Sequence Labeling & more! (III)

CS 690N, Spring 2017

Advanced Natural Language Processing

http://people.cs.umass.edu/~brenocon/anlp2017/

Brendan O'Connor

College of Information and Computer Sciences

University of Massachusetts Amherst

Random project idea

- Temporal relations
- Goal: extract events from text and have on timeline -- or at least a partial order

- e.g. the TimeBank-Dense dataset
  - "Before <u>1993</u>, she <u>attended</u>..."
    - => *BEFORE(attend, 1993-01-01)*
  - "I <u>got</u> in a car and <u>drove</u>"
    - => *BEFORE(car, drove)*
- Aspectual/subordinate/factive relations
  - Hold between events
  - *"I <u>doubt</u> I <u>left</u> them there"*

Temporal Relations

- BEFORE, AFTER, DURING
  - R(evt, time)
  - R(evt, evt)
- Logical implications: e.g. transitivity

Relation	Illustration	Interpretation
$X < Y$ $Y > X$	X Y	X takes place before Y
$X \mathbf{m} Y$ $Y \mathbf{mi} X$	X Y	X meets Y (*i* stands for *inverse*)
$X \mathbf{o} Y$ $Y \mathbf{oi} X$	X Y	X overlaps with Y
$X \mathbf{s} Y$ $Y \mathbf{si} X$	X Y	X starts Y
$X \mathbf{d} Y$ $Y \mathbf{di} X$	X Y	X during Y
$X \mathbf{f} Y$ $Y \mathbf{fi} X$	X Y	X finishes Y
$X = Y$	X Y	X is equal to Y

[Allen's interval algebra]

Forward-Backward

- Purpose: compute
  - Tag marginals $p(y_t \mid w)$
  - Pair marginals $p(y_{t-1}, y_t \mid w)$
- Why?
  - Min Bayes Risk decoding
    - For each *t*, choose: $\mathrm{argmax}_k\ p(y_t=k \mid w)$
  - E-step for EM learning of unsupervised HMM
  - Feature expectations for supervised CRF

Generalized CRF

$$\psi_c(y_c) = \theta^\mathsf{T} f_c(y_c, x)$$

$$p(y \mid x) \propto \exp\left(\sum_c \psi_c(y_c)\right)$$

- Clique *c*: set of random variables
- ψ_C: soft constraint (logprob) among y_c
- Linear chain CRF: neighboring cliques only
- Many others possible!
  - Higher order Markov
  - Global document information
    - e.g. repeated words tend to have same label: one-sense-per-discourse or coreference

Learning a CRF

$$\log p_\theta(y \mid w) = \theta^\mathsf{T} f(y, w) - \log \sum_{y'} \exp(\theta^\mathsf{T} f(y, w))$$

$$\frac{\partial \log p_\theta(...)}{\partial \theta_j} = f_j(y, w) - \sum_{y'} p_\theta(y' \mid w) f_j(y', w)$$

- Apply local decomposition

Learning a CRF

$$\log p_\theta(y \mid w) = \theta^\mathsf{T} f(y, w) - \log \sum_{y'} \exp(\theta^\mathsf{T} f(y, w))$$

$$\frac{\partial \log p_\theta(...)}{\partial \theta_j} = f_j(y, w) - \sum_{y'} p_\theta(y' \mid w) f_j(y', w)$$

- Apply local decomposition

$$= \left(\sum_t f_j(y_{t-1}, y_t, w_t) \right) - \sum_{y'} p_\theta(y' \mid w) \sum_t f_j(y'_{t-1}, y'_t, w_t)$$

Learning a CRF

$$\log p_\theta(y \mid w) = \theta^\mathsf{T} f(y, w) - \log \sum_{y'} \exp(\theta^\mathsf{T} f(y, w))$$

$$\frac{\partial \log p_\theta(...)}{\partial \theta_j} = f_j(y, w) - \sum_{y'} p_\theta(y' \mid w) f_j(y', w)$$

- Apply local decomposition

$$= \left(\sum_t f_j(y_{t-1}, y_t, w_t) \right) - \sum_{y'} p_\theta(y' \mid w) \sum_t f_j(y'_{t-1}, y'_t, w_t)$$

$$= \sum_t \left(f_j(y_{t-1}, y_t, w_t) - \sum_{y'_t, y'_{t-1}} p_\theta(y'_{t-1}, y'_t \mid w) f_j(y'_{t-1}, y'_t, w_t) \right)$$

Learning a CRF

$$\log p_\theta(y \mid w) = \theta^\mathsf{T} f(y,w) - \log \sum_{y'} \exp(\theta^\mathsf{T} f(y,w))$$

$$\frac{\partial \log p_\theta(\ldots)}{\partial \theta_j} = f_j(y,w) - \sum_{y'} p_\theta(y' \mid w) f_j(y',w)$$

- Apply local decomposition

$$= \left(\sum_t f_j(y_{t-1}, y_t, w_t) \right) - \sum_{y'} p_\theta(y' \mid w) \sum_t f_j(y'_{t-1}, y'_t, w_t)$$

Real feature value

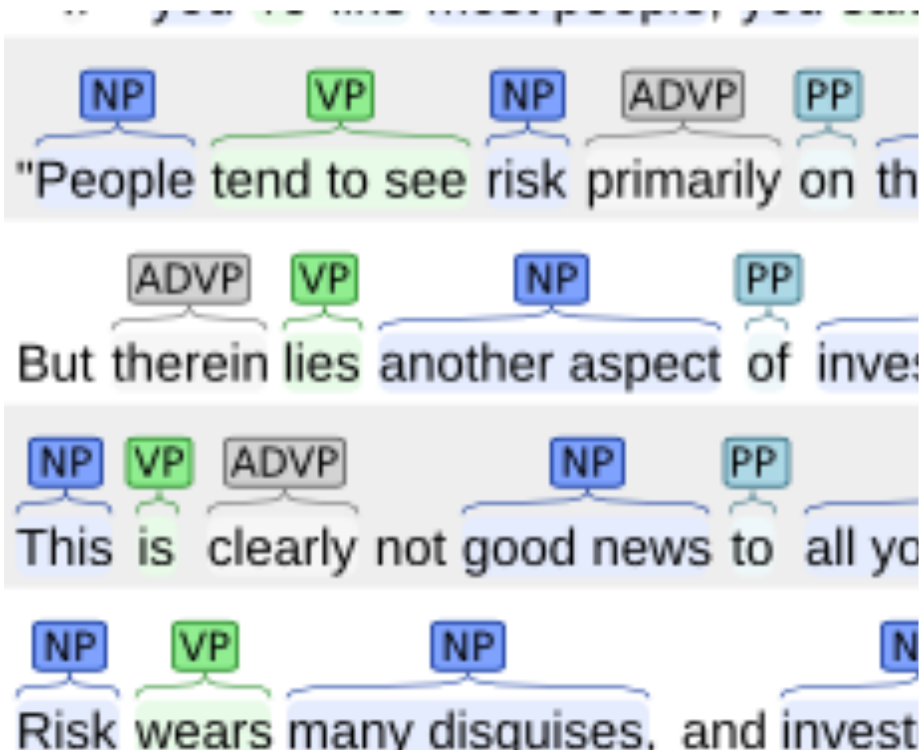
Expected feature value

$$= \sum_t \left(f_j(y_{t-1}, y_t, w_t) - \sum_{y'_t, y'_{t-1}} p_\theta(y'_{t-1}, y'_t \mid w) f_j(y'_{t-1}, y'_t, w_t) \right)$$

Tag marginals (to compute: forward-backward)

Semi-Markov CRF

- *[Sarawagi and Cohen, 2004]*
- Instead of sequence labels, assume variable length *segments*
  - s_j = (start, end, label)
  - All positions covered by non-overlapping segments



- Allows natural whole-segment features, e.g. "are all words in this span capitalized?"
- Inference for max-length L: LKN
(contrast to L-th order Markov model)

Inference

- Viterbi: V(i,y) = best prob of path up to *i*, starting segment *y*

$$V(i,y) = \begin{cases} \max_{y',d=1\ldots L} V(i-d,y') + \mathbf{W}\cdot\mathbf{g}(y,y',\mathbf{x},i-d,i) & \text{if } i>0 \\ 0 & \text{if } i=0 \\ -\infty & \text{if } i<0 \end{cases}$$

- Forward: a(i,y) = sum of path probs up to *i*, where *i* is starting a segment *y*

$$\alpha(i,y) = \sum_{d=1}^{L}\sum_{y'\in\mathcal{Y}} \alpha(i-d,y')e^{\mathbf{W}\cdot\mathbf{g}(y,y',\mathbf{x},i-d,i)}$$

Alternate training

- Standard CRF training: NLL loss

$$-\log p(y \mid x) = -\theta^{\mathsf{T}} f(x,y) + \log \sum_{y'} \exp\left(\theta^{\mathsf{T}} f(x,y')\right)$$

$$\frac{\partial}{\partial \theta}(-\log p(y \mid x)) = -f(x,y) + E_{y' \sim p_\theta(y|x)}[f(x,y')]$$

Alternate training

- Standard CRF training: NLL loss

$$-\log p(y \mid x) = -\theta^\mathsf{T} f(x,y) + \log \sum_{y'} \exp\left(\theta^\mathsf{T} f(x,y')\right)$$

$$\frac{\partial}{\partial \theta}(-\log p(y \mid x)) = -f(x,y) + E_{y' \sim p_\theta(y|x)}[f(x,y')]$$

- Structured perceptron loss

$$L_{perc}(y) = -\theta^\mathsf{T} f(x,y) + \max_{y'} \theta^\mathsf{T} f(x,y')$$

- => gradient:

Alternate training

- Standard CRF training: NLL loss

$$-\log p(y \mid x) = -\theta^\mathsf{T} f(x,y) + \log \sum_{y'} \exp\left(\theta^\mathsf{T} f(x,y')\right)$$

$$\frac{\partial}{\partial \theta}(-\log p(y \mid x)) = -f(x,y) + E_{y' \sim p_\theta(y|x)}[f(x,y')]$$

- Structured perceptron loss

$$L_{perc}(y) = -\theta^\mathsf{T} f(x,y) + \max_{y'} \theta^\mathsf{T} f(x,y')$$

- => gradient:

$$\frac{\partial}{\partial \theta} L_{perc}(y) = -f(x,y) + f(x,y^*)$$

$$y^* \rightarrow \arg\max_{y'} \theta^\mathsf{T} f(x,y)$$

Alternate training

- Standard CRF training: NLL loss

$$-\log p(y \mid x) = -\theta^\intercal f(x,y) + \log \sum_{y'} \exp\left(\theta^\intercal f(x,y')\right)$$

$$\frac{\partial}{\partial\theta}(-\log p(y \mid x)) = -f(x,y) + E_{y' \sim p_\theta(y|x)}[f(x,y')]$$

- Structured perceptron loss

$$L_{perc}(y) = -\theta^\intercal f(x,y) + \max_{y'} \theta^\intercal f(x,y')$$

- => gradient:

$$\frac{\partial}{\partial\theta} L_{perc}(y) = -f(x,y) + f(x,y^*)$$

$$\arg\max_{y'} \theta^\intercal f(x,y)$$

- SGD => the *structured perceptron* algorithm *[Collins 2002]*
- Advantage: only need a Viterbi algorithm
- Better variant: Cost-augmented perceptron (structured hinge/SVM loss)