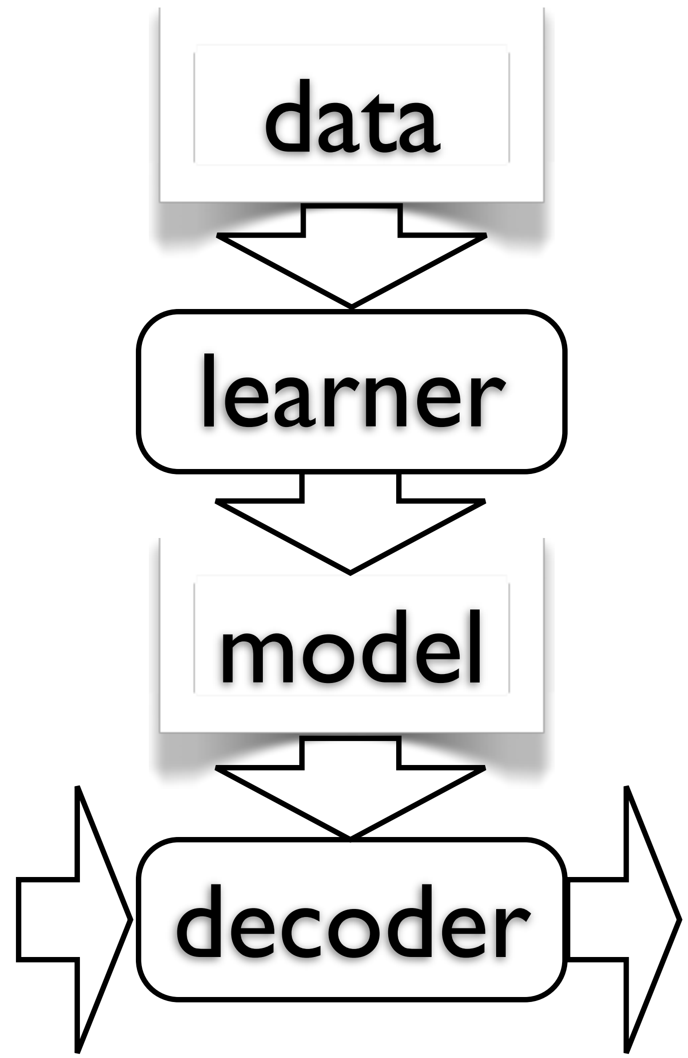
Statistical Machine Translation 2012

Taro Watanabe
taro.watanabe @ nict.go.jp

HIT-MSRA Summer School 2012

Machine Translation



黑山头口岸联检部门将原来要二至三天办完的出入境手续改为一天办完。

The United Inspection Department of Heishantou Port has shortened the procedures for leaving and entering the territory from originally 2 - 3 days to 1 day.

- A data-driven approach to MT
- We learn parameters from data assuming a “model”

Bilingual Data

1.上海浦东开发与法制建设同步
2.新华社上海二月十日电（记者谢金虎、张持坚）
3.上海浦东近年来颁布实行了涉及经济、贸易、建设、规划、科技、文教等领域的七十一件法规性文件，确保了浦东开发的有序进行。
4.浦东开发开放是一项振兴上海，建设现代化经济、贸易、金融中心的跨世纪工程，因此大量出现的是以前不曾遇到过的新情况、新问题。
5.对此，浦东不是简单的采取"干一段时间，等积累了经验以后再制定法规条例"的做法，而是借鉴发达国家和深圳等特区的经验教训，聘请国内外有关专家学者，积极、及时地制定和推出法规性文件，使这些经济活动一出现就被纳入法制轨道。
6.去年初浦东新区诞生的中国第一家医疗机构药品采购服务中心，正因为一开始就比较规范，运转至今，成交药品一亿多元，没有发现一例回扣。

1.The development of Shanghai's Pudong is in step with the establishment of its legal system
2.Xinhua News Agency, Shanghai, February 10, by wire (reporters Jinhu Xie and Chijian Zhang)
3.In recent years Shanghai's Pudong has promulgated and implemented 71 regulatory documents relating to areas such as economics, trade, construction, planning, science and technology, culture and education, etc., ensuring the orderly advancement of Pudong's development.
4.Pudong's development and opening up is a century-spanning undertaking for vigorously promoting Shanghai and constructing a modern economic, trade, and financial center. Because of this, new situations and new questions that have not been encountered before are emerging in great numbers.
5.In response to this, Pudong is not simply adopting an approach of "work for a short time and then draw up laws and regulations only after waiting until experience has been accumulated." Instead, Pudong is taking advantage of the lessons from experience of developed countries and special regions such as Shenzhen by hiring appropriate domestic and foreign specialists and scholars, by actively and promptly formulating and issuing regulatory documents, and by ensuring that these economic activities are incorporated into the sphere of influence of the legal system as soon as they appear.
6.Precisely because as soon as it opened it was relatively standardized, China's first drug purchase service center for medical treatment institutions, which came into being at the beginning of last year in the Pudong new region, in operating up to now, has concluded transactions for drugs of over 100 million yuan and hasn't had one case of kickback.

(part of LDC2007T02, English translation of Chinese treebank)

Channel Model

Channel Model

+ noise

encoder $\Rightarrow Y \Rightarrow$ channel $\Rightarrow X \Rightarrow$ decoder

$$
\begin{aligned}
\hat{y} &= \operatorname*{argmax}_y Pr(y|x) \\
&= \operatorname*{argmax}_y \frac{Pr(x|y)Pr(y)}{Pr(x)} \\
&= \operatorname*{argmax}_y Pr(x|y)Pr(y)
\end{aligned}
$$

f = source

e = target

$$\hat{\mathbf{e}} = \operatorname*{argmax}_{\mathbf{e}} Pr(\mathbf{f}|\mathbf{e})Pr(\mathbf{e})$$

- Employed in: ASR, OCR, MT...

Translation Model

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} \underbrace{\boxed{Pr(\mathbf{f}|\mathbf{e})}}_{\text{Translation Model}} \underbrace{\boxed{Pr(\mathbf{e})}}_{\text{Language Model}}$$

(Brown et al., 1990)

- Translation Model: adequacy of translation
- Language Model: grammatical correctness, consistent style, fluency

Language Model

$$
\begin{aligned}
Pr(\text{I do not know}) &= ? \\
Pr(\text{I not do know}) &= ?
\end{aligned}
$$

- Likelihood of a string of English words
- Usually modeled by ngrams

$$W = w_1, w_2, w_3, \cdots w_N$$

$$
\begin{aligned}
p(W) &= p(w_1, w_2, w_3, \cdots, w_N) \\
&= p(w_1)p(w_2|w_1)p(w_3|w_1, w_2)\cdots \\
&\qquad p(w_N|w_1, w_2, w_3, \cdots, w_{N-1})
\end{aligned}
$$

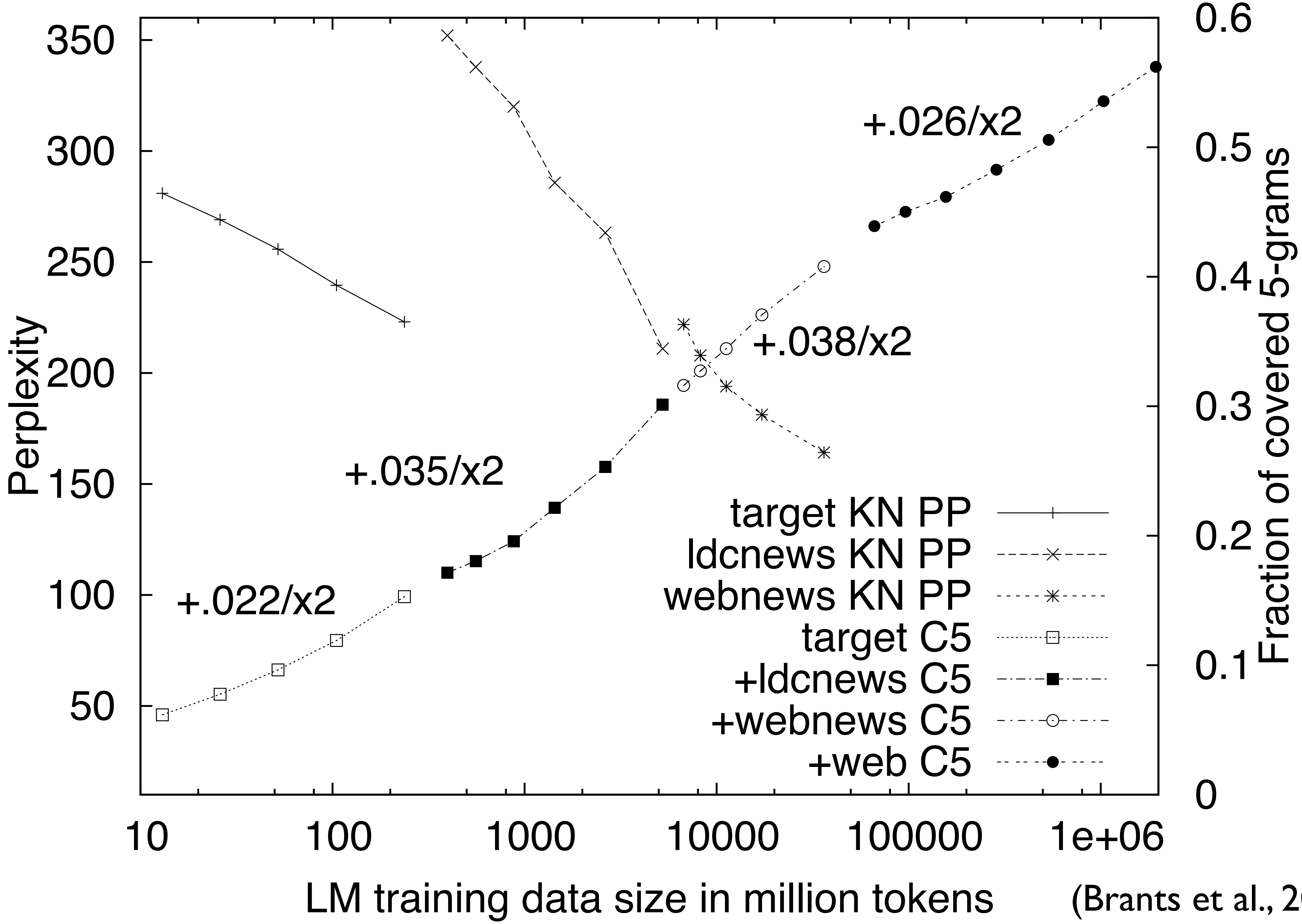
ngram Language Model

- Markov assumption: only n-word history is memorized
- Bigram:

$$p(\text{I do not know}) \quad = \quad p(\text{I})p(\text{do}|\text{I})p(\text{not}|\text{do})p(\text{know}|\text{not})$$

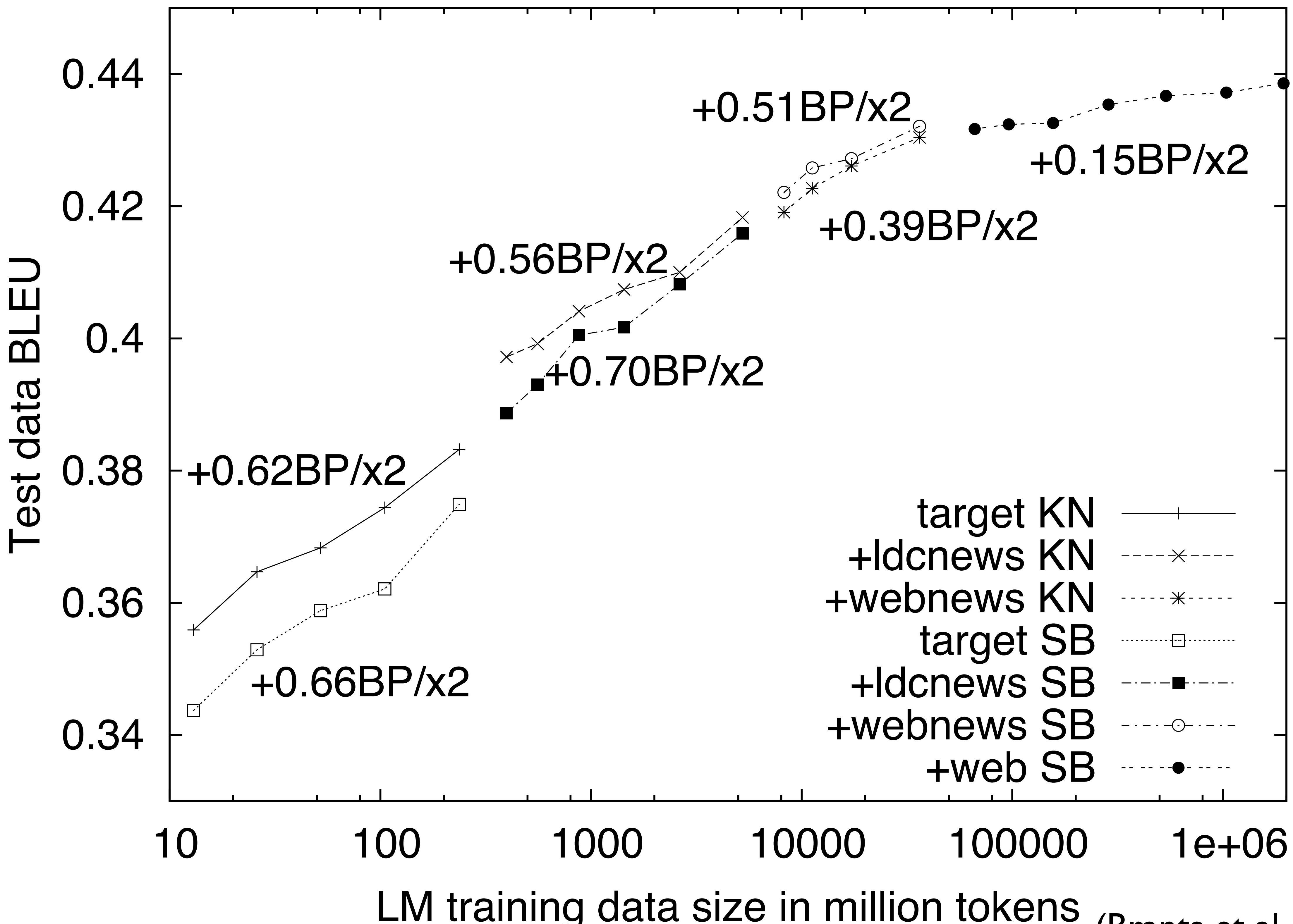
- Training: Maximum likelihood estimate + smoothing (Good-Turing, Witten-Bell, Kneser-Ney etc.)

Larger Data, Better LM



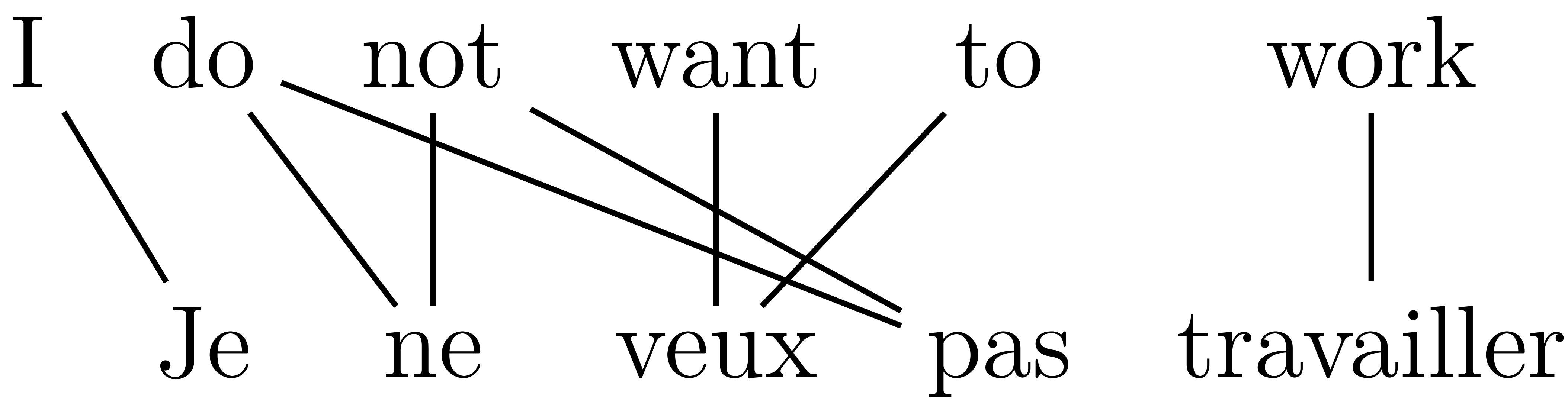
(Brants et al., 2007)

Better LM, Better MT



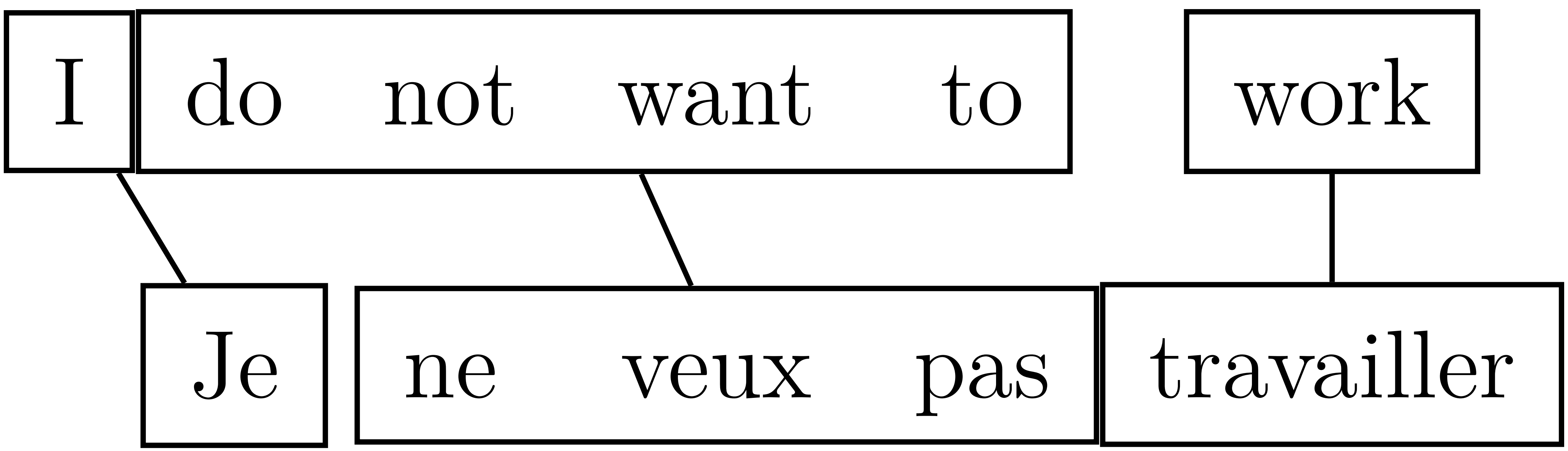
(Brants et al., 2007)

Word-based MT



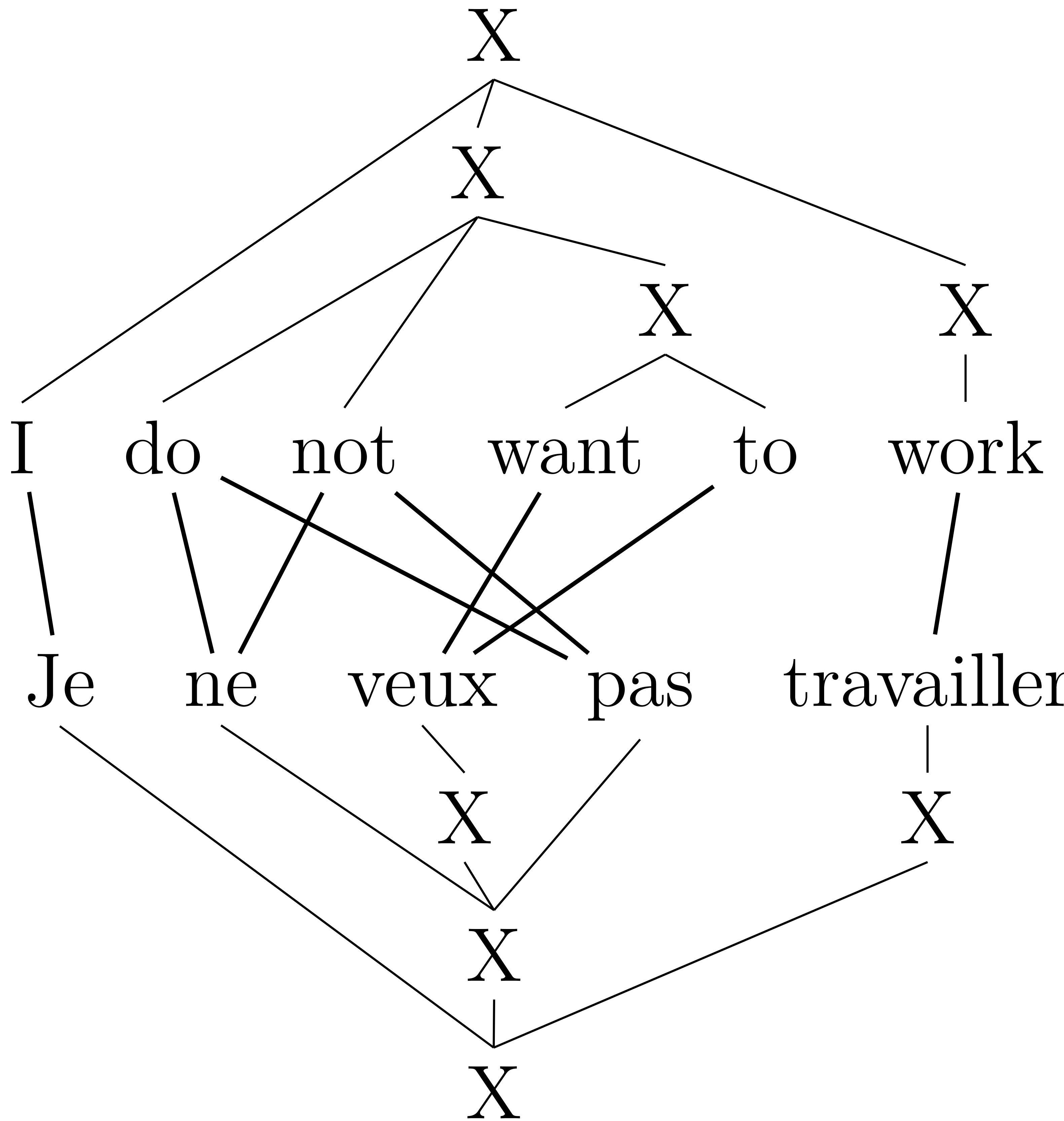
(Brown et al., 1993)

Phrase-based MT



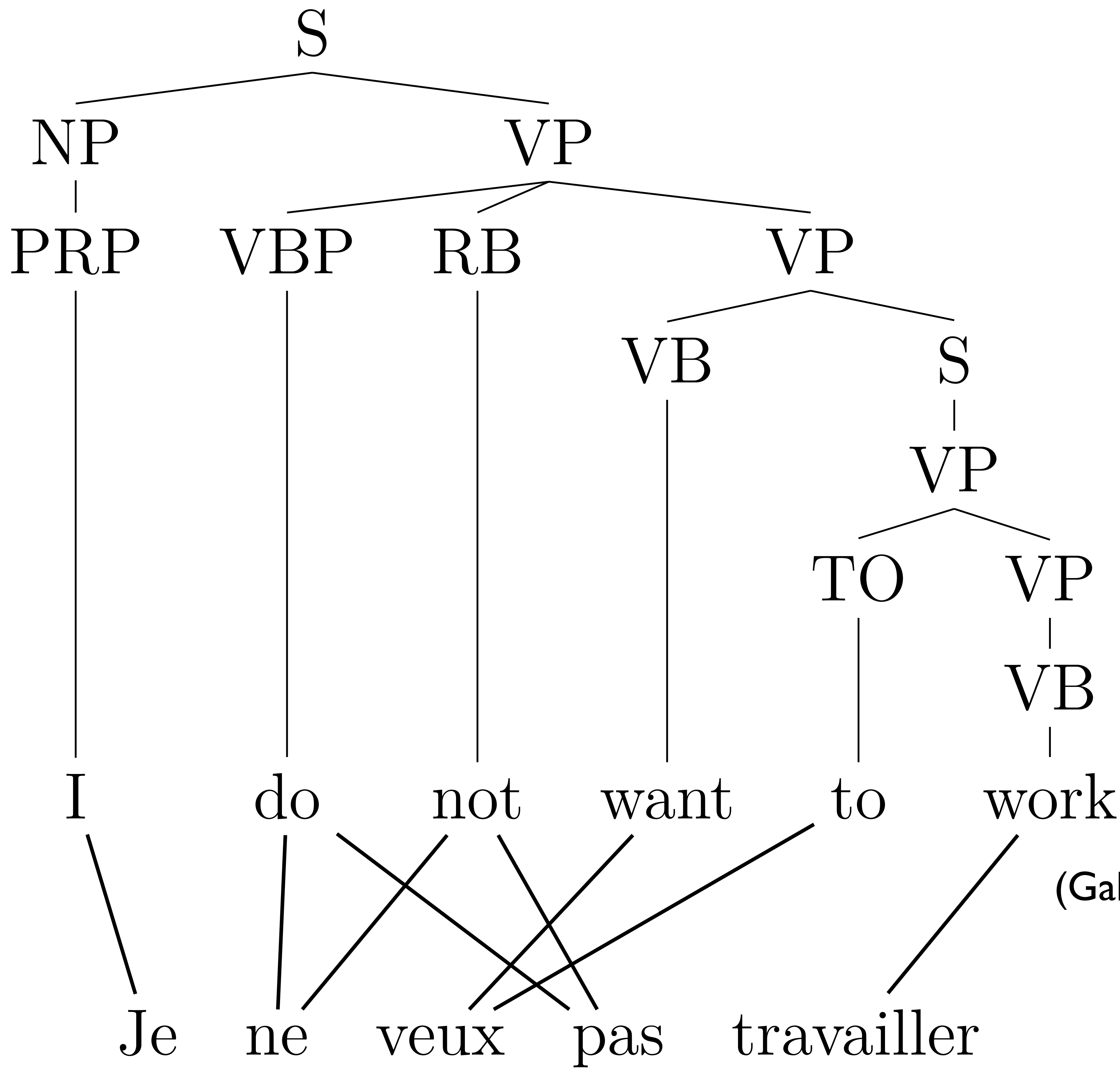
(Koehn et al., 2003)

Hierarchical PBMT



(Chiang, 2007)

Syntax-based MT



(Galley et al., 2004)

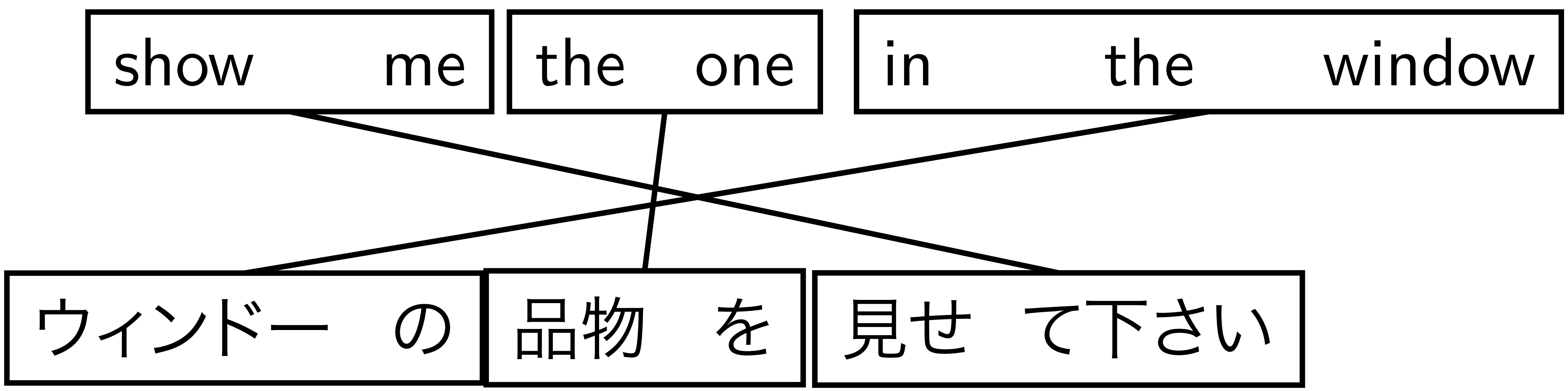
SMT2012

- Tutorial
  - Phrase-based MT
  - Tree-based MT
- Recent Topics
  - Phrase/rule induction
  - Tuning

Why Phrases?

- Grammar-less approach to MT
- Use phrases as a unit of translations
  - Directly handle many-to-many word correspondence + local reordering
  - Allow local context + non-compositional phrases
- Employed in many systems, including Google, NICT(VoiceTra, TexTra) and open-source, Moses (http://www.statmt.org/moses/)

Phrase-based Model



- Generative story:
  - f is segmented into phrases
  - Each phrase is translated
  - Translated phrases are reordered

Phrase-based Model

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} \frac{\exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\right)}{\sum_{\mathbf{e}', \phi'} \exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}', \phi', \mathbf{f})\right)} \\ &= \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\end{aligned}$$

- Maximization of a log-linear combination of multiple feature functions h(e, Φ, f)
- Φ: phrasal partition of f and e
- w: weight of feature functions

Questions

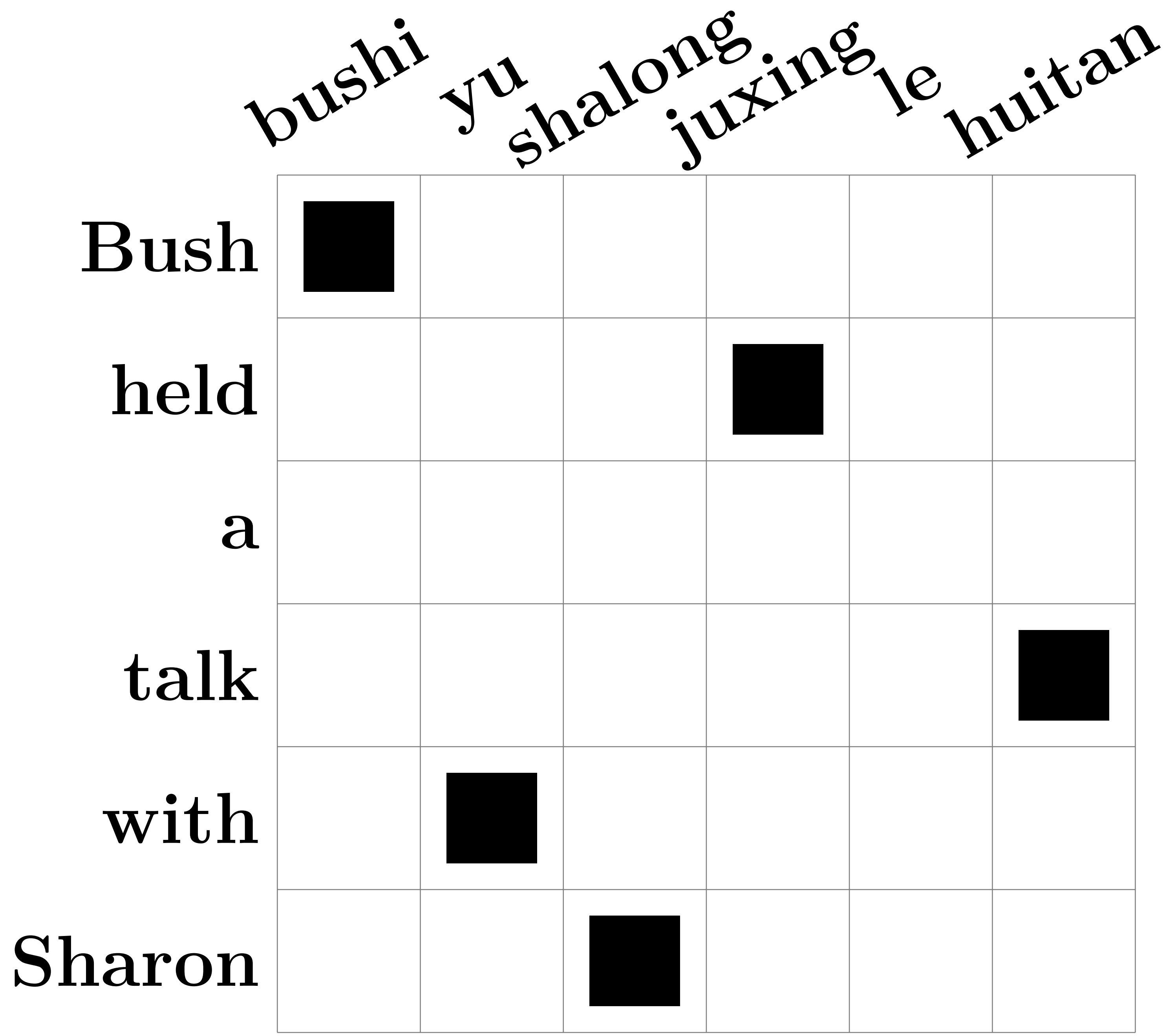
$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}}\, \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})$$

- Training: How to learn phrases and parameters (Φ and h)?
- Decoding (or search): How to find the best translation (argmax)?
- Tuning (or optimization): How to learn the scaling of features (w)?

Training

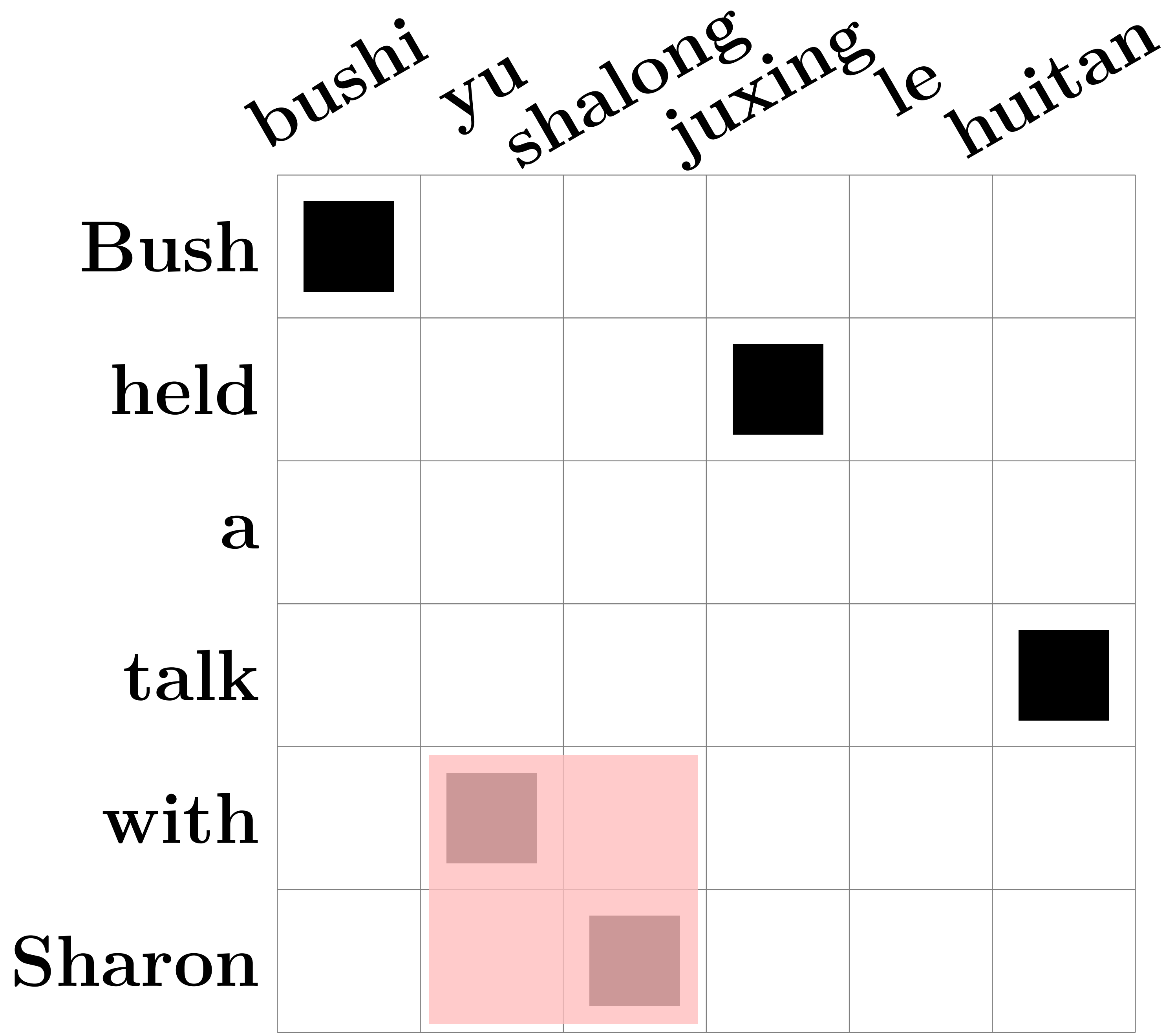
- Learn phrase pairs from $\mathcal{D} = \langle \mathcal{F}, \mathcal{E} \rangle$
- A standard heuristic approach (Koehn et al., 2003)
  - Compute word alignment
  - Extract phrase pairs
  - Score phrases

Word alignment



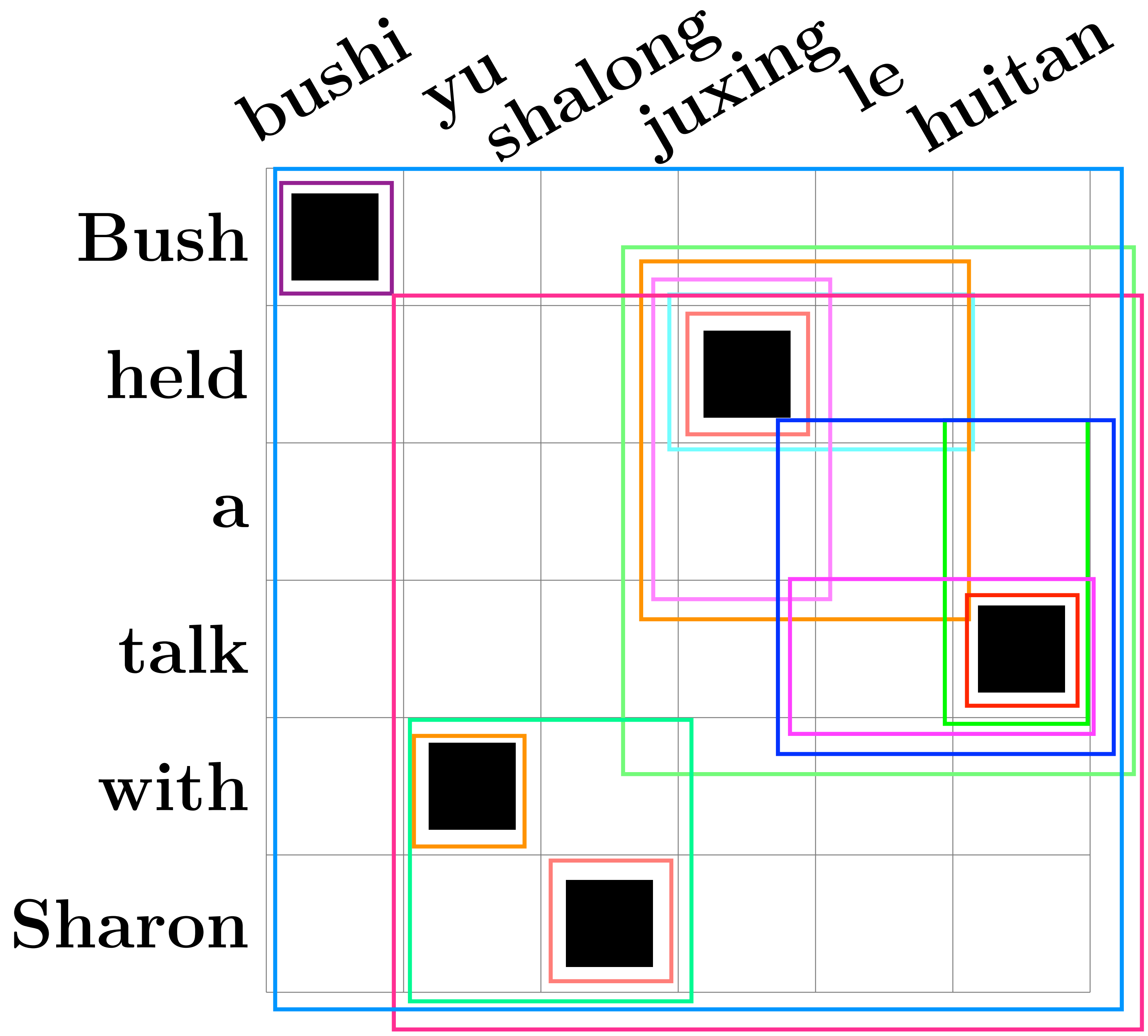
(Example from Huang and Chiang, 2007)

Extract Phrase Pairs



- From word alignment, extract a phrase pair consistent with word alignment

Exhaustive Extraction



- Exhaustively extract phrases from f, e

Features from Phrases

$$\log p_\phi(\bar{\mathbf{f}}|\bar{\mathbf{e}}) = \log \frac{\text{count}(\bar{\mathbf{e}}, \bar{\mathbf{f}})}{\sum_{\bar{\mathbf{f}}'} \text{count}(\bar{\mathbf{e}}, \bar{\mathbf{f}}')}$$

$$\log p_\phi(\bar{\mathbf{e}}|\bar{\mathbf{f}}) = \log \frac{\text{count}(\bar{\mathbf{e}}, \bar{\mathbf{f}})}{\sum_{\bar{\mathbf{e}}'} \text{count}(\bar{\mathbf{e}}', \bar{\mathbf{f}})}$$

- Collect all the phrase pairs from the data
- Maximum likelihood estimates by relative frequencies
- Employ scores in two directions

Features from Alignment

$$\log p_{lex}(\bar{\mathbf{f}}|\bar{\mathbf{e}},\bar{\mathbf{a}}) = \log \prod_{i}^{|\bar{\mathbf{e}}|} \frac{1}{|\{j|(i,j) \in \bar{\mathbf{a}}\}|} \sum_{\forall (i,j) \in \bar{\mathbf{a}}} t(e_i|f_j)$$

$$\log p_{lex}(\bar{\mathbf{e}}|\bar{\mathbf{f}},\bar{\mathbf{a}}) = \log \prod_{j}^{|\bar{\mathbf{f}}|} \frac{1}{|\{i|(j,i) \in \bar{\mathbf{a}}\}|} \sum_{\forall (j,i) \in \bar{\mathbf{a}}} t(f_j|e_i)$$

- Lexical weighing which scores by word translation probabilities
- Idea: counts for rare phrase pairs are unreliable
  - Smoothing effect by decomposing into word pairs

Example: Phrase Table

一直 往 里 走 ||| go along the inside to ||| -1.2729656758 -14.9932759574 0.0 -15.5778679932

一直 往 里 走 ||| go along the inside to the ||| -1.9195928407 -18.045853471 0.0 -15.6358281685

一直 往 里 走 ||| go inside and find it in the ||| -1.9195928407 -21.0681860363 0.0 -16.7303209435

一直 往 里 走 ||| go straight inside to ||| -1.2729656758 -9.7770695282 0.0 -12.525701484

一直 往 里 走 ||| go straight inside to the ||| -1.9195928407 -12.8296470418 0.0 -12.5836616593

不 熟悉 ||| 'm not familiar ||| -1.4859937213 -7.2301988107 -0.3036824138 -3.0311892056

不 熟悉 ||| do n't know ||| -1.2064088591 -5.3571402084 -3.4402617349 -6.8870595804

不 熟悉 ||| i 'm not familiar ||| -2.522085653 -9.1804032749 -1.06784063 -3.0311892056

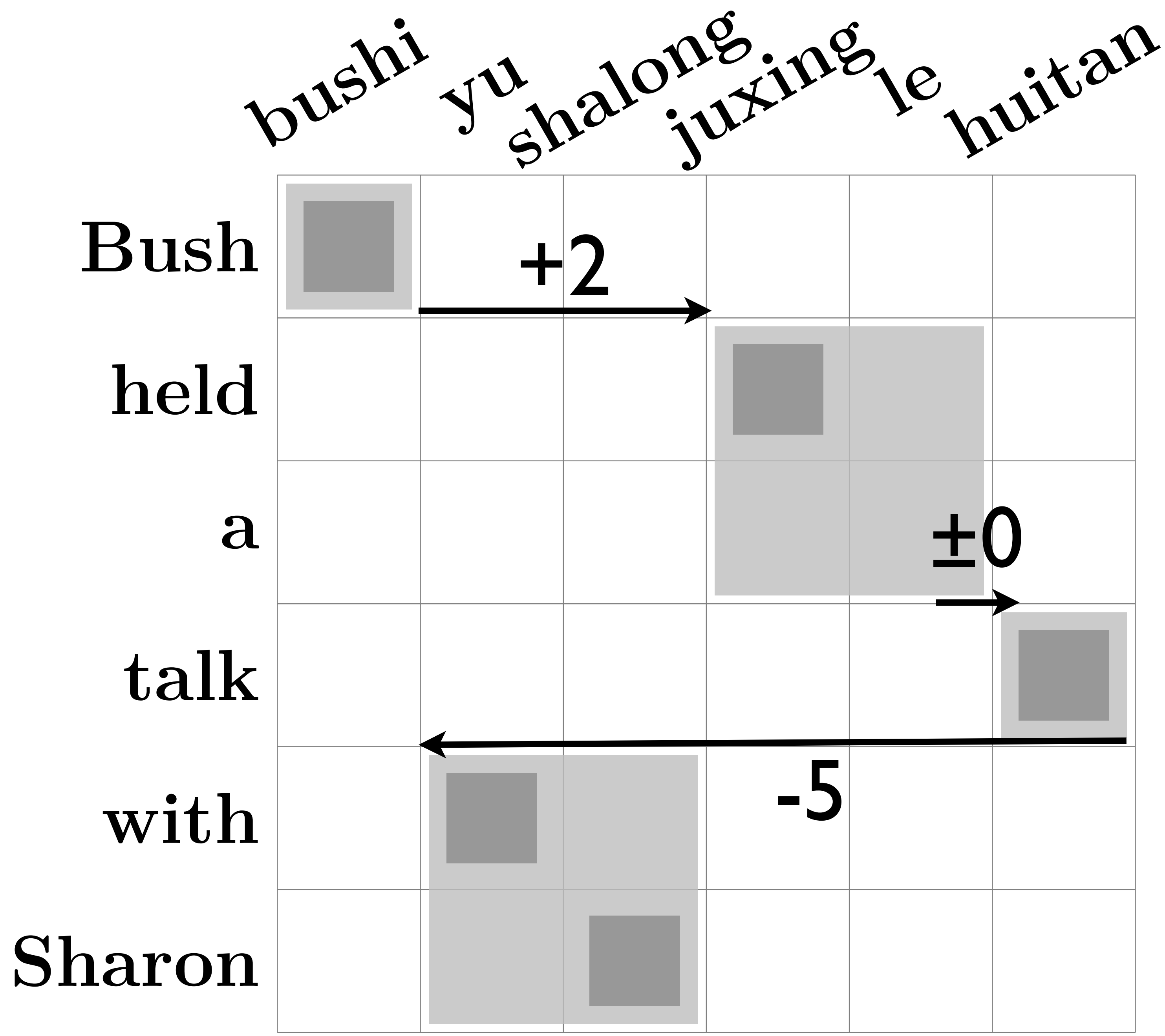
不 熟悉 ||| it will be great ||| -2.522085653 -20.871716142 0.0 -11.4593095552

不 熟悉 ||| not accustomed ||| -2.522085653 -5.5628513514 -0.6931471806 -2.2177906617

不 熟悉 ||| not accustomed to ||| -2.522085653 -8.5631752395 0.0 -2.2177906617

不 熟悉 ||| not familiar ||| -1.8754584881 -3.4150084505 -0.4212134651 -2.4210642434

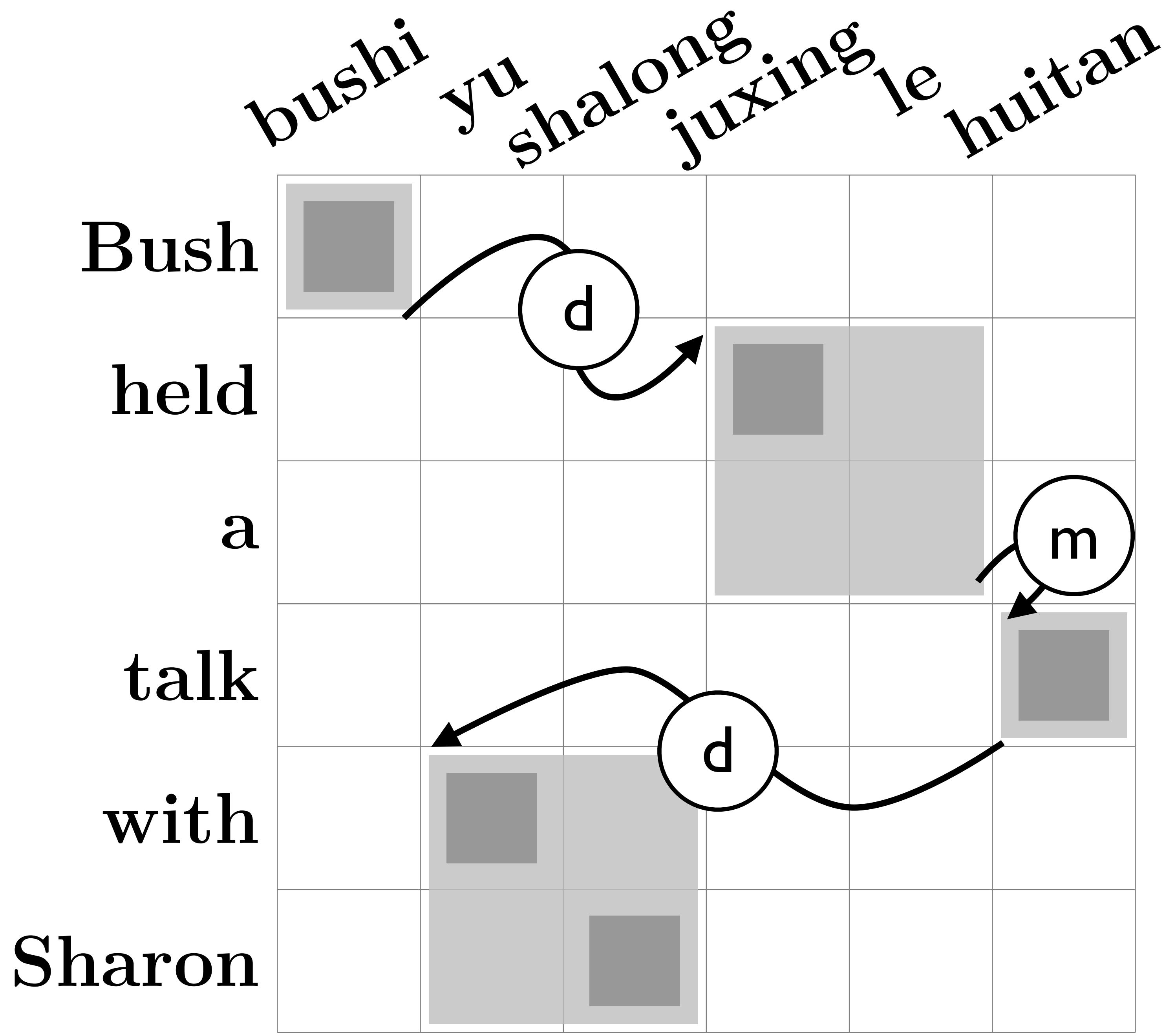
Features for Distortion



- Distance-based distortion modeling

$$d(\mathbf{f}, \phi, \mathbf{e}) = |+2| + |0| + |-5| = 7$$

Features for Reordering



- Fine grained reordering features: $\log p_o(o \in \{m, s, d\} \,|\bar{\mathbf{f}}, \bar{\mathbf{e}})$
- Either monotone, swap, discontinuous

Other Features

- log of ngram language model(s)
- word count: bias for ngram language model(s)
- phrase count: shorter or longer phrases

Direct Training

- Instead of word alignment + extraction pipeline, directly learn phrase-pairs (Marcu and Wong, 2002)
- Bayesian approach + blocked Gibbs sampling to learn parameters (Blunsom et al., 2009)
- Exhaustively memorize longer phrases (Neubig et al., 2011)

Questions

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}}\, \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})$$

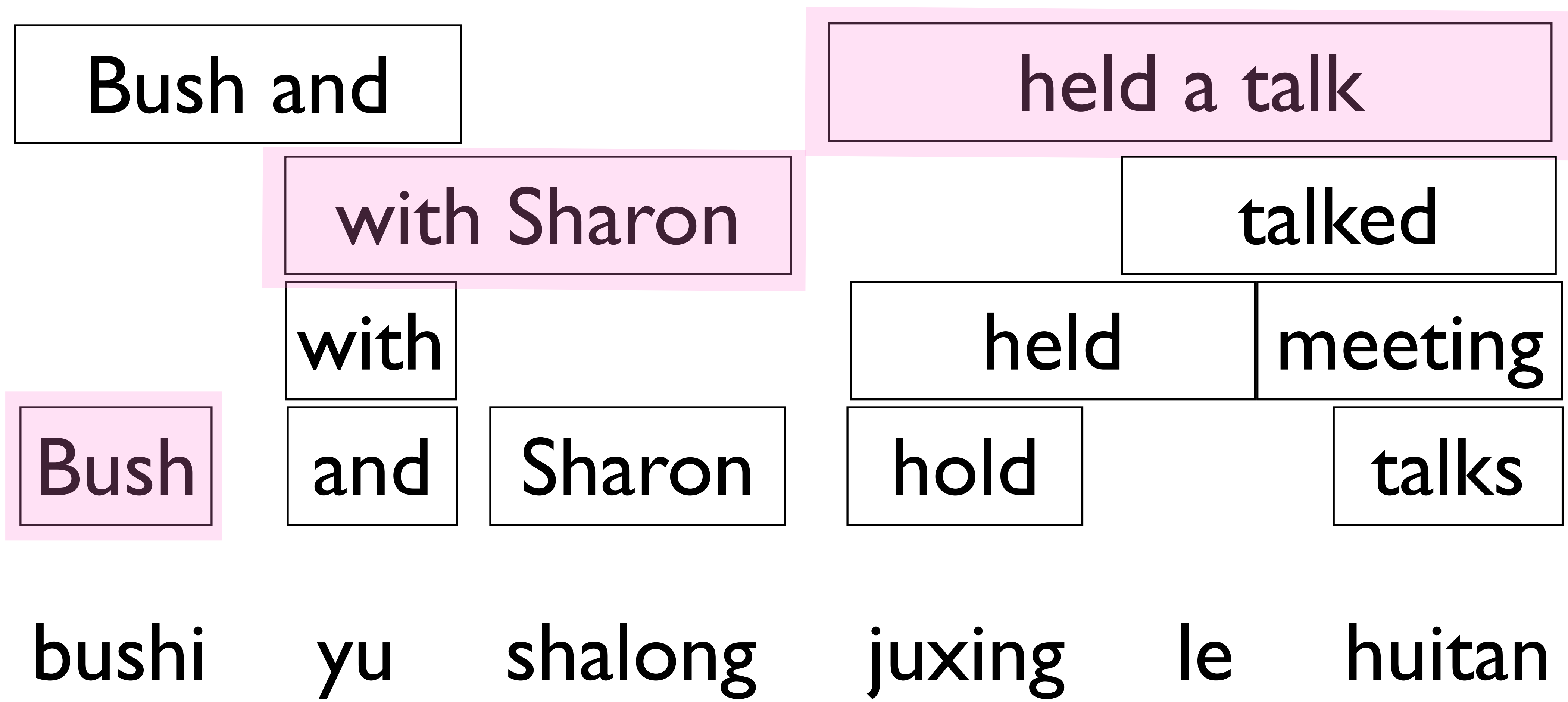
- Training: How to learn phrases and parameters (Φ and h)?
- Decoding (or search): How to find the best translation (argmax)?
- Tuning (or optimization): How to learn the scaling of features (w)?

Decoding

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} \frac{\exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\right)}{\sum_{\mathbf{e}', \phi'} \exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}', \phi', \mathbf{f})\right)} \\ &= \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\end{aligned}$$

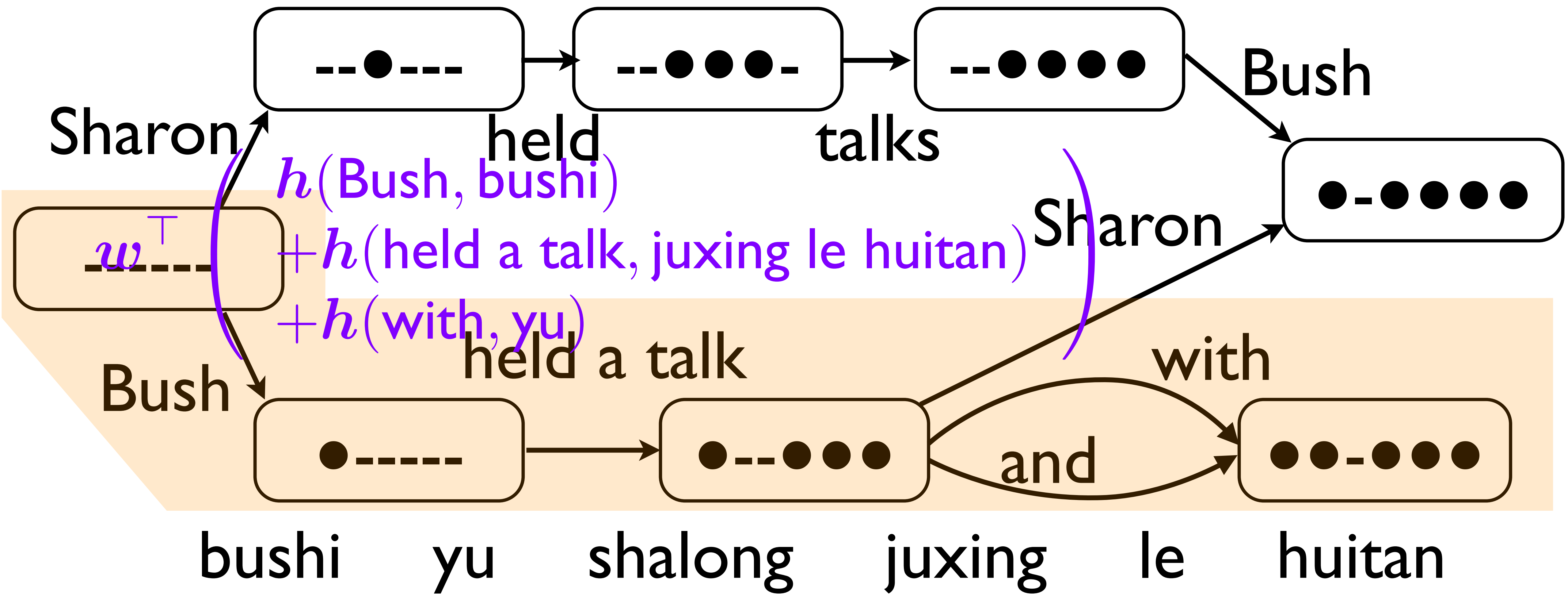
- Given an input sentence f and phrasal model h and w, find e with the highest score
- Potential errors:
  - Search error: we cannot find the best scored hypothesis
  - Translation error: highest scored hypothesis is bad

Enumerate Phrase Pairs



- Given a input sentence f, we can enumerate all possible phrases that match with the source side
- Choose the best phrase pair + ordering

Phrase-based Search Space



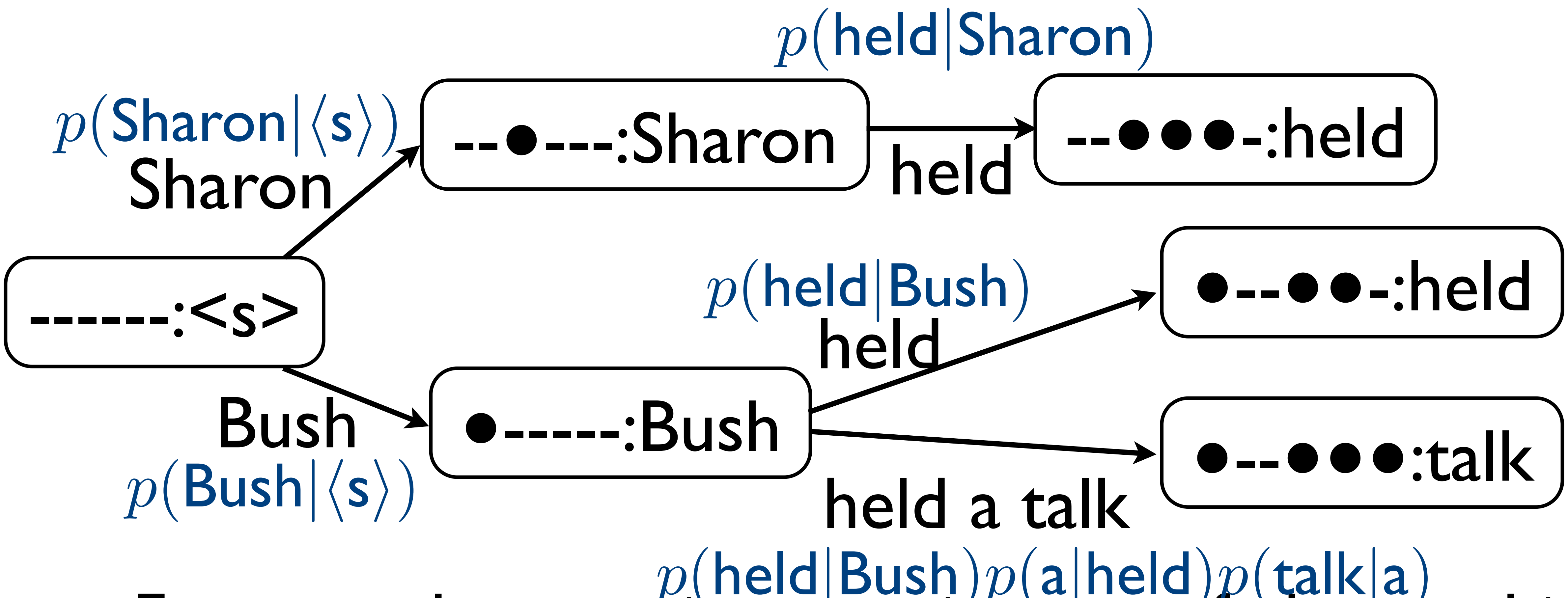
- Node: bit-vector representing covered source words
- Edge: phrasal translations, strictly left-to-right + score
- Search space: $O(2^n)$, Time: $O(2^n n^2)$ (Why?)

Traveling Salesman Problem

- NP-hard problem: visit each city only once
- MT as a Traveling Salesman Problem (Knight, 1999)
  - Each source word corresponds to a city
  - A Dynamic Programming solution:
    - State: visited cities (bit-vector)
    - Search space: $O(2^n)$
  - Distortion limit to reduce search space

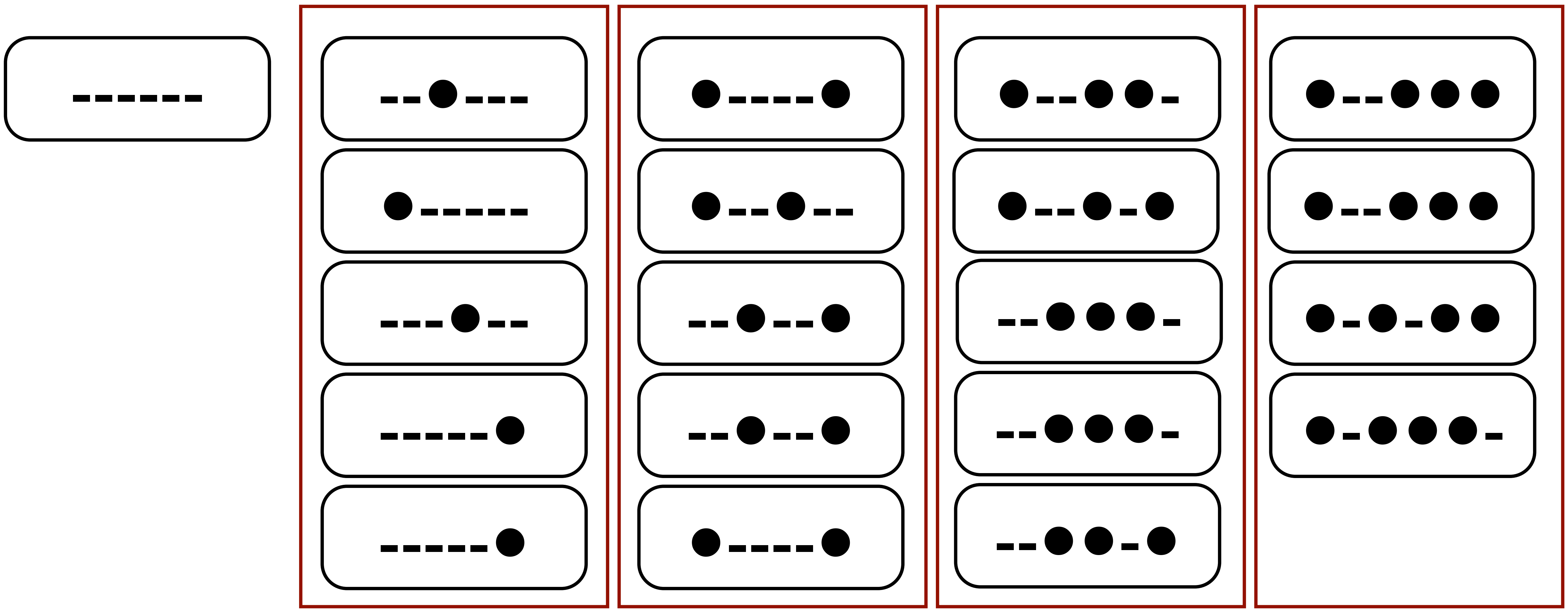
    i.e. long distortion: ●----- → ●----●

Non-local features



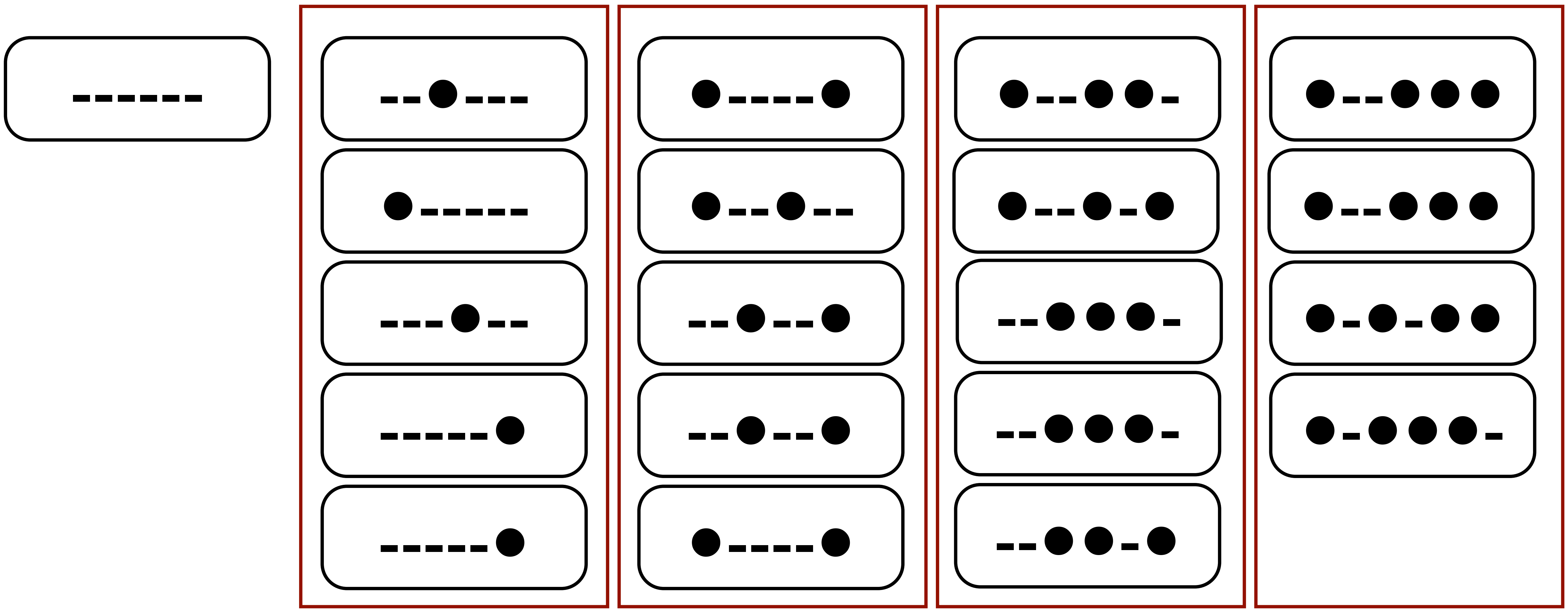
- Features that requires scoring out of phrases: bigram language model
- Additional state representation required for "future scoring": 1-word for bigram LM
- Space: $O(2^n V^{m-1})$, Time: $O(2^n V^{m-1} n^2)$ for m-gram LM

Phrase-based Decoding



- Re-organize the search space by the cardinality (= # of covered source words)
- Expand hypotheses from the smallest cardinality first

Pruning



- Prune low scored hypotheses in a bin sharing the same cardinality
- Expand survived hypotheses only (Koehn et al., 2003; Och and Ney, 2004)

Questions

$$\hat{\mathbf{e}} = \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})$$

- Training: How to learn phrases and parameters (Φ and h)?
- Decoding (or search): How to find the best translation (argmax)?
- Tuning (or optimization): How to learn the scaling of features (w)?

Tuning

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} \frac{\exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\right)}{\sum_{\mathbf{e}', \phi'} \exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}', \phi', \mathbf{f})\right)} \\ &= \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\end{aligned}$$

- Three popular objectives (in SMT) for tuning w
  - (Direct) Error Minimization (Och, 2003)
  - Maximum Entropy (Och and Ney, 2002)
  - Large Margin (Watanabe et al., 2007; Chiang et al., 2008; Hopkins and May, 2011)

(Direct) Minimum Error

$$\hat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}} \sum_{s=1}^{S} l(\operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}, \mathbf{f}_s), \mathbf{e}_s)$$

- MERT (Minimum ERror Training)
- Standard in SMT (but not in other NLP areas, such as tagging etc.)
  - We can incorporate arbitrary error functions, l
  - "Summation" can be replaced by document-wise BLEU specific summation
  - 10+ real valued features

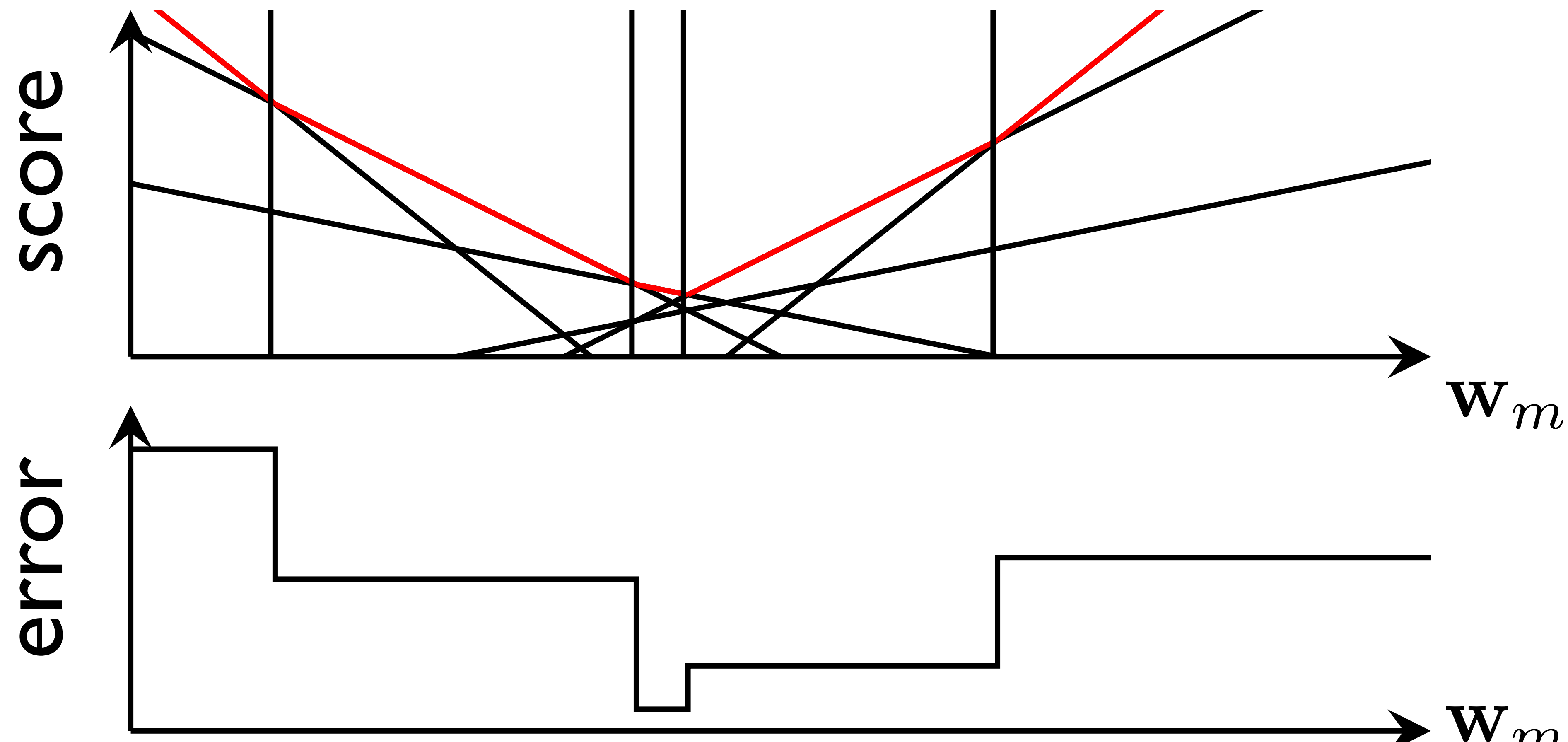
n-best Approximation

procedure MERT($\{(\mathbf{e}_s, \mathbf{f}_s)\}_{s=1}^{S}$)
    for $n = 1 \ldots N$ **do**
        Decode and generate nbest list using $\mathbf{w}$
        Merge nbest list
        for $k = 1 \ldots K$ **do**
            for each parameter $m = 1 \ldots M$ **do**
                Solve one dimensional optimization
            end for
            update $\mathbf{w}$
        end for
    end for
end procedure

- N iterations, with each iteration, n-bests are generated and merged
  - K iterations, with each iteration, M dimensions are tried (M = # of features), and w is updated

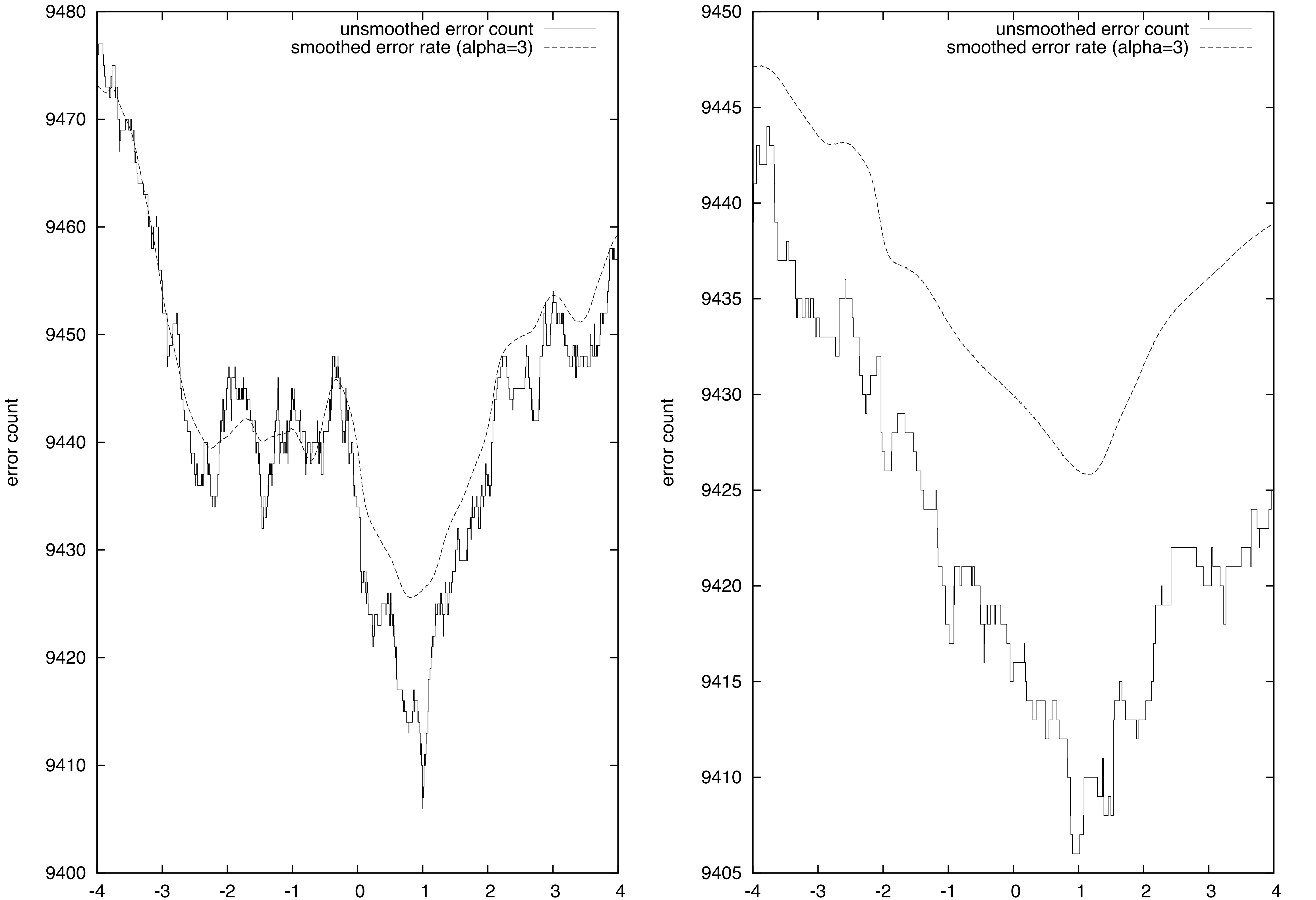
Efficient Line Search

$$\hat{\mathbf{e}} = \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}_m^\top \cdot \underbrace{\mathbf{h}_m(\mathbf{e}, \mathbf{f}_s)}_{\text{slope}} + \underbrace{\mathbf{w}_{m\_}^\top \cdot \mathbf{h}_{m\_}(\mathbf{e}, \mathbf{f}_s)}_{\text{constant}}$$



- If we choose one dimension m, and others fixed, we can treat each hypothesis e as a "line"
- Compute convex hull of a set of "lines"

Error Surface



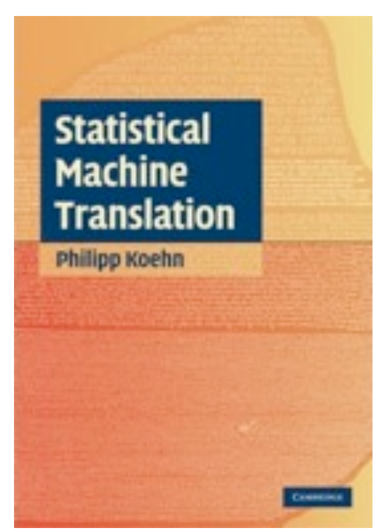
(Och, 2003)

MERT in Practice

- Many random starting points (Macherey et al., 2008; Moore and Quirk, 2008)
- Many random directions (Macherey et al., 2008)
- Error count smoothing (Cer et al., 2008)
- Regularization (Hayashi et al., 2009)
- Multi-dimensional search by efficiently computing convex hull (Galley and Quirk, 2011)
- MERT at least 3 times, and report average BLEU (Clark et al., 2011)

Answered?

- Grammar-less model (but very strong)
- Fast decoding
- Why MERT? (Good for non-binary, numerical features)
- Software: Moses: http://www.statmt.org/moses/
- Further readings:



SMT2012

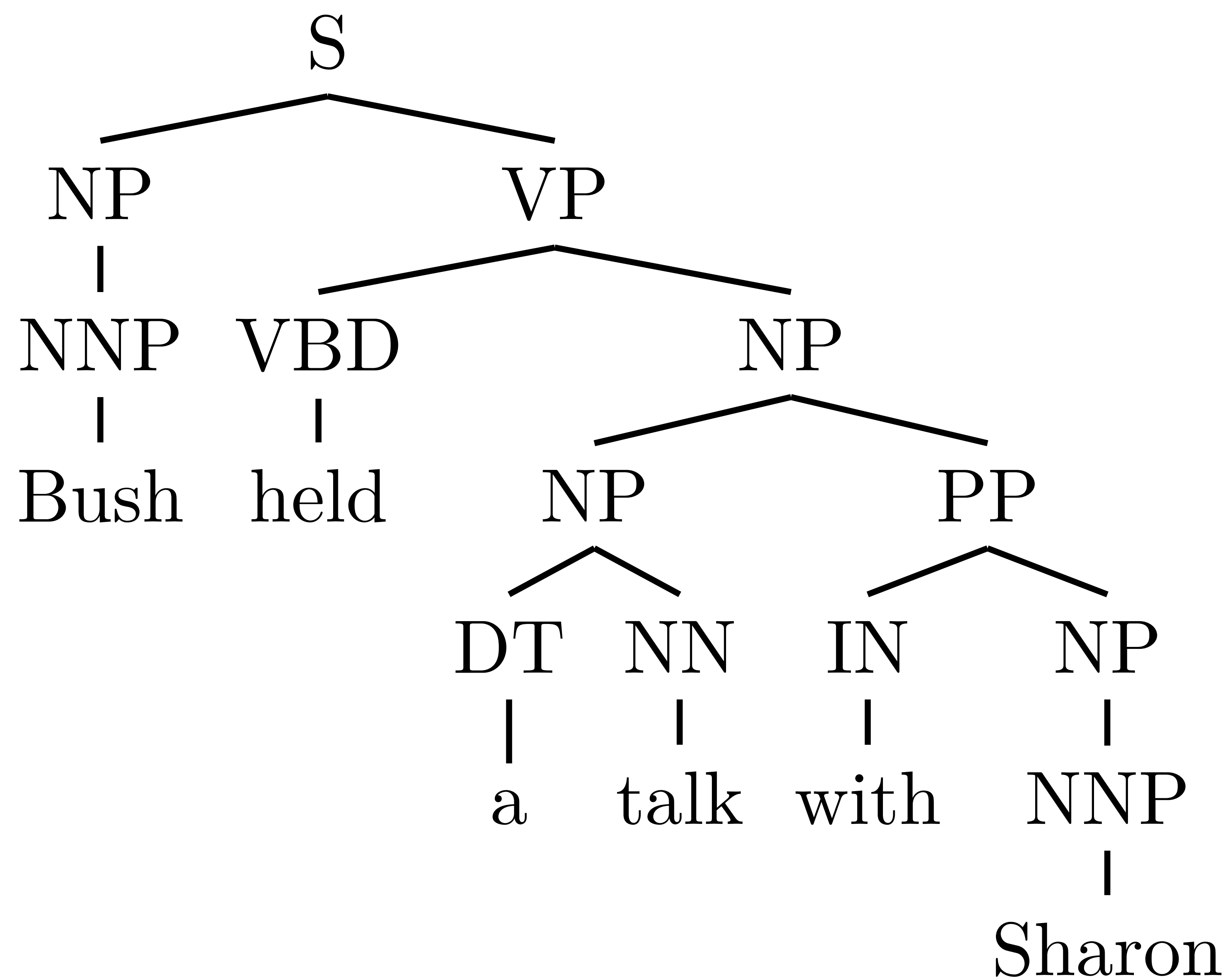
- Tutorial
  - Phrase-based MT
  - Tree-based MT
- Recent Topics
  - Phrase/rule induction
  - Tuning

Tree-based MT

- Backgrounds
  - CFG, parsing, hypergraph, deductive system semirings
- Tree-based SMT
  - Synchronous-CFG
  - String-to-Tree, Tree-to-String

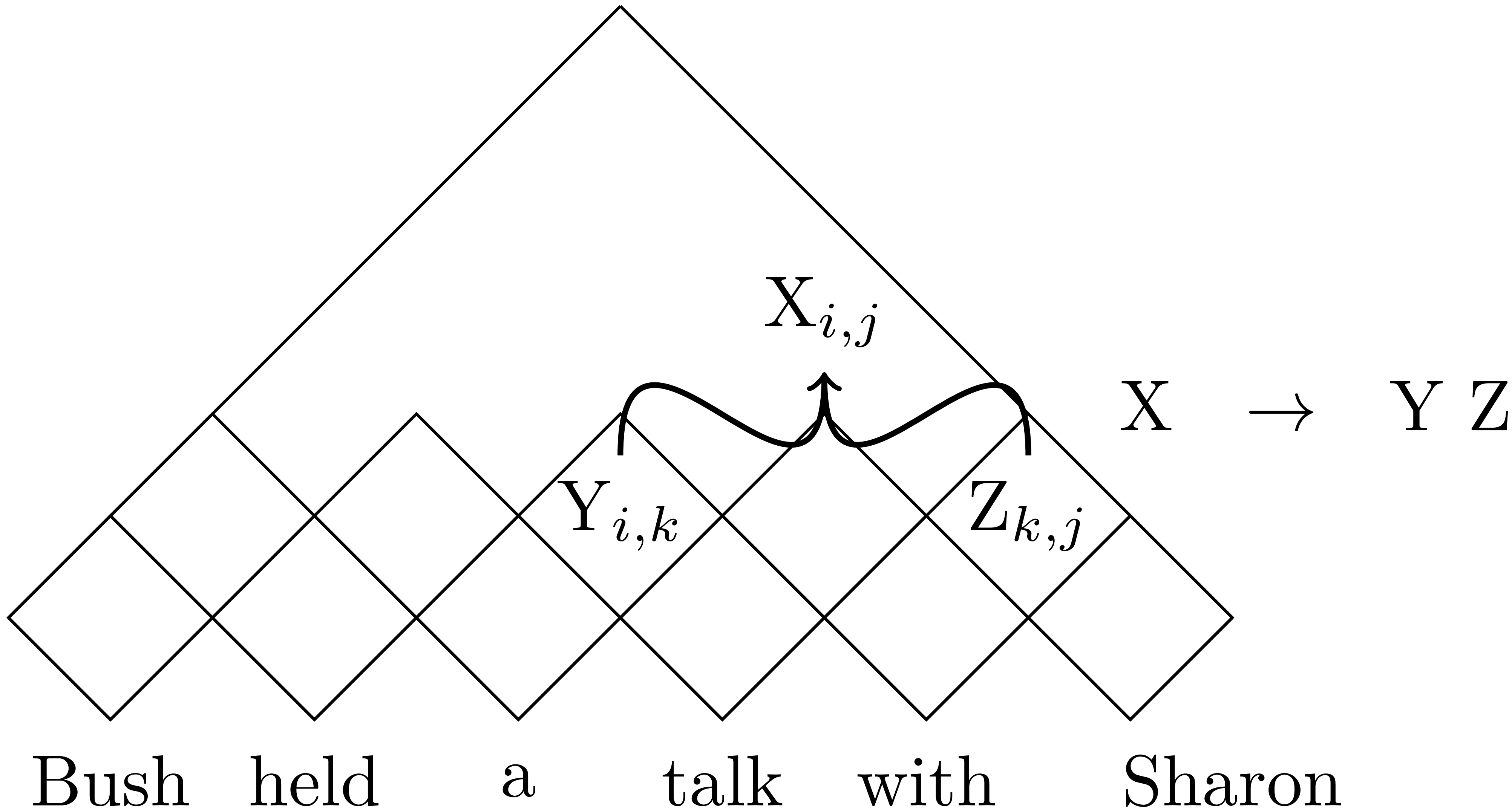
Backgrounds: CFG

$$
\begin{aligned}
\mathrm{S} &\rightarrow \mathrm{NP\ VP} \\
\mathrm{NP} &\rightarrow \mathrm{NNP} \\
\mathrm{NP} &\rightarrow \mathrm{NP\ PP} \\
\mathrm{NP} &\rightarrow \mathrm{DP\ NN} \\
\mathrm{NP} &\rightarrow \mathrm{DT\ NN} \\
\mathrm{VP} &\rightarrow \mathrm{VBD\ NP} \\
\mathrm{NNP} &\rightarrow \mathrm{Bush} \\
\mathrm{VBD} &\rightarrow \mathrm{held} \\
&\vdots
\end{aligned}
$$



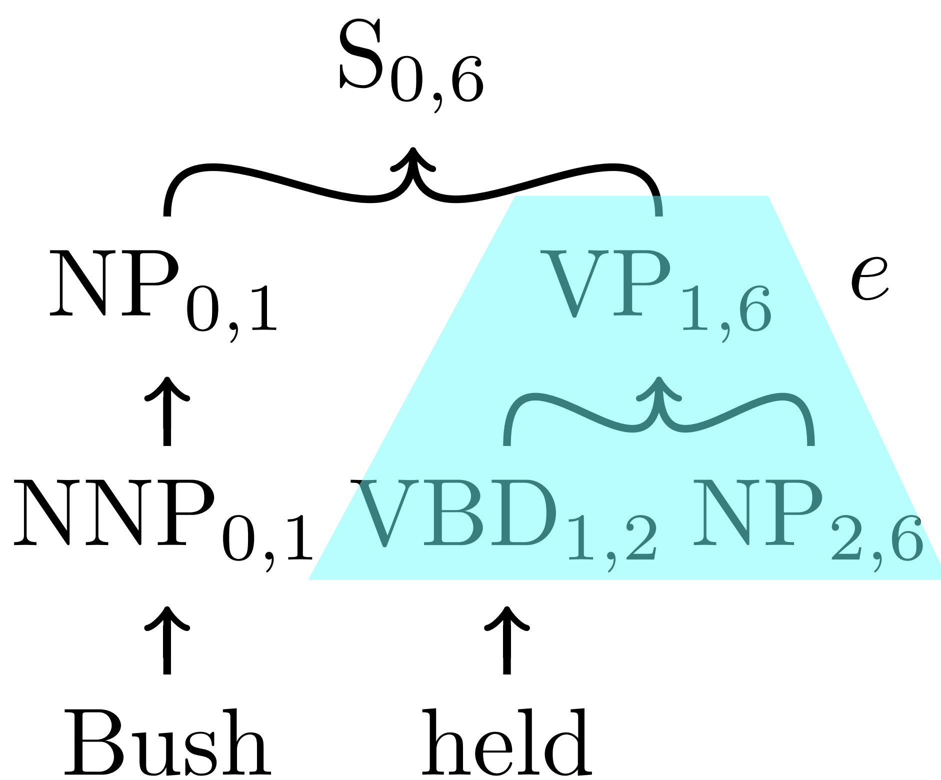
- parsing = intersection of CFG with a string (regular grammar)

Parsing: CKY

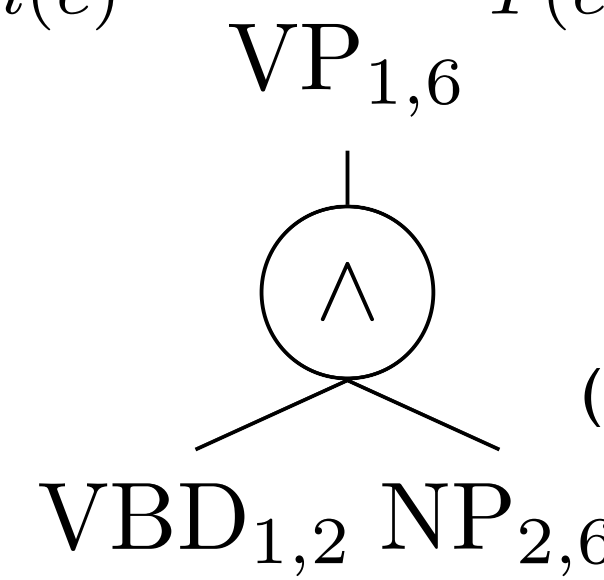


- $O(n^3)$: For each length n, for each position i, for each rule X → Y Z, for each split point k
- (Bottom-up) topological order

Hypergraph



$$e = \langle \underbrace{\mathrm{VP}_{1,6}}_{h(e)}, \underbrace{\{\mathrm{VBD}_{1,2}, \mathrm{NP}_{2,6}\}}_{T(e)} \rangle$$



(Klein and Manning, 2001)

- Generalization of graphs:
  - h(e): head node of hyperedge e
  - T(e): tail node(s) of hyperedge e, arity = |T(e)|
  - hyperedge = instantiated rule
- Represented as and-or graphs

Deductive System

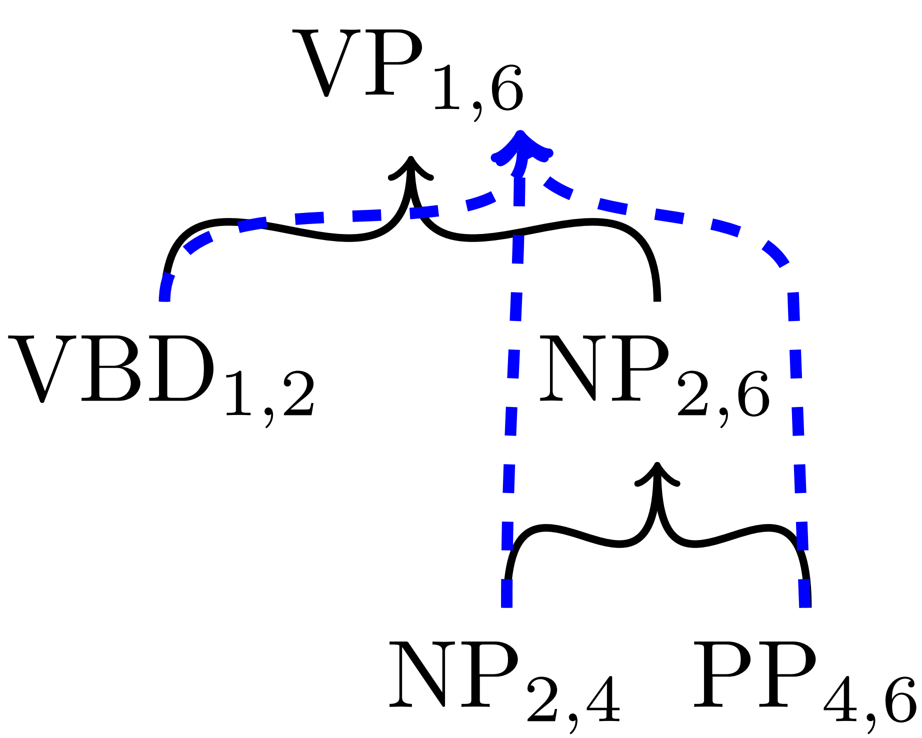
$$\underbrace{\mathrm{VBD}_{1,2} \quad \mathrm{NP}_{2,6}}_{\mathrm{VP}_{1,6}}$$

$$\frac{\overbrace{\mathrm{VBD}_{1,2}\ \mathrm{NP}_{2,6}}^{\text{antecedents}}}{\underbrace{\mathrm{VP}_{1,6}}_{\text{consequent}}}\ \mathrm{VP}_{[i,j]} \rightarrow \mathrm{VBZ}_{[j,k]}\ \mathrm{NP}_{[i,k]}$$

(Shieber et al., 1995)

- Parsing algorithm as a deductive system
- We start from initial items (axioms) until we reach a goal item
- If antecedents are proved, its consequent is proved
- deduction = hyperedge

Packed Forest



$$\frac{\mathrm{VBD}_{1,2} \quad \dfrac{\mathrm{NP}_{2,4}\ \mathrm{PP}_{4,6}}{\mathrm{NP}_{2,6}}}{\mathrm{VP}_{1,6}}$$

$$\frac{\mathrm{VBD}_{1,2}\ \mathrm{NP}_{2,4}\ \mathrm{PP}_{4,6}}{\mathrm{VP}_{1,6}}$$

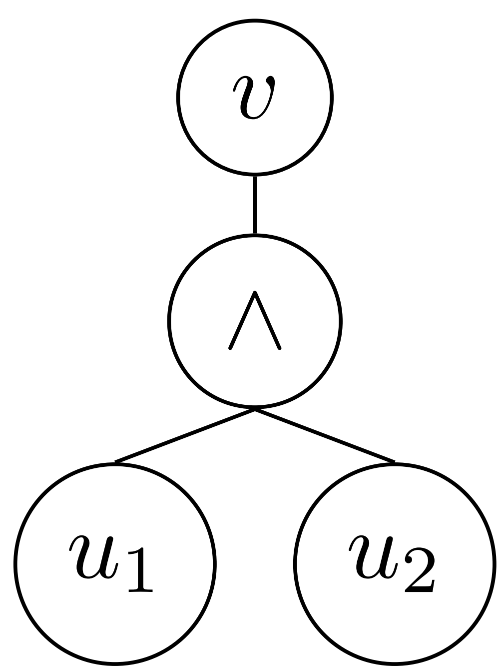
(Klein and Manning, 2001; Huang and Chiang, 2005)

- A polynomial space encoding of exponentially many parses by sharing common sub-derivations
- Single derivation = tree

Summary of Formalisms

hypergraph	AND/OR graph	CFG	deductive system
vertex source-vertex target-vertex	OR-node leaf OR-node root OR-node	symbol terminal start symbol	item axiom goal item
hyperedge	AND-node	production	instantiated deduction

$\langle v, \{u_1, u_2\}\rangle$



$v \rightarrow u_1\ u_2$

$$\frac{u_1\ \ u_2}{v}$$

Weights and Semirings

$$\mathrm{VP} \xrightarrow{w_1} \mathrm{VBD}\ \mathrm{NP}$$

$$\mathrm{NP} \xrightarrow{w_2} \mathrm{NP}\ \mathrm{PP}$$

$$\overbrace{\mathrm{VBD}_{1,2} : c \quad \mathrm{NP}_{2,6} : d}^{\mathrm{VP}_{1,6} : w_1 \otimes c \otimes d}$$

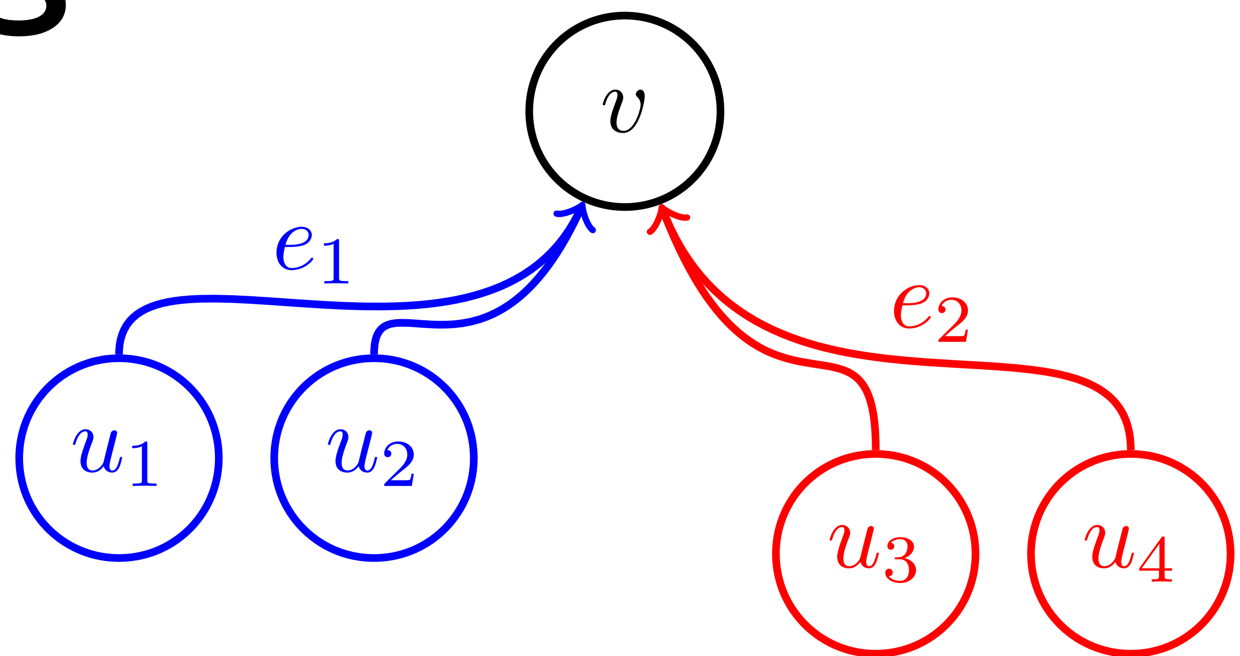
$$\overbrace{\mathrm{NP}_{2,4} : a \quad \mathrm{PP}_{4,6} : b}^{\mathrm{NP}_{2,6} : w_2 \otimes a \otimes b}$$

$$\frac{\mathrm{VBD}_{1,2} : c \ \ \mathrm{NP}_{2,6} : d}{\mathrm{VP}_{1,6} : w_1 \otimes c \otimes d} : w_1$$

$$\frac{\mathrm{NP}_{2,4} : a \ \ \mathrm{PP}_{4,6} : b}{\mathrm{NP}_{2,6} : w_2 \otimes a \otimes b} : w_2$$

- Associate weights as in WFST
- $\otimes$: extension (multiplicative), $\oplus$: summary (additive)

Weights and Semirings



$$
\begin{aligned}
d(v) \quad &= \quad (w(e_1, u_1, u_2) \otimes d(u_1) \otimes d(u_2)) \\
&\quad \oplus (w(e_2, u_3, u_4) \otimes d(u_3) \otimes d(u_4))
\end{aligned}
$$

- The weight of a hyperedge is dependent on antecedents (non-monotonic)
- The weight of a derivation is the product of hyperedge weights
- The weight of a vertex is the summary of (sub-)derivation weights

Semirings

$$\mathbf{K} = \langle K, \oplus, \otimes, \mathbf{0}, \mathbf{1} \rangle$$

semiring	K	$\oplus$	$\otimes$	0	1
Viterbi	[0,1]	max	×	0	1
Real	R	+	x	0	1
Log	R	logsumexp	+	$+\infty$	0
Tropical	R	min	+	$+\infty$	0
Expectation	$<P,R>$	$<p_1 \oplus p_2, r_1 \oplus r_2>$	$<p_1 \otimes p_2, p_1 \otimes r_2 \oplus p_2 \otimes r_1>$	$<0,0>$	$<1,0>$

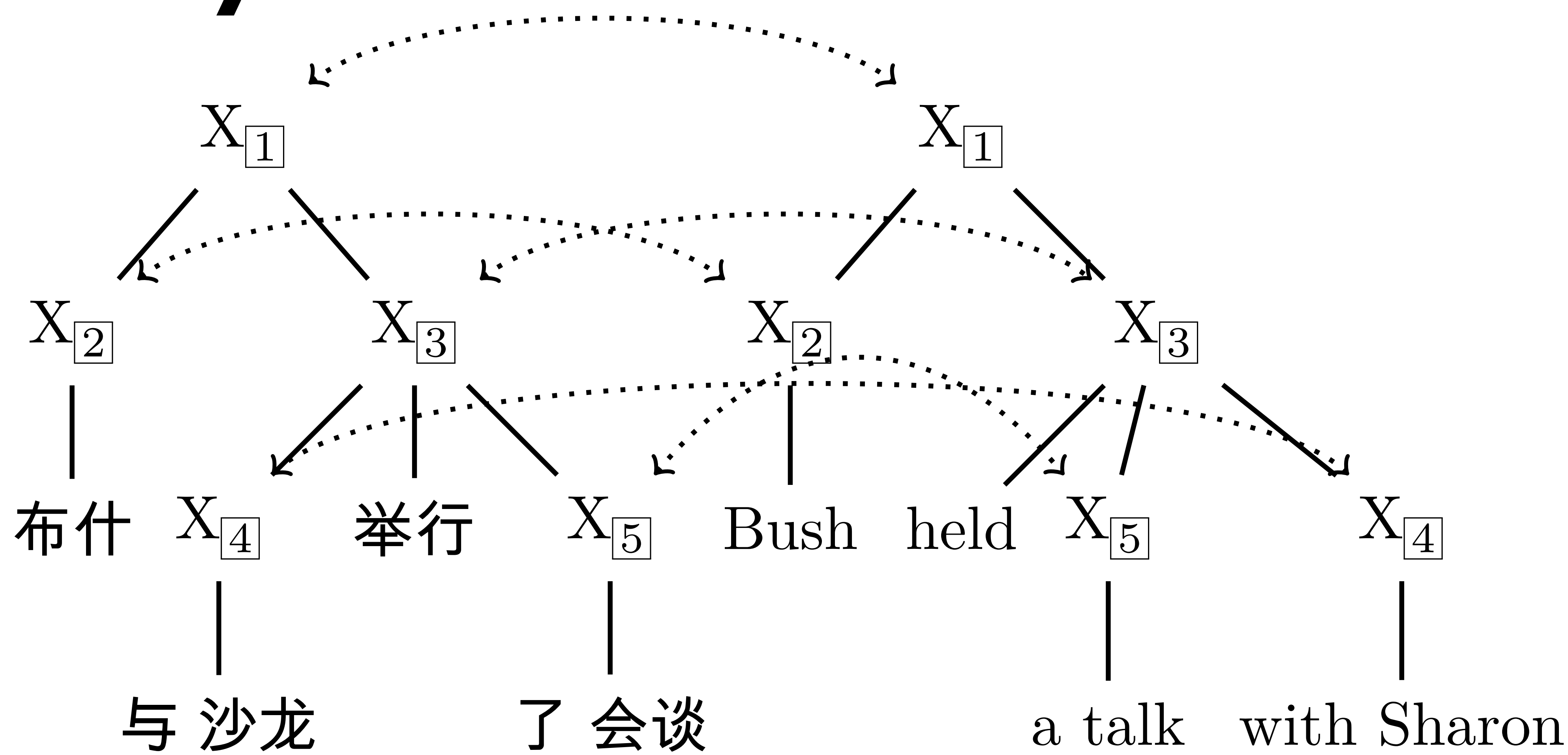
Conclusion

- Review important concepts from “parsing”
  - CFG, parsing, hypergraph, deductive system, weights, semirings

Tree-based MT

- Backgrounds
  - CFG, parsing, hypergraph, deductive system semirings
- Tree-based SMT
  - Synchronous-CFG
  - String-to-Tree, Tree-to-String

Synchronous-CFG



(Chiang, 2007)

$$
\begin{aligned}
\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} \frac{\exp\left(\mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}, D, \mathbf{f})\right)}{\sum_{\mathbf{e}', D'} \exp\left(\mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}', D', \mathbf{f})\right)} \\
&= \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}, D, \mathbf{f})
\end{aligned}
$$

- D: a single derivation constructed by intersecting SCFG with input string

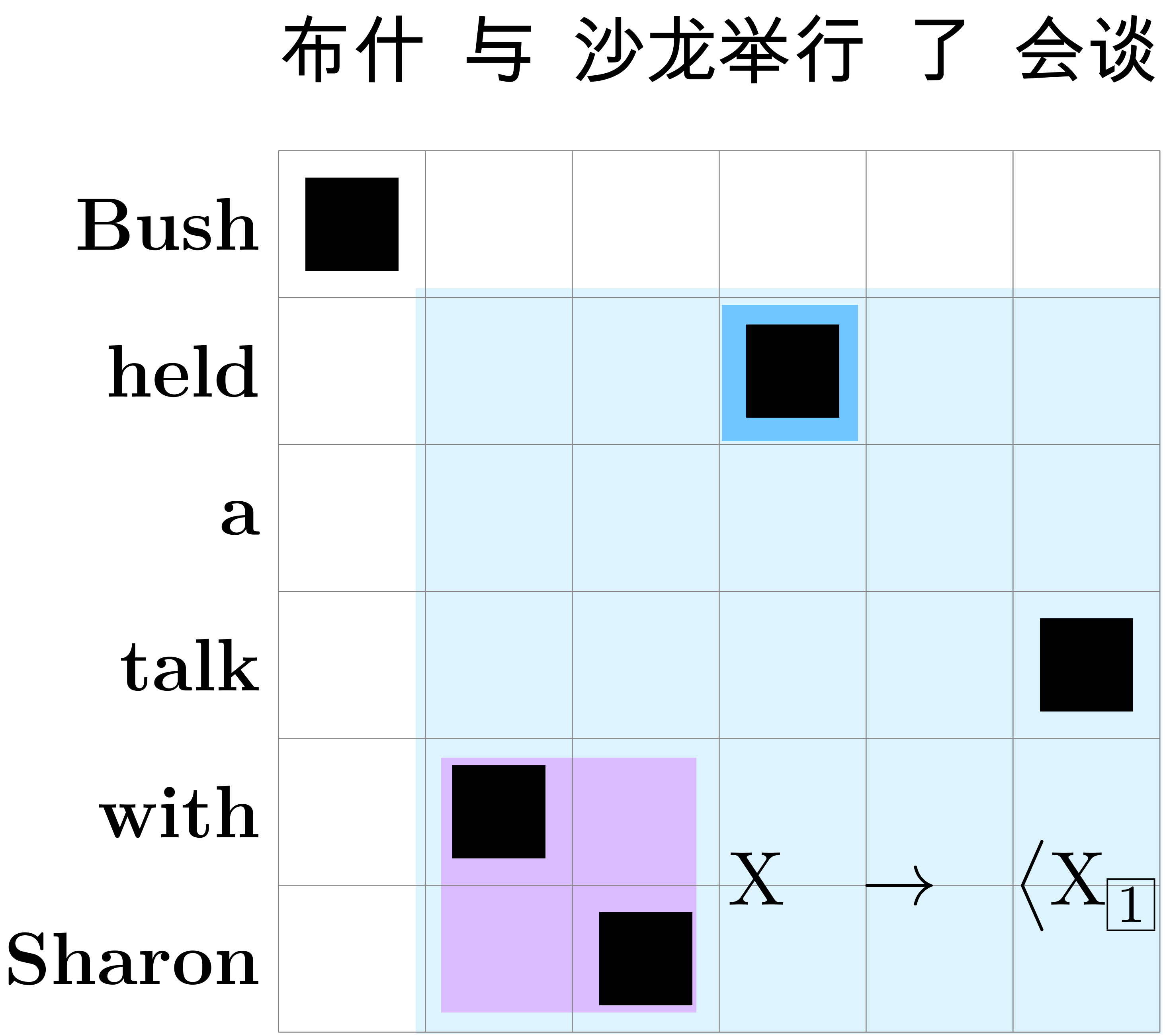
Synchronous-CFG: Model

$$
\begin{aligned}
S &\rightarrow \langle S_{\boxed{1}}\ X_{\boxed{2}}, S_{\boxed{1}}\ X_{\boxed{2}} \rangle \\
S &\rightarrow \langle X_{\boxed{1}}, X_{\boxed{1}} \rangle \\
X &\rightarrow \langle X_{\boxed{1}}\ \text{举行}\ X_{\boxed{2}}, \text{hold}\ X_{\boxed{2}}\ X_{\boxed{1}} \rangle \\
X &\rightarrow \langle \text{与 沙龙}, \text{with Sharon} \rangle
\end{aligned}
$$

$$
\begin{aligned}
VP &\rightarrow \langle VBD_{\boxed{1}}\ NP_{\boxed{2}},\ NP_{\boxed{2}}\ VBD_{\boxed{1}} \rangle \\
NP &\rightarrow \langle NP_{\boxed{1}}\ PP_{\boxed{2}},\ NP_{\boxed{1}}\ PP_{\boxed{2}} \rangle \\
VP &\rightarrow \langle VBD_{\boxed{1}}\ NP_{\boxed{2}}\ PP_{\boxed{3}},\ NP_{\boxed{2}}\ PP_{\boxed{3}}\ VBD_{\boxed{1}} \rangle
\end{aligned}
$$

- We use two categories, S and X (Chiang, 2007)
- Or, borrow linguistic categories from syntactic parse (Zollman and Venugopal, 2006)

Rule Extraction



$\langle$held a talk with Sharon, 与 沙龙 举行 了 会谈$\rangle$

$\langle$with Sharon, 与 沙龙$\rangle$

$\langle$held, 举行$\rangle$

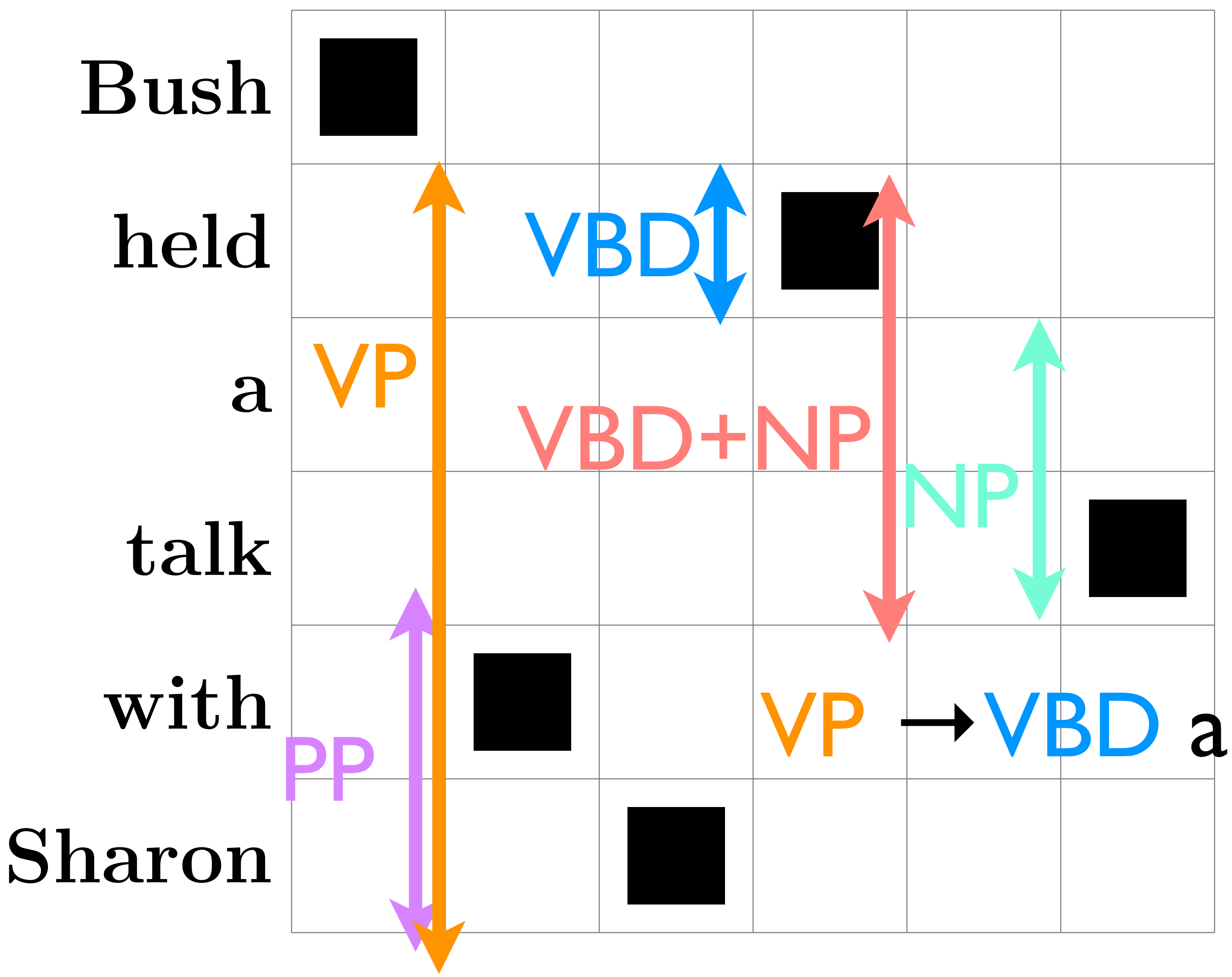
$\mathrm{X} \rightarrow \langle \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{2}}$ 了 会谈, $\mathrm{X}_{\boxed{2}}$ a talk $\mathrm{X}_{\boxed{1}} \rangle$

(Example from Huang and Chiang, 2007)

- As in phrase-based models, extract phrases then, use sub-phrases as non-terminals, aka Hiero (Chiang, 2007)

Syntactic Categories

布什 与 沙龙举行 了 会谈



⟨held a talk with Sharon, 与 沙龙 举行 了 会谈⟩

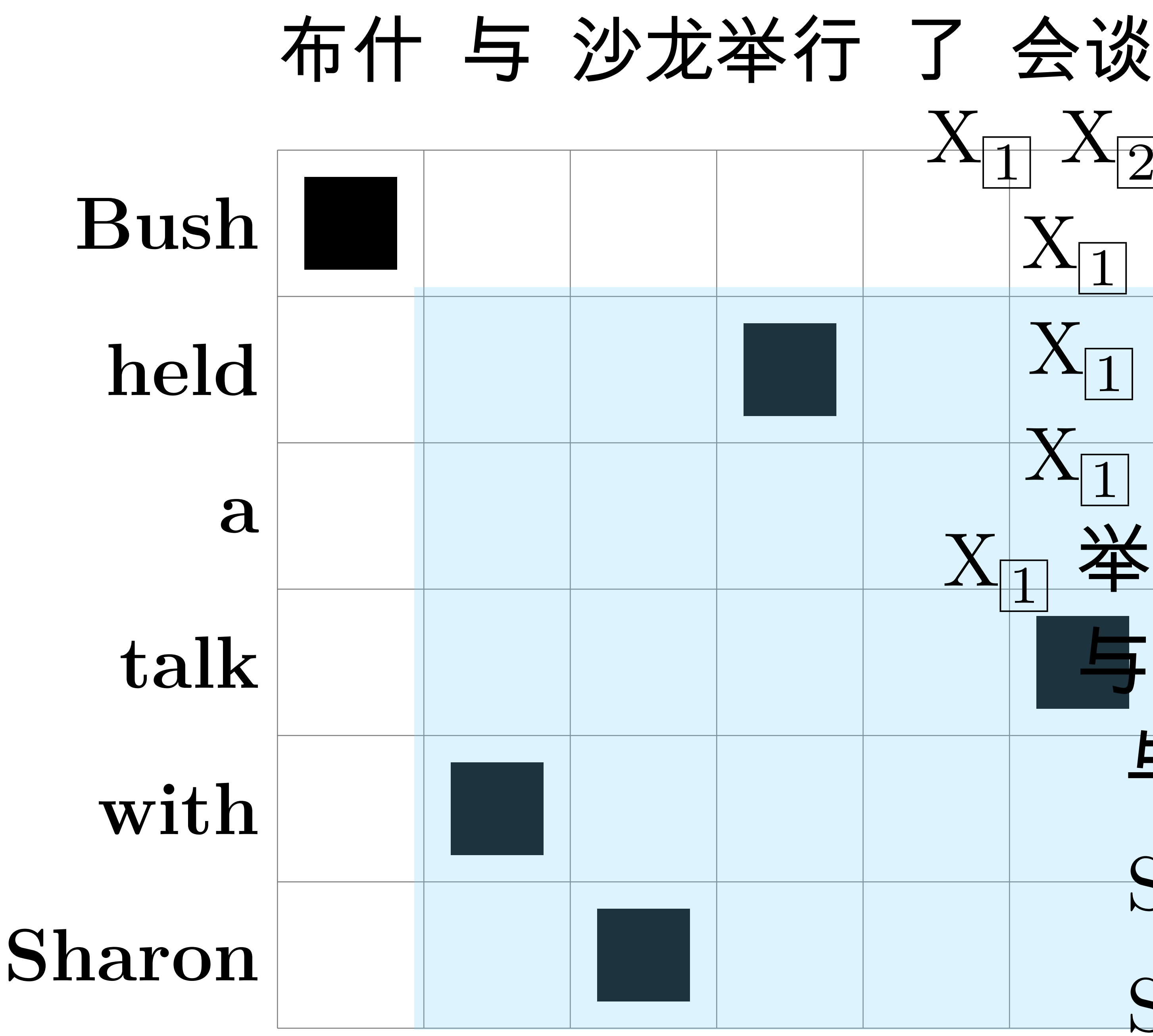
⟨with Sharon, 与 沙龙⟩

⟨held, 举行⟩

VP → VBD a talk PP, PP VBD 了 会谈

- Borrow syntactic categories either from source/target side, aka SAMT(Zollman and Venugopal, 2006)

Exhaustive Extraction



$X_{\boxed{1}}\ X_{\boxed{2}}$ 了 会谈	$X_{\boxed{2}}$ a talk $X_{\boxed{1}}$
$X_{\boxed{1}}\ X_{\boxed{2}}$ 会谈	$X_{\boxed{2}}$ a talk $X_{\boxed{1}}$
$X_{\boxed{1}}\ X_{\boxed{2}}$ 会谈	$X_{\boxed{2}}$ talk $X_{\boxed{1}}$
$X_{\boxed{1}}$ 举行 $X_{\boxed{2}}$	held $X_{\boxed{2}}\ X_{\boxed{1}}$
$X_{\boxed{1}}$ 举行 了 $X_{\boxed{2}}$	held a $X_{\boxed{2}}\ X_{\boxed{1}}$
与 沙龙 $X_{\boxed{1}}$	$X_{\boxed{1}}$ with Sharon
与 $X_{\boxed{1}}\ X_{\boxed{2}}$	$X_{\boxed{2}}$ with $X_{\boxed{1}}$

$$S \rightarrow \langle S_{\boxed{1}}\ X_{\boxed{2}}, S_{\boxed{1}}\ X_{\boxed{2}} \rangle$$

$$S \rightarrow \langle X_{\boxed{1}}, X_{\boxed{1}} \rangle$$

- Exhaustively extract rules as in phrase-based MT
- + glue rules

Features from Rules

$$\begin{aligned}
\log p_r(\bar{\alpha}|\bar{\beta}) &= \log \frac{\text{count}(\bar{\beta}, \bar{\alpha})}{\sum_{\bar{\alpha'}} \text{count}(\bar{\beta}, \bar{\alpha'})} \\
\log p_r(\bar{\beta}|\bar{\alpha}) &= \log \frac{\text{count}(\bar{\beta}, \bar{\alpha})}{\sum_{\bar{\beta'}} \text{count}(\bar{\beta'}, \bar{\alpha})}
\end{aligned}$$

- Collect all the rules (α, β) from the data:
  - α = source side string, β = target side string
- Maximum likelihood estimates by relative frequencies
- Employ scores in two directions

Example: Grammar

[x] ||| [x,1] 给 我 [x,2] 。 ||| [x,1] 'd like some [x,2] . ||| -1.4853690183 -10.1479974813 0.0 -3.7423198799
[x] ||| [x,1] 给 我 [x,2] 。 ||| [x,1] 'll have [x,2] . ||| -1.6548831288 -7.0498958791 0.0 -4.2061890092
[x] ||| [x,1] 给 我 [x,2] 。 ||| [x,1] show me [x,2] . ||| -1.6145807498 -5.0981314097 0.0 -1.7266717936
[x] ||| [x,1] 给 我 [x,2] 。 ||| [x,2] , [x,1] . ||| -0.9584345257 -1.4907203037 -1.0686157177 -3.958028322

[x] ||| 我 不 [x,1] 说 过 [x,2] 了 ||| i said i [x,2] n't [x,1] ||| 0.0 -5.3472963389 0.0 -8.2260811313
[x] ||| 我 不 [x,1] 说 过 [x,2] 了 吗 ||| i said i [x,2] n't [x,1] it ||| 0.0 -8.7156056227 0.0 -11.0837696086
[x] ||| 我 不 [x,1] 说 过 不要 [x,2] 吗 ||| i said [x,2] do n't [x,1] it ||| 0.0 -5.7738835319 0.0 -9.4922428063
[x] ||| 我 不 [x,1] 说 过 不要 了 [x,2] ||| i said i do n't [x,1] [x,2] ||| 0.0 -5.3472963389 0.0 -10.4427474019
[x] ||| 我 不 [x,1] 说 过 不要 了 吗 ？ ||| i said i do n't [x,1] it . ||| 0.0 -11.9166721472 0.0 -17.1218716285
[x] ||| 我 不 是 [x,1] 不要 了 [x,2] ||| i [x,1] i do n't need [x,2] ||| 0.0 -8.3177521678 0.0 -9.4746990247
[x] ||| 我 不 是 [x,1] 不要 了 吗 ？ ||| i [x,1] i do n't need it . ||| 0.0 -14.8871279762 0.0 -16.1538232513
[x] ||| 我 不 是 [x,1] 吗 [x,2] ||| i [x,1] n't need it [x,2] ||| 0.0 -8.7443393608 0.0 -6.3075281585

[x] ||| 可以 ||| can ||| -1.1143606456 -0.5135029225 -0.716677678 -1.1056222421
[x] ||| 可以 ||| can i ||| -1.1143606456 -1.9504120512 -0.4212134651 -1.1056222421
[x] ||| 可以 ||| may ||| -1.7609878106 -1.4225938319 0.0 0.0
[x] ||| 可以 ||| may i ||| -1.7609878106 -2.8595029606 0.0 0.0

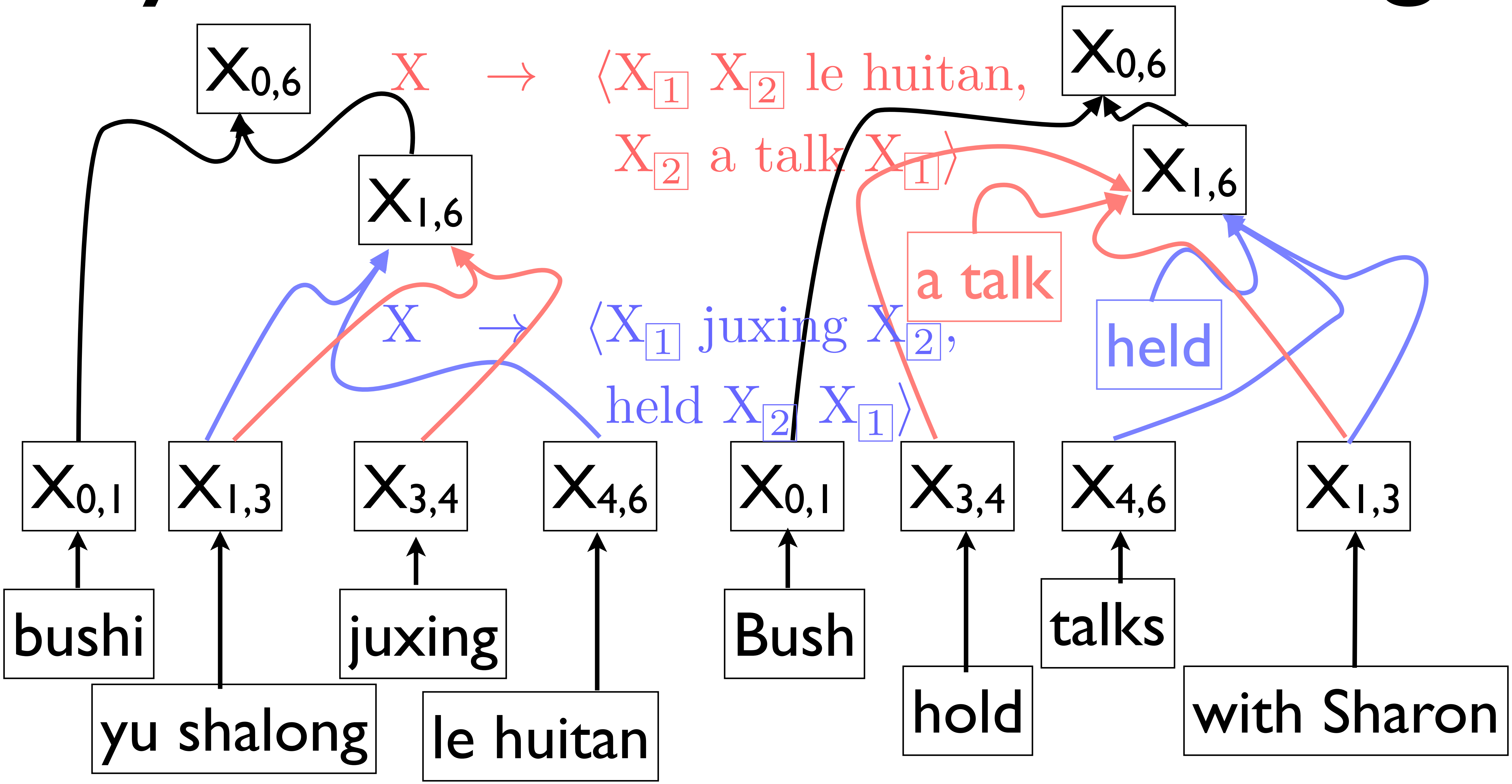
Remarks on Rules

- Too many rules extracted (Chiang, 2007):
  - at most two non-terminal symbols
  - at least one terminal between non-terminals in the source side
  - Span at most 15 words for "holes"
- Fractional counts (Chiang, 2007):
  - Each phrases counted in phrase-based MT
  - Fractional counts for rules sharing the same source/target span

Other Features

- Lexical weights as used in phrase-based MT
- ngram language model(s)
- word count: bias for ngram language model(s)
- rule count: shorter or longer phrases
- glue-rule counts: bias for monotonic glue rules

Synchronous-CFG: Parsing



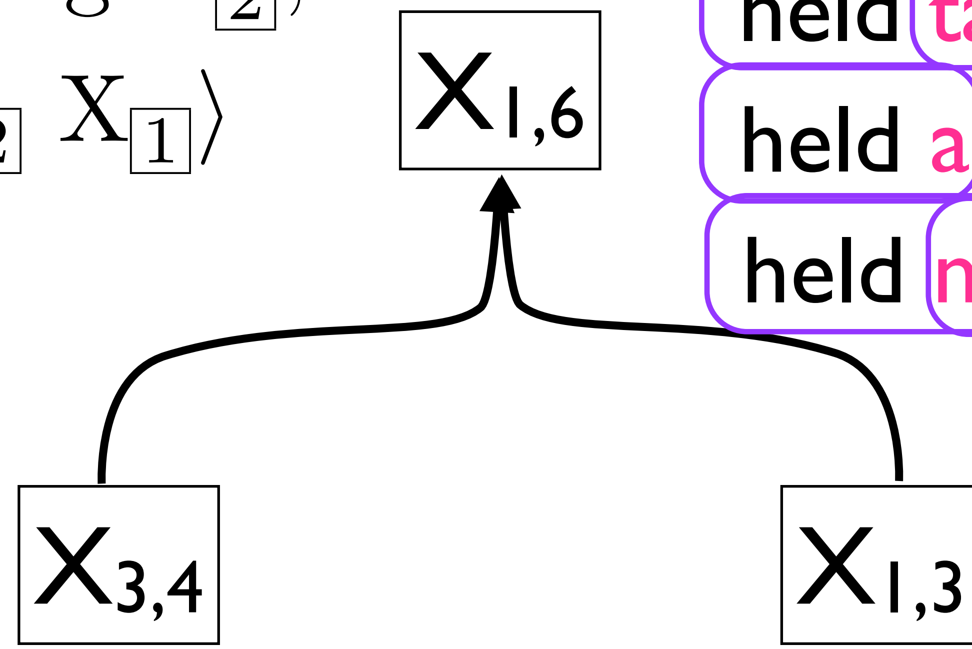
- Parse input sentence using the source side, and construct a translation forest by target side

Synchronous-CFG: Parsing

- Translation by SCFG = monolingual parsing using the source side grammar
- Construct forest by the projected target side
- From forests, compute the best derivation (Huang and Chiang, 2005)
- Complexity: $O(n^3)$ as in monolingual CKY

Non-Local Features

$$X \rightarrow \langle X_{\boxed{1}} \text{ juxing } X_{\boxed{2}},\ \text{held } X_{\boxed{2}}\ X_{\boxed{1}} \rangle$$



held a talk with Sharon

held talks with Sharon

held a talk and Sharon

held meeting Sharon with

Update boundary words only

p(talk | a) a talk

talks

meeting

meetings

with Sharon p(Sharon | with)

and Sharon p(Sharon | and)

Sharon with p(with | Sharon)

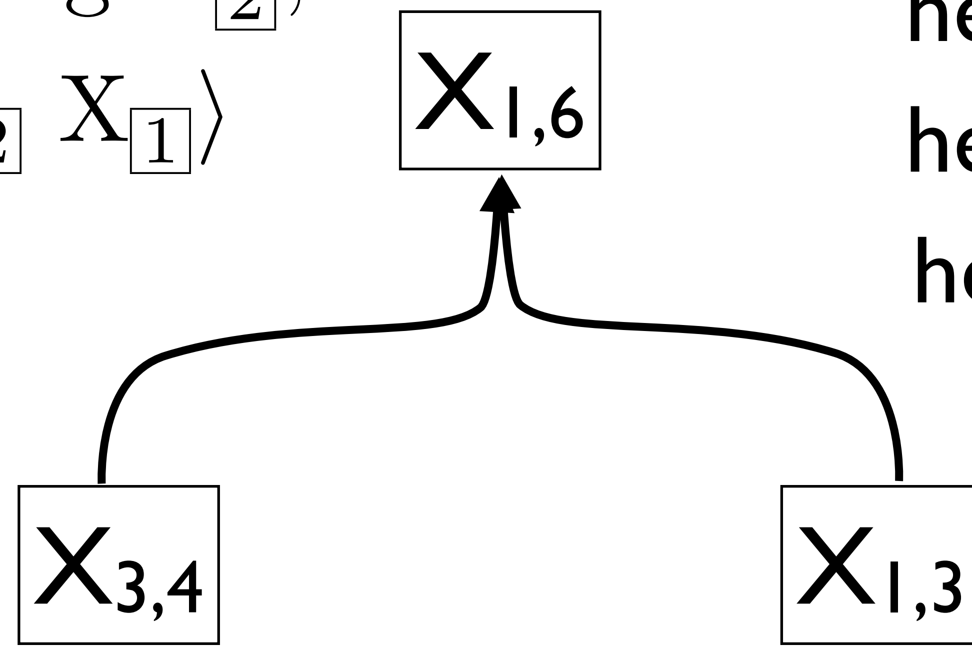
Sharon and p(and | Sharon)

- non-local features which requires out-of-span context, i.e. bigram LM

Bigram Features

$$X \quad \rightarrow \quad \langle X_{\boxed{1}} \text{ juxing } X_{\boxed{2}}, \; \text{held } X_{\boxed{2}} \; X_{\boxed{1}} \rangle$$

held * Sharon
held * Sharon
held * Sharon
held * with



a * talk
talks
meeting
meetings

with * Sharon
and * Sharon
Sharon * with
Sharon * and

- We keep only bigram states: (Why 2 words?)

Language Model Updates

- Each hypothesis keeps two contexts:
  - Prefix: ngrams to be scored with antecedents
  - Suffix: contexts for future ngrams (i.e. Phrase-based MT)
- Complexity: $O(n^3V^{2(m-1)})$
- Very inefficient: we need to explicitly enumerate all the hypotheses in antecedents

Forest Rescoring

- Translation by SCFG = monolingual parsing using the source side grammar
- Construct forest by the projected target side + Rescore with non-local features
- From forests, compute the best derivation (Huang and Chiang, 2005)
- ~~Complexity: $O(n^3)$ as in monolingual CKY~~

Cube Pruning

$$X \rightarrow \langle X_{\boxed{1}} \text{ juxing } X_{\boxed{2}}, \text{ held } X_{\boxed{2}}\ X_{\boxed{1}} \rangle$$

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	2.5	2.7	3.6	4.2
talks	1.3	2.8	3.0	3.9	4.5
meeting	2.2	3.7	3.9	4.8	5.4
meetings	2.6	4.1	4.3	5.2	5.8

- For each hyperedge, create a “cube” representing combinations of antecedents (Huang and Chiang, 2007)

Cube Pruning

$$X \rightarrow \langle X_{\boxed{1}} \text{ juxing } X_{\boxed{2}},\ \text{held } X_{\boxed{2}}\ X_{\boxed{1}} \rangle$$

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	2.5 +0.5	2.7 +1.0	3.6 +1.5	4.2 +1.5
talks	1.3	2.8 +0.3	3.0 +1.5	3.9 +2.0	4.5 +2.0
meeting	2.2	3.7 +0.5	3.9 +1.0	4.8 +1.5	5.4 +1.5
meetings	2.6	4.1 +0.3	4.3 +1.5	5.2 +2.0	5.8 +2.0

- Bigrams require contexts from antecedents: non-monotonic scoring

Cube Pruning

queue: (0,0)

k-best:

		with * Sharon 1.5	and * Sharon 1.7	Sharon * with 2.6	Sharon * and 3.2
a * talk	1.0	3.0			
talks	1.3				
meeting	2.2				
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue:

k-best: (0,0)

		with * Sharon 1.5	and * Sharon 1.7	Sharon * with 2.6	Sharon * and 3.2
a * talk	1.0	3.0			
talks	1.3				
meeting	2.2				
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (0,1)(1,0)

k-best: (0,0)

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	3.0	3.7		
talks	1.3	3.1			
meeting	2.2				
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (1,0)

k-best: (0,0)(0,1)

		with * Sharon 1.5	and * Sharon 1.7	Sharon * with 2.6	Sharon * and 3.2
a * talk	1.0	3.0	3.7		
talks	1.3	3.1			
meeting	2.2				
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (1,0) (0,2) (1,1)

k-best: (0,0) (0,1)

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	3.0	3.7		
talks	1.3	3.1	4.5		
meeting	2.2	4.2			
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (0,2) (1,1)

k-best: (0,0) (0,1) (1,0)

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	3.0	3.7		
talks	1.3	3.1	4.5		
meeting	2.2	4.2			
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (0,2) (1,1) (3,0)

k-best: (0,0) (0,1) (1,0)

		with * Sharon 1.5	and * Sharon 1.7	Sharon * with 2.6	Sharon * and 3.2
a * talk	1.0	3.0	3.7	5.1	
talks	1.3	3.1	4.5		
meeting	2.2	4.2			
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

queue: (1,1) (3,0)

k-best: (0,0) (0,1) (1,0) (0,2)

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	3.0	3.7	5.1	
talks	1.3	3.1	4.5		
meeting	2.2	4.2			
meetings	2.6				

- Starting from the upper-left corner, enumerate antecedent combinations

Cube Pruning

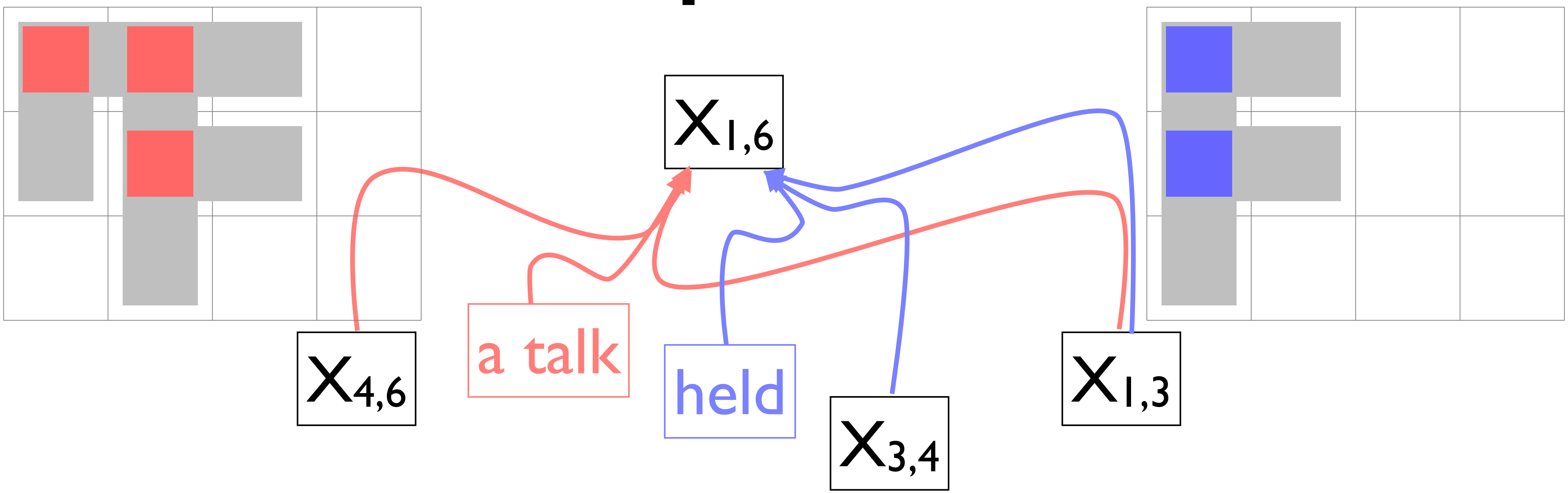
queue: (0,4) (1,1)(1,2) (3,0)

k-best: (0,0)(0,1) (1,0) (0,2)

		with * Sharon	and * Sharon	Sharon * with	Sharon * and
		1.5	1.7	2.6	3.2
a * talk	1.0	3.0	3.7	5.1	
talks	1.3	3.1	4.5		
meeting	2.2	4.2	4.9		
meetings	2.6	4.4			

- Starting from the upper-left corner, enumerate antecedent combinations

Multiple Rules



- Multiple rules sharing the same span are queued
  - Each rule is associated with a cube
  - hypothesis = hyperedge + cube-position

Further Faster Pruning

- Cube Growing (Huang and Chiang, 2007)
  - Top-down pruning combined with heuristic estimates
- Faster Cube Pruning (Gesmundo and Henderson, 2010)
  - Eliminate bookkeeping for inserted hypotheses by determining the ordering of cube enumerations
  - Push minimum hypotheses by looking up ancestors
- Top-down decoding (Watanabe et al., 2006; Huang and Mi, 2010; Yang et al., 2012)

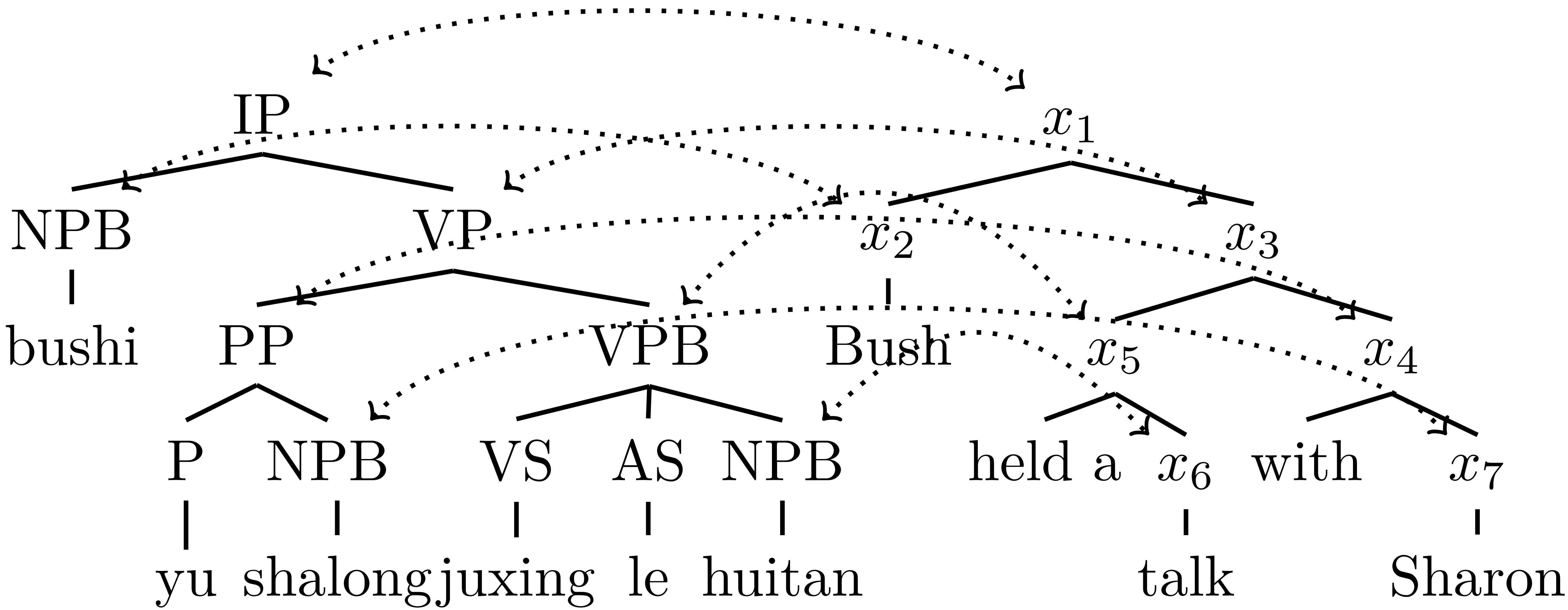
Conclusion

- Synchronous-CFG
  - paired CFG + shared non-terminal symbols
- Training is based on phrase-based MT by treating sub-phrase as a non-terminal
- Decoding: monolingual parsing
  - An efficient antecedent combination via cube-pruning

Tree-based MT

- Backgrounds
  - CFG, parsing, hypergraph, deductive system semirings
- Tree-based SMT
  - Synchronous-CFG
  - String-to-Tree, Tree-to-String

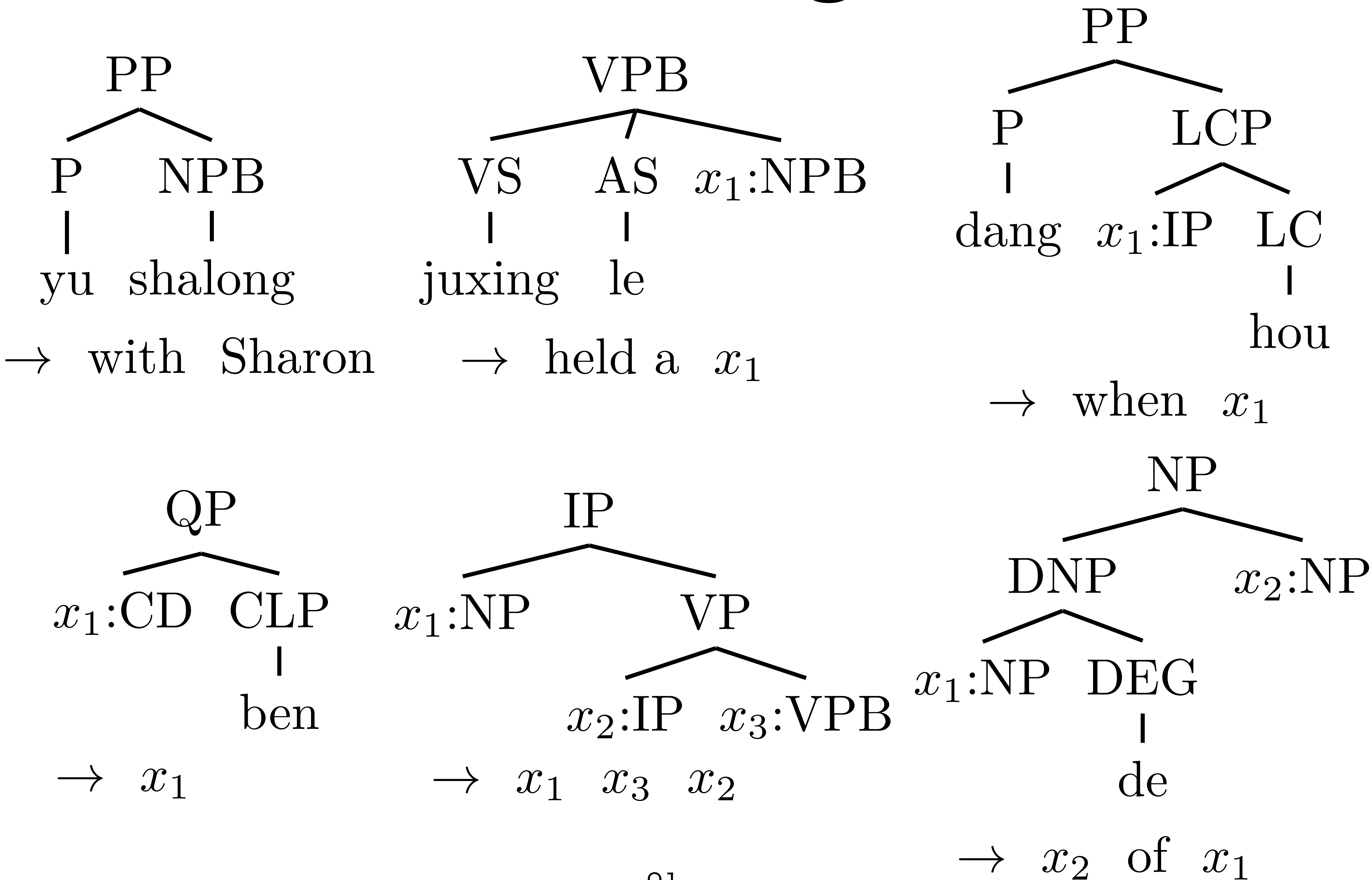
{Tree,String}-to-{Tree,String}



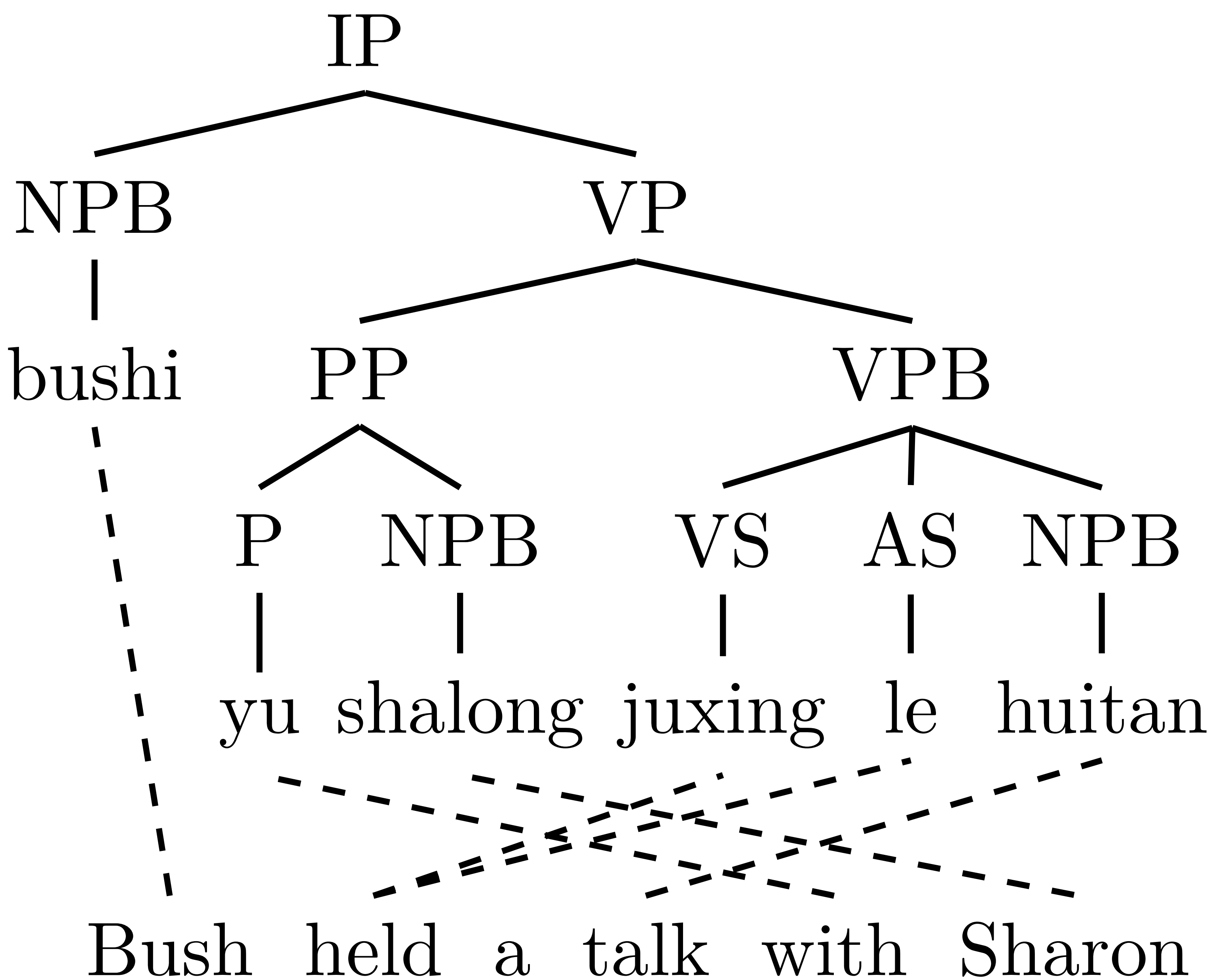
(Galley et al., 2004)

- Each synchronous rule has a subtree structure
- Flat structure + sharing the same non-terminal symbols = synchronous-CFG

Tree-to-String Rules



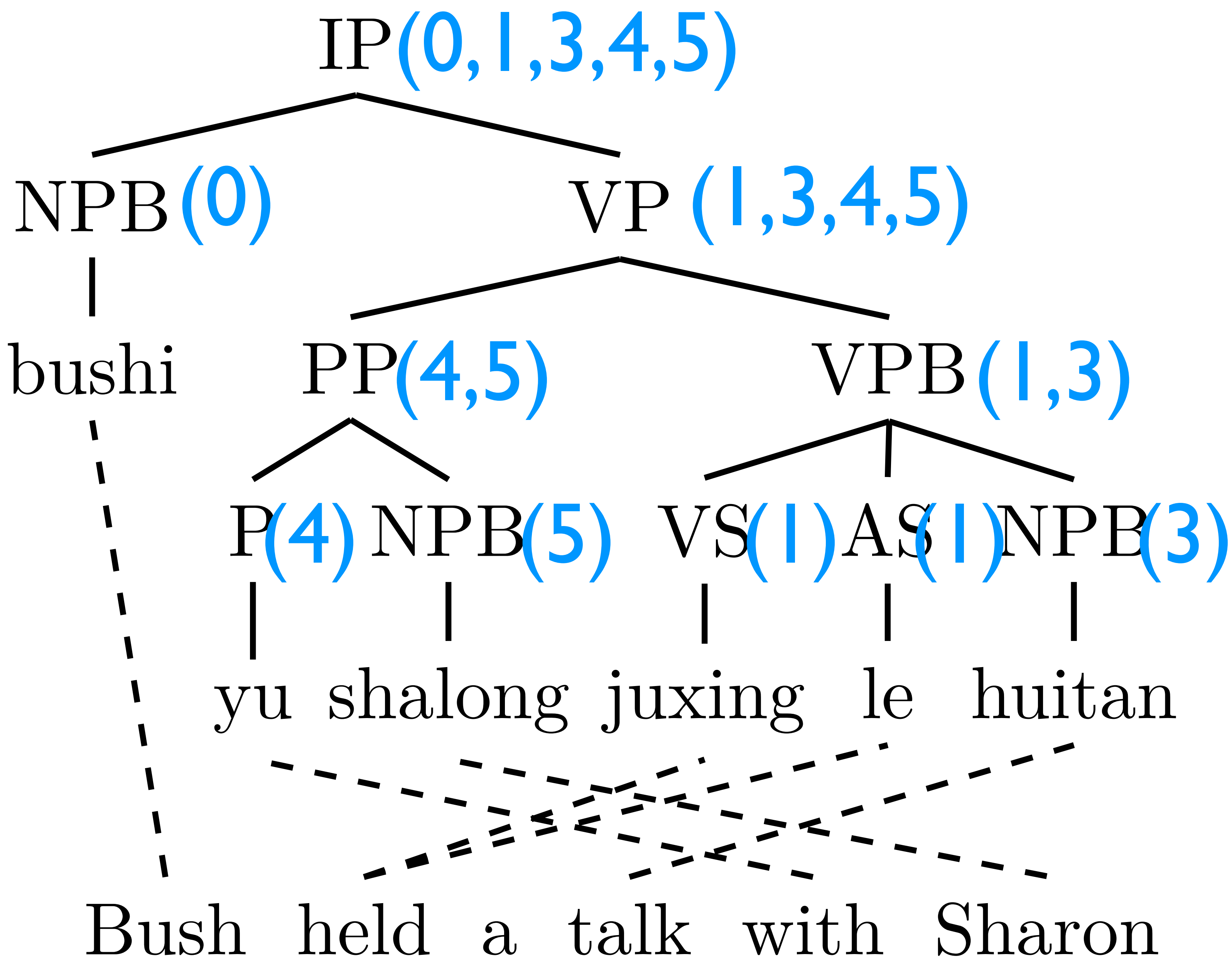
Rule Extraction



(Galley et al., 2004)

- Compute "minimum rules" as in phrase-based MT (or, compute phrasal-match)

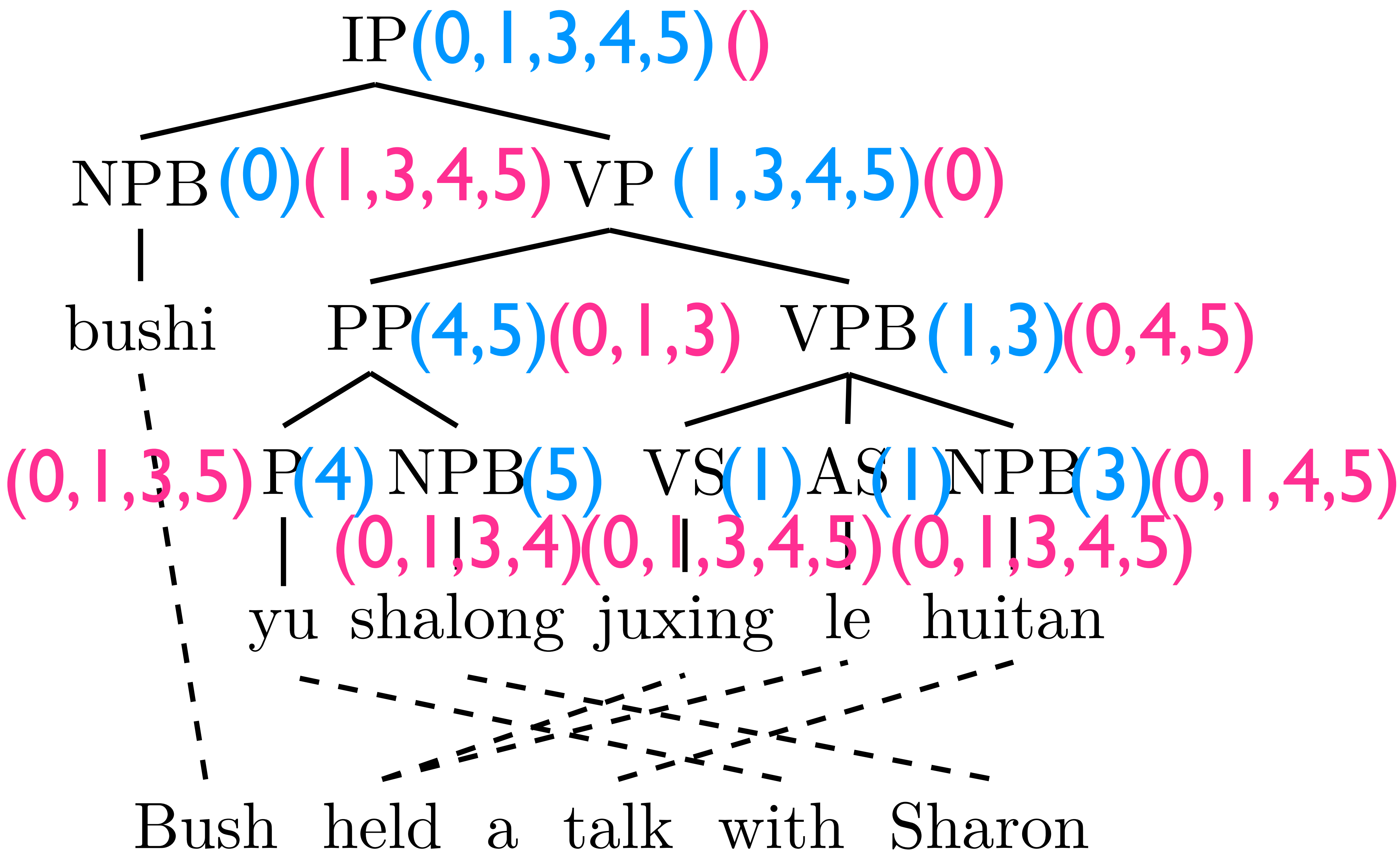
Rule Extraction



(Galley et al., 2004)

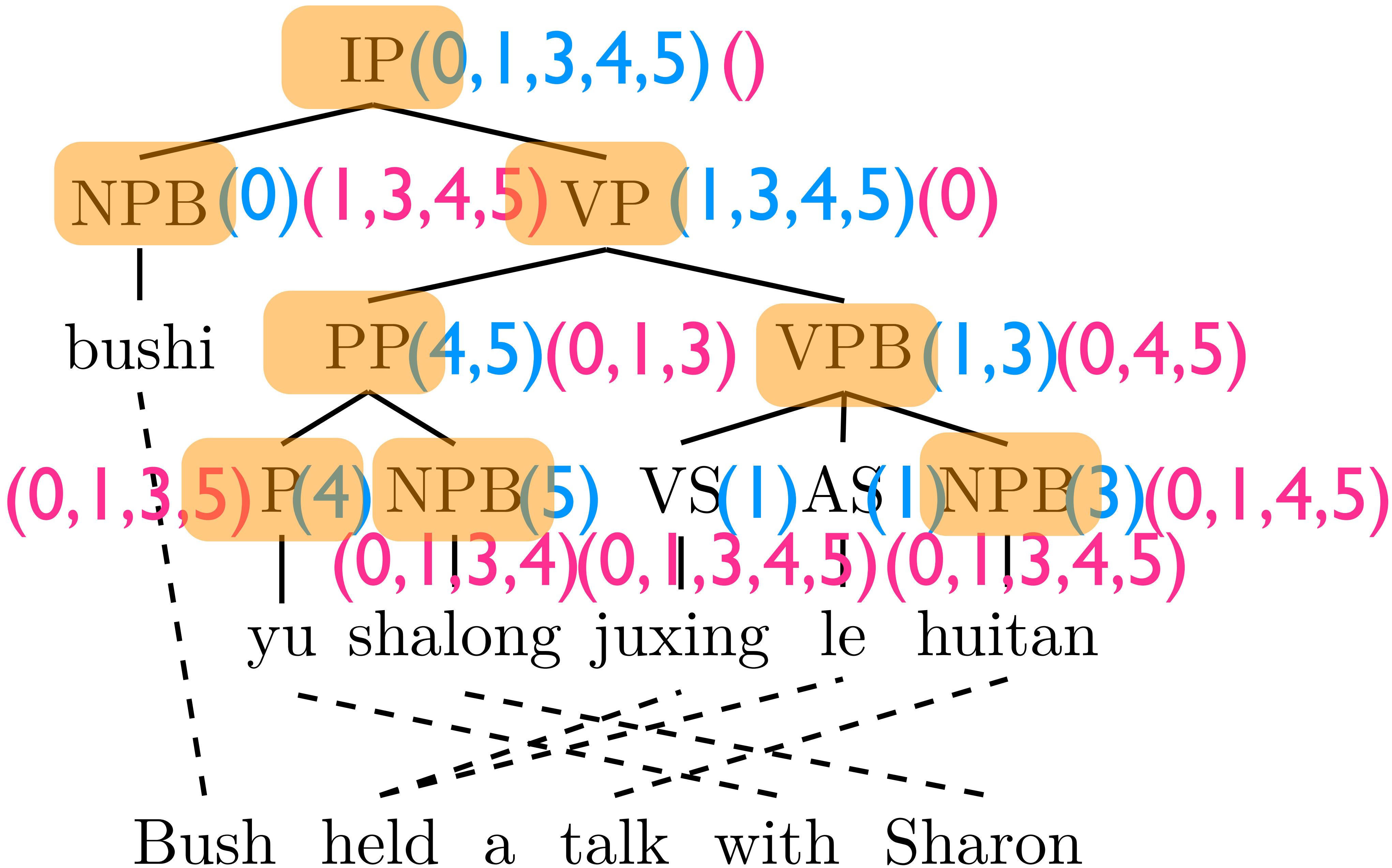
- Compute "spans" by propagating alignment in bottom-up

Rule Extraction



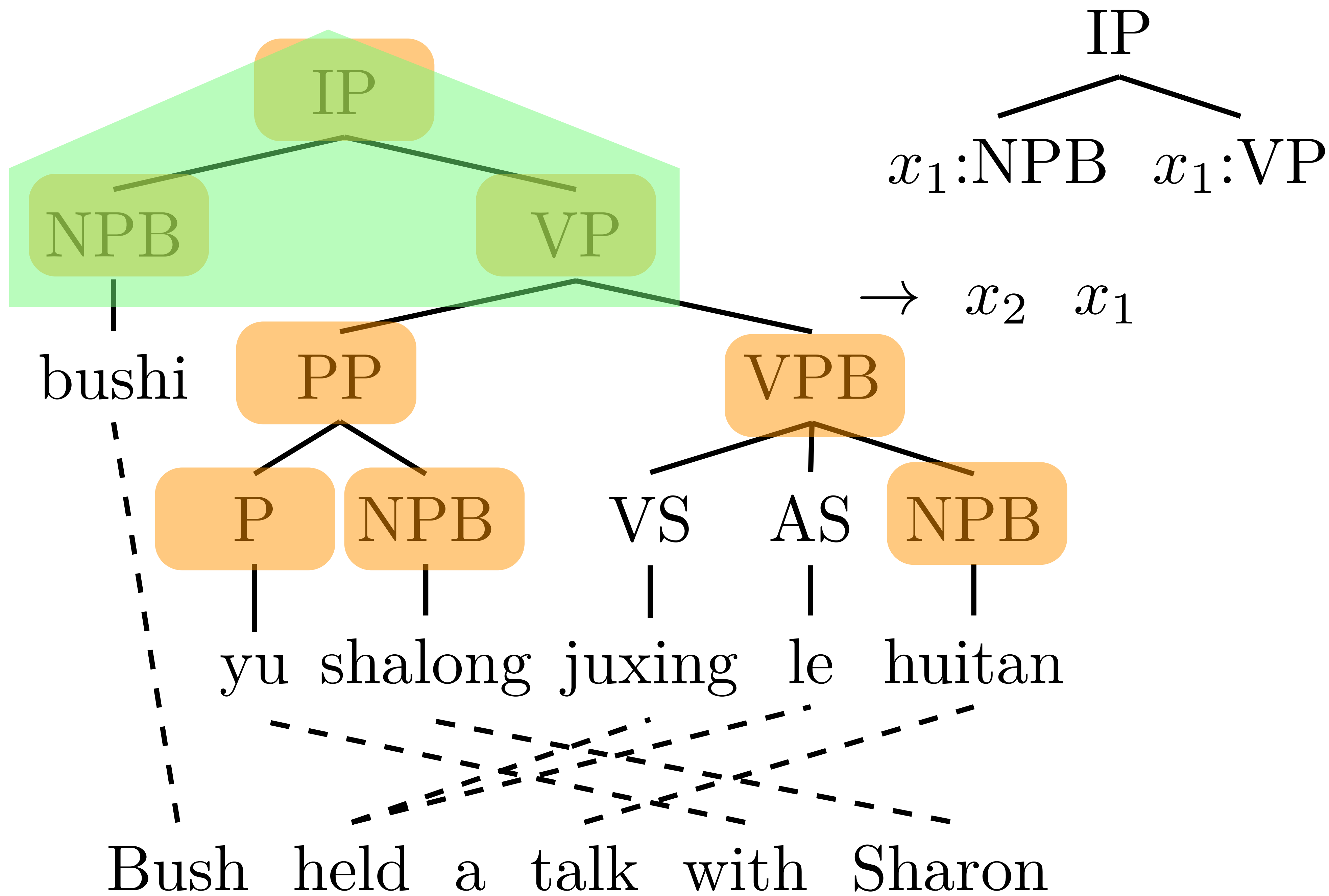
- Compute "complements" in top-down

Rule Extraction



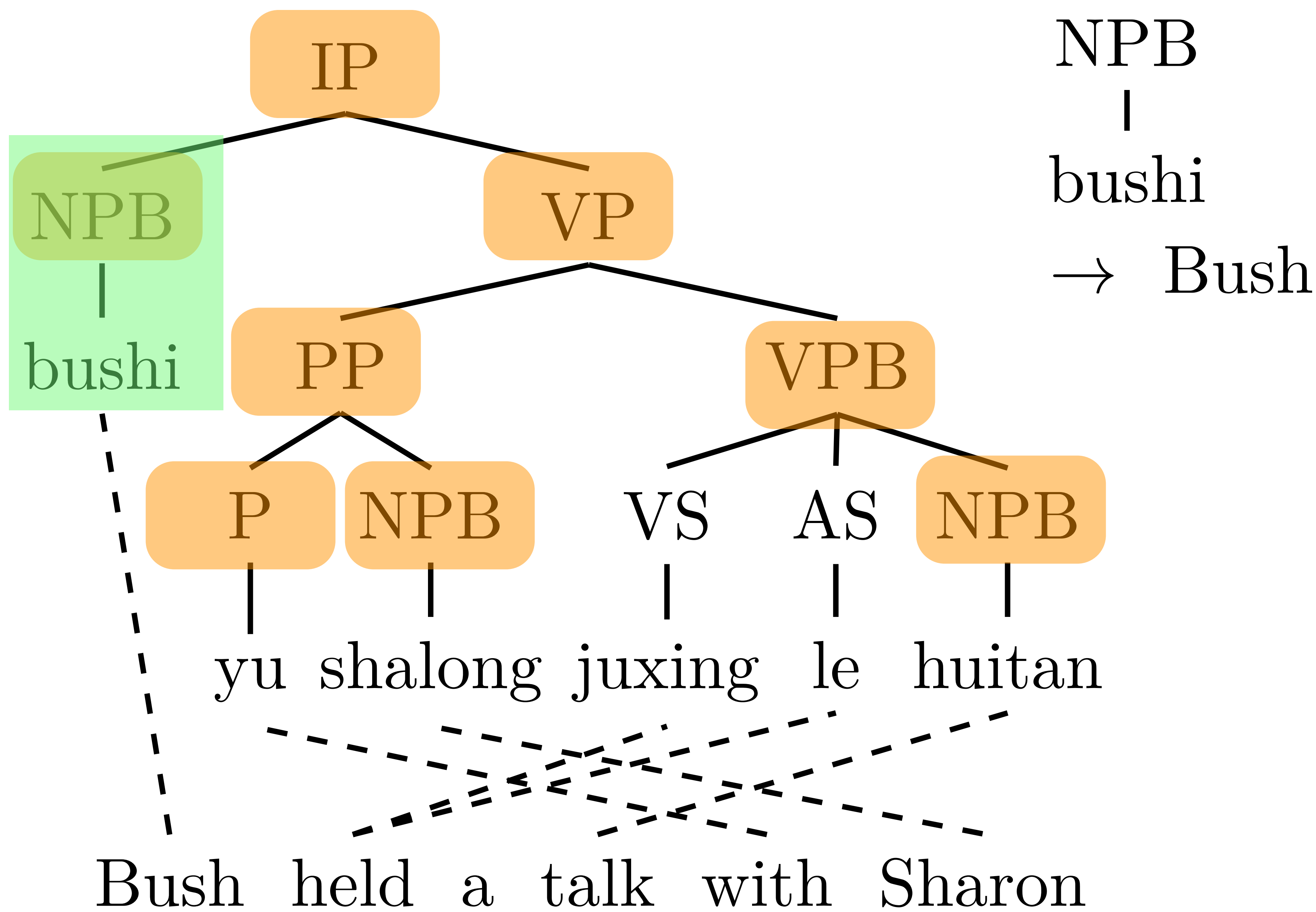
- Compute "frontiers": The nodes in which the intersection of "spans" and "complements" is empty

Rule Extraction



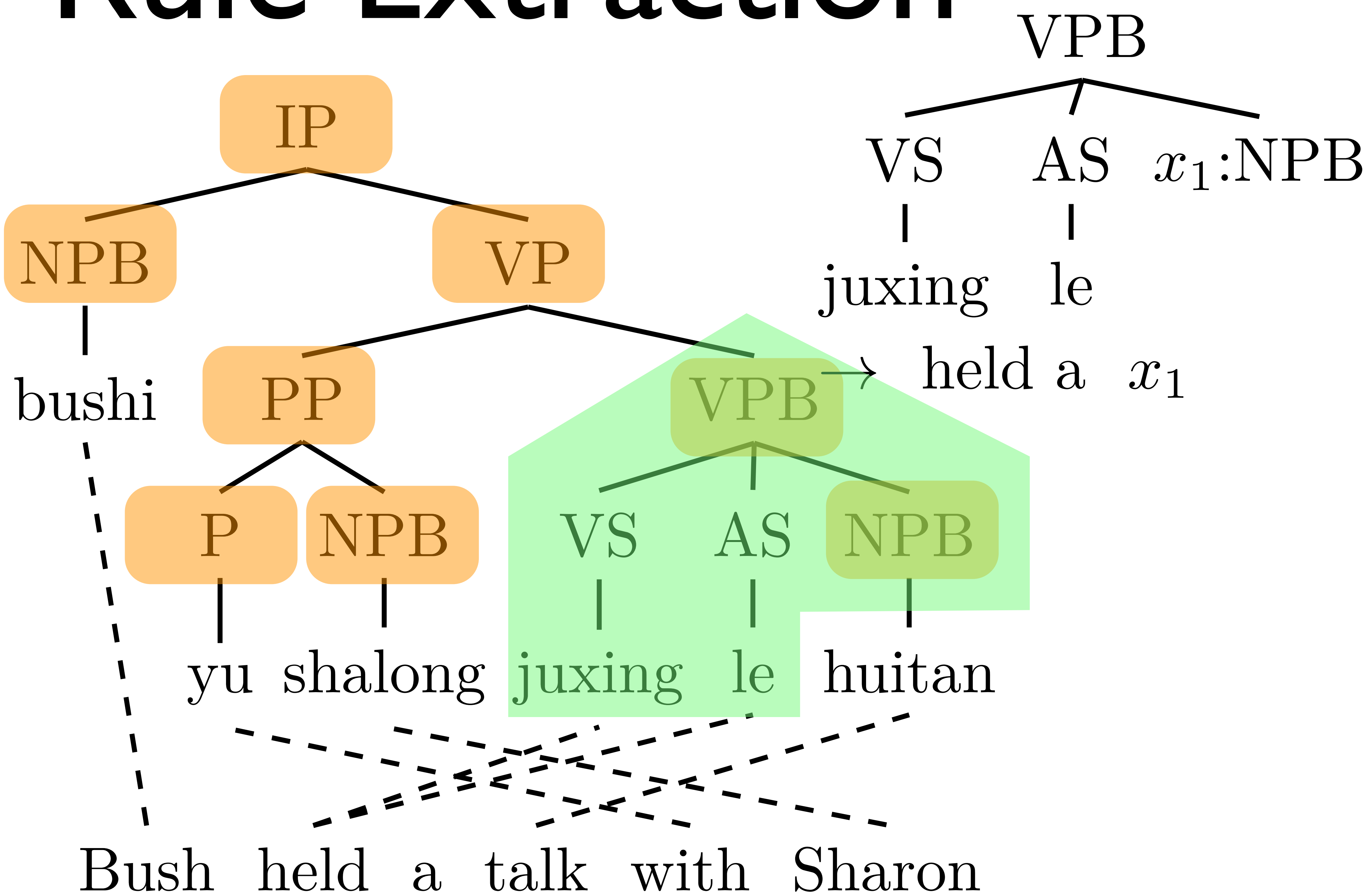
- Extract minimum rules using frontiers

Rule Extraction



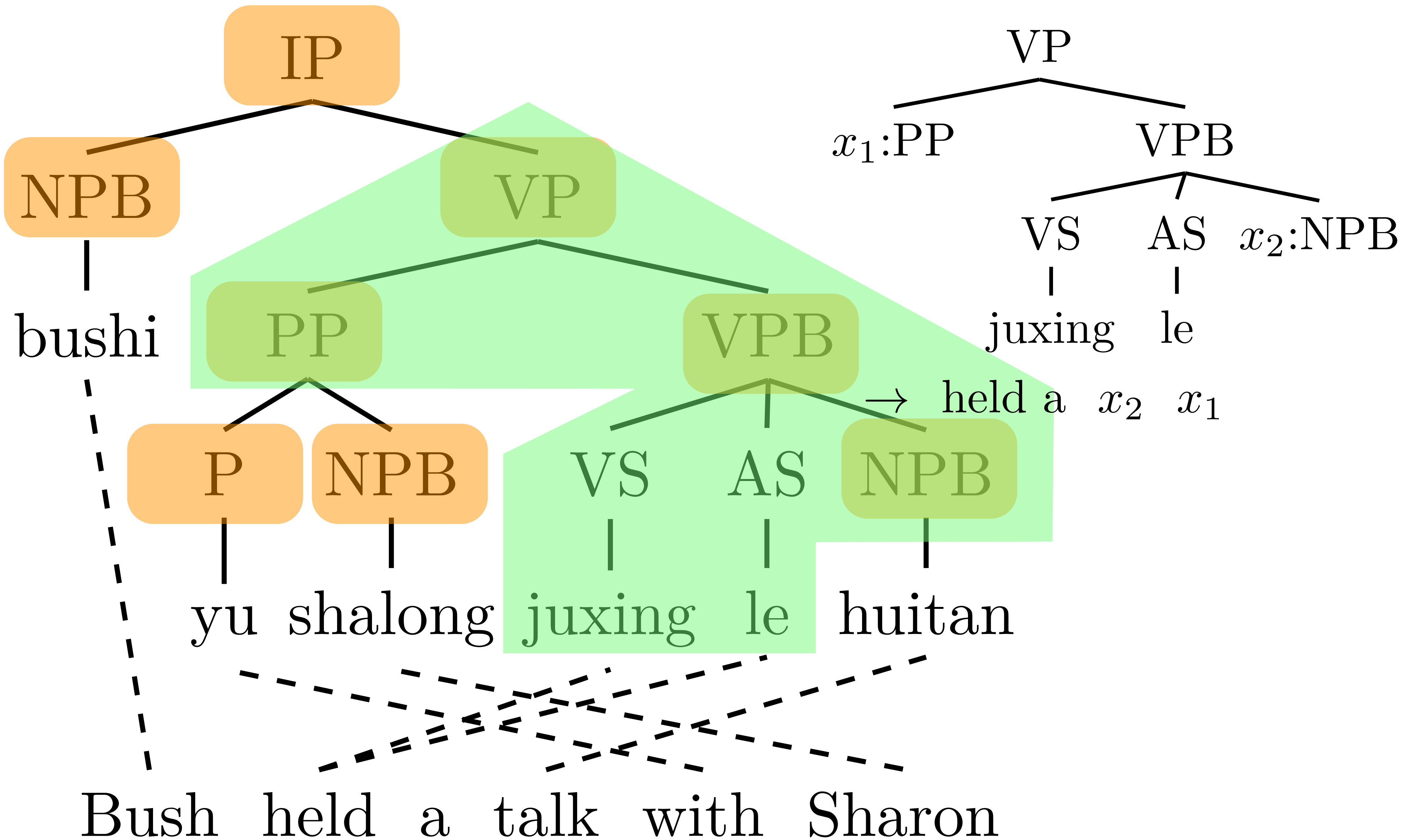
- Extract minimum rules using frontiers

Rule Extraction



- Extract minimum rules using frontiers

Rule Extraction



(Galley et al., 2006)

- Extract "compound rules" by combining minimum rules (i.e. longer phrases)

Example: Grammar

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](小))) ||| [x]([x,1] minimum) ||| 0.0 -6.0493246701 -1.2862109026 -2.0951197555 -8.895629627l -13.5950647026

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](所有))) ||| [x]([x,1] all) ||| 0.0 -0.4491434583 -4.3555426458 -2.8899812337 -9.5422567921 -11.1594765255

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](抱歉))) ||| [x]([x,1] , but) ||| -1.9195928407 -5.5287820969 -2.1972245773 -4.0379367632 -7.6774721878 -13.3468850731

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](抱歉))) ||| [x]([x,1] but) ||| -1.9195928407 -2.6030938064 -2.9348235205 -4.0379367632 -7.6774721878 -12.6054711516

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](抱歉))) ||| [x]([x,1] sorry but) ||| -1.9195928407 -3.3153733014 -2.2655438213 -2.3741149248 -7.6774721878 -13.2426240628

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](抱歉))) ||| [x]([x,1] sorry) ||| -1.9195928407 -0.712279495 -4.3786117196 -1.7804642018 -7.6774721878 -11.0996500752

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](抱歉))) ||| [x](sorry [x,1]) ||| -0.8835009091 -0.712279495 -4.1415974471 -1.7804642018 -7.6774721878 -10.312883511

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](正式))) ||| [x]([x,1] formal) ||| 0.0 -0.4767629775 -1.6959115104 -1.4531858426 -8.5061648604 -12.7763402147

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](现代))) ||| [x]([x,1] modern) ||| 0.0 -0.4035171952 0.0 -0.3294117036 -9.5422567921 -15.4598493069

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](耐用))) ||| [x]([x,1] durable) ||| 0.0 -0.324508026 0.0 -0.5357159117 -9.5422567921 -15.4598493069

[ADJP]([ADVP]([AD,1]) [ADJP]([JJ](重要))) ||| [x]([x,1] important) ||| 0.0 -0.3380531821 -2.8699628863 -0.0405166154 -9.5422567921 -12.6054711516

[NP]([ADJP]([JJ,1]) [NP]([NN](曲线) [NN](球))) ||| [x]([x,1] ball) ||| -0.1953087523 -0.3300015301 -0.1953087523 -2.6137082997 -11.7142174428 -13.7468707155

[NP]([ADJP]([JJ,1]) [NP]([NN](曲线) [NN](球))) ||| [x](ball [x,1]) ||| -1.7292391122 -0.3300015301 0.0 -2.6137082997 -11.7142174428 -15.4598493069

[NP]([ADJP]([JJ,1]) [NP]([NN](机构))) ||| [x]([x,1] branch) ||| 0.0 -1.7874404239 -2.929980896 -4.6254673582 -13.4271960342 -12.5544396716

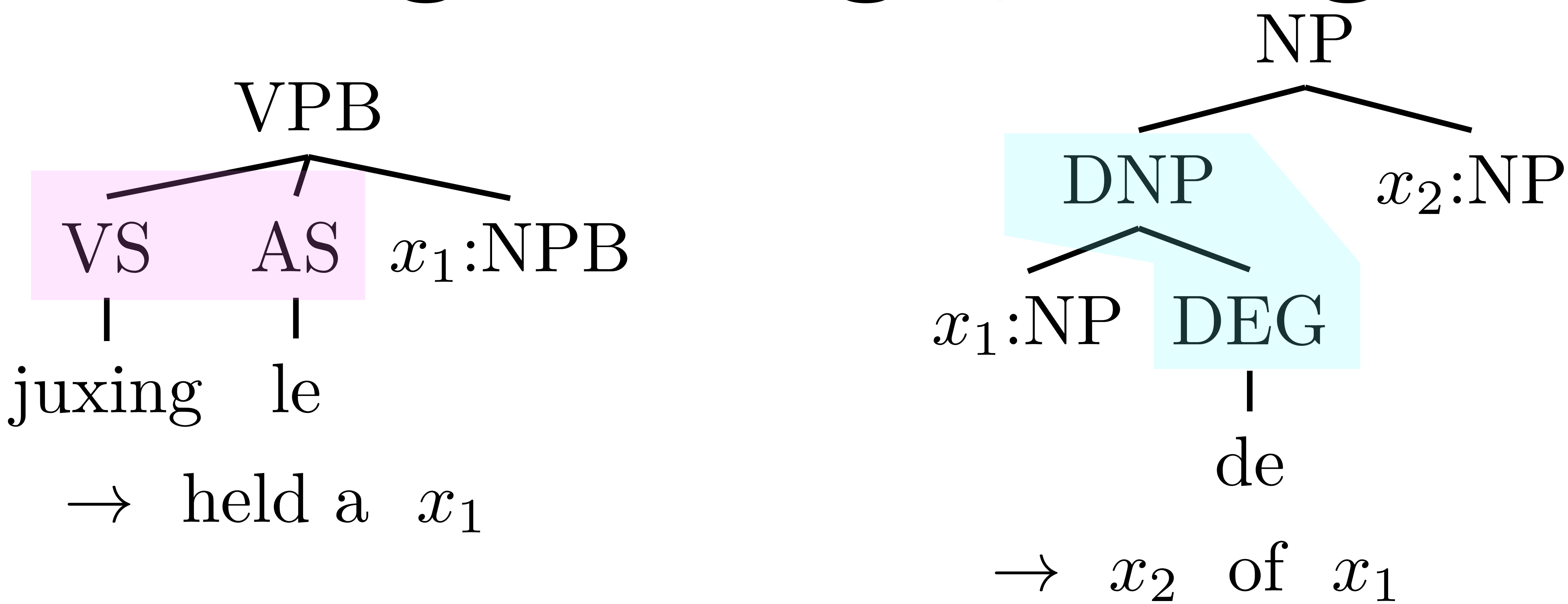
[NP]([ADJP]([JJ,1]) [NP]([NN](机票))) ||| [x]([x,1] airplane ticket) ||| -0.3079667436 -4.0611521158 0.0 -2.0898991524 -11.4306421523 -13.7468707155

[NP]([ADJP]([JJ,1]) [NP]([NN](机票))) ||| [x]([x,1] ticket already issued) ||| -2.020945335 -12.2678308429 0.0 -2.9984937374 -11.4306421523 -15.4598493069

[NP]([ADJP]([JJ,1]) [NP]([NN](机票))) ||| [x]([x,1] ticket) ||| -2.020945335 -0.7597181833 -3.6774507583 -2.1219967549 -11.4306421523 -11.815111746

[NP]([ADJP]([JJ,1]) [NP]([NN](杯))) ||| [x]([x,1] ,) ||| 0.0 -2.7742246718 -7.3946394365 -6.4934017099 -12.7805688693 -7.4339465274

Decoding: String-{String, Tree}



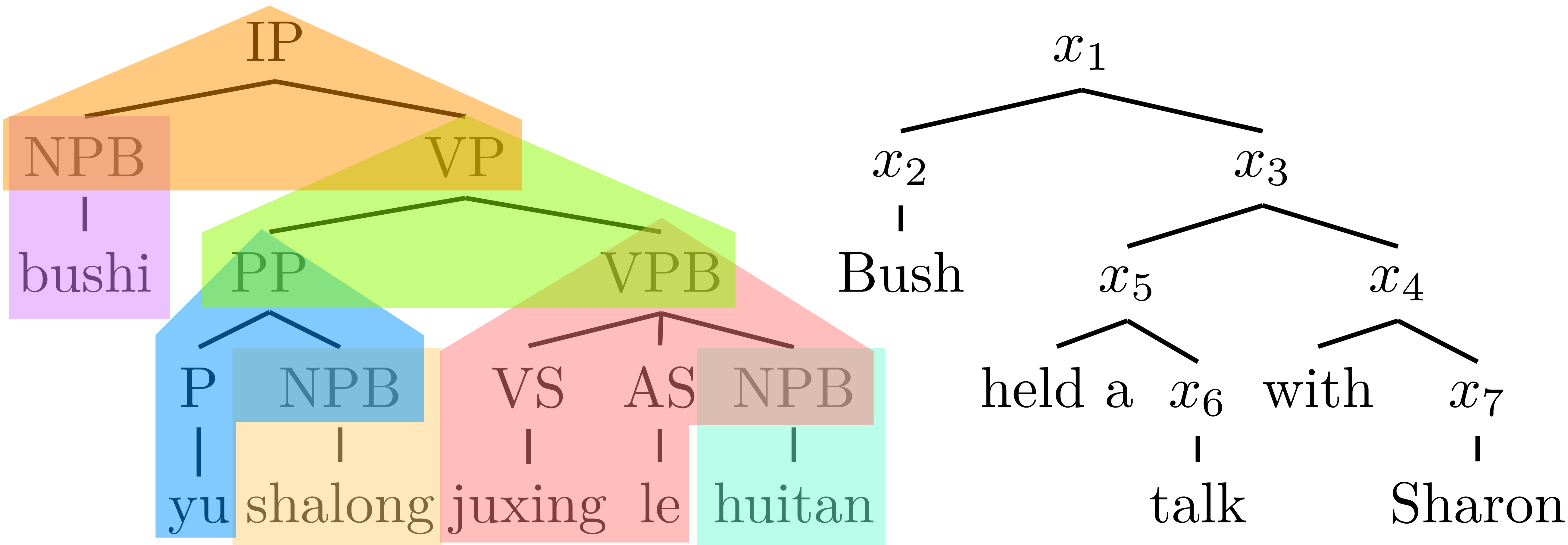
$$\langle \text{VPB} \rightarrow \text{juxing le NPB}_1, \quad x \rightarrow \text{hold a } x_1 \rangle$$

$$\langle \text{NP} \rightarrow \text{NP}_1 \text{ de NP}_2, \quad x \rightarrow x_2 \text{ of } x_1 \rangle$$

(Galley et al., 2004)

- Similar to SCFG decoding: Use the "collapsed" source side rule to perform CKY parsing
- Construct a translation forest using the target side

Decoding: Tree-{String, Tree}



(Huang et al., 2006)

- First, an input sentence is parsed
- Input tree is transformed into a translation forest by tree rewriting (Huang et al., 2006; Zhang et al., 2009)

Forest Rescoring

- Translation by {tree,string}-to-{tree,string}
  - string-to-{tree,sting}: parsing using the source-side grammar
  - tree-to-{tree,string}: parse input sentences + tree-match-rewrite
- Construct forest by the projected target side
- From forests, compute the best derivation (Huang and Chiang, 2005)

Conclusion

- {String, Tree}-to-{String, Tree} translation models
- Rules extraction by GHKM (Galley et al., 2004)
  - Galley M, Hopkins M, Knight K, Marcu D, 2004
- Decoding:
  - String-to-{String, Tree} by CKY
  - Tree-to-{String, Tree} by tree-rewrite

More on Tree-based Models

- Forest-based approach: instead of 1-best parse, use forest encoding k-bests (Mi and Huang, 2008; Mi et al., 2008)
- "Binarized forest" as an alternative to represent multiple parses (Zhang et al., 2011)
- Fuzzy tree-to-tree as a way to overcome "stricktness" of tree-based models (Chiang, 2010)
- Use of dependency (Mi and Liu, 2010; Xie et al., 2011)
- Grammar encoding (Zhang et al., 2009; Ghodke et al., 2011)

Software

- synchronous-CFG
  - Moses: http://www.statmt.org/moses/
  - cdec: http://cdec-decoder.org/
  - joshua: http://joshua-decoder.org/
  - jane: http://www.hltpr.rwth-aachen.de/jane
- synchronous-TSG
  - NONE (You can find a private implementation, though)

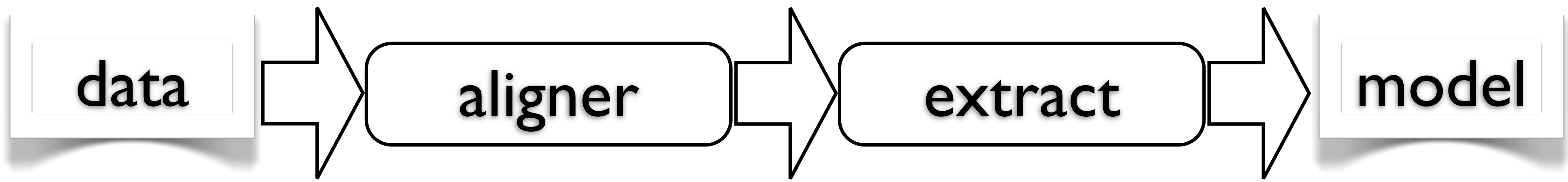
Tree-based MT

- Backgrounds
  - CFG, parsing, hypergraph, deductive system semirings
- Tree-based SMT
  - Synchronous-CFG
  - String-to-Tree, Tree-to-String

SMT2012

- Tutorial
  - Phrase-based MT
  - Tree-based MT
- Recent Topics
  - Phrase/rule induction
  - Tuning

Phrase/Rule Induction



$$p_\phi(\bar{\boldsymbol{f}}|\bar{\boldsymbol{e}}) = \frac{\mathrm{count}(\bar{\boldsymbol{e}}, \bar{\boldsymbol{f}})}{\sum_{\bar{\boldsymbol{f}}'} \mathrm{count}(\bar{\boldsymbol{e}}, \bar{\boldsymbol{f}}')}$$

- Starting from bilingual data, annotate word alignment, extract phrases/rules.... Is it correct?
- MaxLike estimate from counts... Is it correct?
- A solution: non-parametric Bayesian approach

non-parametric Bayesian

- Tutorial (w/o theory)
- Phrase-pair induction

Dice...

- If you throw a dice 6 times and observed only "2"
- What is the probability of observing 2?

$$\begin{aligned} \text{observed data} \quad X &= \quad 2, 2, 2, 2, 2, 2 \\ \text{parameter} \quad \theta &= \quad P(X = 2) = ??? \end{aligned}$$

Dice...

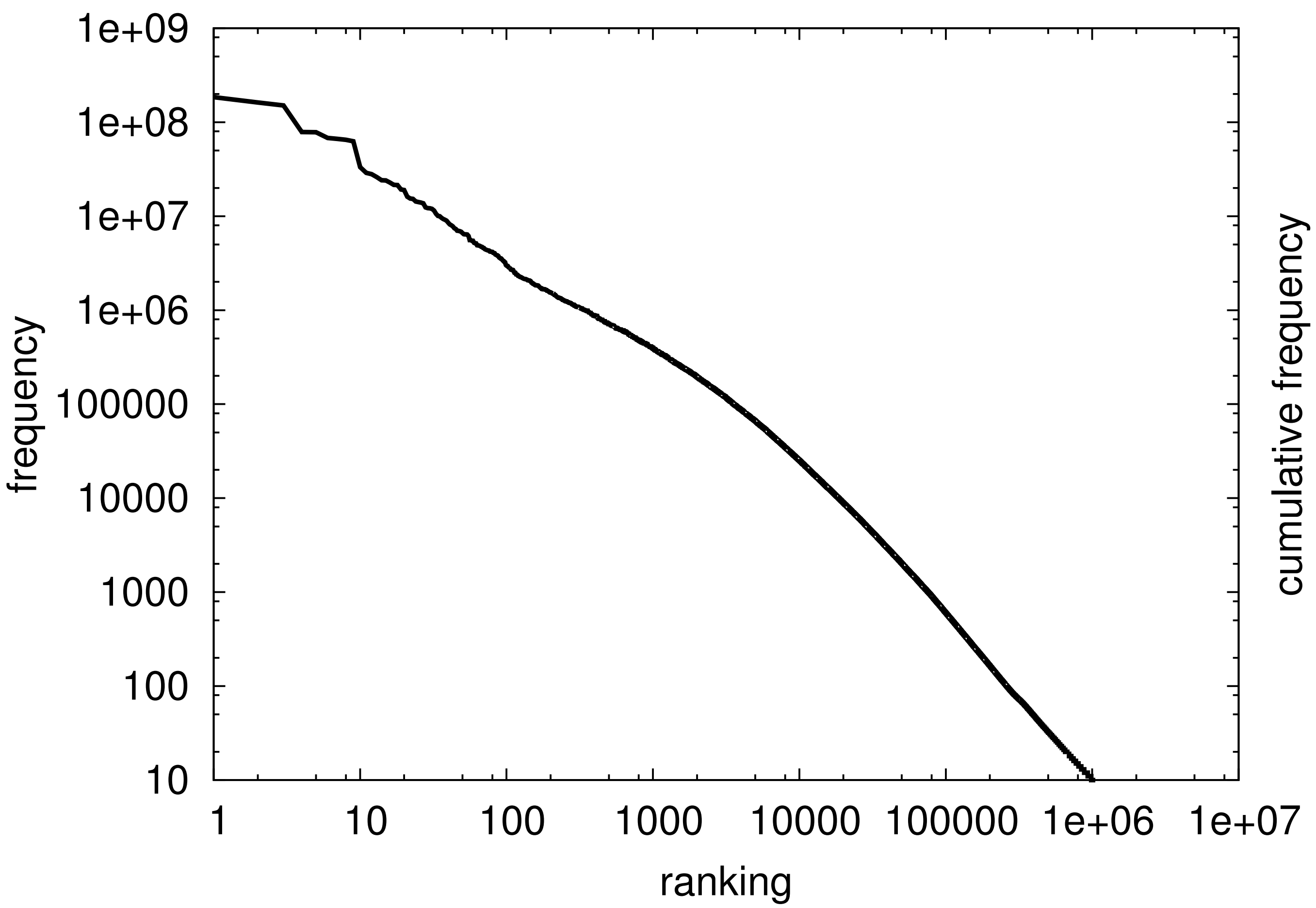
- Maximum Likelihood(ML): $P(X=2)=\theta=\frac{6}{6}=1$
- Maximum A Posterior(MAP):
  - We know a dice has 6 faces: prior distribution of parameter $P(\theta=\frac{1}{6})=0.999$
  - This dice may be skewed: observation likelihood $P(X|\theta)$
  - Derive a posterior: $P(\theta|X)\propto P(X|\theta)P(\theta)$
  - Take a maximum: $P(X=2)=\hat{\theta}=\arg\max_{\theta} P(X|\theta)P(\theta)$

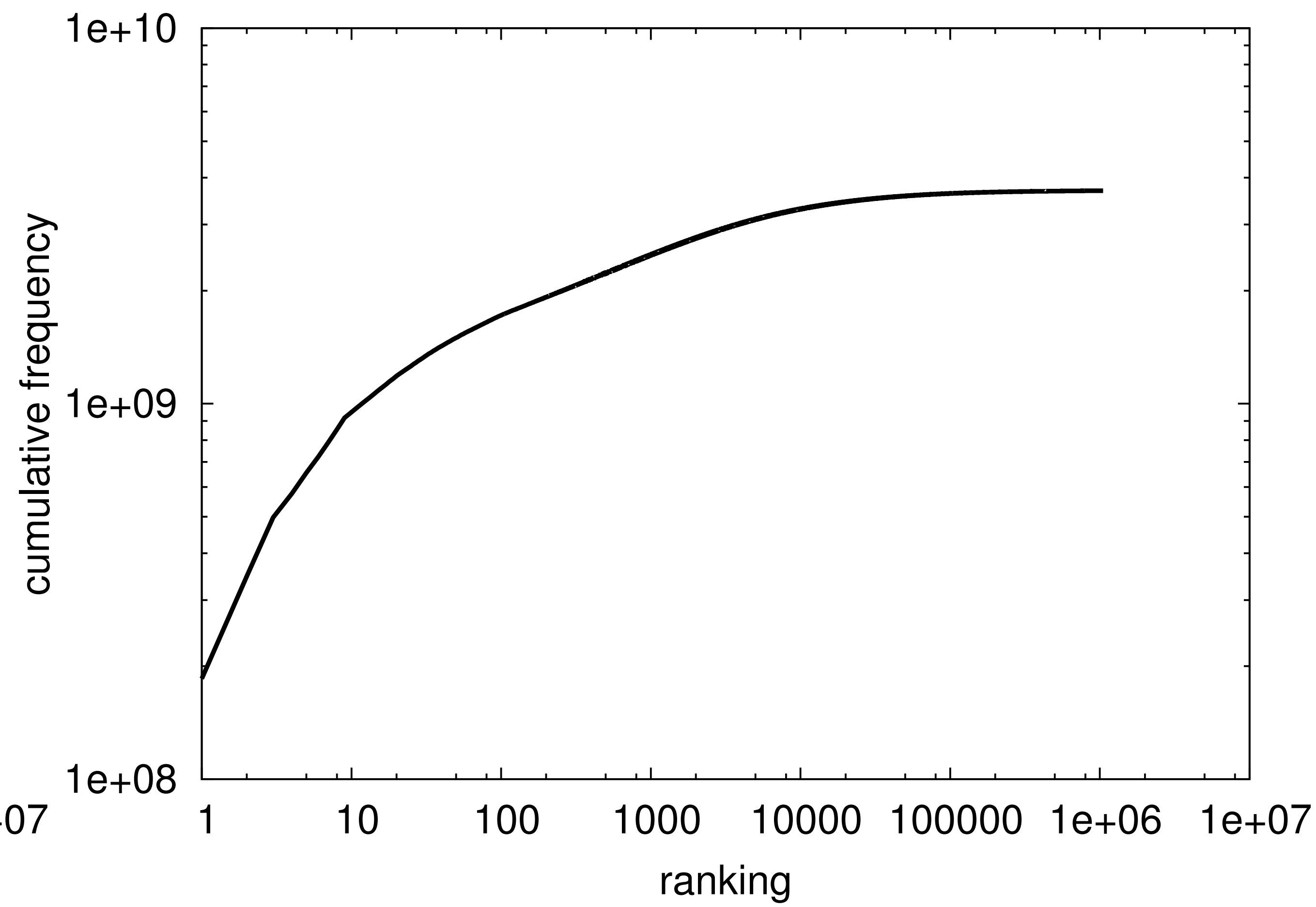
Dice...

- MAP considers a single Θ
- Bayesian:
  - Consider all possible Θ
  - a dice may be old/new, defects etc.

$$P(X=2) = \int_\theta P(X|\theta)P(\theta)\,\mathrm{d}\theta$$

Bayesian for NLP





- Zipf's law: power-low distribution, richer-get-richer effect for word distribution

Probabilities for words

$$
\begin{aligned}
P(x|x_1, x_2, \ldots, x_N) &= \int_\theta P(x|\theta)P(\theta|x_1, x_2, \ldots, x_N)\,\mathrm{d}\theta \\
&= ?
\end{aligned}
$$

- Assign probabilities for words x given $x_1, x_2, \ldots x_N$
- We cannot explicitly compute summation of all possible parameters.
- Obtain “samples” (a sequence of x) from a model assuming a “distribution of a distribution”.

Dirichlet Process

$$P_{\mathrm{DP}}(x|\ldots) = \frac{c(x)}{c(\_) + s} + \frac{s}{c(\_) + s} P_{base}(x|\ldots)$$

$$\begin{aligned} c(x) &= \text{frequency of } x \\ c(\_) &= \sum_x c(x) \end{aligned}$$

- DP can be viewed as back-off smoothing
  - We ignore theoretical details...
- If "strength" s = 0? $P_{\mathrm{ML}}(x|\ldots) = \frac{c(x)}{c(\_)}$
- What is $P_{base}(x)$?

Base Measure

$$
\begin{aligned}
\{\text{the}, \text{dog}, \text{blue}\} &= \{0.33, 0.33, 0.33\} \\
&= \{0.7, 0.15, 0.15\} \\
&= \{?, ?, ?\}
\end{aligned}
$$

- Prior beliefs on the distribution of words:
  - All the words are equally likely.
  - the, a, of, etc. appears more frequently.
  - Newswire starts with headline, timeline etc.

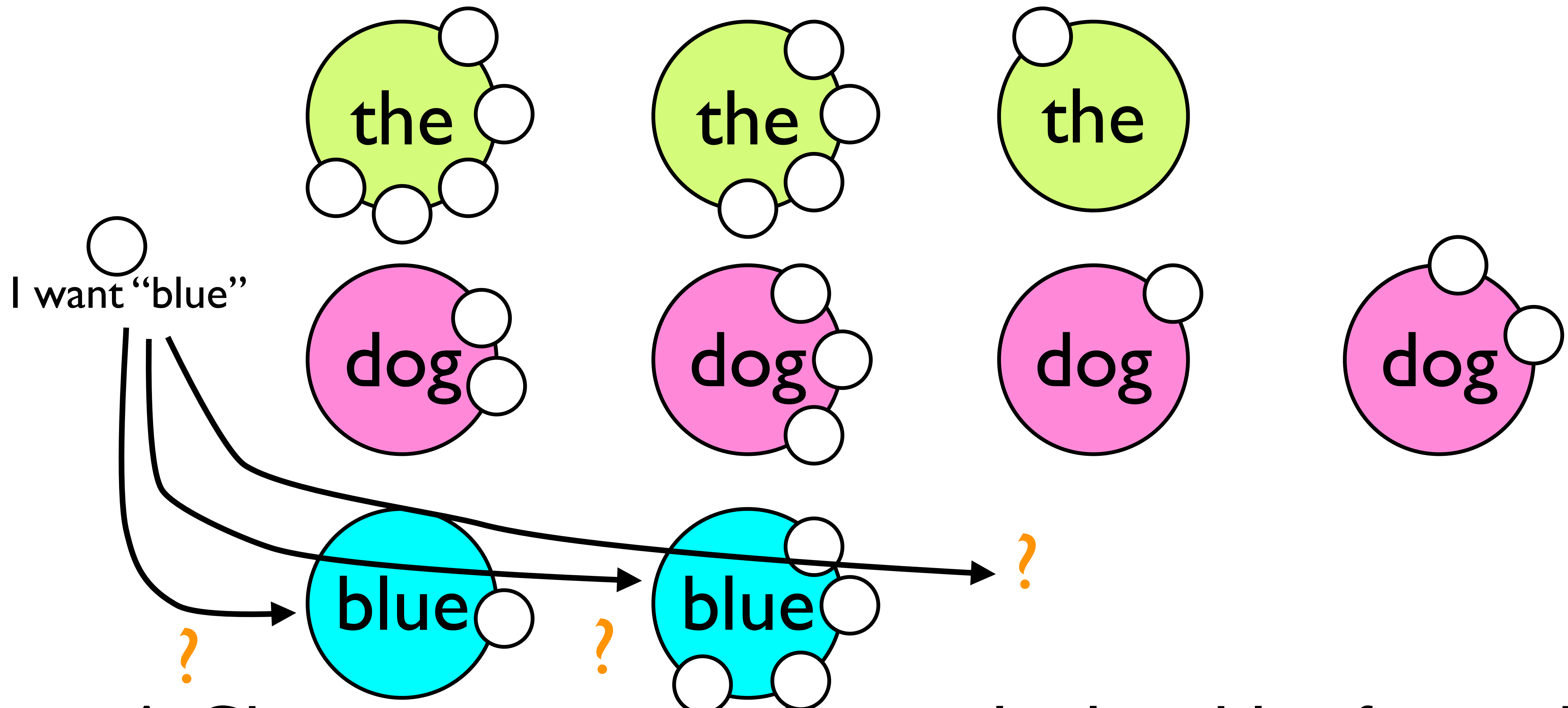
Pitman-Yor Process

$$P_{\mathrm{PY}}(x|\dots) = \frac{c(x) - d \cdot t(x)}{c(\_) + s} + \frac{d \cdot t(\_) + s}{c(\_) + s} P_{base}(x|\dots)$$

$$\theta \sim \mathrm{PY}(d, s, P_{base})$$

- PYP can be viewed as "better" back-off smoothing
- We will often write: Θ (a set of parameters for $P_{PY}(x|\dots)$) is "drawn" from PY process.
- d = discount, s = strength
- What are t(x) and t(_) ?

Chinese Restaurant Process



- A Chinese restaurant has multiple tables for each "dish" (= token type) which is served for each customer
- t(x) = # of tables for x, t(_) = total # of tables in the restaurant
- What is P_{PY}(bleu)? (assume P_{base}=0.25, d=0.9, s=1)

Chinese Restaurant Process

$$P_{\mathrm{PY}}(x|\ldots) = \frac{c(x) - d \cdot t(x)}{c(\_) + s} + \frac{d \cdot t(\_) + s}{c(\_) + s} P_{base}(x|\ldots)$$

- When a new customer x enters the restaurant:
  - Seat at an existing table by: $\frac{c(x) - d \cdot t(x)}{c(\_) + s}$
    - And, select the i_{th} table proportional to its popularity (richer-get-richer!) by: $\frac{c_i(x) - d}{c(x) - d \cdot t(x)}$
  - Or, create a new table by: $\frac{d \cdot t(\_) + s}{c(\_) + s} P_{base}(x|\ldots)$
- NOTE: we need to re-normalize the probabilities for the existing/new selection

Chinese Restaurant Process

$$P_{\mathrm{PY}}(x|\ldots) = \frac{c(x) - d \cdot t(x)}{c(\_) + s} + \frac{d \cdot t(\_) + s}{c(\_) + s} P_{base}(x|\ldots)$$

- When a customer x exit the restaurant:
    - Choose a customer x equally likely... (Customers have no choice!)
    - If a table becomes empty, remove it.
    - If we consider the discount (d), this will lead to biased distribution.

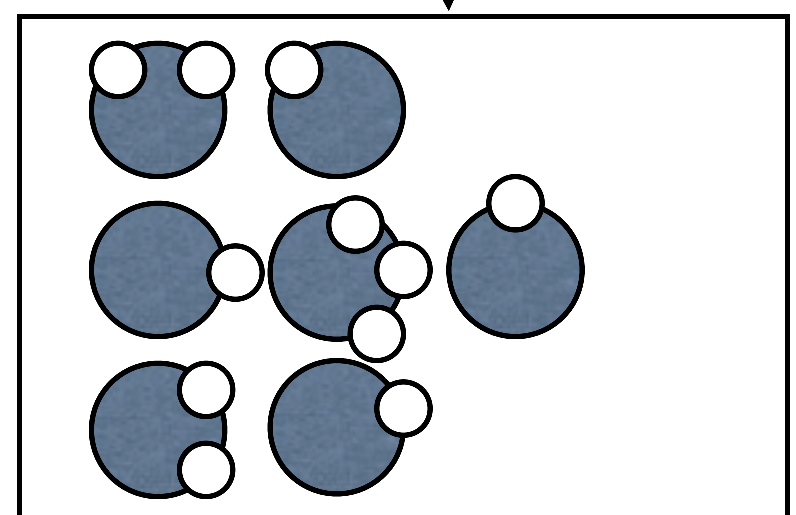
Gibbs Sampling

$x_1, x_2, \ldots x_N$

Choose a word, x

remove it from the restaurant

--c(x)
(and --t(x),
when required)



++c(x)
(and ++t(x),
when required)

add it in the restaurant

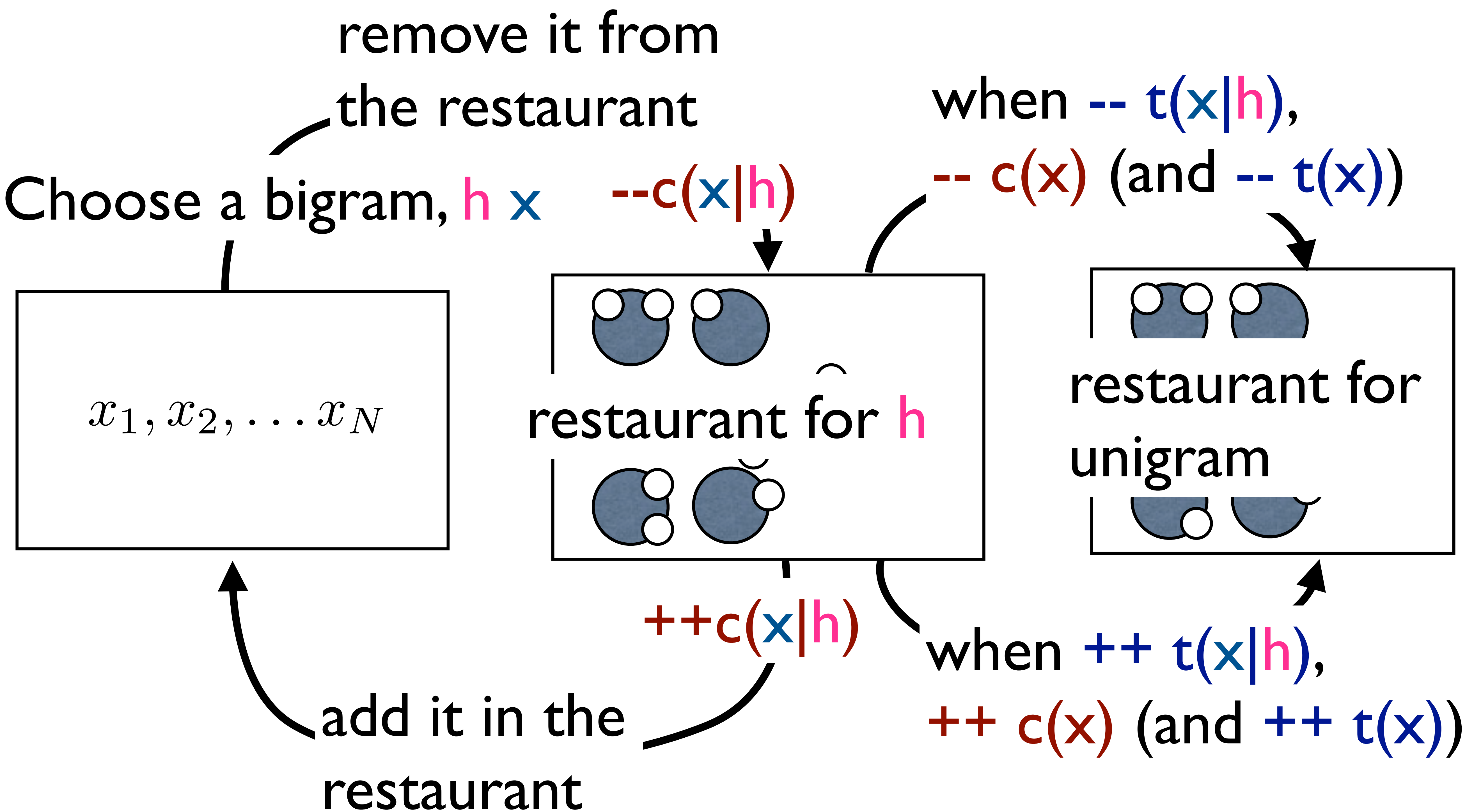
Unigram to Bigram

$$P^2_{\mathrm{PY}}(x|\dots) = \frac{c(x|h) - d^2 \cdot t(x|h)}{c(\_|h) + s^2} + \frac{d^2 \cdot t(\_|h) + s^2}{c(\_|h) + s^2} P^1_{PY}(x|\dots$$

$$P^1_{\mathrm{PY}}(x|\dots) = \frac{c(x) - d^1 \cdot t(x)}{c(\_) + s^1} + \frac{d^1 \cdot t(\_) + s^1}{c(\_) + s^1} P_{base}(x|\dots)$$

- h = previous word (in our example, x_N)
- bigram model can fallback to unigram model (as in n-gram language models)
- Restaurant for each h
- If t(x) and t(x|h)is always 1, then it is identical to Kneser-Ney smoothing (Kneser and Ney, 1995)

Hierarchical Update



PYP n-gram Language Model

$$
\begin{aligned}
\theta^{n}_{h_1,\ldots,h_{n-1}} &\sim \mathrm{PY}(d^n, s^n, \theta^{n-1}_{h_2,\ldots,h_{n-1}}) \\
\theta^{n-1}_{h_2,\ldots,h_{n-1}} &\sim \mathrm{PY}(d^{n-1}, s^{n-1}, \theta^{n-2}_{h_3,\ldots,h_{n-1}}) \\
&\vdots \\
\theta^{2}_{h_{n-1}} &\sim \mathrm{PY}(d^2, s^2, \theta^1) \\
\theta^{1} &\sim \mathrm{PY}(d^1, s^1, \theta^0)
\end{aligned}
$$

- Hierarchically draw parameters for n-gram language model (with uniform distribution: Θ^0) (Teh, 2006)

Hyperparameters: d, s

$$
\begin{aligned}
d^n &\sim \mathrm{Beta}(\alpha^n, \beta^n) \\
s^n &\sim \mathrm{Gamma}(a^n, b^n)
\end{aligned}
$$

- We can manually set hyperparameters: d and s
- A standard practice is to sample from “restaurants” assuming some distributions... (But we will omit it for brevity... see Teh (2006))
- NOTE: We have multiple gamma distribution definitions, Here: a = shape parameter, b = rate parameter (or, inverse scale parameter)

Software

- LM-related implementation
  - Latent word LM (Deschacht and Moens, 2009): http://chasen.org/~daiti-m/dist/lwlm/
  - Word segmentation LM (Mochihashi et al., 2009): http://www.phontron.com/latticelm/

non-parametric Bayesian

- Tutorial (w/o theory)
- Phrase-pair induction

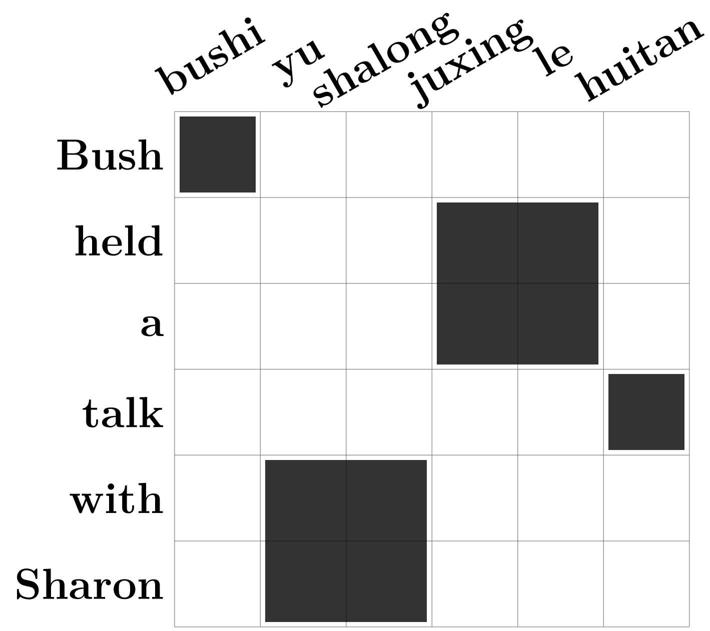
Phrase-pair Induction

$$
\begin{aligned}
P(\theta|\langle F, E\rangle) &\propto P(\langle F, E\rangle|\theta)P(\theta) \\
&= \prod_{\langle f, e\rangle \in \langle F, E\rangle} P(\langle f, e\rangle|\theta)P(\theta)
\end{aligned}
$$

- Directly learns Θ for phrase pairs from bilingual data w/o word alignment (DeNero et al., 2008; Arun et al., 2009; Neubig et al., 2011)
- EM-algorithm suffers serious de-generation problem (Marcu and Wong, 2002)
- non-parametric Bayesian for assuming priors for parameters (or, place stronger preferences for de-composed phrases)

PYP for Phrase Pairs

$$P_{\mathrm{PY}}(x|\ldots) = \frac{c(x) - d \cdot t(x)}{c(\_) + s} + \frac{d \cdot t(\_) + s}{c(\_) + s} P_{base}(x|\ldots)$$



$= \Phi$

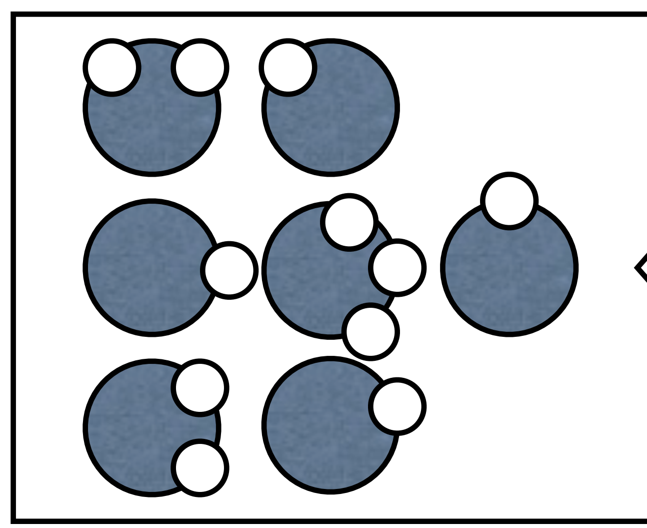
- x: phrase pair, $P_{\text{base}}(x)$: IBM Model 1 + unigram LM
- In contrast with the n-gram language model, phrasal segmentation is "hidden"
- We compute the derivations Φ for all the bilingual data by Gibbs sampling

Gibbs for Phrase Pairs

Bilingual data with derivations

$$\{\ldots, \langle f, e, \phi \rangle, \ldots\}$$

1. Choose data
2. Choose operator
3. Choose a part of the derivation Φ_i



decrement for Φ_i

Compute $P_{PY}(x)$ for all possible derivations

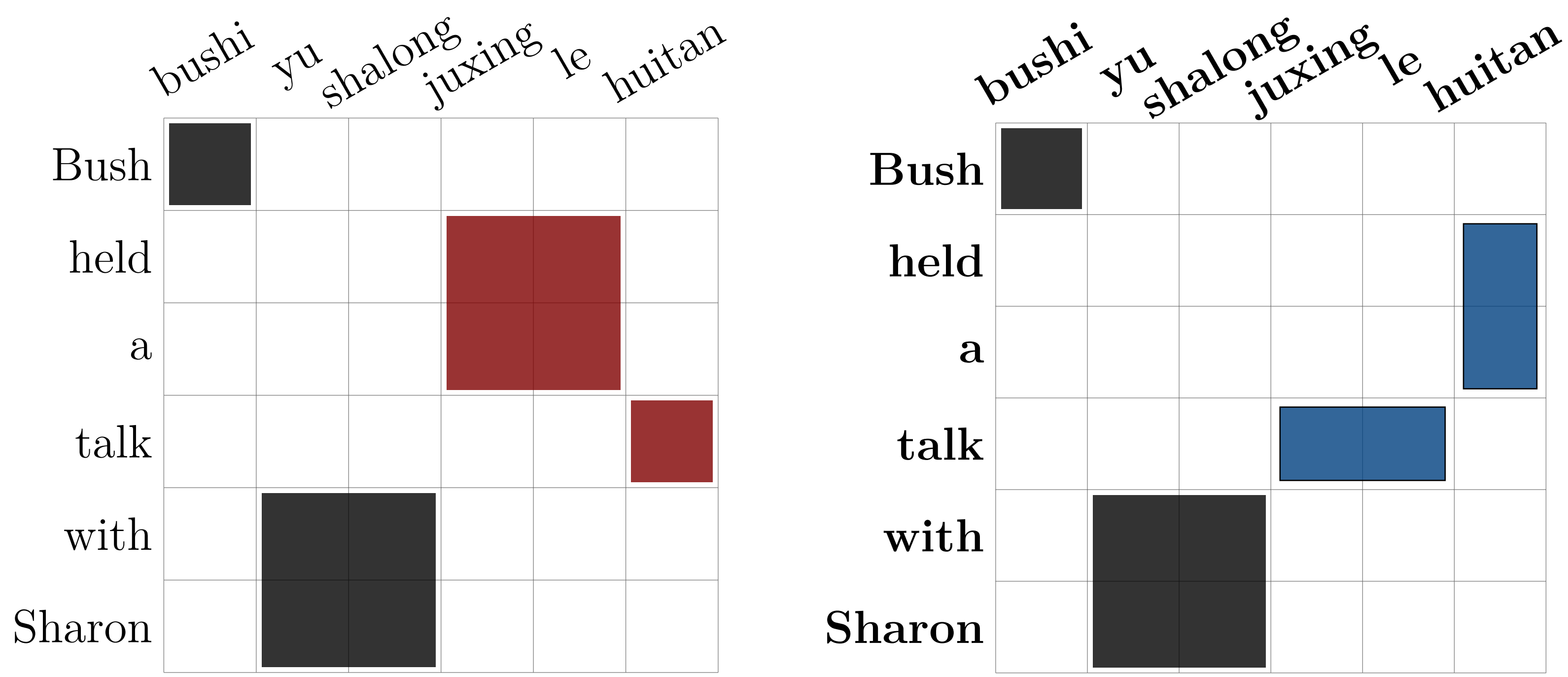
Choose new Φ_i

increment for new Φ_i

Update derivation

A Sample Gibbs (SWAP)

Φ_i and Φ_{i+1}



Φ'_i and Φ'_{i+1}

(DeNero et al., 2008)

- Decrement old Φ_i and Φ_{i+1}
- Choose by $P_{PY}(\Phi_i, \Phi_{i+1})$ and $P_{PY}(\Phi'_i, \Phi'_{i+1})$ (normalize the probabilities before selection!)
- Φ_i, Φ_{i+1}, Φ'_i, Φ'_{i+1}: do not affect other derivations!
- Increment new Φ_i and Φ_{i+1}

Sampling for Phrase Pairs

- Swap, Flip, Toggle, Move operators (DeNero et al., 2008)
- Requires many (100 to 1,000) iterations
- Essentially, NP-hard problem for phrase alignment
- ITG to restrict the alignment with a Polynomial algorithm

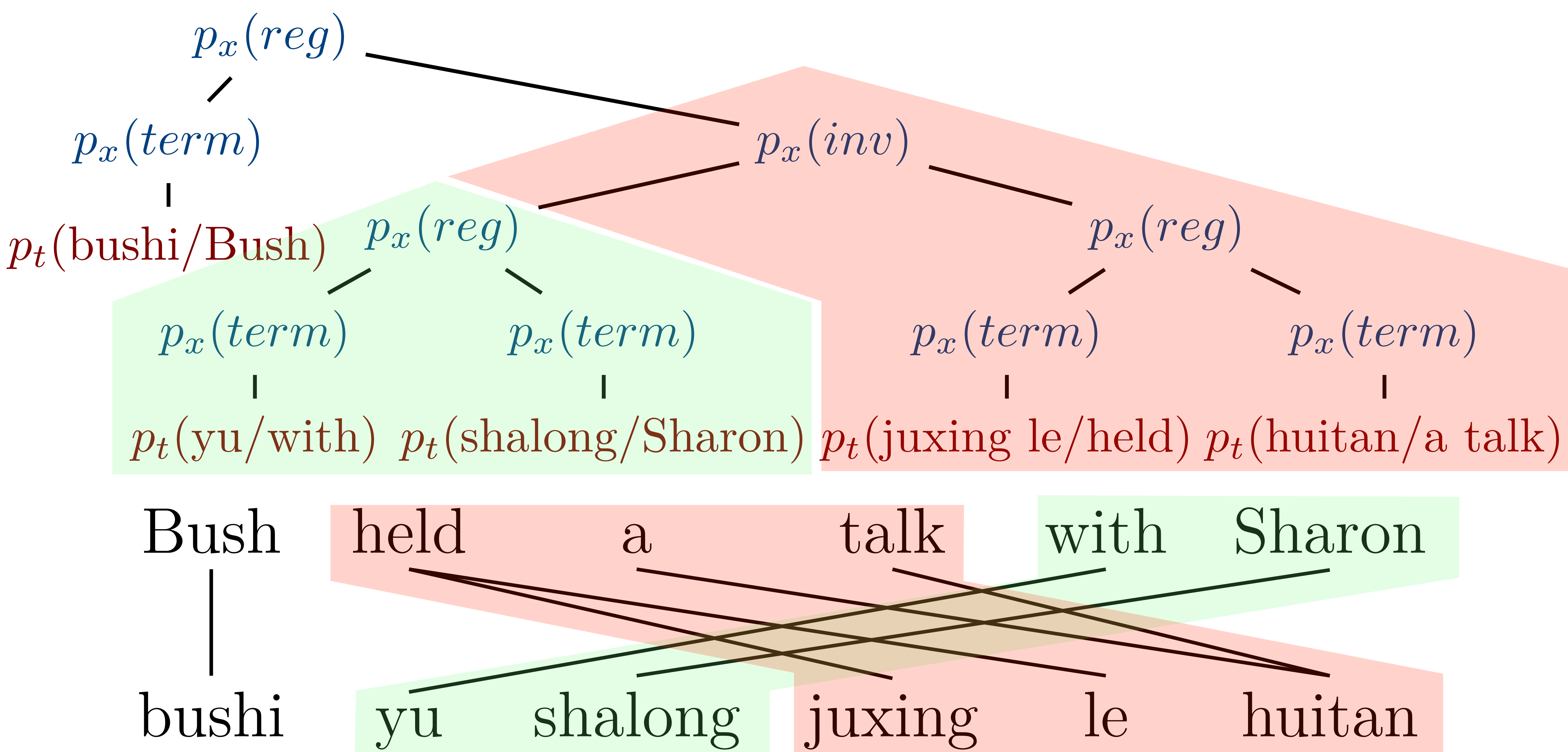
ITG

$$
\begin{aligned}
\mathrm{X} &\rightarrow \langle \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{2}}, \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{2}} \rangle \\
\mathrm{X} &\rightarrow \langle \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{2}}, \mathrm{X}_{\boxed{2}}\ \mathrm{X}_{\boxed{1}} \rangle \\
\mathrm{X} &\rightarrow \langle f, e \rangle
\end{aligned}
$$

$$
\mathrm{X} \rightarrow [X\ \ X] \mid \langle X\ \ X \rangle \mid f/e
$$

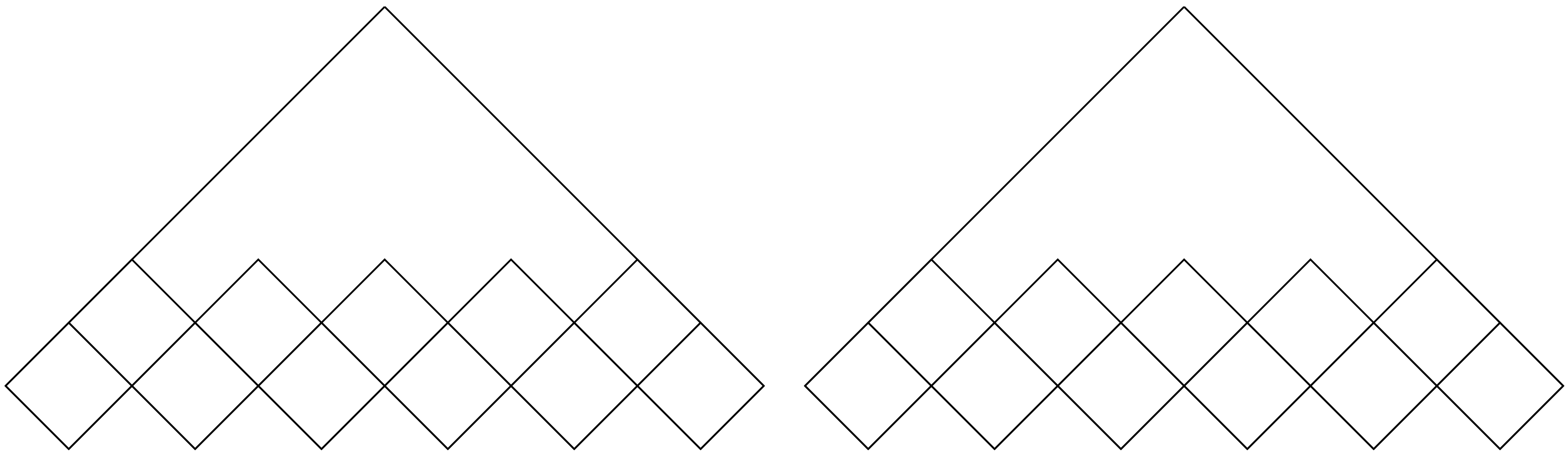
- Inversion Transduction Grammar (ITG) (Wu, 1997) which is an instance of synchronous-CFG
- Exploited for word alignment (Wu, 1997; Zhang and Gildea, 2005; Haghighi et al., 2009), phrase alignment (Cherry and Lin, 2007; Zhang et al., 2008), constraints for decoding (Zens and Ney, 2003; Zens et al., 2004; Cherry et al., 2012)

ITG for Phrase Induction



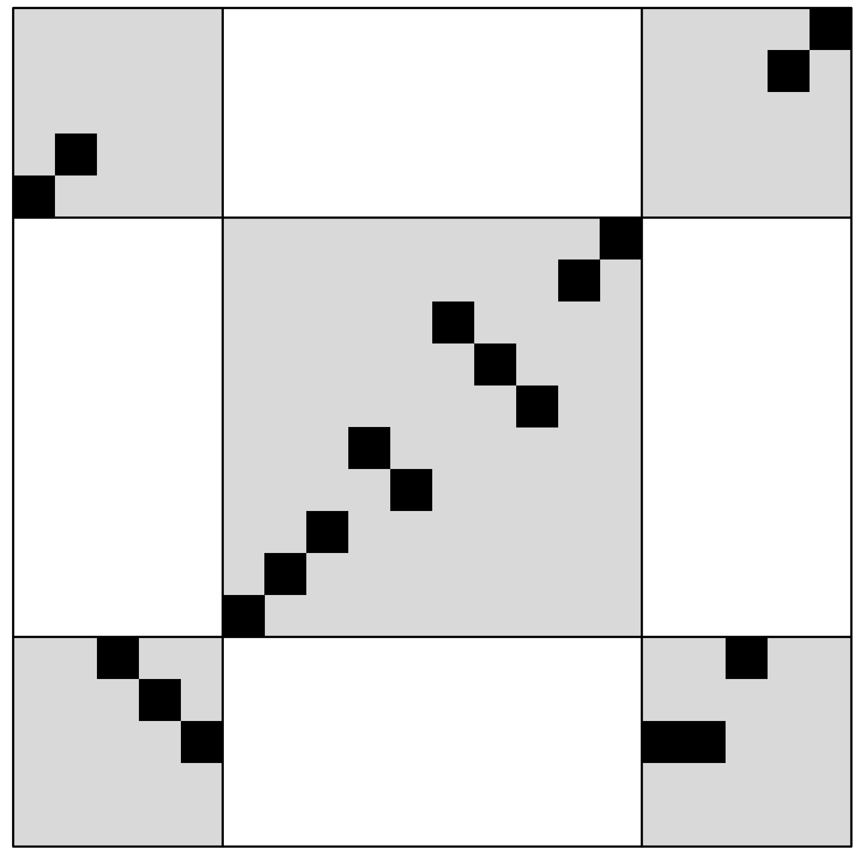
- Phrasal alignment by ITG (Cherry and Lin, 2007)
- "parsing" for efficient sampling

Bitext Parsing



- Intersection between SCFG and two texts
- $O(N^3 M^3)$ for ITG (Wu, 1997)
  - For each length n and m, for each position i and j, for each rule X ➔ YZ、for each split k and l

Span Pruning



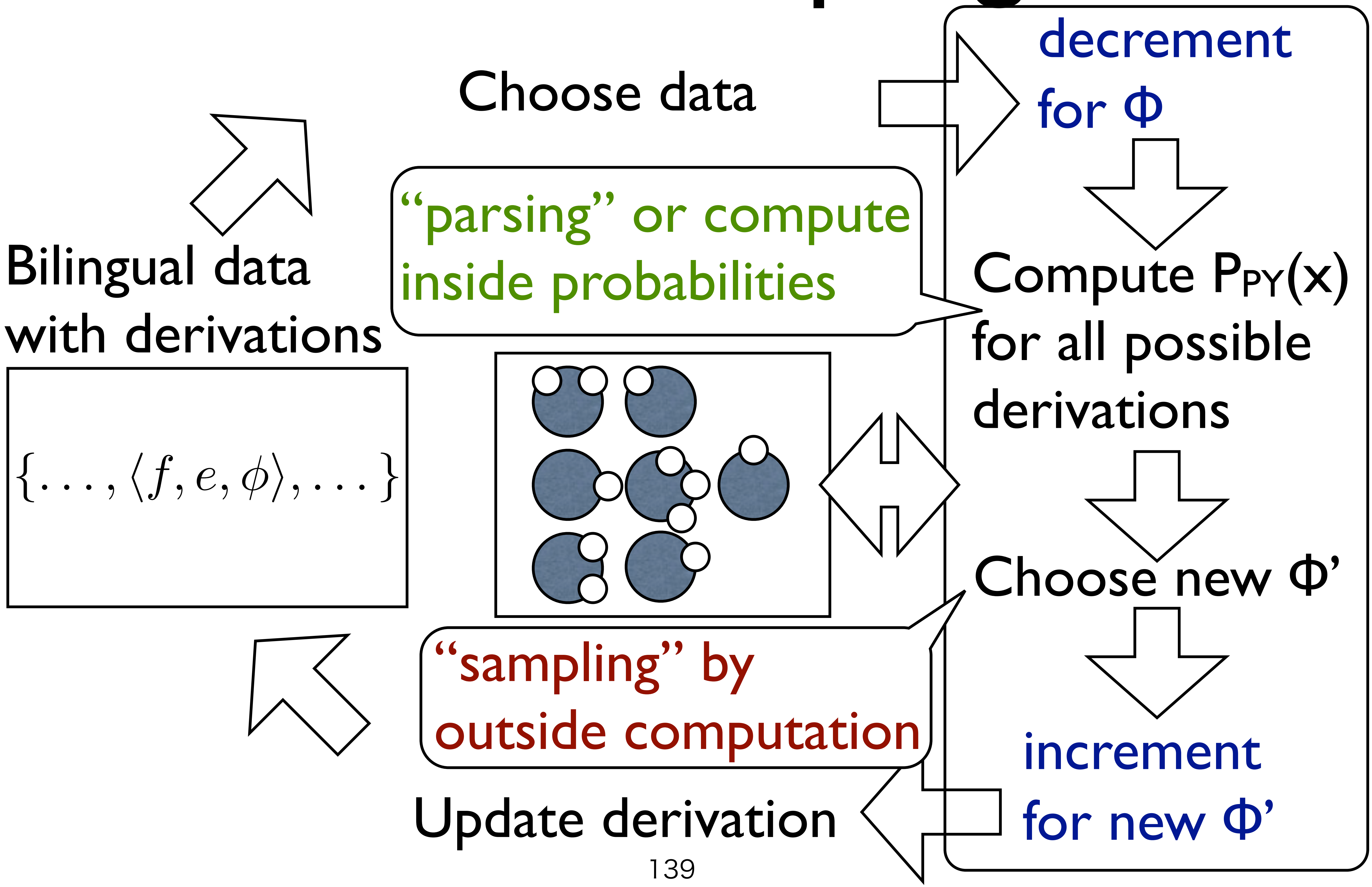
- You do not have to visit all the span pairs
- Use figure-of-merit to prune spans
  - $O(n^4)$ for a naive algorithm (Zhang and Gildea, 2005)
  - $O(n^3)$ for a DP-based algorithm (Zhang et al., 2008)

Beam Pruning

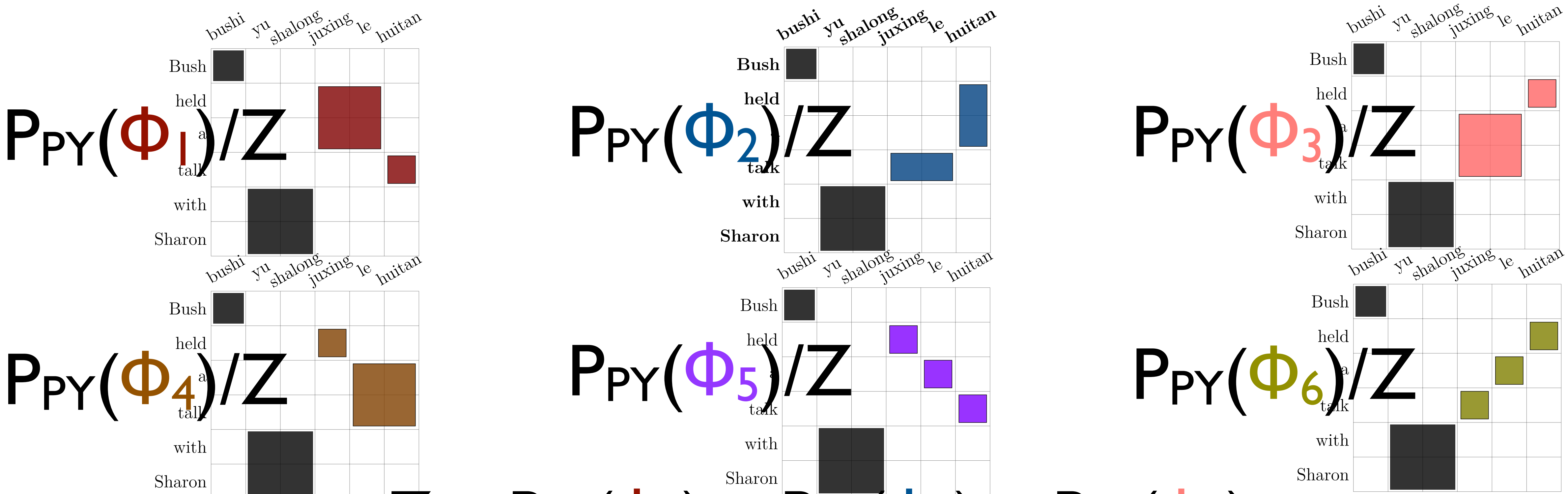
--●---,------	--●---,--●---	--●---,--●●--	--●●--,--●●--
-●----,------	-●----,--●---	-●----,--●●--	●●----,--●●--
------,----●-	---●--,----●-	---●●-,----●-	---●●-,---●●-
------,---●--	------,--●●--	---●--,--●●--	---●●-,--●●--
------,-●----	---●--,-●----	--●●--,-●----	--●●--,●●----

- Re-organize the search space by the cardinality (= # of source/target words parsed) (Saers et al., 2009)
- Prune by the cardinality: Complexity $O(bn^3)$
- Simple look-ahead (Neubig et al., 2012)

Block Sampling



Block Sampling



$P_{PY}(\Phi_1)/Z$ $P_{PY}(\Phi_2)/Z$ $P_{PY}(\Phi_3)/Z$

$P_{PY}(\Phi_4)/Z$ $P_{PY}(\Phi_5)/Z$ $P_{PY}(\Phi_6)/Z$

$$Z = P_{PY}(\Phi_1) + P_{PY}(\Phi_2) + P_{PY}(\Phi_3) + P_{PY}(\Phi_4) + P_{PY}(\Phi_5) + P_{PY}(\Phi_6)$$

- Instead of considering a single variable, or a single operator, sample a new "sentence-wise" derivation Φ'
- Bottom-up to compute span-probabilities, top-down for sampling using the computed span-probabilities
  - Span probabilities are normalized, then, draw sample

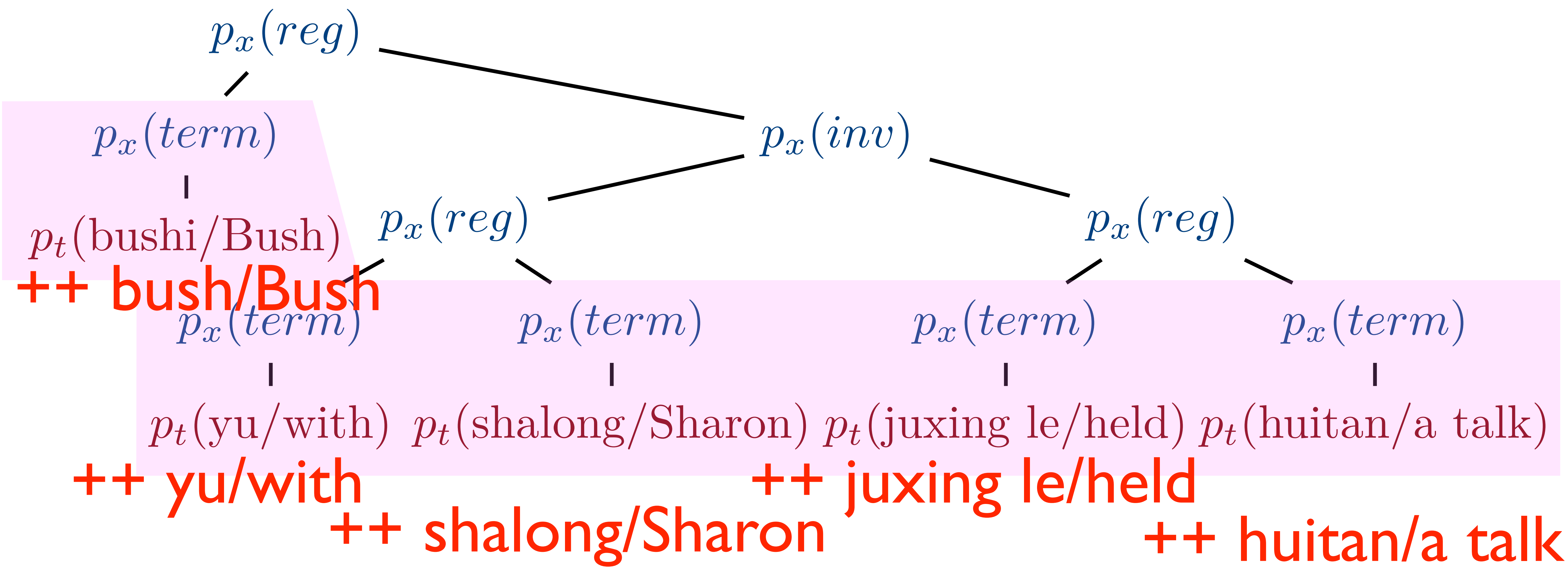
MH Step

$$\text{accept } \phi'? \sim \min\left\{1, \frac{\pi_1 q(\phi|\phi')}{\pi_0 q(\phi'|\phi)}\right\}$$

$$\begin{aligned}\pi_0 &= P_{\mathrm{PY}}(\phi|\text{current restaurant})\\ \pi_1 &= P_{\mathrm{PY}}(\phi'|\text{proposal restaurant})\end{aligned}$$

- Metropolis-Hastings to "judge" whether to accept the proposal distribution with new Φ' in the restaurant
- q: a distribution from which we draw Φ' (normalized inside scores)
- Why? Heuristic pruning may draw parameters from an unknown distribution: MH step to assure sampling from the model

Minimum Phrases

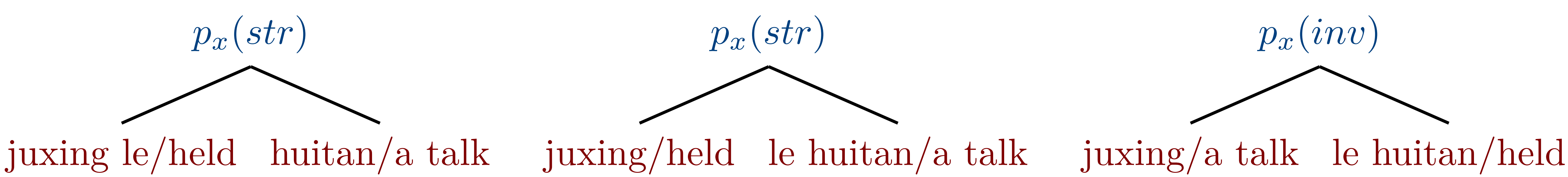


- Sampled derivations contain only minimum phrases
- Longer phrases are heuristically extracted (DeNero et al., 2008; Zhang et al., 2008; Blunsom et al., 2009)

Fallback Modeling

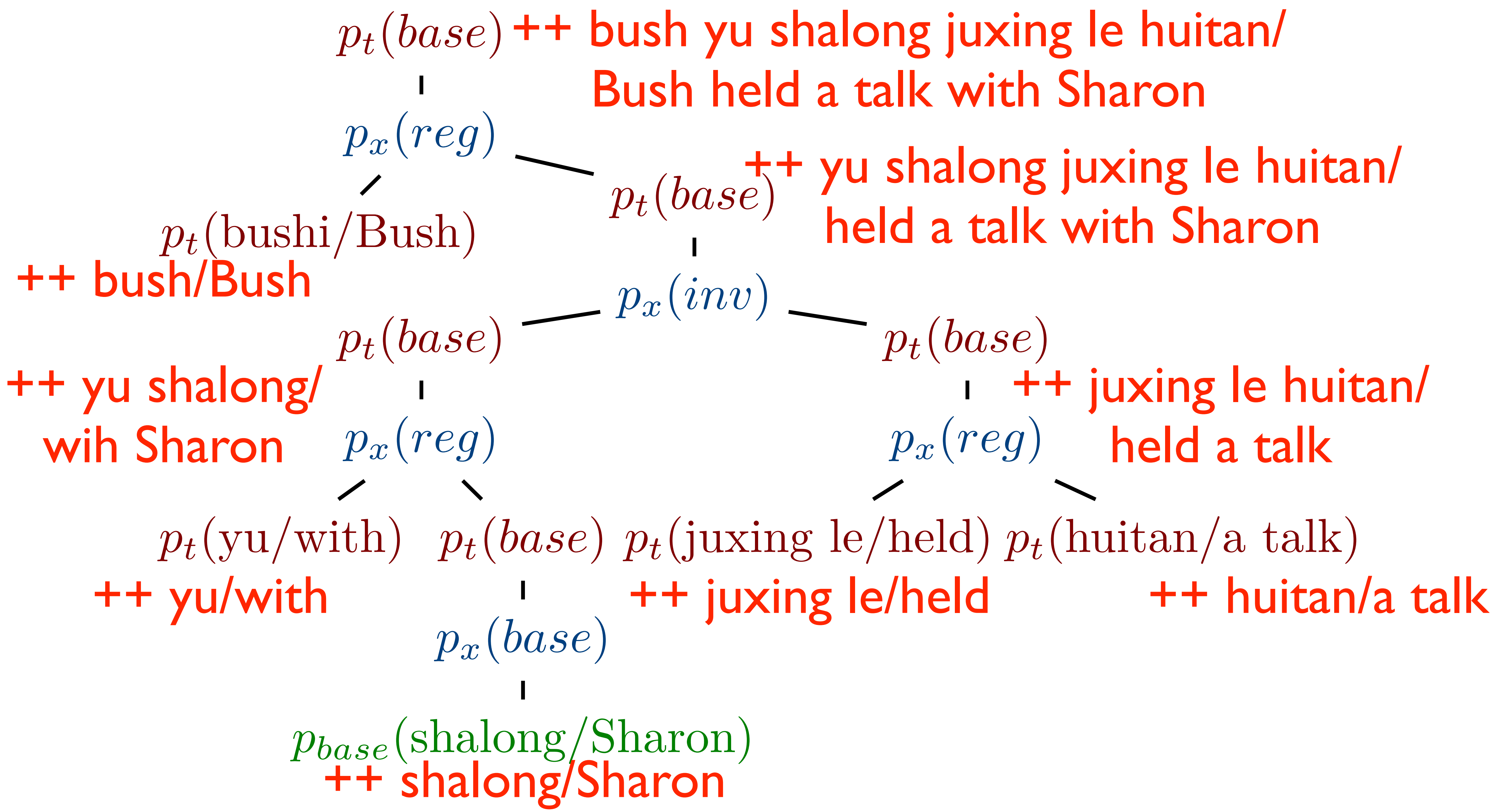
$$P_{\mathrm{PY}}(x|\ldots) = \frac{c(x) - d \cdot t(x)}{c(\_) + s} + \frac{d \cdot t(\_) + s}{c(\_) + s} P_{dac}(x|\ldots)$$

$$P_{dac}(x|\ldots) = \begin{cases} P_x(base) P_{base}(x|\ldots) \\ P_x(str) P_{\mathrm{PY}}(y|\ldots) P_{\mathrm{PY}}(z|\ldots) \\ P_x(inv) P_{\mathrm{PY}}(y'|\ldots) P_{\mathrm{PY}}(z'|\ldots) \end{cases}$$



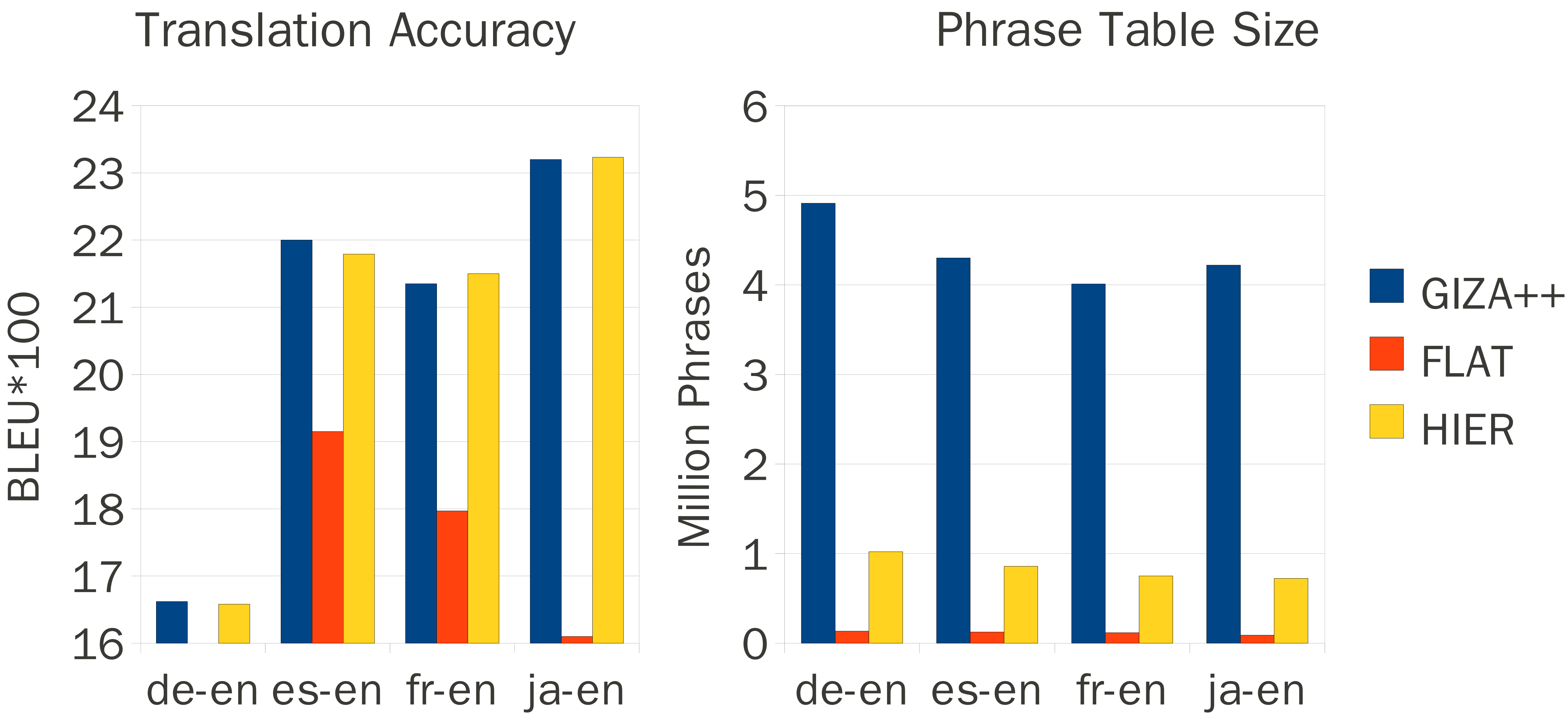
- Hierarchical PYP as in PYP n-gram LM!(Neubig et al., 2011; Neubig et al., 2012)
- “Base measure” encodes multiple splitting choice

Exhaustive ITG Phrases



- Recursively divide-and-conquer
- Increment table-count at all the granularities

Experiments



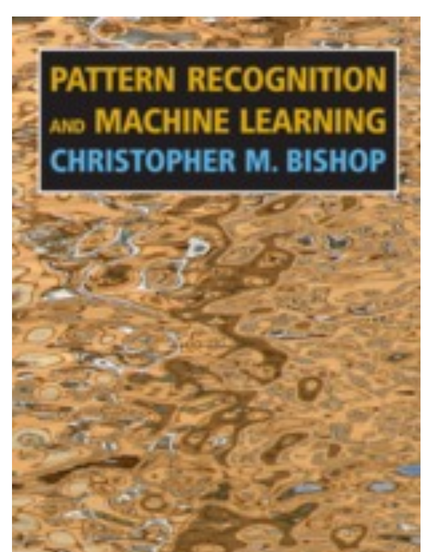
(Neubig et al., 2011)

- Smaller model, competitive to baselines

http://www.phontron.com/pialign

Conclusion

- non-parametric Bayesian is a powerful method for unsupervised learning
- Already exploited for: synchronous-CFG (Blunsom et al., 2009; Levenberg et al., 2012) and synchronous-TSG (Cohn and Blunsom, 2009) in a limited fashion
- Further readings:



SMT2012

- Tutorial
  - Phrase-based MT
  - Tree-based MT
- Recent Topics
  - Phrase/rule induction
  - Tuning

Tuning

$$
\begin{aligned}
\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} \frac{\exp\left(\mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})\right)}{\sum_{\mathbf{e}', \phi'} \exp\left(\mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}', \phi', \mathbf{f})\right)} \\
&= \operatorname*{argmax}_{\mathbf{e}} \mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}, \phi, \mathbf{f})
\end{aligned}
$$

- Linear model: features are scaled by w
- Problem 1: many alternative translations (e) possible with many alternative "hidden variables" (Φ)
  - We cannot enumerate all possible variables
- Problem 2: translation error metric is corpus-wise, not sentence-wise (i.e. BLEU; Papineni et al., 2002)

Evaluation: ngram precision

Well , I 'd like to stay five nights beginning October twenty-fifth to thirty .

- I 'd like to stay there for five nights , from October twenty fifth to the thirtieth .
- I want to stay for five nights , from October twenty fifth to the thirtieth .
- I 'd like to stay for five nights , from October twenty fifth to the thirtieth .
- I would like to reserve a room for five nights , from October twenty fifth to the thirtieth .

Evaluation: ngram precision

Well , I 'd like stay five nights beginning October twenty-fifth to thirty .

$$p_1 = \frac{11}{15}$$

- I 'd like to stay there for five nights , from October twenty fifth to the thirtieth .
- I want to stay for five nights , from October twenty fifth to the thirtieth .
- I 'd like to stay for five nights , from October twenty fifth to the thirtieth .
- I would like to reserve a room for five nights , from October twenty fifth to the thirtieth .

Evaluation: ngram precision

Well , I 'd like to stay five nights beginning October twenty-fifth to thirty .

$$p_1 = \frac{11}{15} \qquad p_2 = \frac{5}{14}$$

- I 'd like to stay there for five nights , from October twenty fifth to the thirtieth .
- I want to stay for five nights , from October twenty fifth to the thirtieth .
- I 'd like to stay for five nights , from October twenty fifth to the thirtieth .
- I would like to reserve a room for five nights , from October twenty fifth to the thirtieth .

Evaluation: ngram precision

Well , I 'd like to stay five nights beginning October twenty-fifth to thirty .

$$p_1 = \frac{11}{15} \qquad p_2 = \frac{5}{14} \qquad p_3 = \frac{3}{13}$$

- I 'd like to stay there for five nights , from October twenty fifth to the thirtieth .
- I want to stay for five nights , from October twenty fifth to the thirtieth .
- I 'd like to stay for five nights , from October twenty fifth to the thirtieth .
- I would like to reserve a room for five nights , from October twenty fifth to the thirtieth .

Evaluation: ngram precision

Well , I 'd like to stay five nights beginning October twenty-fifth to thirty .

$$p_1 = \frac{11}{15} \qquad p_2 = \frac{5}{14} \qquad p_3 = \frac{3}{13} \qquad p_4 = \frac{2}{12}$$

- I 'd like to stay there for five nights , from October twenty fifth to the thirtieth .
- I want to stay for five nights , from October twenty fifth to the thirtieth .
- I 'd like to stay for five nights , from October twenty fifth to the thirtieth .
- I would like to reserve a room for five nights , from October twenty fifth to the thirtieth .

Evaluation: BLEU

$$\exp\left(\sum_{n=1}^{N} w_n \log p_n + \min(1 - \frac{r}{c}, 0)\right)$$

- (Uniformly) weighted combination of precision (Papineni et al., 2002)
- brevity penalty: penalize too short sentences
  - r = reference length, c = candidate length
  - If we have multiple "r", choose the closest-shortest reference to "c"
- Both factors are computed over the whole document

Why BLEU?

- Used as a standard metric for more than 10 years: Progress of SMT is due by BLEU!
  - Easy to compute ngram statistics
  - However, non-linear decomposition into sentences: corpus-wise metric, thus, harder to optimize
  - BP-problem(Chiang et al., 2009): You can generate spuriously long translations together with a highly confident short translations.
- An alternative to BLEU is a good research topic!

A Bad Example

"we come from the land of the ice and snow"
"from the midnight sun where the hot springs flow"

system 1

xxx xxx xxx xxx land xxx xxx ice xxx snow
xxx xxx midnight xxx xxx xxx hot xxx flow

system 2

x come x x land x x ice x snow x x x x x
from xxx sun xxx

- Both shared the same # of words, and the same # of matches

Tuning

- Batch learning
- Online learning

k-best approximation

procedure BatchLearn($\langle F, E \rangle = \left\{ \langle \boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)} \rangle \right\}_{i=1}^{N}$)
  $\boldsymbol{w}^{(0)} \leftarrow \emptyset$
  $C = \{\emptyset\}_{i=1}^{N}$ ▷ k-best list
  for $t \in \{1 \ldots T\}$ **do**
    for $i \in \{1 \ldots N\}$ **do**
      $kbest^{(i)} \leftarrow \mathrm{GEN}(\boldsymbol{f}^{(i)}, \boldsymbol{w}^{(t-1)})$ ▷ decode using $\boldsymbol{w}^{(t-1)}$
      $\boldsymbol{c}^{(i)} \leftarrow \boldsymbol{c}^{(i)} \cup kbest^{(i)}$ ▷ merge k-best
    end for
    $\boldsymbol{w}^{(t)} \leftarrow \arg\min_{\boldsymbol{w} \in \mathcal{W}} \ell(F, E, C; \boldsymbol{w}) + \lambda \Omega(\boldsymbol{w})$ ▷ optimize
  end for
  return $\boldsymbol{w}^{(T)}$
end procedure

- k-best merging approach (Och and Ney, 2002)
- We can plug-in any loss + optimization algorithms

Maximum Entropy

$$\hat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|^2 - \sum_{s=1}^{S} \log \frac{\sum_{\mathbf{e}^* \in \mathsf{ORACLE}(\mathbf{f}_s)} \exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}^*, \mathbf{f}_s)\right)}{\sum_{\mathbf{e}' \in \mathsf{GEN}(\mathbf{f}_s)} \exp\left(\mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}', \mathbf{f}_s)\right)}$$

- Minimize negative conditional log-likelihood (Och and Ney, 2002)
- Derive ORACLE, a set of correct translation candidates, from GEN, a k-best list
- A standard optimization package: LBFGS, SGD
- Many sparse features

Why Not MaxEnt?

error criterion used in training	mWER [%]	mPER [%]	BLEU [%]	NIST	# words
confidence intervals	+/- 2.7	+/- 1.9	+/- 0.8	+/- 0.12	-
MMI	**68.0**	*51.0*	11.3	5.76	21933
mWER	*68.3*	*50.2*	13.5	6.28	22914
smoothed-mWER	*68.2*	*50.2*	13.2	6.27	22902
mPER	*70.2*	*49.8*	15.2	*6.71*	24399
smoothed-mPER	*70.0*	**49.7**	15.2	*6.69*	24198
BLEU	76.1	53.2	**17.2**	6.66	28002
NIST	73.3	*51.5*	*16.4*	**6.80**	26602

(Och, 2003)

- They select single oracle translation by WER(Och and Ney, 2002): This is difficult (non-decomposable to sentence-wise metric)
- summation problem: k-best (merging) approximation is not a true sample from the model (parameter)

All Derivations

System	Test (BLEU)
Discriminative max-derivation	25.78
Hiero (p_d, gr, rc, wc)	26.48
Discriminative max-translation	27.72
Hiero ($p_d, p_r, p_d^{lex}, p_r^{lex}, gr, rc, wc$)	28.14
Hiero ($p_d, p_r, p_d^{lex}, p_r^{lex}, gr, rc, wc, lm$)	32.00

(Blunsom et al., 2008)

- Blunsom et al. (2008): Optimized toward multiple derivations computed from a forest
- However, the correct translations are those exactly matched with reference translations (not computed by BLEU)

Ranking Approach

$$\hat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|^2 + \sum_{s=1}^{S}\sum_{\mathbf{e}''_s}\sum_{\mathbf{e}'_s}\xi_{s,\mathbf{e}''_s,\mathbf{e}'_s}$$

$$-\log\left(1+\exp(-\mathbf{w}^\top\cdot\Delta\mathbf{h}_{\mathbf{e}''_s,\mathbf{e}'_s})\right) \geq -\xi_{s,\mathbf{e}''_s,\mathbf{e}'_s}$$

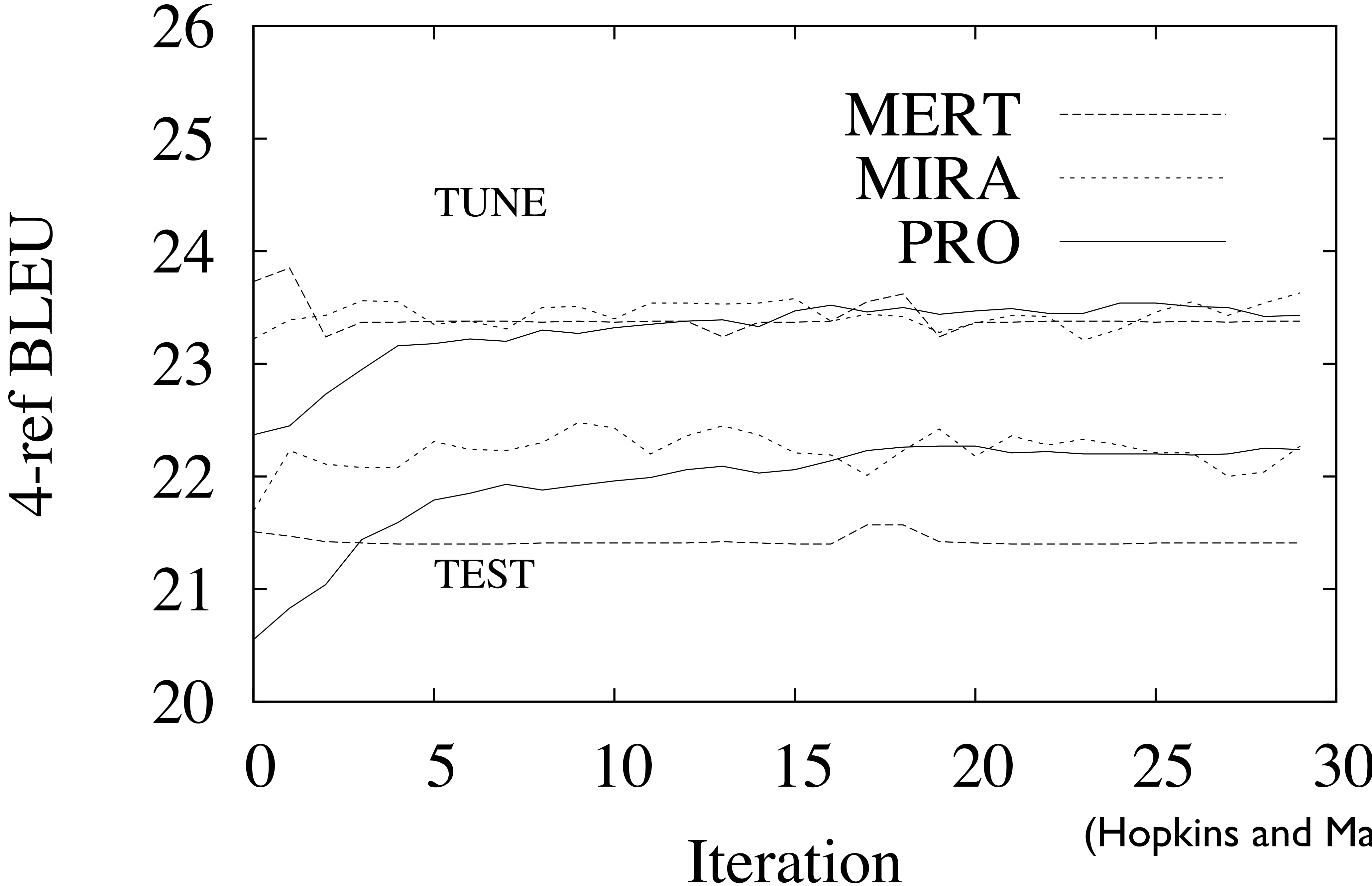
$$\mathbf{e}''_s, \mathbf{e}'_s \in \mathsf{GEN}(\mathbf{f}_s)$$

$$\ell(\mathbf{e}'_s, \mathbf{e}''_s) > 0$$

$$\Delta\mathbf{h}_{\mathbf{e}''_s,\mathbf{e}'_s} = \mathbf{h}(\mathbf{e}''_s, \mathbf{f}_s) - \mathbf{h}(\mathbf{e}'_s, \mathbf{f}_s)$$

- pair-wise comparison via (smoothed) sentence-BLEU + sampling (Hopkins and May, 2011)
- Use any binary classifier (here, logistic-loss) + linearly interpolated with parameters from previous iterations

Results



(Hopkins and May, 2011)

- Performed similarly with MIRA, MERT

Risk Minimization

$$\min_{\gamma, \mathbf{w}} \mathbb{E}_{p_{\gamma,\mathbf{w}}}[\ell(\mathbf{e}_s)] - T \cdot H(p_{\gamma,\mathbf{w}})$$

$$\begin{aligned}
\mathbb{E}_{p_{\gamma,\mathbf{w}}}[\ell(\mathbf{e}_s)] &= \sum_s \sum_i \ell(\mathbf{e}_s^i) p_{\gamma,\mathbf{w}}(\mathbf{e}_s^i | \mathbf{f}_s) \\
p_{\gamma,\mathbf{w}}(\mathbf{e}_s^i | \mathbf{f}_s) &= \frac{\exp\left(\gamma \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}_s^i, \mathbf{f}_s)\right)}{\sum_{i'} \exp\left(\gamma \mathbf{w}^\top \cdot \mathbf{h}(\mathbf{e}_s^{i'}, \mathbf{f}_s)\right)}
\end{aligned}$$

- smoothing by γ, regularization by entropy H(.), cooling by temperature T (Smith and Eisner, 2006)
- How to compute loss?: BLEU is non-linear!

Tailor series approximation

$$\log \text{Bleu} \approx \sum_{n=1}^{4} \frac{1}{4} \log \frac{c_n}{c_0} + \min\left(1 - \frac{r}{c_0}, 0\right)$$

$$\log \text{Bleu}' - \log \text{Bleu} \approx \sum_{n=0}^{4} (c'_n - c_n) \left. \frac{\partial \log \text{Bleu}'}{\partial c'_n} \right|_{c'_n = c_n}$$

$$= -\frac{c'_0 - c_0}{c_0} + \frac{1}{4} \sum_{n=1}^{4} \frac{c'_n - c_n}{c_n}$$

- Approximate the gain by BLEU by changing the statistics from c_n into c'_n (Tromble et al., 2008)
- Smith and Eisner (2006) approximated BLEU itself

Results

Training scheme	dev	test
MERT (Nbest, small)	42.6	47.7
MR (Nbest, small)	40.8	47.7
MR+DA (Nbest, small)	41.6	47.8
MR (hypergraph, small)	41.3	48.4
MR+DA (hypergraph, small)	41.9	48.3
MR (hypergraph, large)	42.3	**48.7**

(Li and Eisner, 2009)

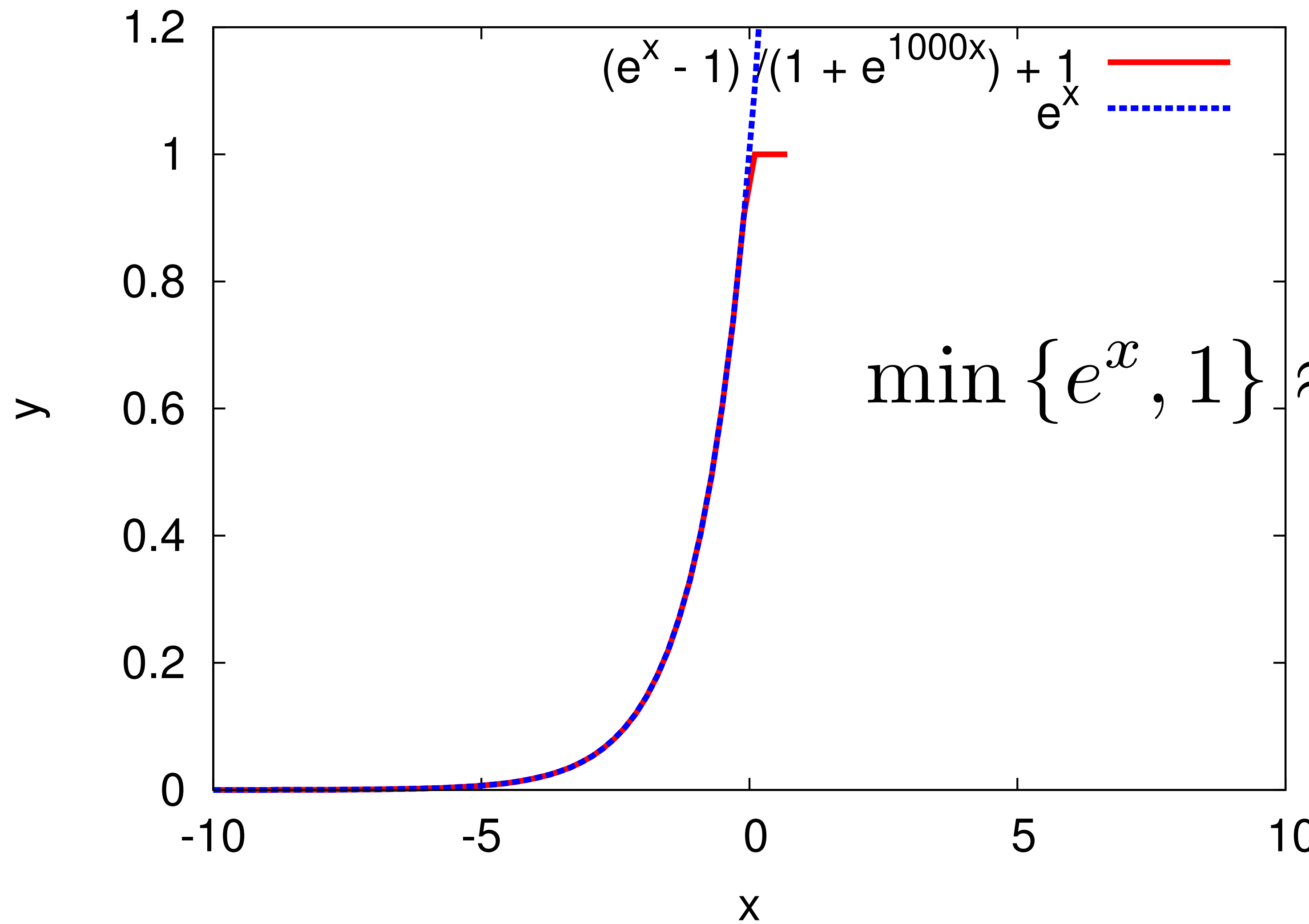
- Similar to MERT
- Better results by computing over hypergraph

Expected BLEU

$$\prod_{n=1}^{4} \left(\frac{\min\left\{\sum_s \sum_i \sum_{g_n \in \mathbf{e}_s^i} \mathbb{E}_{\gamma,\mathbf{w}}[c(g_n)], c^*(g_n)\right\}}{\sum_s \sum_i \sum_{g_n \in \mathbf{e}_s^i} \mathbb{E}_{\gamma,\mathbf{w}}[c(g_n)]} \right)^{\frac{1}{4}} \times \min\left\{\exp\left(1 - \frac{\sum_s r_s}{\sum_s \sum_i \sum_{g_1 \in \mathbf{e}_s^i} \mathbb{E}_{\gamma,\mathbf{w}}[c(g_1)]}\right), 1\right\}$$

- Maximize expected BLEU (Pauls et al., 2009; Rosti et al., 2010; Rosti et al., 2011)
  - compute BLEU from the expectation E[.] of ngram g_n
  - Similar to Smith and Eisner (2006)

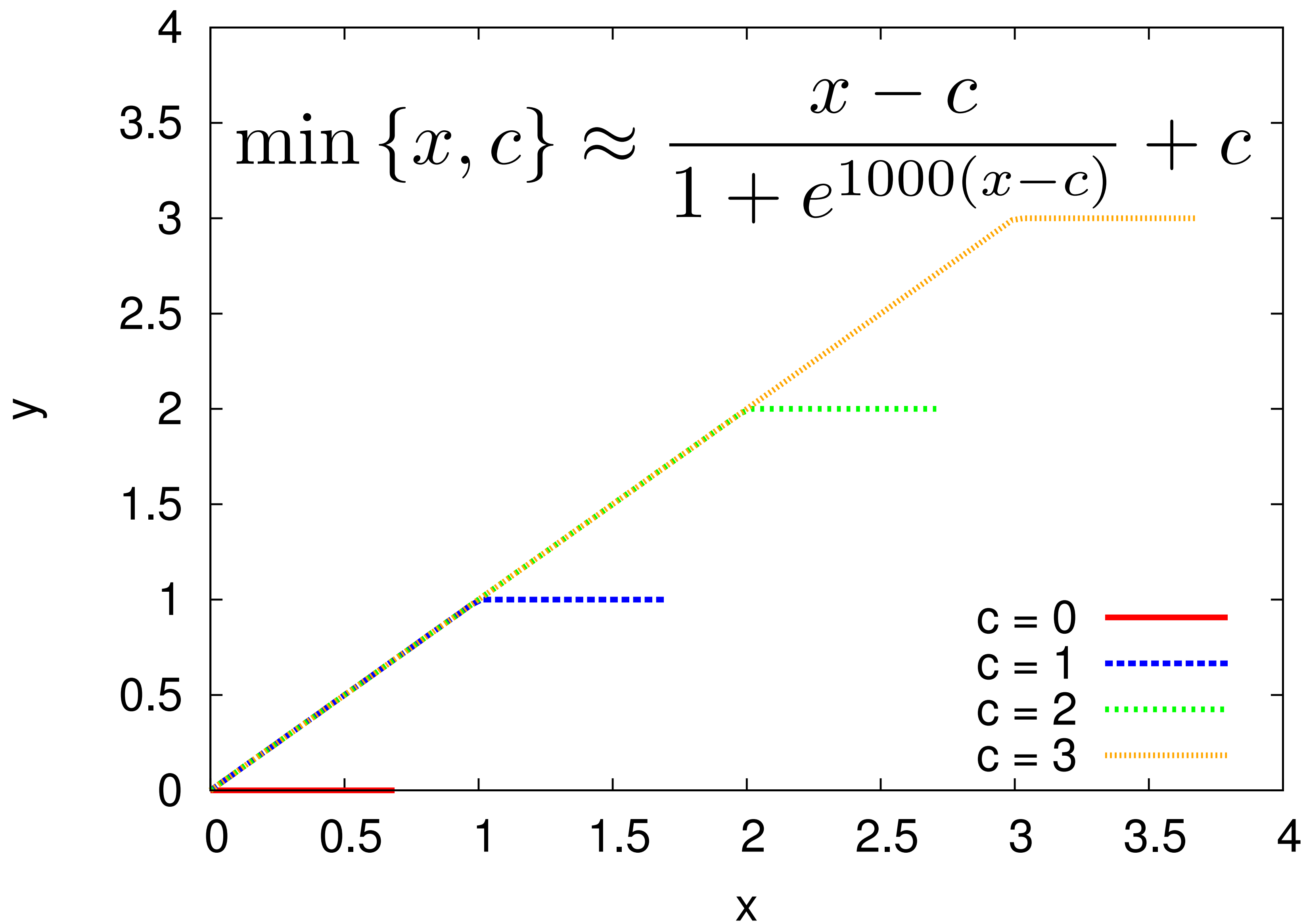
BP?



$$\min\{e^x, 1\} \approx \frac{e^x - 1}{1 + e^{1000x}} + 1$$

- They tried many alternatives by matlab(Rosti et al., 2010; Rosti et al., 2011)
- Ignore BP(Tromble et al., 2008)
- Ignore min(Pauls et al., 2009)

clip?



- Required for the expected BLUE over lattice/ forest(Rosti et al., 2011)
- NOTE: BUG in equation (15) of Rosti et al. (2011)

Results

test	cz-en		de-en		es-en		fr-en	
System	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU
worst	65.35	17.69	69.03	15.83	61.22	19.79	62.36	21.36
best	52.21	29.54	58.00	24.16	50.15	30.14	50.15	30.32
latBLEU	52.80	29.89	55.87	26.22	48.29	33.91	48.51	32.93
nbExpBLEU	52.97	29.93	55.77	26.52	48.39	33.86	48.25	32.94
latExpBLEU	52.68	29.99	55.74	26.62	48.30	34.10	48.17	32.91

- System combination result by optimization over lattice (Rosti et al., 2011)
- Efficient computation by expectation-semiring

Tuning

- Batch learning
- Online learning

Online Learning

procedure ONLINELEARN($\langle F, E\rangle = \left\{\langle \boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)}\rangle\right\}_{i=1}^{N}$)

  $\boldsymbol{w}^{(0)} \leftarrow \emptyset$

  $j \leftarrow 1$

  for $t \in \{1 \dots T\}$ **do**

    Choose $B_t = \{\boldsymbol{b}_1^{(t)}, \cdots \boldsymbol{b}_M^{(t)}\}$ ▷ randomly choose M batch

    for $\boldsymbol{b} \in B_t$ **do** ▷ $\boldsymbol{b} = \{\dots, \langle \boldsymbol{f}, \boldsymbol{e}\rangle, \dots\}$

      $\boldsymbol{c} \leftarrow \mathrm{GEN}(\boldsymbol{b}, \boldsymbol{w}^{(j-1)})$ ▷ decode using $\boldsymbol{w}^{(j-1)}$

      $\boldsymbol{w}^{(j)} \leftarrow \arg\min_{\boldsymbol{w} \in \mathcal{W}} \ell(\boldsymbol{b}, \boldsymbol{c}; \boldsymbol{w}) + \lambda\Omega(\boldsymbol{w})$ ▷ optimize

      $j \leftarrow j + 1$

    end for

  end for

  return $\boldsymbol{w}^{(T \cdot M)}$

end procedure

- Randomly split training data into M batches
- Decode sentences in a batch, and optimize (+ parallel training)

Online Large Margin

$$\hat{\mathbf{w}} = \operatorname*{argmin}_{\mathbf{w}'} \frac{\lambda}{2} \|\mathbf{w}' - \mathbf{w}\|^2 + \max\left(\ell_s - \mathbf{w}'^{\top} \cdot \Delta\mathbf{h}_s\right)$$

$$\begin{aligned} \hat{\mathbf{e}}_s &= \operatorname*{argmax}_{e} \mathbf{w}^{\top} \cdot \mathbf{h}(\mathbf{e}, \mathbf{f}_s) \\ \ell_s &= \ell(\hat{\mathbf{e}}_s) - \ell(\mathbf{e}^*_s) \\ \Delta\mathbf{h}_s &= \mathbf{h}(\hat{\mathbf{e}}_s, \mathbf{f}_s) - \mathbf{h}(\mathbf{e}^*, \mathbf{f}_s) \end{aligned}$$

- Optimize by MIRA(Crammer et al., 2006) (Watanabe et al., 2007; Chiang et al., 2008)
- Defined as an instance of structured Ramp loss minimization (Gimpel and Smith, 2012)
- How to compute BLUE?

Pseudo BLEU

$$\mathsf{GEN}(\mathbf{f}_s, \mathbf{w})$$

$$\mathbf{e}_1^*, \cdots, \begin{pmatrix} \mathbf{e}_s^1 \\ \vdots \\ \mathbf{e}_s^i \\ \vdots \\ \mathbf{e}_s^n \end{pmatrix}, \cdots, \mathbf{e}_S^*$$

- Memorize BLEU statistics for each sentence (1-best, or oracle candidate)
- Given a new k-best list, update the pseudo document statistics (Watanabe et al., 2007)

Decayed Pseudo BLEU

$$\begin{aligned} \mathbf{b} &\leftarrow 0.9 \times (\mathbf{b} + \mathbf{c}(\mathbf{e})) \\ l &\leftarrow 0.9 \times (l + |\mathbf{f}|) \end{aligned}$$

$$B(\mathbf{e}) = (l + |\mathbf{f}|) \times \mathsf{Bleu}(\mathbf{b} + \mathbf{c}(\mathbf{e}))$$

$$\begin{aligned} \hat{\mathbf{e}}_s &= \operatorname*{argmax}_{\mathbf{e}} -B(\mathbf{e}) + \hat{\mathbf{w}} \cdot \mathbf{h}(\mathbf{e}, \mathbf{f}_s) \\ \mathbf{e}^*_s &= \operatorname*{argmax}_{\mathbf{e}} +B(\mathbf{e}) + \hat{\mathbf{w}} \cdot \mathbf{h}(\mathbf{e}, \mathbf{f}_s) \end{aligned}$$

- Previously merged BLEU statistics are “decayed” (Chiang et al., 2008)
- argmax considering error counts

Results

System	Training	Features	#	Tune	Test
Hiero	MERT	baseline	11	35.4	36.1
	MIRA	syntax, distortion	56	35.9	36.9*
		syntax, distortion, discount	61	36.6	37.3**
		all source-side, discount	10990	38.4	37.6**
Syntax	MERT	baseline	25	38.6	39.5
	MIRA	baseline	25	38.5	39.8*
		overlap	132	38.7	39.9*
		node count	136	38.7	40.0**
		all target-side, discount	283	39.6	40.6**

(Chiang et al., 2009)

- statistically significant better results over the MERT baseline with feature engineering

Intricacy of BLEU

- Optimization for a sentence-wise BLEU $\neq$optimal for a document-wise BLEU
- BLEU on a larger batch: better document-wise BLEU estimates
- However, requiring more iterations
- Previous work: Pseudo-document, Decayed BLEU (Watanabe et al., 2007, Chiang et al., 2008)

SGD

$$\arg\min_{\mathbf{w}} \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \ell(\mathbf{w}; b)$$

$$\mathbf{w}_{k+\frac{1}{2}} \leftarrow (1 - \lambda\eta_k)\mathbf{w}_k + \sum_{(f,\mathbf{e})\in b, e^*, e'} \frac{\eta_k}{M(\mathbf{w}_k; b)} \Phi(f, e^*, e')$$

- Solve a "batch local" objective in each update
- When updating: set learning rate + update by a sub-gradient + projection into a L_2-ball
- hinge-loss: each loss-term is scaled by a constant

Optimized Update

$$\mathbf{w}_{k+\frac{1}{4}} \leftarrow (1 - \lambda\eta_k)\mathbf{w}_k$$

$$\arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{w}_{k+\frac{1}{4}}\|_2^2 + \eta_k \sum_{(f,\mathbf{e})\in b, e^*, e'} \xi_{f,e^*,e'}$$

$$\mathbf{w}^\top \Phi(f, e^*, e') \geq 1 - \xi_{f,e^*,e'}$$

$$\xi_{f,e^*,e'} \geq 0$$

- 2-step update: suffer sub-gradient from L_2 + solve a QP (Watanabe, 2012)
- Similar to MIRA: global L_2 + directly use the learning rate as a hyperparameter

Rescale Sub-Gradients

$$\mathbf{w}_{k+\frac{1}{2}} \leftarrow \mathbf{w}_{k+\frac{1}{4}} + \sum_{(f,\mathbf{e})\in b, e^*, e'} \tau_{e^*,e'} \Phi(f, e^*, e')$$

$$\sum_{(f,\mathbf{e})\in b, e^*, e'} \tau_{e^*,e'} \leq \eta_k$$

- We use Dual Coordinate Descent (Hsieh et al., 2008)
- If τ is set to η/M, then, we recover the original update formula

$$\mathbf{w}_{k+\frac{1}{2}} \leftarrow \mathbf{w}_{k+\frac{1}{4}} + \sum_{(f,\mathbf{e})\in b, e^*, e'} \frac{\eta_k}{M(\mathbf{w}_k; b)} \Phi(f, e^*, e')$$

Parallel Learning

$\mathbf{w}^1 \leftarrow \mathbf{0}$

for $t = 1, ..., T$ **do**

  $\mathbf{w}^{t,s} \leftarrow \mathbf{w}^t$

  Each shard learns $\mathbf{w}^{t+1,s}$ using D_s

  $\mathbf{w}^{t+1} \leftarrow 1/S \sum_s \mathbf{w}^{t+1,s}$

end for

return $\mathbf{w}^{T+1}$

- Split data into S shards (McDonald et al., 2010)
- Each shard learns locally
- Averaging in each round

Additional Line Search

$\mathbf{w}^1 \leftarrow \mathbf{0}$
for $t = 1, ..., T$ **do**
    $\mathbf{w}^{t,s} \leftarrow \mathbf{w}^t$
    Each shard learns $\mathbf{w}^{t+1,s}$ using D_s
    $\mathbf{w}^{t+\frac{1}{2}} \leftarrow 1/S \sum_s \mathbf{w}^{t+1,s}$
    $\mathbf{w}^{t+1} \leftarrow (1-\rho)\mathbf{w}^t + \rho\mathbf{w}^{t+\frac{1}{2}}$
end for
return $\mathbf{w}^{T+1}$

- Line search to determine ρ (Watanabe, 2012)
- The same as the procedure in MERT
  - Directly use document-BLEU as an objective using n-bests in each round

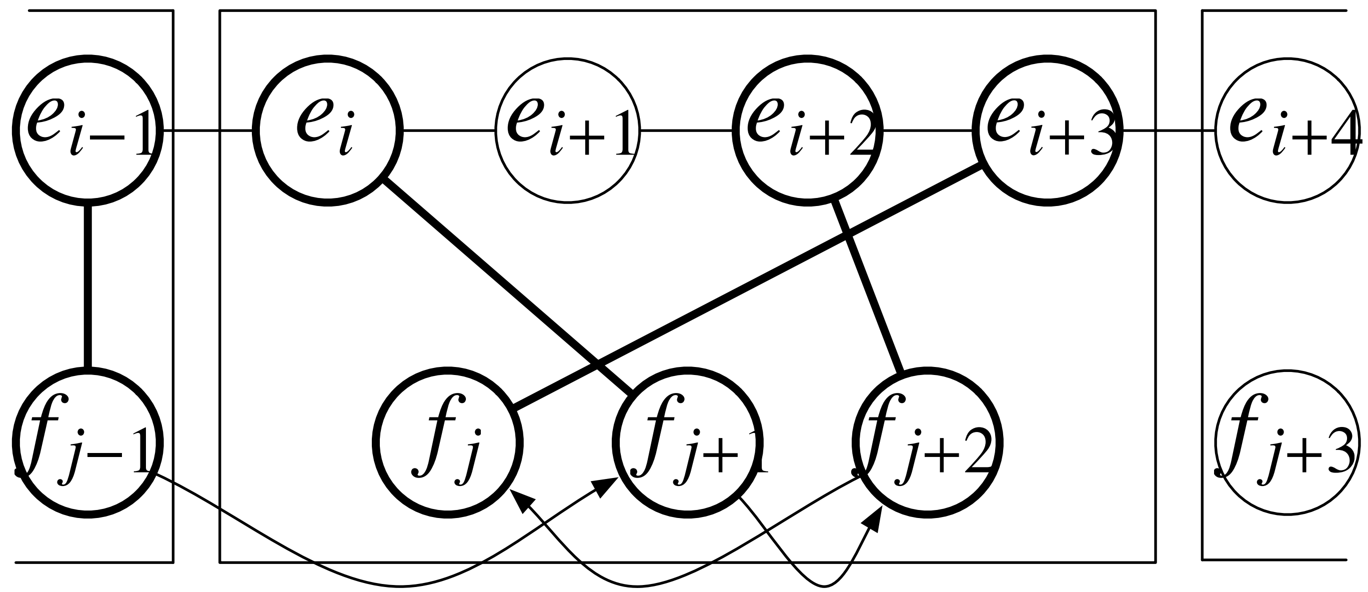
Experiments

	MT06	MT08
MERT	31.45†	24.13†
PRO	31.76†	24.43†
MIRA-L	31.42†	24.15†
ORO-L_{hinge}	29.76	21.96
O-ORO-L_{hinge}	**32.06**	**24.95**
ORO-$L_{softmax}$	30.77	23.07
O-ORO-$L_{softmax}$	31.16†	23.20

(Watanabe, 2009)

- NIST Chinese-to-English translation task
- Tune on MT02, development testing on MT06, testing on MT08

Feature Selections



- Watanabe et al. (2007) presented millions of feature approach
- However, pre-selecting features are better (Chiang et al., 2008, Chiang et al., 2009, Xiao et al., 2011)
- Any automatic way to select features?

Tuning as Multitask Learning

$$\operatorname*{arg\,min}_{\boldsymbol{w}^{(1)},\ldots,\boldsymbol{w}^{(N)}} \sum_{i=1}^{N} \ell(\boldsymbol{f}^{(i)}, \boldsymbol{e}^{(i)}, \boldsymbol{c}^{(i)}; \boldsymbol{w}^{(i)}) + \lambda\Omega(\boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(N)})$$

$$\begin{array}{r c ccccc c c | c ccccc c c}
 & & w_1 & w_2 & w_3 & w_4 & w_5 & & & & w_1 & w_2 & w_3 & w_4 & w_5 & & \\
\mathbf{w}_{z_1} & [& 6 & 4 & 0 & 0 & 0 &] & & [& 6 & 4 & 0 & 0 & 0 &] & \\
\mathbf{w}_{z_2} & [& 0 & 0 & 3 & 0 & 0 &] & & [& 3 & 0 & 0 & 0 & 0 &] & \\
\mathbf{w}_{z_3} & [& 0 & 0 & 0 & 2 & 3 &] & & [& 2 & 3 & 0 & 0 & 0 &] & \\
\text{column } \ell_2 \text{ norm:} & & 6 & 4 & 3 & 2 & 3 & & & & 7 & 5 & 0 & 0 & 0 & & \\
\ell_1 \text{ sum:} & & & & & & \Rightarrow & & 18 & & & & & & \Rightarrow & & 12
\end{array}$$

(Simianer et al., 2012)

- Separate w for each sentence and enforce agreement by a regularizer (Duh et al., 2010)
- l_1/l_2 regularization to derive "agreed features"

Features among Shards

$\mathbf{w}^1 \leftarrow \mathbf{0}$
for $t = 1, \ldots, T$ **do**
    $\mathbf{w}^{t,s} \leftarrow \mathbf{w}^t$
    Each shard learns $\mathbf{w}^{t+1,s}$ using D_s
    $\mathbf{W} = \left[\mathbf{w}^{t+1,1} | \ldots | \mathbf{w}^{t+1,S}\right]$
    Choose top K features by column-ℓ_2 norm of $\mathbf{W}$
    $\mathbf{w}^{t+1} \leftarrow 1/S \sum_s \mathbf{w}^{t+1,s}$
end for
return $\mathbf{w}^{T+1}$

- Compute column-l_2 of parameters among shards
- Keep only K-best features (Simianer et al., 2012)

Results

Algorithm	Tuning set	Features	#Features	devtest-*nc*	test-*nc*
MIRA	dev-*nc*	default	12	–	27.10
1	dev-*nc*	default	12	**25.88**	28.0
	dev-*nc*	+id	137k	25.53	27.6[†23]
	dev-*nc*	+ng	29k	25.82	27.42[†234]
	dev-*nc*	+shape	51	25.91	28.1
	dev-*nc*	+id,ng,shape	180k	25.71	**28.15**[34]
2	train-*nc*	default	12	25.73	27.86
	train-*nc*	+id	4.1M	25.13	27.19[†134]
	train-*nc*	+ng	354k	**26.09**	**28.03**[134]
	train-*nc*	+shape	51	26.07	27.91[3]
	train-*nc*	+id,ng,shape	4.7M	26.08	27.86[34]
3	train-*nc*	default	12	26.09 @2	27.94[†]
	train-*nc*	+id	3.4M	26.1 @4	27.97[†12]
	train-*nc*	+ng	330k	26.33 @4	28.34[12]
	train-*nc*	+shape	51	26.39 @9	28.31[2]
	train-*nc*	+id,ng,shape	4.7M	**26.42** @9	**28.55**[124]
4	train-*nc*	+id	100k	25.91 @7	27.82[†2]
	train-*nc*	+ng	100k	26.42 @4	28.37[†12]
	train-*nc*	+id,ng,shape	100k	**26.8** @8	**28.81**[123]

(Simianer et al., 2012)

Conclusion

- Batch/online training for tuning
- The "hidden variable" for MT is very large
  - translation error metric approximation
  - k-best merging approximation
  - online approximation

SMT2012

- Tutorial
  - Phrase-based MT
  - Tree-based MT
- Recent Topics
  - Phrase/rule induction
  - Tuning

Research on MT

- Reading: at least 50 papers are related to MT “every year”
- Solve a sub-problem as a specialist
- Keep the whole picture

References

Arun, A., Dyer, C., Haddow, B., Blunsom, P., Lopez, A., & Koehn, P. (2009, June). Monte carlo inference and maximization for phrase-based translation. In *Proc. of conll-2009* (pp. 102--110). Boulder, Colorado. Retrieved from http://www.aclweb.org/anthology/W09-1114

Blunsom, P., Cohn, T., Dyer, C., & Osborne, M. (2009, August). A gibbs sampler for phrasal synchronous grammar induction. In *Proc. of acl/ijcnlp 2009* (pp. 782--790). Suntec, Singapore: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P/P09/P09-1088

Blunsom, P., Cohn, T., & Osborne, M. (2008, June). A discriminative latent variable model for statistical machine translation. In *Proc. of acl-08: Hlt* (pp. 200--208). Columbus, Ohio. Retrieved from http://www.aclweb.org/anthology/P/P08/P08-1024

Brants, T., Popat, A. C., Xu, P., Och, F. J., & Dean, J. (2007, June). Large language models in machine translation. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (emnlp-conll)* (pp. 858--867). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D/D07/D07-1090

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., … Roossin, P. S. (1990, June). A statistical approach to machine translation. *Computational Linguistics*, *16*(2), 79--85. Retrieved from http://dl.acm.org/citation.cfm?id=92858.92860

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., & Mercer, R. L. (1993, June). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, *19*(2), 263--311. Retrieved from http://dl.acm.org/citation.cfm?id=972470.972474

Cer, D., Jurafsky, D., & Manning, C. D. (2008, June). Regularization and search for minimum error rate training. In *Proc. of smt 2008* (pp. 26--34). Columbus, Ohio. Retrieved from http://www.aclweb.org/anthology/W/W08/W08-0304

Cherry, C., & Lin, D. (2007, April). Inversion transduction grammar for joint phrasal translation modeling. In *Proc. of ssst 2007* (pp. 17--24). Rochester, New York. Retrieved from http://

www.aclweb.org/anthology/W/W07/W07-0403
Cherry, C., Moore, R. C., & Quirk, C. (2012, June). On hierarchical re-ordering and permutation parsing for phrase-based decoding. In *Proceedings of the seventh workshop on statistical machine translation* (pp. 200--209). Montréal, Canada: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W12-3125
Chiang, D. (2007, June). Hierarchical phrase-based translation. *Computational Linguistics*, *33*(2), 201--228. Retrieved from http://dx.doi.org/10.1162/coli.2007.33.2.201 doi: 10.1162/coli.2007.33.2.201
Chiang, D. (2010, July). Learning to translate with source and target syntax. In *Proc. of acl 2010* (pp. 1443--1452). Uppsala, Sweden. Retrieved from http://www.aclweb.org/anthology/P10-1146
Chiang, D., Knight, K., & Wang, W. (2009, June). 11,001 new features for statistical machine translation. In *Proc. of naacl-hlt 2009* (pp. 218--226). Boulder, Colorado. Retrieved from http://www.aclweb.org/anthology/N/N09/N09-1025
Chiang, D., Marton, Y., & Resnik, P. (2008, October). Online large-margin training of syntactic and structural translation features. In *Proc. of emnlp 2008* (pp. 224--233). Honolulu, Hawaii. Retrieved from http://www.aclweb.org/anthology/D08-1024
Clark, J. H., Dyer, C., Lavie, A., & Smith, N. A. (2011, June). Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proc. of acl 2011* (pp. 176--181). Portland, Oregon, USA.
Cohn, T., & Blunsom, P. (2009, August). A Bayesian model of syntax-directed tree to string grammar induction. In *Proc. of emnlp 2009* (pp. 352--361). Singapore. Retrieved from http://www.aclweb.org/anthology/D/D09/D09-1037
Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006, March). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7, 551--585.
DeNero, J., Bouchard-Côté, A., & Klein, D. (2008, October). Sampling alignment structure under a Bayesian translation model. In *Proc. of emnlp 2008* (pp. 314--323). Honolulu, Hawaii: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D08-1033

Deschacht, K., & Moens, M.-F. (2009, August). Semi-supervised semantic role labeling using the Latent Words Language Model. In *Proceedings of the 2009 conference on empirical methods in natural language processing* (pp. 21--29). Singapore: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/D/D09/D09-1003`

Duh, K., Sudoh, K., Tsukada, H., Isozaki, H., & Nagata, M. (2010, July). N-best reranking by multitask learning. In *Proceedings of the joint fifth workshop on statistical machine translation and metricsmatr* (pp. 375--383). Uppsala, Sweden: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/W10-1757`

Feng, Y., Liu, Y., Liu, Q., & Cohn, T. (2012, July). Left-to-right tree-to-string decoding with prediction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning* (pp. 1191--1200). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/D12-1109`

Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W., & Thayer, I. (2006, July). Scalable inference and training of context-rich syntactic translation models. In *Proc. of acl/coling 2006* (pp. 961--968). Sydney, Australia. Retrieved from `http://www.aclweb.org/anthology/P06-1121` doi: 10.3115/1220175.1220296

Galley, M., Hopkins, M., Knight, K., & Marcu, D. (2004, May 2 - May 7). What's in a translation rule? In D. M. Susan Dumais & S. Roukos (Eds.), *Proc. of hlt-naacl 2004* (pp. 273--280). Boston, Massachusetts, USA: Association for Computational Linguistics.

Galley, M., & Quirk, C. (2011, July). Optimal search for minimum error rate training. In *Proc. of emnlp 2011* (pp. 38--49). Edinburgh, Scotland, UK.. Retrieved from `http://www.aclweb.org/anthology/D11-1004`

Gesmundo, A., & Henderson, J. (2010). Faster Cube Pruning. In *Proc. of iwslt 2010* (pp. 267--274).

Ghodke, S., Bird, S., & Zhang, R. (2011, November). A breadth-first representation for tree matching in large scale forest-based translation. In *Proceedings of 5th international joint conference on natural language processing* (pp. 785--793). Chiang Mai, Thailand: Asian Federation of Natural Language Processing. Retrieved from `http://www.aclweb.org/anthology/I11-1088`

Gimpel, K., & Smith, N. A. (2012, June). Structured ramp loss minimization for machine translation.

In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 221--231). Montréal, Canada: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/N12-1023

Haghighi, A., Blitzer, J., DeNero, J., & Klein, D. (2009, August). Better word alignments with supervised itg models. In *Proc. of acl/ijcnlp 2009* (pp. 923--931). Suntec, Singapore. Retrieved from http://www.aclweb.org/anthology/P/P09/P09-1104

Hayashi, K., Watanabe, T., Tsukada, H., & Isozaki, H. (2009). Structural Support Vector Machines for Log-Linear Approach in Statistical Machine Translation. In *Proc. of iwslt 2009* (p. 144-151). Tokyo, Japan.

Hopkins, M., & May, J. (2011, July). Tuning as ranking. In *Proc. of emnlp 2011* (pp. 1352--1362). Edinburgh, Scotland, UK.: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D11-1125

Hsieh, C.-J., Chang, K.-W., Lin, C.-J., Keerthi, S. S., & Sundararajan, S. (2008). A dual coordinate descent method for large-scale linear svm. In *Proc. of icml '08* (pp. 408--415). Helsinki, Finland. Retrieved from http://doi.acm.org/10.1145/1390156.1390208 doi: http://doi.acm.org/10.1145/1390156.1390208

Huang, L., & Chiang, D. (2005, October). Better k-best parsing. In *Proc. of iwpt'05* (pp. 53--64). Vancouver, British Columbia. Retrieved from http://www.aclweb.org/anthology/W/W05/W05-1506

Huang, L., & Chiang, D. (2007, June). Forest rescoring: Faster decoding with integrated language models. In *Proc. of acl 2007* (pp. 144--151). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P07-1019

Huang, L., Knight, K., & Joshi, A. (2006). Statistical syntax-directed translation with extended domain of locality. In *In proc. amta 2006* (pp. 66--73).

Huang, L., & Mi, H. (2010, October). Efficient incremental decoding for tree-to-string translation. In *Proc. of emnlp 2010* (pp. 273--283). Cambridge, MA. Retrieved from http://www.aclweb.org/anthology/D10-1027

Klein, D., & Manning, C. D. (2001). Parsing and hypergraphs. In *Proc. of iwpt-2001* (pp. 123--134).

Kneser, R., & Ney, H. (1995, May). Improved backing-off for m-gram language modeling. In *In proceedings of the ieee international conference on acoustics, speech and signal processing* (Vol. I, p. 181-184). Detroit, Michigan.

Knight, K. (1999, December). Decoding complexity in word-replacement translation models. *Computational Linguistics*, *25*, 607--615. Retrieved from `http://portal.acm.org/citation.cfm?id=973226.973232`

Koehn, P., Och, F. J., & Marcu, D. (2003, May-June). Statistical pharse-based translation. In *Proc. of HLT-NAACL 2003* (pp. 48--54). Edmonton.

Levenberg, A., Dyer, C., & Blunsom, P. (2012, July). A bayesian model for learning scfgs with discontiguous rules. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning* (pp. 223--232). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/D12-1021`

Li, Z., & Eisner, J. (2009, August). First- and second-order expectation semirings with applications to minimum-risk training on translation forests. In *Proc. of emnlp 2009* (pp. 40--51). Singapore. Retrieved from `http://www.aclweb.org/anthology/D/D09/D09-1005`

Macherey, W., Och, F., Thayer, I., & Uszkoreit, J. (2008, October). Lattice-based minimum error rate training for statistical machine translation. In *Proc. of emnlp 2008* (pp. 725--734). Honolulu, Hawaii. Retrieved from `http://www.aclweb.org/anthology/D08-1076`

Marcu, D., & Wong, D. (2002, July). A phrase-based,joint probability model for statistical machine translation. In *Proceedings of the 2002 conference on empirical methods in natural language processing* (pp. 133--139). Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/W02-1018` doi: 10.3115/1118693.1118711

McDonald, R., Hall, K., & Mann, G. (2010, June). Distributed training strategies for the structured perceptron. In *Proc. of naacl-hlt 2010* (pp. 456--464). Los Angeles, California.

Mi, H., & Huang, L. (2008, October). Forest-based translation rule extraction. In *Proc. of emnlp 2008* (pp. 206--214). Honolulu, Hawaii. Retrieved from `http://www.aclweb.org/anthology/D08-1022`

Mi, H., Huang, L., & Liu, Q. (2008, June). Forest-based translation. In *Proc. of acl-08: Hlt* (pp. 192--199).

Columbus, Ohio. Retrieved from http://www.aclweb.org/anthology/P/P08/P08-1023
Mi, H., & Liu, Q. (2010, July). Constituency to dependency translation with forests. In *Proc. of acl 2010* (pp. 1433--1442). Uppsala, Sweden. Retrieved from http://www.aclweb.org/anthology/P10-1145
Mochihashi, D., Yamada, T., & Ueda, N. (2009, August). Bayesian unsupervised word segmentation with nested pitman-yor language modeling. In *Proceedings of the joint conference of the 47th annual meeting of the acl and the 4th international joint conference on natural language processing of the afnlp* (pp. 100--108). Suntec, Singapore: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P/P09/P09-1012
Moore, R. C., & Quirk, C. (2008, August). Random restarts in minimum error rate training for statistical machine translation. In *Proc. of coling 2008* (pp. 585--592). Manchester, UK. Retrieved from http://www.aclweb.org/anthology/C08-1074
Neubig, G., Watanabe, T., Mori, S., & Kawahara, T. (2012, July). Machine translation without words through substring alignment. In *Proceedings of the 50th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 165--174). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P12-1018
Neubig, G., Watanabe, T., Sumita, E., Mori, S., & Kawahara, T. (2011, June). An unsupervised model for joint phrase alignment and extraction. In *Proc. of acl-hlt 2011* (pp. 632--641). Portland, Oregon, USA: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P11-1064
Och, F. J. (2003, July). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st annual meeting of the association for computational linguistics* (pp. 160--167). Sapporo, Japan: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P03-1021 doi: 10.3115/1075096.1075117
Och, F. J., & Ney, H. (2002, July). Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of acl 2002* (pp. 295--302). Philadelphia, Pennsylvania, USA. Retrieved from http://www.aclweb.org/anthology/P02-1038 doi: 10.3115/1073083.1073133
Och, F. J., & Ney, H. (2004, December). The alignment template approach to statistical machine

translation. *Computational Linguistics*, *30*(4), 417--449. Retrieved from http://dx.doi.org/10.1162/0891201042544884 doi: 10.1162/0891201042544884
Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002, July). Bleu: a method for automatic evaluation of machine translation. In *Proc. of acl 2002* (pp. 311--318). Philadelphia, Pennsylvania, USA: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P02-1040 doi: 10.3115/1073083.1073135
Pauls, A., Denero, J., & Klein, D. (2009, August). Consensus training for consensus decoding in machine translation. In *Proc. of emnlp 2009* (pp. 1418--1427). Singapore. Retrieved from http://www.aclweb.org/anthology/D/D09/D09-1147
Rosti, A.-V., Zhang, B., Matsoukas, S., & Schwartz, R. (2010, July). Bbn system description for wmt10 system combination task. In *Proceedings of the joint fifth workshop on statistical machine translation and metricsmatr* (pp. 321--326). Uppsala, Sweden: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W10-1748
Rosti, A.-V., Zhang, B., Matsoukas, S., & Schwartz, R. (2011, July). Expected bleu training for graphs: Bbn system description for wmt11 system combination task. In *Proceedings of the sixth workshop on statistical machine translation* (pp. 159--165). Edinburgh, Scotland: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W11-2119
Saers, M., Nivre, J., & Wu, D. (2009, October). Learning stochastic bracketing inversion transduction grammars with a cubic time biparsing algorithm. In *Proc. of iwpt'09* (pp. 29--32). Paris, France. Retrieved from http://www.aclweb.org/anthology/W09-3804
Shieber, S. M., Schabes, Y., & Pereira, F. C. N. (1995, July--August). Principles and implementation of deductive parsing. *Journal of Logic Programming*, *24*(1--2), 3--36.
Simianer, P., Riezler, S., & Dyer, C. (2012, July). Joint feature selection in distributed stochastic learning for large-scale discriminative training in smt. In *Proceedings of the 50th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 11--21). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P12-1002
Smith, D. A., & Eisner, J. (2006, July). Minimum risk annealing for training log-linear models. In *Proc. of the coling/acl 2006* (pp. 787--794). Sydney, Australia. Retrieved from http://www.aclweb.org/

anthology/P/P06/P06-2101

Teh, Y. W. (2006a). *A bayesian interpretation of interpolated kneser-ney* (Tech. Rep.). School of Computing, National University of Singapore. (Technical Report TRA2/06)

Teh, Y. W. (2006b, July). A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the association for computational linguistics* (pp. 985--992). Sydney, Australia: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P06-1124 doi: 10.3115/1220175.1220299

Tromble, R., Kumar, S., Och, F., & Macherey, W. (2008, October). Lattice Minimum Bayes-Risk decoding for statistical machine translation. In *Proc. of emnlp 2008* (pp. 620--629). Honolulu, Hawaii. Retrieved from http://www.aclweb.org/anthology/D08-1065

Watanabe, T. (2012, June). Optimized online rank learning for machine translation. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 253--262). Montréal, Canada: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/N12-1026

Watanabe, T., Suzuki, J., Tsukada, H., & Isozaki, H. (2007, June). Online Large-Margin Training for Statistical Machine Translation. In *Proc. of emnlp-conll 2007* (pp. 764--773). Prague, Czech Republic.

Watanabe, T., Tsukada, H., & Isozaki, H. (2006, July). Left-to-Right Target Generation for Hierarchical Phrase-Based Translation. In *Proc. of acl/coling 2006* (pp. 777--784). Sydney, Australia. Retrieved from http://www.aclweb.org/anthology/P06-1098 doi: 10.3115/1220175.1220273

Wu, D. (1997, September). Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, *23*(3), 377--403. Retrieved from http://dl.acm.org/citation.cfm?id=972705.972707

Xiao, X., Liu, Y., Liu, Q., & Lin, S. (2011, July). Fast generation of translation forest for large-scale smt discriminative training. In *Proceedings of the 2011 conference on empirical methods in natural language processing* (pp. 880--888). Edinburgh, Scotland, UK.: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D11-1081

Xie, J., Mi, H., & Liu, Q. (2011, July). A novel dependency-to-string model for statistical machine

translation. In *Proc. of emnlp 2011* (pp. 216--226). Edinburgh, Scotland, UK.. Retrieved from http://www.aclweb.org/anthology/D11-1020
Zens, R., & Ney, H. (2003, July). A comparative study on reordering constraints in statistical machine translation. In *Proceedings of the 41st annual meeting of the association for computational linguistics* (pp. 144--151). Sapporo, Japan: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P03-1019 doi: 10.3115/1075096.1075115
Zens, R., Ney, H., Watanabe, T., & Sumita, E. (2004, Aug 23--Aug 27). Reordering Constraints for Phrase-Based Statistical Machine Translation. In *Proc. of COLING 2004* (pp. 205--211). Geneva, Switzerland.
Zhang, H., Fang, L., Xu, P., & Wu, X. (2011, June). Binarized forest to string translation. In *Proc. of acl-hlt 2011* (pp. 835--845). Portland, Oregon, USA. Retrieved from http://www.aclweb.org/anthology/P11-1084
Zhang, H., & Gildea, D. (2005, June). Stochastic lexicalized inversion transduction grammar for alignment. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)* (pp. 475--482). Ann Arbor, Michigan: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P05-1059 doi: 10.3115/1219840.1219899
Zhang, H., Quirk, C., Moore, R. C., & Gildea, D. (2008, June). Bayesian learning of non-compositional phrases with synchronous parsing. In *Proceedings of acl-08: Hlt* (pp. 97--105). Columbus, Ohio: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/P/P08/P08-1012
Zhang, H., Zhang, M., Li, H., & Tan, C. L. (2009, August). Fast translation rule matching for syntax-based statistical machine translation. In *Proceedings of the 2009 conference on empirical methods in natural language processing* (pp. 1037--1045). Singapore: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D/D09/D09-1108
Zollmann, A., & Venugopal, A. (2006, June). Syntax augmented machine translation via chart parsing. In *Proceedings on the workshop on statistical machine translation* (pp. 138--141). New York City: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W/W06/W06-3119