Statistical Machine Translation

-tree-based models-

Artem Sokolov
Computerlinguistik
Universität Heidelberg
Sommersemester 2015

material from P. Koehn, S. Riezler

- Traditional statistical models operate on sequences of words
- Linguistic theories tell us that a deeper structure exists
- 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
  - ➡ trend: as technology matures even more pairs are better handled by tree-based SMT

Idea: word groups should correspond to constituents of certain roles and functions

- 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, angry with the high prices, faster than you, ...
- Weighted Context-free Grammars (CFG)
  - $G =< N, T, (P, \pi), (S, \sigma) >$
  - non-terminal symbols N: phrase structure labels, part-of-speech tags
  - terminal symbols T: words
  - production rules P: $N \rightarrow (N \cup T)^*$
  - weights of production rules: $\pi : P \rightarrow K$ (K is a semiring)
  - start symbol S
  - weights of start states: $\sigma : S \rightarrow K$ (K is a semiring) example: $\textsc{np} \rightarrow \textsc{det nn}$
    example: $\textsc{np} \rightarrow \textsc{det}$ house

Weighted Synchronous Context-free Grammars (SCFG)

- $G =< N, T^1, T^2, (P, \pi), (S, \sigma) >$
- non-terminal symbols N
- source and target terminal symbols T^1, T^2
- production rules P: $N \rightarrow (N \cup T^1)^* \times (\{\boxed{1}, \boxed{2}, \ldots\} \cup T^2)^*$
- weights of production rules: $\pi : P \rightarrow K$ (K is a semiring)
- start symbol S
- weights of start states: $\sigma : S \rightarrow K$ (K is a semiring)

- Nonterminal rules

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

- Terminal rules

$$\mathrm{N} \rightarrow \textsf{maison} \mid \textsf{house}$$
$$\mathrm{NP} \rightarrow \textsf{la maison bleue} \mid \textsf{the blue house}$$

- Mixed rules

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

- 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
- Each rule is weighted (definition)
- Total translation probability

$$\text{SCORE}(\text{TREE}, \text{E}, \text{F}) = \prod_i \text{RULE}_i$$

- Many ways to assign probabilities to rules (as there are many parses possible)

$$\Sigma = \{a, b, c\} \quad \Delta = \{x, y, z\} \quad V = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\} \quad S = \{\mathrm{C}\} \quad \sigma(\mathrm{C}) = 1.0$$

$$
\begin{array}{llll}
R = \{ & \mathrm{A} & \stackrel{0.5}{\rightarrow} & \langle \mathrm{A}\ a\ \mathrm{B},\ \boxed{2}\,\boxed{1}\,x\rangle, \\
& \mathrm{A} & \stackrel{0.5}{\rightarrow} & \langle b\ \mathrm{C},\ y\,\boxed{1}\rangle, \\
& \mathrm{B} & \stackrel{0.6}{\rightarrow} & \langle a\ b\ c,\ z\ z\rangle, \\
& \mathrm{B} & \stackrel{0.4}{\rightarrow} & \langle a\ b\ c,\ x\ y\ z\rangle, \\
& \mathrm{C} & \stackrel{0.8}{\rightarrow} & \langle \mathrm{A}\ \mathrm{A},\ \boxed{1}\,\boxed{2}\rangle, \\
& \mathrm{C} & \stackrel{0.2}{\rightarrow} & \langle c, z\rangle \qquad \}
\end{array}
$$

Figure 2.4: Example of a weighted synchronous context-free grammar (WSCFG). Note that C is the start symbol.

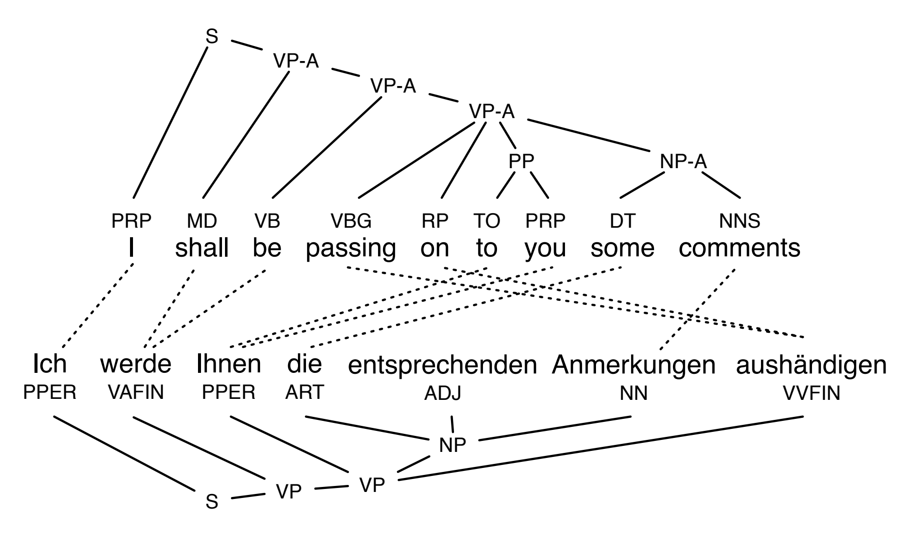
Yield	$\alpha \rightarrow \langle\beta,\gamma\rangle$	Weight
$\langle \mathrm{C},\ \boxed{1}\rangle \Rightarrow$	$\mathrm{C} \rightarrow \langle \mathrm{A}\ \mathrm{A},\ \boxed{1}\,\boxed{2}\rangle$	1.0×0.8
$\langle \mathrm{A}\ \mathrm{A},\ \boxed{1}\,\boxed{2}\rangle \Rightarrow$	$\mathrm{A} \rightarrow \langle \mathrm{A}\ a\ \mathrm{B},\ \boxed{2}\,\boxed{1}\,x\rangle$	$\times 0.5$
$\langle \mathrm{A}\ a\ \mathrm{B}\ \mathrm{A},\ \boxed{2}\,\boxed{1}\,x\,\boxed{3}\rangle \Rightarrow$	$\mathrm{A} \rightarrow \langle b\ \mathrm{C},\ y\,\boxed{1}\rangle$	$\times 0.5$
$\langle b\ \mathrm{C}\ a\ \mathrm{B}\ \mathrm{A},\ \boxed{2}\,y\,\boxed{1}\,x\,\boxed{3}\rangle \Rightarrow$	$\mathrm{C} \rightarrow \langle c,\ z\rangle$	$\times 0.2$
$\langle b\ c\ a\ \mathrm{B}\ \mathrm{A},\ \boxed{1}\,y\ z\ x\,\boxed{2}\rangle \Rightarrow$	$\mathrm{B} \rightarrow \langle a\ b\ c,\ x\ y\ z\rangle$	$\times 0.4$
$\langle b\ c\ a\ a\ b\ c\ \mathrm{A},\ x\ y\ z\ y\ z\ x\,\boxed{1}\rangle \Rightarrow$	$\mathrm{A} \rightarrow \langle b\ \mathrm{C},\ y\,\boxed{1}\rangle$	$\times 0.5$
$\langle b\ c\ a\ a\ b\ c\ b\ \mathrm{C},\ x\ y\ z\ y\ z\ x\ y\,\boxed{1}\rangle \Rightarrow$	$\mathrm{C} \rightarrow \langle c,\ z\rangle$	$\times 0.2$
$\langle b\ c\ a\ a\ b\ c\ b\ c,\ x\ y\ z\ y\ z\ x\ y\ z\rangle$		$= 0.0016$

Figure 2.5: Example synchronous derivation using the WSCFG shown in Figure 2.4.

- so far assumed no particular syntax theory
- can use real syntactic annotation
- will need to store internal structure in the rule

Benefits:

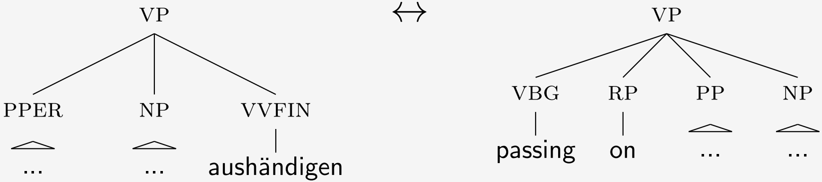
- input language syntax puts some constrains on the extracted rules
- output language can have a better-formed (syntactically) structure



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

Reordering Rule

- Subtree alignment



- Synchronous grammar rule

$$\text{VP} \rightarrow \text{PPER}_1 \ \text{NP}_2 \ \textsf{aushändigen} \ \Big| \ \textsf{passing on} \ \text{PP}_1 \ \text{NP}_2$$

- Note:
  - ➡ one word *aushändigen* mapped to two words *passing on*
  - ➡ effortless capture of reordering
  - ➡ but: fully non-terminal rule not possible
    (one-to-one mapping constraint for nonterminals)

- Subtree alignment

$$\text{PRO} \to \text{Ihnen} \quad \leftrightarrow \quad \text{PP} \to (\text{TO} \to \text{to})\ (\text{PRP} \to \text{you})$$

- Synchronous grammar rule (stripping out English internal structure)

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

- Rule with internal structure

$$\text{PRO}/\text{PP} \quad \to \quad \text{Ihnen} \quad \Big| \quad (\text{TO} \to \text{to}) \quad (\text{PRP} \to \text{you})$$

- Translation of German werde to English shall be

- Translation rule needs to include mapping of VP

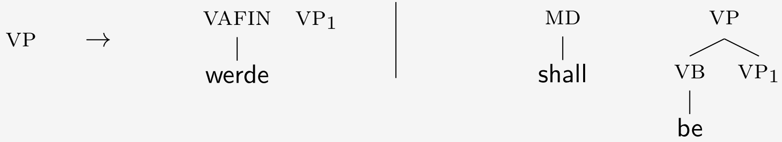
⇒ Complex rule

- Stripping out internal structure

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

$\Rightarrow$ synchronous context free grammar

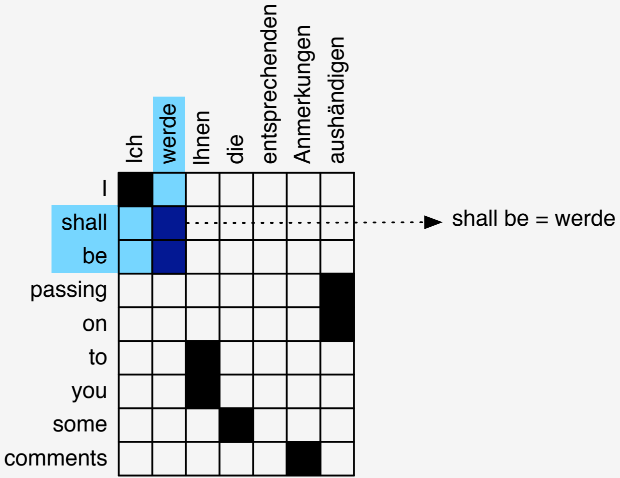
- Maintaining internal structure

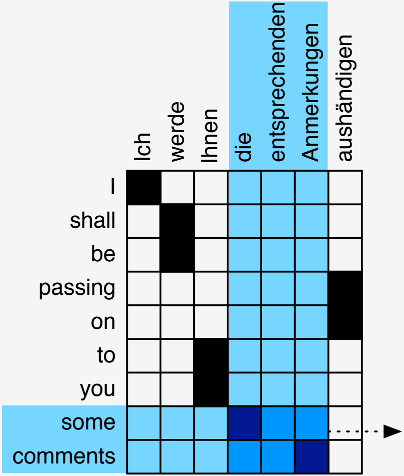


$\Rightarrow$ synchronous tree substitution grammar

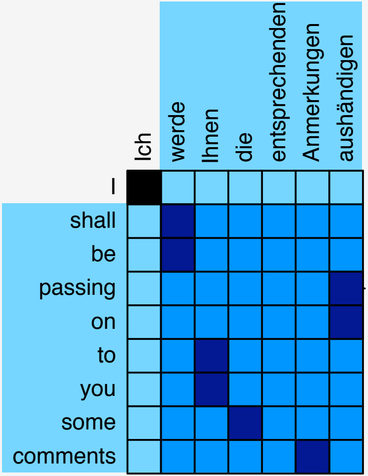
- Extracting rules from a word-aligned parallel corpus
- First: Hierarchical phrase-based model
  - ➡ only **one** non-terminal symbol X
  - ➡ no linguistic syntax, just a formally syntactic model
- Then: Synchronous phrase structure model
  - ➡ non-terminals for words and phrases: NP, VP, PP, ADJ, ...
  - ➡ corpus must be parsed with syntactic parser

- Suppose we want to learn a rule for *werde ... aushändigen*
- phrase-based SMT will probably fail here
  - ➡ the gap is too large, likely inconsistent
  - ➡ if extracted will contain all words in between (rarely applicable)

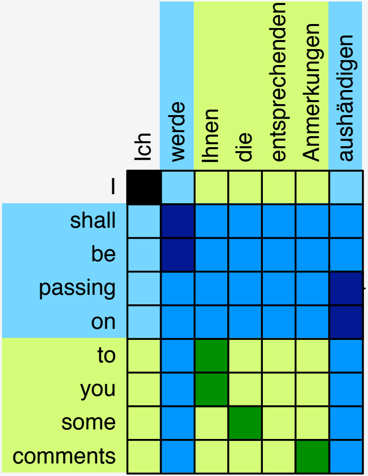

Ich
werde
Ihnen
die
entsprechenden
Anmerkungen
aushändigen
I
shall
be
passing
on
to
you
some
comments
shall be = werde



some comments =
die entsprechenden Anmerkungen



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



subtracting subphrase

werde X aushändigen
= shall be passing on X

- 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 $(\bar{e}, \bar{f})$

- Extend recursively:

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

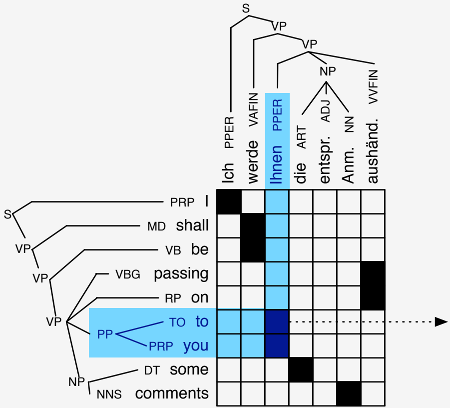
  (note: any of e_{PRE}, e_{POST}, f_{PRE}, or f_{POST} may be empty)
- Set of hierarchical phrase pairs is the closure under this extension mechanism

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

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

- Typical restrictions to limit complexity [Chiang, 2005], to avoid exponential explosion
  - ➡ at most 2 nonterminal symbols
  - ➡ at least 1 but at most 5 words per language
  - ➡ span at most 15 words (counting gaps)
  - ➡ no 2 non-terminals are next to each other in both languages

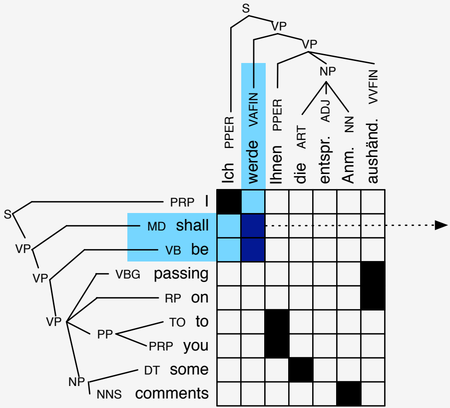
Even without syntax tree-based models often gain about 1-2 BLEU points over phrase-based systems.


S
VP
VP
VP
NP
Ich PPER
werde VAFIN
Ihnen PPER
die ART
entspr. ADJ
Anm. NN
aushänd. VVFIN
S
VP
VP
VP
PRP I
MD shall
VB be
VBG passing
RP on
PP
TO to
PRP you
NP
DT some
NNS comments

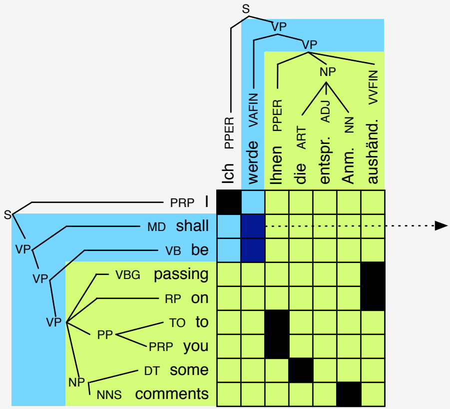

PRO
Ihnen
=
PP
TO
PRP
to
you

- Hierarchical: rule can cover any span
  $\Leftrightarrow$ syntactic rules must cover constituents in the tree $\Rightarrow$ 1 node on top
- Hierarchical: gaps may cover any span
  $\Leftrightarrow$ gaps must cover constituents in the tree

- Moving up the tree introduces non-terminals
- Much less rules are extracted (all things being equal)

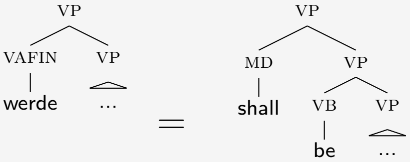


English span not a constituent
no rule extracted



Rule with this phrase pair

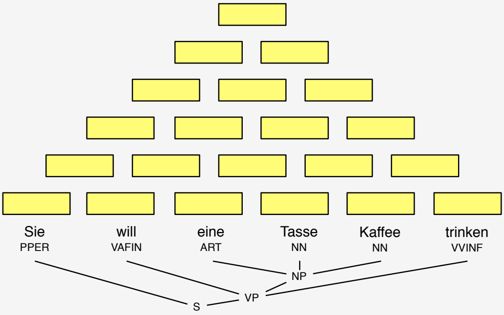
requires syntactic context



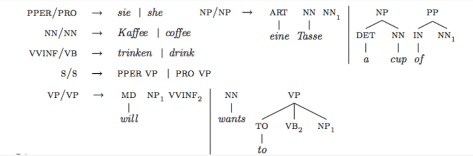
- Extract all rules from corpus
- Score based on counts
  - joint rule probability: $p(\mathrm{LHS}, \mathrm{RHS}_f, \mathrm{RHS}_e)$
  - rule application probability: $p(\mathrm{RHS}_f, \mathrm{RHS}_e|\mathrm{LHS})$
  - direct translation probability: $p(\mathrm{RHS}_e|\mathrm{RHS}_f, \mathrm{LHS})$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

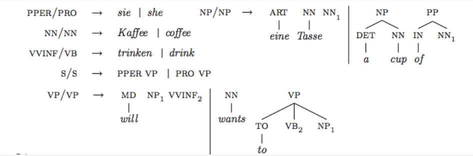
Inspired by monolingual syntactic chart parsing:

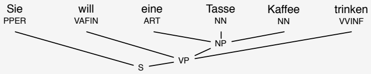
During decoding of the source sentence,
a chart with translations for the $O(n^2)$ spans has to be filled

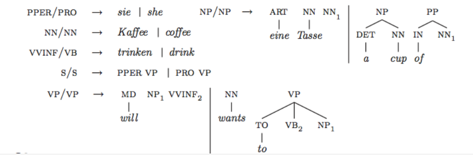


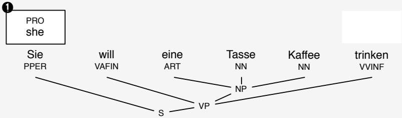
note: constrains limit the branching factor


PPER/PRO → sie | she
NN/NN → Kaffee | coffee
VVINF/VB → trinken | drink
S/S → PPER VP | PRO VP
VP/VP → MD NP1 VVINF2
will
NN
wants
VP
TO VB2 NP1
to
NP/NP → ART NN NN1
eine Tasse
NP
DET NN
a cup
PP
IN NN1
of

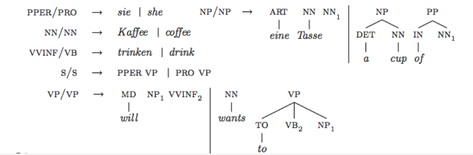

PPER/PRO → sie | she
NN/NN → Kaffee | coffee
VVINF/VB → trinken | drink
S/S → PPER VP | PRO VP
VP/VP → MD NP$_1$ VVINF$_2$ | NN VP
will
wants
TO VB$_2$ NP$_1$
to
NP/NP → ART NN NN$_1$ | NP PP
eine Tasse
DET NN IN NN$_1$
a cup of

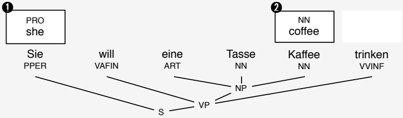

Sie
PPER
will
VAFIN
eine
ART
Tasse
NN
Kaffee
NN
trinken
VVINF
NP
VP
S



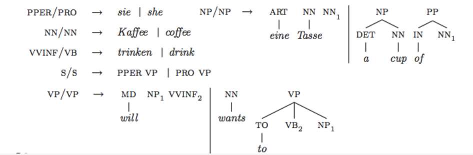


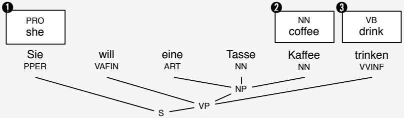
Purely lexical rule: filling a span with a translation (a constituent in the chart)



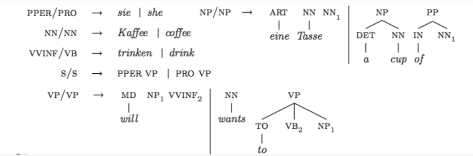


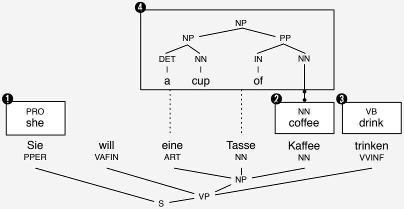
Purely lexical rule: filling a span with a translation (a constituent in the chart)



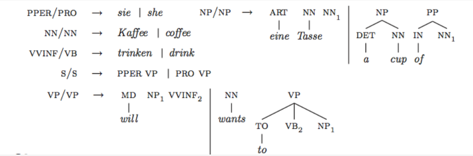


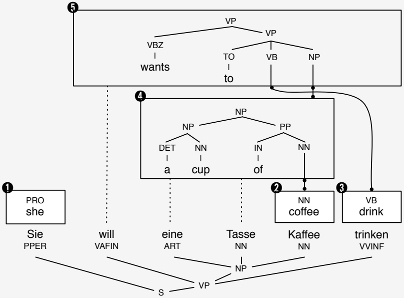
Purely lexical rule: filling a span with a translation (a constituent in the chart)



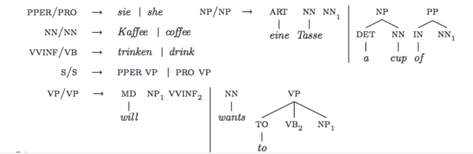


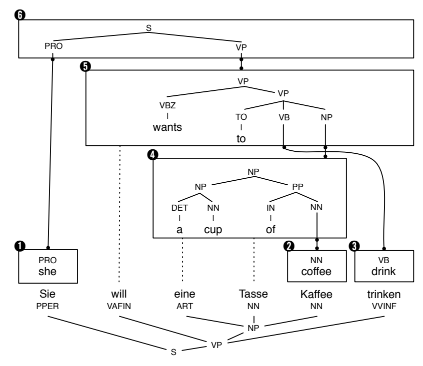
Complex rule: matching underlying constituent spans, and covering words

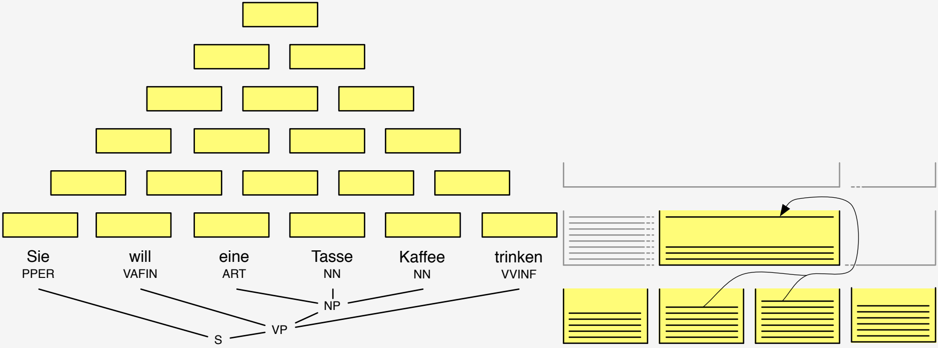




Complex rule with reordering

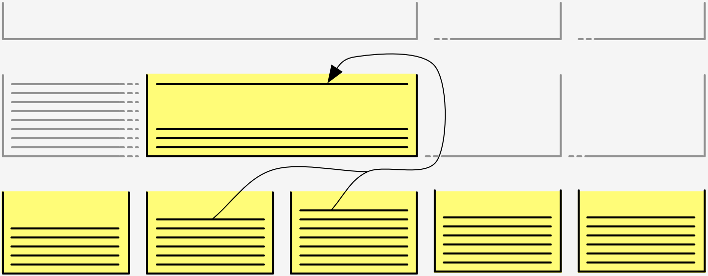

PPER/PRO → sie | she
NN/NN → Kaffee | coffee
VVINF/VB → trinken | drink
S/S → PPER VP | PRO VP
VP/VP → MD NP1 VVINF2 will
NN wants
VP TO VB2 NP1 to
NP/NP → ART NN NN1 eine Tasse
NP PP DET NN IN NN1 a cup of


S
PRO
VP
VP
VP
VBZ
wants
TO
to
VB
NP
NP
NP
PP
DET
a
NN
cup
IN
of
NN
PRO
she
NN
coffee
VB
drink
Sie
PPER
will
VAFIN
eine
ART
Tasse
NN
Kaffee
NN
trinken
VVINF
NP
VP
S



- Chart consists of cells that cover contiguous spans over the input sentence
- Each cell contains a set of hypotheses
- Hypothesis = translation of span with target-side constituent

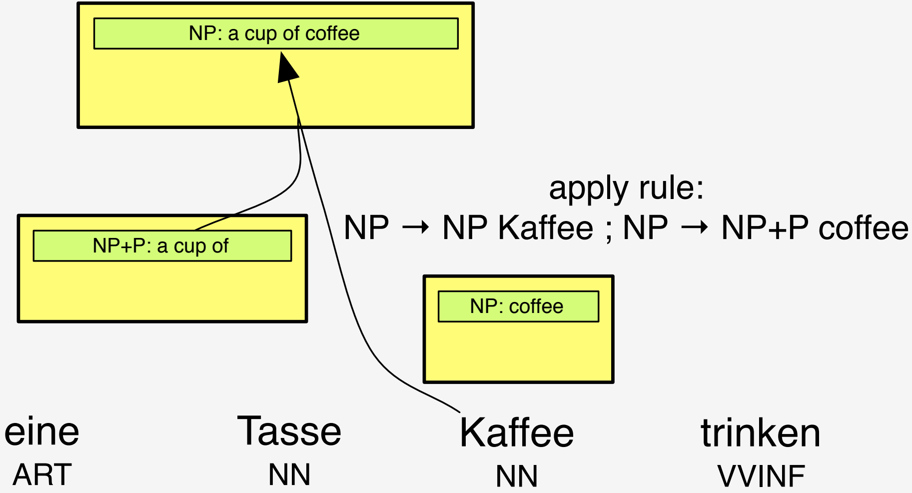
- For each span, a stack of (partial) translations is maintained
- Bottom-up: a higher stack is filled, once underlying stacks are complete



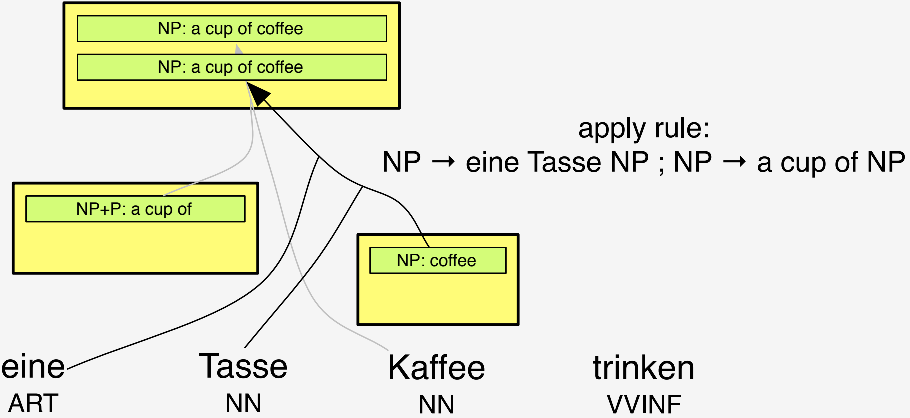
Input: Foreign sentence $\mathbf{f} = f_1, ... f_{l_f}$, with syntax tree
Output: English translation **e**

```
for all spans [start,end] (bottom up) do
    for all sequences s of hypotheses and words in span [start,end] do
        for all rules r do
            if rule r applies to chart sequence s then
                create new hypothesis c
                add hypothesis c to chart
            end if
        end for
    end for
end for
return English translation e from best hypothesis in span [0,l_f]
```

Applying rule creates new hypothesis



Another hypothesis



Both hypotheses are indistiguishable in future search

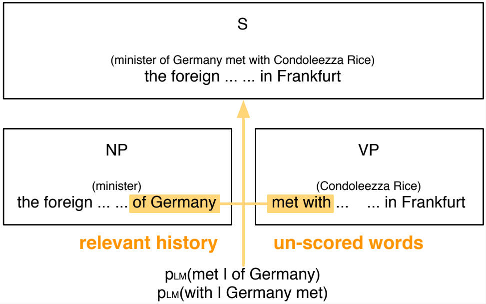
→ can be recombined

Hypotheses have to match in

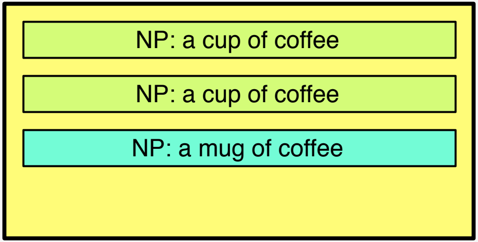
- span of input words covered
- output constituent label
- first n–1 output words (not properly scored, since they lack context)
- last n–1 output words (still affect scoring of subsequently added words, just like in phrase-based decoding)

(n is the order of the n-gram language model)

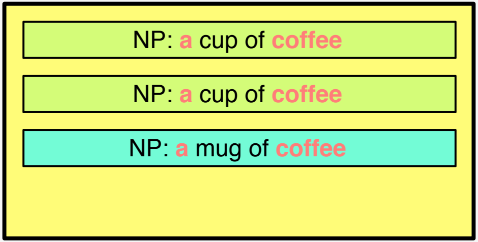
When merging hypotheses, internal language model contexts are absorbed



Recombinable?



Recombinable?



Yes, iff max. 2-gram language model is used