

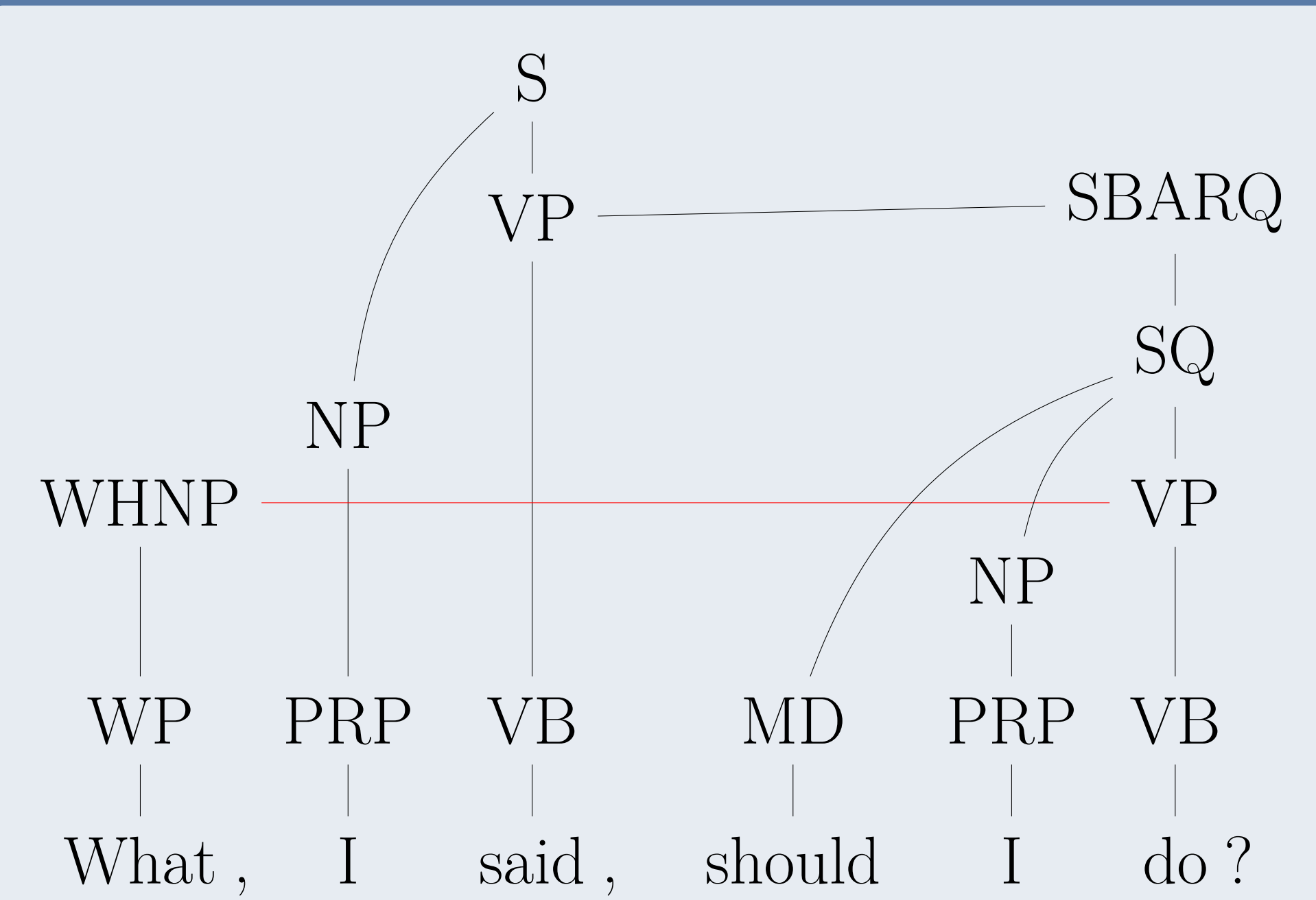
Efficient Discontinuous Phrase-Structure Parsing via the Generalized Maximum Spanning Arborescence

Caio Corro, Joseph Le Roux, Mathieu Lacroix

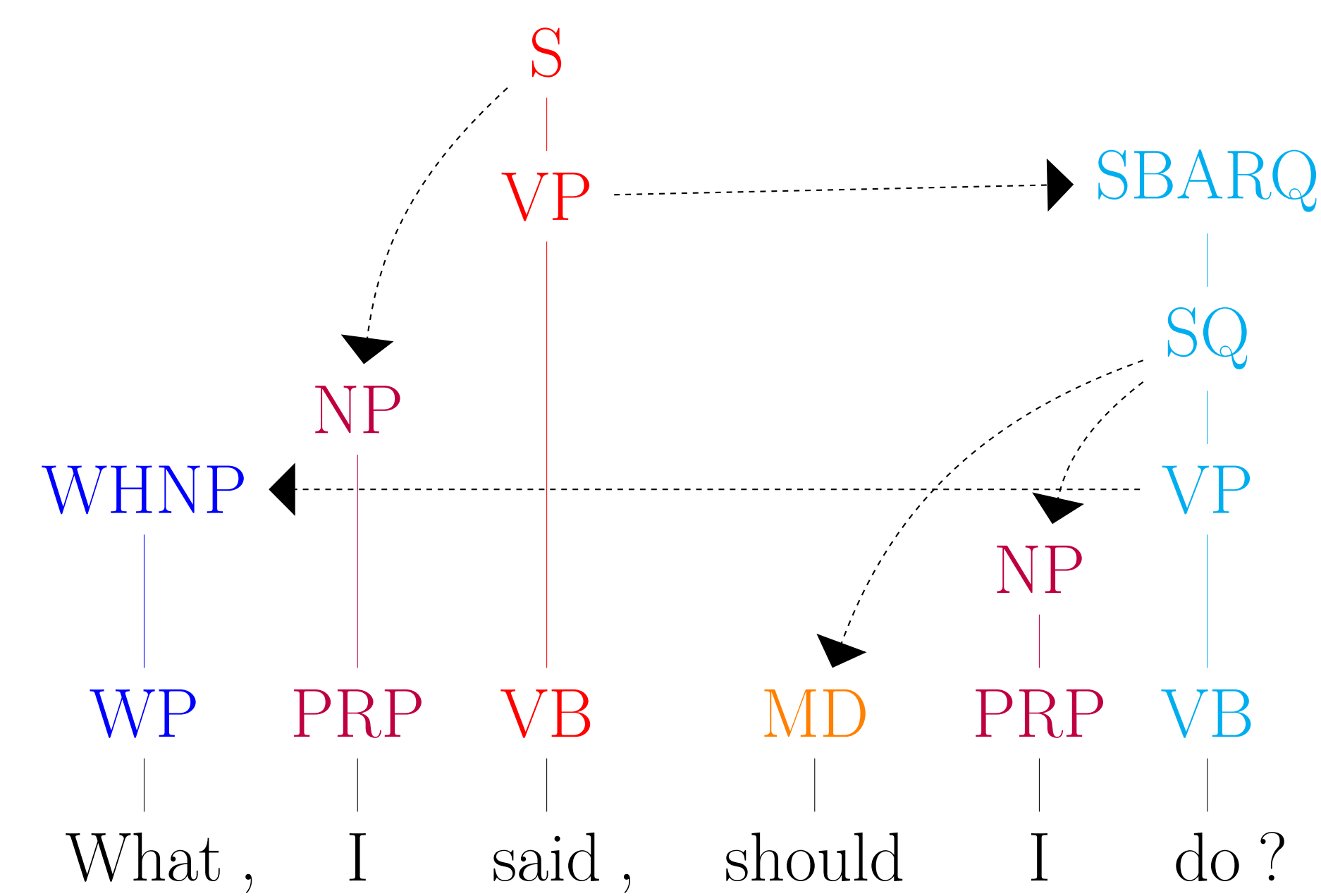
Laboratoire Informatique de Paris Nord (LIPN), Université Paris 13 - SPC, CNRS UMR 7030



Discontinuous Phrase Structure



Discontinuous Lexicalized Spinal Grammar



Parsing: Joint Problem

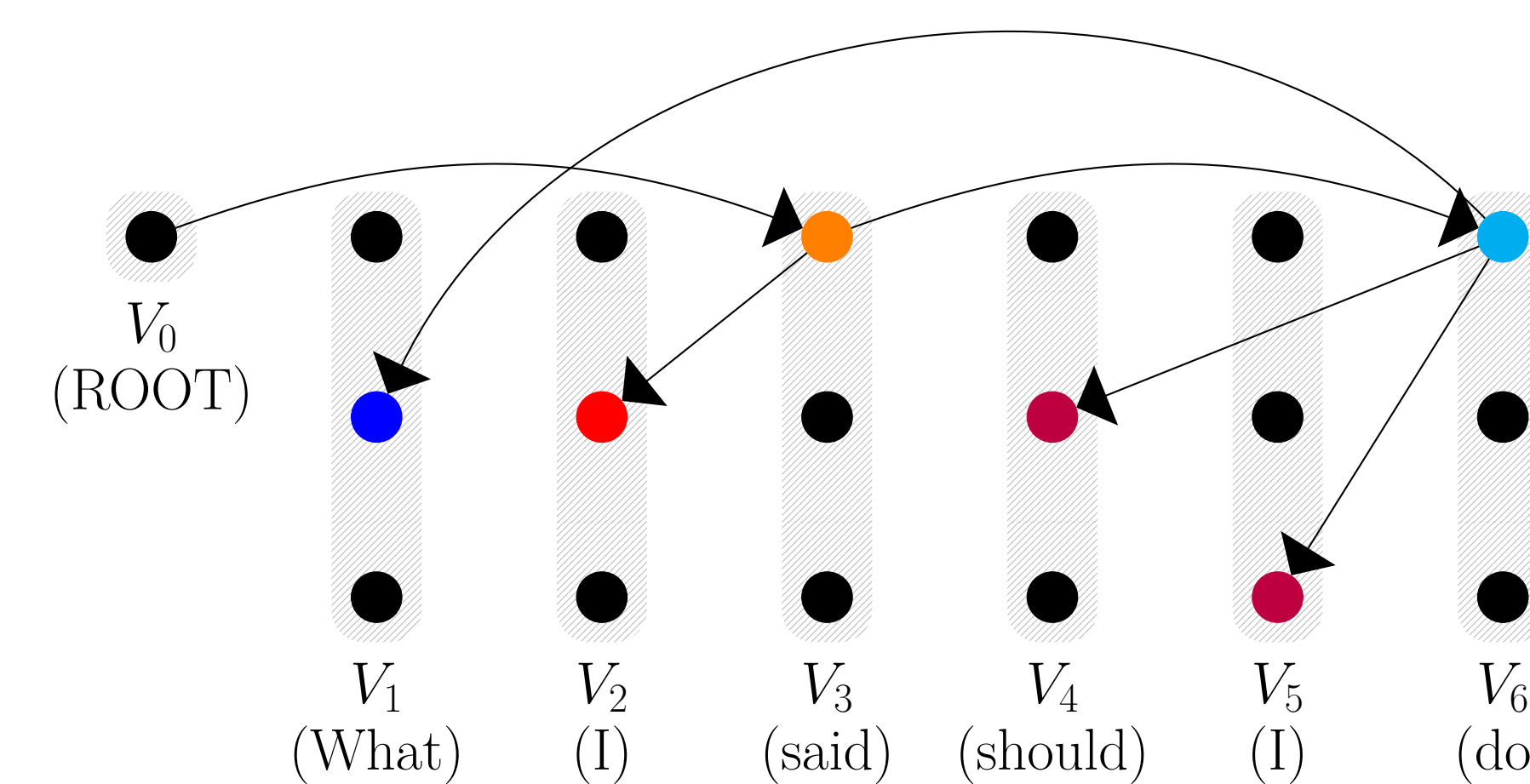
- Spine tagging
- Non-projective dependency parsing

Benefits

- Reduces to a known optimization problem
- Based on two well-studied tasks in NLP

Generalized Maximum Spanning Arborescence

Joint tagging and parsing **Generalized Spanning Arborescence**
 One spine per word ⇔ One node per cluster
 Non-projective dependencies ⇔ Spanning arborescence over clusters



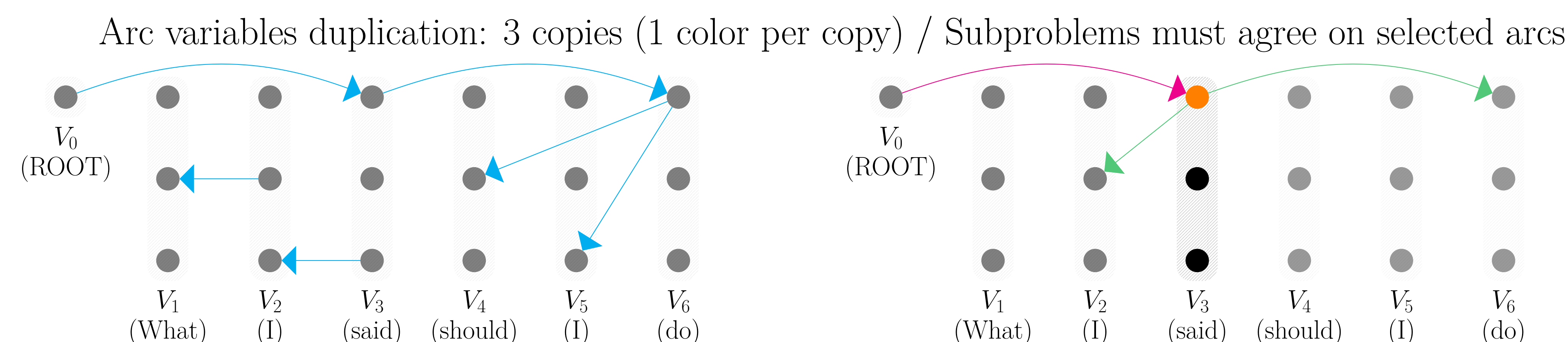
Integer Linear Programming Formulation

Variables:
 y_a : selection of arc $a \in A$
 x_v : selection of vertex $v \in V$

Optimization program:
 (1): structure with maximum weight under an arc-factored model
 (2)-(4): y is a spanning arborescence over clusters
 (5): vertex x_v included if it is adjacent to an arc in y
 (6): exactly one vertex per cluster

$$\begin{aligned} \max_{x,y} \quad & \phi^\top y & (1) \\ \text{s.t.} \quad & y(\delta^-(V_0)) = 0 & (2) \\ & y(\delta^-(V_k)) = 1 & (3) \\ & y(\delta^-(\bigcup_{V_k \in \pi'} V_k)) \geq 1 & (4) \\ & x_v \geq y_a & (5) \\ & x_v(V_k) = 1 & (6) \end{aligned}$$

Fast Decoding via Dual Decomposition and Subgradient Descent



Maximum Spanning Arborescence over clusters
 ⇒ No constraint on adjacent vertices

One subproblem per cluster
 ⇒ Exactly one adjacent vertex per cluster

Neural Parametrization

\mathbf{d} : dependencies (h, m) : a single dependency
 \mathbf{s} : spine tags s_m : a single tag
 \mathbf{w} : input sentence

$$\begin{aligned} P(\mathbf{d}, \mathbf{s} | \mathbf{w}) &= P_\alpha(\mathbf{d} | \mathbf{w}) \times P_\nu(\mathbf{s} | \mathbf{d}, \mathbf{w}) \\ &\approx \prod_{(h,m) \in \mathbf{d}} P_\alpha(h | m, \mathbf{w}) \times P_\nu(s_m | m, \mathbf{d}, \mathbf{w}) \\ &\approx \prod_{(h,m) \in \mathbf{d}} P_\alpha(h | m, \mathbf{w}) \times P_\nu(s_m | m, h, \mathbf{w}) \end{aligned}$$

Neural network: Parameter estimation:
 Stacked bi-LSTMs • Log-likelihood max.
 + Biaffine classifiers • Dropout: 0.5 on inputs
 + Softmax

Experimental Results

Discontinuous PTB (English)		
	LF	Time
Short sentences only		
This work	89.85	≈ 4
van Cranenburgh <i>et al.</i>	87.00	≈ 180
Full test set		
This work	89.17	≈ 5.5
TIGER (German)		
	LF	Time
This work	81.63	≈ 11
Coavoux <i>et al.</i>	81.60	≈ 2.5

Future work

- Max-margin training
- High-order scoring models:
 - Bi-gram
 - Sibling and grand-father
- Application to other joint tagging and parsing problems