Unsupervised Latent Tree Induction with Deep Inside Outside Recursive Autoencoders

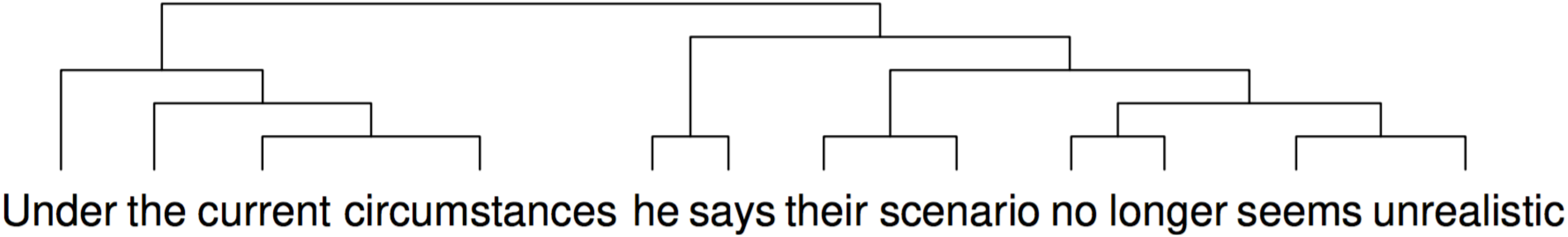
Wang Ge

Group Meeting

2019/10/23

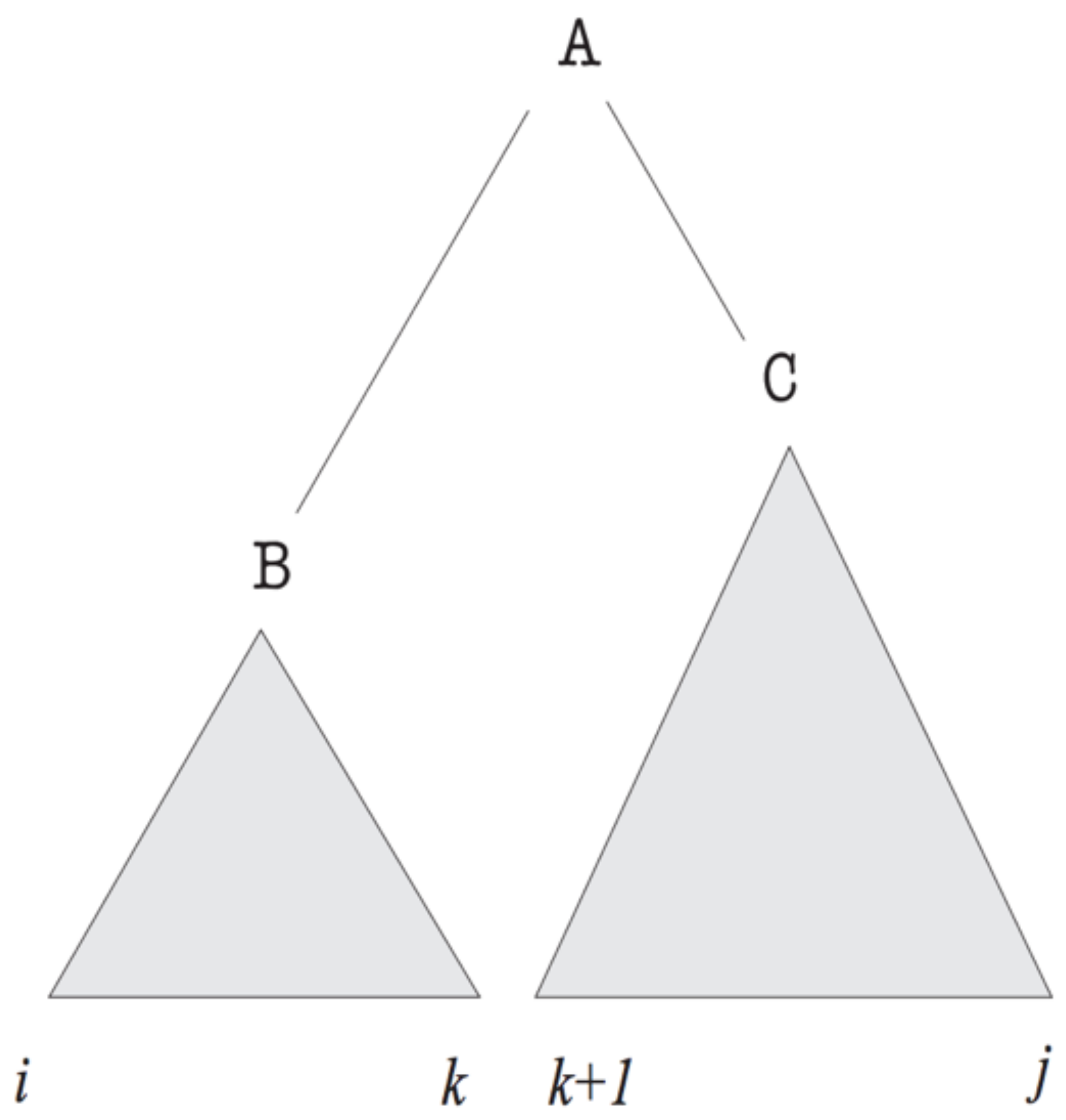
Drozdov et al. NAACL 2019

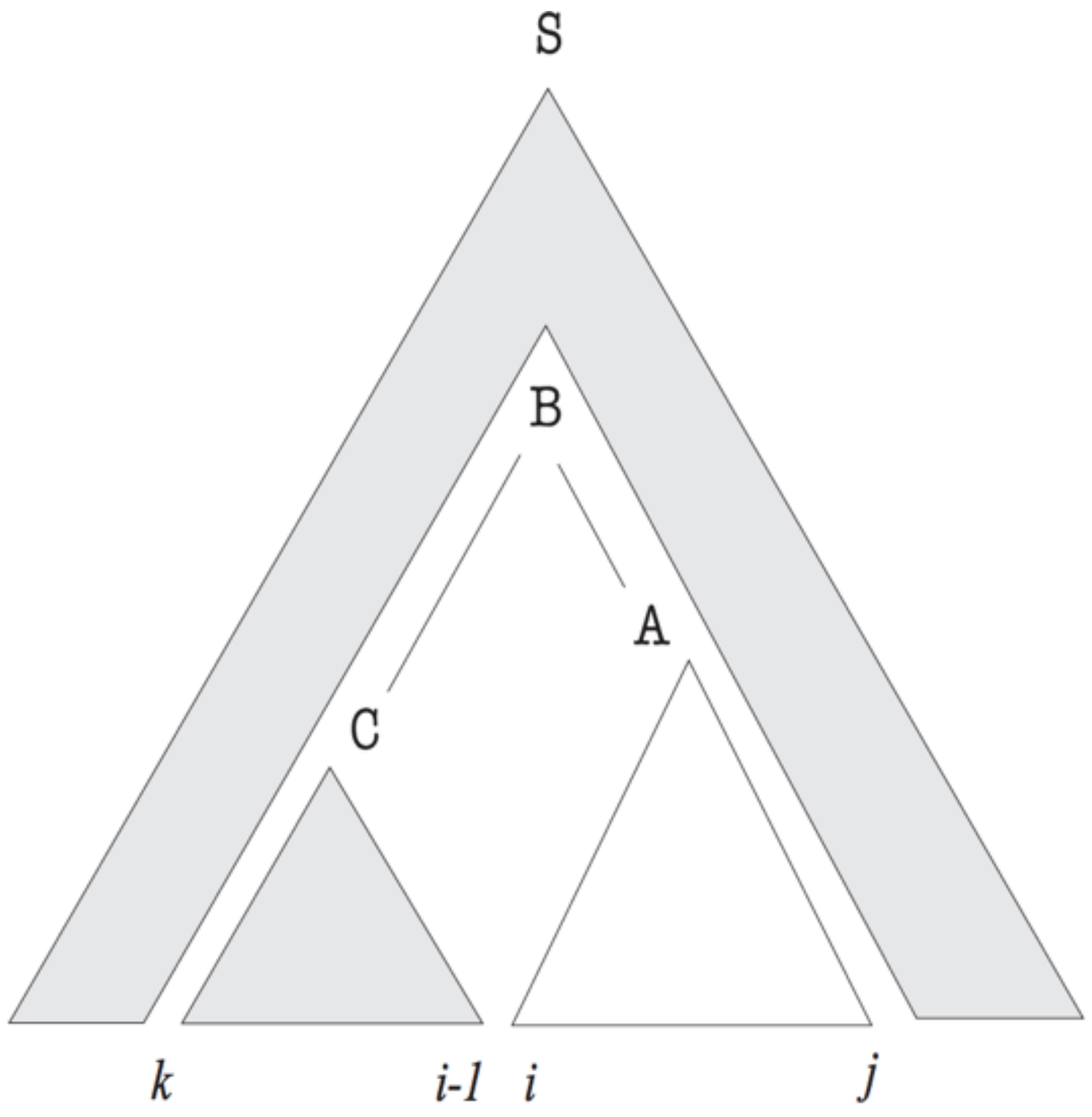
Unsupervised Constituent Parsing



- A task long been dominated by classic methods like CCM
- Recent researches focus on downstream works
- This work tries to integrate traditional methods with deep learning

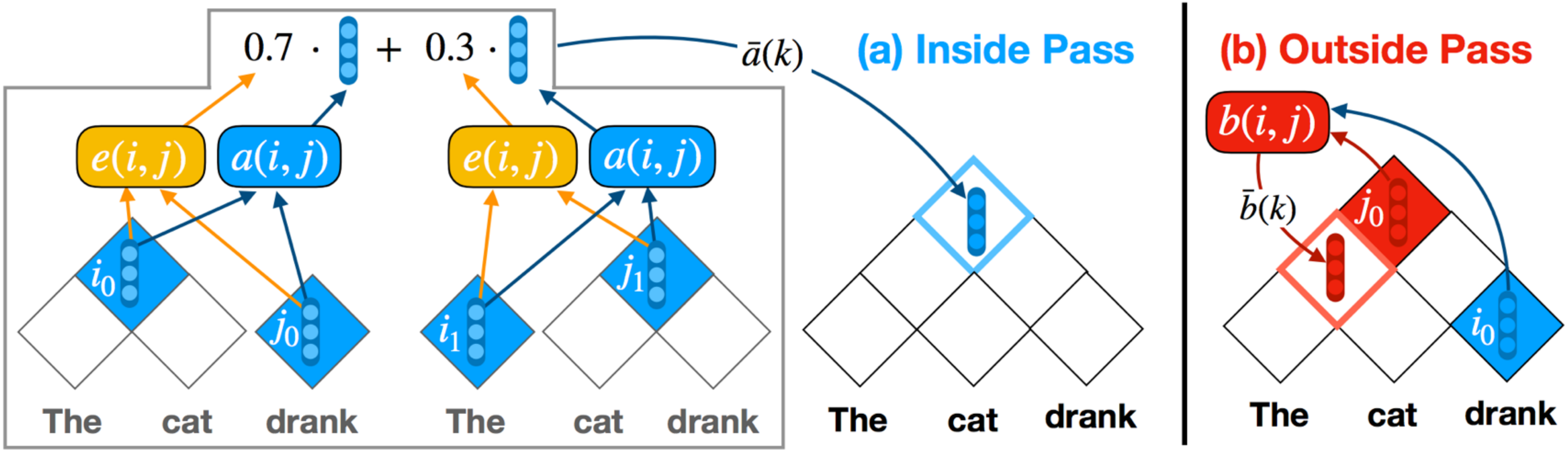
Inside-Outside Algorithm





The key point of this work is to construct a neuralized inside-outside chart

Neural Inside Outside Pass



Neural representation required for score computation

Inside Pass

$$\begin{bmatrix} x \\ o \\ u \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \tanh \end{bmatrix} (U_\psi v_k + b)$$

$$\bar{a}(k) = o + \tanh(x \odot u)$$

$$\bar{e}(k) = 0$$

$$e(i,j) = \frac{\exp(\hat{e}(i,j))}{\sum_{\hat{i},\hat{j} \in \{k\}} \exp(\hat{e}(\hat{i},\hat{j}))}$$

$$\hat{e}(i,j) = \phi(\bar{a}(i), \bar{a}(j); S_\alpha) + \bar{e}(i) + \bar{e}(j)$$

$$a(i,j) = \mathrm{Compose}_\alpha(\bar{a}(i), \bar{a}(j))$$

$$\bar{a}(k) = \sum_{i,j \in \{k\}} e(i,j)\, a(i,j)$$

$$\bar{e}(k) = \sum_{i,j \in \{k\}} e(i,j)\, \hat{e}(i,j)$$

where Compose is TreeLSTM
or 2-layer MLP

Outside Pass

$$\bar{b}(k) = \sum_{i,j \in \{k\}} f(i,j)\, b(i,j)$$

$$\bar{f}(k) = \sum_{i,j \in \{k\}} f(i,j)\, \hat{f}(i,j)$$

$$b(i,j) = \mathrm{Compose}_\beta(\bar{a}(i), \bar{b}(j))$$

$$\hat{f}(i,j) = \phi(\bar{a}(i), \bar{b}(j); S_\beta) + \bar{e}(i) + \bar{f}(j)$$

- The representation for root is learned as a bias vector
- Some parameters are shared with that of the inside pass

Training and Parsing

- Training use an auto encoding objective

- Aiming to maximize the inside-outside scores for basic spans, negative sampling is used

$$L_{\boldsymbol{x}} = \sum_{i=0}^{T-1} \sum_{i*=0}^{N-1} \max(0, 1 - \bar{b}(i) \cdot \bar{a}(i) + \bar{b}(i) \cdot \bar{a}(i^*))$$

$$Z^* = \sum_{i*=0}^{N-1} \exp(\bar{b}(i) \cdot \bar{a}(i^*))$$

$$L_{\boldsymbol{x}} = -\sum_{i=0}^{T-1} \log \frac{\exp(\bar{b}(i) \cdot \bar{a}(i))}{\exp(\bar{b}(i) \cdot \bar{a}(i)) + Z^*}$$

- The parse tree is extracted by Viterbi decoding from the inside table

Banalized WSJ and NLI Data Results

Model	$\mathbf{F1}_{\mu}$	$\mathbf{F1}_{max}$	δ
LB	13.1	13.1	12.4
RB	16.5	16.5	12.4
Random	21.4	21.4	5.3
Balanced	21.3	21.3	4.6
RL-SPINN†	13.2	13.2	-
ST-Gumbel - GRU†	22.8 ± 1.6	25.0	-
PRPN-UP	38.3 ± 0.5	39.8	5.9
PRPN-LM	35.0 ± 5.4	42.8	6.2
ON-LSTM	47.7 ± 1.5	49.4	5.6
DIORA	48.9 ± 0.5	49.6	8.0
PRPN-UP^{+PP}	-	45.2	6.7
PRPN-LM^{+PP}	-	42.4	6.3
DIORA^{+PP}	**55.7** ± 0.4	**56.2**	8.5

Model	$\mathbf{F1}_{median}$	$\mathbf{F1}_{max}$	δ
Random	27.0	27.0	4.4
Balanced	21.3	21.3	3.9
PRPN-UP	48.6	-	4.9
PRPN-LM	50.4	-	5.1
DIORA	51.2	53.3	6.4
PRPN-UP^{+PP}	-	54.8	5.2
PRPN-LM^{+PP}	-	50.4	5.1
DIORA^{+PP}	59.0	**59.1**	6.7

WSJ-10,40 Parsing Results;Full WSJ Segmentation Results

Model	WSJ-10		WSJ-40	
	F1$_{\mu}$	**F1**$_{max}$	**F1**$_{\mu}$	**F1**$_{max}$
UB	87.8	87.8	85.7	85.7
LB	28.7	28.7	12.0	12.0
RB	61.7	61.7	40.7	40.7
CCM†	-	63.2	-	-
CCM$_{gold}$†	-	71.9	-	33.7
PRLG †	-	**72.1**	-	54.6
PRPN$_{NLI}$	66.3 $\pm$0.8	68.5	-	-
PRPN‡	70.5 $\pm$0.4	71.3	-	52.4
ON-LSTM‡	65.1 $\pm$1.7	66.8	-	-
DIORA	67.7 $\pm$0.7	68.5	60.6 $\pm$0.2	**60.9**

Label	Count	DIORA	P-UP	P-LM
NP	297,872	**0.767**	0.687	0.598
VP	168,605	**0.628**	0.393	0.316
PP	116,338	0.595	0.497	**0.602**
S	87,714	**0.798**	0.639	0.657
SBAR	24,743	**0.613**	0.403	0.554
ADJP	12,263	**0.604**	0.342	0.360
QP	11,441	**0.801**	0.336	0.545
ADVP	5,817	**0.693**	0.392	0.500
PRN	2,971	**0.546**	0.127	0.144
SINV	2,563	0.926	0.904	**0.932**

Phrase Similarity Results

Model	Dim	CoNLL 2000			CoNLL 2012		
		P@1	P@10	P@100	P@1	P@10	P@100
Random	800	0.684	0.683	0.680	0.137	0.133	0.135
ELMo_{CI}	1024	0.962	0.955	0.957	0.708	0.643	0.544
ELMo_{SI}	4096	0.970	0.964	0.955	0.660	0.624	0.533
ELMo	4096	0.987	0.983	0.974	**0.896**	**0.847**	**0.716**
$\text{DIORA}_{In/Out}$	800	**0.990**	**0.985**	**0.979**	0.860	0.796	0.646