Sequence Labeling, Contd

Prof. Sameer Singh

CS 295: STATISTICAL NLP

WINTER 2017

February 2, 2017

Outline

Marginal Inference in HMMs: F/B algorithm

Maximum Entropy Markov Models

Conditional Random Fields

Neural Sequence Tagging

Outline

Marginal Inference in HMMs: F/B algorithm

Maximum Entropy Markov Models

Conditional Random Fields

Neural Sequence Tagging

Expectation Maximization

K-Means

Initialization

$e(x_i|y_i) \sim \text{uniform}$

$t(y_i|y_{i-1}) \sim \text{uniform}$

Pick K random centroids

Label Data from the Model

Fix e, t

Compute $p(\vec{y}|\vec{x})$

Cluster all the points

Update the Model from Data

Fix $p(y|x)$

Update e, t

Update centroids

Label Data from the Model

Hard-EM

$$y^* = \arg\max_y p(y|x) = \arg\max_y p(y, x)$$

$$\hat{e}(x_i|y_i) = \frac{\#\, x_i \wedge y_i \text{ in } y^*}{\#\, y_i \text{ in } y^*} \qquad \vec{t} = \ldots$$

Soft-EM

$$p(y_i|\vec{x}) = \sum_{\substack{\vec{y}_j \\ y_{ij} = y_i}} p(\vec{y}_j|\vec{x})$$

$$p(x_i, y_i) = \sum_{y_j} p(x_j = x_i, y_{ij} = y_i)$$

Dynamic Programming

$$p(x_1 \ldots x_n, y_i) = \sum_{y_1 \ldots y_{i-1}} \sum_{y_{i+1} \ldots y_n} p(x_1 \ldots x_n, y_1 \ldots y_n)$$

$$= p(x_1 \ldots x_i, y_i)\, p(x_{i+1} \ldots x_n \mid y_i)$$

$$\alpha(i, y_i) = p(x_1 \ldots x_i, y_i) = \sum_{y_1 \ldots y_{i-1}} p(x_1 \ldots x_i, y_1 \ldots y_i)$$

$$= \sum_{y_{i-1}} e(x_i \mid y_i)\, t(y_i \mid y_{i-1})\, \alpha(i-1, y_{i-1})$$

$$\beta(i, y_i) = p(x_{i+1} \ldots x_n \mid y_i) = \sum_{y_{i+1} \ldots y_n} p(x_{i+1} \ldots x_n, y_{i+1} \ldots y_n)$$

$$= \sum_{y_{i+1}} e(x_{i+1}, y_{i+1})\, t(y_{i+1} \mid y_i)\, \beta(i+1, y_{i+1})$$

Forward Backward Algorithm

Forward

$$\alpha(0, s) = 1$$

$$\alpha(i, y_i) = \sum_{y_{i-1}} e(x_i \mid y_i)\, t(y_i \mid y_{i-1})\, \alpha(i-1, y_{i-1})$$

Backward

$$\beta(n, e) = 1$$

$$\beta(i, y_i) = \sum_{y_{i+1}} e(x_{i+1} \mid y_{i+1})\, t(y_{i+1} \mid y_i)\, \beta(i+1, y_{i+1})$$

Updating the Model from Data

Hard-EM

$$e(x_i \mid y_i) = \frac{\#\, x_i \text{ tagged as } y_i}{\#\, y_i} \qquad t(y_i \mid y_{i-1}) = \frac{\#\, y_{i-1} \rightarrow y_i}{\#\, y_{i-1}}$$

Soft-EM

$$e(x_i \mid y_i) = \sum_j p(x_j = x_i, y_j = y_i \mid x_1 \ldots x_n)$$

$$= \sum_{\substack{j \\ x_j = x_i}} p(y_j = y_i \mid x_1 \ldots x_n)$$

$$p(x_1 \ldots x_n, y_i) = \alpha(i, y_i)\beta(i, y_i)$$

$$p(x_1 \ldots x_n, y_i, y_{i+1}) = \alpha(i, y_i)\, t(y_{i+1} \mid y_i)\, e(x_{i+1} \mid y_{i+1})\, \beta(i+1, y_{i+1})$$

Outline

Marginal Inference in HMMs: F/B algorithm

Maximum Entropy Markov Models

Conditional Random Fields

Neural Sequence Tagging

Named Entity Recognition

George	W.	Bush	spoke	from	the	White	House	today	.
PER	PER	PER	O	O	O	LOC	LOC	O	O
B-PER	I-PER	I-PER	O	O	O	B-LOC	I-LOC	O	O
B-PER	I-PER	E-PER	O	O	O	B-LOC	E-LOC	O	O

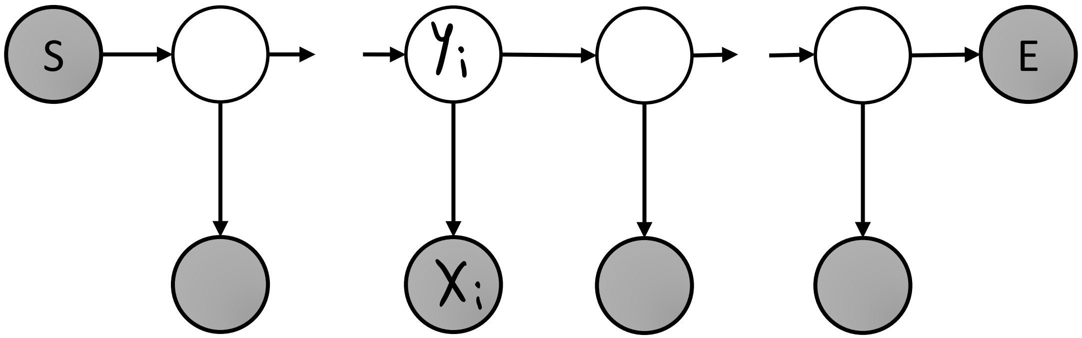
Max. Entropy Markov Models

$$p(\vec{y}|\vec{x}) = \prod_i p(y_i | x_i, y_{i-1})$$

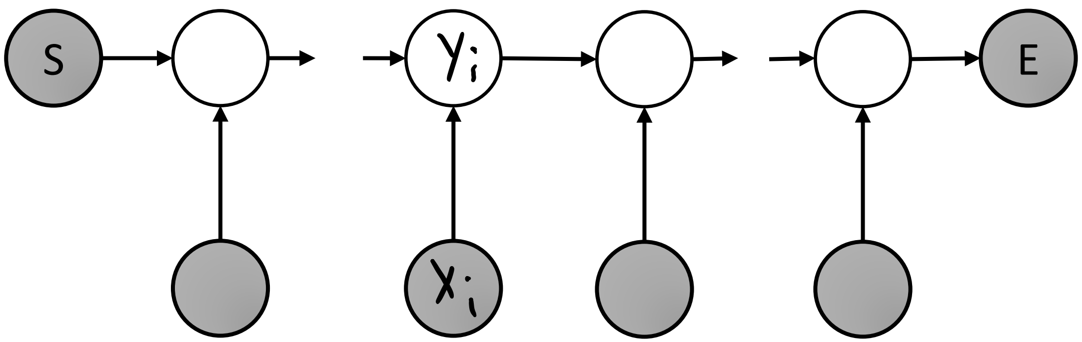
$$p(y_i | x_i, y_{i-1}) = \frac{e^{\theta \cdot \phi(x_i, y_i, y_{i-1})}}{\sum_y e^{\theta \cdot \phi(x_i, y, y_{i-1})}}$$

Graphical Model Notation

HMMs



MEMMs



Adding Features (for POS)

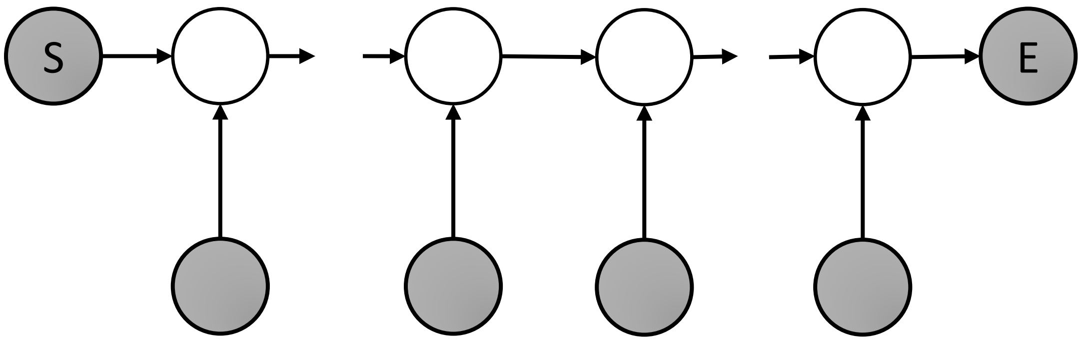
Current Word

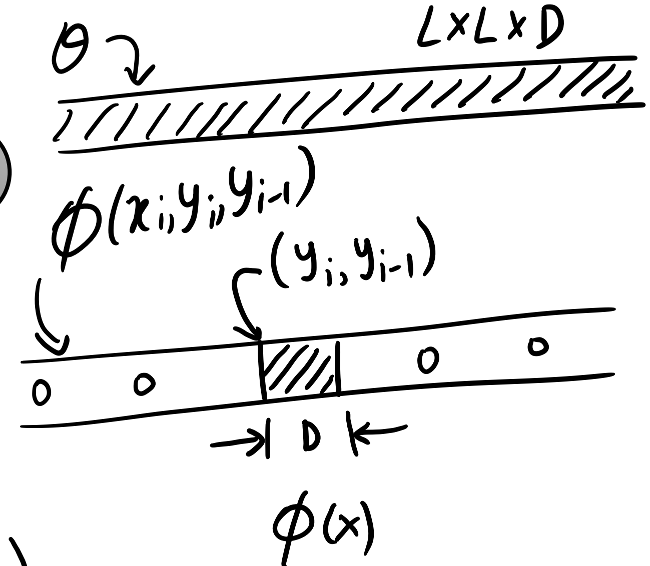
- Word — the: the → DT
- Lowercased word — Importantly: importantly → RB
- Prefixes — unfathomable: un- → JJ
- Suffixes — Surprisingly: -ly → RB
- Capitalization — Meridian: CAP → NNP
- Word shapes — 35-year: d-x → JJ

Window Words

- Add in previous / next word the __
- Previous / next word shapes — X __ X
- Occurrence pattern features — [X: x X occurs]
- Crude entity detection — __ ….. (Inc.|Co.)
- Phrasal verb in sentence? put …… __
- Conjunctions of these things

Adding Features





$$p(y_i \mid x_i, y_{i-1}) = \frac{e^{\theta \cdot \phi(y_i, x_i, y_{i-1})}}{\sum_{y} e^{\theta \cdot \phi(y, x_i, y_{i-1})}}$$

Predictions Using MEMMs

Greedy

$$y_1 = \arg\max_y p(y \mid x_1, s) \quad y_2 = \arg\max_y p(y \mid x_2, y_1) \quad \ldots$$

Viterbi Decoding

$$\vec{y} = \arg\max_y \prod_{i=1}^{n} p(y_i \mid x_i, y_{i-1})$$

$$\pi(i, y_i) = \arg\max_{y_1 \ldots y_{i-1}} \prod_{j=1}^{i-1} p(y_j \mid x_j, y_{j-1})$$

$$= \arg\max_{y_{i-1}} p(y_i \mid x_i, y_{i-1}) \, \pi(i-1, y_{i-1})$$

Training MEMMs

$$\mathcal{L}(\theta, D) = \sum_{d \in D} \log P(\vec{y} \,|\, \vec{x})$$

$$= \sum_{d} \sum_{i} \log \underbrace{P(y_i \,|\, x_i, y_{i-1})}_{\text{Independent}}$$

Train using off the shelf classifiers!

Outline

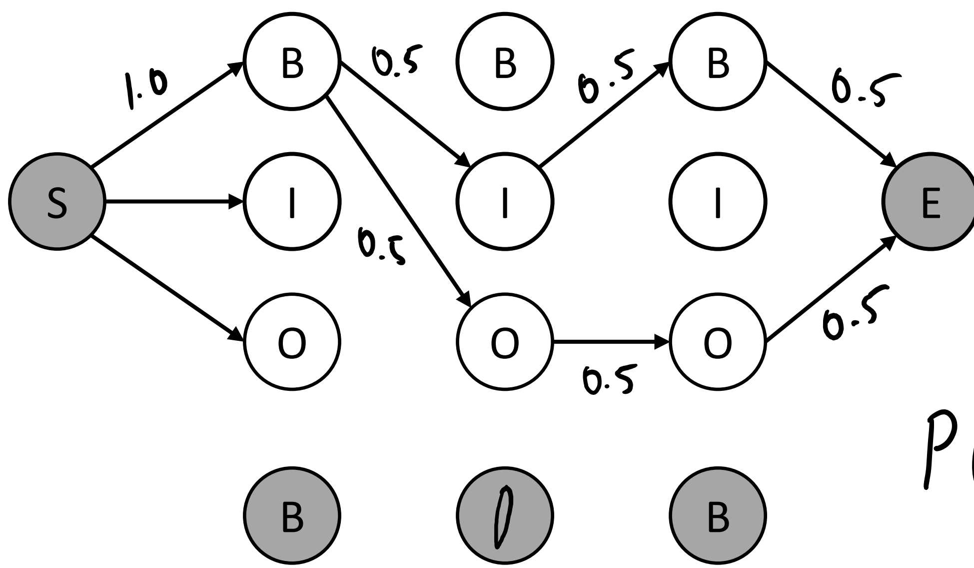
Marginal Inference in HMMs: F/B algorithm

Maximum Entropy Markov Models

Conditional Random Fields

Neural Sequence Tagging

Label Bias Problem



$P(B|\text{"B"}) = 1$

$P(I|\text{"O"}, B) = 0.5$

$P(O|\text{"O"}, B) = 0.5$

$P(B|\text{"B"}, I) = 0.5$

$P(B|B) = 0.5$

$P(BOO) = 0.5$

XX $\longrightarrow$ $P(O|\text{"B"}, O) = 0.5$

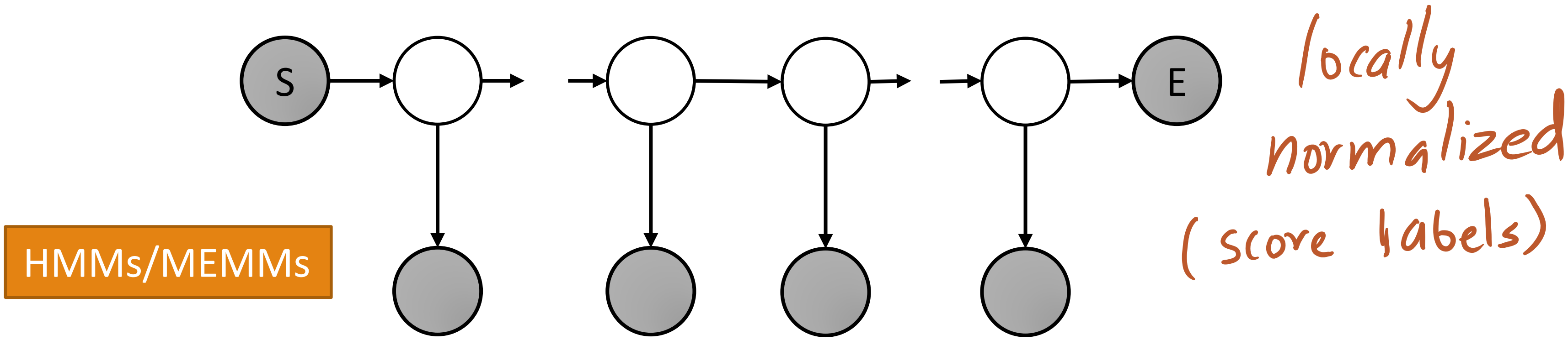
ignores "B"

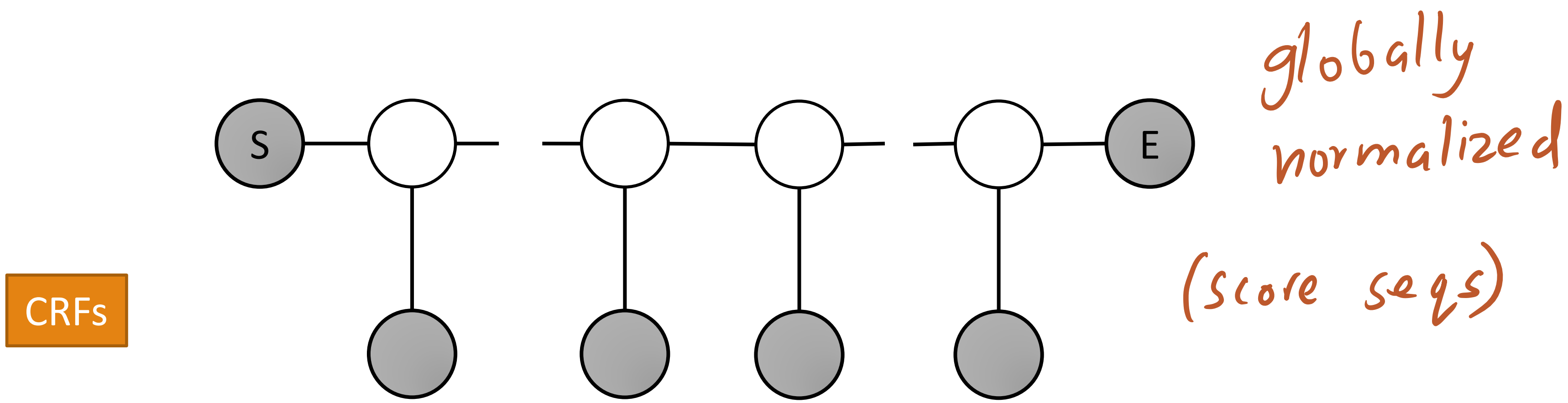
Conditional Random Fields

$$p(\vec{y}|\vec{x}) = \frac{e^{\theta \cdot \Phi(\vec{x},\vec{y})}}{\sum_{\vec{y}'} e^{\theta \cdot \Phi(\vec{x},\vec{y}')}}$$

$$\Phi(\vec{x},\vec{y}) = \overbrace{\underbrace{\sum_{i=1}^{n} \phi_x(x_i, y_i)}_{\text{arbitrary features}} + \underbrace{\sum_{i=2}^{n} \phi_t(y_i, y_{i-1})}_{\text{transition table}}}^{\text{sometimes combined } \phi(x_i, y_i, y_{i-1})}$$

Graphical Model Notation





Predictions Using CRFs

$$\vec{y} = \underset{y}{\operatorname{argmax}}\ p(y|x)$$

$$= \underset{y}{\operatorname{argmax}}\ e^{\theta \cdot \Phi(x,y)} = \underset{y}{\operatorname{argmax}}\ \theta \cdot \Phi(x,y)$$

$$= \underset{y}{\operatorname{argmax}}\ \theta \cdot \sum_i \phi(x_i, y_i, y_{i-1}) = \underset{y}{\operatorname{argmax}} \sum_i \theta \cdot \phi(x_i, y_i, y_{i-1})$$

Viterbi

$$\pi(i, y_i) = \max_{y_1 \cdots y_i} \sum_{j=1}^{i-1} \theta \cdot \phi(x_j, y_j, y_{j-1})$$

$$= \max_{y_i}\ \theta \cdot \phi(x_i, y_i, y_{i-1}) + \pi(i-1, y_{i-1})$$

Likelihood Training of CRFs

$$\mathcal{L}(\theta, D) = \log \prod_j P(\vec{y}_j, \vec{x}_j) = \sum_j \log \frac{e^{\theta \cdot \Phi(\vec{x}, \vec{y})}}{\sum_y e^{\theta \cdot \Phi(\vec{x}, \vec{y})}}$$

$$= \sum_j \theta \cdot \Phi(\vec{x}, \vec{y}) - \log \sum_y e^{\theta \cdot \Phi(\vec{x}, \vec{y})}$$

$$\frac{\partial \mathcal{L}(\theta, D)}{\partial \theta_k} = \sum_j \Phi_k(\vec{x}, \vec{y}) - \sum_y p(y \mid x_j)\, \Phi_k(\vec{x}_j, y)$$

Likelihood Training of CRFs

$$\sum_{y} p(\vec{y}|\vec{x})\,\Phi_k(\vec{x},\vec{y}) = \sum_{y} p(\vec{y}|\vec{x}) \sum_{i} \phi_k(x_i, y_i, y_{i-1})$$

$$= \sum_{i}\sum_{y} p(y|x)\,\phi_k(x_i, y_i, y_{i-1}) = \sum_{i}\sum_{\substack{y_i \\ y_{i-1}}}\sum_{y/y_i, y_{i-1}} p(y|x)\,\phi_k(x_i, y_i, y_{i-1})$$

$$= \sum_{i}\sum_{\substack{y_i \\ y_{i-1}}} \phi_k(x_i, y_i, y_{i-1})\,\underline{q_i(y_i, y_{i-1})} \qquad q_i(a,b) = \sum_{\substack{y \\ y_i = a \\ y_{i+1} = b}} p(y|\vec{x})$$

$$\mu_i(a,b) = \sum_{\substack{y \\ y_i = a \\ y_{i+1} = b}} \overset{e^{\theta\Phi(\vec{x},\vec{y})}}{\underline{\Psi}}(\vec{y}) \qquad q_i(a,b) = \frac{\mu_i(a,b)}{\sum_{a,b} \mu_i(a,b)}$$

Forward-Backward Algorithm

$$\alpha(i, y_i) = p(y_i \mid x_1 \cdots x_i) = \sum_{y_1 \cdots y_{i-1}} \psi(x_i, y_i, y_{i-1}) \prod_{j=1}^{i-1} \underbrace{\psi(x_j, y_j, y_{j-1})}_{e^{\theta \cdot \phi(x_j, y_j, y_{j-1})}}$$

$$= \sum_{y_{i-1}} \psi(x_i, y_i, y_{i-1}) \, \alpha(i-1, y_{i-1})$$

$$\beta(i, y_i) = p(y_i \mid x_{i+1} \cdots x_n) = \sum_{y_{i+1} \cdots y_n} \prod_{j=i+1}^{n} \psi(x_j, y_j, y_{j-1})$$

$$Z = \sum_y \alpha(n, y)$$

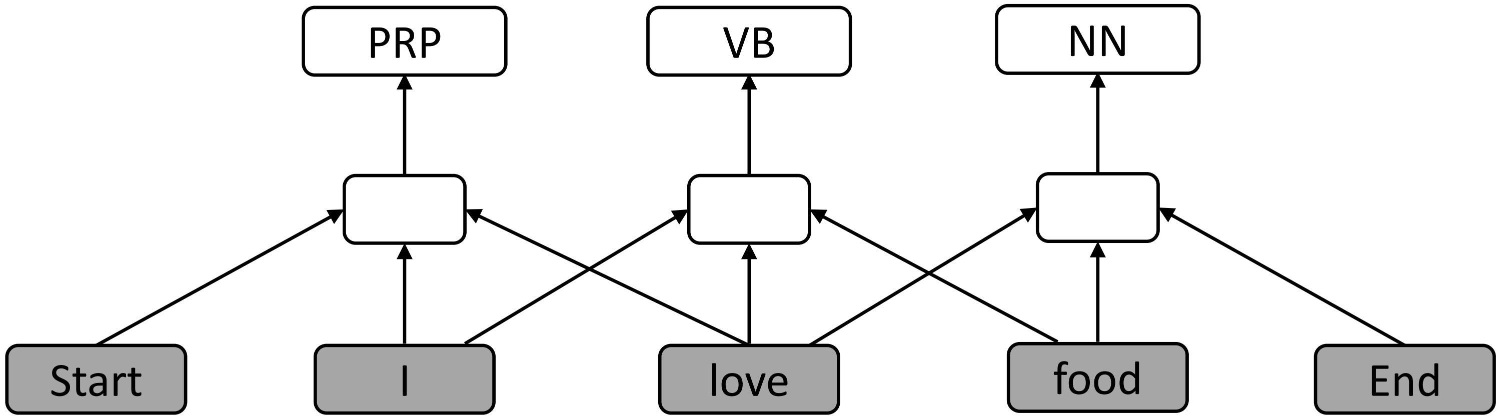
$$= \sum_{y_{i+1}} \psi(x_{i+1}, y_{i+1}, y_i) \, \beta(i+1, y_{i+1})$$

$$\mu_j(a, b) = \alpha(j, a) \, \psi(x_j, b, a) \, \beta(j+1, b)$$

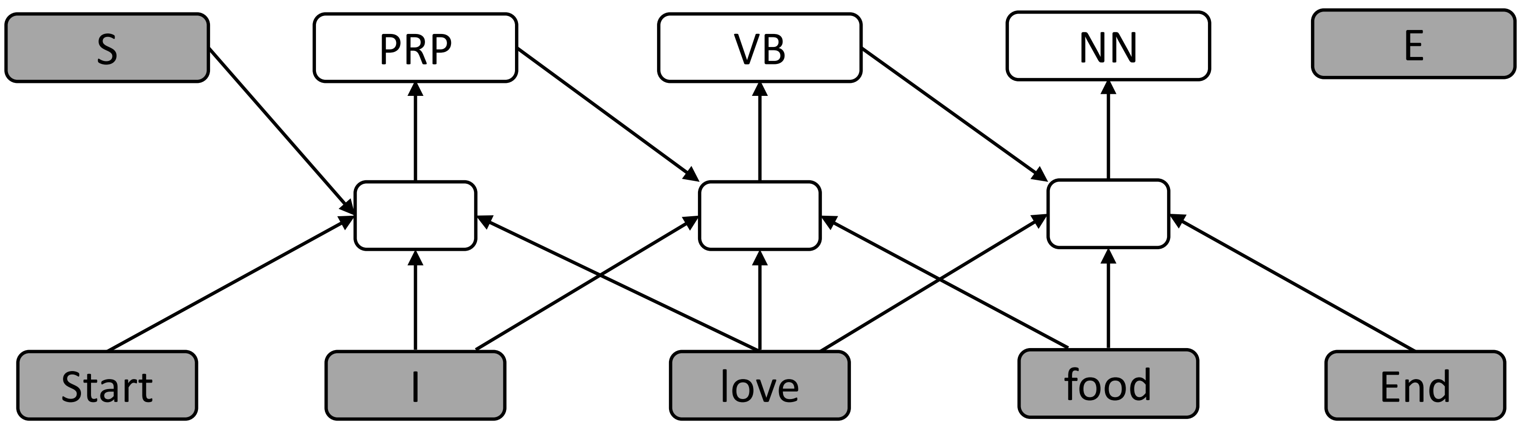
Outline

- Marginal Inference in HMMs: F/B algorithm
- Maximum Entropy Markov Models
- Conditional Random Fields
- **Neural Sequence Tagging**

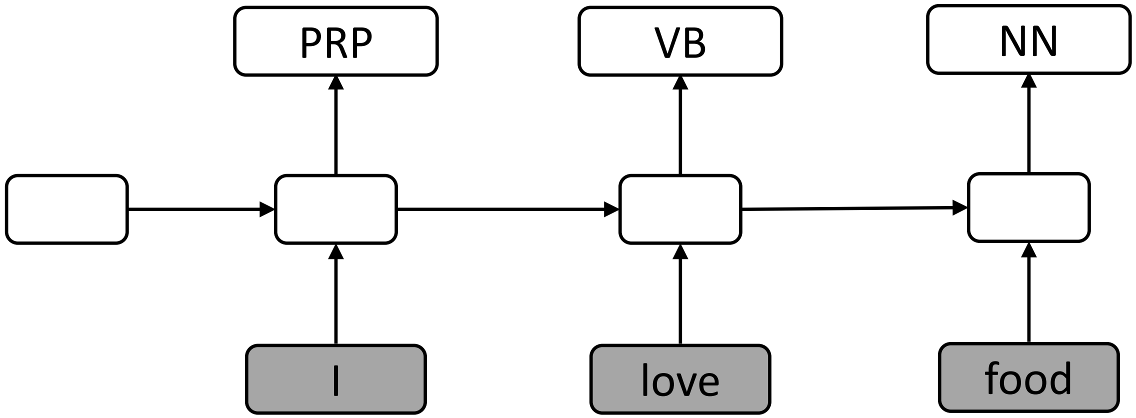
Simple Neural Tagger



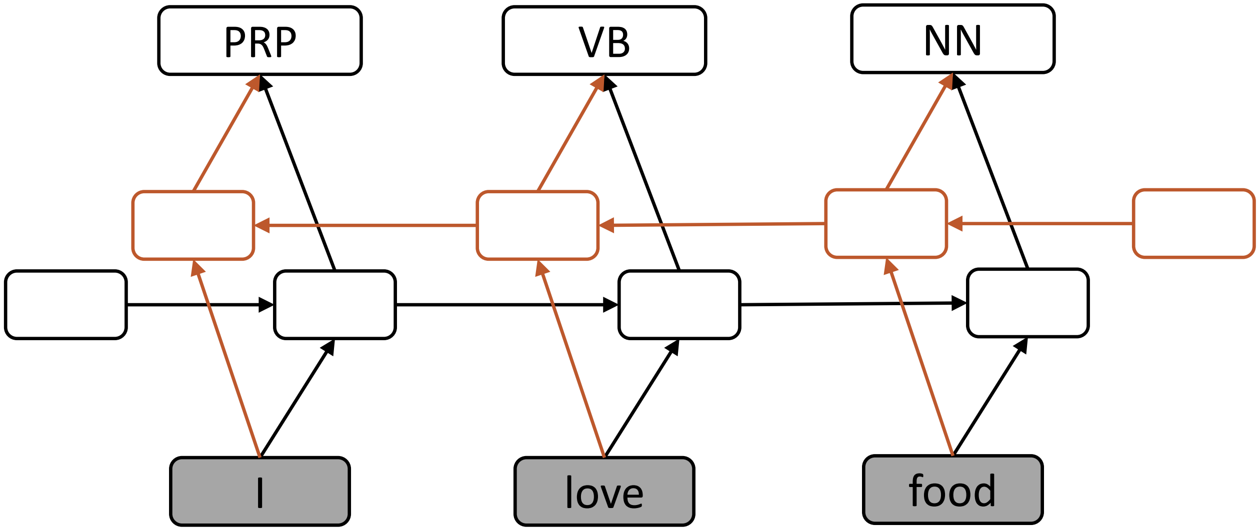
MEMM-*ish* Neural Tagger



Recurrent Neural Tagger



Bidirectional RNN Tagger



Upcoming...

Homework

- Homework 2 is due (~10 days): February 13, 2017
- Write-up, data, and code for Homework 2 is up
- Ask questions early!

Project

- Proposal is due on Tuesday: February 7, 2017
- Only 2 pages

Summaries

- Paper summaries: February 17, February 28, March 14
- Only 1 page each