



Language
Technologies
Institute

Carnegie
Mellon
University

Multimodal Machine Learning

Lecture 9.1: Discriminative Graphical Models

Louis-Philippe Morency

* Original version co-developed with Tadas Baltrusaitis

Lecture Objectives

- Markov Random Fields
 - Boltzmann/Gibbs distribution
 - Factor graphs
- Conditional Random Fields
 - Multi-View Conditional Random Fields
- CRFs and Deep Learning
 - DeepConditional Neural Fields
 - CRF and Bilinear LSTM
- Continuous and Fully-Connected CRFs



Administrative Stuff



Upcoming Schedule

- First project assignment:
 - Proposal presentation (10/1 and 10/3)
 - First project report (Sunday 10/6)
- Midterm project assignment
 - Midterm presentations (11/5 and 11/7)
 - Midterm report (Sunday 11/10)
- Final project assignment
 - Final presentation (12/3 & 12/5)
 - Final report (Sunday 12/8)



Midterm Project Report Instructions

- **Goal:** Evaluate state-of-the-art models on your dataset and identify key issues through a detailed error analysis
 - It will inform the design of your new research ideas
- **Report format:** 8 pages, 2 column (ICML template)
 - The report should follow a similar structure to a research paper
- **Number of SOTA models**
 - Teams of 3 should have at least two baseline models
 - Teams of 4 or 5 should have at least three baseline models
- **Error analysis**
 - This is one of the most important part of this report. You need to understand where previous models can be improved.

Midterm Poster Presentations

Title and authors		
Motivation	Main baseline model	New idea 1
Dataset	Main results	New idea 2
Experimental methodology	Error analysis	New idea 3



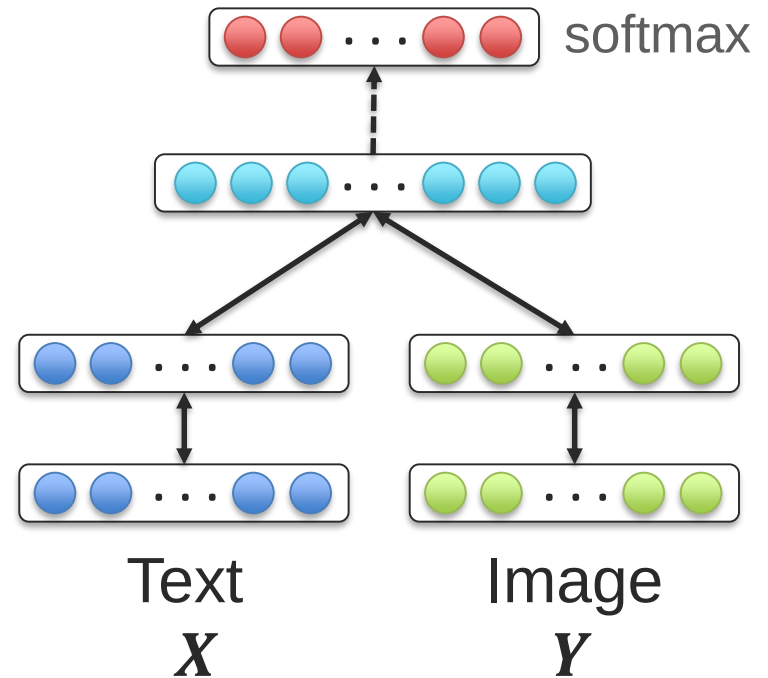
Quick Recap



Multimodal Representation Learning

Learn (unsupervised) a joint representation between multiple modalities where similar unimodal concepts are closely projected.

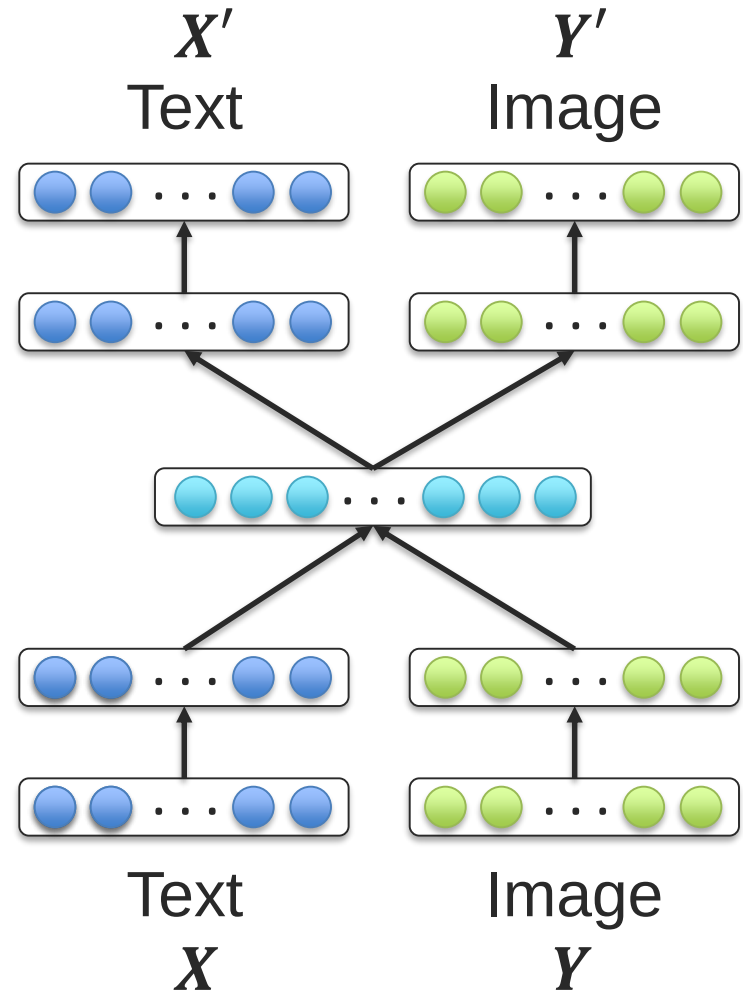
- Deep Multimodal Boltzmann machines



Multimodal Representation Learning

Learn (unsupervised) a joint representation between multiple modalities where similar unimodal concepts are closely projected.

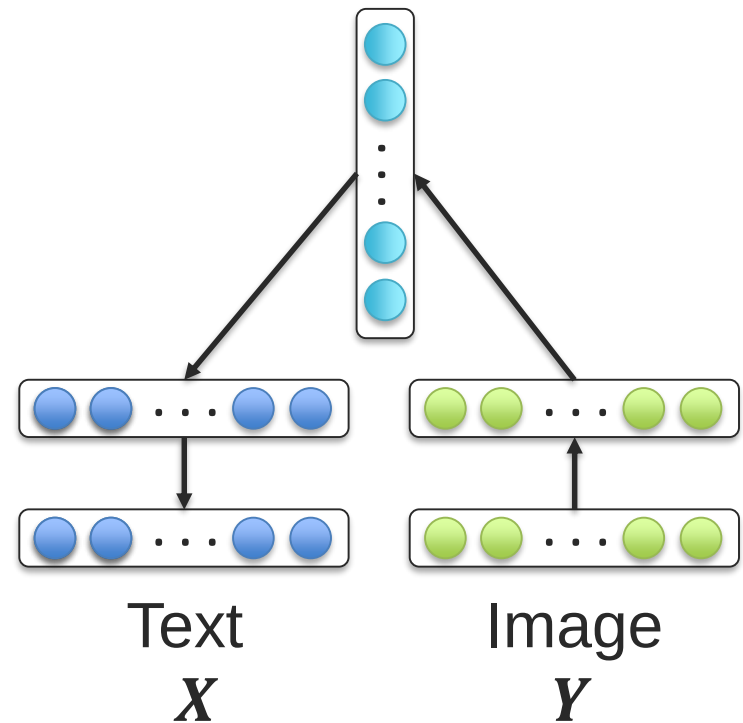
- ❑ Deep Multimodal Boltzmann machines
- ❑ Stacked Autoencoder



Multimodal Representation Learning

Learn (unsupervised) a joint representation between multiple modalities where similar unimodal concepts are closely projected.

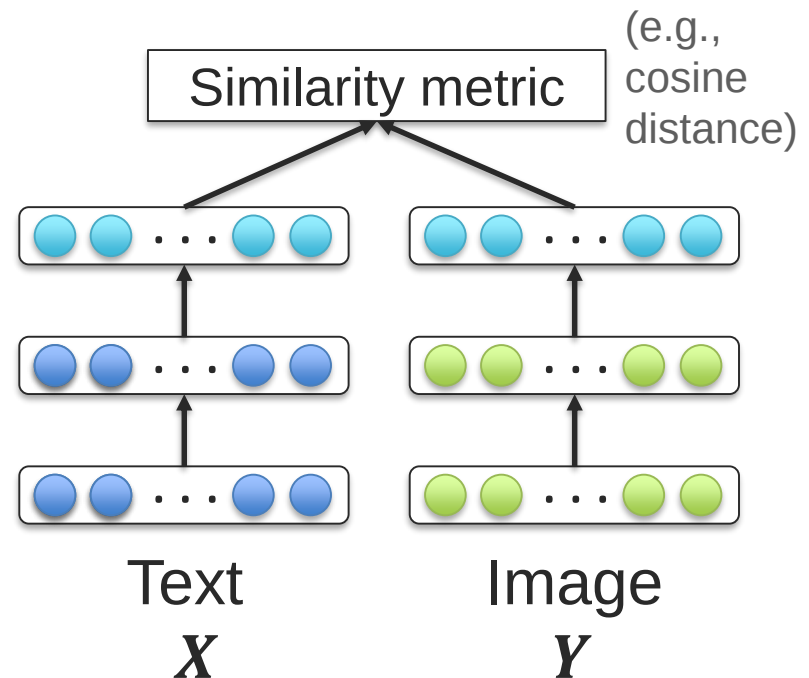
- ❑ Deep Multimodal Boltzmann machines
- ❑ Stacked Autoencoder
- ❑ Encoder-Decoder



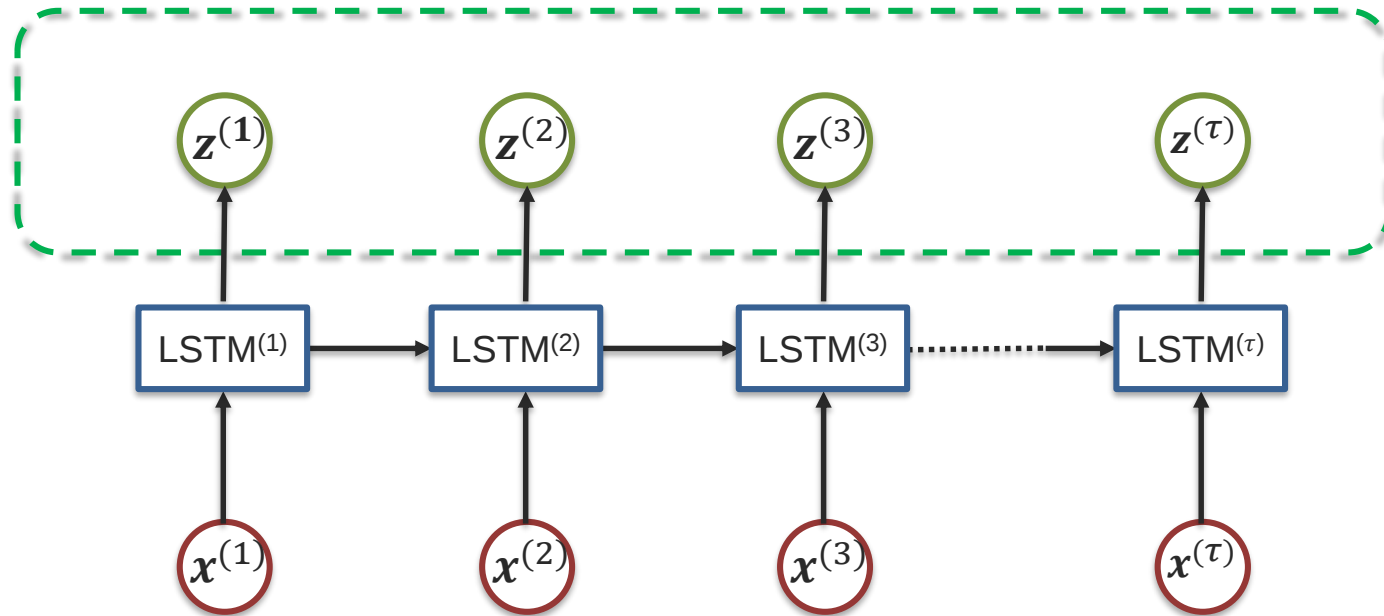
Multimodal Representation Learning

Learn (unsupervised) a joint representation between multiple modalities where similar unimodal concepts are closely projected.

- ❑ Deep Multimodal Boltzmann machines
- ❑ Stacked Autoencoder
- ❑ Encoder-Decoder
- ❑ “Minimum-distance” Multimodal Embedding



Recurrent Neural Network using LSTM Units



How can we improve reasoning by including prior domain knowledge?

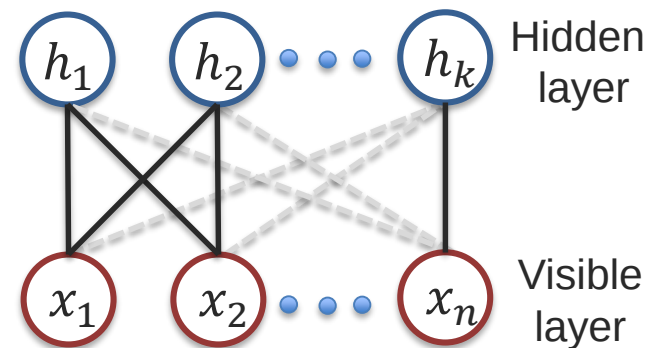


Markov Random Fields



Recap: Restricted Boltzmann Machine (RBM)

- Undirected Graphical Model
- A generative rather than discriminative model
- Connections from every hidden unit to every visible one
- No connections across units (hence Restricted), makes it easier to train and do inference



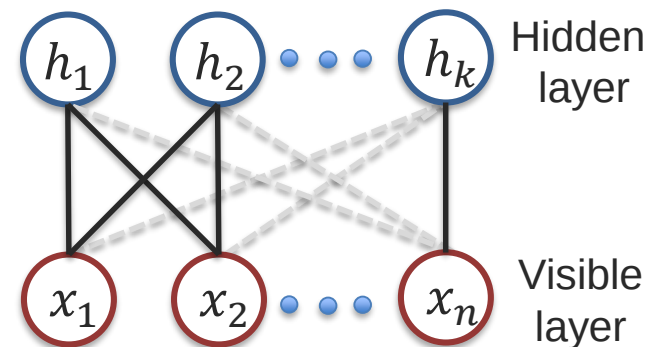
Recap: Restricted Boltzmann Machine (RBM)

$$p(\mathbf{x}, \mathbf{h}; \theta) = \frac{\exp(-E(\mathbf{x}, \mathbf{h}; \theta))}{\sum_{\mathbf{x}'} \sum_{\mathbf{h}'} \exp(-E(\mathbf{x}', \mathbf{h}'; \theta))} \quad \leftarrow \text{Partition function } Z$$

- Hidden and visible layers are binary (e.g. $\mathbf{x} = \{0, \dots, 1, 0, 1\}$)
- Model parameters $\theta = \{W, \mathbf{b}, \mathbf{a}\}$

$$E = -\mathbf{x}W\mathbf{h} - \mathbf{b}\mathbf{x} - \mathbf{a}\mathbf{h}$$

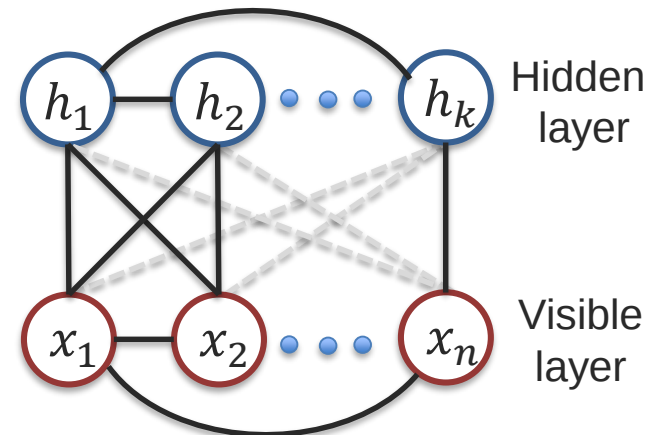
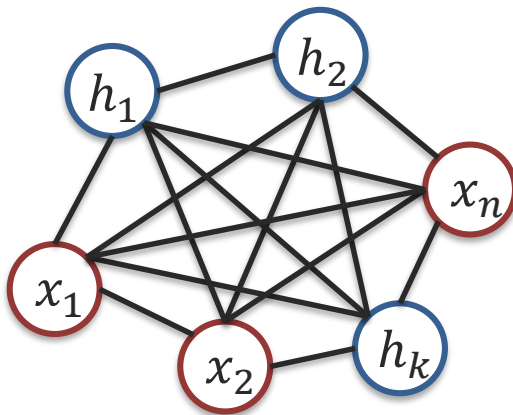
$$E = - \underbrace{\sum_i \sum_j w_{i,j} x_i h_j}_{\text{Interaction term}} - \underbrace{\sum_i b_i x_i}_{\text{Bias terms}} - \underbrace{\sum_j a_j h_j}_{\text{Bias terms}}$$



Recap: Boltzmann Machine

$$p(\mathbf{x}, \mathbf{h}; \theta) = \frac{\exp(-E(\mathbf{x}, \mathbf{h}; \theta))}{\sum_{\mathbf{x}'} \sum_{\mathbf{h}'} \exp(-E(\mathbf{x}', \mathbf{h}'; \theta))}$$

- Hidden and visible layers are binary (e.g. $\mathbf{x} = \{0, \dots, 1, 0, 1\}$)

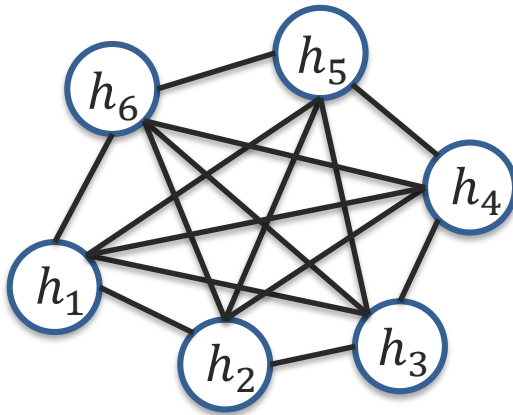


Recap: Boltzmann Distribution

[also called Gibbs measure]

$$p(\mathbf{h}; \theta) = \frac{\exp(-E(\mathbf{h}; \theta)/kT)}{\sum_{\mathbf{h}'} \exp(-E(\mathbf{h}'; \theta)/kT)}$$

- probability distribution that gives the probability that a system will be in a certain state $\mathbf{h}$



$E(\mathbf{h}; \theta)$: Energy of state $\mathbf{h}$

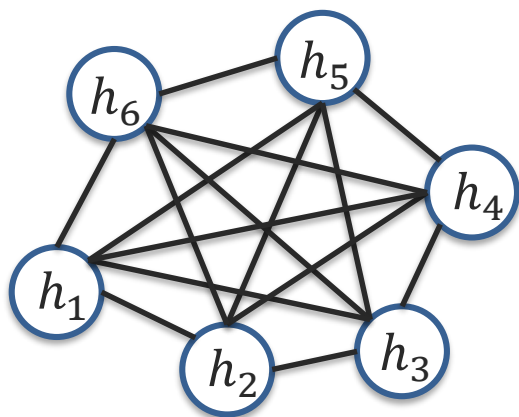
k : Boltzmann constant

T : Thermodynamic temperature

Markov Random Fields

$$p(H = \mathbf{h}; \theta) = \frac{\exp(-E(\mathbf{h}; \theta))}{\sum_{\mathbf{h}'} \exp(-E(\mathbf{h}'; \theta))} = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)}$$

- Set of random variables H having a Markov property described by undirected graph



$$\Phi(\mathbf{h}; \theta) = \prod_k \phi_k(\mathbf{h}; \theta_k)$$

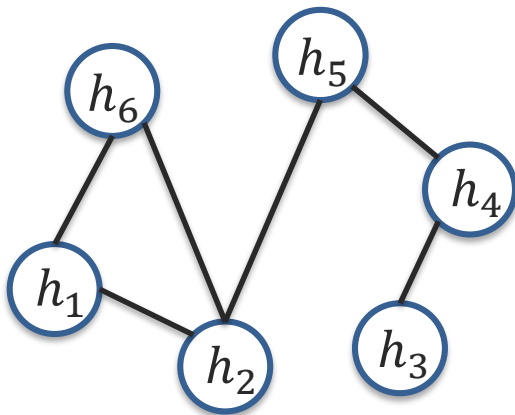
Potential functions
 $\phi_k(\mathbf{h}; \theta) > 0$

$$= \exp \left(- \sum_k E_k(\mathbf{h}; \theta_k) \right)$$

Markov Random Fields

$$p(H = \mathbf{h}; \theta) = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

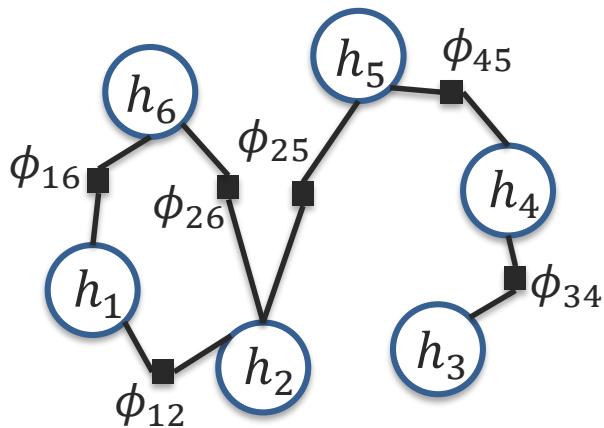
$$\begin{aligned} \Phi(\mathbf{h}; \theta) = & \phi_{12}(h_1, h_2; \theta_{12}) \times \\ & \phi_{16}(h_1, h_6; \theta_{16}) \times \\ & \phi_{26}(h_2, h_6; \theta_{26}) \times \\ & \phi_{25}(h_2, h_5; \theta_{25}) \times \\ & \phi_{45}(h_4, h_5; \theta_{45}) \times \\ & \phi_{34}(h_3, h_4; \theta_{34}) \end{aligned}$$



Markov Random Fields: Factor Graphs

$$p(H = \mathbf{h}; \theta) = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

$$\begin{aligned} \Phi(\mathbf{h}; \theta) = & \phi_{12}(h_1, h_2; \theta_{12}) \times \\ & \phi_{16}(h_1, h_6; \theta_{16}) \times \\ & \phi_{26}(h_2, h_6; \theta_{26}) \times \\ & \phi_{25}(h_2, h_5; \theta_{25}) \times \\ & \phi_{45}(h_4, h_5; \theta_{45}) \times \\ & \phi_{34}(h_3, h_4; \theta_{34}) \end{aligned}$$



Markov Random Fields (Factor Graphs)

$$p(H = \mathbf{h}; \theta) = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

$$\Phi(\mathbf{h}; \theta) = \phi_{12}(h_1, h_2; \theta_{12}) \times$$

$$\phi_{16}(h_1, h_6; \theta_{16}) \times$$

$$\phi_{26}(h_2, h_6; \theta_{26}) \times$$

$$\phi_{25}(h_2, h_5; \theta_{25}) \times$$

$$\phi_{45}(h_4, h_5; \theta_{45}) \times$$

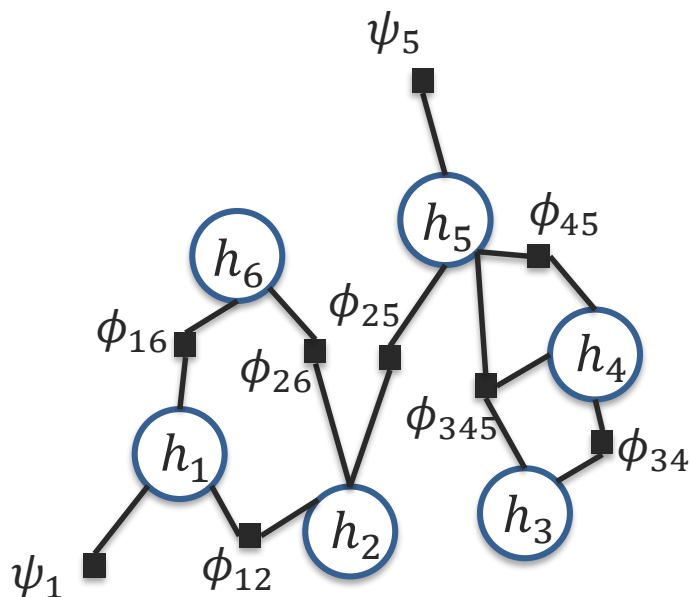
$$\phi_{34}(h_3, h_4; \theta_{34}) \times$$

$$\psi_1(h_1; \theta_1) \times \psi_5(h_5; \theta_5)$$

$$\times \phi_{345}(h_3, h_4, h_5; \theta_{345})$$

pairwise
potentials

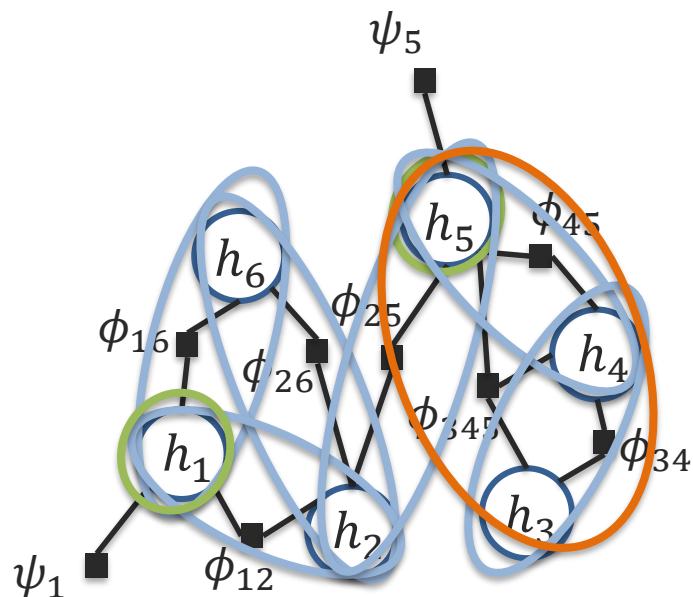
Unary
potentials



Markov Random Fields – Clique Factorization

$$p(H = \mathbf{h}; \theta) = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

Clique factorization



$$\Phi(\mathbf{h}; \theta) = \phi_{12}(h_1, h_2; \theta_{12}) \times$$

$$\phi_{16}(h_1, h_6; \theta_{16}) \times$$

$$\phi_{26}(h_2, h_6; \theta_{26}) \times$$

$$\phi_{25}(h_2, h_5; \theta_{25}) \times$$

$$\phi_{45}(h_4, h_5; \theta_{45}) \times$$

$$\phi_{34}(h_3, h_4; \theta_{34}) \times$$

$$\psi_1(h_1; \theta_1) \times \psi_5(h_5; \theta_5)$$

$$\times \phi_{345}(h_3, h_4, h_5; \theta_{345})$$

pairwise
potentials

Unary
potentials

Chain Markov Random Fields (Factor Graphs)

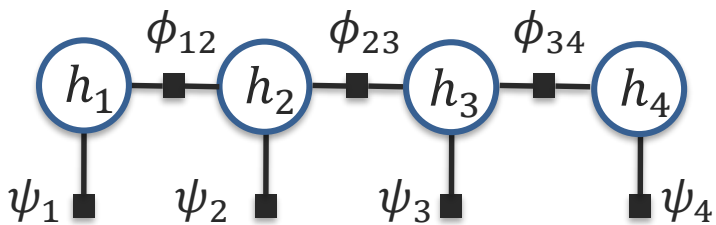
$$p(H = \mathbf{h}; \theta) = \frac{\Phi(\mathbf{h}; \theta)}{\sum_{\mathbf{h}'} \Phi(\mathbf{h}'; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

$$\Phi(\mathbf{h}; \theta) = \phi_{12}(h_1, h_2; \theta_{12}) \times \phi_{23}(h_2, h_3; \theta_{23}) \times \phi_{34}(h_3, h_4; \theta_{34}) \times$$

pairwise potentials

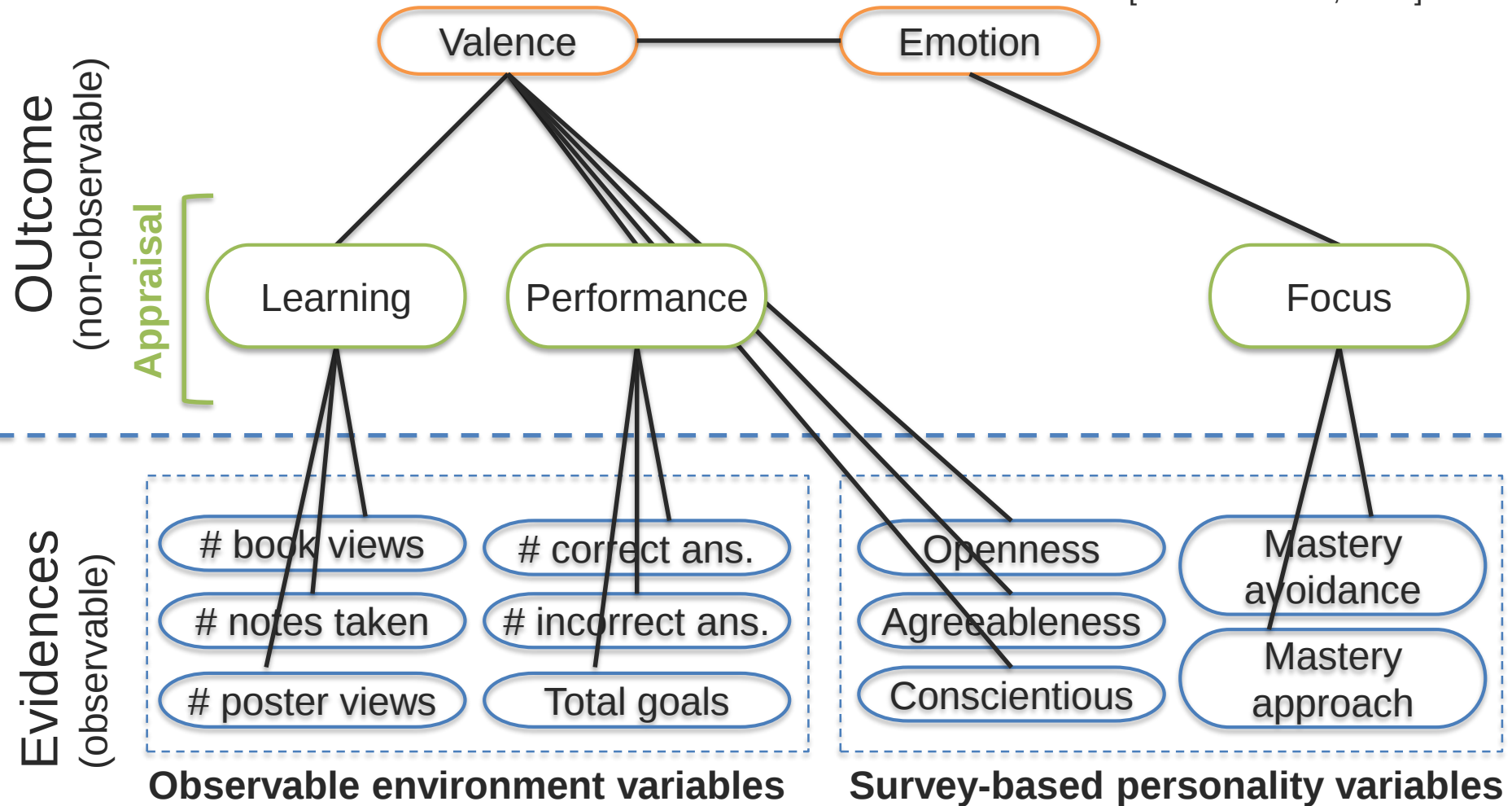
$$\psi_1(h_1; \theta_1) \times \psi_2(h_2; \theta_2) \times \psi_3(h_3; \theta_3) \times \psi_4(h_4; \theta_4)$$

Unary potentials



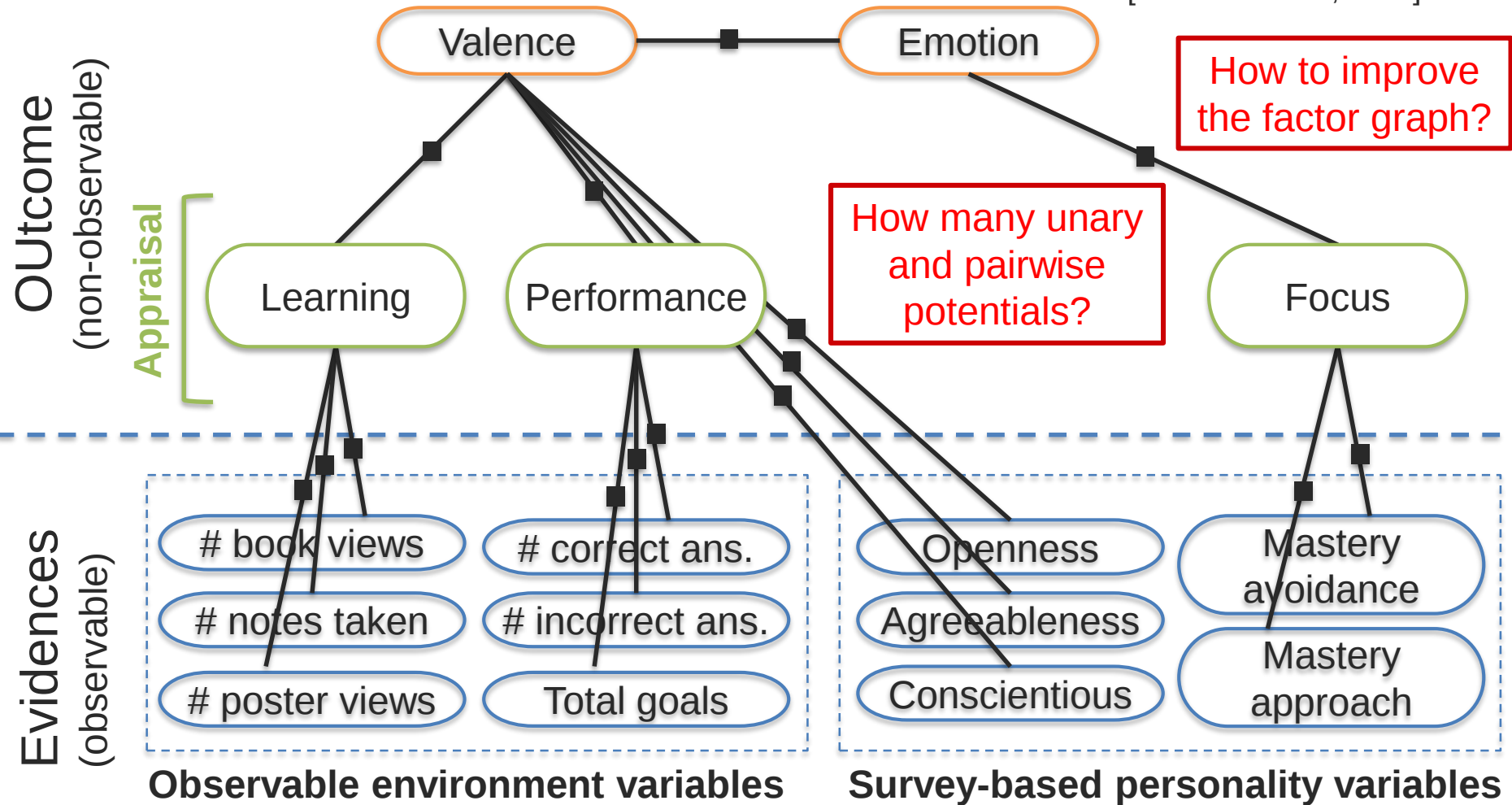
Example: Markov Random Field – Graphical Model

[Sabourin et al., 2011]



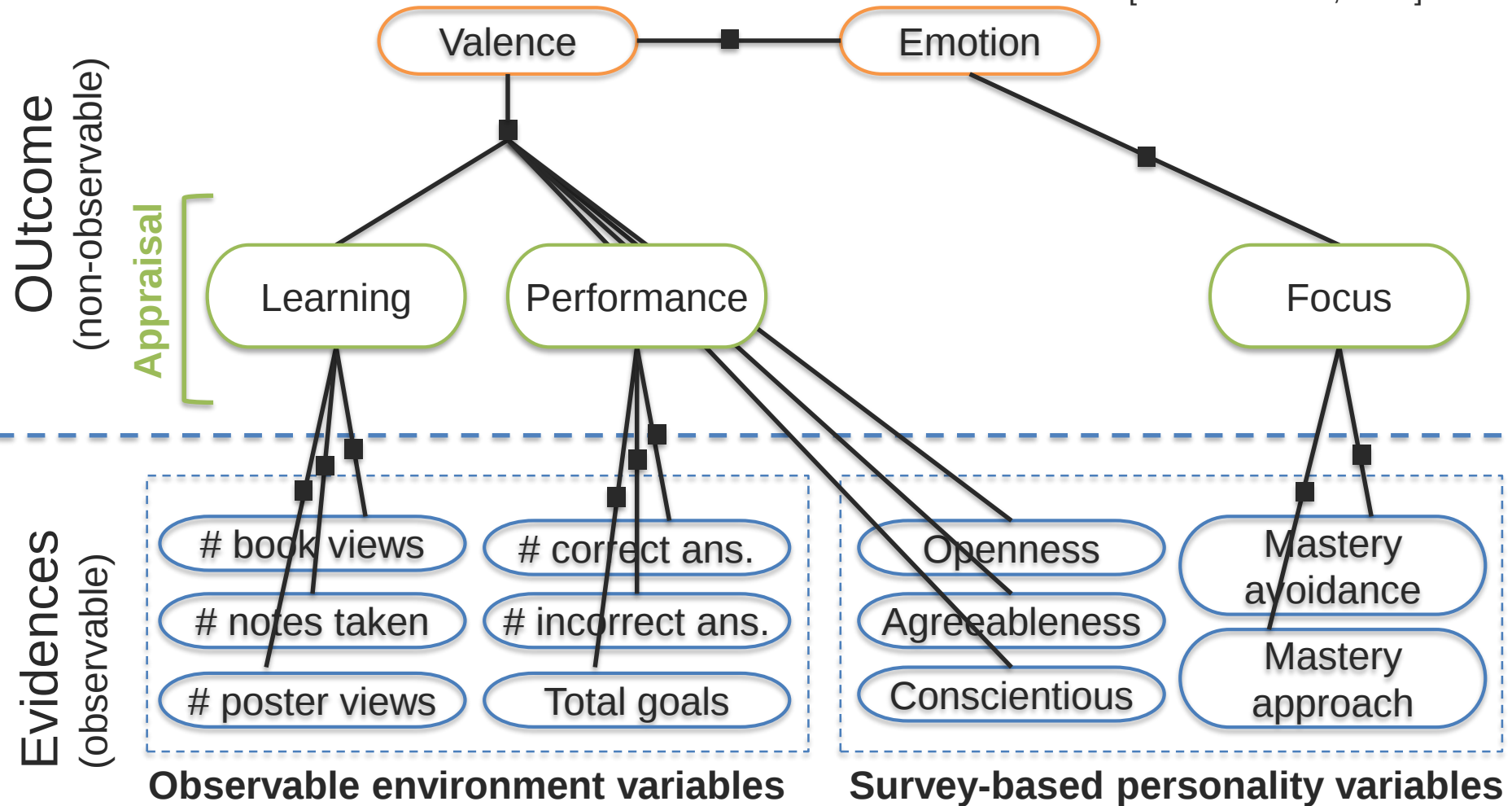
Example: Markov Random Field – Factor Graph

[Sabourin et al., 2011]



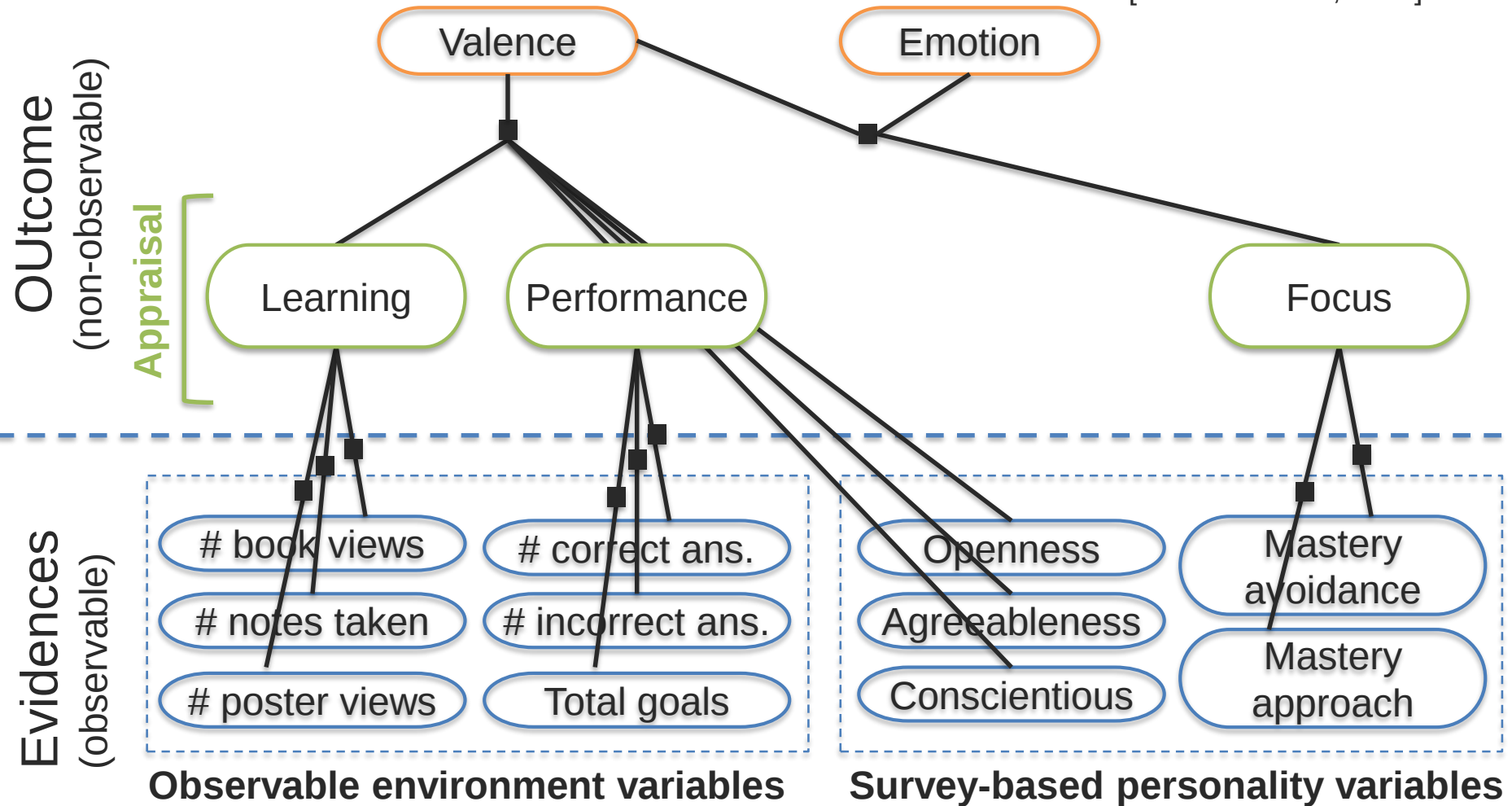
Example: Markov Random Field – Factor Graph

[Sabourin et al., 2011]



Example: Markov Random Field – Factor Graph

[Sabourin et al., 2011]



Conditional Random Fields



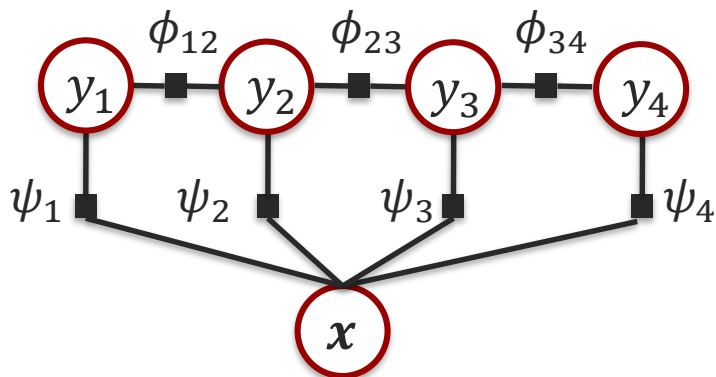
Conditional Random Fields (Factor Graphs)

$$p(\mathbf{y}|\mathbf{x}; \theta) = \frac{\Phi(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \Phi(\mathbf{y}', \mathbf{x}; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

$$\begin{aligned} \Phi(\mathbf{y}, \mathbf{x}; \theta) = & \phi_{12}(y_1, y_2, \mathbf{x}; \theta_{12}) \times \\ & \phi_{23}(y_2, y_3, \mathbf{x}; \theta_{23}) \times \\ & \phi_{34}(y_3, y_4, \mathbf{x}; \theta_{34}) \times \\ & \psi_1(y_1, \mathbf{x}; \theta_1) \times \\ & \psi_2(y_2, \mathbf{x}; \theta_2) \times \\ & \psi_3(y_3, \mathbf{x}; \theta_3) \times \\ & \psi_4(y_4, \mathbf{x}; \theta_4) \end{aligned}$$

pairwise
potentials

Unary
potentials



Conditional Random Fields (Factor Graphs)

$$p(\mathbf{y}|\mathbf{x}; \theta) = \frac{\Phi(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \Phi(\mathbf{y}', \mathbf{x}; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)}$$

$$\Phi(\mathbf{y}, \mathbf{x}; \theta) = \phi_{12}(y_1, y_2, \mathbf{x}; \theta_{12}) \times$$

$$\phi_{23}(y_2, y_3, \mathbf{x}; \theta_{23}) \times$$

$$\phi_{34}(y_3, y_4, \mathbf{x}; \theta_{34}) \times$$

$$\psi_1(y_1, x_1; \theta_1) \times$$

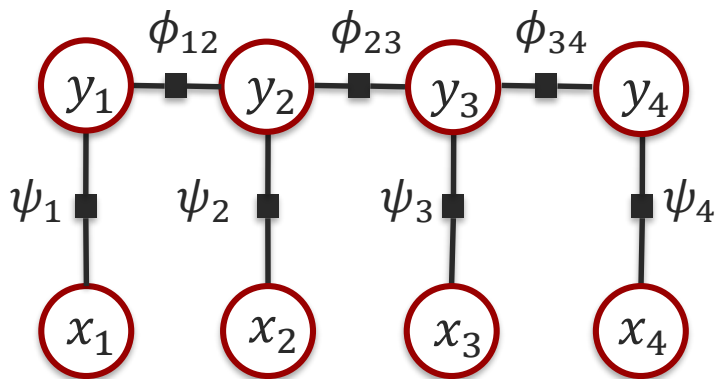
$$\psi_2(y_2, x_2; \theta_2) \times$$

$$\psi_3(y_3, x_3; \theta_3) \times$$

$$\psi_4(y_4, x_4; \theta_4)$$

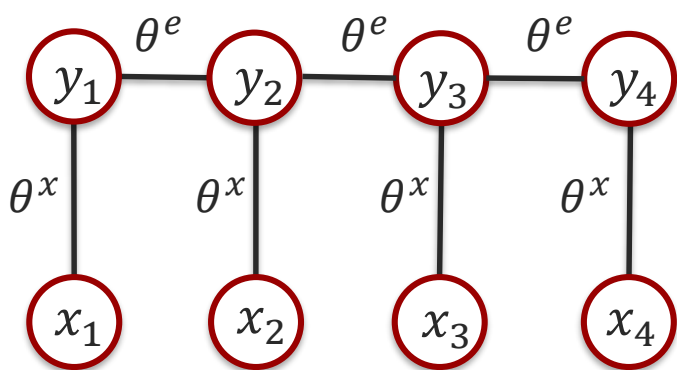
pairwise
potentials

Unary
potentials



Conditional Random Fields (Log-linear Model)

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}; \theta) &= \frac{\Phi(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \Phi(\mathbf{y}', \mathbf{x}; \theta)} = \frac{\sum_k \phi_k(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \sum_k \phi_k(\mathbf{y}', \mathbf{x}; \theta)} \\ &= \frac{\exp(\sum_k \theta_k f_k(\mathbf{y}, \mathbf{x}))}{\sum_{\mathbf{y}'} \exp(\sum_k \theta_k f_k(\mathbf{y}', \mathbf{x}))} \end{aligned}$$



$f_k(\mathbf{y}, \mathbf{x})$: feature function

- Pairwise feature function

$$f_k(y_i, y_j, \mathbf{x}; \theta^e)$$

- Unary feature function

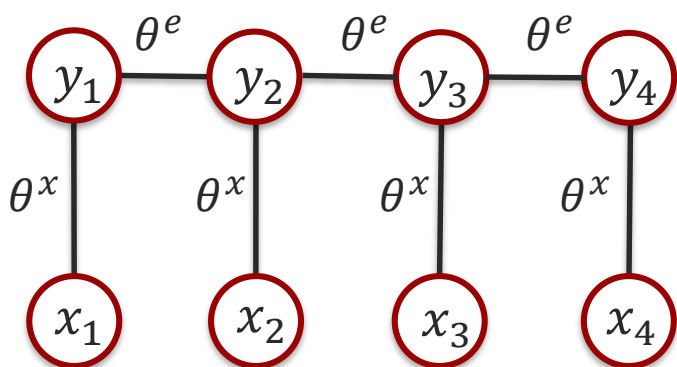
$$f_k(y_i, \mathbf{x}; \theta^x)$$



Learning Parameters of a CRF Model

$$\operatorname{argmax}_{\hat{y}} \log(p(\mathbf{y}|\mathbf{x}; \theta))$$

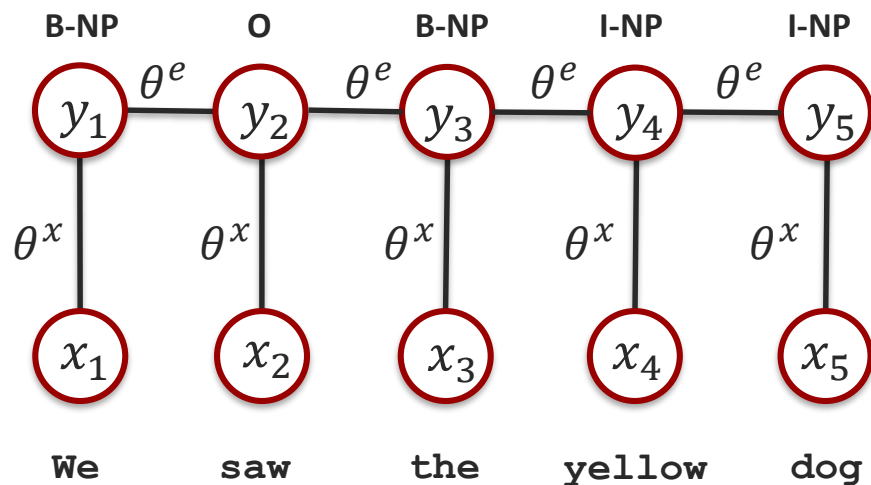
- Gradient can be computed analytically
 - Inference of marginal probabilities using belief propagation (or loopy belief propagation for cyclic graphs)
- Optimized with stochastic or batch approaches



CRFs for Shallow Parsing

$$p(\mathbf{y}|\mathbf{x}; \theta) = \frac{\Phi(\mathbf{y}, \mathbf{x}; \theta)}{\sum_{\mathbf{y}'} \Phi(\mathbf{y}', \mathbf{x}; \theta)} = \frac{\exp(\sum_k \theta_k f_k(\mathbf{y}, \mathbf{x}))}{\sum_{\mathbf{y}'} \exp(\sum_k \theta_k f_k(\mathbf{y}', \mathbf{x}))}$$

- How many θ^x parameters?
- What did θ^x learn?
- What did θ^e learn?



	B-NP	I-NP	O
B-NP	θ_{11}	θ_{21}	θ_{31}
I-NP	θ_{12}	θ_{22}	θ_{32}
O	θ_{13}	θ_{23}	θ_{33}

Labels:

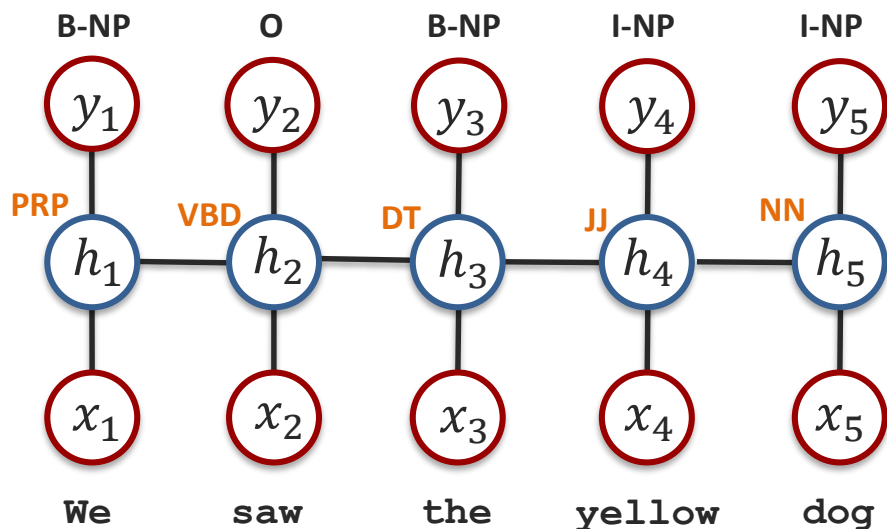
B-NP: Beginning of a noun phrase
 I-NP: Continuation of a noun phrase
 O: Outside a noun phrase

Dictionary size: 10,000 words

Latent-Dynamic CRF

$$p(y|x; \theta) = \sum_h p(y|h; \theta) p(h|x; \theta) \quad \text{where } p(y|h; \theta) = \begin{cases} 1 & \text{if } \forall h_t \in \mathcal{H}_{y_t} \\ 0 & \text{otherwise} \end{cases}$$

$$= \sum_{h: \forall h_t \in \mathcal{H}_{y_t}} p(h|x; \theta) = \sum_{h: \forall h_t \in \mathcal{H}_{y_t}} \frac{\Phi(h, x; \theta)}{\sum_{h'} \Phi(h', x; \theta)}$$



Latent variables (e.g., POS tags)

$h = \{h_1, h_2, h_3, \dots, h_t\}$ where $h_t \in \{\mathcal{H}_{y_t}\}$

For example:

$\mathcal{H} = \{\mathcal{H}_{\text{B-NP}}, \mathcal{H}_{\text{I-NP}}, \mathcal{H}_{\text{O}}\}$

$\mathcal{H} = \{B_1, B_2, B_3, B_4, I_1, I_2, I_3, I_4, O_1, O_2, O_3, O_4\}$

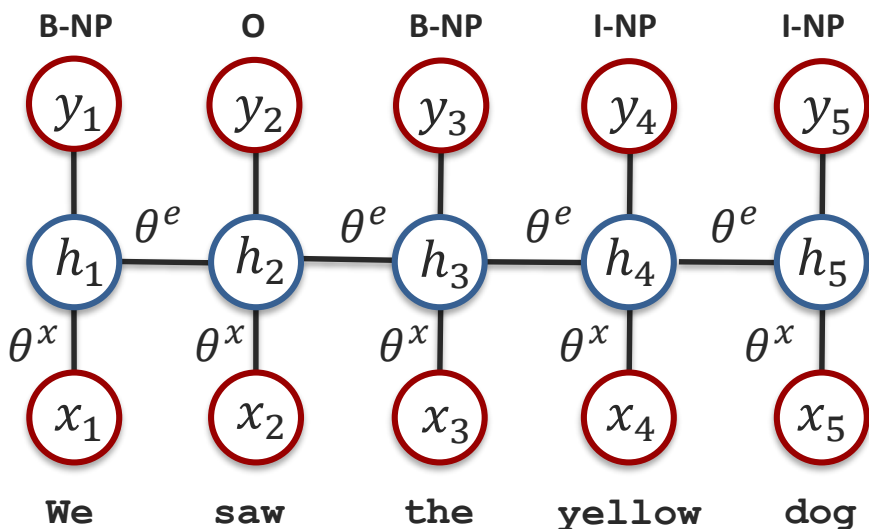
Dictionary size: 10,000 words



Latent-Dynamic CRF

$$p(\mathbf{y}|\mathbf{x}; \theta) = \sum_{\mathbf{h}: \forall h_t \in \mathcal{H}_{y_t}} \frac{\exp(\sum_k \theta_k f_k(\mathbf{h}, \mathbf{x}))}{\sum_{\mathbf{h}'} \exp(\sum_k \theta_k f_k(\mathbf{h}', \mathbf{x}))}$$

- How many θ^x parameters?
- How many θ^e parameters?
- What did θ^x learn?
- What did θ^e learn?



- Intrinsic dynamics
- Extrinsic dynamics

Latent variables (e.g., POS tags)

$\mathbf{h} = \{h_1, h_2, h_3, \dots, h_t\}$ where $h_t \in \{\mathcal{H}_{y_t}\}$

For example:

$\mathcal{H} = \{\mathcal{H}_{\text{B-NP}}, \mathcal{H}_{\text{I-NP}}, \mathcal{H}_{\text{O}}\}$

$\mathcal{H} = \{B_1, B_2, B_3, B_4, I_1, I_2, I_3, I_4, O_1, O_2, O_3, O_4\}$

Dictionary size: 10,000 words



Latent-Dynamic CRF for Shallow Parsing

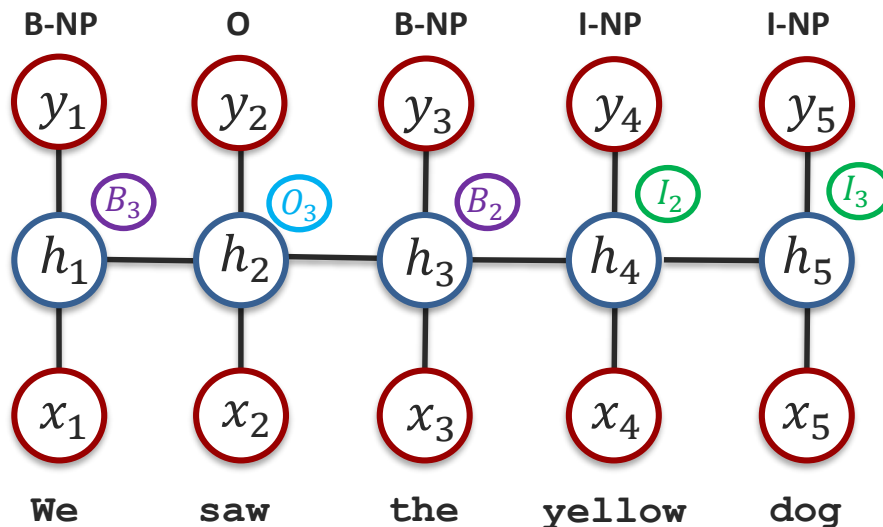
Experiment – Analyzing latent variables

- **Task:** Shallow parsing with CoNLL 2000 dataset
- **Input features:** word feature only
- **Output labels:** Noun phrase labels

- 1) Select hidden state a^* with highest marginal:

$$a^* = \arg \max_a p(h_t = a | x; \theta)$$

- 2) Compute relative frequency for each word



Label	State	Words	POS	Freq.
B	B_1	That	WDT	0.85
		who	WP	0.49
		Who	WP	0.33
		any	DT	1.00
	B_2	an	DT	1.00
		a	DT	0.98
	B_3	They	PRP	1.00
		we	PRP	1.00
		he	PRP	1.00
		Nasdaq	NNP	1.00
	B_4	Florida	NNP	0.99
		cities	NNS	0.99

Label	State	Words	POS	Freq.
O	O_1	but	CC	0.88
		by	IN	0.73
		or	IN	0.67
		4.6	CD	1.00
	O_2	1	CD	1.00
		1 1	CD	0.62
	O_3	were	VBD	0.94
		rose	VBD	0.93
		have	VBP	0.92
		been	VBN	0.97
	O_4	be	VB	0.94
		to	TO	0.92

Latent variables (e.g., POS tags)

$$h = \{h_1, h_2, h_3, \dots, h_t\} \quad \text{where } h_t \in \{\mathcal{H}_{y_t}\}$$

For example:

$$\mathcal{H} = \{\mathcal{H}_{B-NP} \mathcal{H}_{I-NP} \mathcal{H}_O\}$$

$$\mathcal{H} = \{B_1, B_2, B_3, B_4, I_1, I_2, I_3, I_4, O_1, O_2, O_3, O_4\}$$

Dictionary size: 10,000 words



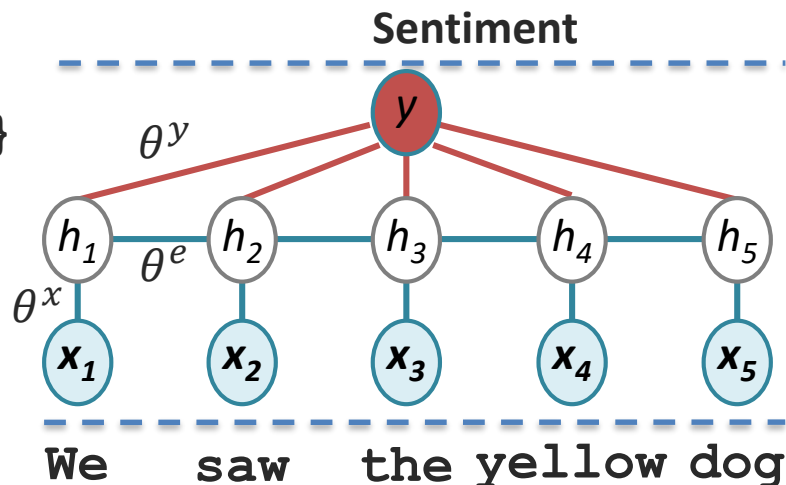
Hidden Conditional Random Field

Sequence label:

$y \in \mathcal{Y}$ for example, $\mathcal{Y}: \{\text{positive, negative}\}$

Latent variables with shared hidden states:

$\mathbf{h} = \{h_1, h_2, h_3, \dots, h_t\}$ where $h_t \in \mathcal{H}$



$$p(\mathbf{y}, \mathbf{h} \mid \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{Z(\mathbf{x}; \boldsymbol{\theta})} \exp \left\{ \sum_t \boldsymbol{\theta}^x \cdot f^x(h_t, \mathbf{x}_t) + \sum_t \boldsymbol{\theta}^e \cdot f^e(h_t, h_{t-1}, \mathbf{y}) + \sum_t \boldsymbol{\theta}^y \cdot f^y(\mathbf{y}, h_t) \right\}$$

$$p(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) = \sum_{\mathbf{h}} p(\mathbf{y}, \mathbf{h} \mid \mathbf{x}; \boldsymbol{\theta})$$

- Inference is tractable: $O(YH^2T)$
 - Linear in sequence length T !
- Parameter learning $(\boldsymbol{\theta}^x, \boldsymbol{\theta}^e, \boldsymbol{\theta}^y)$:
 - Gradient descent or L-BFGS

Shared hidden states



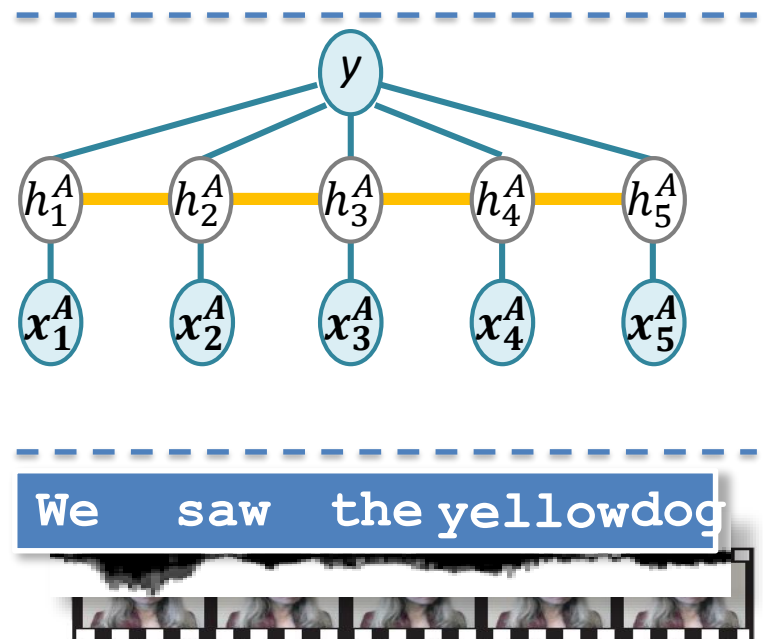
Learning Multimodal Structure

Modality-*private* structure

- Internal grouping of observations

Modality-*shared* structure

- Interaction and synchrony



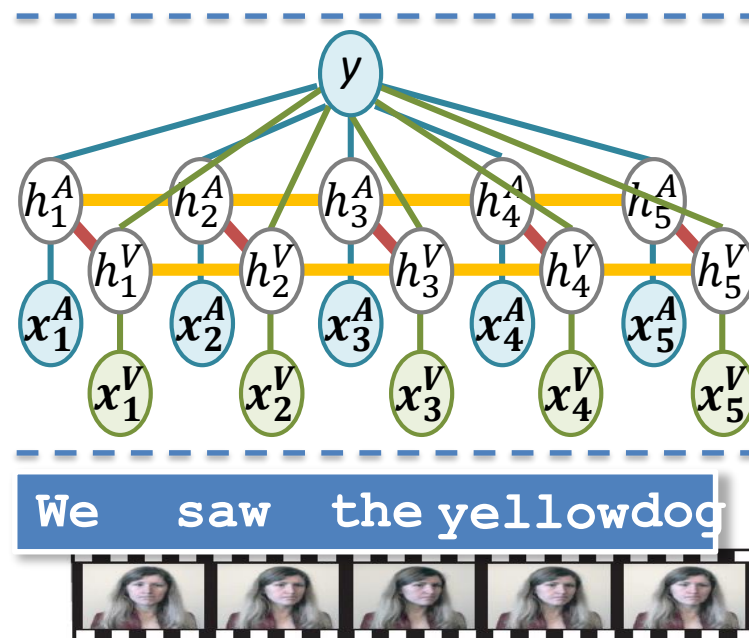
Multi-view Latent Variable Discriminative Models

Modality-*private* structure

- Internal grouping of observations

Modality-*shared* structure

- Interaction and synchrony



$$p(y | \mathbf{x}^A, \mathbf{x}^V; \boldsymbol{\theta}) = \sum_{\mathbf{h}^A, \mathbf{h}^V} p(y, \mathbf{h}^A, \mathbf{h}^V | \mathbf{x}^A, \mathbf{x}^V; \boldsymbol{\theta})$$

- Approximate inference using loopy-belief

CRFs and Deep Learning

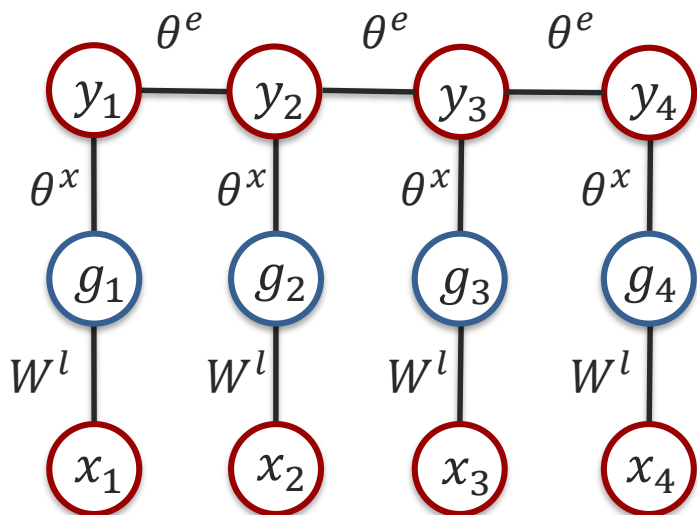


Conditional Neural Fields

$$\mathcal{G}^l(\mathbf{x}_i, \mathbf{W}^l) = [g_1^l(\mathbf{x}_i \cdot \mathbf{W}_1^l), g_2^l(\mathbf{x}_i \cdot \mathbf{W}_i^l), \dots, g_n^l(\mathbf{x}_i \cdot \mathbf{W}_n^l)]$$

$$p(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) \propto \exp \left\{ \sum_i \boldsymbol{\theta}^x \cdot \mathbf{f}^x(y_i, \mathbf{x}_i) + \sum_i \boldsymbol{\theta}^e \cdot \mathbf{f}^e(y_i, y_{i-1}) \right\}$$

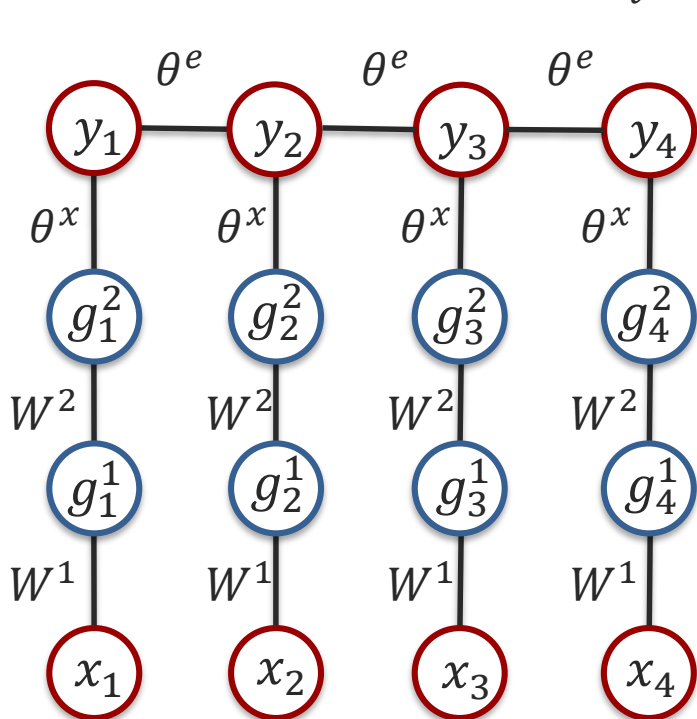
$$\mathbf{f}^x(y_i, \mathbf{x}_i) = \mathbb{I}[y_i = y'] \cdot \mathcal{G}(\mathbf{x}_i, \mathbf{W}^l)$$



Deep Conditional Neural Fields

$$\mathbf{g}^l(\mathbf{x}_i, \mathbf{W}^l) = [g_1^l(\mathbf{x}_i \cdot \mathbf{W}_1^l), g_2^l(\mathbf{x}_i \cdot \mathbf{W}_i^l), \dots, g_n^l(\mathbf{x}_i \cdot \mathbf{W}_n^l)]$$

$$p(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) \propto \exp \left\{ \sum_i \boldsymbol{\theta}^x \cdot \mathbf{f}^x(y_i, \mathbf{x}_i) + \sum_i \boldsymbol{\theta}^e \cdot \mathbf{f}^e(y_i, y_{i-1}) \right\}$$



$$\mathbf{f}^x(y_i, \mathbf{x}_i) = \mathbb{I}[y_i = y'] \cdot \mathcal{G}(\mathbf{a}_i^{m-1}, \mathbf{W}^l)$$

$$\mathbf{a}^l = \mathcal{G}(\mathbf{a}_i^{l-1}, \boldsymbol{\theta}^g) \quad \text{for } l = 2 \dots m - 1$$

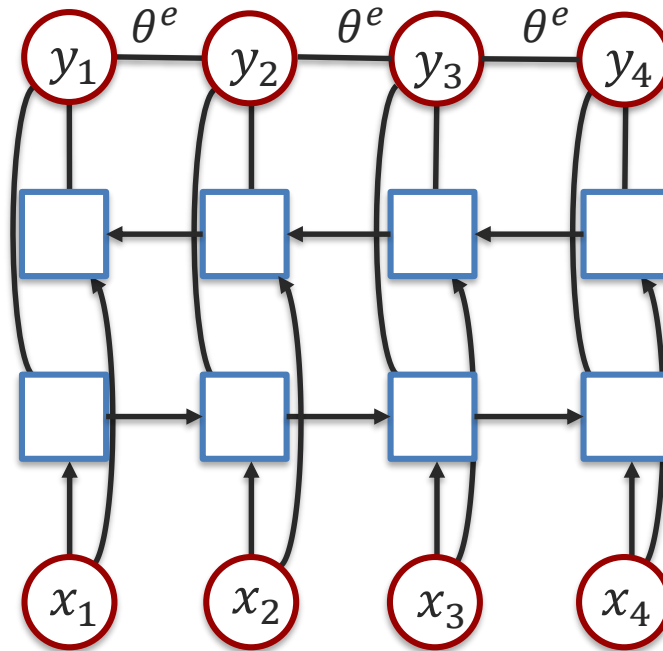
Iterate

CRF and Bilinear LSTM

[Dyer, 2016]

Learning:

1. Feedforward
2. Gradient
 - a) Belief propagation
3. Backpropagation



Output labels:

- Name entities

Input features:

- Word embedding

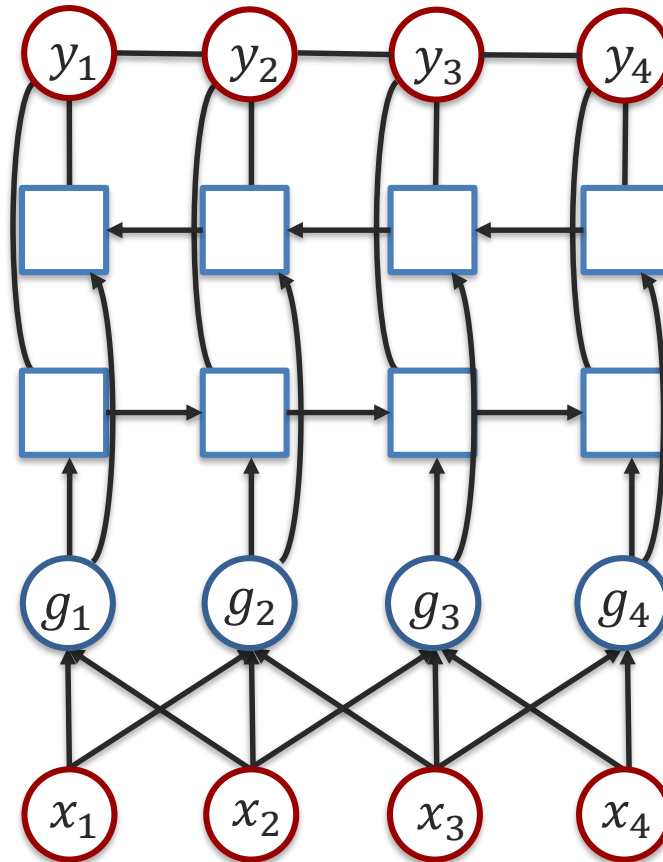
- What did θ^e parameters learn?
- What does LSTM parameters learn?



CNN and CRF and Bilinear LSTM [Hovy, 2016]

Learning:

1. Feedforward
2. Gradient
 - a) Belief propagation
3. Backpropagation



Output labels:

- Name entities

Input features:

- Character embedding



Continuous and Fully-Connected CRFs



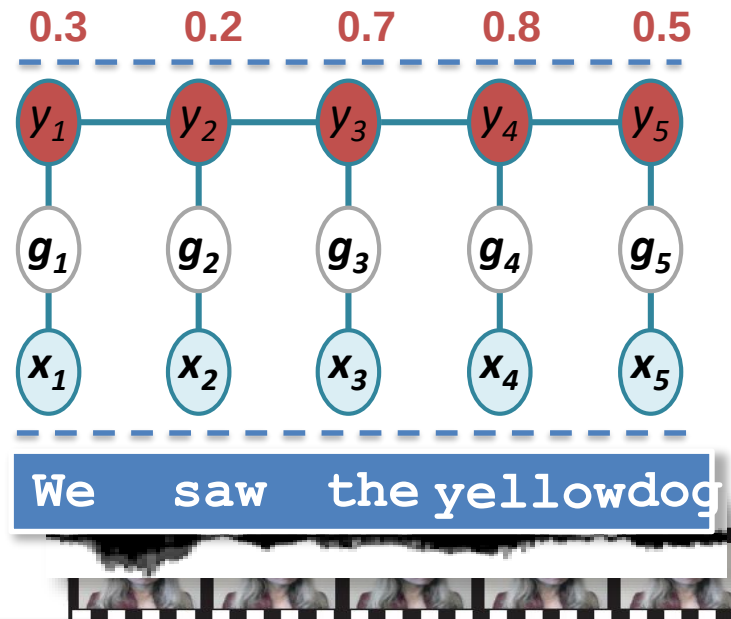
Continuous Conditional Neural Field [Baltrusaitis 2014]

Continuous output variables: (e.g., continuous emotional label)

$$\mathbf{y} = \{y_1, y_2, y_3, \dots, y_t\} \text{ where } y_t \in \mathbb{R}$$

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{Z(\mathbf{x}; \boldsymbol{\theta})} \exp \left\{ \sum_t \boldsymbol{\theta} \cdot F(y_t, y_{t-1}, \mathbf{x}_t, \boldsymbol{\theta}^g) \right\}$$

$$Z(\mathbf{x}; \boldsymbol{\theta}) = \int_{-\infty}^{\infty} \exp \left\{ \sum_t \boldsymbol{\theta} \cdot F(y_t, y_{t-1}, \mathbf{x}_t, \boldsymbol{\theta}^g) \right\} d\mathbf{y}$$



➤ How to solve

Multivariate Gaussian integral:

$$\int_{-\infty}^{\infty} \exp \left\{ \frac{1}{2} \mathbf{y}^T \Sigma^{-1} \mathbf{y} + \mathbf{y} \Sigma^{-1} \boldsymbol{\mu} \right\} d\mathbf{y}$$

$$= \frac{(2\pi)^{n/2}}{|\Sigma^{-1}|^{1/2}} \exp \left(\frac{1}{2} \boldsymbol{\mu} \Sigma^{-1} \boldsymbol{\mu} \right)$$

[Radosavljevic et al., 2010]



Continuous Conditional Neural Field

Continuous output variables: (e.g., continuous emotional label)

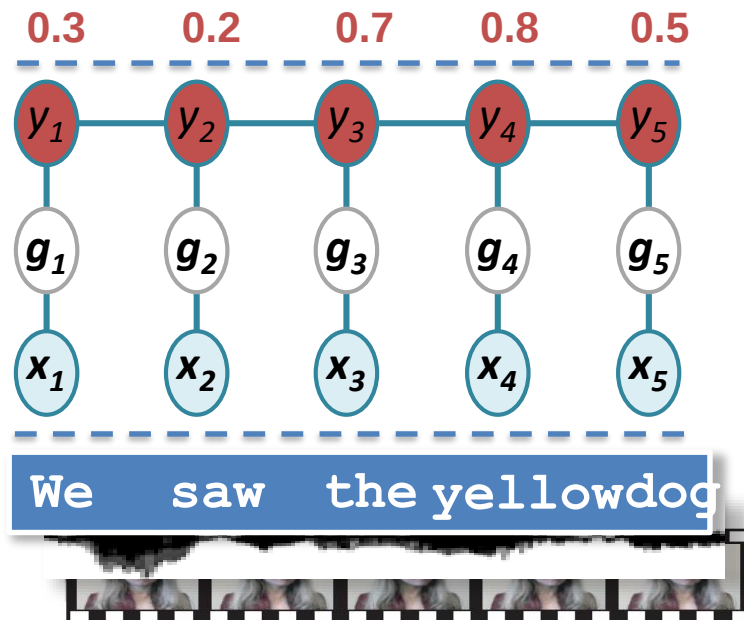
$\mathbf{y} = \{y_1, y_2, y_3, \dots, y_t\}$ where $y_t \in \mathbb{R}$

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{Z(\mathbf{x}; \boldsymbol{\theta})} \exp \left\{ \sum_t \boldsymbol{\theta} \cdot F(y_t, y_{t-1}, \mathbf{x}_t, \boldsymbol{\theta}^g) \right\}$$

$$Z(\mathbf{x}; \boldsymbol{\theta}) = \int_{-\infty}^{\infty} \exp \left\{ \sum_t \boldsymbol{\theta} \cdot F(y_t, y_{t-1}, \mathbf{x}_t, \boldsymbol{\theta}^g) \right\} d\mathbf{y}$$

$$f^x(y_t, x_t, \boldsymbol{\theta}^g) = -(y_t - g_k(x_t, \boldsymbol{\theta}_k^g))^2$$

$$f^e(y_t, y_{t-1}) = -\frac{1}{2}(y_t - y_{t-1})^2$$



Continuous Conditional Neural Field

Continuous output variables: (e.g., continuous emotional label)

$$\mathbf{y} = \{y_1, y_2, y_3, \dots, y_t\} \text{ where } y_t \in \mathbb{R}$$

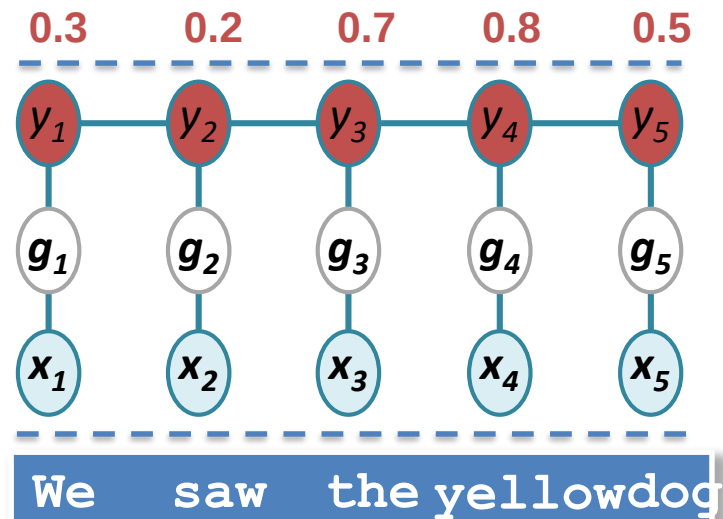
Multivariate Gaussian distribution:

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \boldsymbol{\mu})\right)$$

where $\boldsymbol{\Sigma}$

matrix $\mathbf{V}$

and $\boldsymbol{\mu} = \boldsymbol{\Sigma} \mathbf{V}$



Since CCNF can be viewed as a multivariate Gaussian, the prediction of $\mathbf{y}'$ is simply the mean value of distribution:

$$\mathbf{y}' = \arg \max_{\mathbf{y}} (P(\mathbf{y} | \mathbf{x})) = \boldsymbol{\mu}$$

➤ Optimized using gradient ascent or BFGS.



High-Order Continuous Conditional Neural Field

Continuous output variables: (e.g., continuous emotional label)

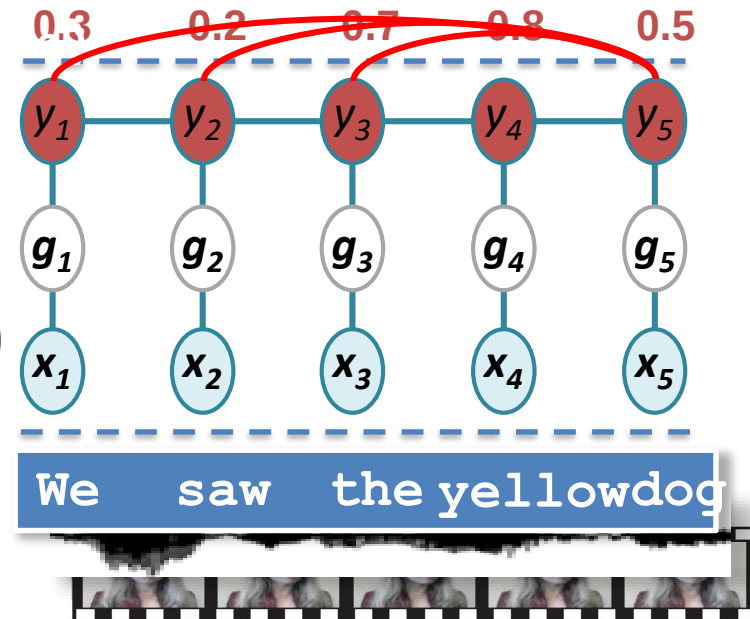
$$\mathbf{y} = \{y_1, y_2, y_3, \dots, y_t\} \text{ where } y_t \in \mathbb{R}$$

Multivariate Gaussian distribution:

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \boldsymbol{\mu})\right)$$

k -order potential functions:

$$f^{e_k}(y_t, y_{t-k}) = -\frac{1}{2}(y_t - y_{t-k})^2$$



Fully-Connected Continuous Conditional Neural Field

Continuous output variables: (e.g., continuous emotional label)

$$\mathbf{y} = \{y_1, y_2, y_3, \dots, y_t\} \text{ where } y_t \in \mathbb{R}$$

Multivariate Gaussian distribution:

$$p(\mathbf{y} | \mathbf{x}; \boldsymbol{\theta}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{y} - \boldsymbol{\mu})\right)$$

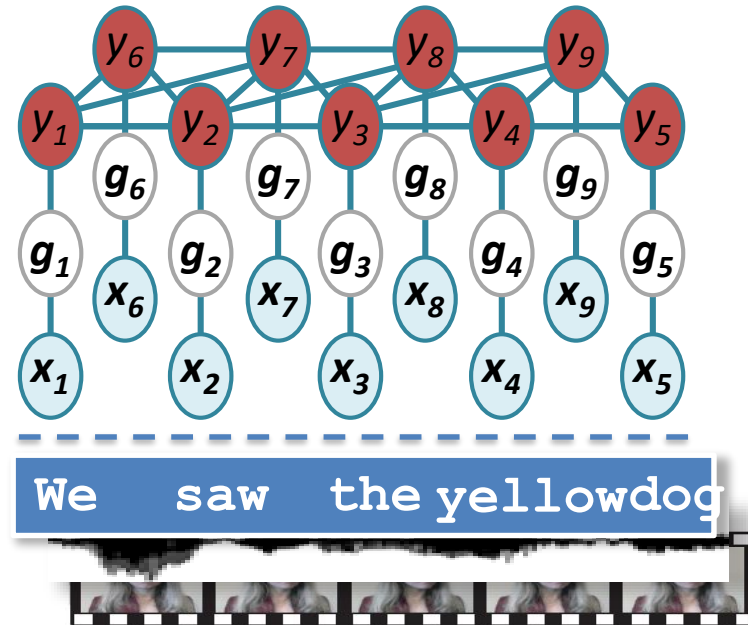
k -order potential functions:

$$f^{e_k}(y_t, y_{t-k}) = -\frac{1}{2}(y_t - y_{t-k})^2$$

Grid potential functions:

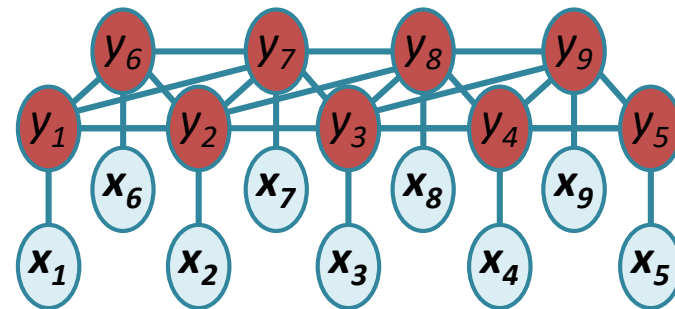
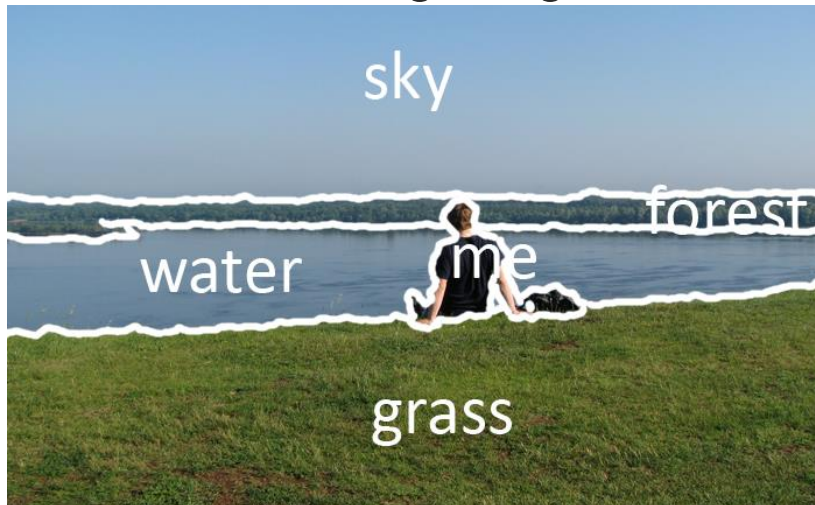
$$f^{2D}(y_i, y_j) = -\frac{1}{2} s_{ij} (y_i - y_j)^2$$

where $s_{i,j}$ specifies which nodes are connected.



Fully-Connected CRF [Krahenbuhl and Koltun, 2013]

“Semantic” image segmentation



y_i : object class label

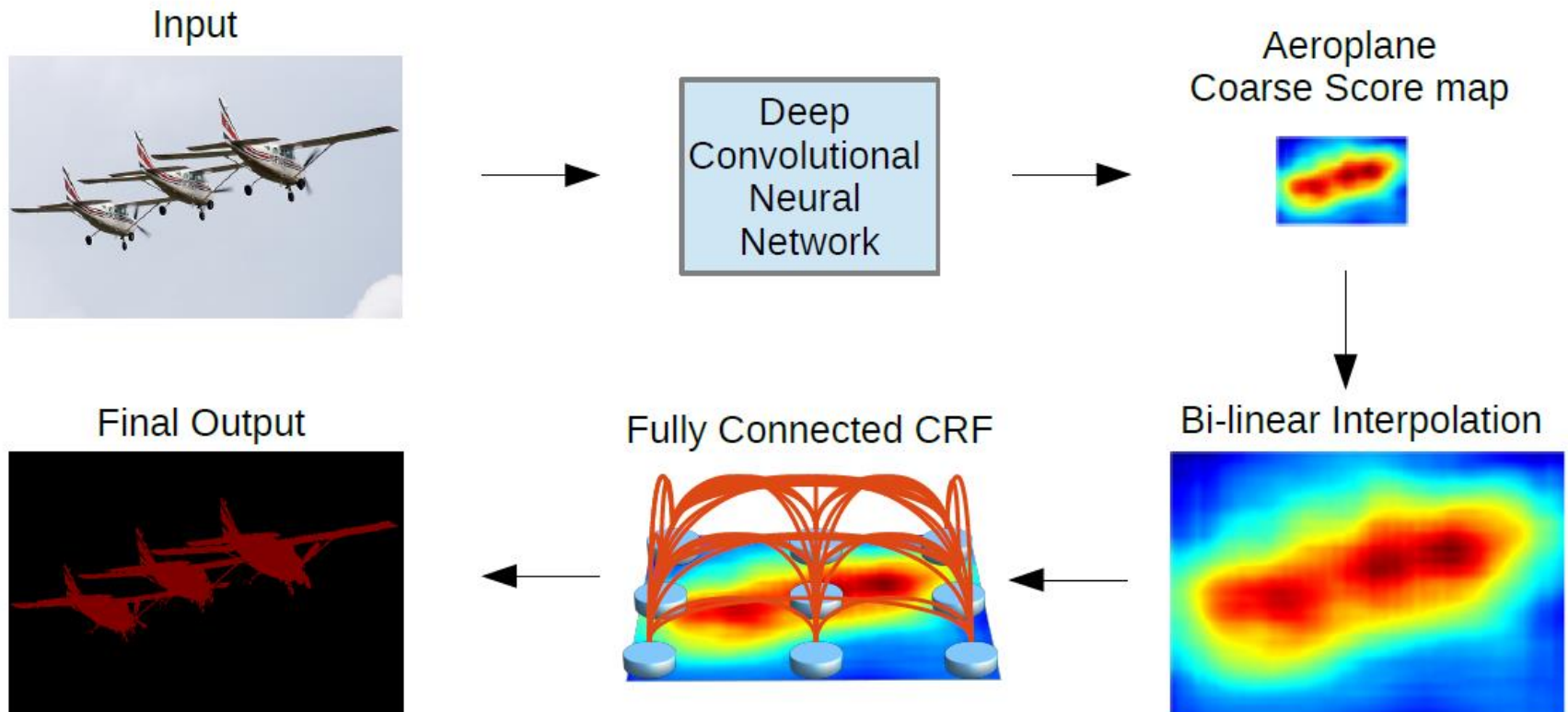
x_i : local pixel features

$$p(\mathbf{y}|\mathbf{x}; \theta) = \frac{\Phi(\mathbf{y},; \theta)}{\sum_{\mathbf{y}'} \Phi(\mathbf{y}', \mathbf{x}; \theta)}$$

where

$$\Phi_{ij}(y_i, y_j; \boldsymbol{\theta}) = \sum_{m=1}^c \underbrace{u^{(m)}(y_i, y_j | \boldsymbol{\theta}) k^{(m)}(\mathbf{x}_i, \mathbf{x}_j)}_{\text{Mixture of kernels}}$$

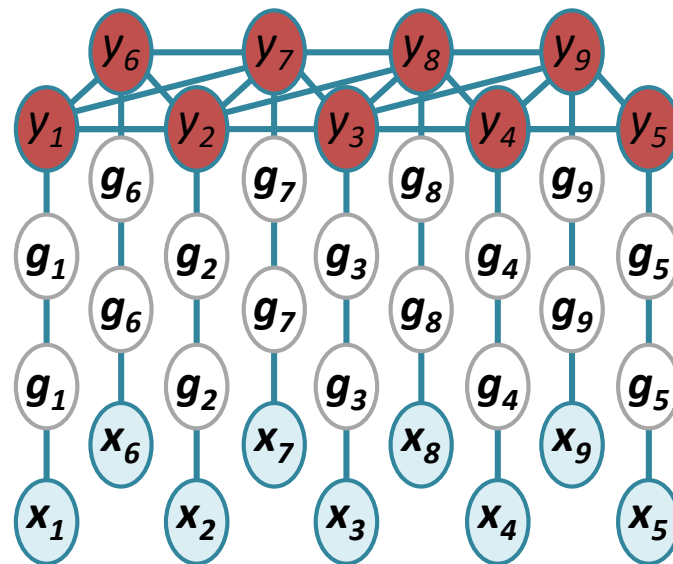
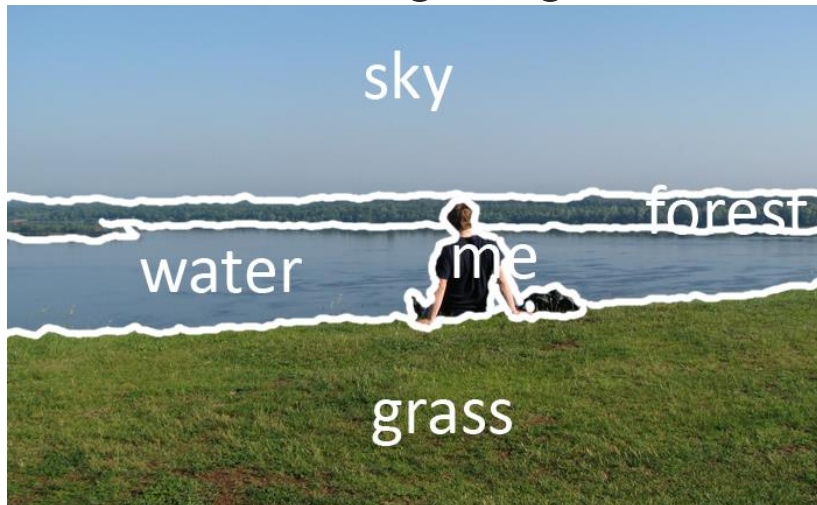
CNN and Fully-Connected CRF [Chen et al., 2014]



Fully Connected Deep Structured Networks

[Zheng et al., 2015; Schwing and Urtasun, 2015]

“Semantic” image segmentation



Algorithm: Learning Fully Connected Deep Structured Models

Repeat until stopping criteria

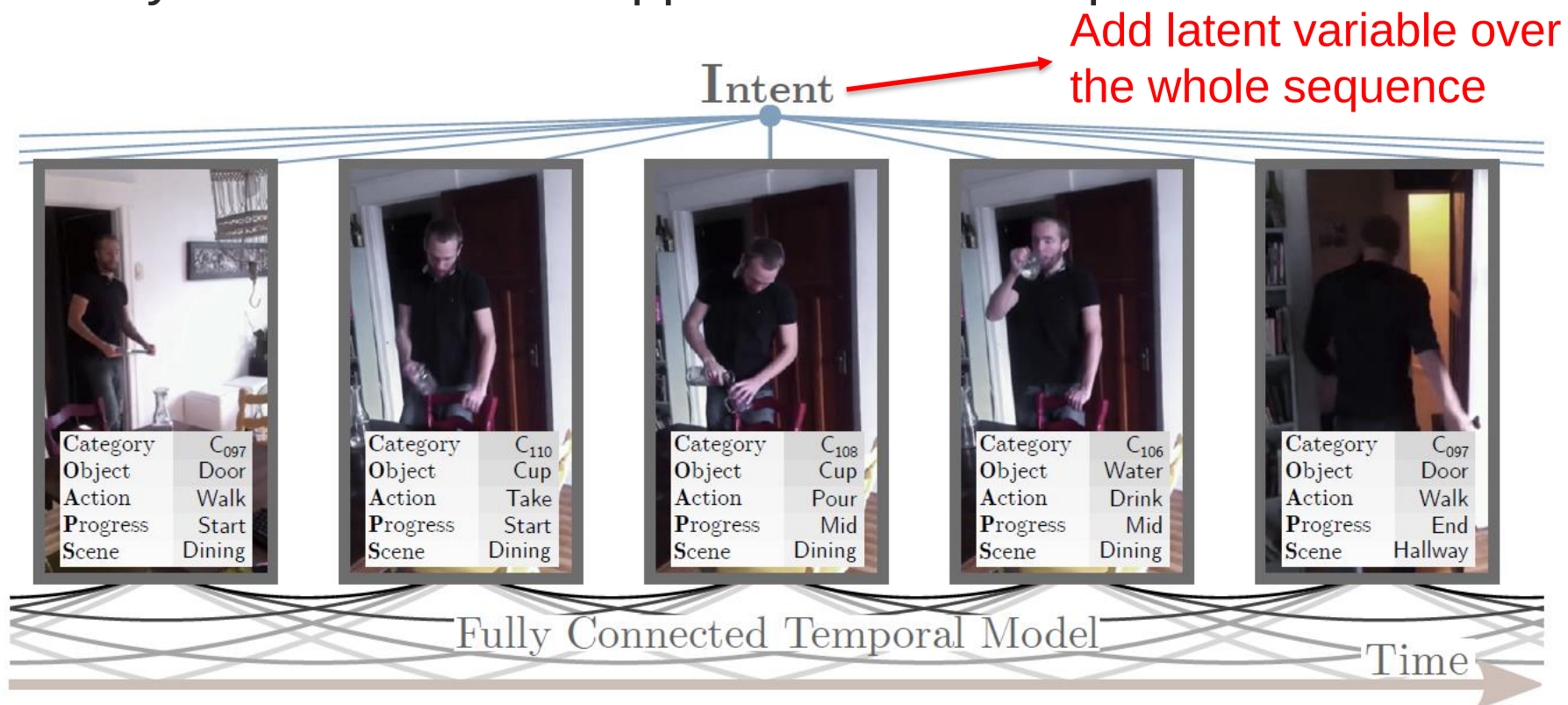
1. Forward pass to compute $f_r(x, \hat{y}_r; w) \forall r \in \mathcal{R}, y_r \in \mathcal{Y}_r$
2. Computation of marginals $q_{(x,y),i}^t(\hat{y}_i)$ via filtering for $t \in \{1, \dots, T\}$
3. Backtracking through the marginals $q_{(x,y),i}^t(\hat{y}_i)$ from $t = T - 1$ down to $t = 1$
4. Backward pass through definition of function via chain rule
5. Parameter update

Using mean field approximation



Fully-Connected Temporal CRF

Fully-connected CRF applied to video sequence:



Sigurdsson et al., Asynchronous Temporal Fields for Action Recognition, CVPR 2017