CRF Autoencoder for Unsupervised Dependency Parsing

Jiong Cai · Yong Jiang · Kewei Tu

School of Information Science and Technology
ShanghaiTech University, Shanghai, China

Motivation

Almost all previous work on upsupervised dependency parsing focuses on learning generative models. We propose an discriminative and tractable model for this task.

Proposed Solution

We develop an unsupervised dependency parsing model based on the CRF autoencoder. We propose an exact algorithm for parsing as well as a tractable learning algorithm.

Empirical Results

We evaluated the performance of our model on eight multilingual treebanks and found that our model achieved comparable performance with state-of-the-art approaches.

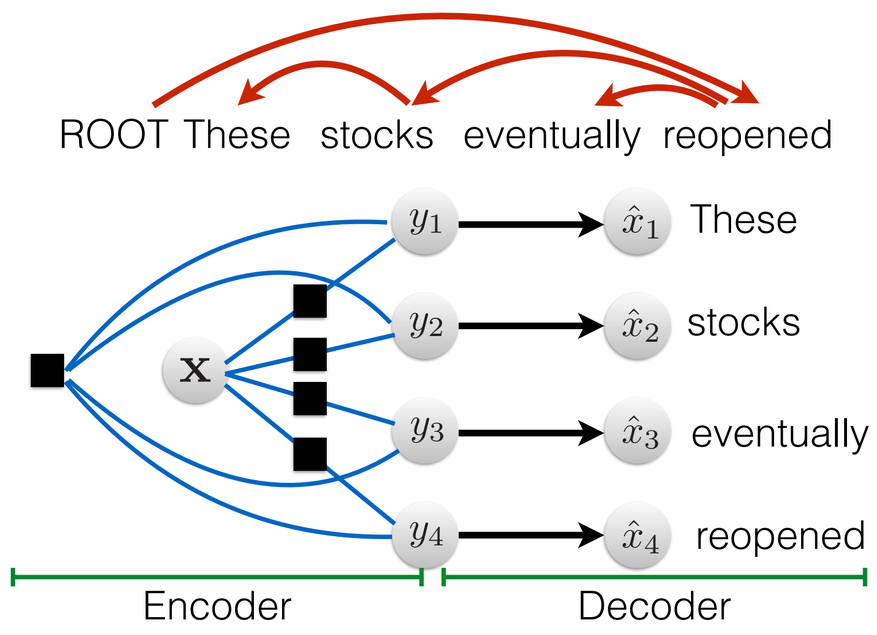
Model

Our model based on CRF autoencoder consists of two part:

- Encoder: a log-linear model represented by a first-order dependency parser. The score of a dependency tree can be factorized as the sum of scores of its dependencies. The encoder models the distribution $P(\mathbf{y}|\mathbf{x})$.
- Decoder: a set of categorical conditional distributions $\theta_{x|t}$, which represents the probability of generating token x conditioned on the token of its parent t based on its parse tree y. The decoder models the distribution $P(\hat{\mathbf{x}}|\mathbf{y})$.

The conditional distribution of $\hat{\mathbf{x}}, \mathbf{y}$ given $\mathbf{x}$ is modeled by the whole model.

$$P(\mathbf{y}, \hat{\mathbf{x}}|\mathbf{x}) = P(\mathbf{y}|\mathbf{x})P(\hat{\mathbf{x}}|\mathbf{y})$$



Training

The goal of the training procedure is to maximize the Vertibi-like conditional reconstruction probability.

Our objective function is

$$J(\mathbf{w}, \theta) = -\sum_{i=1}^{N} \log \left(\max_{\mathbf{y} \in \mathcal{Y}(\mathbf{x}_i)} P(\hat{\mathbf{x}}_i, \mathbf{y}|\mathbf{x}_i) Q^{\alpha}(\mathbf{x}_i, \mathbf{y}) \right) + \lambda \Omega(\mathbf{w})$$

- λ: a hyper-parameter controlling the strength of regularization
- $\Omega(\mathbf{w})$: the regularization term of the encoder parameter $\mathbf{w}$
- α: a hyper-parameter controlling the strength of the constraint factor
- $Q(\mathbf{x}_i, \mathbf{y})$: a soft constraint factor over the parse tree
  - The purpose of adding the soft constraint factor $Q(\mathbf{x}_i, \mathbf{y})$ to the objective function is to encourage learning of dependency relations that satisfy universal linguistic knowledge. The constraint factor $Q(\mathbf{x}_i, \mathbf{y})$ is calculated based on the universal syntactic rules following Naseem et al. (2010) and Grave et al. (2015).

We apply coordinate descent to minimize the objective function, which alternately updates parameters of encoder $\mathbf{w}$ and parameters of decoder θ.

- In each optimization step of $\mathbf{w}$, we run several epochs of stochastic gradient descent,
- In each optimization step of θ, we run several iterations of the Viterbi EM algorithm.

Parsing

$$\mathbf{y}^* = \arg \max_{\mathbf{y} \in \mathcal{Y}(\mathbf{x})} \log P(\hat{\mathbf{x}}, \mathbf{y}|\mathbf{x})$$

The conditional reconstruction probability is decomposable by edges and therefore our model can be seen as a first-order graph-based dependency parser. We can use Eisner's algorithm for projective dependency parsing and the MST algorithm for non-projective dependency parsing.

Results on the WSJ dataset

We evaluated our model on the Wall Street Journal corpus. We trained our model on section 2-21, tuned the hyperparameters on section 22, and tested our model on section 23. We compare the performances with the other methods having similar settings.

Methods	WSJ10	WSJ
Basic Setup		
Feature DMV	63.0	-
UR-A E-DMV	71.4	57.0
Neural E-DMV	69.7	52.5
Neural E-DMV (Good Init)	72.5	57.6
Basic Setup + Universal Linguistic Prior		
Convex-MST	60.8	48.6
HDP-DEP	71.9	-
CRFAE	71.7	55.7
Systems Using Extra Info		
LexTSG-DMV	67.7	55.7
CS	72.0	64.4
MaxEnc	73.2	65.8

Table 1. Comparison of recent unsupervised dependency parsing systems on English. Basic setup is the same as our setup except that linguistic prior is not used. Extra info includes lexicalization, longer training sentences, etc.

Results on Seven Other Languages

We evaluated our model on seven languages from the PASCAL Challenge on Grammar Induction.

	Basque	Czech	Danish	Dutch	Portuguese	Slovene	Swedish	Avg
Length: ≤ 10								
DMV(EM)	41.1	31.3	50.8	47.1	36.7	36.7	43.5	41.0
DMV(Viterbi)	47.1	27.1	39.1	37.1	32.3	23.7	42.6	35.5
Neural DMV (EM)	46.5	33.1	**55.6**	**49.0**	30.4	42.2	44.3	43.0
Neural DMV (Viterbi)	48.1	28.6	39.8	37.2	36.5	39.9	47.9	39.7
Convex-MST (No Prior)	29.4	36.5	49.3	31.3	46.4	33.7	35.5	37.4
Convex-MST (With Prior)	30.0	46.1	51.6	35.3	55.4	**63.7**	50.9	47.5
CRFAE (No Prior)	49.0	33.9	28.8	39.3	47.6	34.7	51.3	40.6
CRFAE (With Prior)	**49.9**	**48.1**	53.4	43.9	**68.0**	52.5	**64.7**	**54.3**
Length: All								
DMV(EM)	31.2	28.1	40.3	44.2	23.5	25.2	32.0	32.0
DMV(Viterbi)	40.9	20.4	32.6	33.0	26.9	16.5	36.2	29.5
Neural DMV (EM)	38.5	29.3	**46.1**	**46.2**	16.2	36.6	32.8	35.1
Neural DMV (Viterbi)	**41.8**	23.8	34.2	33.6	29.4	30.8	40.2	33.4
Convex-MST (No Prior)	30.5	33.4	44.2	29.3	38.3	32.2	28.3	33.7
Convex-MST (With Prior)	30.6	**40.0**	45.8	35.6	46.3	**51.8**	40.5	41.5
CRFAE (No Prior)	39.8	25.4	24.2	35.2	52.2	26.4	40.0	34.7
CRFAE (With Prior)	41.4	36.8	40.5	38.6	**58.9**	43.3	**48.5**	**44.0**

Table 2. Parsing accuracy on seven languages. Our model is compared with DMV, Neural DMV, and Convex-MST

Future work

- Using higher order dependency models as the encoder.
- Introducing neural networks into the encoder.