Second-Order Semantic Dependency Parsing with End-to-End Neural Networks

Xinyu Wang · Jingxian Huang · Kewei Tu

School of Information Science and Technology, ShanghaiTech University

Paper:

Code:

Semantic Dependency Parsing

Semantic dependency parsing aims to identify semantic relationships between words in a sentence that form a graph.

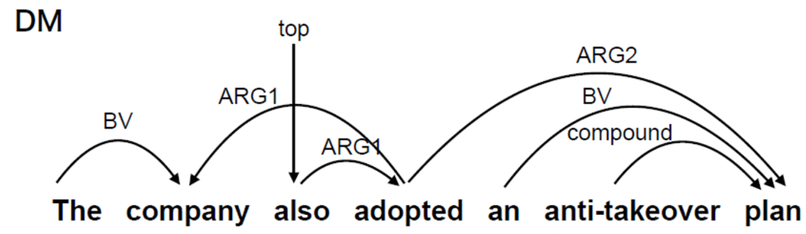


Figure 1. An Example of DM formalism.

First-Order vs. Second-Order

A first-order parser takes only individual dependency edges into consideration while a second-order parser considers the interations between edges.

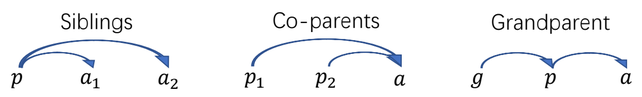


Figure 2. Second-order parts used in our model.

Part Scoring

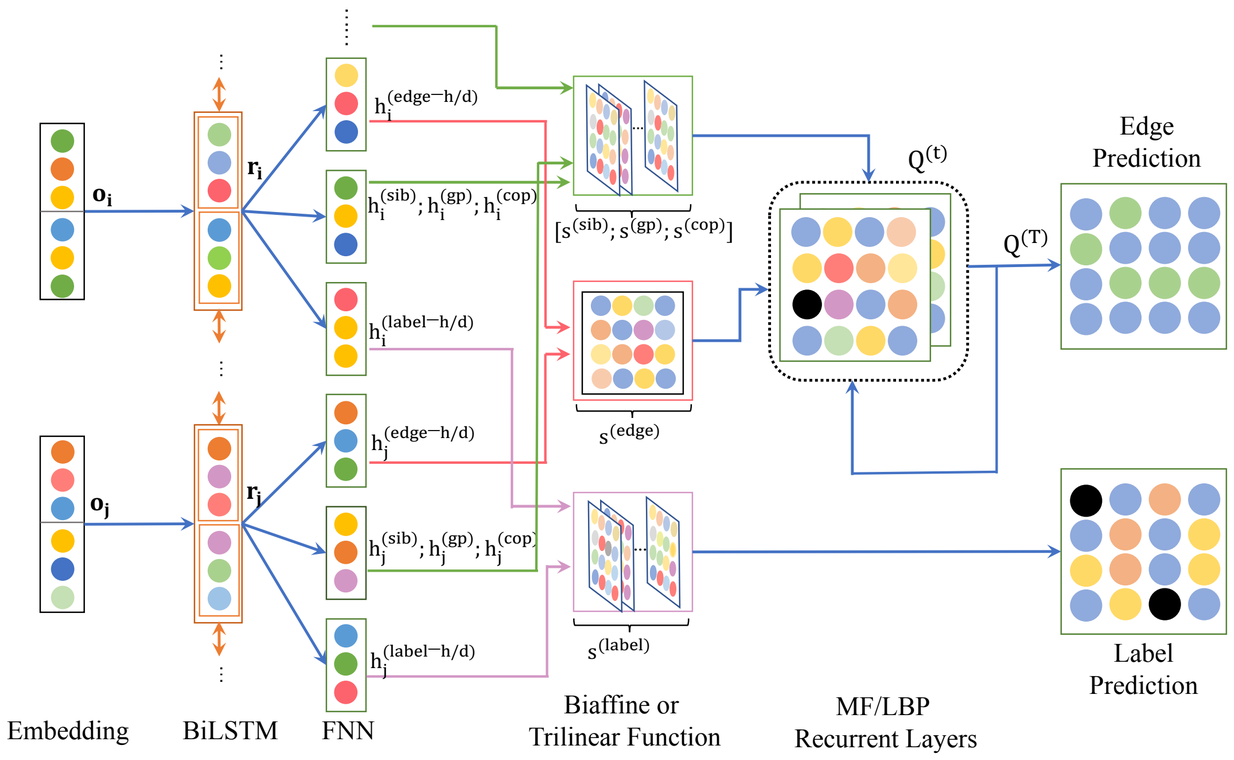
We apply a BiLSTM to the input sentence and then apply a biaffine and a trilinear function to the BiLSTM outputs **h** to compute scores of first-order and second-order parts respectively.

$$\mathbf{Biaff}(\mathbf{h}_1, \mathbf{h}_2) := \mathbf{h}_1^T \mathbf{U} \mathbf{h}_2 + \mathbf{b}$$

$$\mathbf{Trilin}(\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3) := \mathbf{h}_3^T \mathbf{h}_1^T \mathbf{W} \mathbf{h}_2$$

U and **W** are weight tensors.

Model Structure



Conditional Random Field

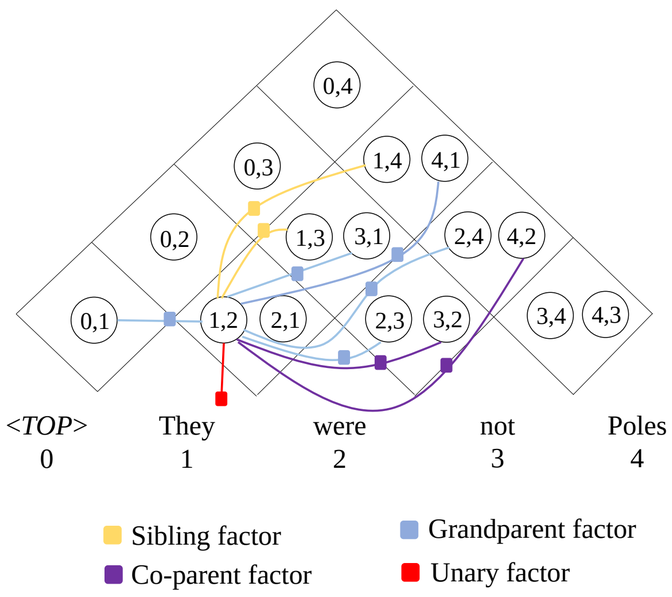


Figure 3. In the edge-prediction module, computing the edge probabilities can be seen as doing posterior inference on a Conditional Random Field (CRF). The boolean variable (i, j) indicates whether the directed edge (i, j) exists.

Inference as RNN

Mean Field Variational Inference:

$$\mathcal{F}_{ij}^{(t-1)} = \sum_{k \neq i,j} Q_{ik}^{(t-1)}(1) s_{ij,ik}^{(sib)} + Q_{kj}^{(t-1)}(1) s_{ij,kj}^{(cop)} + Q_{jk}^{(t-1)}(1) s_{ij,jk}^{(gp)} + Q_{ki}^{(t-1)}(1) s_{ki,ij}^{(gp)}$$

$$Q_{ij}^{(t)}(0) \propto 1$$

$$Q_{ij}^{(t)}(1) \propto \exp\{s_{ij}^{(\text{edge})} + \mathcal{F}_{ij}^{(t-1)}\}$$

Each iteration can be seen as a recurrent neural network layer. Therefore, the whole model is an end-to-end neural network trainable by gradient descent. Inference with loopy belief propagation can be similarly converted to an RNN.

Experiments and Analysis

	DM		PAS		PSD		Avg	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD
Du et al. (2015)	89.1	81.8	91.3	87.2	75.7	73.3	85.3	80.8
A&M, (2015)	88.2	81.8	90.9	86.9	76.4	74.8	85.2	81.2
WCGL, (2018)	90.3	84.9	91.7	87.6	78.6	75.9	86.9	82.8
PTS17: Basic	89.4	84.5	92.2	88.3	77.6	75.3	86.4	82.7
PTS17: Freda3	90.4	85.3	92.7	89.0	78.5	76.4	87.2	83.6
D&M, (2018): Basic	91.4	86.9	93.9	90.8	79.1	77.5	88.1	85.0
Baseline: Basic	92.6	88.0	94.1	91.0	80.6	78.5	89.1	85.8
MF: Basic	**93.0**	**88.4**	**94.3**	**91.5**	80.9	**78.9**	**89.4**	**86.3**
LBP: Basic	92.9	**88.4**	**94.3**	**91.5**	**81.0**	78.8	**89.4**	86.2
D&M, (2018): +Char +Lemma	93.7	88.9	93.9	90.6	81.0	79.4	89.5	86.3
Baseline: +Char +Lemma	93.7	89.4	94.1	90.9	81.0	79.5	89.6	86.6
MF: +Char +Lemma	**94.0**	**89.7**	94.1	**91.3**	**81.4**	**79.6**	**89.8**	**86.9**
LBP: +Char +Lemma	93.9	89.5	**94.2**	**91.3**	**81.4**	79.5	**89.8**	86.8

Figure 5. Comparison of labeled F1 scores achieved by our model and previous state-of-the-arts.

	DM		PAS		PSD		Avg	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD
Baseline: 70%	92.0	87.0	93.8	90.6	79.8	77.7	88.5	85.1
MF: 70%	**92.4**	**87.5**	93.9	90.8	**80.2**	78.0	**88.8**	85.4
LBP: 70%	92.3	87.4	**94.0**	**90.9**	**80.2**	**78.1**	**88.8**	**85.5**
Baseline: 40%	90.8	85.5	93.2	89.6	78.4	76.4	87.4	83.8
MF: 40%	**91.2**	**86.0**	93.4	**90.0**	**78.9**	76.7	87.8	84.2
LBP: 40%	**91.2**	**86.0**	**93.5**	**90.0**	**78.9**	**76.8**	**87.9**	**84.3**
Baseline: 10%	86.1	80.0	90.8	86.4	73.5	71.2	83.4	79.2
MF: 10%	**86.9**	**81.0**	**91.3**	**87.1**	**74.5**	72.1	**84.2**	**80.1**
LBP: 10%	86.8	80.9	**91.3**	87.0	**74.5**	**72.3**	**84.2**	**80.1**

Figure 6. Comparison of labeled F1 scores achieved by our model and our baseline on 10%, 40%, 70% of the training data.

Case Study

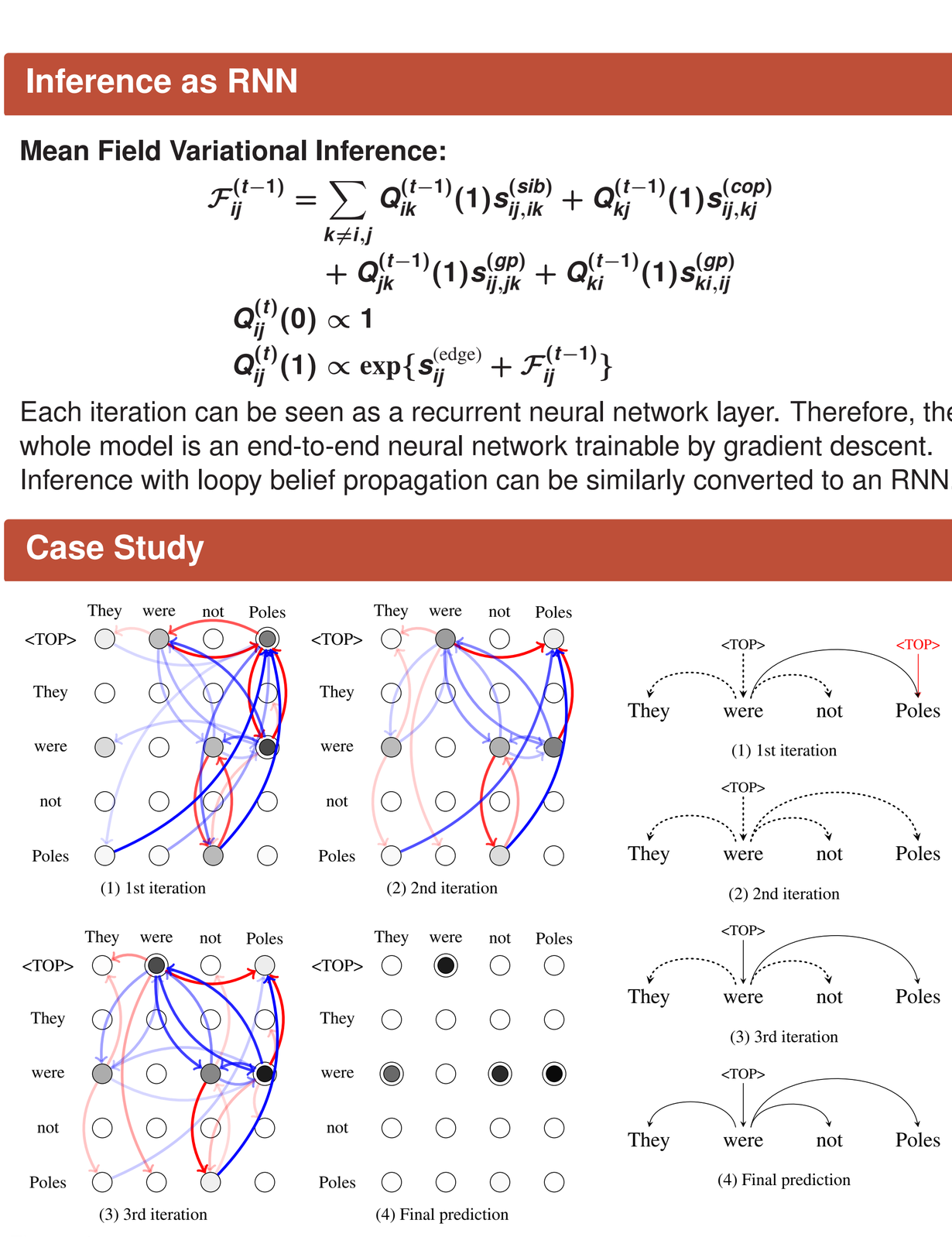


Figure 4. An example of message passing and the corresponding graph parses (right) in our second-order parser with mean field variational inference. Blue arcs and red arcs on the left represent positive messages and negative messages respectively. Blackness of each nodes represents the probability of edge existence. A Node with a double circle means the corresponding edge is predicted to exist. Dotted arcs and red arcs on the right represent missed predictions and wrong predictions compared to the golden parse.

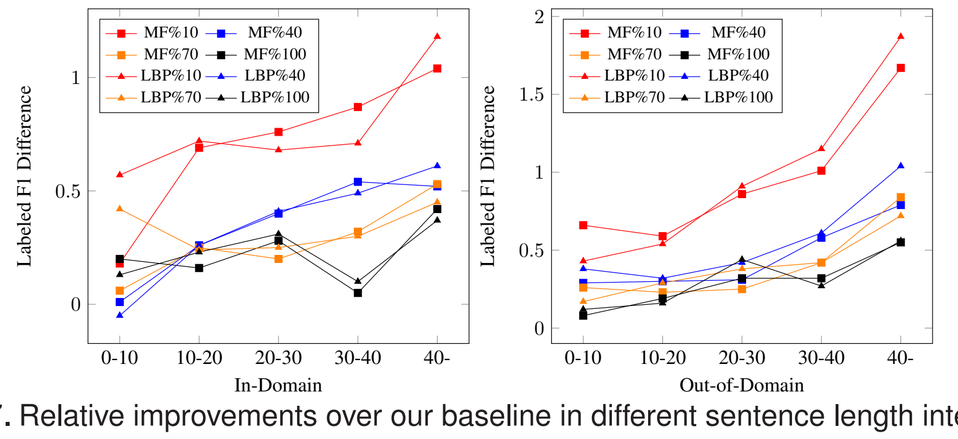


Figure 7. Relative improvements over our baseline in different sentence length intervals with different training data sizes.