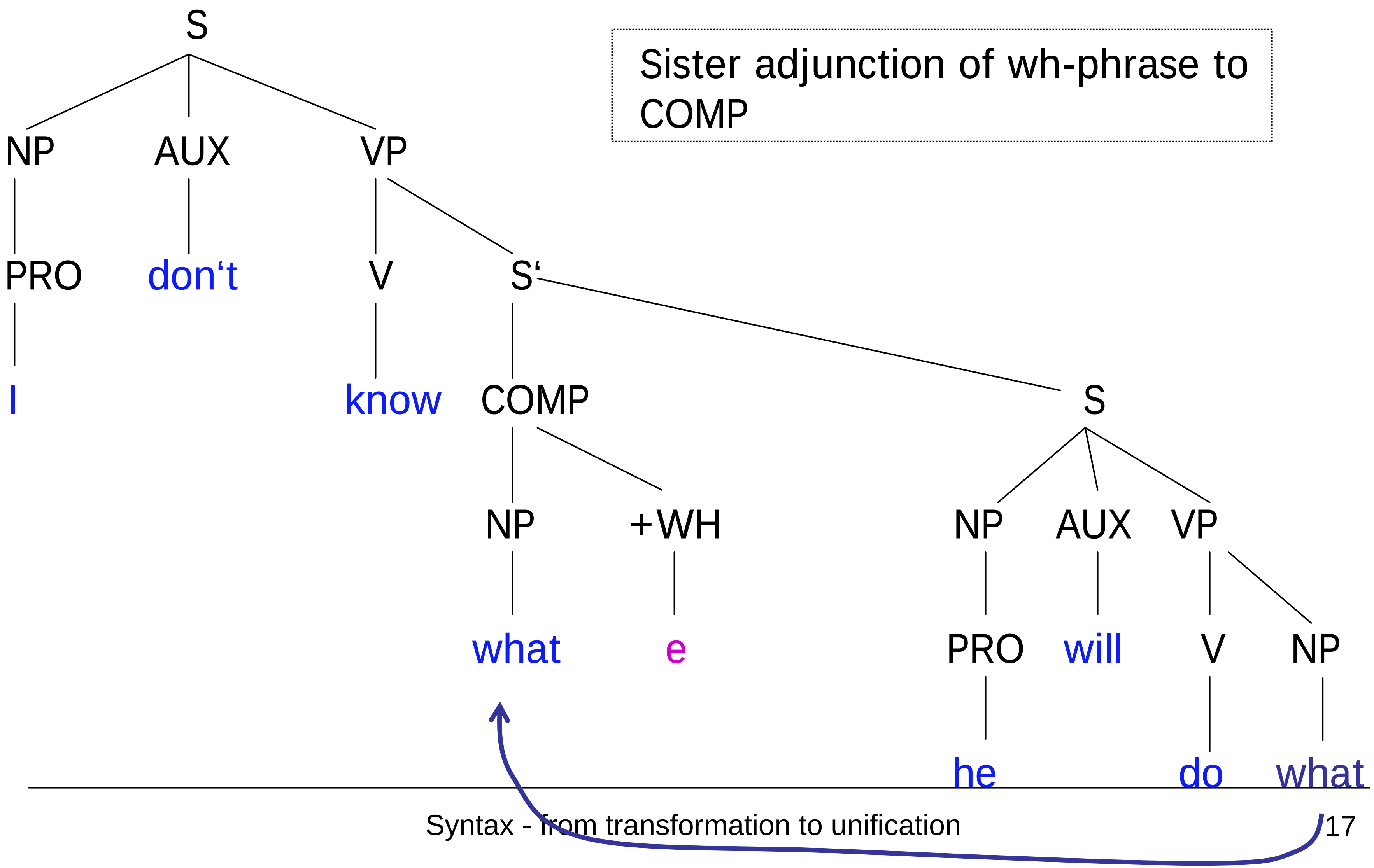
Adjunction: Example: *I don‘t know what he will do.*



Tree Structure of sentence

Examples for transformations

Adjunction: Example: *I don‘t know what he will do.*



Grammar theories and formalisms

Binary rules

Parser

Top-down:

S -> VP + PUNC
VP -> NP + VP
NP -> DET + NP
NP-> ADJ + NP
VP -> V + NP
...

Bottom-up:

DET + NP -> NP
ADJ + NP -> NP
V + NP -> VP
NP + VP -> VP
VP + PUNC -> S
...

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Binary rules

The dog chases the cat.

Top-down:

S -> VP + PUNC
VP -> NP + VP
NP -> DET + NP
NP-> N
VP -> VP + NP
VP-> V
…

S ->The dog chases the cat (VP) + . (PUNC)
VP-> the dog (NP) + chases the cat (VP)
NP-> the(DET) + dog (NP)
NP-> dog (N)
VP-> chases (VP) + the cat (NP)
VP-> chases (V)
NP-> the (DET) + cat (NP)
NP-> cat (N)

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Binary rules

The dog chases the cat.

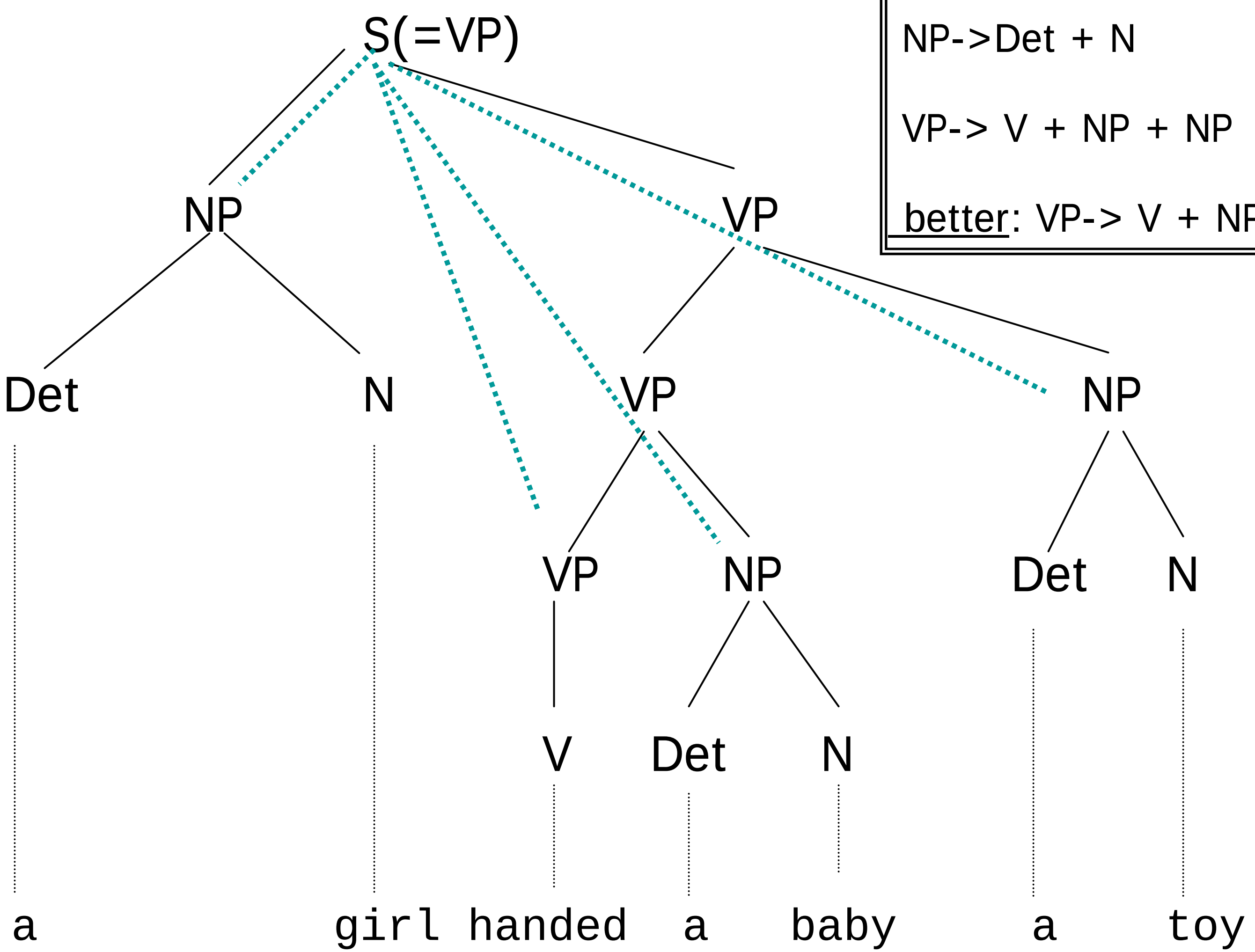
Bottom-up:

DET + NP -> NP
N-> NP
VP + NP -> VP
V->VP
NP + VP -> VP
VP + PUNC -> S
...

dog (N) -> NP
the (DET) + dog (NP) -> NP
chases (V)-> VP
cat (N)-> NP
the (DET) + cat (NP) -> NP
chases (VP) + the cat (NP) -> VP
the dog (NP) + chases the cat (VP) -> VP
the dog chases the cat (VP) + .(PUNC) -> S

constituent structure / PS-tree:

A girl handed a baby a toy.



S -> NP + VP

NP->Det + N

VP-> V + NP + NP

<u>better</u>: VP-> V + NP (recursive!)

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Why binary rules ?

Claim: „ the set of well-formed sentences in any language is *infinite*“

(Chomsky)

Claim: „ there is no (theoretical) upper limit on the length of sentences in any language“ (though there are of course performance limitations)

Example 1: We can have indefinitely many attributive adjectives qualifying a noun in English:

John is a handsome man.
John is a dark, handsome man.
John is a tall, dark, handsome man.
John is a sensitive, tall, dark, handsome man.
John is an intelligent, sensitive, tall, dark, handsome man.
etc.

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Why binary rules ?

Example 2: There is in principle no upper limit to the number of quantifying expressions we can use to modify an adjective in English:

Debbie Harry is very attractive.
Debbie Harry is very, very attractive.
Debbie Harry is very, very, very attractive.
Debbie Harry is very , very, very, very attractive.
Debbie Harry is very, very , very, very, very attractive.
Debbie Harry is very, very, very , very, very, very attractive.
etc.

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Why binary rules ?

Example 3: There isn‘t any limit on the number of times that we can use one clause as the complement of another in English:

John said that Mary was ill.
Fred said that John said that Mary was ill.
Harry said that Fred said that John said that Mary was ill.
etc.

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Why binary rules ?

Example 4: There is no limit on the number of phrases that we can conjoin together by *and* (or *or*) in English:

I met Debbie and Harry.
I met Debbie, Noam, and Harry.
I met Debbie, Noam, the dustman, and Harry.
I met Debbie, Noam, the dustman, the president, and Harry.
etc.

Grammar theories and formalisms

Why binary rules ?

Example 5: There is no upper limit on the number of relative clauses a sentence can contain in English:

I chased the dog.
I chased the dog that chased the cat.
I chased the dog that chased the cat that chased the rat.
I chased the dog that chased the cat that chased the rat that chased the mouse.
etc.

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Why binary rules ?

In order to reach the goal of writing a set of finite rules to describe an infinite set of well-formed sentences, we need to think „binary“ in order to be able to program rules efficiently – otherwise we would have an infinite set of rules (…and our task of writing a grammar would be an endless story…)

That is why „binary rules“ !

Grammar theories and formalisms

References

Jensen, Karen. 1987. Binary rules and non-binary trees: Breaking down the concept of phrase structure. In *Mathematics of language*, ed. Alexis Manaster-Ramer, pages 65-86. Amsterdam, John Benjamins Publishing Company

Jensen, Karen. Issues in Parsing.

Grammar theories and formalisms

Genealogy of grammar theories and formalisms (and their influence on AI)

3. Phrase structure grammar (PS-grammar)

= a finite set of PS-rules

A PS-grammar consists of:

- an auxiliary dictionary (the node names)
- the final vocabulary (the lexicon)
- a start node (= S)
- a set of PS-rules

Grammar theories and formalisms

1. Dependency Grammar
2. Transformational Grammar
3. Phrase Structure Grammar
4. Case Grammar
5. Unification Based Grammar

Genealogy of grammar theories and formalisms (and their influence on AI)

3. Phrase structure grammar

context sensitive

X -> Y/w_z

V_trans/_obj

VP

V_intrans/_#

context free

X -> Y

S	-> NP + VP
NP	-> Det + NP
VP	-> V_trans + NP
VP	-> V_intrans

(a) PS-Rules

S -> NP + VP

NP -> PRO

VP -> V + (S)

RULES

1. Dependency Grammar
2. Transformational Grammar
3. **Phrase Structure Grammar**
4. Case Grammar
5. Unification Based Grammar

(b) Lexical Insertion Rules

PRO -> I, she

V -> wonder, snores

COMP -> whether

INTERFACE :

RULES-LEXICON

(c) I -> (PRO, +1sg, +human, +animate,...)

she -> (PRO, +3sg, +human, +animate,...)

LEXICON

wonder -> (VERB, +1sg, -3sg, +wh-clause, +cognitive,...)

snores -> (VERB, +3sg, +intrans, +active, +animate,...)

Lexicon: parts of speech, subcategorisational features (morpho-syntactic), selectional features (semantic)

Syntax

grammar rules

electronic dictionary