

A brief introduction to Conditional Random Fields

Mark Johnson
Brown University

April, 2005

Talk outline

- Graphical models
- Maximum likelihood and maximum conditional likelihood estimation
- Naive Bayes and Maximum Entropy Models
- Hidden Markov Models
- Conditional Random Fields

Classification with structured labels

- Classification: predicting label \mathbf{y} given features \mathbf{x}

$$\mathbf{y}^*(\mathbf{x}) = \arg \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x})$$

- Naive Bayes and Maxent models: label \mathbf{y} is atomic, \mathbf{x} can be structured (e.g., set of features)
- HMMs and CRFs are extensions of Naive Bayes and Maxent models where \mathbf{y} is structured too
- HMMs and CRFs model dependencies between components \mathbf{y}_i of label \mathbf{y}
- Example: Part of speech tagging: \mathbf{x} is a sequence of words, \mathbf{y} is corresponding sequence of parts of speech (e.g., noun, verb, etc.)

Why graphical models?

- Graphical models depict *factorizations of probability distributions*
- Statistical and computational properties depend on the factorization
 - complexity of dynamic programming is size of a certain cut in the graphical model
- Two different (but related) graphical representations
 - *Bayes nets* (directed graphs; products of conditionals)
 - *Markov Random Fields* (undirected graphs; products of arbitrary terms)
- Each random variable X_i is represented by a node

Bayes nets (directed graph)

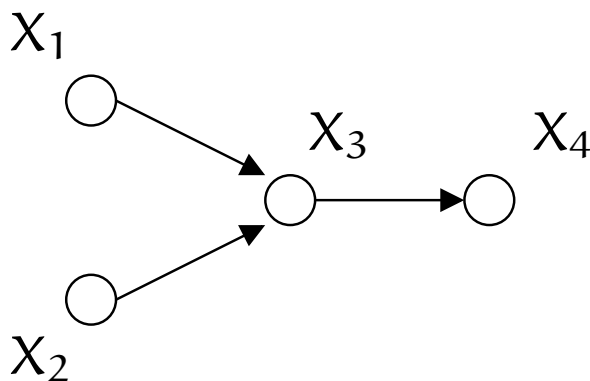
- Factorize *joint* $P(X_1, \dots, X_n)$ into *product of conditionals*

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | X_{\text{Pa}(i)})$$

where $\text{Pa}(i) \subseteq (X_1, \dots, X_{i-1})$

- The *Bayes net* contains an arc from each $j \in \text{Pa}(i)$ to i

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2)P(X_3|X_1, X_2)P(X_4|X_3)$$



Markov Random Field (undirected)

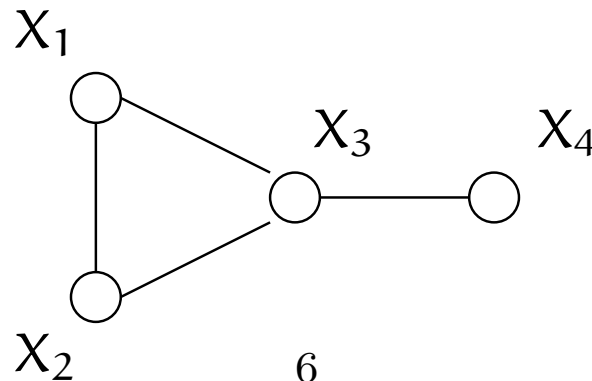
- Factorize $P(X_1, \dots, X_n)$ into product of *potentials* $g_c(X_c)$, where $c \subseteq \{1, \dots, n\}$ and $c \in \mathcal{C}$ (a set of tuples of indices)

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{c \in \mathcal{C}} g_c(X_c)$$

- If $i, j \in c \in \mathcal{C}$, then an edge connects i and j

$$\mathcal{C} = \{(1, 2, 3), (3, 4)\}$$

$$P(X_1, X_2, X_3, X_4) = \frac{1}{Z} g_{123}(X_1, X_2, X_3) g_{34}(X_3, X_4)$$



A rose by any other name ...

- MRFs have the same form as *Maximum Entropy models*, *Exponential models*, *Log-linear models*, *Harmony models*, ...

$$\begin{aligned} P(\mathbf{X}) &= \frac{1}{Z} \prod_{c \in \mathcal{C}} g_c(\mathbf{X}_c) \\ &= \frac{1}{Z} \prod_{c \in \mathcal{C}, \mathbf{x}_c \in \mathcal{X}_c} (\theta_{\mathbf{x}_c = \mathbf{x}_c})^{\llbracket \mathbf{X}_c = \mathbf{x}_c \rrbracket}, \text{ where } \theta_{\mathbf{x}_c = \mathbf{x}_c} = g_c(\mathbf{x}_c) \\ &= \frac{1}{Z} \exp \sum_{c \in \mathcal{C}, \mathbf{x}_c \in \mathcal{X}_c} \llbracket \mathbf{X}_c = \mathbf{x}_c \rrbracket \phi_{\mathbf{x}_c = \mathbf{x}_c}, \text{ where } \phi_{\mathbf{x}_c = \mathbf{x}_c} = \log g_c(\mathbf{x}_c) \end{aligned}$$

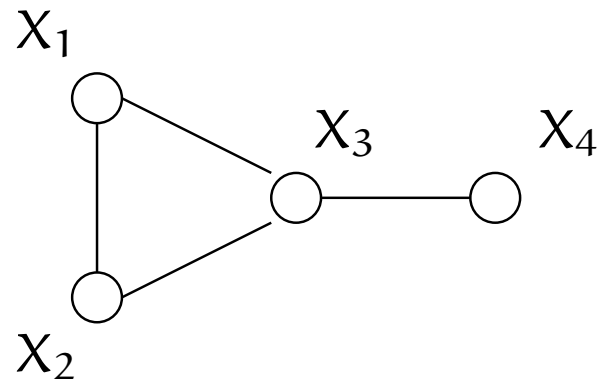
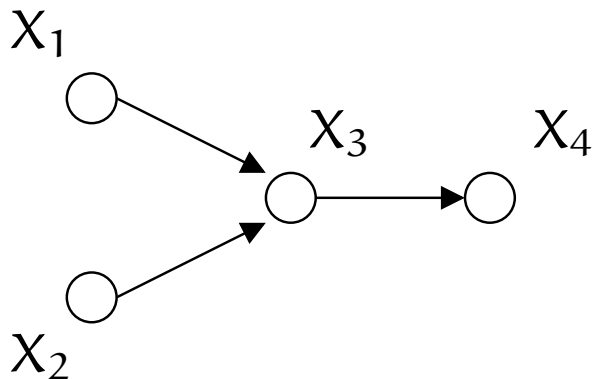
$$\begin{aligned} P(\mathbf{X}) &= \frac{1}{Z} g_{123}(\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3) g_{34}(\mathbf{X}_3, \mathbf{X}_4) \\ &= \frac{1}{Z} \exp \left(\begin{array}{l} \llbracket \mathbf{X}_{123} = 000 \rrbracket \phi_{000} + \llbracket \mathbf{X}_{123} = 001 \rrbracket \phi_{001} + \dots \\ \llbracket \mathbf{X}_{34} = 00 \rrbracket \phi_{00} + \llbracket \mathbf{X}_{34} = 01 \rrbracket \phi_{01} + \dots \end{array} \right) \end{aligned}$$

Bayes nets and MRFs

- MRFs are more general than Bayes nets
- Its easy to find the MRF representation of a Bayes net

$$P(X_1, X_2, X_3, X_4) = \underbrace{P(X_1)P(X_2)P(X_3|X_1, X_2)}_{g_{123}(X_1, X_2, X_3)} \underbrace{P(X_4|X_3)}_{g_{34}(X_3, X_4)}$$

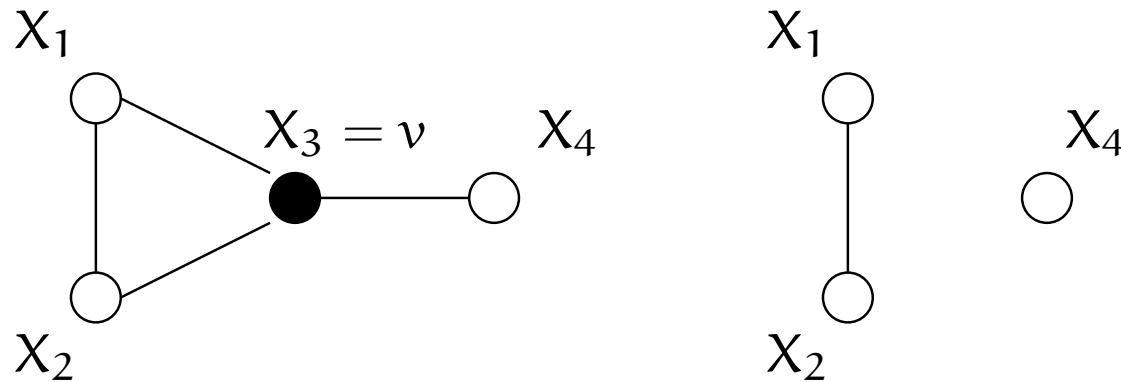
- *Moralization*, i.e, “marry the parents”



Conditionalization in MRFs

- Conditionalization is *fixing the value of some variables*
- To get a MRF representation of the conditional distribution, *delete nodes whose values are fixed and arcs connected to them*

$$\begin{aligned} P(X_1, X_2, X_4 | X_3 = v) &= \frac{1}{Z} \frac{1}{P(X_3 = v)} g_{123}(X_1, X_2, v) g_{34}(v, X_4) \\ &= \frac{1}{Z'(v)} g'_{12}(X_1, X_2) g'_4(X_4) \end{aligned}$$



Classification

- Given value of X , predict value of Y
- Given a probabilistic model $P(Y|X)$, predict:

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

- In general we must learn $P(Y|X)$ from data
 $D = ((x_1, y_1), \dots, (x_n, y_n))$
- Restrict attention to a *parametric model class* P_θ parameterized by parameter vector θ
 - learning is estimating θ from D

ML and CML Estimation

- Maximum likelihood estimation (MLE) picks the θ that makes the data $D = (\mathbf{x}, \mathbf{y})$ as *likely as possible*

$$\hat{\theta} = \arg \max_{\theta} P_{\theta}(\mathbf{x}, \mathbf{y})$$

- Conditional maximum likelihood estimation (CMLE) picks the θ that maximizes *conditional likelihood* of the data $D = (\mathbf{x}, \mathbf{y})$

$$\hat{\hat{\theta}} = \arg \max_{\theta} P_{\theta}(\mathbf{y}|\mathbf{x})$$

- $P(X, Y) = P(X)P(Y|X)$, so CMLE *ignores* $P(X)$

MLE and CMLE example

- $X, Y \in \{0, 1\}$, $\theta \in [0, 1]$, $P_\theta(X = 1) = \theta$, $P_\theta(Y = X|X) = \theta$

Choose X by flipping a coin with weight θ , then set Y to same value as X if flipping same coin again comes out 1.

- Given data $D = ((x_1, y_1), \dots, (x_n, y_n))$,

$$\hat{\theta} = \frac{\sum_i^n \mathbb{I}[x_i = 1] + \mathbb{I}[x_i = y_i]}{2n}$$

$$\hat{\hat{\theta}} = \frac{\sum_i^n \mathbb{I}[x_i = y_i]}{n}$$

- CMLE *ignores* $P(X)$, so *less efficient* if model *correctly* relates $P(Y|X)$ and $P(X)$
- But if model *incorrectly relates* $P(Y|X)$ and $P(X)$, MLE converges to wrong θ
 - e.g., if x_i are chosen by some different process entirely

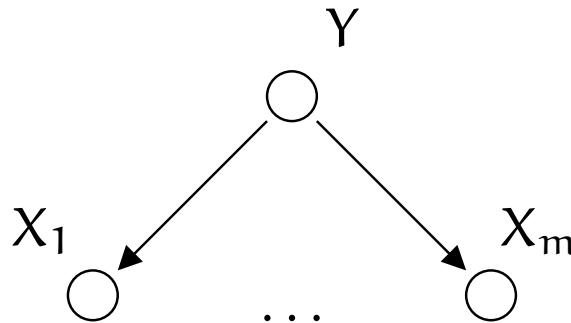
Complexity of decoding and estimation

- Finding $\mathbf{y}^*(\mathbf{x}) = \arg \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x})$ is *equally hard* for Bayes nets and MRFs with similar architectures
- A Bayes net is a product of independent conditional probabilities
 - ⇒ MLE is *relative frequency* (easy to compute)
 - *no closed form for CMLE* if conditioning variables have parents
- A MRF is a product of arbitrary potential functions \mathbf{g}
 - estimation involves learning values of each \mathbf{g} takes
 - partition function Z changes as we adjust \mathbf{g}
- ⇒ usually *no closed form for MLE and CMLE*

Multiple features and Naive Bayes

- Predict label Y from features X_1, \dots, X_m

$$\begin{aligned} P(Y|X_1, \dots, X_m) &\propto P(Y) \prod_{j=1}^m P(X_j|Y, X_1, \dots, X_{j-1}) \\ &\approx P(Y) \prod_{j=1}^m P(X_j|Y) \end{aligned}$$

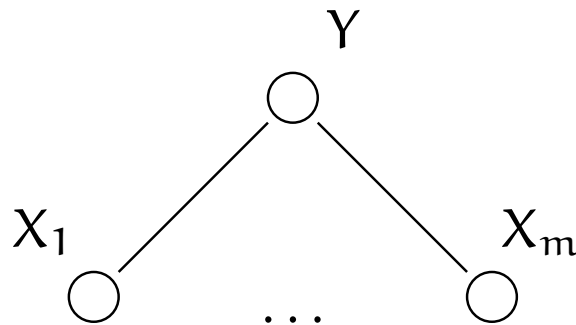


- Naive Bayes estimate is MLE $\hat{\theta} = \arg \max_{\theta} P(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{y})$
 - Trivial to compute (relative frequency)
 - May be poor if X_j aren't really conditionally independent

Multiple features and MaxEnt

- Predict label Y from features X_1, \dots, X_m

$$P(Y|X_1, \dots, X_m) \propto \prod_{j=1}^m g_j(X_j, Y)$$



- MaxEnt estimate is CMLE $\hat{\hat{\theta}} = \arg \max_{\theta} P(\mathbf{y}|\mathbf{x}_1, \dots, \mathbf{x}_m)$
 - Makes *no assumptions* about $P(\mathbf{X})$
 - Difficult to compute (iterative numerical optimization)

Sequence labeling

- Predict labels Y_1, \dots, Y_m given features X_1, \dots, X_m
- Example: Parts of speech

$Y =$ DT JJ NN VBS JJR

$X =$ the big dog barks loudly

- Example: Named entities

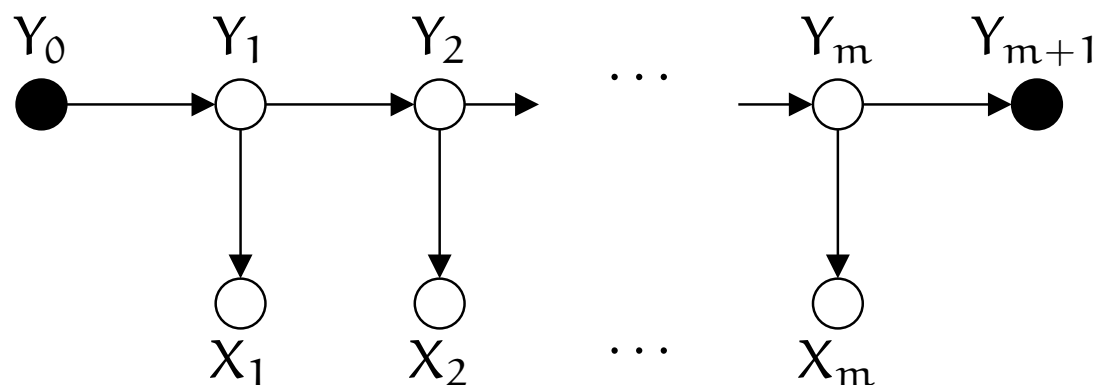
$Y =$ [NP NP NP] — —

$X =$ the big dog barks loudly

- Example: X_1, \dots, X_m are image regions, each X_j is labeled Y_j

Hidden Markov Models

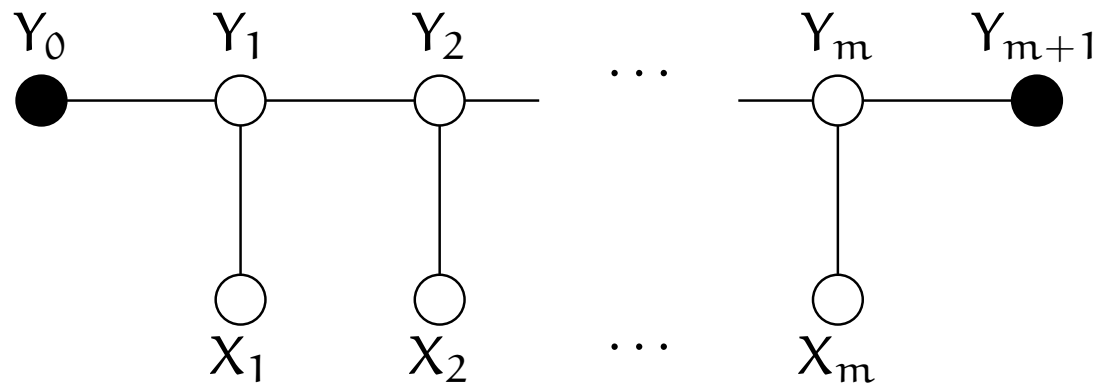
$$P(X, Y) = \left(\prod_{j=1}^m P(Y_j|Y_{j-1})P(X_j|Y_j) \right) P(Y_m, \text{stop})$$



- Usually assume *time invariance* or *stationarity* i.e., $P(Y_j|Y_{j-1})$ and $P(X_j|Y_j)$ do not depend on j
- HMMs are Naive Bayes models with compound labels Y
- Estimator is MLE $\hat{\theta} = \arg \max_{\theta} P_{\theta}(x, y)$

Conditional Random Fields

$$P(Y|X) = \frac{1}{Z(x)} \left(\prod_{j=1}^m f(Y_j, Y_{j-1}) g(X_j, Y_j) \right) f(Y_m, \text{stop})$$



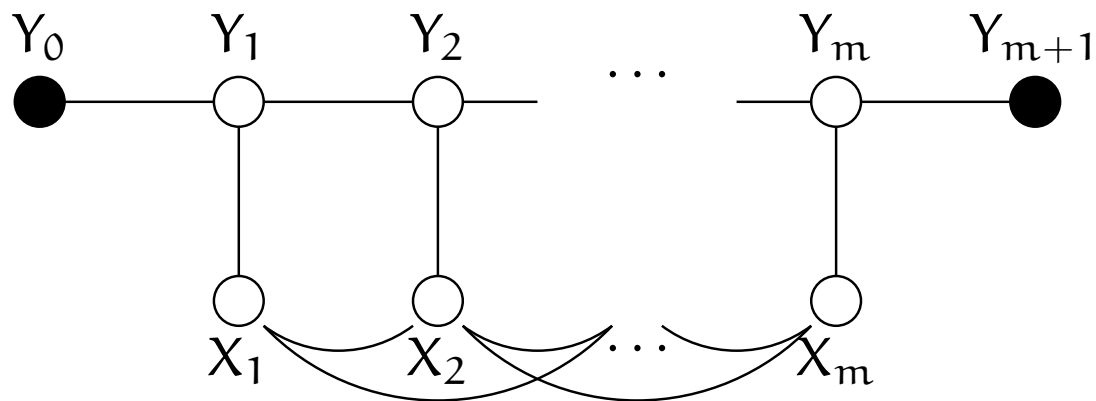
- *time invariance* or *stationarity*, i.e., f and g don't depend on j
- CRFs are MaxEnt models with compound labels Y
- Estimator is CMLE $\hat{\theta} = \arg \max_{\theta} P_{\theta}(y|x)$

Decoding and Estimation

- HMMs and CRFs have *same complexity of decoding* i.e., computing $\mathbf{y}^*(\mathbf{x}) = \arg \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x})$
 - dynamic programming algorithm (Viterbi algorithm)
- Estimating a HMM from labeled data (\mathbf{x}, \mathbf{y}) is *trivial*
 - HMMs are Bayes nets \Rightarrow MLE is relative frequency
- Estimating a CRF from labeled data (\mathbf{x}, \mathbf{y}) is *difficult*
 - Usually *no closed form* for partition function $Z(\mathbf{x})$
 - Use *iterative numerical optimization procedures* (e.g., Conjugate Gradient, Limited Memory Variable Metric) to maximize $P_{\theta}(\mathbf{y}|\mathbf{x})$

When are CRFs better than HMMs?

- When HMM independence assumptions are wrong, i.e., there are dependences between X_j not described in model

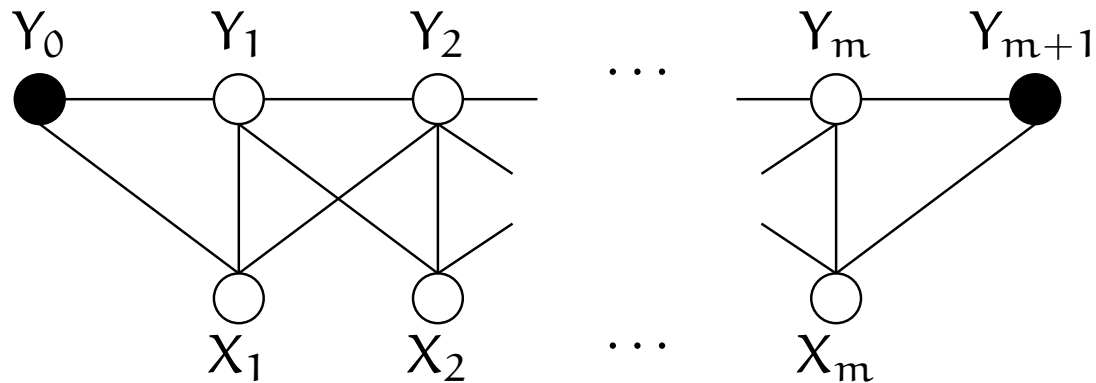


- HMM uses MLE \Rightarrow models joint $P(X, Y) = P(X)P(Y|X)$
- CRF uses CMLE \Rightarrow models conditional distribution $P(Y|X)$
- Because CRF uses CMLE, it makes no assumptions about $P(X)$
- *If $P(X)$ isn't modeled well by HMM, don't use HMM!*

Overlapping features

- Sometimes label Y_j depends on X_{j-1} and X_{j+1} as well as X_j

$$P(Y|X) = \frac{1}{Z(x)} \left(\prod_{j=1}^m f(X_j, Y_j, Y_{j-1}) g(X_j, Y_j, Y_{j+1}) \right)$$



- Most people think this would be difficult to do in a HMM

Summary

- HMMs and CRFs both associate a *sequence* of labels (Y_1, \dots, Y_m) to items (X_1, \dots, X_m)
- HMMs are Bayes nets and estimated by MLE
- CRFs are MRFs and estimated by CMLE
- HMMs assume that X_j are *conditionally independent*
- CRFs do not assume that the X_j are conditionally independent
- The Viterbi algorithm computes $\mathbf{y}^*(\mathbf{x})$ for both HMMs and CRFs
- HMMs are trivial to estimate
- CRFs are difficult to estimate
- It is easier to add new features to a CRF
- There is no EM version of CRF

HMM with longer range dependencies

