Sequence Labeling (III)
Conditional Random Fields

CS 690N, Spring 2018

Advanced Natural Language Processing

http://people.cs.umass.edu/~brenocon/anlp2018/

Brendan O'Connor

College of Information and Computer Sciences

University of Massachusetts Amherst

How to build a POS tagger?

- Sources of information:
    - POS tags of surrounding words: syntactic context
    - The word itself
    - Features, etc.!
        - Word-internal information
        - Features from surrounding words
        - External lexicons
        - Embeddings, LSTM states

← HMM

← Classifier

← CRF

- Seq. labeling as log-linear ***structured prediction***

$$\hat{\boldsymbol{y}}_{1:M} = \underset{\boldsymbol{y}_{1:M} \in \mathcal{Y}(\boldsymbol{w}_{1:M})}{\operatorname{argmax}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}_{1:M}, \boldsymbol{y}_{1:M}),$$

- Example: the ***Hidden Markov model***

$$p(\mathbf{w}, \mathbf{y}) = \prod_t p(y_t \mid y_{t-1}) p(w_t \mid y_t)$$

- Efficiently supports operations via dynamic programming –
  because of **local (Markovian) assumptions**
  - P(w): Likelihood (generative model)
    - Forward algorithm
  - P(y | w): Predicted sequence ("decoding")
    - Viterbi algorithm
  - P(y_m | w): Predicted tag marginals
    - Forward-Backward algorithm
    - Supports EM for unsupervised HMM learning

- Seq. labeling as log-linear ***structured prediction***

$$\hat{\boldsymbol{y}}_{1:M} = \underset{\boldsymbol{y}_{1:M} \in \mathcal{Y}(\boldsymbol{w}_{1:M})}{\operatorname{argmax}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}_{1:M}, \boldsymbol{y}_{1:M}),$$

- Example: the ***Hidden Markov model***

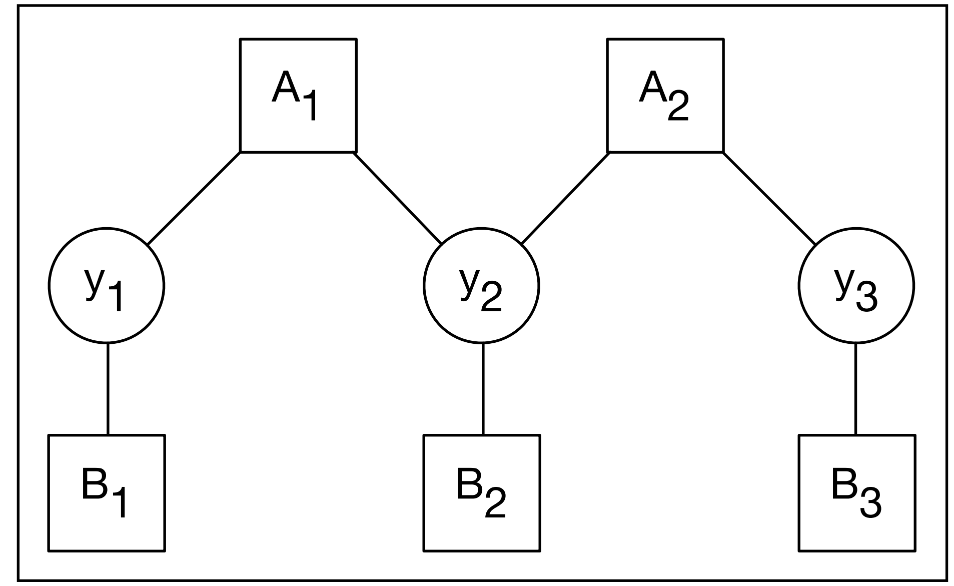
$$p(\mathbf{w}, \mathbf{y}) = \prod_t p(y_t \mid y_{t-1}) p(w_t \mid y_t)$$

- Today: ***Conditional Random Fields***

$$p(\mathbf{y} \mid \mathbf{w}) = \frac{\exp(\theta^\mathsf{T} \mathbf{f}(\mathbf{w}_{1:M}, \mathbf{y}_{1:M}))}{\sum_{\mathbf{y}'_{1:M} \in \mathcal{Y}(\mathbf{w}_{1:M})} \exp(\theta^\mathsf{T} \mathbf{f}(\mathbf{w}_{1:M}, \mathbf{y}'_{1:M}))}$$

* for carefully chosen **f**

HMM as log-linear



$$p(y,w) = \prod_t p(w_y|y_t)\ p(y_t|y_{t-1})$$

$$\underbrace{\log p(y,w)}_{\substack{G(y) \\ \textit{goodness}}} = \sum_t \underbrace{\log p(w_t|y_t)}_{\substack{B_t(y_t) \\ \textit{emission factor score}}} + \underbrace{\log p(y_t|y_{t-1})}_{\substack{A(y_{t-1},y_t) \\ \textit{transition factor score}}}$$

$$\log p(y,w) = \sum_t \underbrace{\phi_t(y_{t-1},y_t)}_{\textit{pair factor score}}$$

Decoding problem (Viterbi algorithm) $\arg \max_{y^* \in outputs(x)} G(y^*)$

HMM as log-linear

- HMM as a joint log-linear model

$$P(y, w) = \prod_t P(y_t \mid y_{t-1}) P(w_t \mid y_t)$$

$$P(y, w) = \exp(\theta^\mathsf{T} f(y, w))$$

$$f(y, w) = \sum_t f(y_{t-1}, y_t, w_t)$$

Local features only!
(Allows efficient inference)

e.g. {(N,V):1, (V,dog):1}
What are the weights?

- This implies the conditional is also log-linear

$$P(y \mid w) \propto \exp(\theta^\mathsf{T} f(y, w))$$

From HMMs to CRFs

- **1. Discriminative learning**: take HMM features, but set weights to maximize *conditional LL* of labels
- **2. More features**: affix, positional, feature templates, embeddings, etc.
    - For efficient inference: make sure to **preserve Markovian structure** within the feature function (e.g. first-order CRF)

Learning a CRF

- Gradient descent on negative **conditional LL**
  - Log-linear gradient:
    sum over all possible predicted structures
    (Forward-Backward for marginalization)
- Non-probabilistic losses: compare gold structure to only one predicted structure
  - Structured perceptron algorithm:
    Collins, 2002 (just got Test of Time award)
  - Structured SVM (hinge loss)
  - (Viterbi for best-structure)

Learning a CRF: max CLL

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

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

- Apply local decomposition

Learning a CRF: max CLL

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

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

- 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: max CLL

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

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

- 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: max CLL

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

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

- Apply local decomposition

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

$$= \sum_t \Bigg(\underbrace{f_j(y_{t-1}, y_t, w_t)}_{\text{Real feature value}} - \underbrace{\sum_{y'_t, y'_{t-1}} \underbrace{p_\theta(y'_{t-1}, y'_t \mid w)}_{\text{Tag marginals (to compute: forward-backward)}} f_j(y'_{t-1}, y'_t, w_t)}_{\text{Expected feature value}} \Bigg)$$

- stopped here 3/27

(Log?-)linear Viterbi

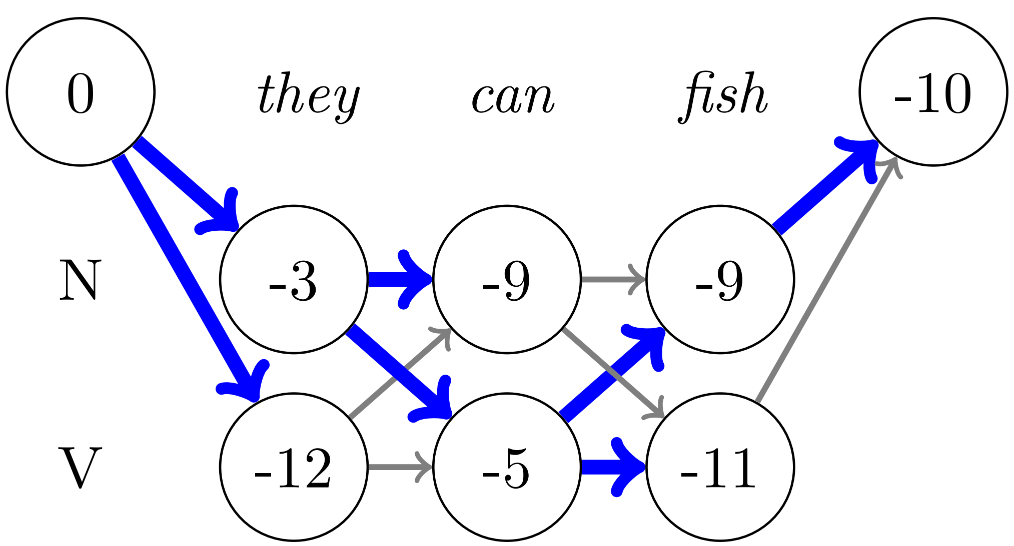
$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{y}) \qquad \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{y}) = \sum_{m=1}^{M} \boldsymbol{f}(\boldsymbol{w}, y_m, y_{m-1}, m).$$

$$\max_{\boldsymbol{y}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{y}) = \max_k v_M(k)$$

Score of best sequence ending in *k*

$$v_m(k) \triangleq \max_{\boldsymbol{y}_{1:m-1}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, k, y_{m-1}, m) + \sum_{n=1}^{m-1} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, y_n, y_{n-1}, n)$$

$$= \max_{y_{m-1}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, k, y_{m-1}, m) + \underbrace{\max_{\boldsymbol{y}_{1:m-2}} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, y_{m-1}, y_{m-2}) + \sum_{n=1}^{m-2} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{w}, y_n, y_{n-1}, n)}_{v_{m-1}(y_{m-1})}$$



	they	*can*	*fish*
N	-2	-3	-3
V	-10	-1	-3

(a) Weights for emission features.

	N	V	♦
◊	-1	-2	$-\infty$
N	-3	-1	-12
V	-1	-3	-1

(b) Weights for transition features.