Toward Fast and Accurate Neural Discourse Segmentation

Yizhong Wang Sujian Li Jingfeng Yang

MOE Key Laboratory of Computational Linguistics,
Peking University

Discourse Segmentation

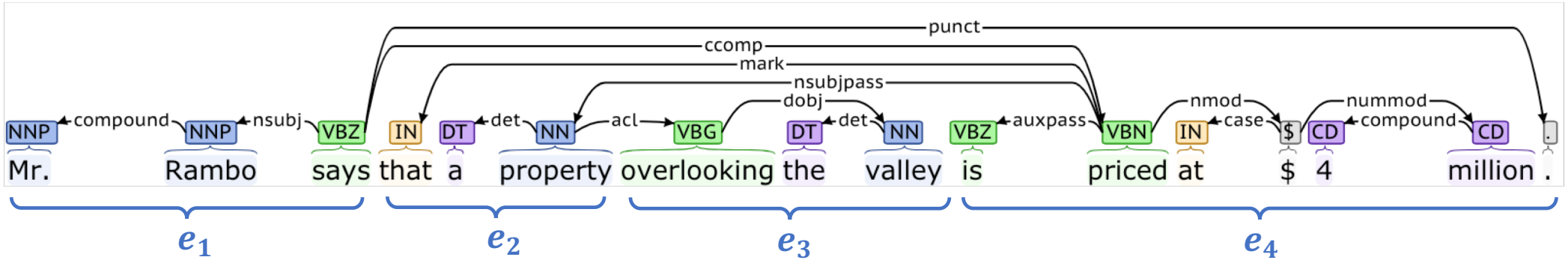
- Segmenting text into Elementary Discourse Units (EDU)

> … [Mr. Rambo says]$_{e_1}$ [*that a 3.2-acre property*]$_{e_2}$ [*overlooking the San Fernando Valley*]$_{e_3}$ [*is priced at $4 million*]$_{e_4}$[*because the late actor Erroll Erroll Flynn once lived there.*]$_{e_5}$ [*"If Flynn hadn't lived there,*]$_{e_6}$[*the property might have been priced $1 million lower,"*]$_{e_7}$ [*says Mr. Rambo,*]$_{e_8}$ [*noting*]$_{e_9}$ [*that Flynn's house has been bulldozed,*]$_{e_{10}}$ [*and only the pool remains.*]$_{e_{11}}$ …

- The first step in RST-style discourse parsing and can be used for many downstream tasks, such as sentence compression or document summarization.

Traditional Feature-based Approach

- Good performance with syntactic features:
  - Part-of-speech tags
  - Parse trees
- These features are important since EDUs are initially designed to be determined with lexical and syntactic clues



Efficiency is the Major Concern!

- Discourse segmentation is a fundamental step in the NLP pipeline.
- But extracting syntactic features takes a long time!

So, why not try Pure Neural Methods?

- Inferior performance without prior knowledge of syntax.
- Labeled data is limited in size to train a large model.

	# of Articles	# of Sentences	# of EDUs
Train Set	347	6132	18765
Test Set	38	991	2346

Our Neural Discourse Segmentor

- Remove all syntactic features!

  ⟹ Speedup!

- Transfer word representations learned from large corpus.

  ⟹ For data insufficiency.

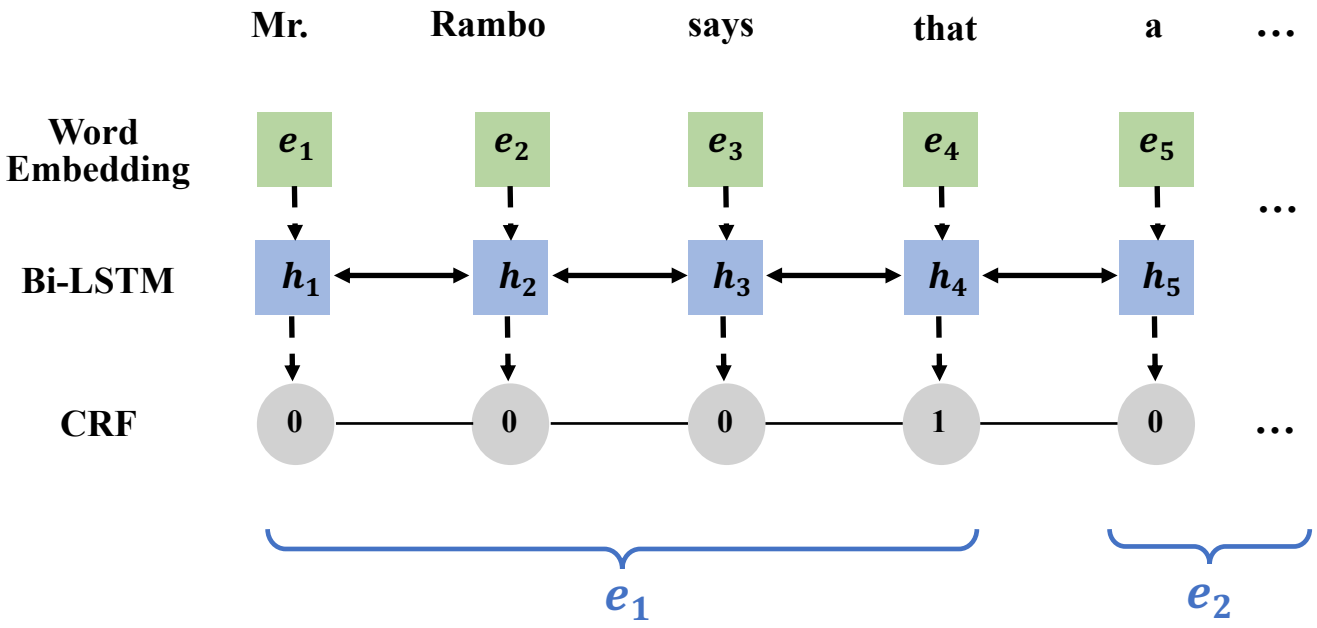
- Use self-attention to model long-range information.

  ⟹ Improve the capacity of our model!

- Restrict the self-attention to a neighborhood.

  ⟹ avoid unnecessary faraway noises.

BiLSTM-CRF for Discourse Segmentation



- Encoding the text with Bi-LSTM:

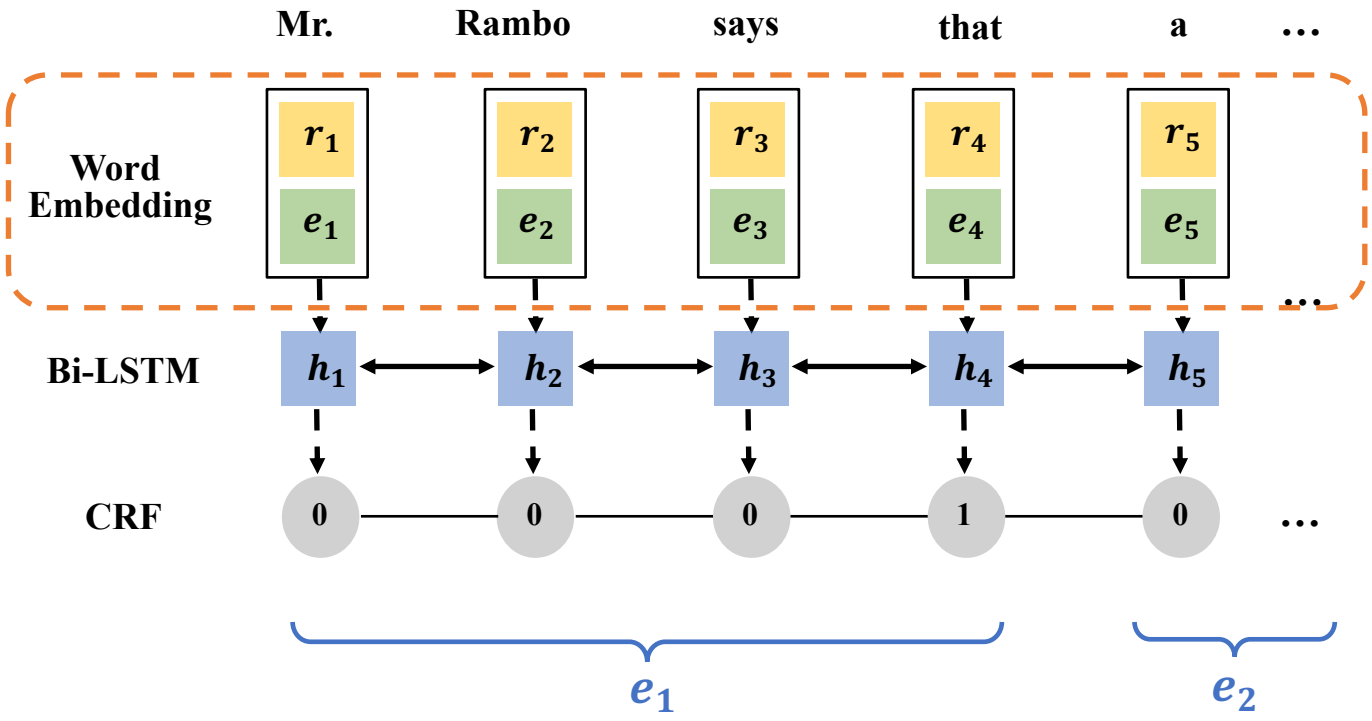
$$\mathbf{h}_t = \mathrm{BiLSTM}(\mathbf{h}_{t-1}, \mathbf{e}_t)$$

- Output layer with CRF:

$$p(\mathbf{y}|\mathbf{h}; \mathbf{W}, \mathbf{b}) = \frac{\prod_{i=1}^{n} \psi_i(y_{i-1}, y_i, \mathbf{h})}{\sum_{y' \in \mathcal{Y}} \prod_{i=1}^{n} \psi_i(y'_{i-1}, y'_i, \mathbf{h})}$$

$$\psi_i(y_{i-1}, y_i, \mathbf{h}) = \exp(\mathbf{w}^T \mathbf{h}_i + b)$$

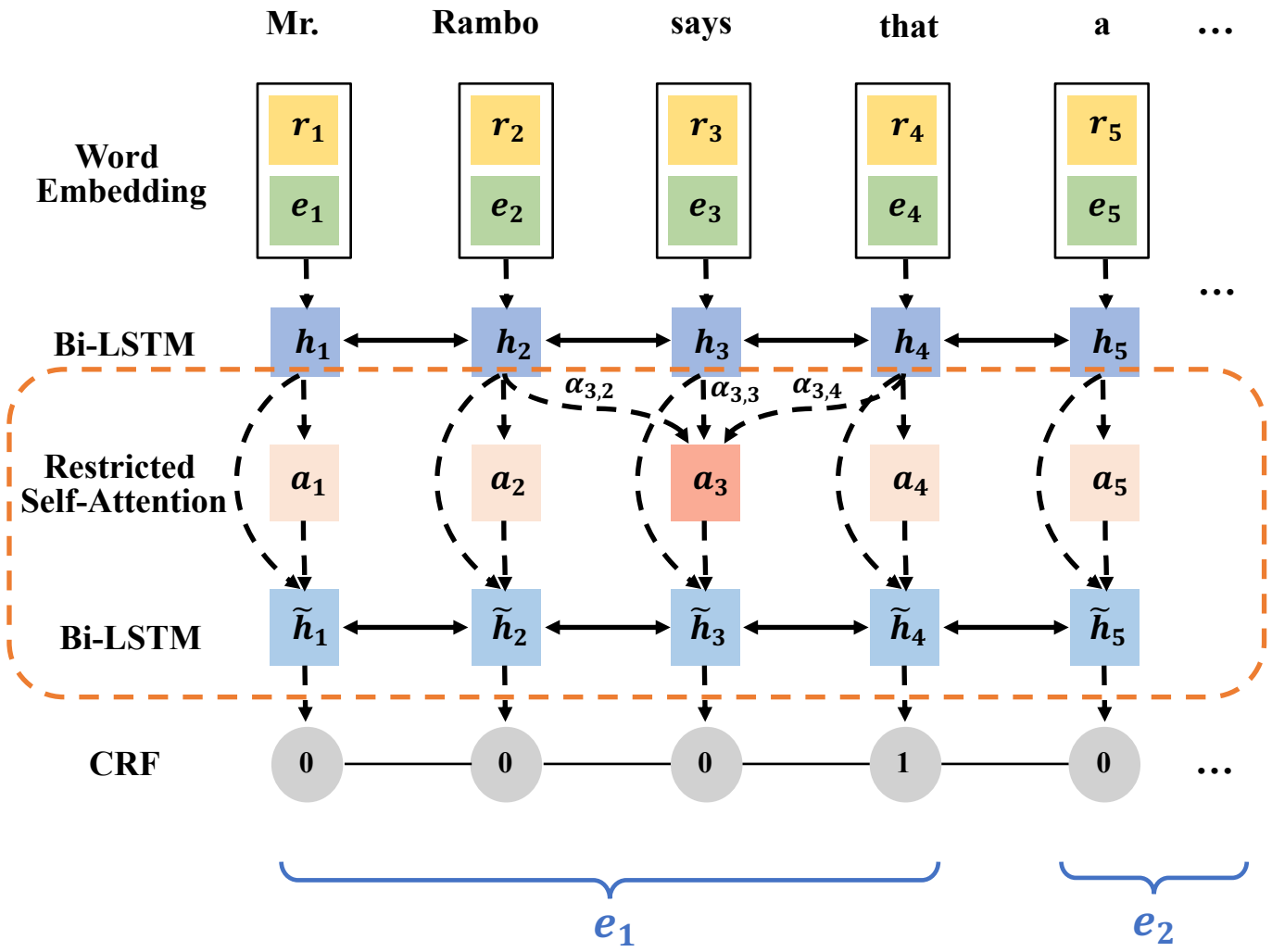
Transferring Representations from LM (ELMo)



- Concatenate ELMo embeddings r_t with GloVe word embeddings e_t :

$$\mathbf{r}_t = \gamma^{\text{LM}} \sum_{l=0}^{3} s_l^{\text{LM}} \mathbf{h}_{t,l}^{\text{LM}}$$

Restricted Self-Attention within a Window



- Compute similarity between current word and nearby words within window K:

$$s_{i,j} = \mathbf{w}_{attn}^T [\mathbf{h}_i, \mathbf{h}_j, \mathbf{h}_i \odot \mathbf{h}_j]$$

- Attention vector a_i is a weighted sum of nearby words:

$$\alpha_{i,j} = \frac{e^{s_{i,j}}}{\sum_{k=-K}^{K} e^{s_{i,i+k}}}$$

$$\mathbf{a}_i = \sum_{j=-K}^{K} \alpha_{i,i+k} \mathbf{h}_{i+k}$$

- Fuse the vectors with another BiLSTM.

Performance on RST Discourse Treebank

Table 2. Performance of our model and other systems

Model	Tree	P(%)	R(%)	F1(%)
SPADE	Gold	84.1	85.4	84.7
NNDS	Gold	85.5	86.6	86.0
CRFSeg	Gold	92.7	89.7	91.2
Reranking	Gold	**93.1**	94.2	93.7
CRFSeg	Stanford	91.0	87.2	89.0
CODRA	BLLIP	88.0	92.3	90.1
Reranking	Stanford	91.5	90.4	91.0
Two-Pass	BLLIP	92.8	92.3	92.6
Our Model	No	92.9	**95.7**	**94.3**
Human	No	98.5	98.2	98.3

- SOTA performance
- F1 + 0.6, compared with methods with gold parse tree
- F1 + 1.7, compared with methods with predicted parse tree

Speed Comparison

Table 3. Speed of our model and two open-source segmentors

System	Speed (Sents/s)	Speedup
Two-Pass	1.39	1.0x
SPADE	3.78	2.7x
Ours (Batch=1)	9.09	6.5x
Ours (Batch=32)	76.23	54.8x

Note: These systems are tested on the same machine (CPU: Intel Xeon E5-2690, GPU: NVIDIA Tesla P100)

Further Analysis

Table 4. Ablation Study

Model	Tree	P(%)	R(%)	F1(%)
Our Model	No	92.9	**95.7**	**94.3**
- Attention	No	92.4	94.8	93.6
- ELMo	No	87.9	84.5	86.2
- Both	No	87.0	82.8	84.8
Human	No	98.5	98.2	98.3

Table 5. Performance with different window size

Window Size	1	5	10	∞
F1-score	94.0	94.3	94.2	93.8

Conclusion

- A pure neural discourse segmentor with:
  - SOTA performance
  - Great speedup
- We show that:
  - Transferred word representations is very useful!
  - Restricted self-attention to a neighborhood can improve the performance.
- Our EDU segmentor is released at:

  https://github.com/yizhongw/neural-edu-segmentation

Thank you!