Tree-based Models

IT Systems Engineering | Universität Potsdam

Dr. Mariana Neves
(adapted from the original slides
of Prof. Philipp Koehn)

January 18th, 2016

Tree-Based Models

- Traditional statistical models operate on sequences of words
- Many translation problems can be best explained by pointing to syntax
  - reordering, e.g., verb movement in German–English translation
  - long distance agreement (e.g., subject-verb) in output

$\Rightarrow$ Translation models based on tree representation of language
  - significant ongoing research
  - state-of-the art for some language pairs

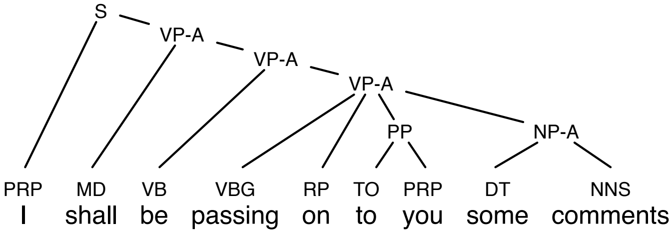
Phrase Structure Grammar

- Phrase structure
  - noun phrases: the big man, a house, ...
  - prepositional phrases: at 5 o'clock, in Edinburgh, ...
  - verb phrases: going out of business, eat chicken, ...
  - adjective phrases, ...

Phrase Structure Grammar

- Context-free Grammars (CFG)
  - non-terminal symbols (NT): phrase structure labels, part-of-speech tags
  - terminal symbols (T): words
  - production rules: $\mathrm{NT} \rightarrow [\mathrm{NT},\mathrm{T}]+$
    example: $\mathrm{NP} \rightarrow \mathrm{DET}\ \mathrm{NN}$

Phrase Structure Grammar



Phrase structure grammar tree for an English sentence
(as produced by Collins' parser)

Synchronous Phrase Structure Grammar

- English/German rule

$$\text{NP} \rightarrow \text{DET JJ NN}$$

- French/Portuguese/Spanish rule

$$\text{NP} \rightarrow \text{DET NN JJ}$$

- Synchronous rule (indices indicate alignment):

$$\text{NP} \rightarrow \text{DET}_1 \ \text{NN}_2 \ \text{JJ}_3 \mid \text{DET}_1 \ \text{JJ}_3 \ \text{NN}_2$$

Synchronous Grammar Rules

- Nonterminal rules

$$\text{NP} \rightarrow \text{DET}_1\ \text{NN}_2\ \text{JJ}_3 \mid \text{DET}_1\ \text{JJ}_3\ \text{NN}_2$$

- Terminal rules

$$\text{N} \rightarrow \text{maison} \mid \text{house}$$

$$\text{NP} \rightarrow \text{la maison bleue} \mid \text{the blue house}$$

- Mixed rules (mixing terminal and non-terminal symbols)

$$\text{NP} \rightarrow \text{la maison JJ}_1 \mid \text{the JJ}_1\ \text{house}$$

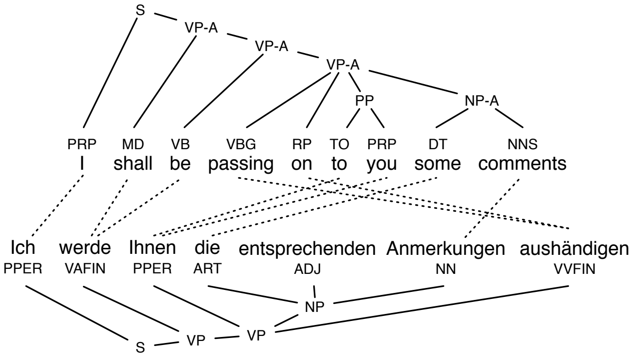
Tree-Based Translation Model

- Translation by parsing
  - synchronous grammar has to parse entire input sentence
  - output tree is generated at the same time
  - process is broken up into a number of rule applications
- Translation probability

$$\textsc{score}(\textsc{tree}, \textsc{e}, \textsc{f}) = \prod_i \textsc{rule}_i$$

- Many ways to assign probabilities to rules

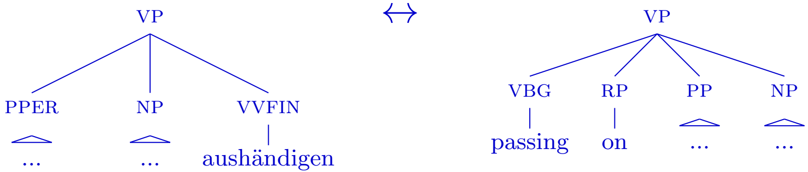
Aligned Tree Pair



Phrase structure grammar trees with word alignment
(German–English sentence pair.)

Reordering Rule

- Subtree alignment



Reordering Rule

- Synchronous grammar rule

$$\mathrm{VP} \rightarrow \mathrm{PPER}_1\ \mathrm{NP}_2\ \text{aushändigen} \mid \text{passing on}\ \mathrm{PP}_1\ \mathrm{NP}_2$$

- Note:
  - one word aushändigen mapped to two words passing on ok
  - but: fully non-terminal rule not possible
    (one-to-one mapping constraint for nonterminals)

Another Rule

- Subtree alignment

$$\begin{array}{c} \text{PRO} \\ | \\ \text{Ihnen} \end{array} \quad \leftrightarrow \quad \begin{array}{c} \text{PP} \\ \text{TO} \quad \text{PRP} \\ | \qquad\quad | \\ \text{to} \qquad \text{you} \end{array}$$

- Synchronous grammar rule (stripping out English internal structure)

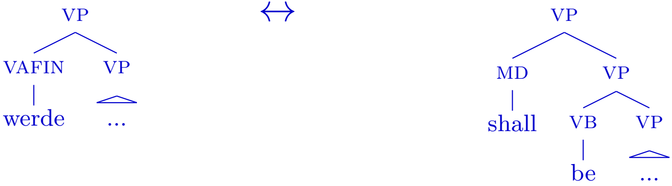
$$\text{PRO/PP} \rightarrow \text{Ihnen} \mid \text{to you}$$

- Rule with internal structure

$$\text{PRO/PP} \rightarrow \text{Ihnen} \quad \Bigg| \quad \begin{array}{cc} \text{TO} & \text{PRP} \\ | & | \\ \text{to} & \text{you} \end{array}$$

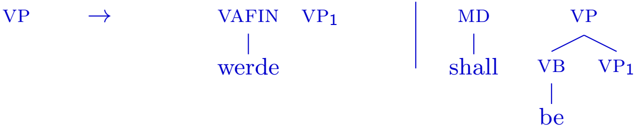
Another Rule

- Translation of German werde to English shall be



- Translation rule needs to include mapping of VP

⇒ Complex rule



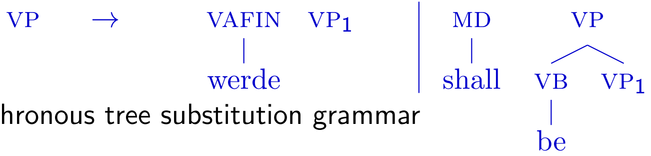
Internal Structure

- Stripping out internal structure

$$\text{VP} \rightarrow \text{werde VP}_1 \mid \text{shall be VP}_1$$

$\Rightarrow$ synchronous context free grammar

- Maintaining internal structure



$\Rightarrow$ synchronous tree substitution grammar

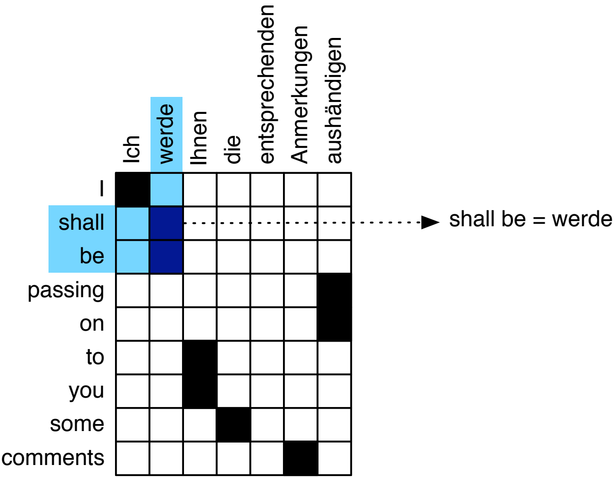
Learning Synchronous Grammars

- Extracting rules from a word-aligned parallel corpus

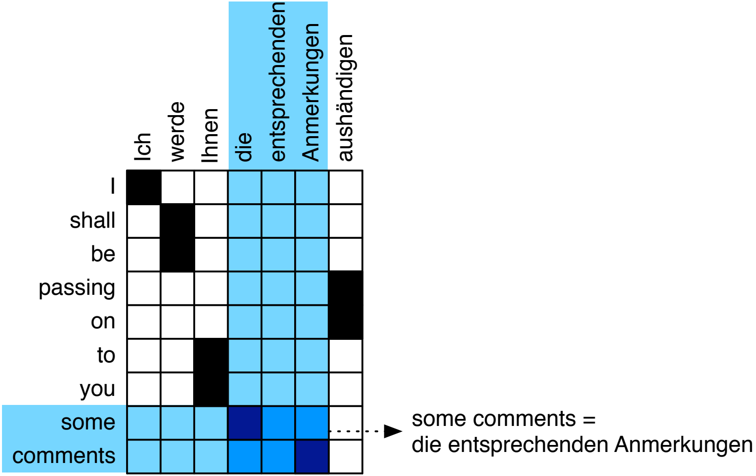
- First: Hierarchical phrase-based model
  - only one non-terminal symbol X
  - no linguistic syntax, just a formally syntactic model

- Then: Synchronous phrase structure model
  - non-terminals for words and phrases: NP, VP, PP, ADJ, ...
  - corpus must also be parsed with syntactic parser

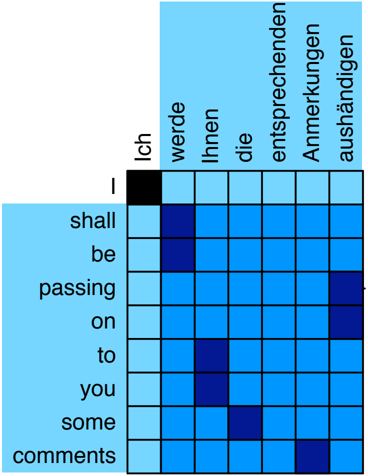
Extracting Phrase Translation Rules



Extracting Phrase Translation Rules

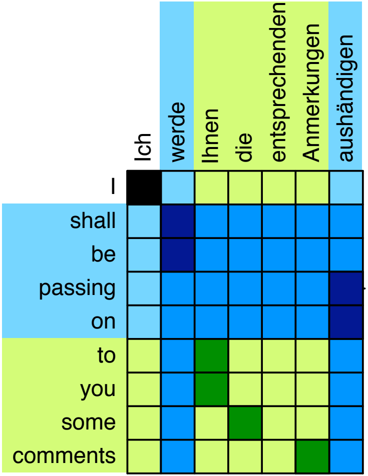


Extracting Phrase Translation Rules



werde Ihnen die entsprechenden Anmerkungen aushändigen
= shall be passing on to you some comments

Extracting Hierarchical Phrase Translation Rules



subtracting subphrase

werde X aushändigen
= shall be passing on X

Formal Definition

- Recall: consistent phrase pairs

$$
\begin{aligned}
(\bar{e}, \bar{f}) \text{ consistent with } A \Leftrightarrow \\
& \forall e_i \in \bar{e} : (e_i, f_j) \in A \rightarrow f_j \in \bar{f} \\
& \text{AND } \forall f_j \in \bar{f} : (e_i, f_j) \in A \rightarrow e_i \in \bar{e} \\
& \text{AND } \exists e_i \in \bar{e}, f_j \in \bar{f} : (e_i, f_j) \in A
\end{aligned}
$$

- Let P be the set of all extracted phrase pairs $(\bar{e}, \bar{f})$

Formal Definition

- Extend recursively:

$$\begin{aligned}\text{if } (\bar{e}, \bar{f}) \in P \text{ AND } (\bar{e}_{\text{SUB}}, \bar{f}_{\text{SUB}}) \in P \\ \text{AND } \bar{e} = \bar{e}_{\text{PRE}} + \bar{e}_{\text{SUB}} + \bar{e}_{\text{POST}} \\ \text{AND } \bar{f} = \bar{f}_{\text{PRE}} + \bar{f}_{\text{SUB}} + \bar{f}_{\text{POST}} \\ \text{AND } \bar{e} \neq \bar{e}_{\text{SUB}} \text{ AND } \bar{f} \neq \bar{f}_{\text{SUB}} \\ \text{add } (e_{\text{PRE}} + \text{X} + e_{\text{POST}}, f_{\text{PRE}} + \text{X} + f_{\text{POST}}) \text{ to } P\end{aligned}$$

  (note: any of e_{PRE}, e_{POST}, f_{PRE}, or f_{POST} may be empty)

- Set of hierarchical phrase pairs is the closure under this extension mechanism

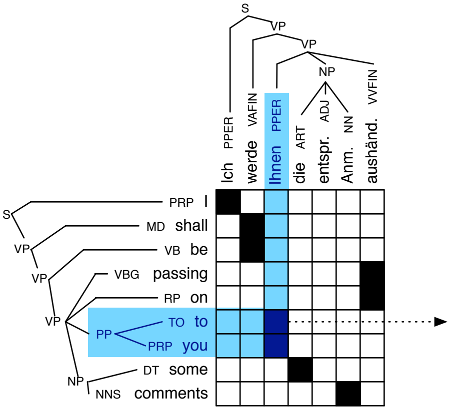
Comments

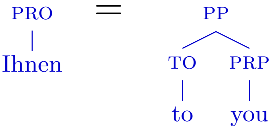
- Removal of multiple sub-phrases leads to rules with multiple non-terminals, such as:

$$\mathrm{Y} \rightarrow \mathrm{X}_1 \ \mathrm{X}_2 \ \mid \ \mathrm{X}_2 \ \textit{of} \ \mathrm{X}_1$$

- Typical restrictions to limit complexity [Chiang, 2005]
  - at most 2 nonterminal symbols
  - at least 1 but at most 5 words per language
  - span at most 15 words (counting gaps)

Learning Syntactic Translation Rules



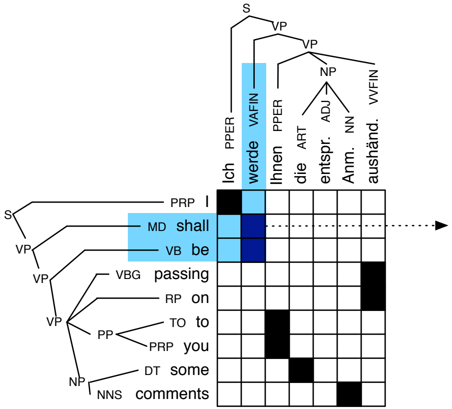


Constraints on Syntactic Rules

- Same word alignment constraints as hierarchical models
- Hierarchical: rule can cover any span
  $\Leftrightarrow$ syntactic rules must cover constituents in the tree
- Hierarchical: gaps may cover any span
  $\Leftrightarrow$ gaps must cover constituents in the tree

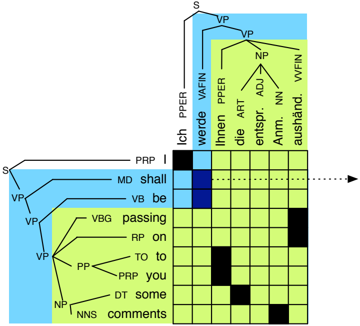
- Much less rules are extracted (all things being equal)

Impossible Rules

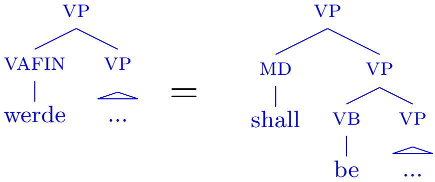


English span not a constituent,
no rule is extracted

Rules with Context



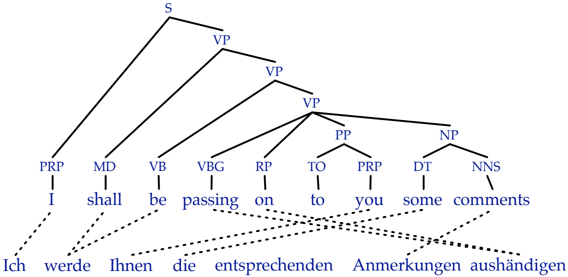
Rule with this phrase pair requires syntactic context



Too Many Rules Extractable

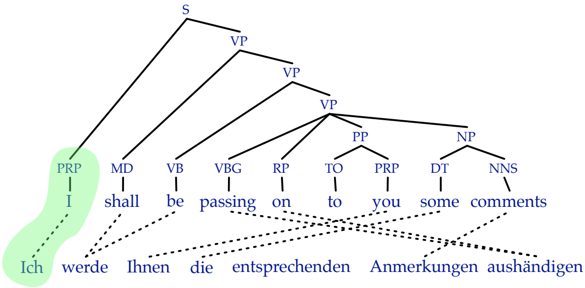
- Huge number of rules can be extracted
  (every alignable node may or may not be part of a rule $\rightarrow$ exponential number of rules)
- Need to limit which rules to extract
- Option 1: similar restriction as for hierarchical model
  (maximum span size, maximum number of terminals and non-terminals, etc.)
- Option 2: only extract minimal rules ("GHKM" rules)

Minimal Rules



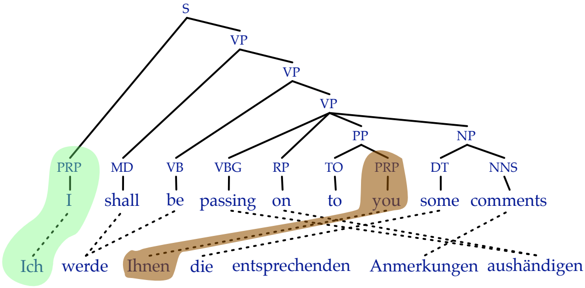
Extract: set of smallest rules required to explain the sentence pair

Lexical Rule



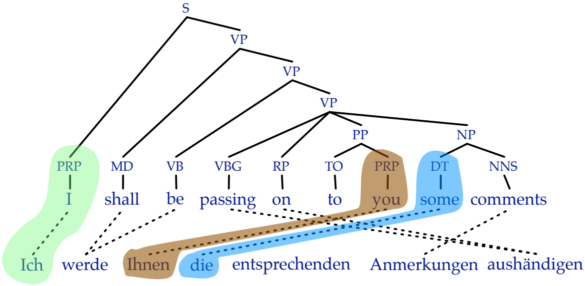
Extracted rule: $\text{PRP} \rightarrow \text{Ich} \mid \text{I}$

Lexical Rule



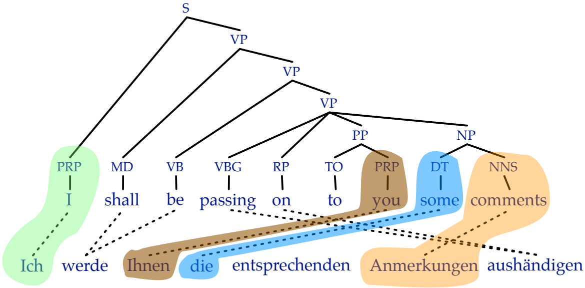
Extracted rule: PRP → Ihnen | you

Lexical Rule



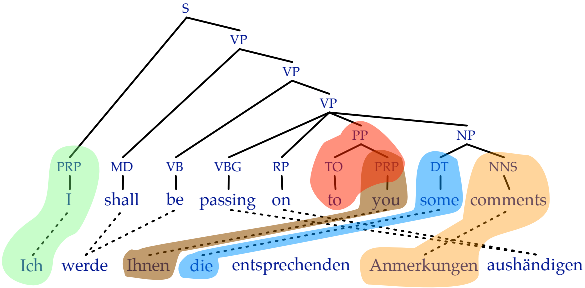
Extracted rule: DT $\rightarrow$ die | some

Lexical Rule



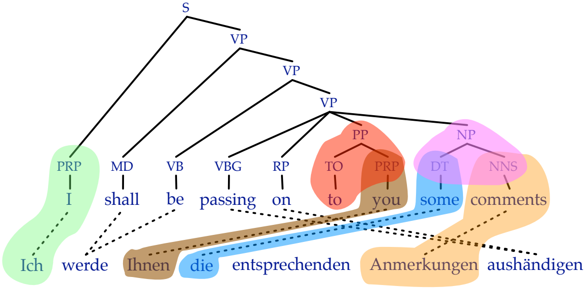
Extracted rule: NNS → Anmerkungen | comments

Insertion Rule



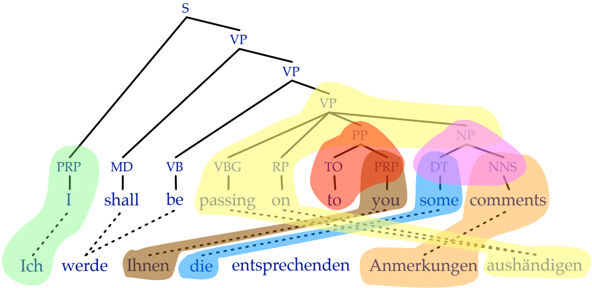
Extracted rule: $\mathrm{PP} \rightarrow \mathrm{X} \mid \text{to PRP}$

Non-Lexical Rule



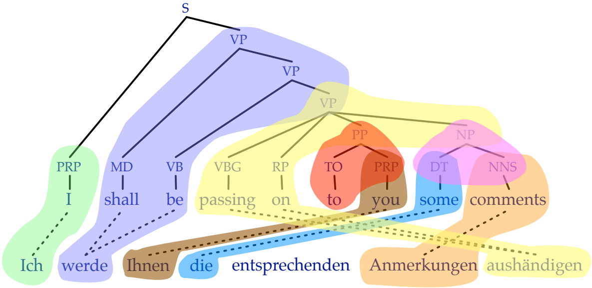
Extracted rule: $\mathrm{NP} \rightarrow \mathrm{X}_1\ \mathrm{X}_2 \mid \mathrm{DT}_1\ \mathrm{NNS}_2$

Lexical Rule with Syntactic Context



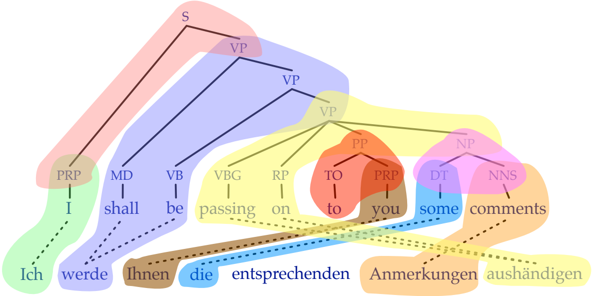
Extracted rule: $\mathrm{VP} \rightarrow \mathrm{X}_1\ \mathrm{X}_2$ aushändigen | passing on $\mathrm{PP}_1\ \mathrm{NP}_2$

Lexical Rule with Syntactic Context



Extracted rule: VP → werde X | shall be VP (ignoring internal structure)

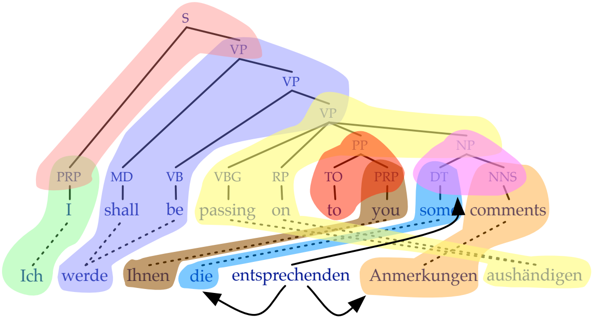
Non-Lexical Rule



Extracted rule: $\text{S} \rightarrow \text{X}_1\ \text{X}_2 \mid \text{PRP}_1\ \text{VP}_2$

DONE — note: one rule per alignable constituent

Unaligned Source Words



Attach to neighboring words or higher nodes → additional rules

Too Few Phrasal Rules?

- Lexical rules will be 1-to-1 mappings (unless word alignment requires otherwise)
- But: phrasal rules very beneficial in phrase-based models
- Solutions
  - combine rules that contain a maximum number of symbols (as in hierarchical models, recall: "Option 1")
  - compose minimal rules to cover a maximum number of non-leaf nodes

Composed Rules

- Current rules

$X_1\ X_2 =$ NP
DT$_1$ NNS$_1$

die = DT
some

entsprechenden Anmerkungen = NNS
comments

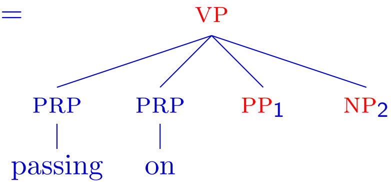
- Composed rule

die entsprechenden Anmerkungen = NP
DT NNS
some comments

(1 non-leaf node: NP)

Minimal rule

x_1 x_2 aushändigen =

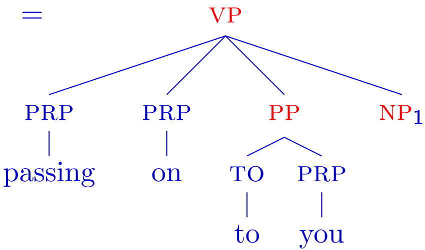


3 non-leaf nodes:

VP, PP, NP

Composed Rules

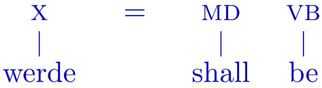
Ihnen x_1 aushändigen =



3 non-leaf nodes:
VP, PP and NP

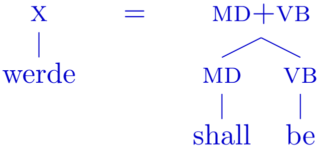
Relaxing Tree Constraints

- Impossible rule



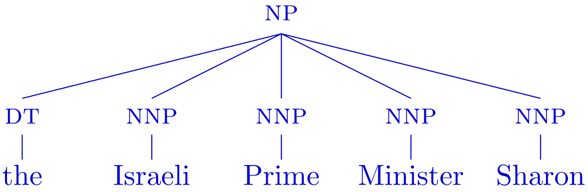
- Create new non-terminal label: MD+VB

⇒ New rule



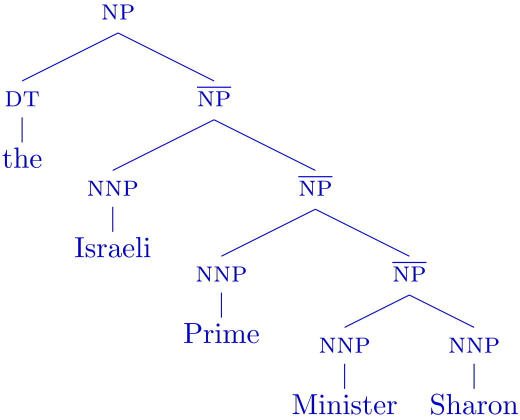
Special Problem: Flat Structures

- Flat structures severely limit rule extraction



- Can only extract rules for individual words or entire phrase

Relaxation by Tree Binarization



More rules can be extracted

Left-binarization or right-binarization?

Scoring Translation Rules

- Extract all rules from corpus
- Score based on counts
  - joint rule probability: $p(\mathrm{LHS}, \mathrm{RHS}_f, \mathrm{RHS}_e)$
  - rule application probability: $p(\mathrm{RHS}_f, \mathrm{RHS}_e | \mathrm{LHS})$
  - direct translation probability: $p(\mathrm{RHS}_e | \mathrm{RHS}_f, \mathrm{LHS})$
  - noisy channel translation probability: $p(\mathrm{RHS}_f | \mathrm{RHS}_e, \mathrm{LHS})$
  - lexical translation probability: $\prod_{e_i \in \mathrm{RHS}_e} p(e_i | \mathrm{RHS}_f, a)$

Suggested reading

- Statistical Machine Translation, Philipp Koehn (chapter 11).