MT System Combination by Confusion Forest

Taro Watanabe and Eiichiro Sumita
NICT

MT System Combination

MT System Combination

- Better translation by combining multiple system outputs:
  - Sentence selection(Nomoto, 2004; etc.)
  - Phrasal combination (Frederking and Nirenburg, 1994; etc.)
  - Word level combination (Bangalore et al., 2001; Matusov et al., 2006; etc.)

MT System Combination

- Better translation by combining multiple system outputs:
  - Sentence selection(Nomoto, 2004; etc.)
  - Phrasal combination (Frederking and Nirenburg, 1994; etc.)
  - Word level combination (Bangalore et al., 2001; Matusov et al., 2006; etc.)
- This Work: Syntactic combination, not word-wise combination

Confusion Network

Confusion Network

I	saw	the	forest	
I	walked	the	blue	forest
I	saw	the	green	trees
the	forest	was	found	

Confusion Network

I	saw	the	forest	
I	walked	the	blue	forest
I	saw	the	green	trees
the	forest	was	found	

- State-of-the-art: Confusion Network
- Choose a skeleton, compute word alignment against the skeleton
  - Edit-distance-based alignment (TER etc.) (Sim et al., 2007)
  - Model-based alignment(GIZA++ etc.) (Matsov et al., 2006)

Confusion Network

★	I	saw	the	forest	
	I	walked	the	blue	forest
	I	saw	the	green	trees
	the	forest	was	found	

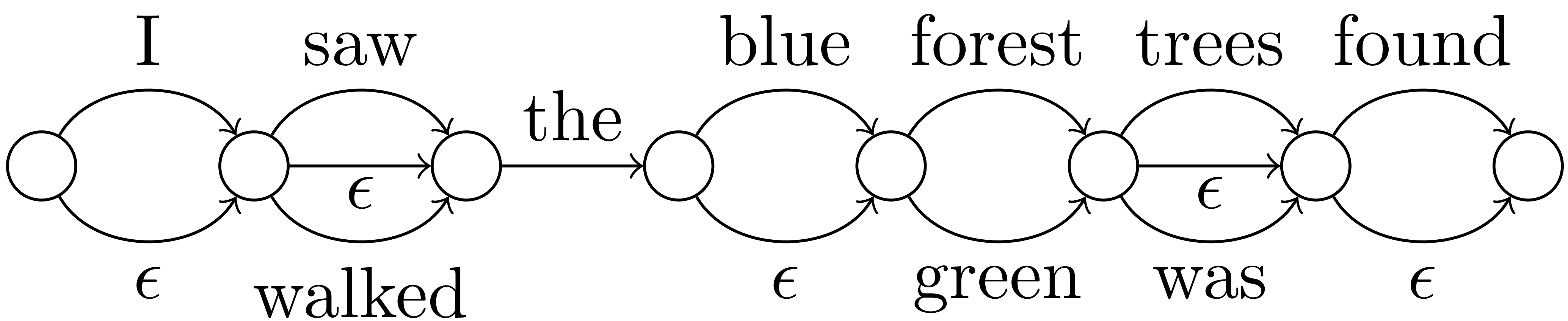
- State-of-the-art: Confusion Network
- Choose a skeleton, compute word alignment against the skeleton
  - Edit-distance-based alignment (TER etc.) (Sim et al., 2007)
  - Model-based alignment(GIZA++ etc.) (Matsov et al., 2006)

Confusion Network

★	I	saw	the		forest		
	I	walked	the	blue	forest		
	I	saw	the		green	trees	
			the		forest	was	found

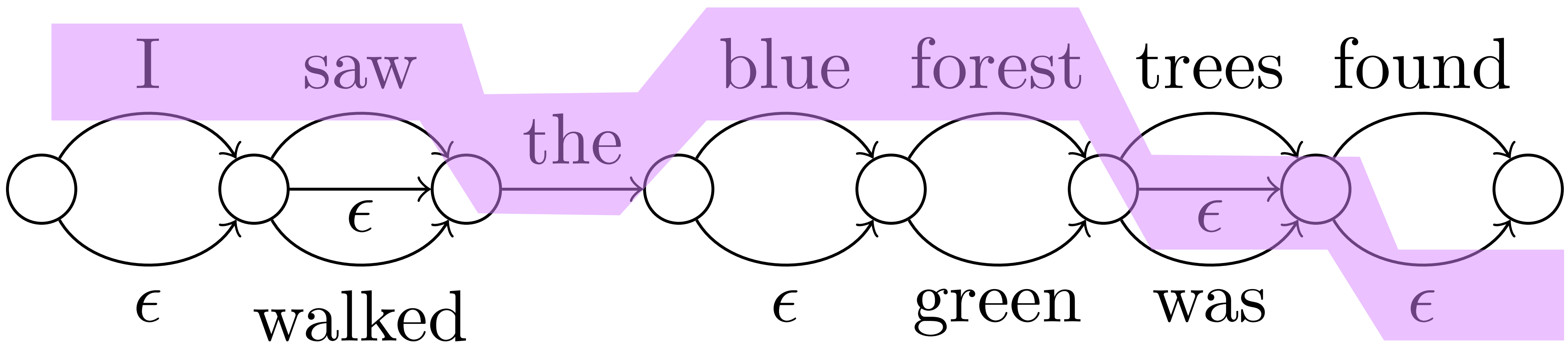
- State-of-the-art: Confusion Network
- Choose a skeleton, compute word alignment against the skeleton
  - Edit-distance-based alignment (TER etc.) (Sim et al., 2007)
  - Model-based alignment(GIZA++ etc.) (Matsov et al., 2006)

Confusion Network



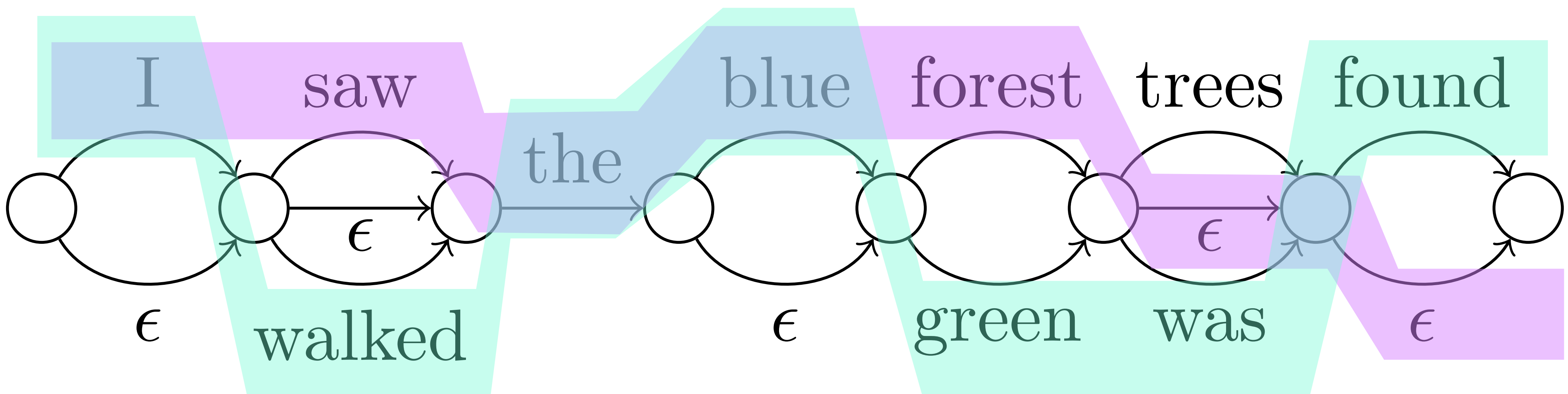
- Construct a network with each arc representing alternative translation
  - Best path = Best translation
  - Syntactically different language pairs: i.e. active/passive voices
  - Spurious insertion/repetition due to alignment error
  - Incremental alignment/construction + merge multiple networks into one (Rosti et al., 2008)

Confusion Network



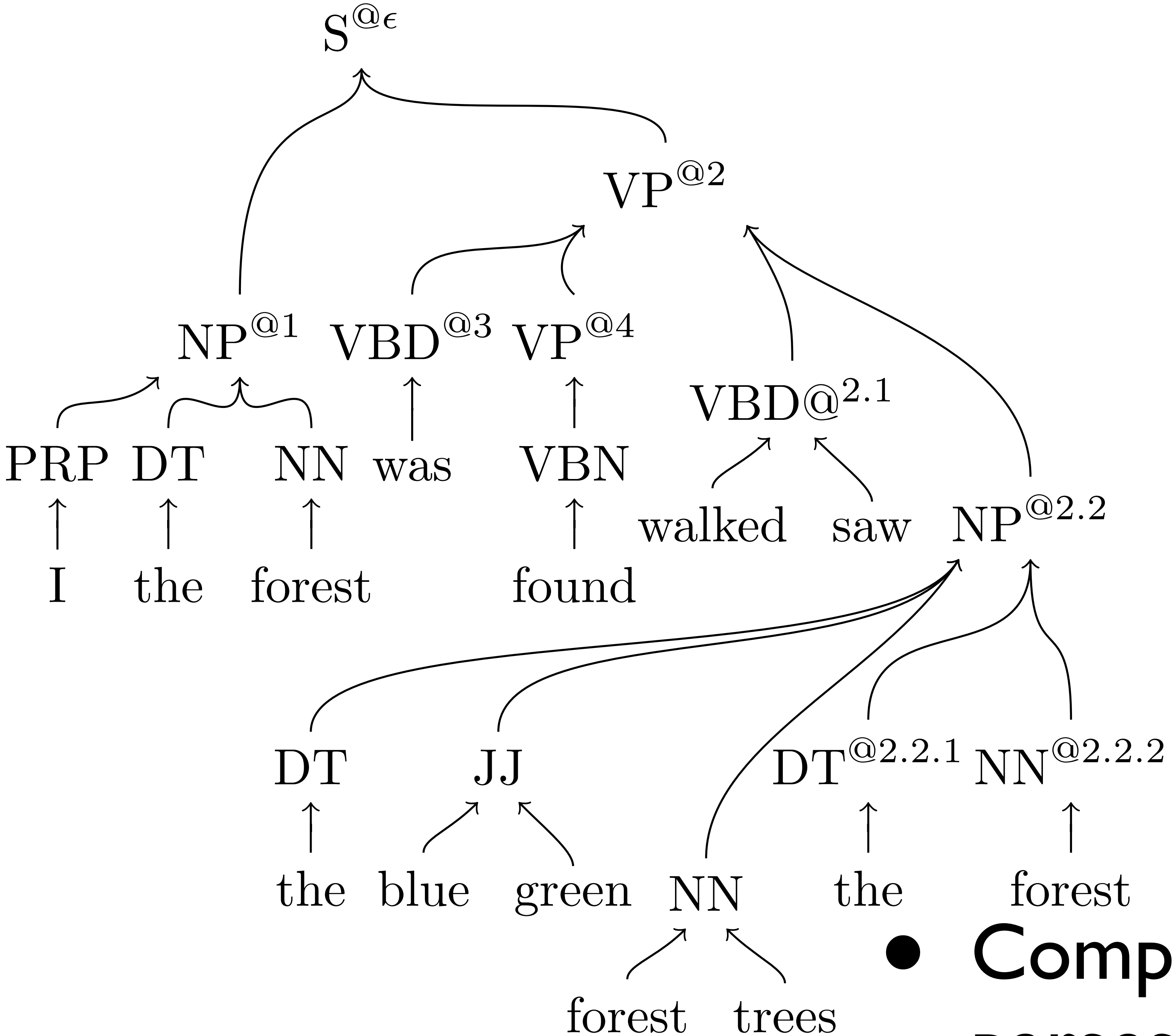
- Construct a network with each arc representing alternative translation
  - Best path = Best translation
  - Syntactically different language pairs: i.e. active/passive voices
  - Spurious insertion/repetition due to alignment error
  - Incremental alignment/construction + merge multiple networks into one (Rosti et al., 2008)

Confusion Network



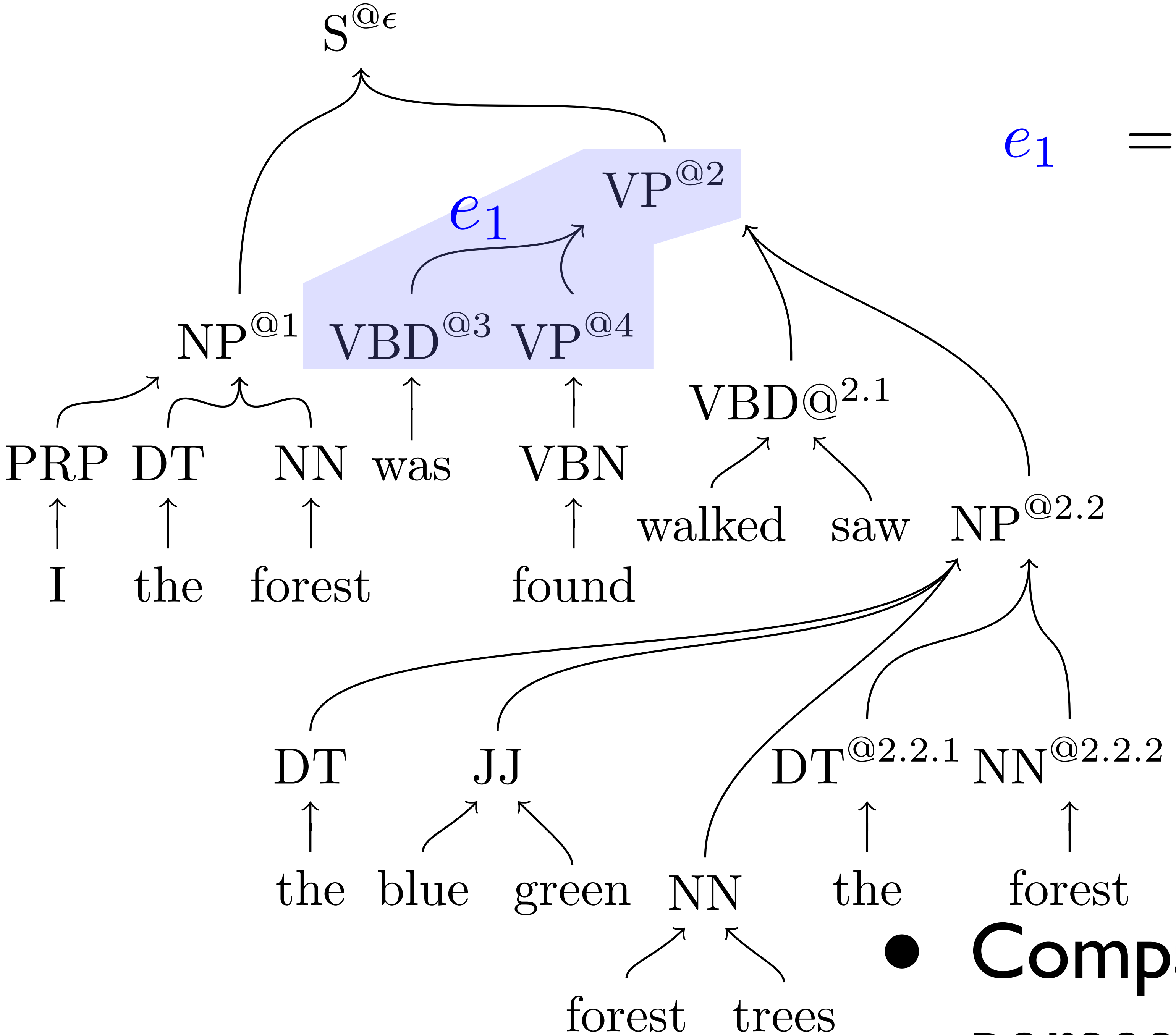
- Construct a network with each arc representing alternative translation
  - Best path = Best translation
  - Syntactically different language pairs: i.e. active/passive voices
  - Spurious insertion/repetition due to alignment error
  - Incremental alignment/construction + merge multiple networks into one (Rosti et al., 2008)

Confusion Forest



- Compactly represent multiple parses by sharing nodes
- Represented by "hypergraph"

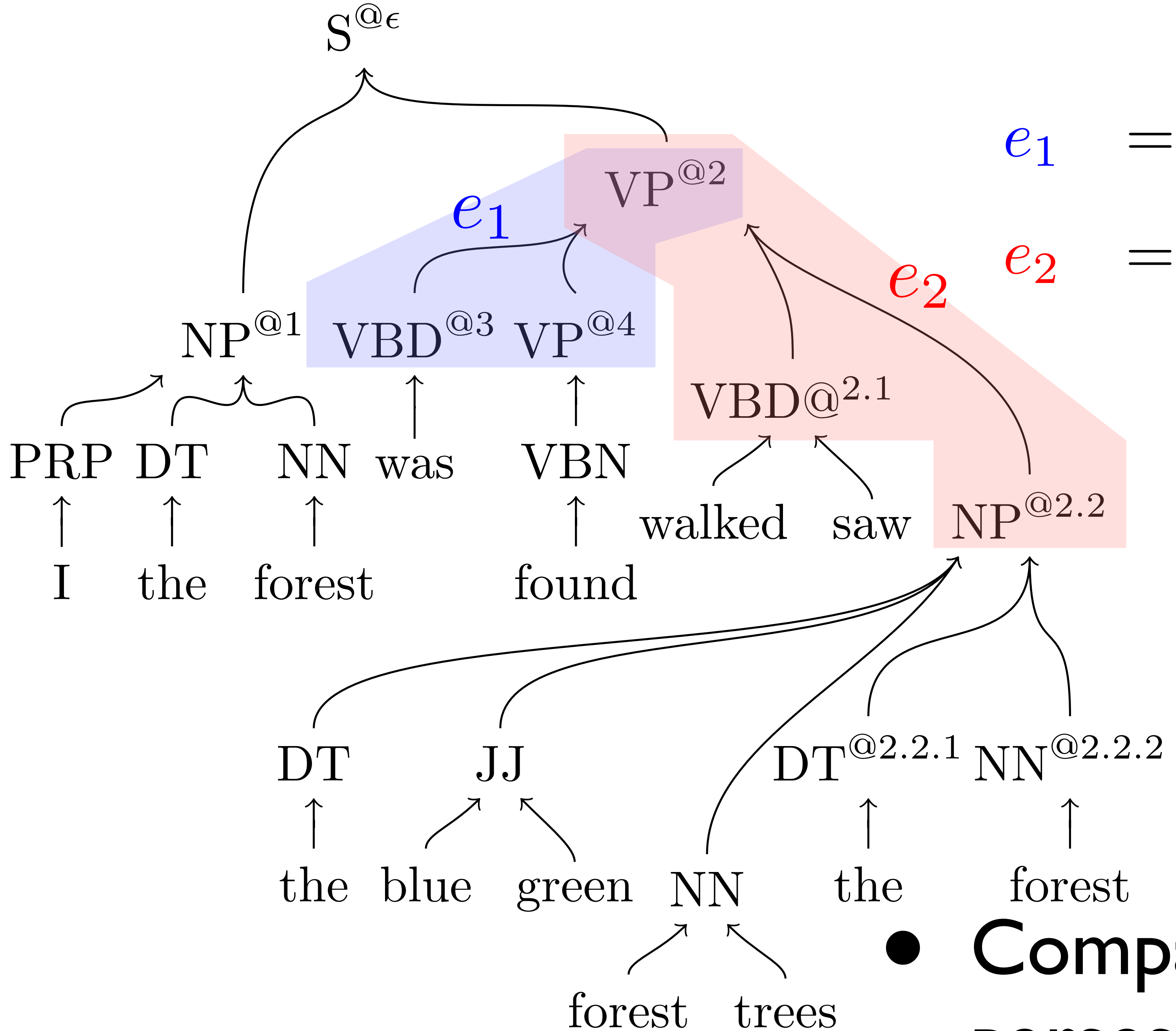
Confusion Forest



$$e_1 = \langle \mathrm{VP}^{@2}, \left\{ \mathrm{VBD}^{@3}, \mathrm{VP}^{@4} \right\} \rangle$$

- Compactly represent multiple parses by sharing nodes
- Represented by "hypergraph"

Confusion Forest

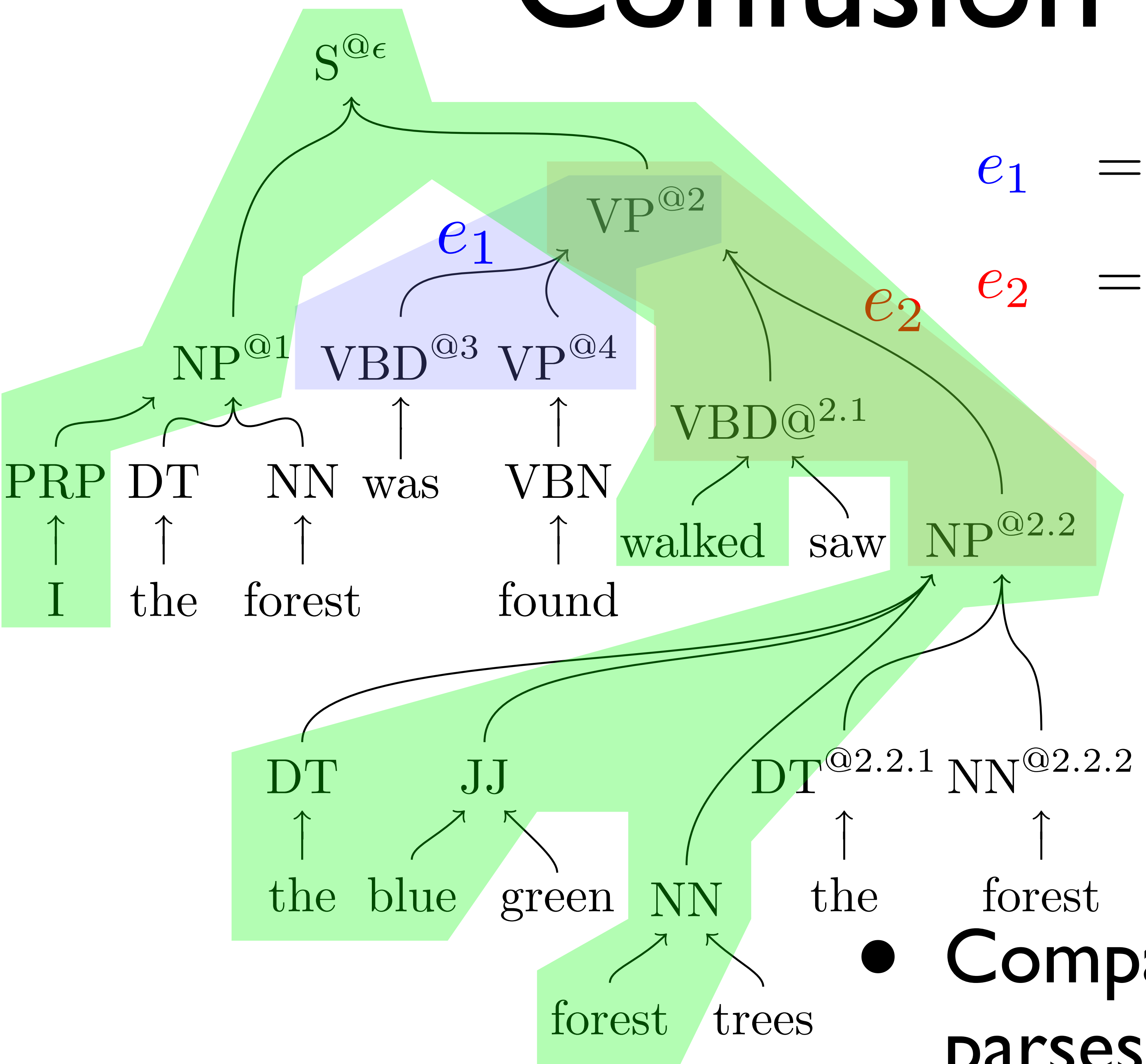


$$e_1 = \langle \mathrm{VP}^{@2}, \{\mathrm{VBD}^{@3}, \mathrm{VP}^{@4}\} \rangle$$

$$e_2 = \langle \mathrm{VP}^{@2}, \{\mathrm{VBD}^{@2.1}, \mathrm{NP}^{@2.2}\} \rangle$$

- Compactly represent multiple parses by sharing nodes
- Represented by “hypergraph”

Confusion Forest

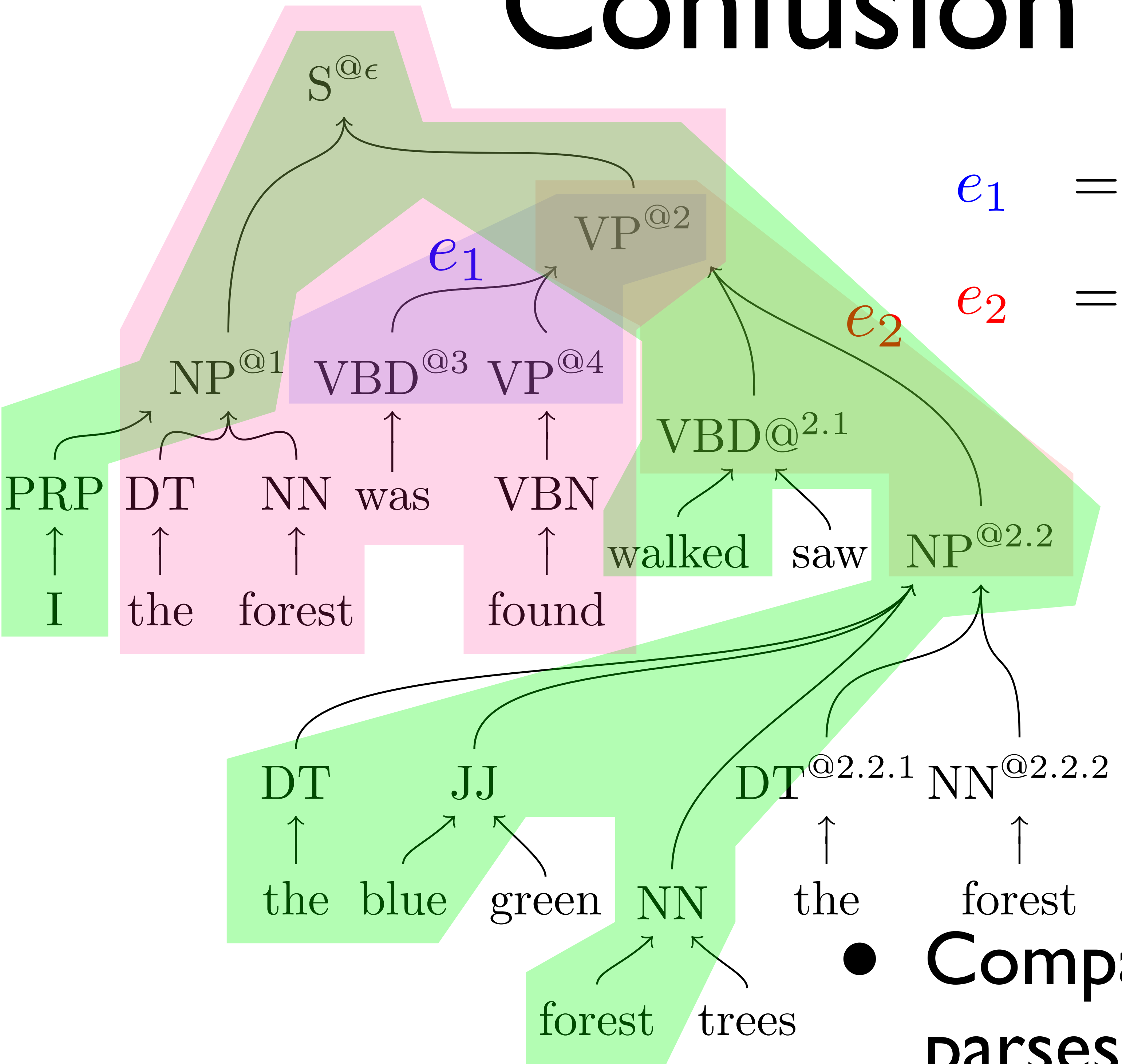


$$e_1 = \langle VP^{@2}, \left\{ VBD^{@3}, VP^{@4} \right\} \rangle$$

$$e_2 = \langle VP^{@2}, \left\{ VBD^{@2.1}, NP^{@2.2} \right\} \rangle$$

- Compactly represent multiple parses by sharing nodes
- Represented by "hypergraph"

Confusion Forest

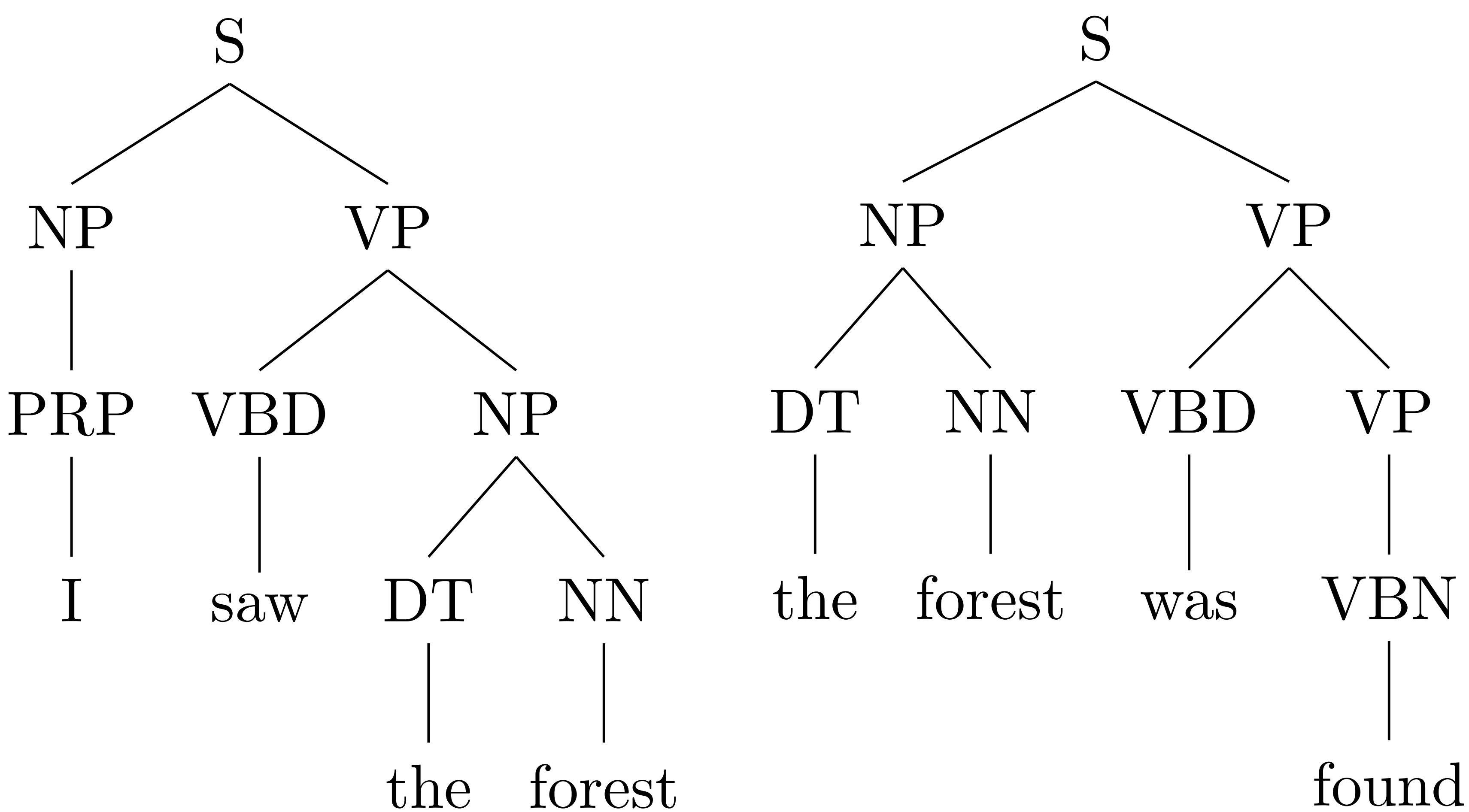


$$e_1 = \langle VP^{@2}, \{VBD^{@3}, VP^{@4}\} \rangle$$

$$e_2 = \langle VP^{@2}, \{VBD^{@2.1}, NP^{@2.2}\} \rangle$$

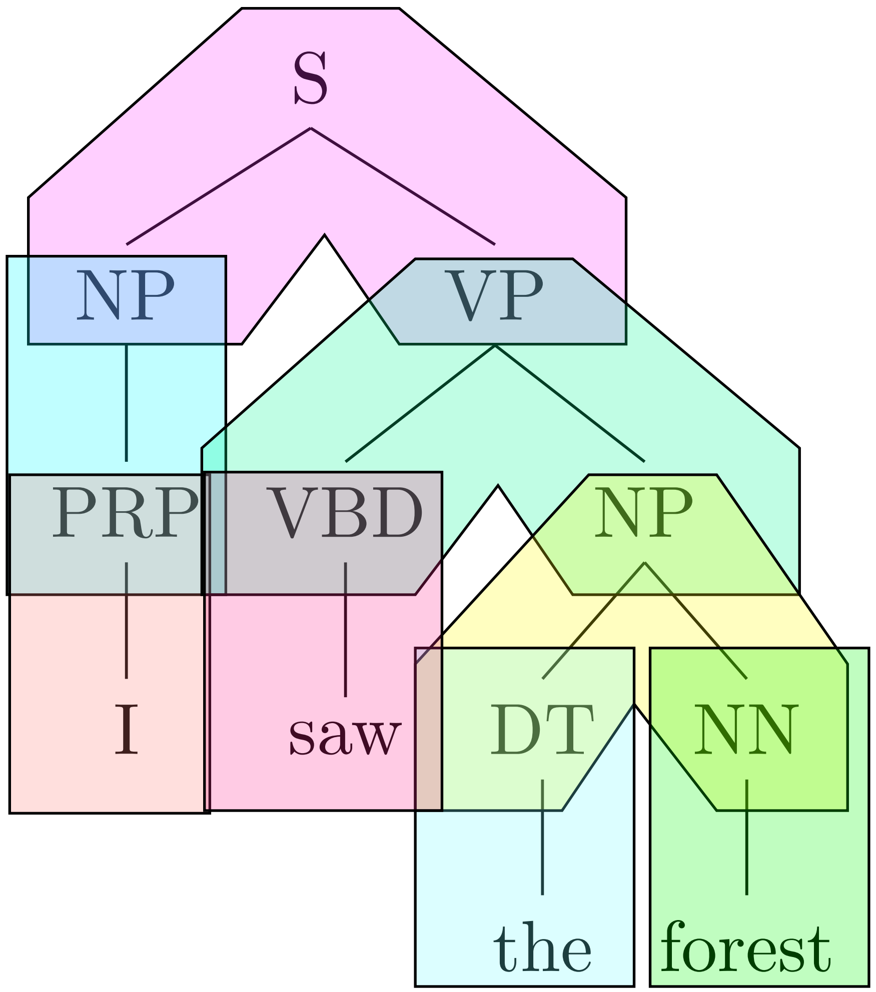
- Compactly represent multiple parses by sharing nodes
- Represented by "hypergraph"

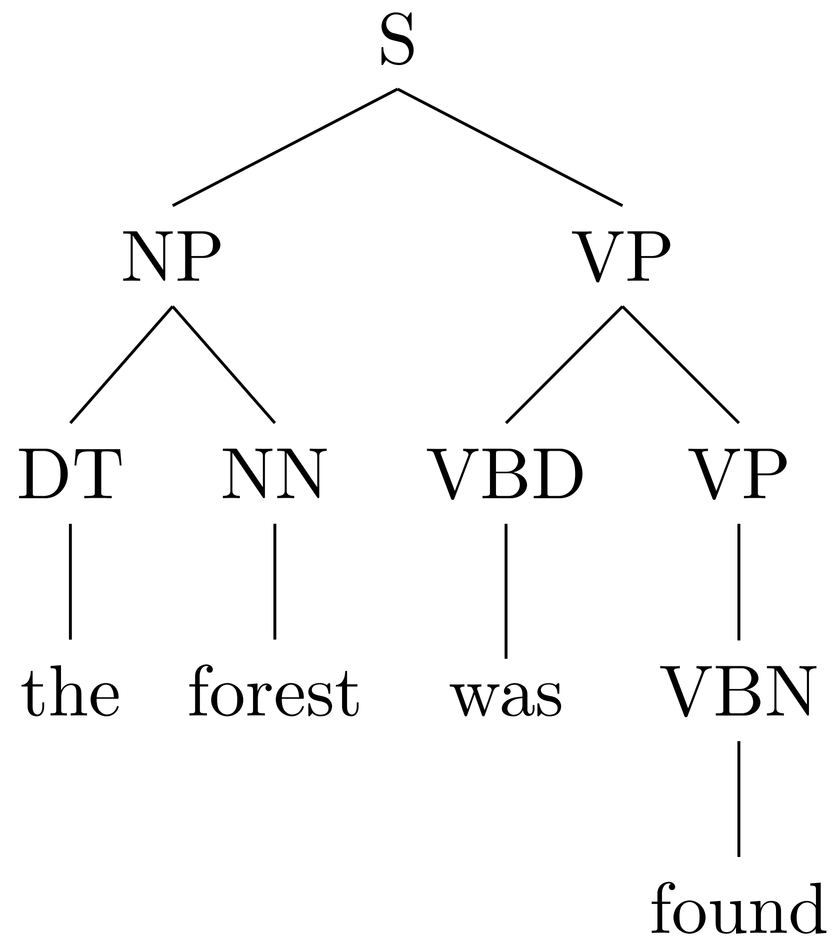
Rule Extraction



- Parse each system output by a parser
- Extract rules from parsed trees: local grammar

Rule Extraction

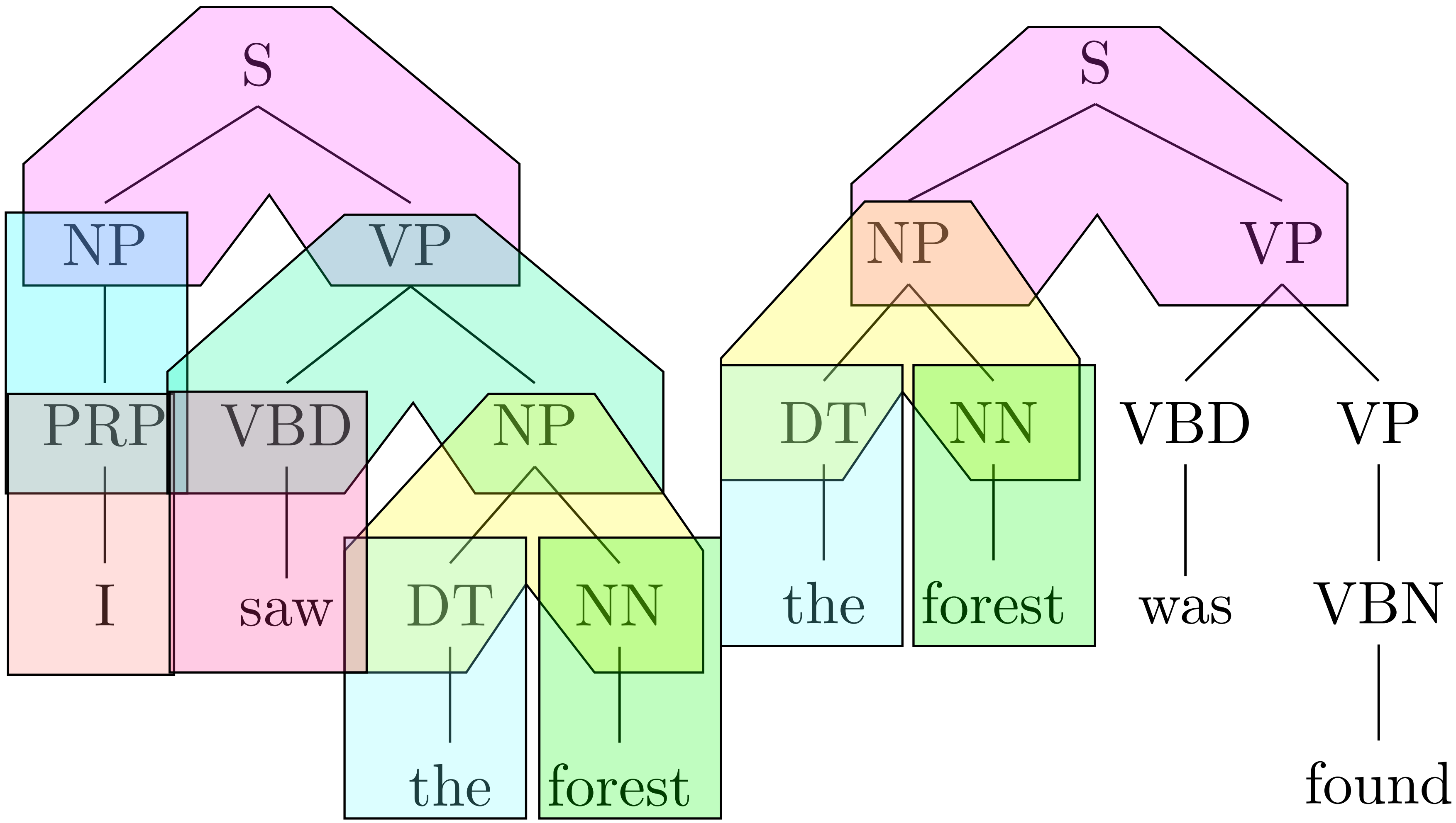




S	→	NP	VP
NP	→	PRP	
PRP	→	I	
VP	→	VBD	NP
VBD	→	saw	
NP	→	DT	NN
DT	→	the	
NN	→	forest	

- Parse each system output by a parser
- Extract rules from parsed trees: local grammar

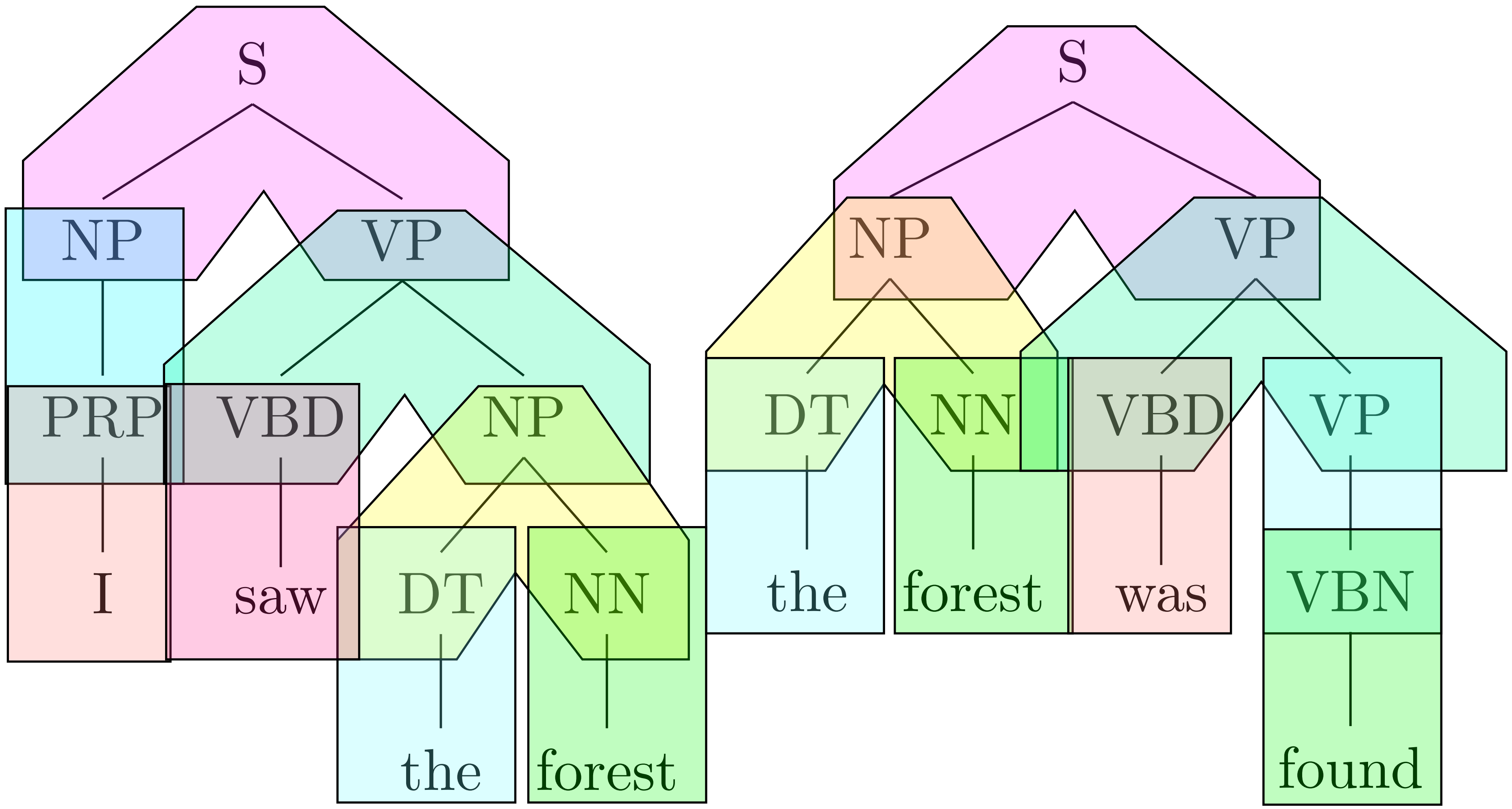
Rule Extraction



S	$\rightarrow$	NP	VP
NP	$\rightarrow$	PRP	
PRP	$\rightarrow$	I	
VP	$\rightarrow$	VBD	NP
VBD	$\rightarrow$	saw	
NP	$\rightarrow$	DT	NN
DT	$\rightarrow$	the	
NN	$\rightarrow$	forest	

- Parse each system output by a parser
- Extract rules from parsed trees: local grammar

Rule Extraction



S	$\rightarrow$	NP	VP
NP	$\rightarrow$	PRP	
PRP	$\rightarrow$	I	
VP	$\rightarrow$	VBD	NP
VBD	$\rightarrow$	saw	
NP	$\rightarrow$	DT	NN
DT	$\rightarrow$	the	
NN	$\rightarrow$	forest	
VP	$\rightarrow$	VBD	VP
VBD	$\rightarrow$	was	
VP	$\rightarrow$	VBN	
VBN	$\rightarrow$	found	

- Parse each system output by a parser
- Extract rules from parsed trees: local grammar

Generation by Earley

Scan:

$$\frac{[\mathrm{X} \to \alpha \bullet x\beta, h] : u}{[\mathrm{X} \to \alpha x \bullet \beta, h] : u}$$

Predict:

$$\frac{[\mathrm{X} \to \alpha \bullet \mathrm{Y}\beta, h]}{[\mathrm{Y} \to \bullet\gamma, h+1] : u} \quad \mathrm{Y} \xrightarrow{u} \gamma \in \mathcal{G}, h < H$$

Complete:

$$\frac{[\mathrm{X} \to \alpha \bullet \mathrm{Y}\beta, h] : u \quad [\mathrm{Y} \to \gamma\bullet, h+1] : v}{[\mathrm{X} \to \alpha\mathrm{Y} \bullet \beta, h] : u \otimes v}$$

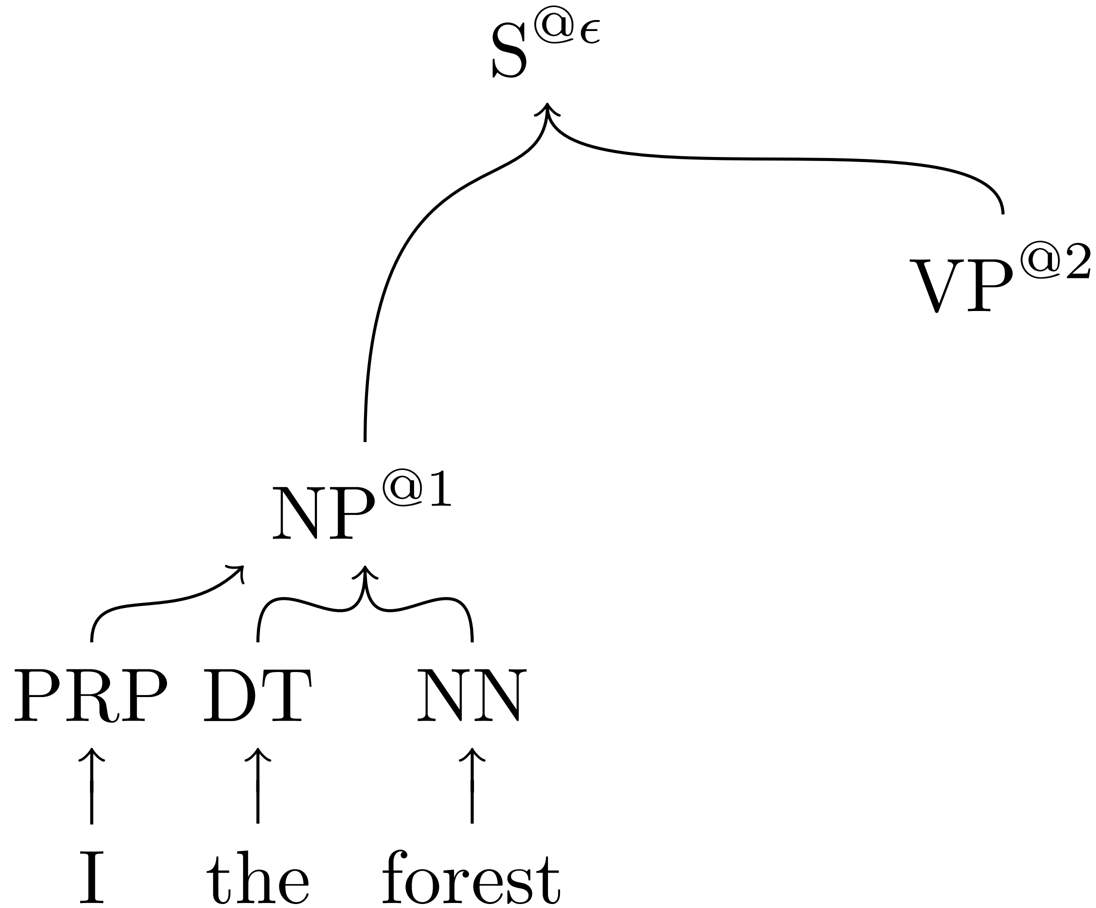
- Generation from the extracted grammar
- Scanning always succeed: constraint by height

Generation by Earley

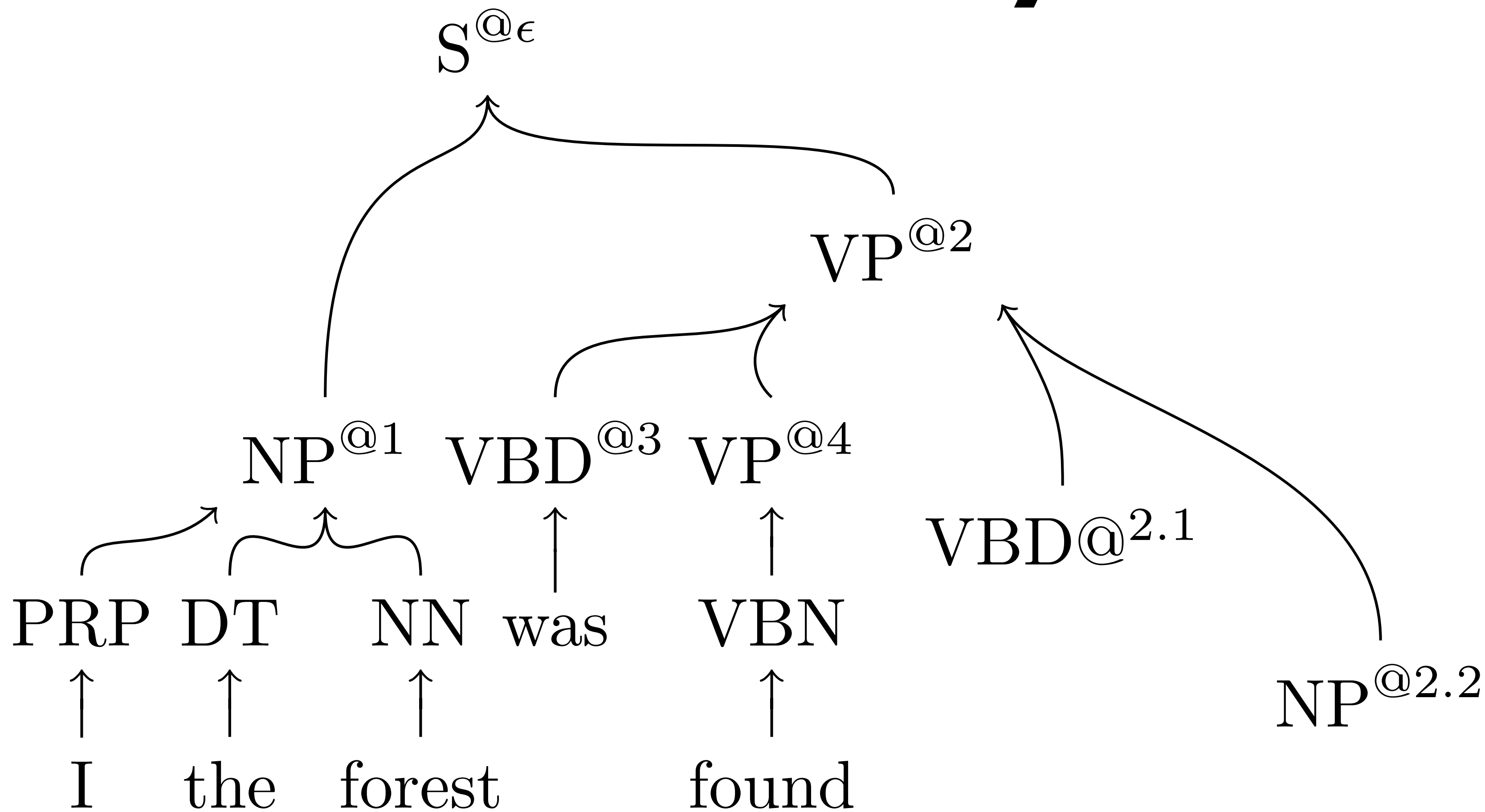
Generation by Earley

$S^{@\epsilon}$

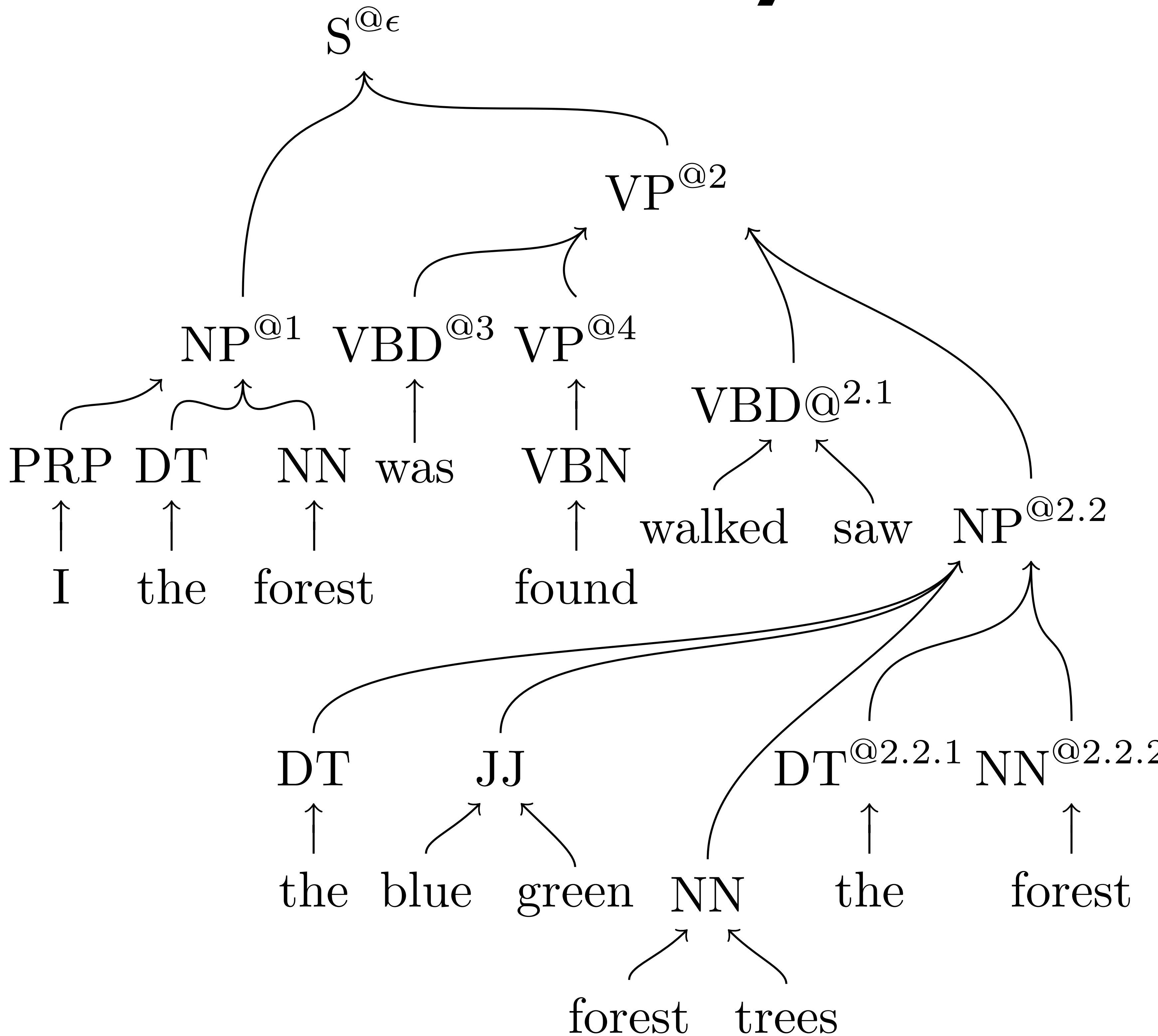
Generation by Earley



Generation by Earley



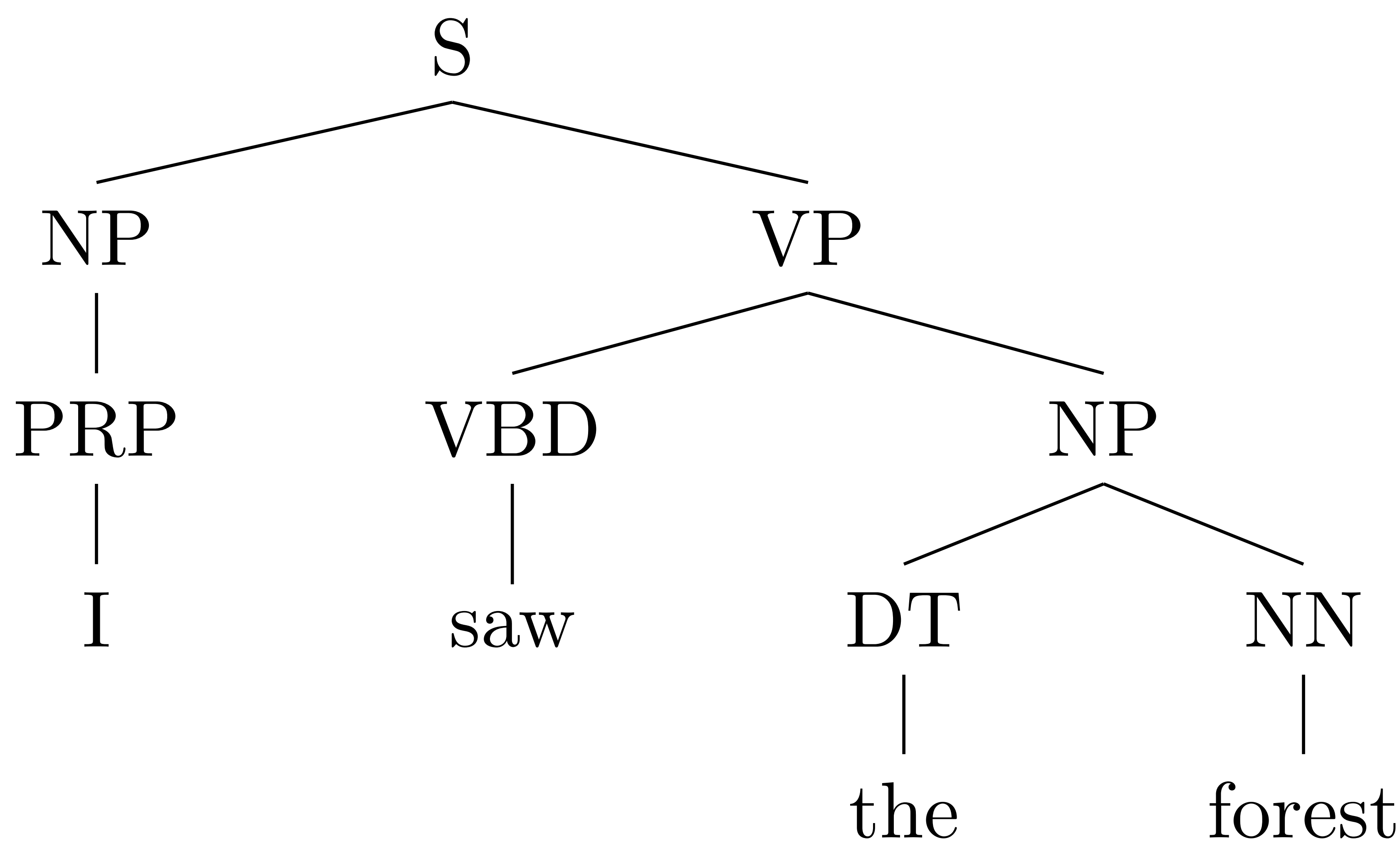
Generation by Earley



Spurious Ambiguity

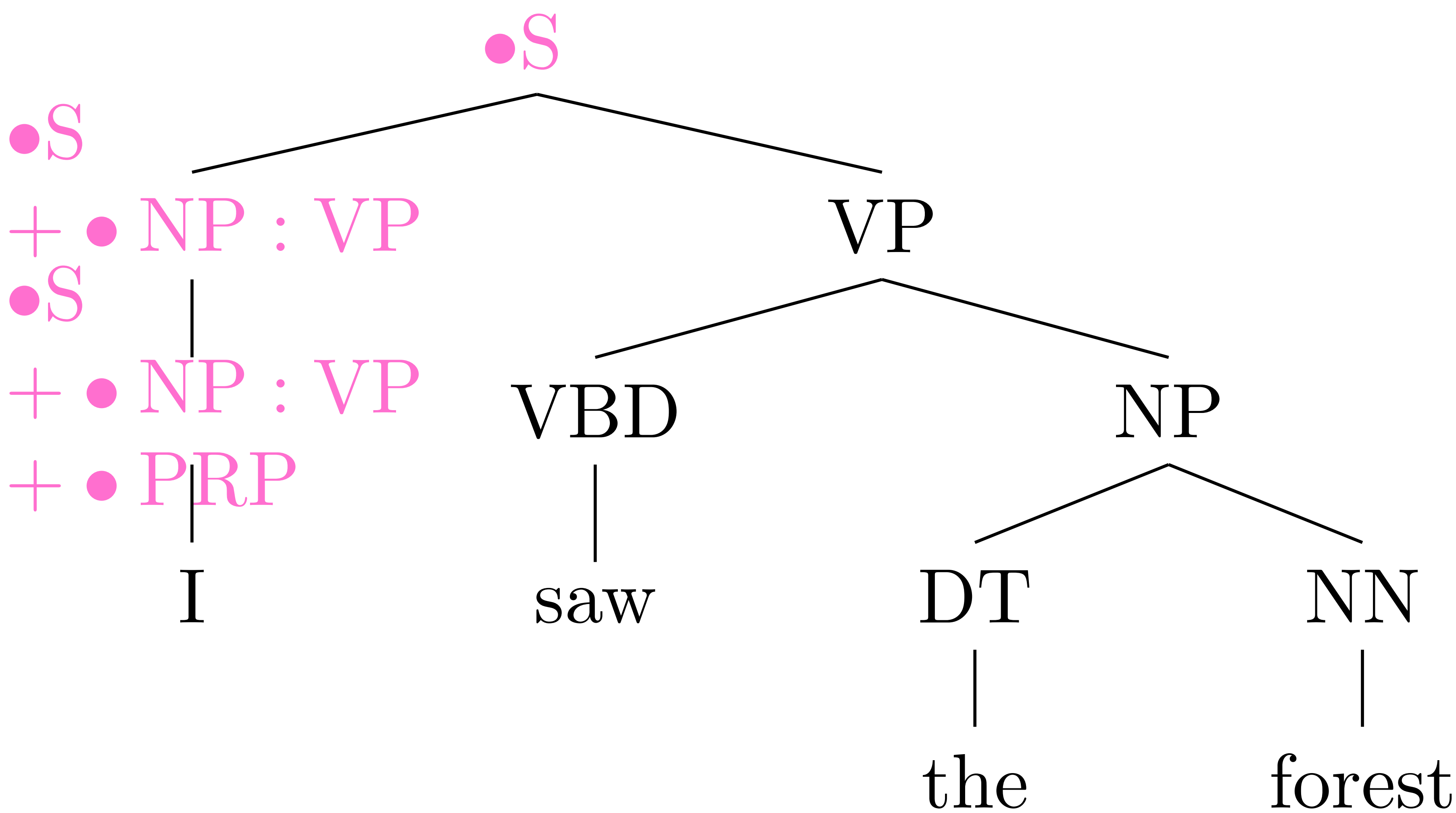
- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
  - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Spurious Ambiguity



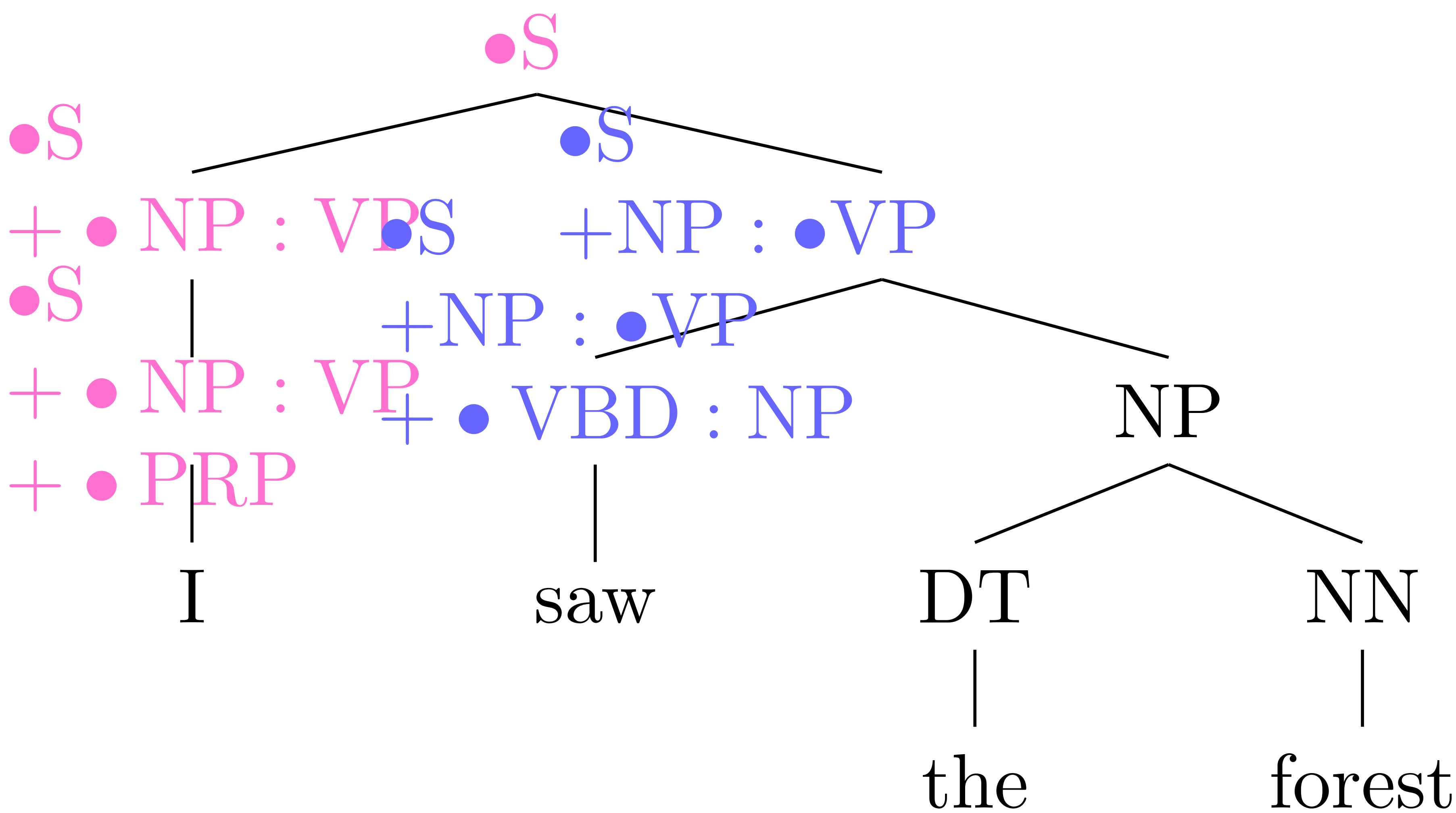
- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
  - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Spurious Ambiguity



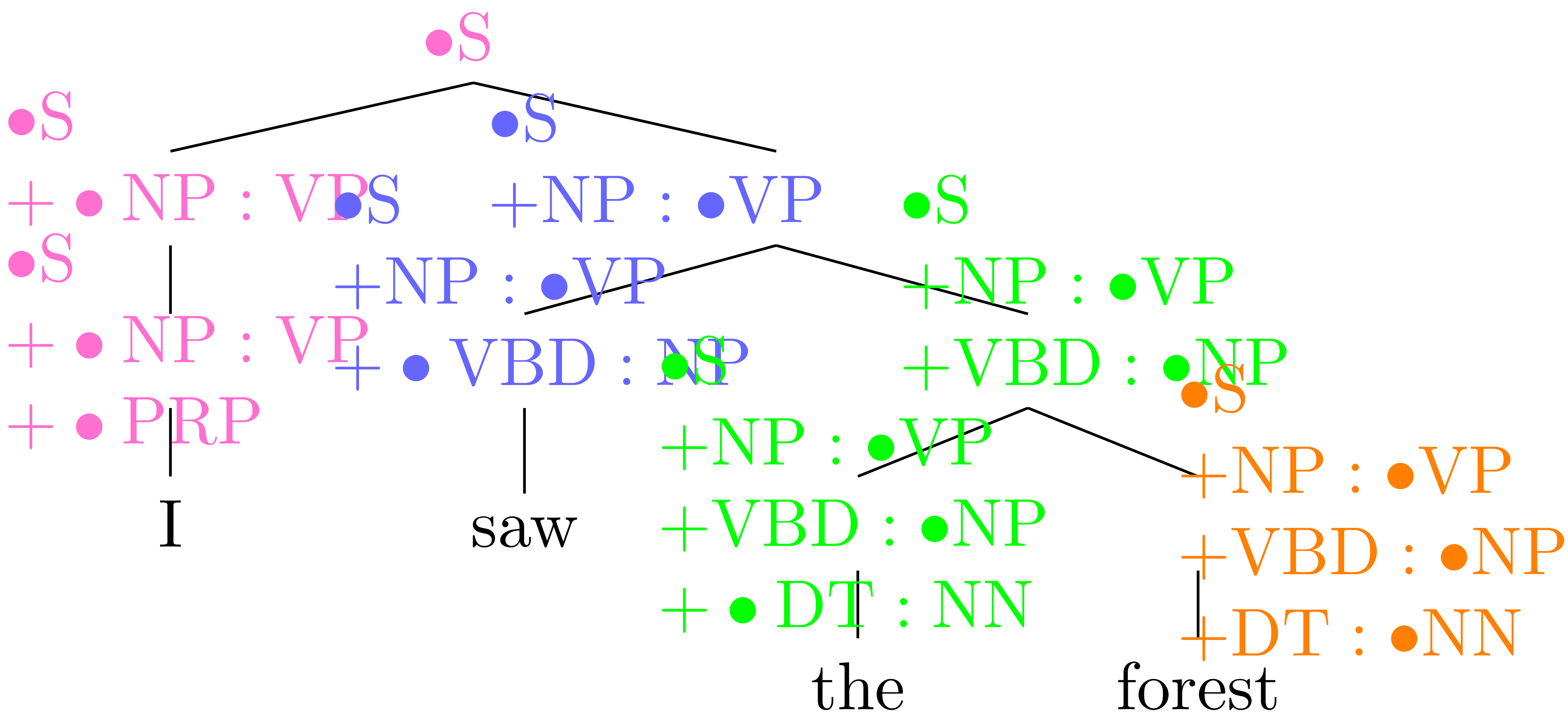
- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
    - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Spurious Ambiguity



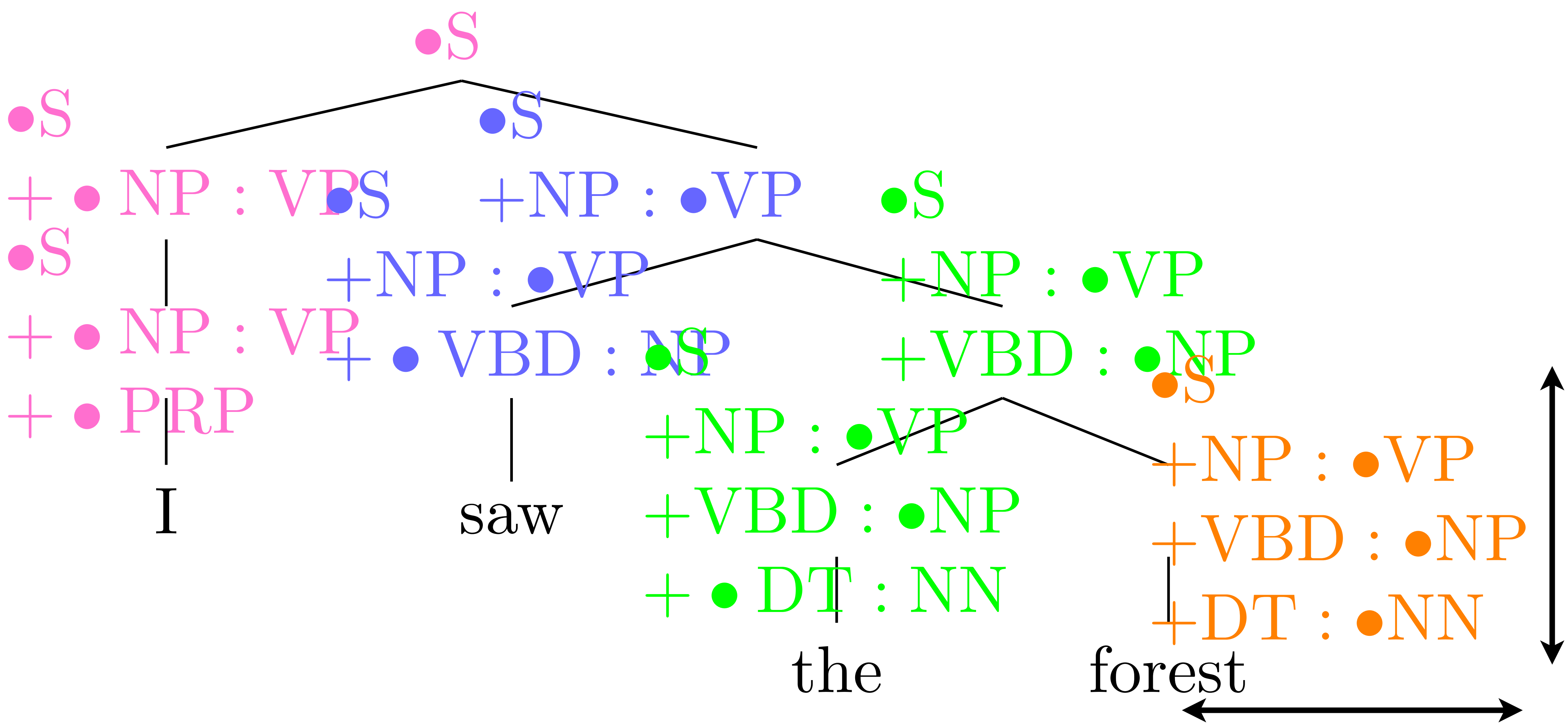
- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
  - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Spurious Ambiguity



- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
  - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Spurious Ambiguity



- Memorize the (partial) tree structures in each node
- Employ the sequence of Ealrye state as a node
  - Horizontal/Vertical Markovization (Klein and Manning, 2003)

Forest Reranking

$$\hat{d} = \arg\max_{d \in D} \mathbf{w}^\top \cdot \mathbf{h}(d, F)$$

- Choose the best derivation d among all possible derivations D in a forest F
  - Terminal yield of the best derivation = the best translation
  - Approximately apply non-local features (ngram language models) by Cube Pruning (Huang and Chiang, 2007)
  - Efficient k-best by Algorithm 3 (Huang and Chiang, 2005)

Experiments

- WMT10 System Combination Task
  - Czech, German, Spanish, French → English
  - tune/test: 455/2,034 sentences

	cz-en	de-en	es-en	fr-en
systems	6	16	8	14
tune	10.6K	10.9K	10.9K	11.0K
test	50.5K	52.1K	52.1K	52.4K

Systems

Systems

- CF: Stanford parser + “cicada” (a hypergraph-based toolkit based on SEMIring parsing framework)
- CN: Single network by merging multiple networks + conversion into hypergraph by lattice parsing

Systems

- CF: Stanford parser + "cicada" (a hypergraph-based toolkit based on SEMIring parsing framework)
- CN: Single network by merging multiple networks + conversion into hypergraph by lattice parsing
- features: tuned by hypergraph-MERT(Kumar et al. 2009)
  - Language Models, # of terminals, # of hyperedges
  - # of rules in a derivation originally in n_{th} system output
  - BLEUs by treating each system output as a reference translation
  - Network distance (only used for CN)

BLEU

	cz-en	de-en	es-en	fr-en
system min	14.09	15.62	21.79	16.79
max	23.44	24.10	29.97	***29.17***
CN	23.70	24.09	***30.45***	***29.15***
CF,v=∞,h=∞	***24.13***	24.18	***30.41***	***29.57***
CF,v=∞,h=2	***24.14***	***24.58***	***30.52***	28.84
CF,v=∞,h=1	***24.01***	23.91	***30.46***	***29.32***

Oracle BLEU

	cz-en	de-en	es-en	fr-en
rerank	29.40	32.32	36.83	36.59
CN	38.52	34.97	47.65	46.37
CF, $v=\infty, h=\infty$	30.51	34.07	38.69	38.94
CF, $v=\infty, h=2$	30.61	34.25	38.87	39.10
CF, $v=\infty, h=1$	31.09	34.65	39.27	39.51

Hypegraph size

	cz-en	de-en	es-en	fr-en
CN	2,222.68	47,231.20	2,932.24	11,969.40
CF,v=∞,h=1	230.08	540.03	262.30	386.79
CF,v=5,h=1	254.45	651.10	302.01	477.51
CF,v=4,h=1	286.01	802.79	349.21	575.17

- Average # of hyperedges
- (rough) estimates for speed

Conclusion

- System combination by Confusion Forest which employs syntactic distance, not word-level distance
- Forest construction by the grammar extracted from system outputs
    - Parser: assign tree structure to the similar expressions
- Compact date structure + comparable performance against Confusion Network
- Future work
    - Syntactic features