11-731 Machine Translation

Syntax-Based Translation Models – Principles, Approaches, Acquisition

Alon Lavie

March 2014

With Acknowledged Contributions from:

- Marcello Federico, Gabriele Musillo and Philipp Koehn (MT Marathon 2011)
- Adam Lopez, Matt Post and CCB (JHU)

Outline

- Syntax-based Translation Models: Rationale and Motivation
- Synchronous Context-Free Grammars (S-CFGs)
- Resource Scenarios and Model Definitions
  - String-to-Tree, Tree-to-String and Tree-to-Tree
- Inversion Transduction Grammars (ITGs)
- Hierarchical Phrase-based Models (Chiang's Hiero)
- Syntax-Augmented Hierarchical Models (Venugopal and Zollmann)
- String-to-Tree Models
  - Phrase-Structure-based Model (Galley et al., 2004, 2006)
- Tree-to-Tree Models
  - Phrase-Structure-based Model (Hanneman, Burroughs and Lavie, 2011)
  - Tree Transduction Models (Yamada and Knight, Gildea et al.)

Syntax-based Models: Rationale

- Phrase-based models model translation at very shallow levels:
  - Translation equivalence modeled at the multi-word lexical level
  - Phrases capture some cross-language local reordering, but only for phrases that were seen in training – No effective generalization
  - Non-local cross-language reordering is modeled only by permuting order of phrases during decoding
  - No explicit modeling of syntax or structural divergences between the two languages
- **Goal:** Improve translation quality using syntax-based models
  - Capture generalizations, reorderings and language divergences at appropriate levels of abstraction
  - Models direct the search during decoding to more accurate translations
  - Still **Statistical MT**: Acquire translation models automatically from (annotated) parallel-data and model them statistically!

Syntax-based Statistical MT

- Building a syntax-based Statistical MT system:
  - Similar in concept to simpler phrase-based SMT methods:
  - **Model Acquisition** from bilingual sentence-parallel corpora
  - **Decoders** that given an input string can find the best translation according to the models
- Our focus this week will be on the different types of models and their **acquisition**
- **Next week (after Spring Break):** Chris Dyer will cover decoding for hierarchical and syntax-based MT

Syntax-based Resources vs. Models

- Important Distinction:
  1. What **structural information** for the parallel-data is available during model acquisition and training?
  2. What **type of translation models** are we acquiring from the annotated parallel data?
- Structure available during Acquisition – Main Distinctions:
  - Syntactic/structural information for the parallel training data:
    - Given by external components (parsers) or inferred from the data?
    - Syntax/Structure available for one language or for both?
    - Phrase-Structure, Dependency-Structure, other annotations?
- What do we extract from parallel-sentences?
  - Sub-sentential units of **translation equivalence** annotated with structure
  - **Rules/structures** that determine how these units combine into full transductions

Structure Available During Acquisition

- What information/annotations are available for the bilingual sentence-parallel training data?
  - (Symetrized) Viterbi Word Alignments (i.e. from GIZA++)
  - (Non-syntactic) extracted phrases for each parallel sentence
  - Parse-trees/dependencies for “source” language
  - Parse-trees/dependencies for “target” language
- **Some major potential issues and problems:**
  - GIZA++ word alignments are not aware of syntax – word-alignment errors can have bad consequences on the extracted syntactic models
  - Using external monolingual parsers is also problematic:
    - Using single-best parse for each sentence introduces parsing errors
    - Parsers were designed for monolingual parsing, not translation
    - Parser design decisions for each language may be very different:
      - Different notions of constituency and structure
      - Different sets of POS and constituent labels

Synchronous Context-Free Grammars (S-CFGs)

TG (Lewis and Stearns, 1968; Aho and Ullman, 1969):

- **two or more strings derived simultaneously**
- more powerful than FSTs
- used in NLP to model **alignments**, unbounded **reordering**, and mappings from surface forms to logical forms

Synchronous Rules:

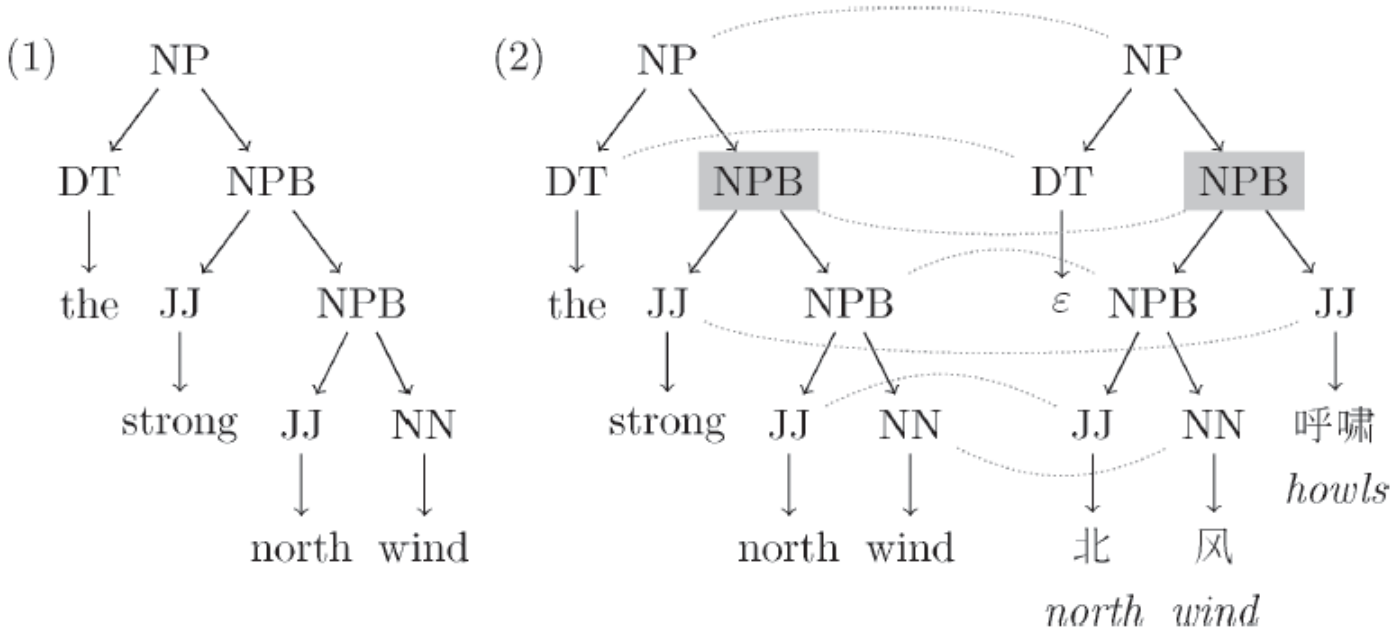
- left-hand side nonterminal symbol associated with **source** and **target** right-hand sides
- **bijection** $[]$ mapping nonterminals in source and target of right-hand sides

$$\begin{cases} E \rightarrow E_{[1]} + E_{[3]} \;/\; + E_{[1]}\ E_{[3]} & \text{infix to Polish notation} \\ E \rightarrow E_{[1]} * E_{[2]} \;/\; * E_{[1]}\ E_{[2]} & \\ E \rightarrow n \;/\; n & n \in N \end{cases}$$

Synchronous Context-Free Grammars (S-CFGs)

$$\begin{aligned}
\text{NP} &\longrightarrow \text{DT}_{[1]}\text{NPB}_{[2]} \;/\; \text{DT}_{[1]}\text{NPB}_{[2]} \\
\text{NPB} &\longrightarrow \text{JJ}_{[1]}\text{NN}_{[2]} \;/\; \text{JJ}_{[1]}\text{NN}_{[2]} \\
\text{NPB} &\longrightarrow \text{NPB}_{[1]}\text{JJ}_{[2]} \;/\; \text{JJ}_{[2]}\text{NPB}_{[1]} \\
\text{DT} &\longrightarrow \text{the} \;/\; \varepsilon \\
\text{JJ} &\longrightarrow \text{strong} \;/\; \text{呼啸} \\
\text{JJ} &\longrightarrow \text{north} \;/\; \text{北} \\
\text{NN} &\longrightarrow \text{wind} \;/\; \text{风}
\end{aligned}$$

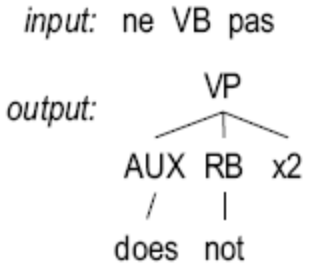
- **1-to-1 correspondence** between nodes
- **isomorphic** derivation trees
- uniquely determined **word alignment**



Syntax-based Translation Models

- **String-to-Tree:**
  - Models explain how to transduce **a string** in the source language into a **structural representation** in the target language
  - During decoding:
    - No separate parsing on source side
    - Decoding results in set of possible translations, each annotated with syntactic structure
    - The best-scoring string+structure can be selected as the translation
- Example:

ne VB pas → (VP (AUX (does)) (RB (not)) x2)



Syntax-based Translation Models

- **Tree-to-String:**
  - Models explain how to transduce a **structural representation** of the source language input into **a string** in the target language
  - During decoding:
    - Parse the source string to derive its structure
    - Decoding explores various ways of decomposing the parse tree into a sequence of composable models, each generating a translation string on the target side
    - The best-scoring string can be selected as the translation
- Examples:

No.	Rule		
(1)	(IP (NP) (VP) (PU))	X_1 X_2 X_3	1:1 2:2 3:3
(2)	(NP (NN 枪手))	The gunman	1:1 1:2
(3)	(VP (SB 被) (VP (NP (NN)) (VV 击毙)))	was killed by X	1:1 2:4 3:2
(4)	(NN 警方)	police	1:1
(5)	(PU 。)	.	1:1

Syntax-based Translation Models

- **Tree-to-Tree:**
  - Models explain how to transduce a **structural representation** of the source language input into **a structural representation** in the target language
  - During decoding:
    - Decoder **synchronously** explores alternative ways of parsing the source-language input string and **transduce** it into corresponding target-language structural output.
    - The best-scoring structure+string can be selected as the translation
- Example:

```
NP::NP [VP 北 CD 有 邦交 ] → [one of the CD countries that VP]
(
;; Alignments
(X1::Y7)
(X3::Y4)
)
```

Inversion Transduction Grammars (ITGs)

BTG (Wu, 1997):

- special form of SCFG
- only one nonterminal X
- nonterminal rules:

$$\begin{cases} X \rightarrow X_{[1]}\ X_{[2]}\ /\ X_{[1]}\ X_{[2]} & \textbf{monotone} \text{ rule} \\ X \rightarrow X_{[1]}\ X_{[2]}\ /\ X_{[2]}\ X_{[1]} & \textbf{inversion} \text{ rule} \end{cases}$$

- preterminal rules where $e \in V_t \cup \{\epsilon\}$ and $f \in V_s \cup \{\epsilon\}$:

$$\{ X \rightarrow f\ /\ e \quad \text{lexical translation rules}$$

Inversion Transduction Grammars (ITGs)

$$\langle X, X \rangle \Rightarrow_{X_1 X_2 / X_2 X_1} \langle X_1 X_2, X_2 X_1 \rangle$$

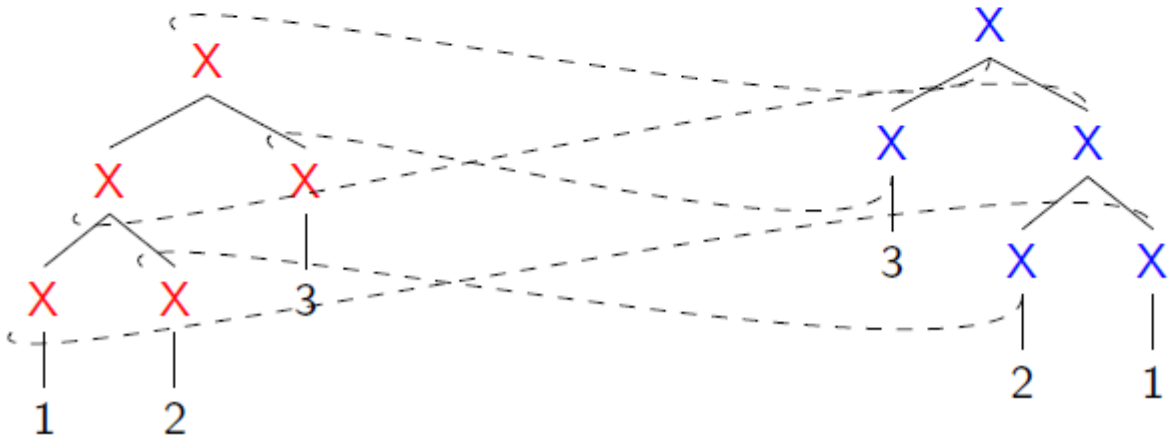
$$\Rightarrow_{X_1 X_2 / X_2 X_1} \langle X_3 X_4 X_2, X_2 X_4 X_3 \rangle$$

re-indexed symbols

$$\Rightarrow_{1/1} \langle 1 X_4 X_2, X_2 X_4 1 \rangle$$

$$\Rightarrow_{2/2} \langle 12 X_2, X_2 21 \rangle$$

$$\Rightarrow_{3/3} \langle 123, 321 \rangle$$



Hierarchical Phrase-Based Models

- Proposed by David Chiang in 2005
- Natural hierarchical extension to phrase-based models
- **Representation:** rules in the form of synchronous CFGs
  - Formally syntactic, but with no direct association to linguistic syntax
  - Single non-terminal “X”
- **Acquisition Scenario:** Similar to standard phrase-based models
  - No independent syntactic parsing on either side of parallel data
  - Uses “symetricized” bi-directional viterbi word alignments
  - Extracts phrases and rules (hierarchical phrases) from each parallel sentence
  - Models the extracted phrases statistically using MLE scores

Hierarchical Phrase-Based Models

Aozhou	shi	yu	Beihan	you	bangjiao	de	shaoshu	guojia	zhiyi	.
Australia	is	with	North Korea	have	dipl. rels.	that	few	countries	one of	.

Typical Phrase-Based Chinese-English Translation:

[Aozhou] [shi]$_1$ [yu Beihan]$_2$ [you] [bangjiao] [de shaoshu guojia zhiyi] [.]

[Australia] [has] [dipl. rels.] [with North Korea]$_2$ [is]$_1$ [one of the few countries] [.]

- Chinese VPs follow PPs / English VPs precede PPs

$$yu\ X_1\ you\ X_2\ /\ have\ X_2\ with\ X_1$$

- Chinese NPs follow RCs / English NPs precede RCs

$$X_1\ de\ X_2\ /\ the\ X_2\ that\ X_1$$

- translation of *zhiyi* construct in English word order

$$X_1\ zhiyi\ /\ one\ of\ X_1$$

Hierarchical Phrase-Based Models

- Example:
  - Chinese-to-English Rules:

(10) $X \rightarrow \langle \text{yu } X_{[1]} \text{ you } X_{[2]}, \text{have } X_{[2]} \text{ with } X_{[1]} \rangle$

(11) $X \rightarrow \langle X_{[1]} \text{ de } X_{[2]}, \text{the } X_{[2]} \text{ that } X_{[1]} \rangle$

(12) $X \rightarrow \langle X_{[1]} \text{ zhiyi}, \text{one of } X_{[1]} \rangle$

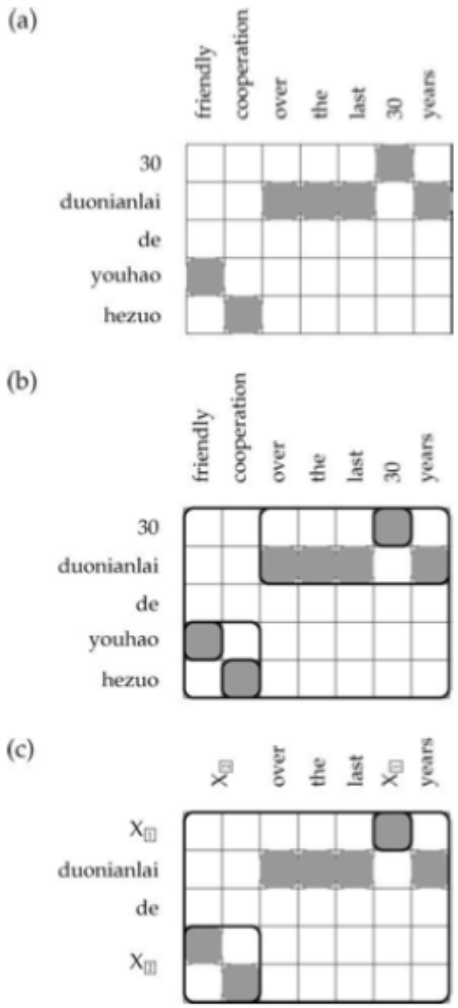
$\langle S_{[1]}, S_{[1]} \rangle \Rightarrow \langle S_{[2]} X_{[3]}, S_{[2]} X_{[3]} \rangle$
$\Rightarrow \langle S_{[4]} X_{[5]} X_{[3]}, S_{[4]} X_{[5]} X_{[3]} \rangle$
$\Rightarrow \langle X_{[6]} X_{[7]} X_{[3]}, X_{[6]} X_{[7]} X_{[3]} \rangle$
$\Rightarrow \langle \text{Aozhou } X_{[7]} X_{[3]}, \text{Australia } X_{[7]} X_{[3]} \rangle$
$\Rightarrow \langle \text{Aozhou shi } X_{[3]}, \text{Australia is } X_{[3]} \rangle$
$\Rightarrow \langle \text{Aozhou shi } X_{[7]} \text{ zhiyi}, \text{Australia is one of } X_{[7]} \rangle$
$\Rightarrow \langle \text{Aozhou shi } X_{[8]} \text{ de } X_{[9]} \text{ zhiyi}, \text{Australia is one of the } X_{[9]} \text{ that } X_{[8]} \rangle$
$\Rightarrow \langle \text{Aozhou shi yu } X_{[1]} \text{ you } X_{[2]} \text{ de } X_{[9]} \text{ zhiyi}, \text{Australia is one of the } X_{[9]} \text{ that have } X_{[2]} \text{ with } X_{[1]} \rangle$

Figure 1: Example partial derivation of a synchronous CFG.

Hierarchical Phrase-Based Models

- Extraction Process Overview:
  1. Start with standard phrase extraction from symetricized viterbi word-aligned sentence-pair
  2. For each phrase-pair, find all embedded phrase-pairs, and create a hierarchical rule for each instance
  3. Accumulate collection of all such rules from the entire corpus along with their counts
  4. Model them statistically using maximum likelihood estimate (MLE) scores:
     - P(target|source) = count(source,target)/count(source)
     - P(source|target) = count(source,target)/count(target)
  5. Filtering:
     - Rules of length < 5 (terminals and non-terminals)
     - At most **two** non-terminals X
     - Non-terminals must be **separated** by a terminal

Hierarchical Phrase-Based Models



Rule Extraction:

a **word-aligned** sentence pair

b extract **initial phrase pairs**

c **replace sub-phrases in phrases with symbol** X

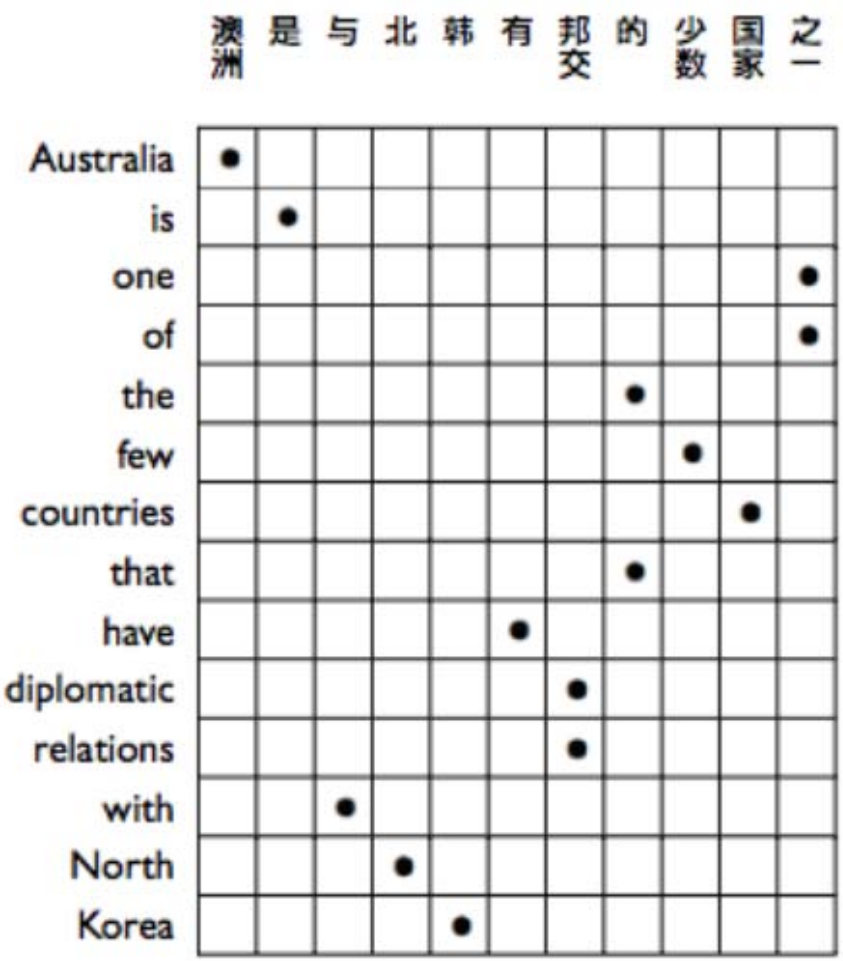
Glue Rules:

$$S \rightarrow S_1 X_2 / S_1 X_2 \quad S \rightarrow X_1 / X_1$$

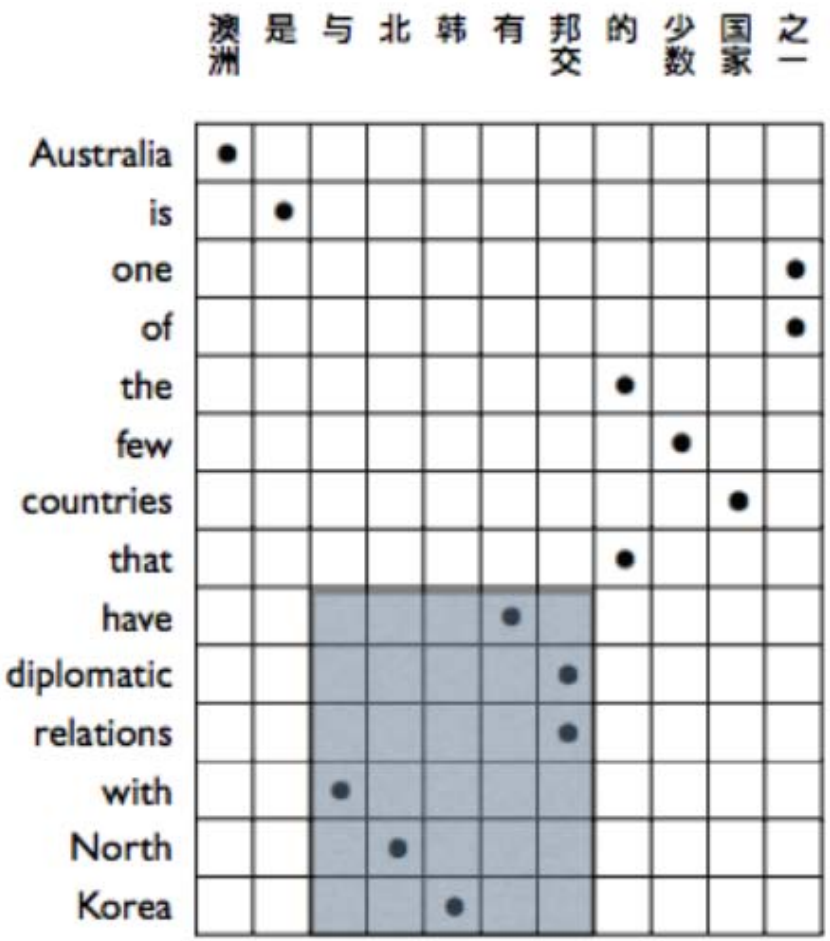
Rule Filtering:

- limited length of initial phrases
- no adjacent nonterminals on source
- at least one pair of aligned words in non-glue rules

Hierarchical Phrase-Based Models

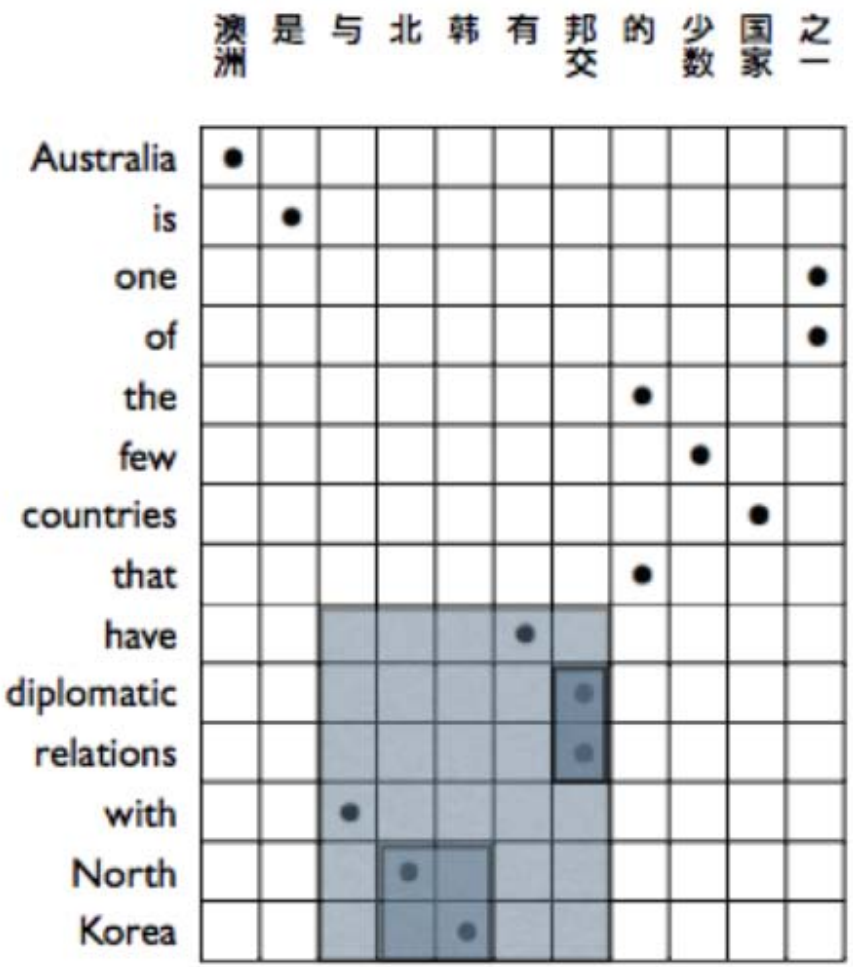


Hierarchical Phrase-Based Models



X → 与 北 韩 有 邦交,
have diplomatic relations
with North Korea

Hierarchical Phrase-Based Models

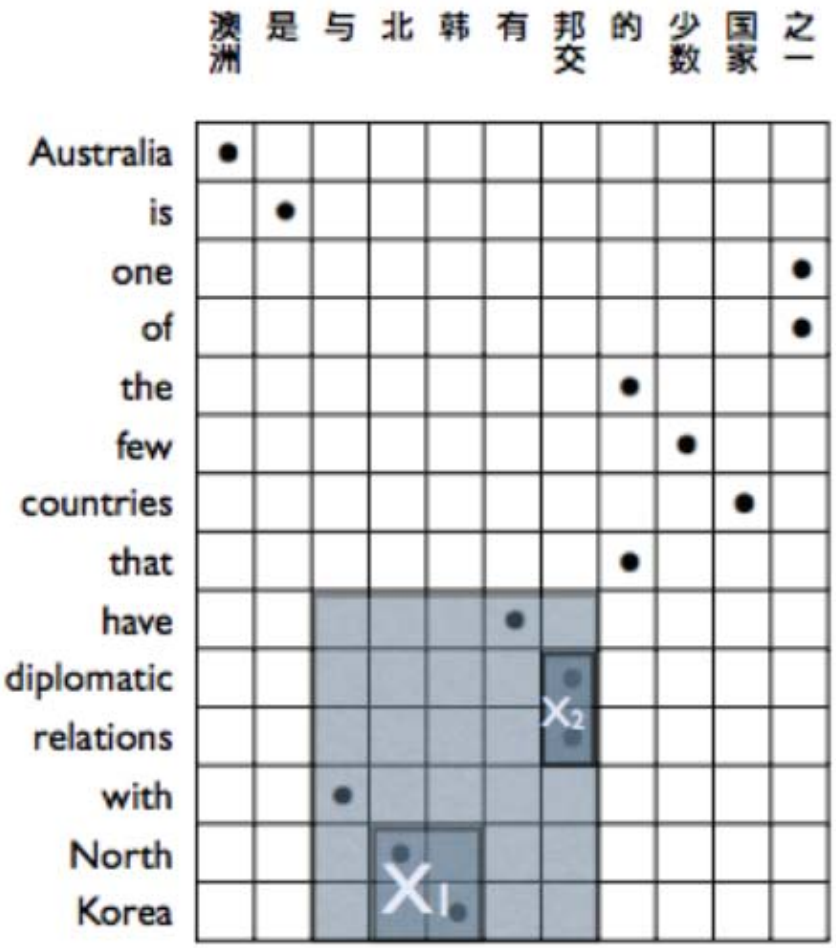


X → 与 北 韩 有 邦交,
have diplomatic relations with North Korea

X → 邦交,
diplomatic relations

X → 北 韩,
North Korea

Hierarchical Phrase-Based Models



X → 与 北 韩 有 邦交,
have diplomatic relations with North Korea

X → 邦交,
diplomatic relations

X → 北 韩,
North Korea

X → 与 X_1 有 X_2,
have X_2 with X_1

Hierarchical Phrase-Based Models

Word Translation Features:

$$h_1(X \rightarrow \alpha/\beta) = \log p(\mathcal{T}_\beta|\mathcal{T}_\alpha)$$

$$h_2(X \rightarrow \alpha/\beta) = \log p(\mathcal{T}_\alpha|\mathcal{T}_\beta)$$

Word Penalty Feature:

$$h_3(X \rightarrow \alpha/\beta) = -|\mathcal{T}_\beta|$$

Synchronous Features:

$$h_4(X \rightarrow \alpha/\beta) = \log p(\beta|\alpha)$$

$$h_5(X \rightarrow \alpha/\beta) = \log p(\alpha|\beta)$$

Glue Penalty Feature:

$$h_6(S \rightarrow S_1X_1/S_1X_1) = -1$$

Phrase Penalty Feature:

$$h_7(X \rightarrow \alpha/\beta) = -1$$

- λ_i tuned on dev set using MERT

Syntax-Augmented Hierarchical Model

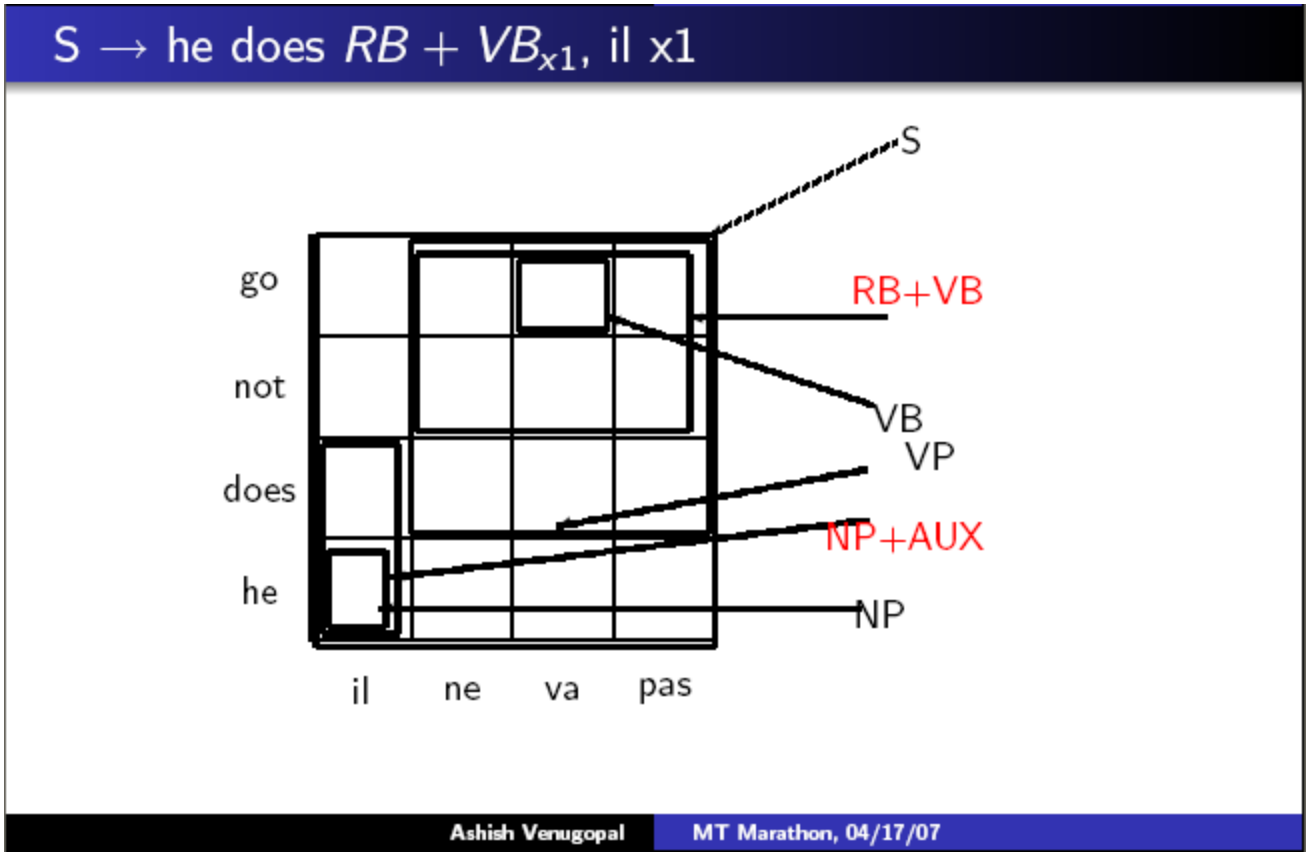
- Proposed by CMU's Venugopal and Zollmann in 2006
- **Representation:** rules in the form of synchronous CFGs
- **Main Goal**: add linguistic syntax to the hierarchical rules that are extracted by the Hiero method:
  - Hiero's "X" labels are completely generic – allow substituting any sub-phrase into an X-hole (if context matches)
  - Linguistic structure has **labeled** constituents – the labels determine what sub-structures are allowed to combine together
  - Idea: use labels that are derived from **parse structures** on one side of parallel data to label the "X" labels in the extracted rules
  - Labels from one language (i.e. English) are "projected" to the other language (i.e. Chinese)
- **Major Issues/Problems**:
  - How to label X-holes that are not complete constituents?
  - What to do about rule "fragmentation" – rules that are the same other than the labels inside them?

Syntax-Augmented Hierarchical Model

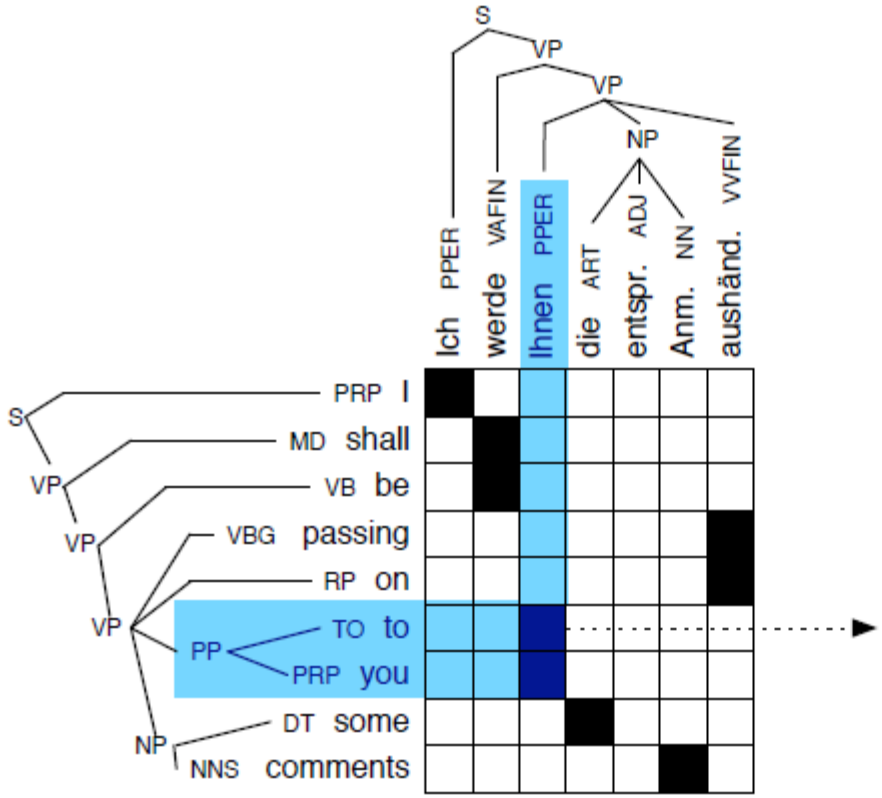
- Extraction Process Overview:
  1. Parse the “strong” side of the parallel data (i.e. English)
  2. Run the Hiero extraction process on the parallel-sentence instance and find all phrase-pairs and all hierarchical rules for parallel-sentence
  3. Labeling: for each X-hole that corresponds to a parse constituent C, label X as C. For all other X-holes, assign combination labels
  4. Accumulate collection of all such rules from the entire corpus along with their counts
  5. Model the rules statistically: Venagopal & Zollman use six different rule score features instead of just two MLE scores.
  6. Filtering: similar to Hiero rule filtering
- Advanced Modeling: **Preference Grammars**
  - **Avoid rule fragmentation**: instead of explicitly labeling the X-holes in the rules with different labels, keep them as “X”, with distributions over the possible labels that could fill the “X”. These are used as features during decoding

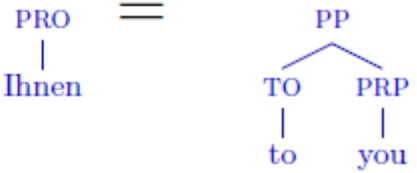
Syntax-Augmented Hierarchical Model

- Example:



Linguistic Syntax-based Models



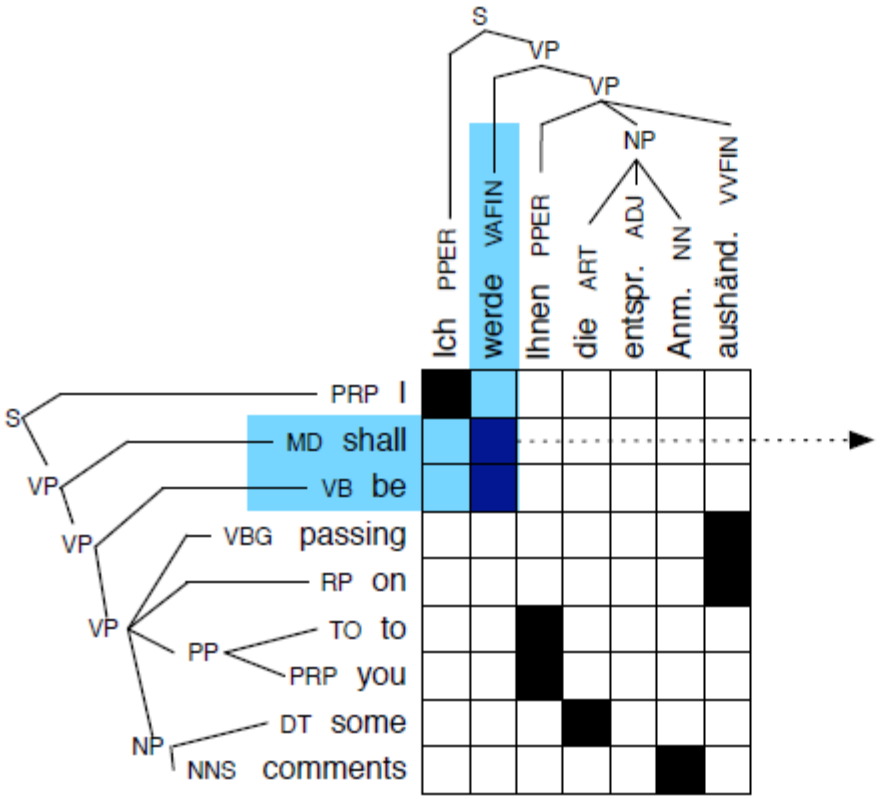


Syntax-based Models

- Syntax and Isomorphism Constraints:
  - Parallel parse-trees are viewed as a compositional combination of syntactic sub-units that are translation equivalents
  - All corresponding source-target units should be translation equivalents: fully supported by the word alignments
  - All lexical phrase-pairs must be valid syntactic constituents in their corresponding trees
  - All non-terminals in extracted rules are decomposition points of smaller syntactic units that are constituents
- All the restrictions we saw in Hierarchical S-CFGs, with additional constraints of syntax from the parse tree(s)
- Fewer phrase-pairs can be extracted
- Much richer grammars

Syntax-based Models

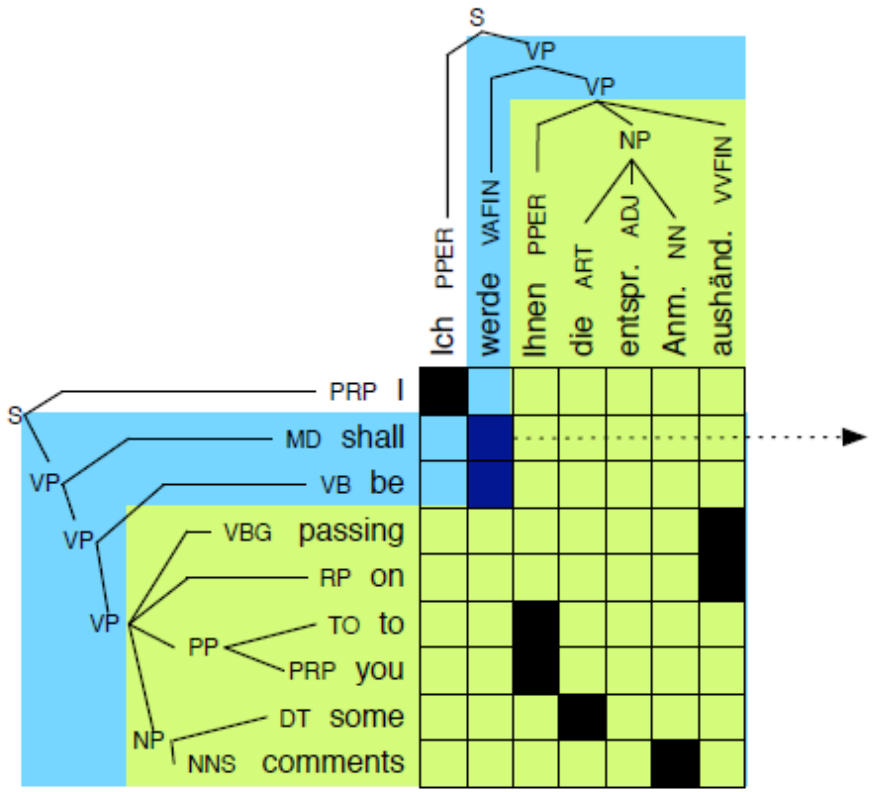
Impossible Rules



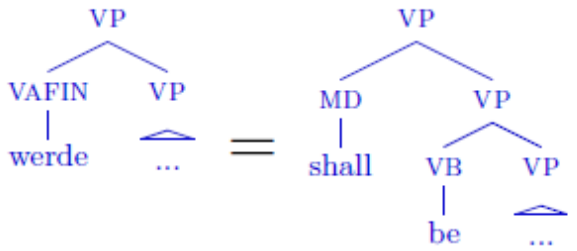
English span not a constituent
no rule extracted

Syntax-based Models

Rules with Context



Rule with this phrase pair requires syntactic context



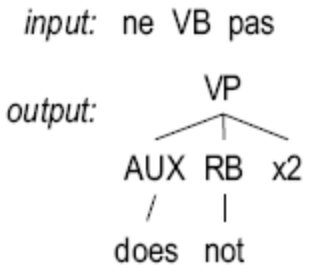
Syntax-based Models

- Extracting all possible syntax-labeled sub-trees and decomposed partial-trees generates an exponential number of phrases and rules
- Typically much too large for effective decoding
- Different approaches to limiting the number of rules:
  - Apply the Hiero-style restrictions on rules: maximum span, at most two non-terminals, at least one lexical anchor, etc.
  - Extract only minimal rules (maximally decompose each training instance): [GHKM 2004] [GHKM 2006]
  - Extract all rules and apply harsh rule-filtering methods: Stat-XFER [Lavie et al, 2009] [Hanneman et al 2011]

String-to-Tree: Galley et al. (GHKM)

- Proposed by Galley et al. in 2004 and improved in 2006
- Idea: model **full syntactic structure** on the **target-side** only in order to produce translations that are more grammatical
- **Representation:** synchronous hierarchical strings on the source side and their corresponding tree fragments on the target side
- **Example:**

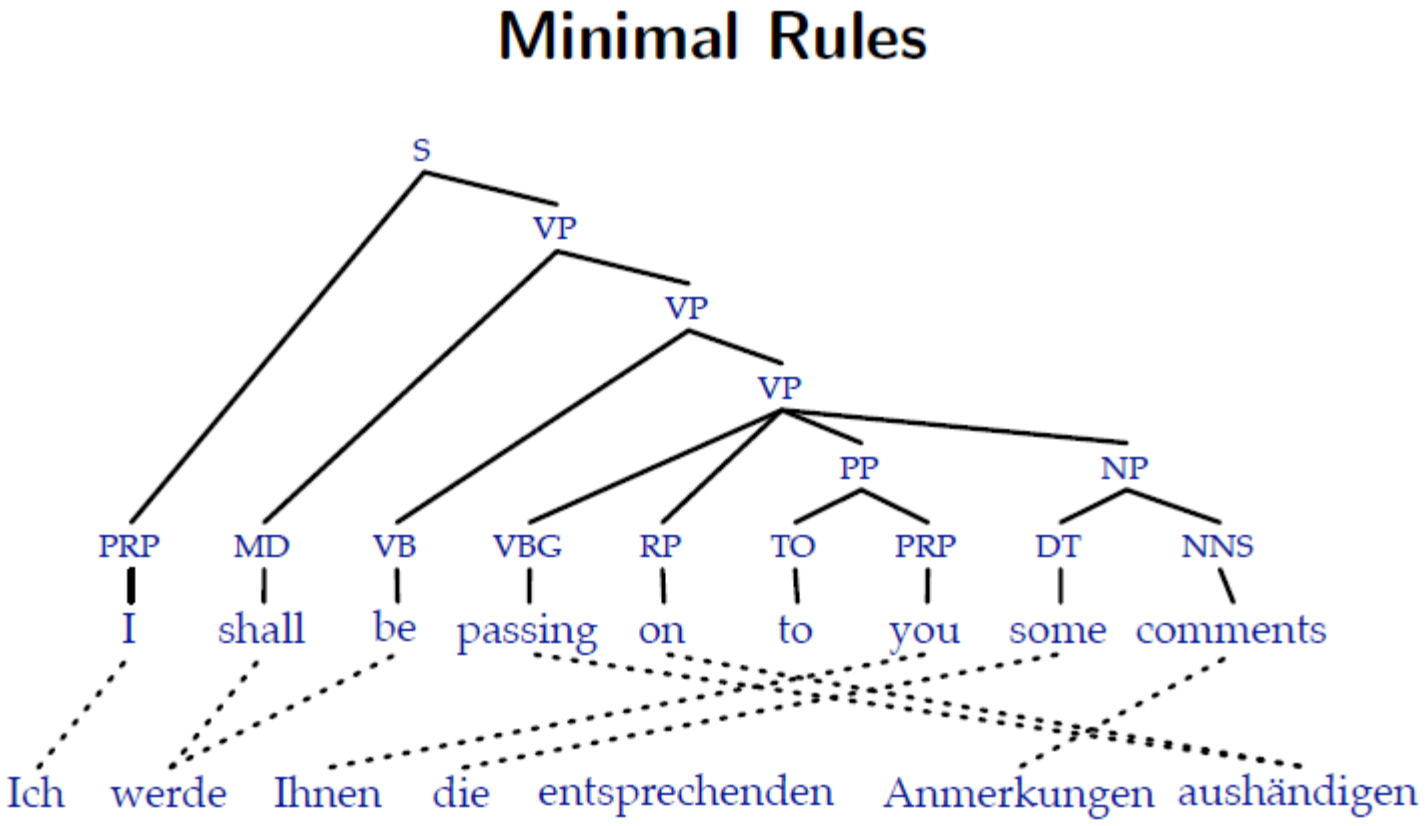
  ne VB pas → (VP (AUX (does) RB (not) x2



String-to-Tree: Galley et al. (GHKM)

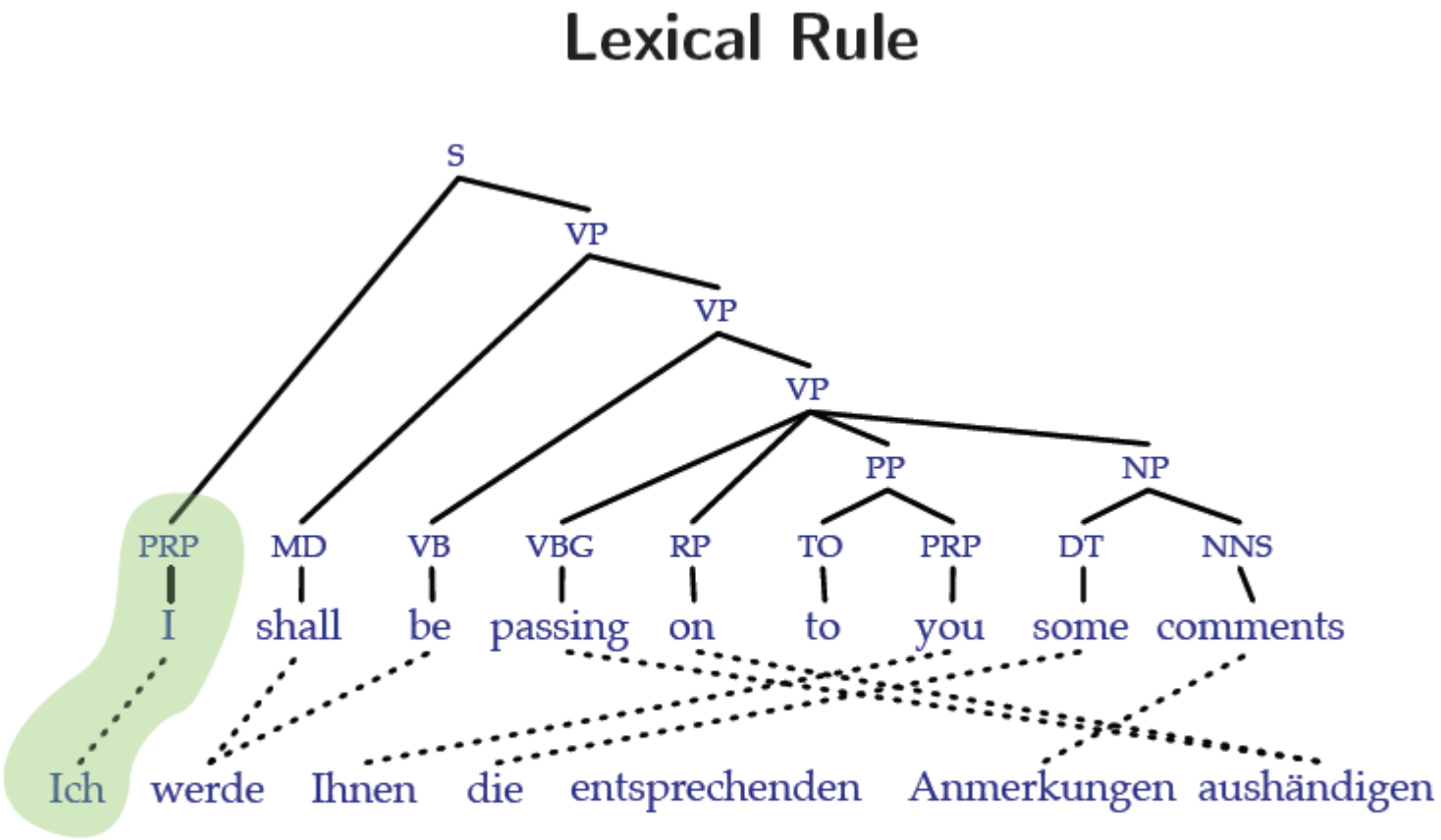
- Overview of Extraction Process:
  1. Obtain symetricized viterbi word-alignments for parallel sentences
  2. Parse the “strong” side of the parallel data (i.e. English)
  3. Find all constituent nodes in the source-language tree that have consistent word alignments to strings in target-language
  4. Treat these as “decomposition” points: extract tree-fragments on target-side along with corresponding “gapped” string on source-side
  5. Labeling: for each “gap” that corresponds to a parse constituent C, label the gap as C.
  6. Accumulate collection of all such rules from the entire corpus along with their counts
  7. Model the rules statistically: initially used “standard” P(tgt|src) MLE scores. Also experimented with other scores, similar to SAMT
- Advanced Modeling: Extraction of **composed rules**, not just **minimal rules**

GHKM Rule Extraction



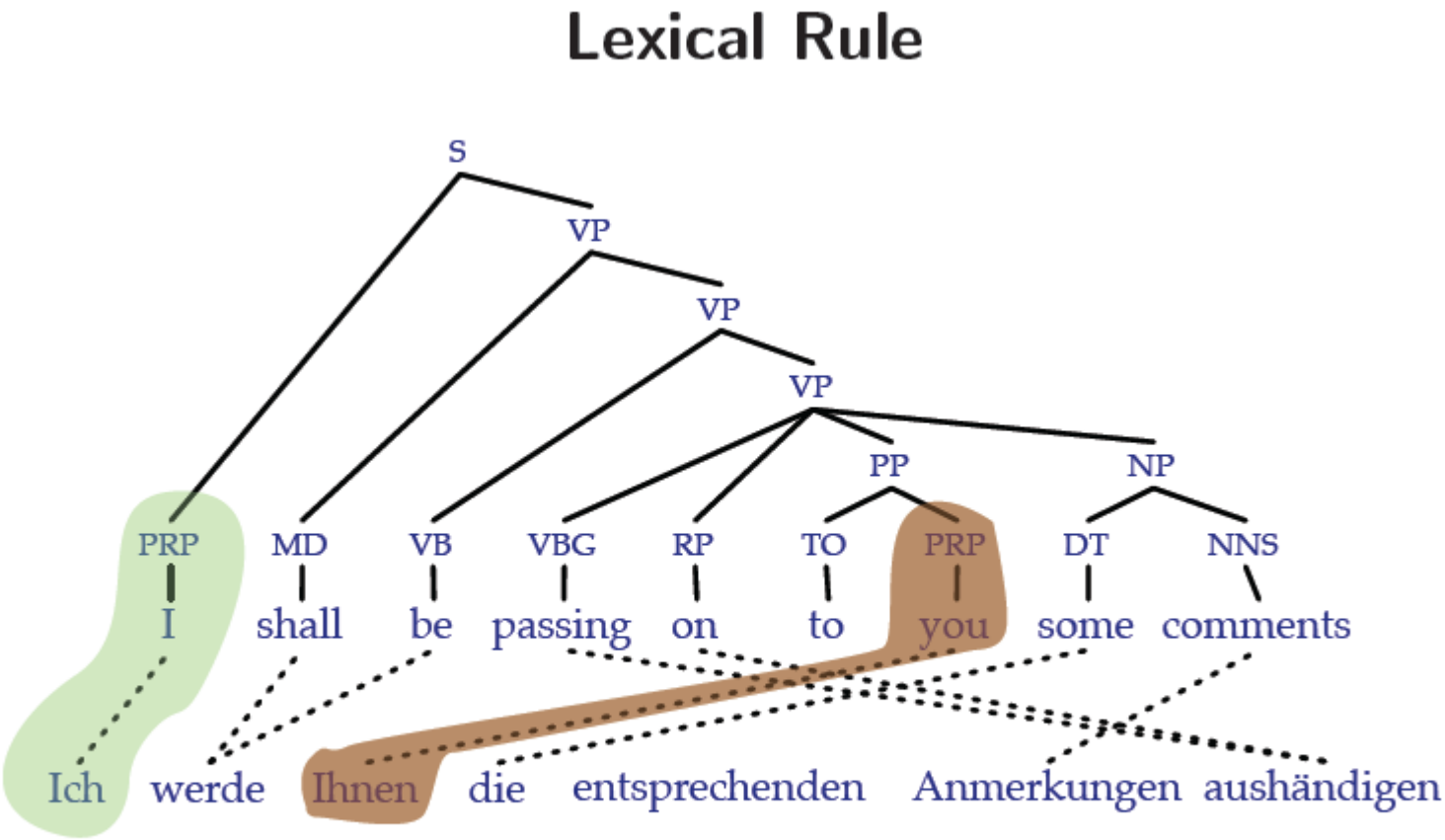
Extract: set of smallest rules required to explain the sentence pair

GHKM Rule Extraction



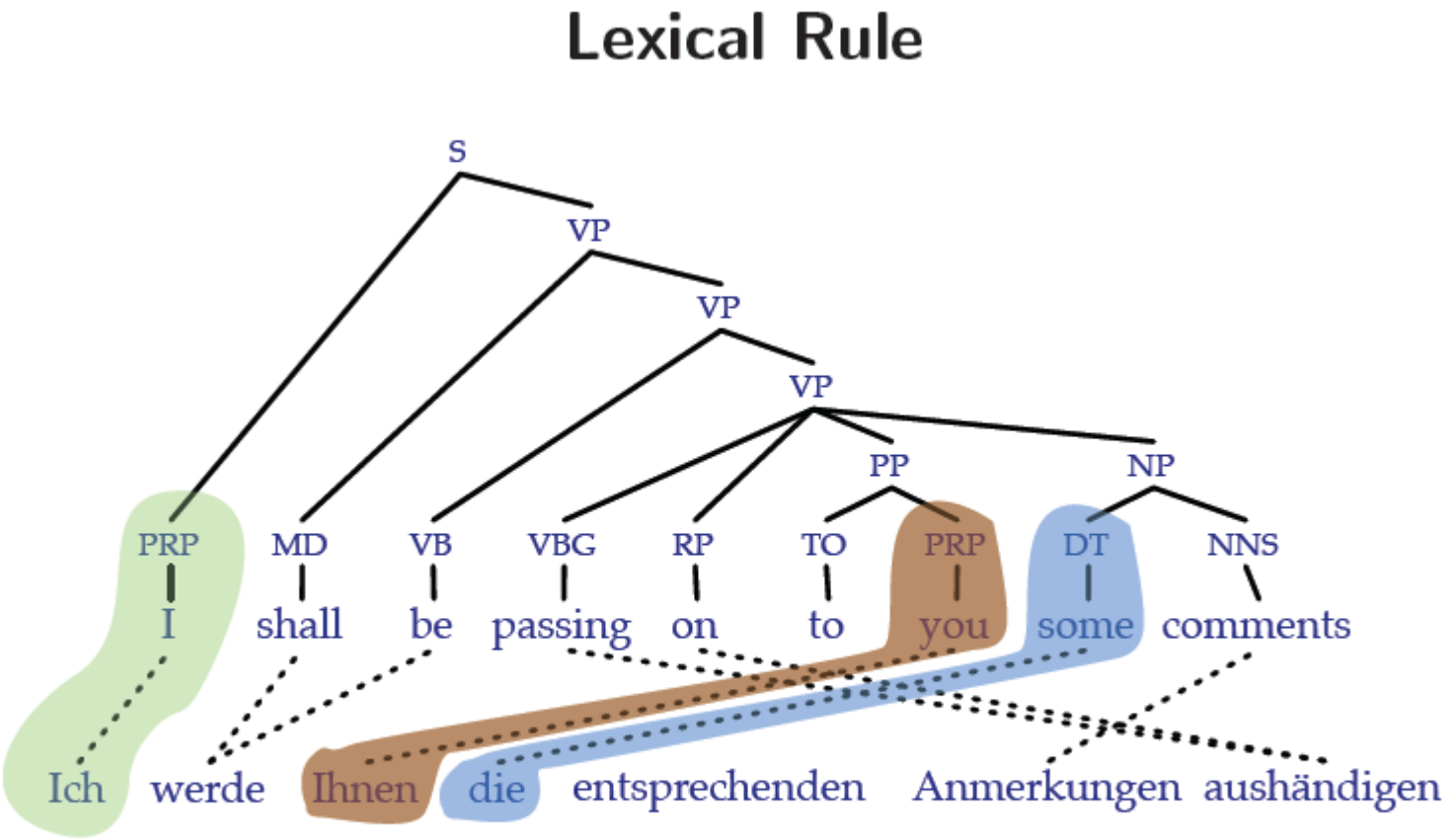
Extracted rule: PRP → Ich | I

GHKM Rule Extraction



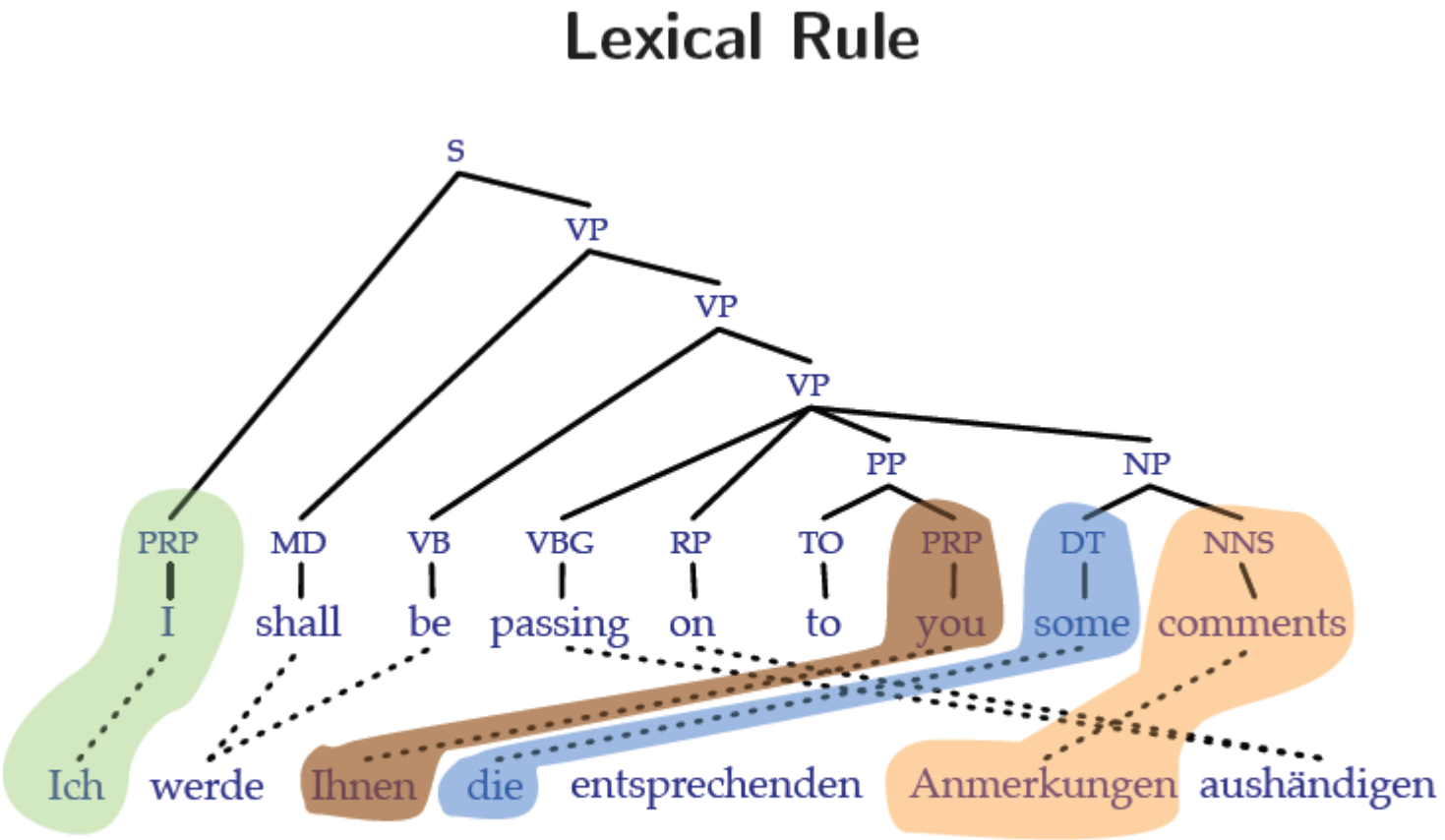
Extracted rule: PRP → Ihnen | you

GHKM Rule Extraction



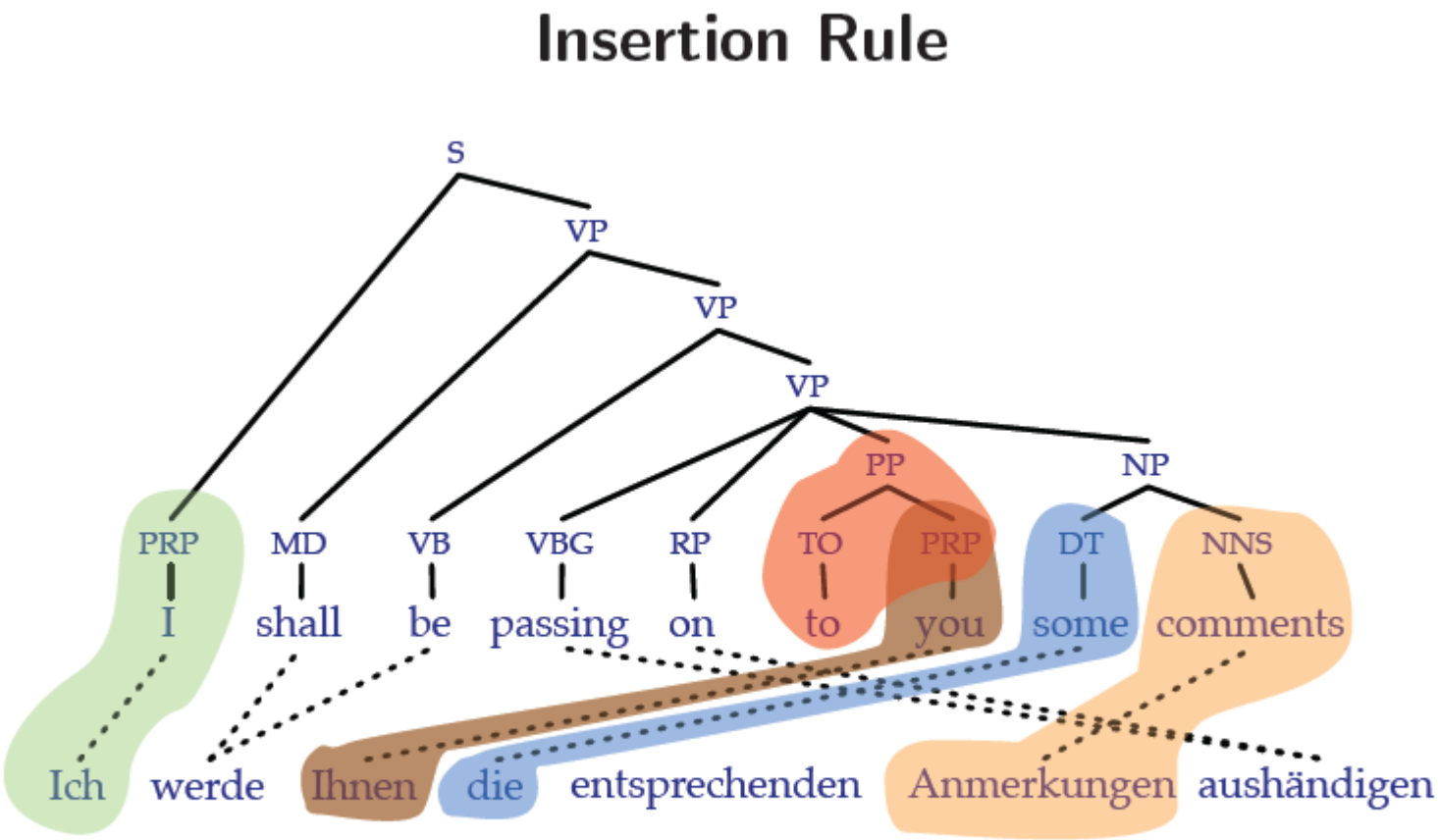
Extracted rule: DT → die | some

GHKM Rule Extraction



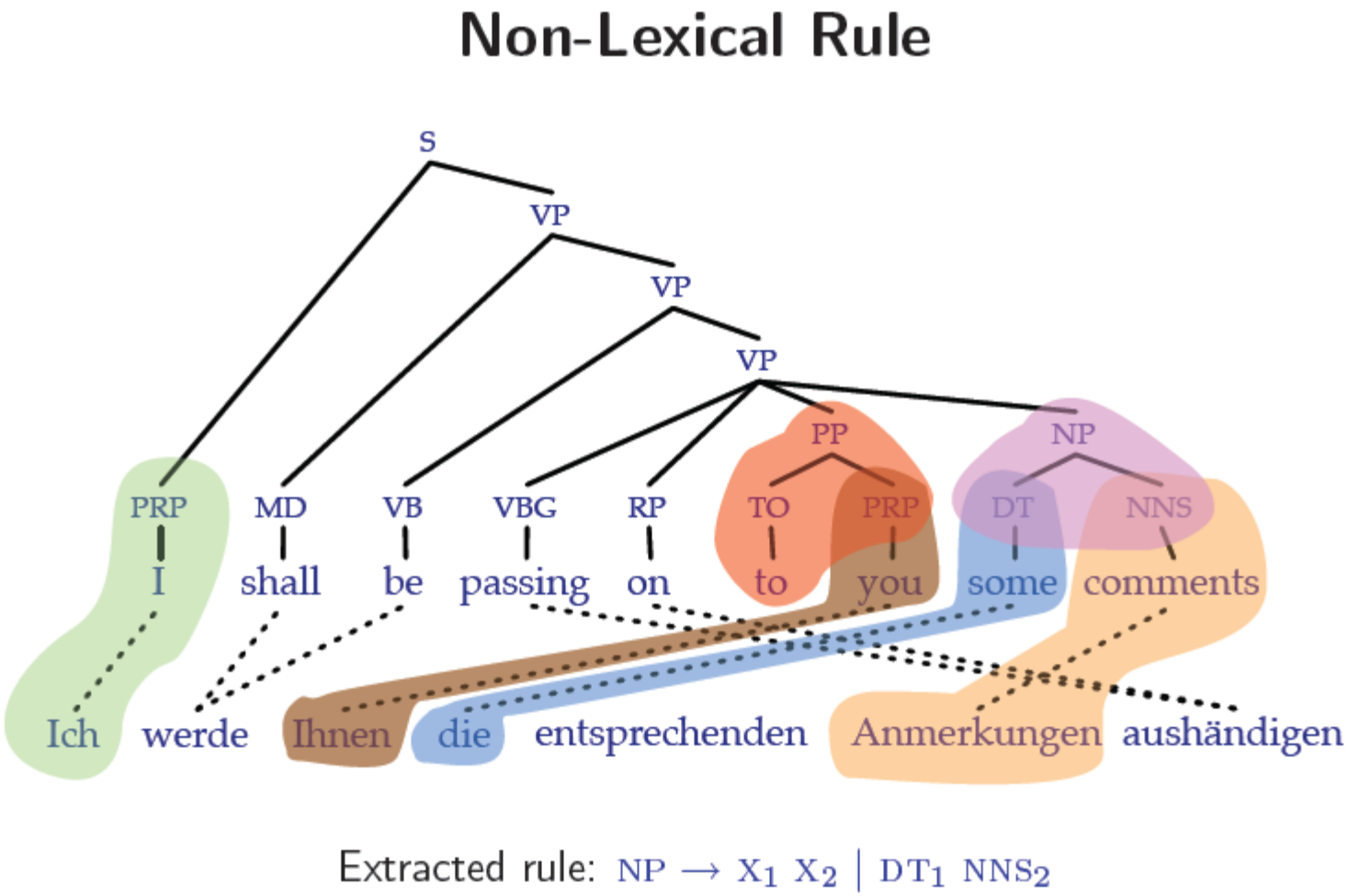
Extracted rule: NNS → Anmerkungen | comments

GHKM Rule Extraction

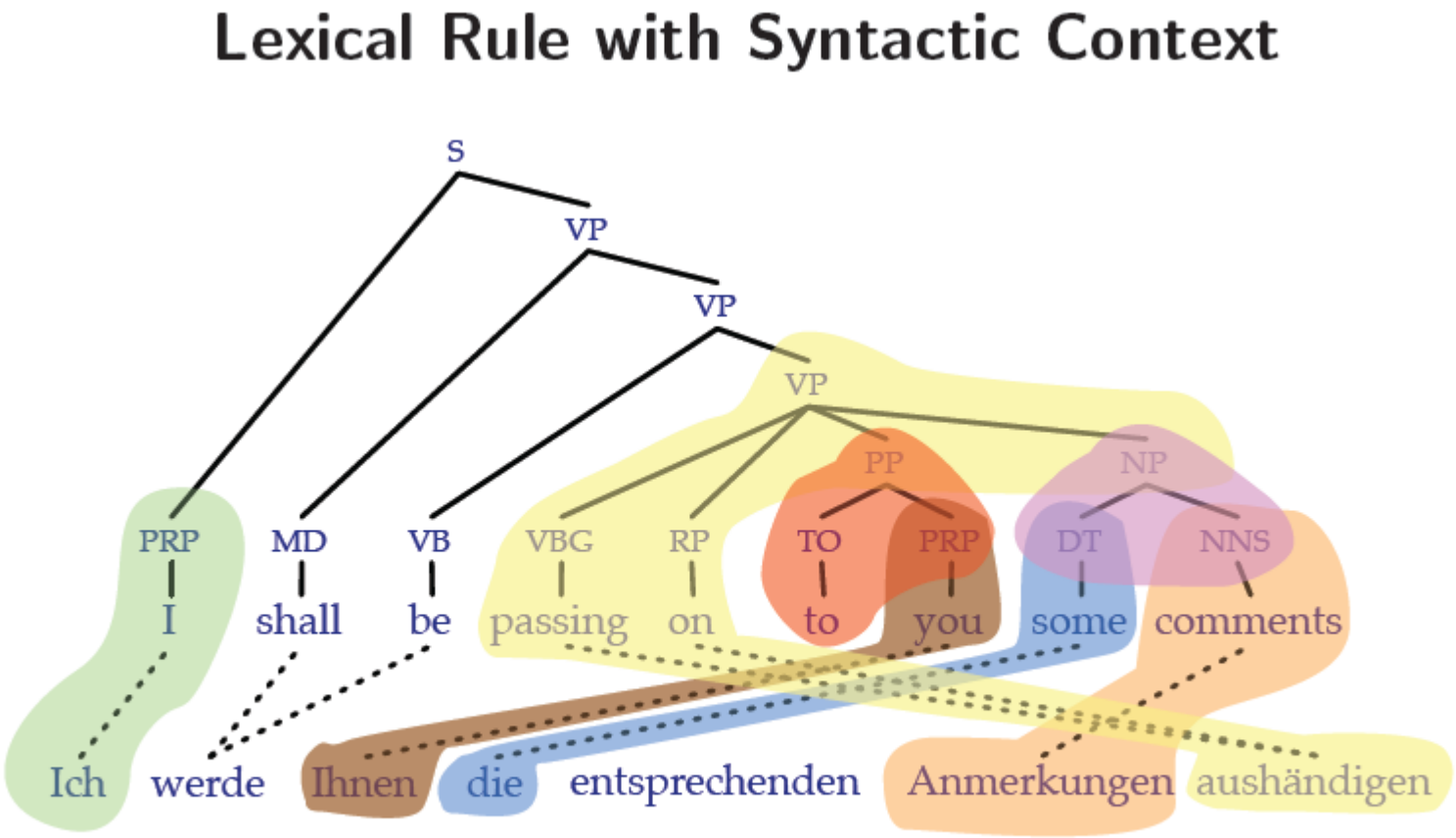


Extracted rule: PP → X | to PRP

GHKM Rule Extraction

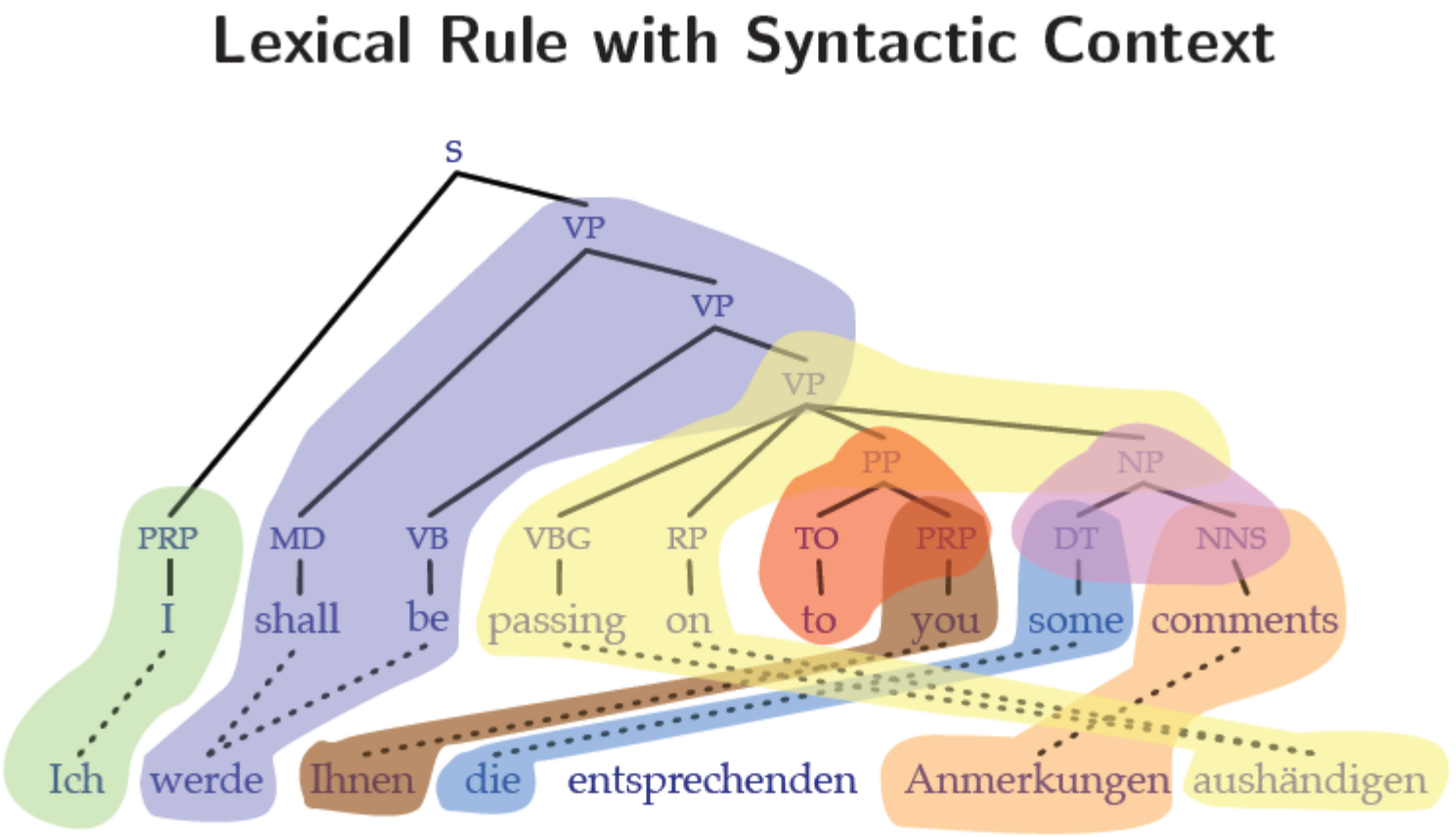


GHKM Rule Extraction



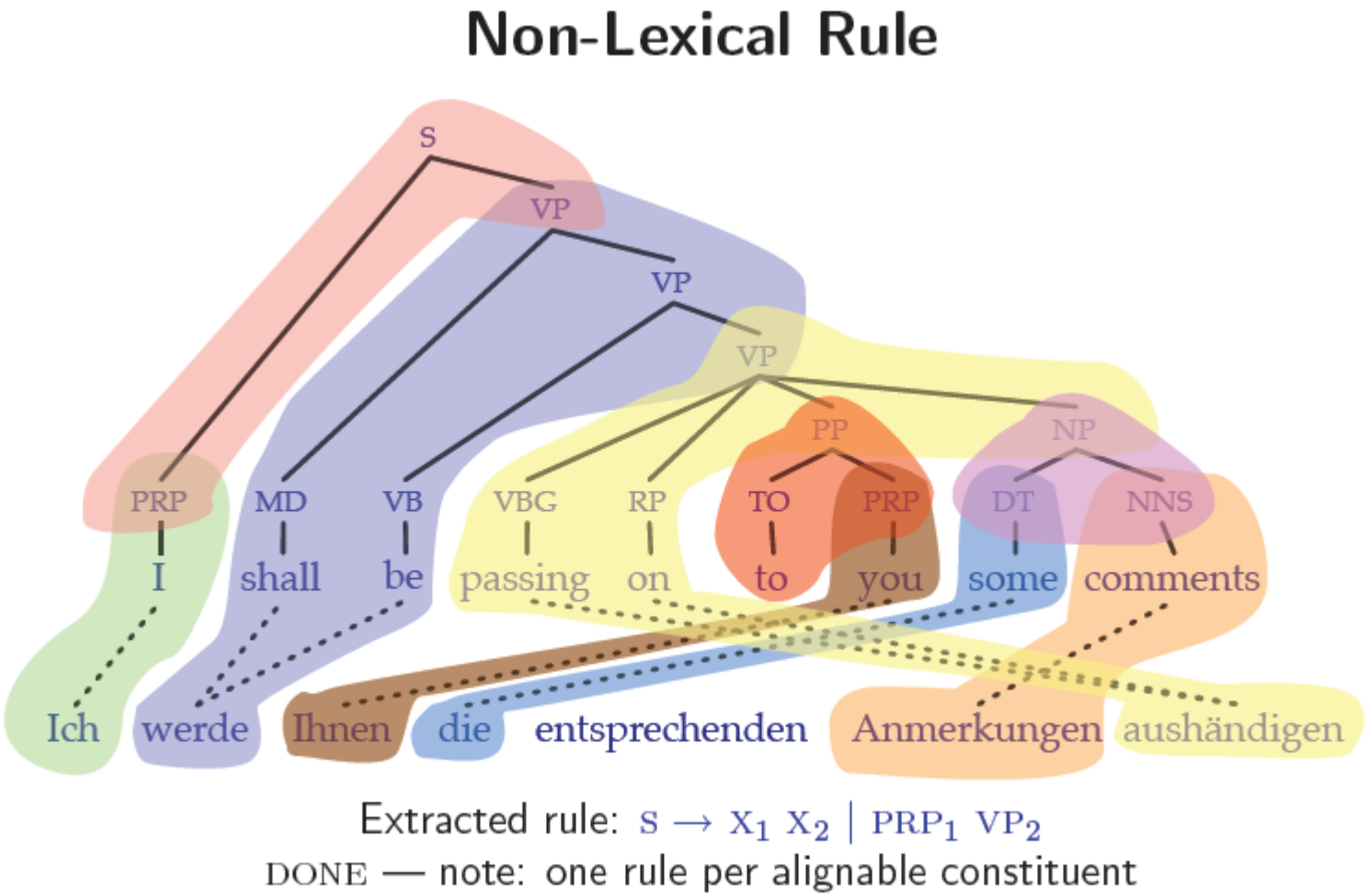
Extracted rule: $\text{VP} \rightarrow \text{X}_1\ \text{X}_2$ aushändigen | passing on $\text{PP}_1\ \text{NP}_2$

GHKM Rule Extraction

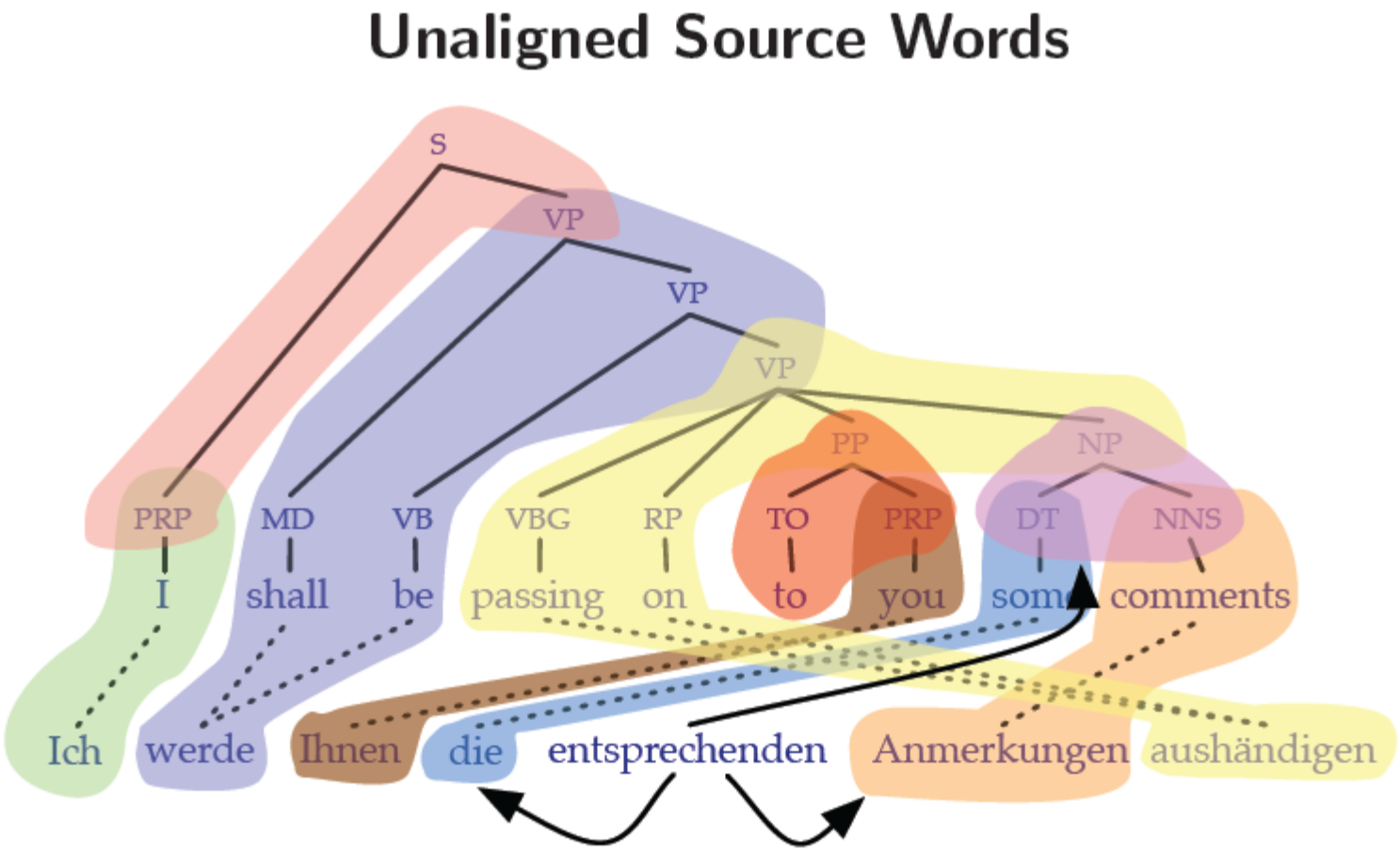


Extracted rule: VP → werde X | shall be VP (ignoring internal structure)

GHKM Rule Extraction



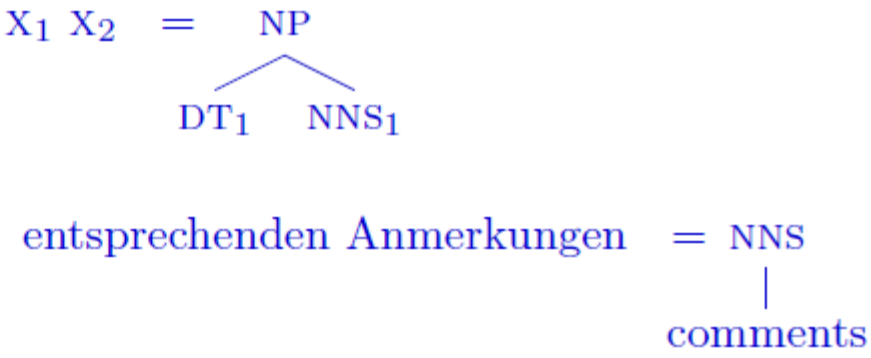
GHKM Rule Extraction



GHKM Rule Extraction

Composed Rules

- Current rules



- Composed rule

(1 non-leaf node: NP)

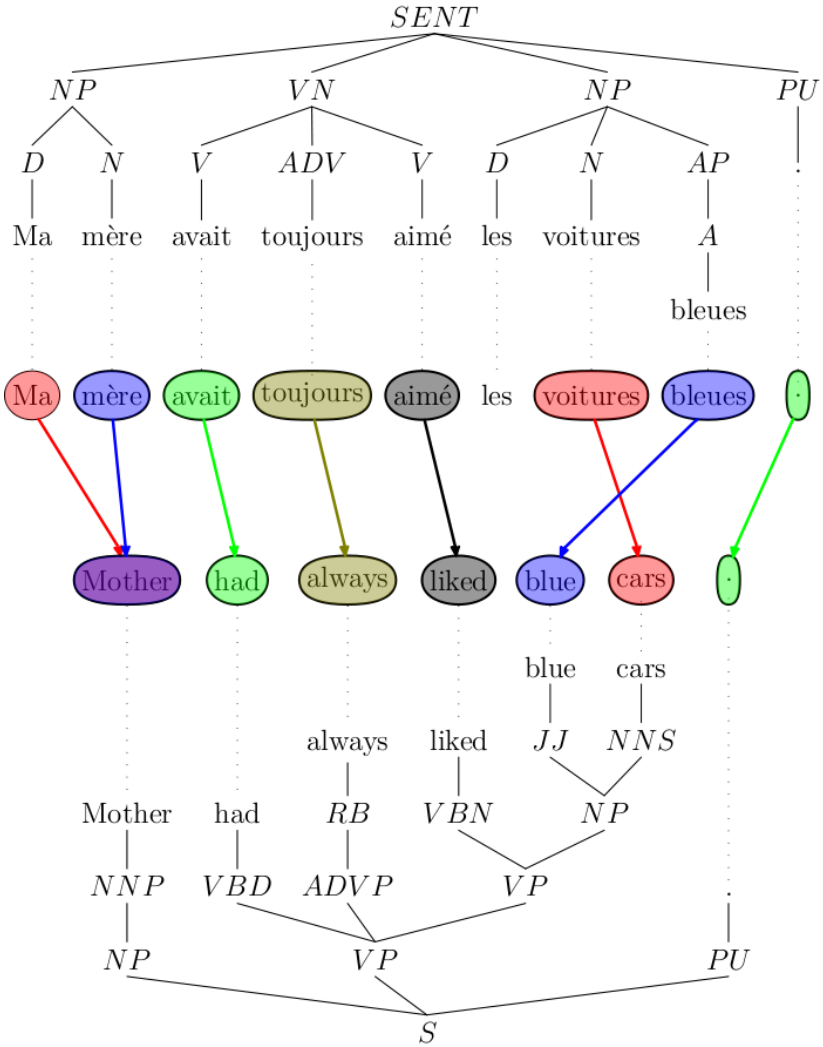
Tree-to-Tree Grammar Extraction

- Developed by [Lavie, Ambati and Parlikar, 2007] and improved in [Hanneman, Burroughs and Lavie, 2011]
- **Goal:** Extract linguistically-supported syntactic phrase-pairs and synchronous transfer rules automatically from parsed parallel corpora
- **Representation:** Synchronous CFG rules with constituent-labels, POS-tags or lexical items on RHS of rules. Syntax-labeled phrases are fully-lexicalized S-CFG rules.
- Acquisition Scenario:
  - Parallel corpus is word-aligned using GIZA++, symetricized.
  - Phrase-structure parses for source and/or target language for each parallel-sentence are obtained using monolingual parsers

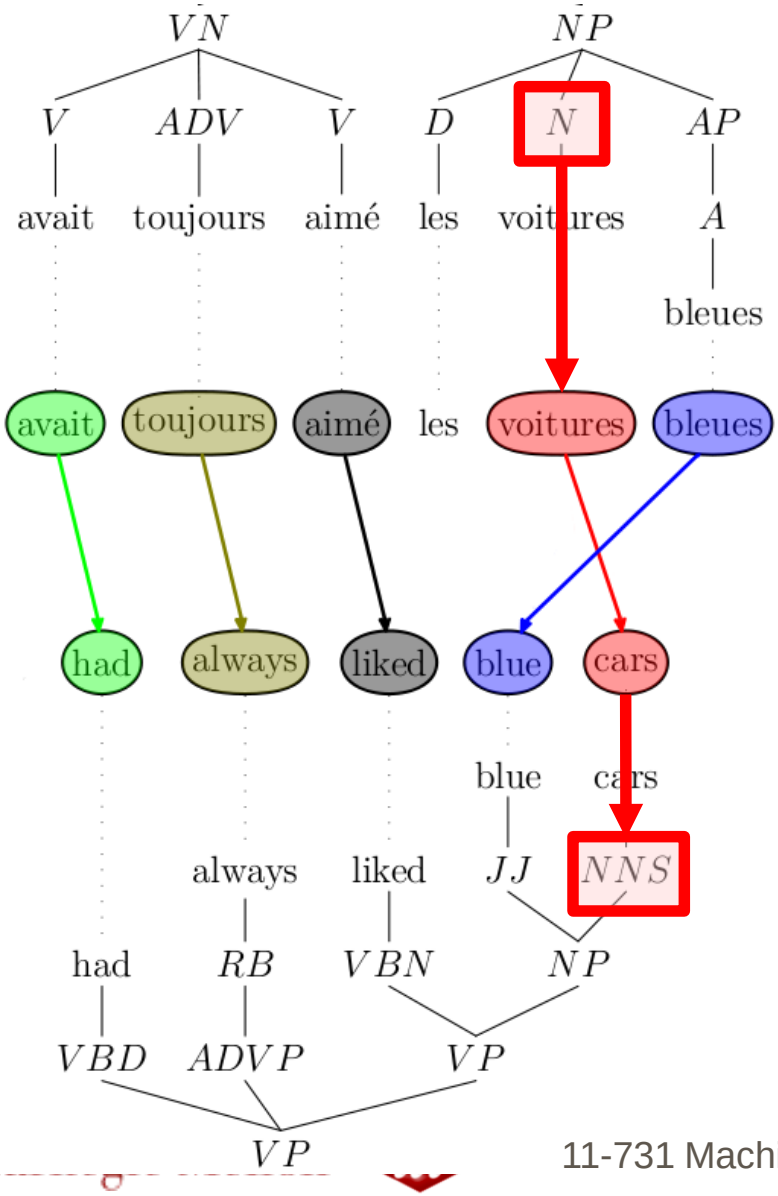
Tree-to-Tree Grammar Extraction

- Goals:
  - Extract all possible rules (minimal and composed) supported by the translation equivalence constraints
  - Do not violate constituent boundaries
  - Allow adding compositional structure
- Accomplished via:
  - Multiple constituent node alignments
  - Virtual constituent nodes
  - Multiple right-hand side decompositions
- First syntax-based grammar extractor to do all three

Tree-to-Tree Grammar Extraction

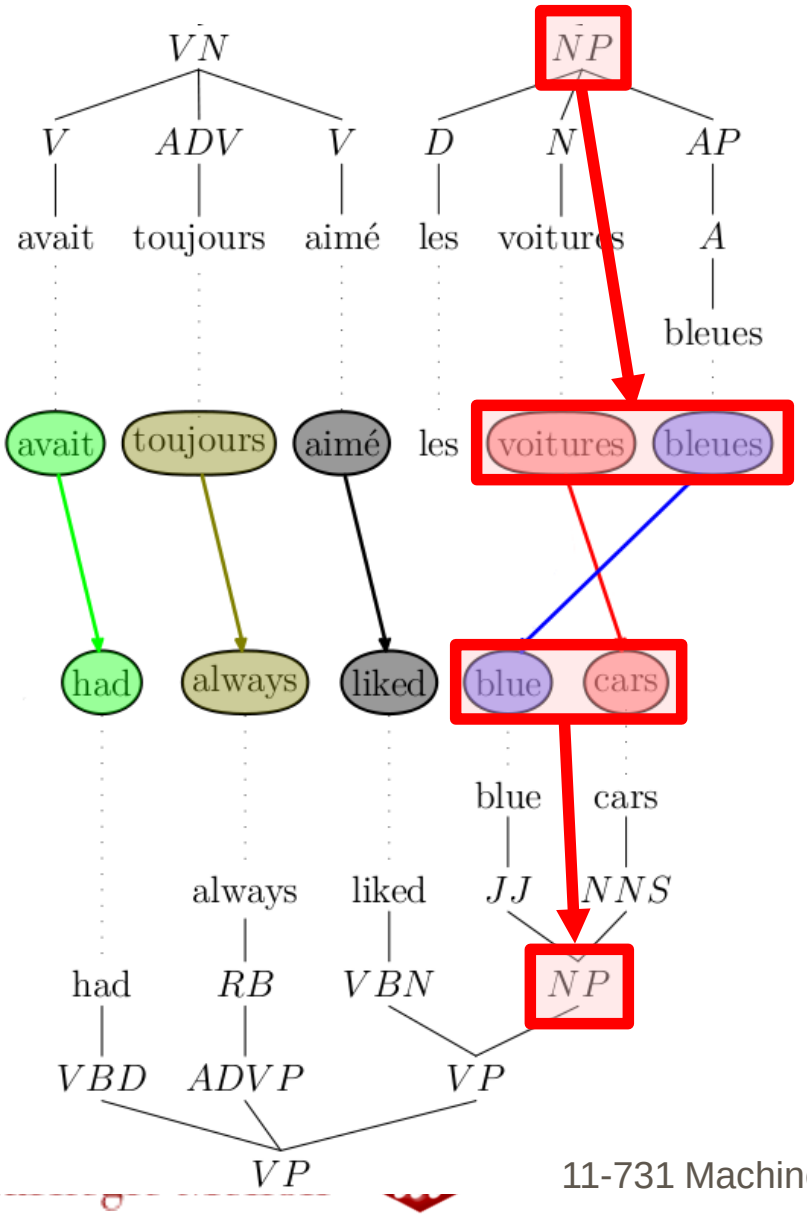


Basic Node Alignment



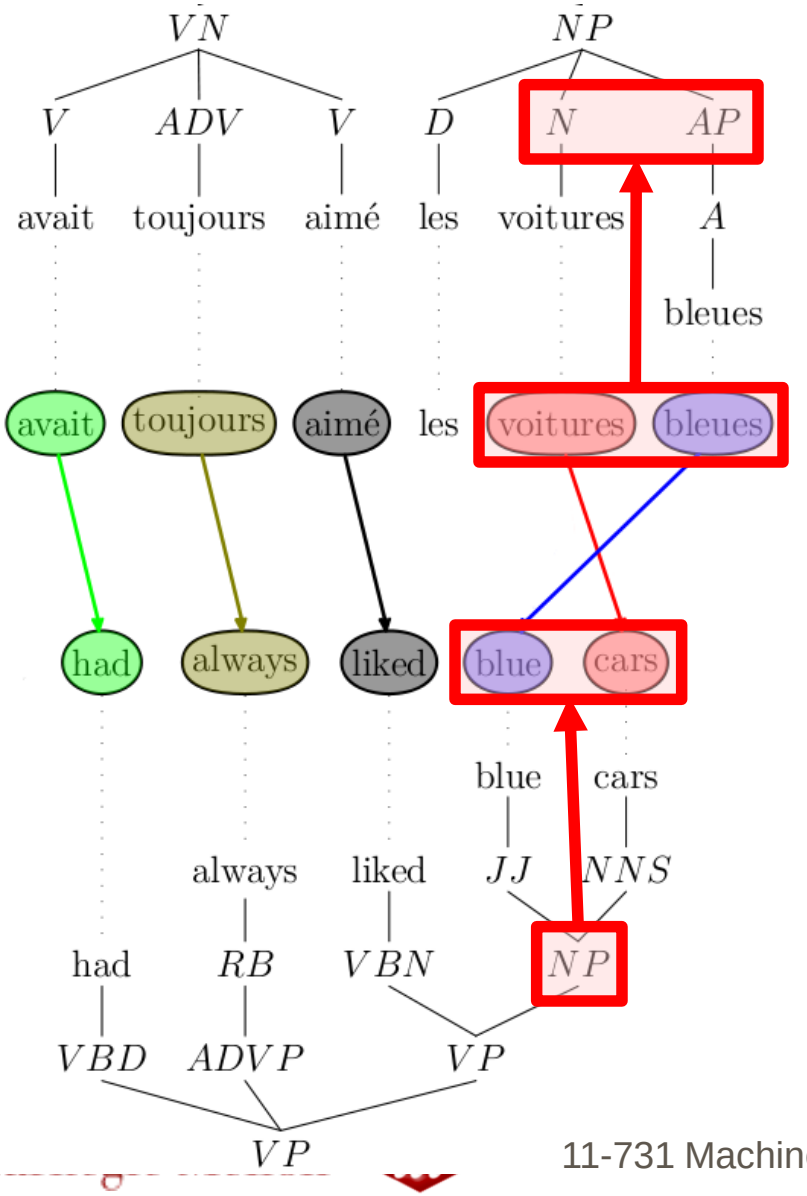
- Word alignment consistency constraint from phrase-based SMT

Basic Node Alignment



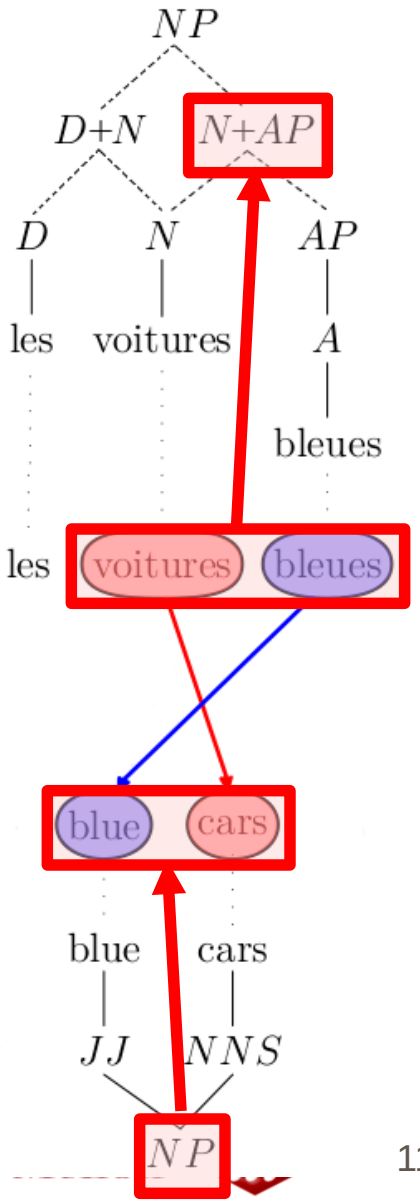
- Word alignment consistency constraint from phrase-based SMT

Virtual Nodes



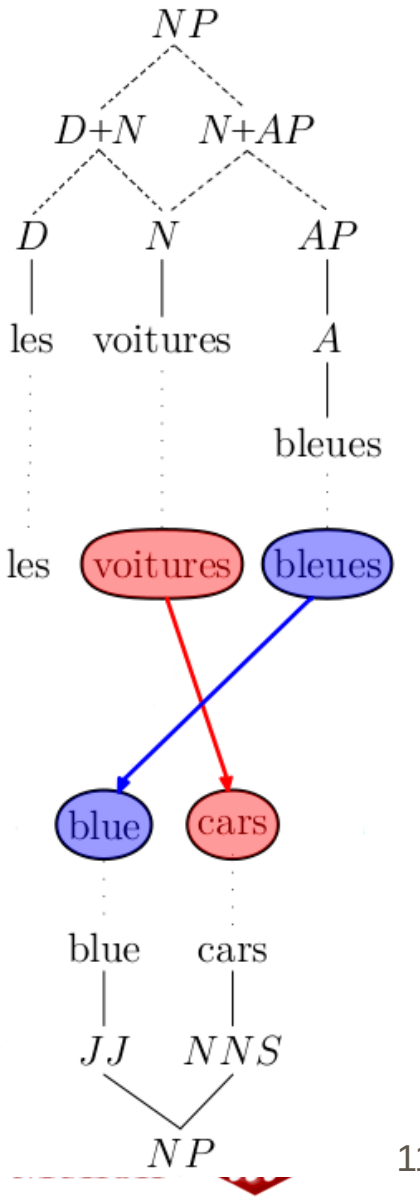
- Consistently aligned consecutive children of the same parent

Virtual Nodes



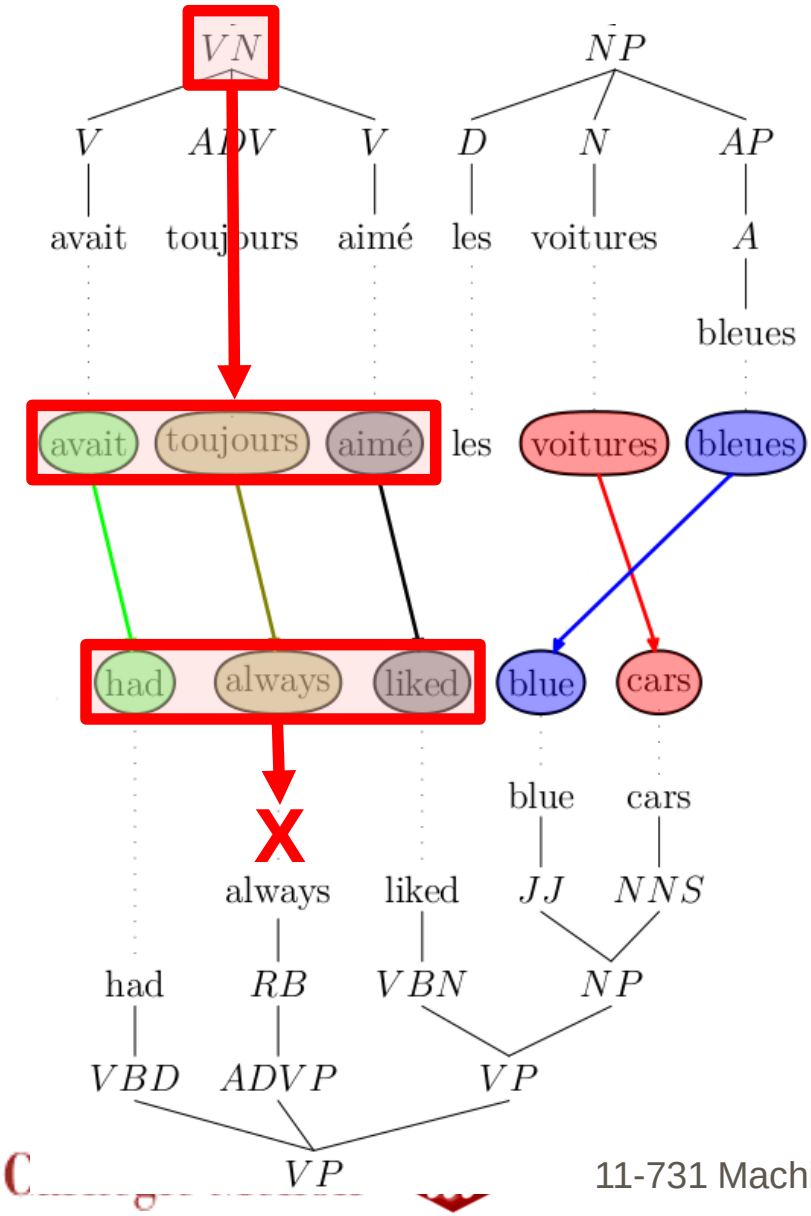
- Consistently aligned consecutive children of the same parent
- New intermediate node inserted in tree

Virtual Nodes



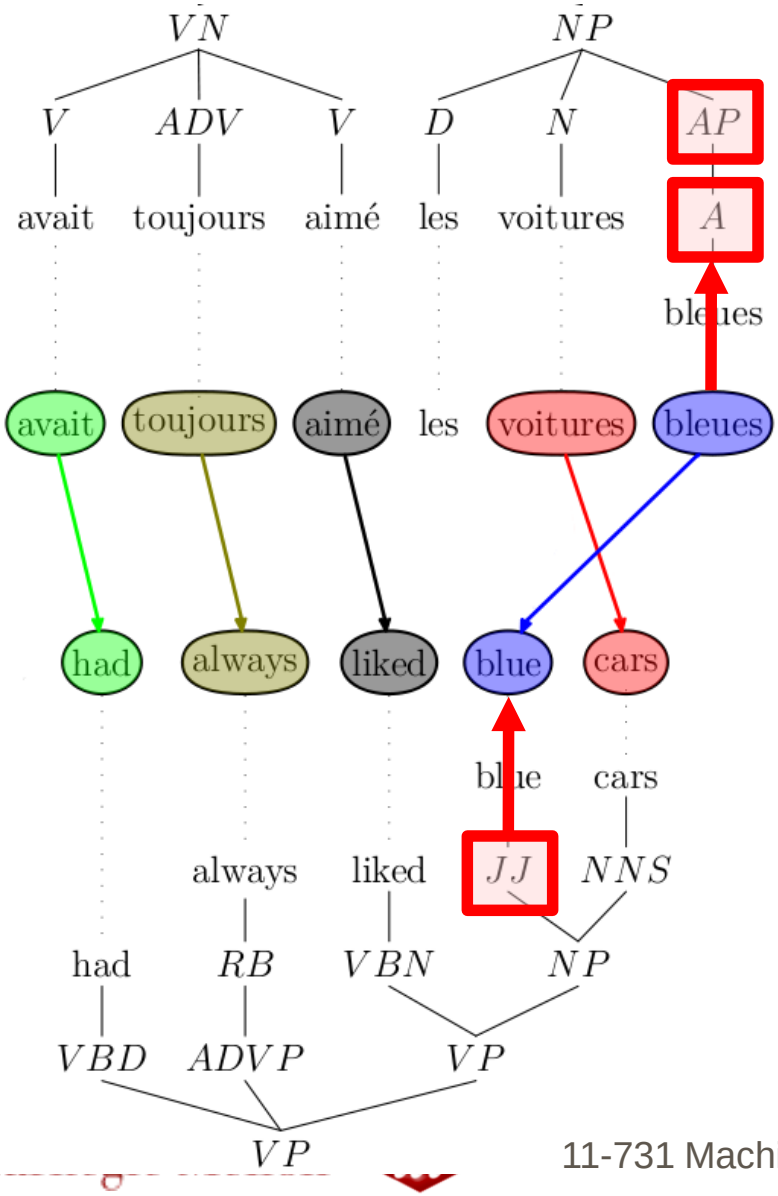
- Consistently aligned consecutive children of the same parent
- New intermediate node inserted in tree
- Virtual nodes may overlap
- Virtual nodes may align to any type of node

Syntax Constraints



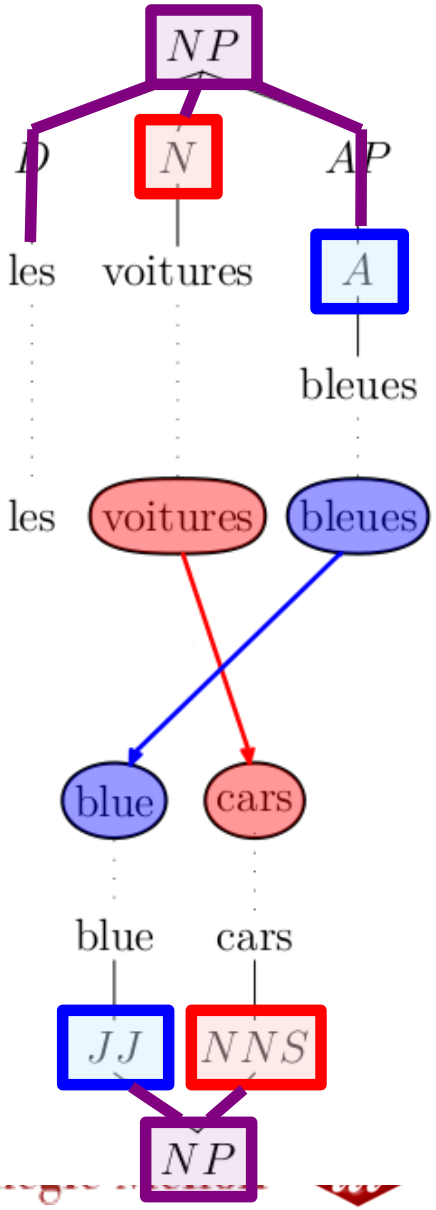
- Consistent word alignments ≠ node alignment
- Virtual nodes may not cross constituent boundaries

Multiple Alignment



- Nodes with multiple consistent alignments
- Keep all of them

Basic Grammar Extraction



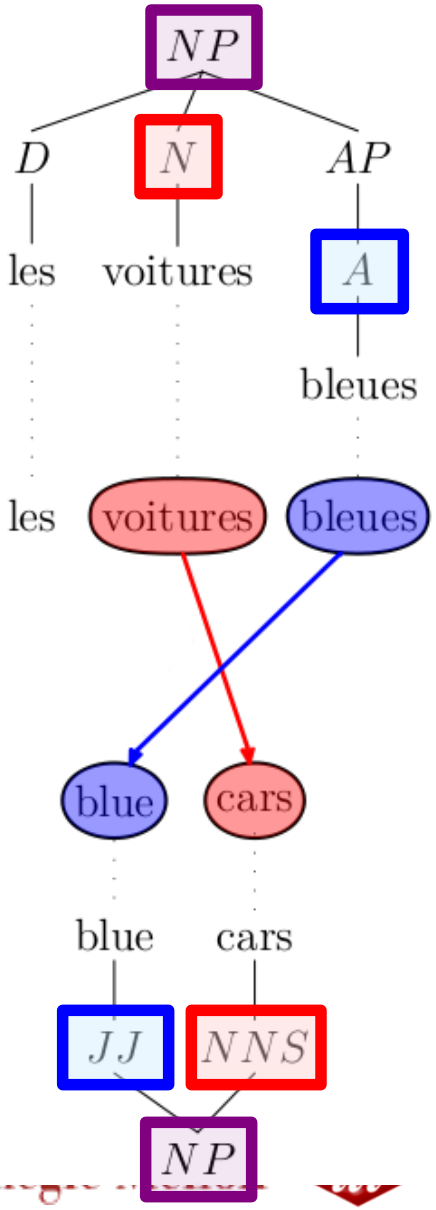
- Aligned node pair is LHS; aligned subnodes are RHS

NP::NP → [les N^1 A^2]::[JJ^2 NNS^1]

N::NNS → [voitures]::[cars]

A::JJ → [bleues]::[blue]

Multiple Decompositions



- All possible right-hand sides are extracted

NP::NP → [les N^1 A^2]::[JJ^2 NNS^1]

NP::NP → [les N^1 bleues]::[blue NNS^1]

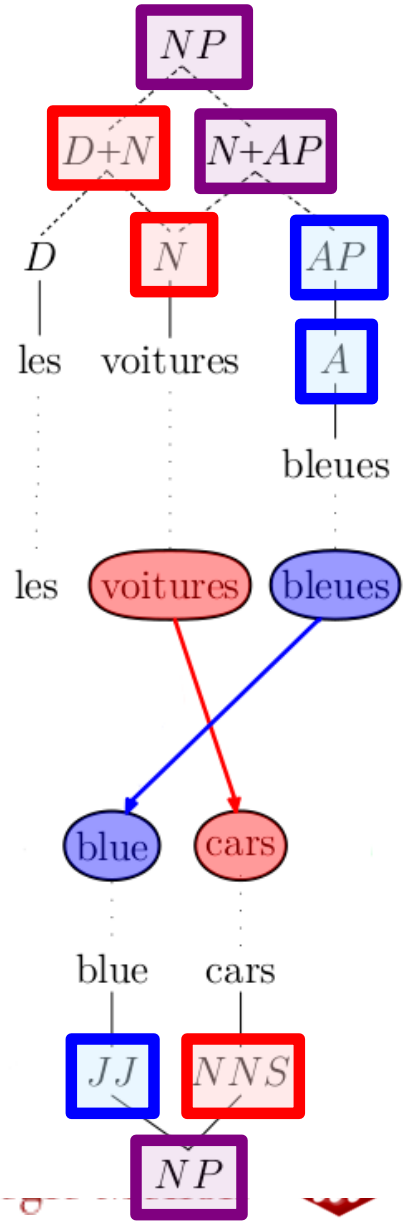
NP::NP → [les voitures A^2]::[JJ^2 cars]

NP::NP → [les voitures bleues]::[blue cars]

N::NNS → [voitures]::[cars]

A::JJ → [bleues]::[blue]

Multiple Decompositions



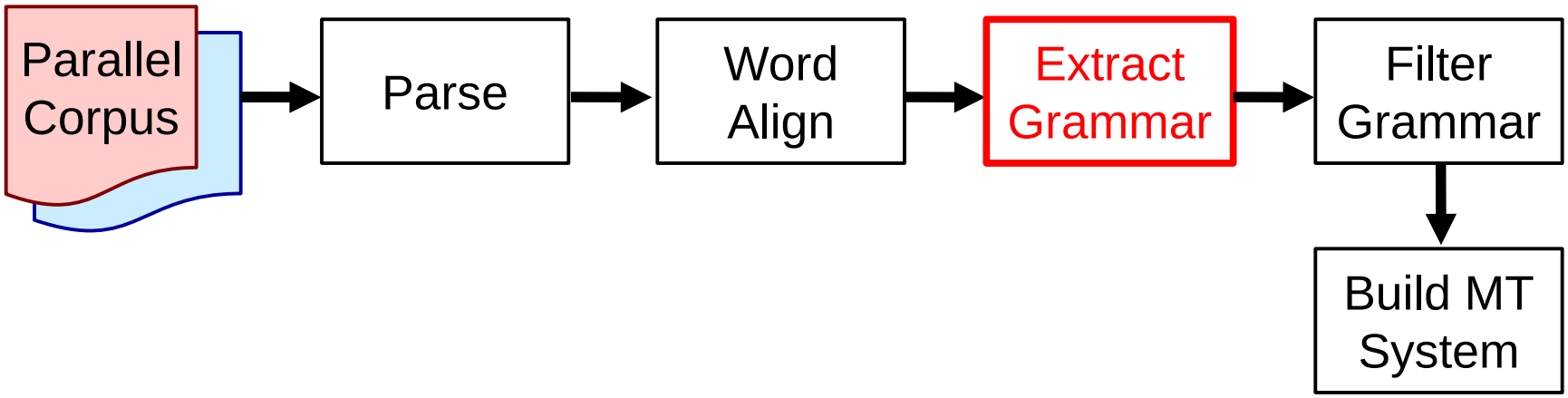
NP::NP → [les N+AP[1]]::[NP[1]]
NP::NP → [D+N[1] AP[2]]::[JJ[2] NNS[1]]
NP::NP → [D+N[1] A[2]]::[JJ[2] NNS[1]]
NP::NP → [les N[1] AP[2]]::[JJ[2] NNS[1]]
NP::NP → [les N[1] A[2]]::[JJ[2] NNS[1]]
NP::NP → [D+N[1] bleues]::[blue NNS[1]]
NP::NP → [les N[1] bleues]::[blue NNS[1]]
NP::NP → [les voitures AP[2]]::[JJ[2] cars]
NP::NP → [les voitures A[2]]::[JJ[2] cars]
NP::NP → [les voitures bleues]::[blue cars]
D+N::NNS → [les N[1]]::[NNS[1]]
D+N::NNS → [les voitures]::[cars]
N+AP::NP → [N[1] AP[2]]::[JJ[2] NNS[1]]
N+AP::NP → [N[1] A[2]]::[JJ[2] NNS[1]]
N+AP::NP → [N[1] bleues]::[blue NNS[1]]
N+AP::NP → [voitures AP[2]]::[JJ[2] cars]
N+AP::NP → [voitures A[2]]::[JJ[2] cars]
N+AP::NP → [voitures bleues]::[blue cars]
N::NNS → [voitures]::[cars]
AP::JJ → [A[1]]::[JJ[1]]
AP::JJ → [bleues]::[blue]
A::JJ → [bleues]::[blue]

Comparison to Related Work

	Tree Constr.	Multiple Aligns	Virtual Nodes	Multiple Decomp.
Hiero	No	—	—	Yes
Stat-XFER	Yes	No	Some	No
GHKM	Yes	No	No	Yes
SAMT	No	No	Yes	Yes
Chiang [2010]	No	No	Yes	Yes
HBL [2011]	Yes	Yes	Yes	Yes

Experimental Setup

- Train: FBIS Chinese–English corpus
- Tune: NIST MT 2006
- Test: NIST MT 2003



Extraction Configurations

- Baseline:
  - Stat-XFER exact tree-to-tree extractor
  - Single decomposition with minimal rules
- Multi:
  - Add multiple alignments and decompositions
- Virt short:
  - Add virtual nodes; max rule length 5
- Virt long:
  - Max rule length 7

Number of Rules Extracted

	Tokens		Types	
	Phrase	**Hierarc.**	**Phrase**	**Hierarc.**
Baseline	6,646,791	1,876,384	1,929,641	767,573
Multi	8,709,589	6,657,590	2,016,227	3,590,184
Virt short	10,190,487	14,190,066	2,877,650	8,313,690
Virt long	10,288,731	22,479,863	2,970,403	15,750,695

Number of Rules Extracted

	Tokens		Types	
	Phrase	Hierarc.	Phrase	Hierarc.
Baseline	6,646,791	1,876,384	1,929,641	767,573
Multi	8,709,589	6,657,590	2,016,227	3,590,184
Virt short	10,190,487	14,190,066	2,877,650	8,313,690
Virt long	10,288,731	22,479,863	2,970,403	15,750,695

- Multiple alignments and decompositions:
  - Four times as many hierarchical rules
  - Small increase in number of phrase pairs

Number of Rules Extracted

	Tokens		Types	
	Phrase	Hierarc.	Phrase	Hierarc.
Baseline	6,646,791	1,876,384	1,929,641	767,573
Multi	8,709,589	6,657,590	2,016,227	3,590,184
Virt short	10,190,487	14,190,066	2,877,650	8,313,690
Virt long	10,288,731	22,479,863	2,970,403	15,750,695

- Multiple decompositions and virtual nodes:
  - 20 times as many hierarchical rules
  - Stronger effect on phrase pairs
  - 46% of rule types use virtual nodes

Number of Rules Extracted

	Tokens		Types	
	Phrase	Hierarc.	Phrase	Hierarc.
Baseline	6,646,791	1,876,384	1,929,641	767,573
Multi	8,709,589	6,657,590	2,016,227	3,590,184
Virt short	10,190,487	14,190,066	2,877,650	8,313,690
Virt long	10,288,731	22,479,863	2,970,403	15,750,695

- Proportion of singletons mostly unchanged
- Average hierarchical rule count drops

Results: Metric Scores

- NIST MT 2003 test set

System	Filter	BLEU	METR	TER
Baseline	**10k**	24.39	54.35	68.01
Multi	**10k**	24.28	53.58	65.30
Virt short	**10k**	25.16	54.33	66.25
Virt long	**10k**	25.74	54.55	65.52

- Strict grammar filtering: extra phrase pairs help improve scores

Tree Transduction Models

- Originally proposed by Yamada and Knight, 2001. Influenced later work by Gildea et al. on Tree-to-String models
- Conceptually simpler than most other models:
  - Learn finite-state transductions on source-language parse-trees in order to map them into well-ordered and well-formed target sentences, based on the viterbi word alignments
- **Representation:** simple local transformations on tree structure, given contextual structure in the tree:
  - Transduce leaf words in the tree from source to target language
  - Delete a leaf-word or a sub-tree in a given context
  - Insert a leaf-word or a sub-tree in a given context
  - Transpose (invert order) of two sub-trees in a given context
  - [Advanced model by Gildea: duplicate and insert a sub-tree]

Tree Transduction Models

- Main Issues/Problems:
  - Some complex reorderings and correspondences cannot be modeled using these simple tree transductions
  - Highly sensitive to errors in the source-language parse-tree and to word-alignment errors

Summary

- Variety of structure and syntax based models: string-to-tree, tree-to-string, tree-to-tree
- Different models utilize different structural annotations on training resources and depend on different independent components (parsers, word alignments)
- Different model acquisition processes from parallel data, but several recurring themes:
  - Finding sub-sentential translation equivalents and relating them via hierarchical and/or syntax-based structure
  - Statistical modeling of the massive collections of rules acquired from the parallel data

Major Challenges

- **Sparse Coverage:** the acquired syntax-based models are often much sparser in coverage than non-syntactic phrases
  - Because they apply additional hard constraints beyond word-alignment as evidence of translation equivalence
  - Because the models fragment the data – they are often observed far fewer times in training data → more difficult to model them statistically
  - Consequently, "pure" syntactic models often lag behind phrase-based models in translation performance – observed and learned again and again by different groups (including our own)
  - This motivates approaches that integrate syntax-based models with phrase-based models
- **Overcoming Pipeline Errors:**
  - Adding independent components (parser output, viterbi word alignments) introduces cumulative errors that are hard to overcome
  - Various approaches try to get around these problems
  - Also recent work on "syntax-aware" word-alignment, "bi-lingual-aware" parsing

Major Challenges

- **Optimizing for Structure Granularity and Labels:**
  - Syntactic structure in MT heavily based on Penn TreeBank structures and labels (POS and constituents) – are these needed and optimal for MT, even for MT into English?
  - Approaches range from single abstract hierarchical "X" label, to fully lexicalized constituent labels. What is optimal? How do we answer this question?
  - Alternative Approaches (i.e. ITGs) aim to overcome this problem by unsupervised inference of the structure from the data
- **Direct Contrast and Comparison of alternative approaches is extremely difficult:**
  - Decoding with these syntactic models is highly complex and computationally intensive
  - Different groups/approaches develop their own decoders
  - Hard to compare anything beyond BLEU (or other metric) scores
- Different groups continue to pursue different approaches – this is at the forefront of current research in Statistical MT

References

- (2008) Vamshi Ambati & Alon Lavie: Improving syntax driven translation models by re-structuring divergent and non-isomorphic parse tree structures. *AMTA-2008. MT at work: Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas*, Waikiki, Hawai'i, 21-25 October 2008; pp.235-244
- (2005) David Chiang: A hierarchical phrase-based model for statistical machine translation. *ACL-2005*: *43rd Annual meeting of the Association for Computational Linguistics*, University of Michigan, Ann Arbor, 25-30 June 2005; pp. 263-270.
- (2004) Michel Galley, Mark Hopkins, Kevin Knight & Daniel Marcu: What's in a translation rule? *HLT-NAACL 2004: Human Language Technology conference and North American Chapter of the Association for Computational Linguistics annual meeting*, May 2-7, 2004, The Park Plaza Hotel, Boston, USA; pp.273-280.
- (2006) Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, & Ignacio Thayer: Scalable inference and training of context-rich syntatic translation models. *Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, 17-21 July 2006; pp.961-968.
- (2011) Hanneman, G., M. Burroughs and A. Lavie. "A General-Purpose Rule Extractor for SCFG-Based Machine Translation" . *In Proceedings of Fifth Workshop on Syntax and Structure in Statistical Translation (SSST-5) at the 49th Meeting of the Association for Computational Linguistics - Human Language Technologies Conference (ACL-HLT-2011)*, Portland, OR, June 2011. Pages 135-144.
- (2008) Alon Lavie, Alok Parlikar, & Vamshi Ambati: Syntax-driven learning of sub-sentential translation equivalents and translation rules from parsed parallel corpora. *Second ACL Workshop on Syntax and Structure in Statistical Translation (ACL-08 SSST-2)*, Proceedings, 20 June 2008, Columbus, Ohio, USA; pp.87-95.
- (2007) Ashish Venugopal & Andreas Zollmann: Hierarchical and syntax structured MT. *First Machine Translation Marathon*, Edinburgh, April 16-20, 2007; 52pp.
- (2001) Kenji Yamada & Kevin Knight: A syntax-based statistical translation model *ACL-EACL-2001*: 39th Annual meeting [of the Association for Computational Linguistics] and 10th Conference of the European Chapter [of ACL], July 9th - 11th 2001, Toulouse, France; pp.523-530.
- (2006) Andreas Zollmann & Ashish Venugopal: Syntax augmented machine translation via chart parsing. *HLT-NAACL 2006: Proceedings of the Workshop on Statistical Machine Translation*, New York, NY, USA, June 2006; pp. 138-141