Extending Hierarchical Machine Translation Using Soft Syntactic Labels

Stephan Peitz

peitz@i6.informatik.rwth-aachen.de

April 14, 2010

Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik 6
Computer Science Department
RWTH Aachen University, Germany

Outline

1 State of the Art

- **hierarchical phrase-based translation:**
  - **A Hierarchical Phrase-based Model for Statistical Machine Translation [Chiang 05] (UMD/ISI, ACL 2005)**
- **explicit linguistic structures:**
  - **Statistical Machine Translation with Syntactified Target Language Phrases [Marcu & Wang^{+} 06] (ISI, EMNLP 2006)**
  - **Well-formed Dependency Structure [Shen & Xu^{+} 08] (BBN, ACL 2008)**
  - **Preference Grammars: Softening Syntactic Constraints to Improve Statistical Machine Translation**
    [Venugopal & Zollmann^{+} 09] (CMU 2009)

2 Hierarchical Phrase-based Translation

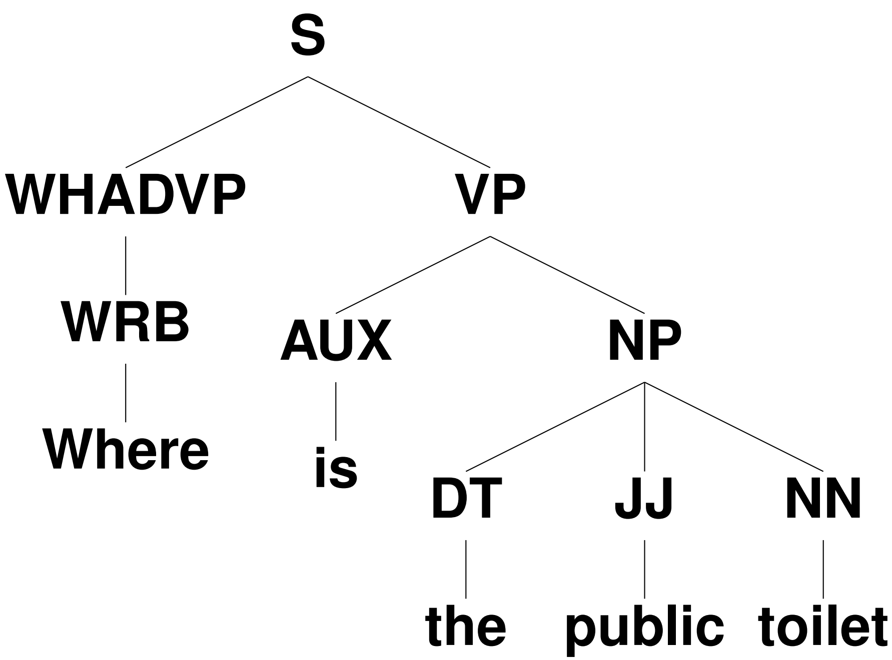
- formalization as a parallel stochastic context-free grammar
- consider the generation of a translation as probabilistic parsing procedure (CYK+)
- rules of the form $X \rightarrow \langle \gamma, \alpha, \sim \rangle$, where:
  - X is a non-terminal
  - γ and α are strings of terminals and non-terminals
  - $\sim$ is a one-to-one correspondence between the non-terminals of α and γ
- example:

$$X \rightarrow \langle \text{Ich stimme } X^{\sim 1} \text{ zu, I agree with } X^{\sim 1} \rangle$$
$$X \rightarrow \langle \text{weil andere } X^{\sim 1} \text{ nicht } X^{\sim 2}\text{, because others have not } X^{\sim 2}\ X^{\sim 1} \rangle$$

- problem: only use a generic non-terminal
- no further information to guide the translation process

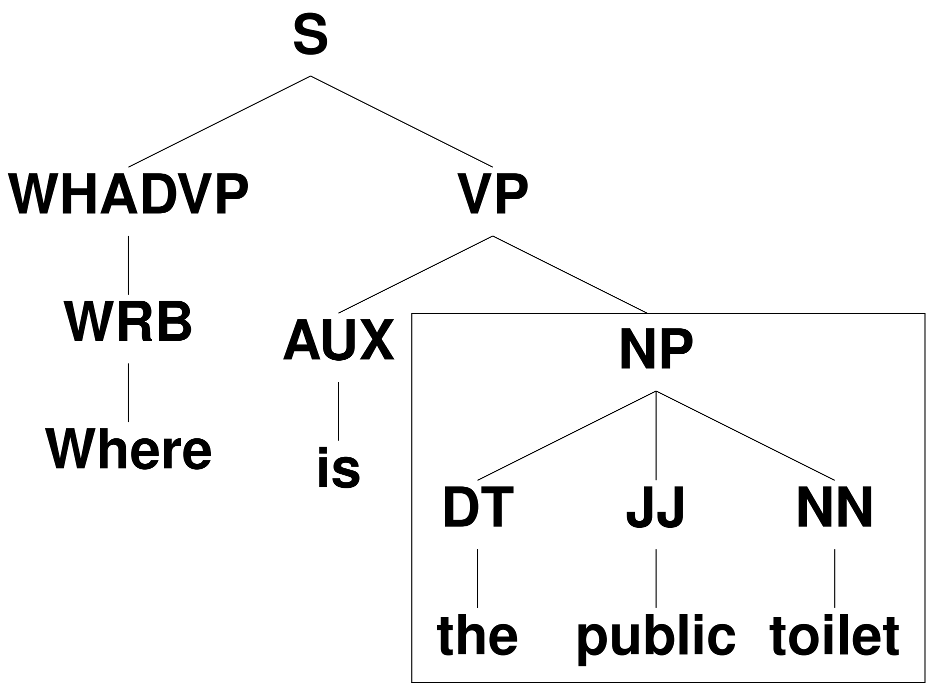
3 Soft Syntactic Labels

- **idea: extend hierarchical translation with an additional model using syntax information**
  - **additional information is extracted from deep syntactic parse tree of the target language**
- **goal: get more fluent, structured and grammatically correct translation**



Syntactic Analysis

- use labels from deep syntactic parse trees to replace the generic non-terminals in the translation process
- each sentence of the target language is parsed
- resulting syntax trees are used in the rule extraction process
  - for each phrase of a given sentence, find the node in the parse tree that matches the phrase best



Rule Extraction

- $\mathcal{H} = \{NP, PP, NN, DT\ \ldots\}$ **is a set of labels used in the additional model**
- **for each rule r_i:**
  - **define a probability distribution $p(\mathrm{h}|r_i)$ over vectors of labels h**
  - **h replaces in the additional model the generic non-terminals in the rule r_i**

Example

- rules with soft syntactic labels:
  - hierarchical rule

$$r_0 : X \rightarrow \langle X^{\sim 1} \textbf{ Zweideutigkeit}, \; X^{\sim 1} \textbf{ ambiguity} \rangle$$
$$\left\{ \begin{array}{rcl} p((NP, DT)|r_0) & = & 0.5 \\ p((PP, PP)|r_0) & = & 0.3 \\ p((NP, NP)|r_0) & = & 0.2 \end{array} \right\}$$

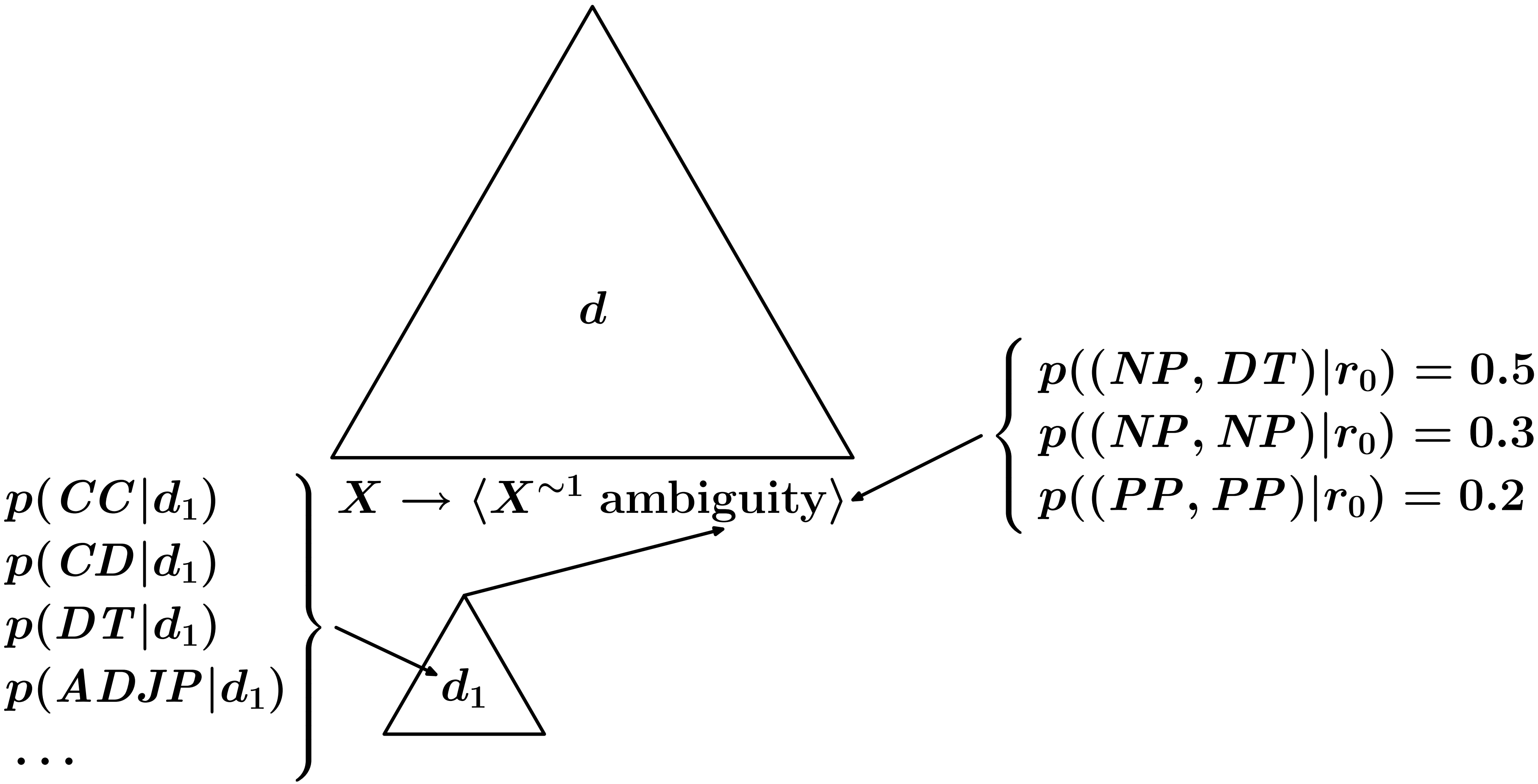
  - lexical rule

$$r_1 : X \rightarrow \langle \textbf{diese}, \; \textbf{this} \rangle$$
$$\left\{ \, p((DT)|r_1) \; = \; 1 \, \right\}$$

4 Tree Well-Formedness Probability Model

- introduce additional model to measure the compatibility between two rules
  - rules with high mutual match should get a high probability
- used factors in the computation of the additional feature $p_{syntax}(d)$ for a derivation d:
  - distribution for each rule computed in the rule extraction
  - distribution over all labels for each sub-derivation

Visualization



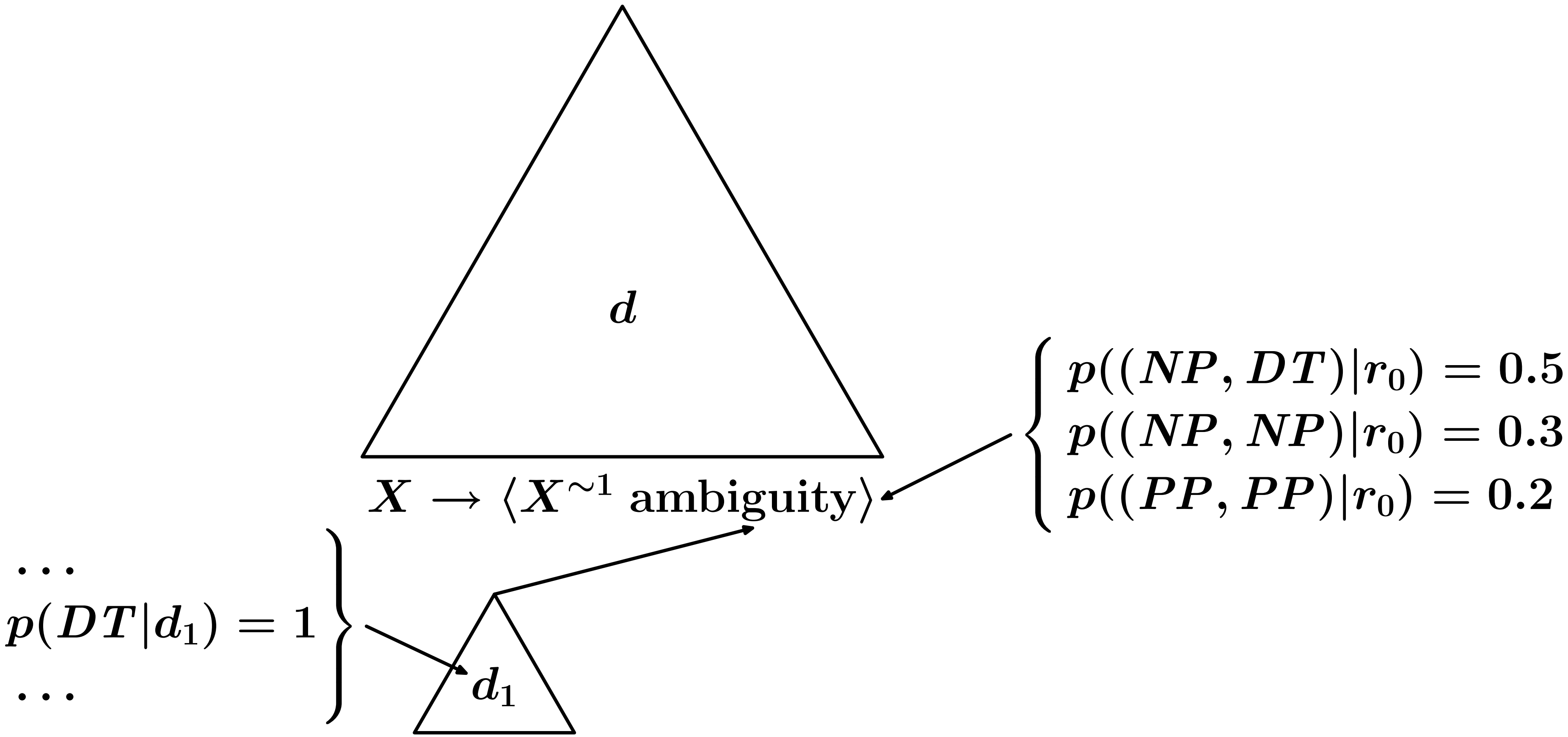
- $p(h_0|d_1)$ **is a computed distribution over all labels** $h_0 \in \mathcal{H}$ **for sub-derivation** d_1
- $p(\mathrm{h}|r_0)$ **is the distribution computed in the rule extraction for rule** r_0

Example

$$
\begin{aligned}
r_0 : X &\rightarrow \langle X^{\sim 1}\ \textbf{Zweideutigkeit},\ X^{\sim 1}\ \textbf{ambiguity}\rangle \\
&\left\{ \begin{array}{rcl} p((NP, DT)|r_0) &=& 0.5 \\ p((PP, PP)|r_0) &=& 0.3 \\ p((NP, NP)|r_0) &=& 0.2 \end{array} \right\} \\
r_1 : X &\rightarrow \langle \textbf{diese},\ \textbf{this}\rangle \\
&\left\{ p((DT)|r_1) = 1 \right\} \\
r_2 : X &\rightarrow \langle \textbf{diese},\ \textbf{such}\rangle \\
&\left\{ \begin{array}{rcl} p((JJ)|r_2) &=& 0.7 \\ p((PDT)|r_2) &=& 0.3 \end{array} \right\}
\end{aligned}
$$

Visualization

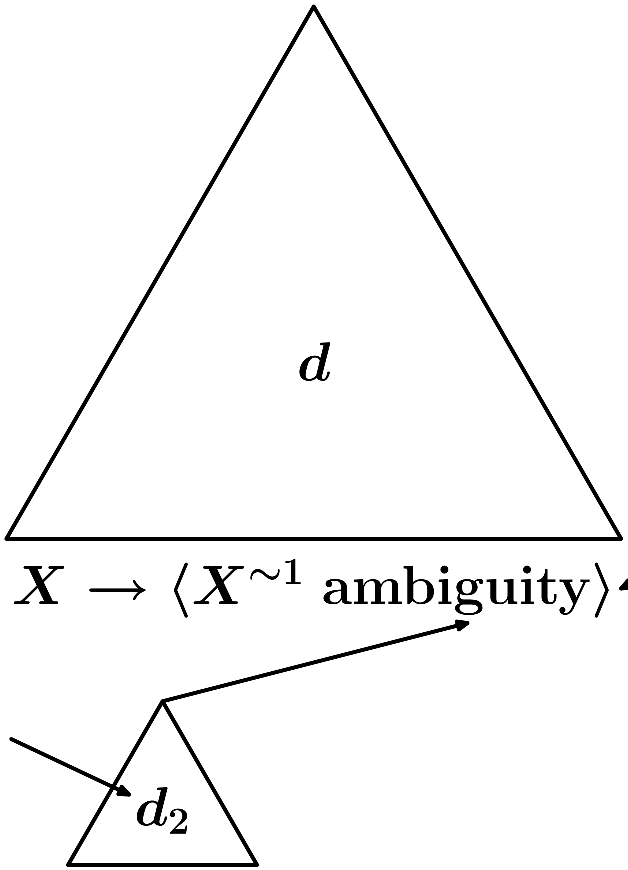
- sentence "diese Zweideutigkeit ..."
- translation "this ambiguity ..."



- $p_{syntax}(d) = 0.5 \cdot 1 + 0.3 \cdot 0 + 0.2 \cdot 0 = 0.5$

Visualization

- sentence "diese Zweideutigkeit ..."
- translation "such ambiguity ..."



$$X \rightarrow \langle X^{\sim 1} \text{ ambiguity} \rangle$$

$$\left\{ \begin{array}{l} p((NP, DT)|r_0) = 0.5 \\ p((NP, NP)|r_0) = 0.3 \\ p((PP, PP)|r_0) = 0.2 \end{array} \right.$$

$$\left. \begin{array}{l} \ldots \\ p(JJ|d_2) = 0.7 \\ p(PDT|d_2) = 0.3 \\ \ldots \end{array} \right\}$$

- $p_{syntax}(d) = 0.5 \cdot 0 + 0.3 \cdot 0 + 0.2 \cdot 0 = 0$

5 Results

- QUAERO 09 German-English (1.5 M training sentences)

	dev		test	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]
baseline	24.4	59.0	26.1	56.4
+ soft syntactic labels	24.8	58.6	26.1	56.4

- NIST 09 Chinese-English (1.1 M training sentences)

	NIST'06		NIST'08	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]
baseline	27.6	66.2	22.2	69.3
+ soft syntactic labels	28.4	65.6	22.6	69.2

6 Conclusion

- extension of the hierarchical system with soft syntactic labels
  - use syntax information of the target language to guide translation process
- small improvement on non-monotonic language pair Chinese-English
- outlook: analysis of the syntax parser influence
  - casing, tokenization, categories
  - domain depedence? (Stanford trained on Wall Street data)

thank you for your attention

References

[Chiang 05] D. Chiang. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pp. 263–270, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. 3

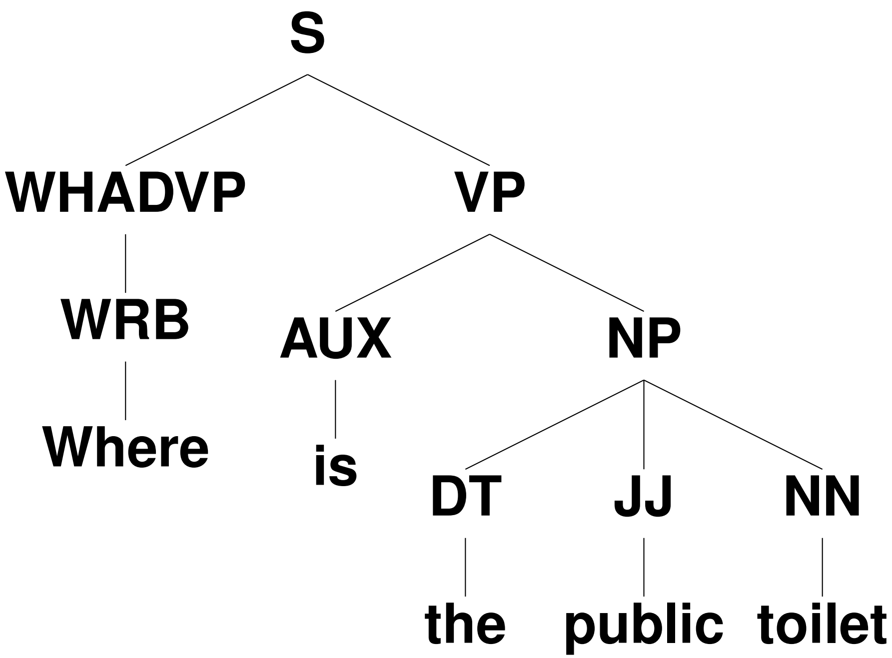
[Marcu & Wang+ 06] D. Marcu, W. Wang, A. Echihabi, K. Knight. Spmt: statistical machine translation with syntactified target language phrases. In *EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 44–52, Morristown, NJ, USA, 2006. Association for Computational Linguistics. 3

[Shen & Xu+ 08] L. Shen, J. Xu, R. Weischedel. A New String-to-Dependency Machine Translation Algorithm with a Target Dependency Language Model. In *46rd Annual Meeting on Association for Computational Linguistics*, pp. 577–585, Columbus, Ohio, June 2008. 3

[Venugopal & Zollmann+ 09] A. Venugopal, A. Zollmann, N. A. Smith, S. Vogel. Preference grammars: softening syntactic constraints to improve statistical machine translation. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American* *of the Association for Computational Linguistics*, pp. 236–244, Morristown, NJ, USA, 2009. Association for Computational Linguistics. 3

“Best match” node

- find the node in the parse tree that best matches the phrase
- minimize the number of words to be deleted or added to a phrase, so that it fits the yield of a node

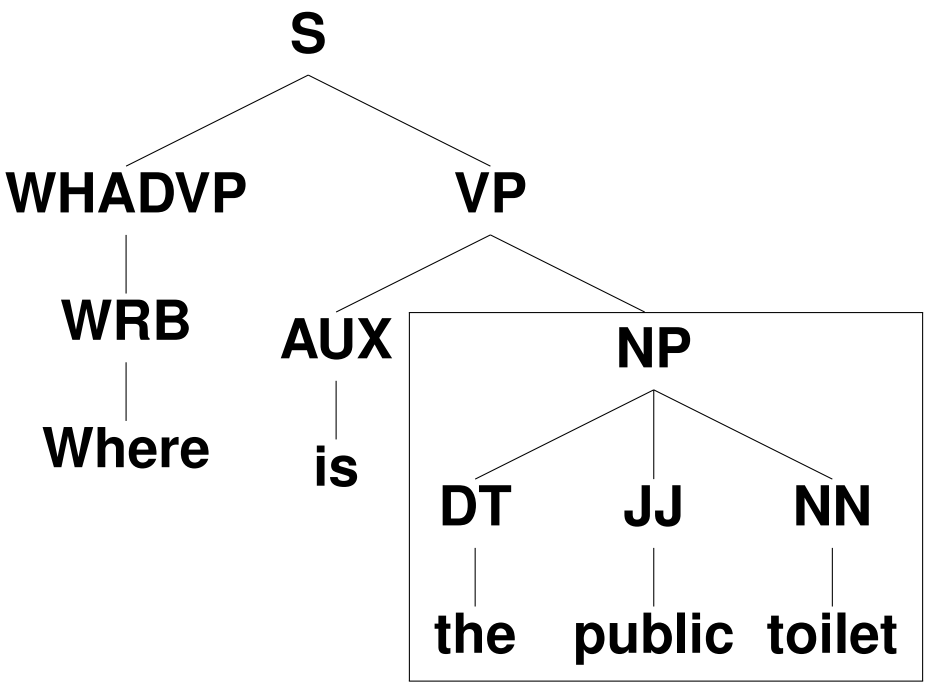


source phrases:

- public toilet
- is the

"Best match" node

- find the node in the parse tree that best matches the phrase
- minimize the number of words to be deleted or added to a phrase, so that it fits the yield of a node

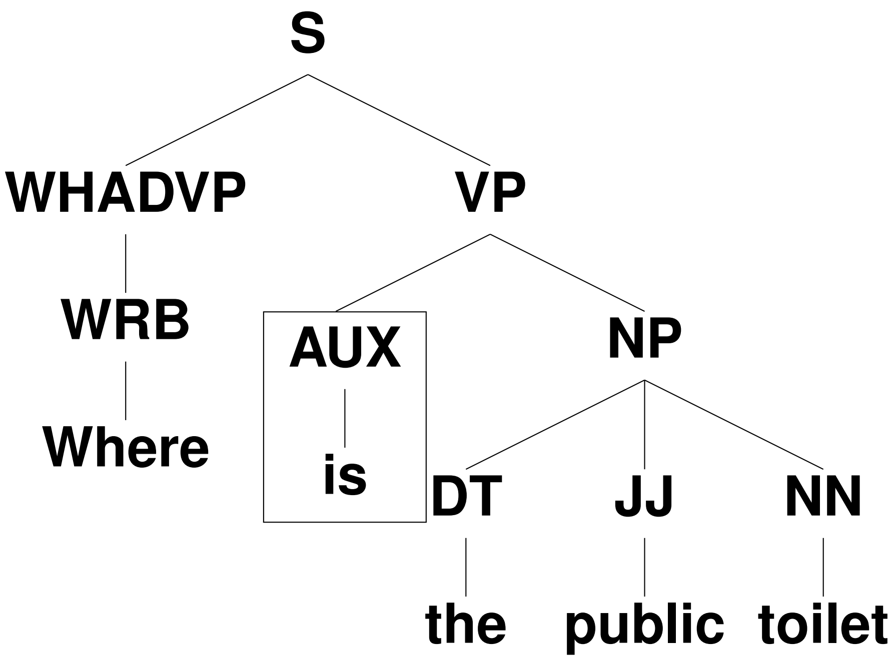


source phrases:

- public toilet: Node NP
- is the

“Best match” node

- find the node in the parse tree that best matches the phrase
- minimize the number of words to be deleted or added to a phrase, so that it fits the yield of a node



source phrases:

- public toilet: Node NP
- is the: Node AUX

Tree Well-Formedness Probability Model

- introduce additional model to measure the compatibility between two rules
  - rules with high mutual match should get a high probability
- d is a derivation using rules $r_1 \ldots r_i \ldots r_{|d|}$
- Let $p_{syntax}(d) = \prod_i^{|d|} p_{syntax}(r_i | r_{i+1}^{|d|})$ be the additional feature

Computation

- **1. $r_{|d|}$ is a lexical rule $X \rightarrow w$**
  - **calculate a probability distribution p for the non-terminals**

$$\begin{aligned} \forall h_0 \in \mathcal{H} : p(h_0|r_{|d|}) &= p((h_0)|r_{|d|}) \\ p_{syntax}(r_{|d|}) &= 1 \end{aligned}$$

Computation

- **2. r_i is a hierarchical rule $X \rightarrow wXv$**
  - **calculate new probability distribution p for the non-terminals**

$$
\begin{aligned}
\tilde{p}(h_0|r_i) &= \sum_{(h_0,h_1)\in H(r_i)} p((h_0,h_1)|r_i) \cdot p(h_1|r_{i+1}^{|d|}) \\
\forall h_0 \in \mathcal{H} : p(h_0|r_i) &= \frac{\tilde{p}(h_0|r_i)}{\sum_{h_0' \in \mathcal{H}} \tilde{p}(h_0'|r_i)} \\
p_{syntax}(r_i|r_{i+1}^{|d|}) &= \sum_{\mathrm{h}\in H(r_i)} p(\mathrm{h}|r_i) \cdot p(h_1|r_{i+1}^{|d|})
\end{aligned}
$$

QUAERO corpus statistics

		German	English
train:	Sentences	1 521 715	
	Running Words	41 009 835	41 695 098
	Vocabulary	177 031	119 140
	Singletons	66 985	45 575
dev:	Sentences	2 121	
	Running Words	56 029	45 211
	Vocabulary	9 454	10 325
	OOVs	1 121	6 131
test:	Sentences	2 007	
	Running Words	53 654	43 797
	Vocabulary	9 375	9 999
	OOVs	1 341	5 881

NIST corpus statistics

		Chinese	English
train:	Sentences	1 165 478	
	Running Words	30 545 919	31 351 263
	Vocabulary	69 804	180 921
	Singletons	15 782	82 502
dev:	Sentences	1 664	
	Running Words	42 930	194 885
	Vocabulary	6 387	9 673
	OOVs	1 897	6 935
test:	Sentences	1 357	
	Running Words	36 114	149 057
	Vocabulary	6 418	17 877
	OOVs	1 449	43 595