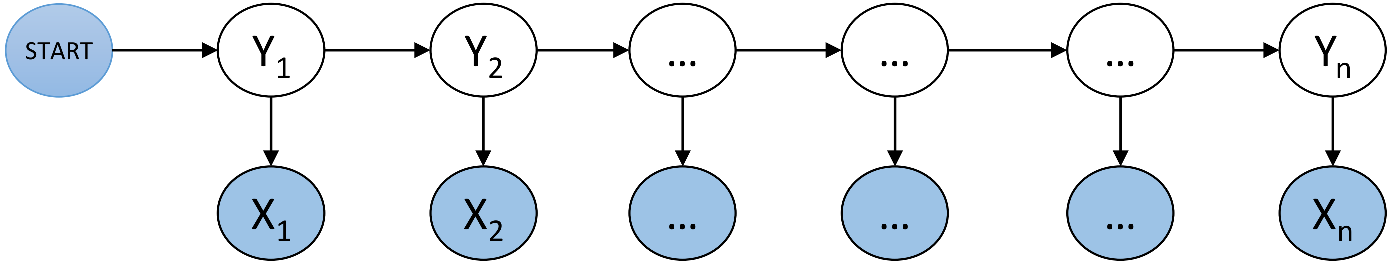
CS60075
Natural Language Processing
Autumn 2020

Module 4: Part 3
CRF for POS Tagging

Sep 25 2020

Hidden Markov Model



$$p(s,x) = p(s_1)p(x_1 \mid s_1)\prod_{i=2}^{n} p(s_i \mid s_{i-1})p(x_i \mid s_i)$$

HMM models capture dependences between each state and only its corresponding observation

Cannot represent multiple interacting features or long range dependences between observed elements.

Discriminative Vs. Generative

$p(\mathbf{y}, \mathbf{x})$

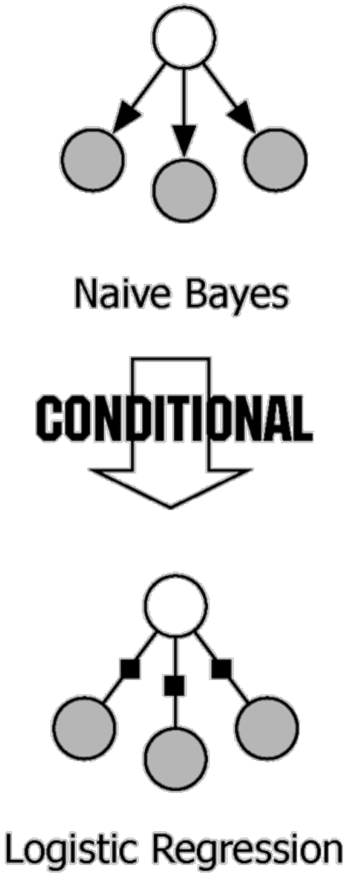
- **Generative Model:** A model that generate observed data randomly
- **Naïve Bayes:** once the class label is known, all the features are independent

$$p(y, \mathbf{x}) = p(y) \prod_{k=1}^{K} p(x_k | y)$$

$p(\mathbf{y} | \mathbf{x})$

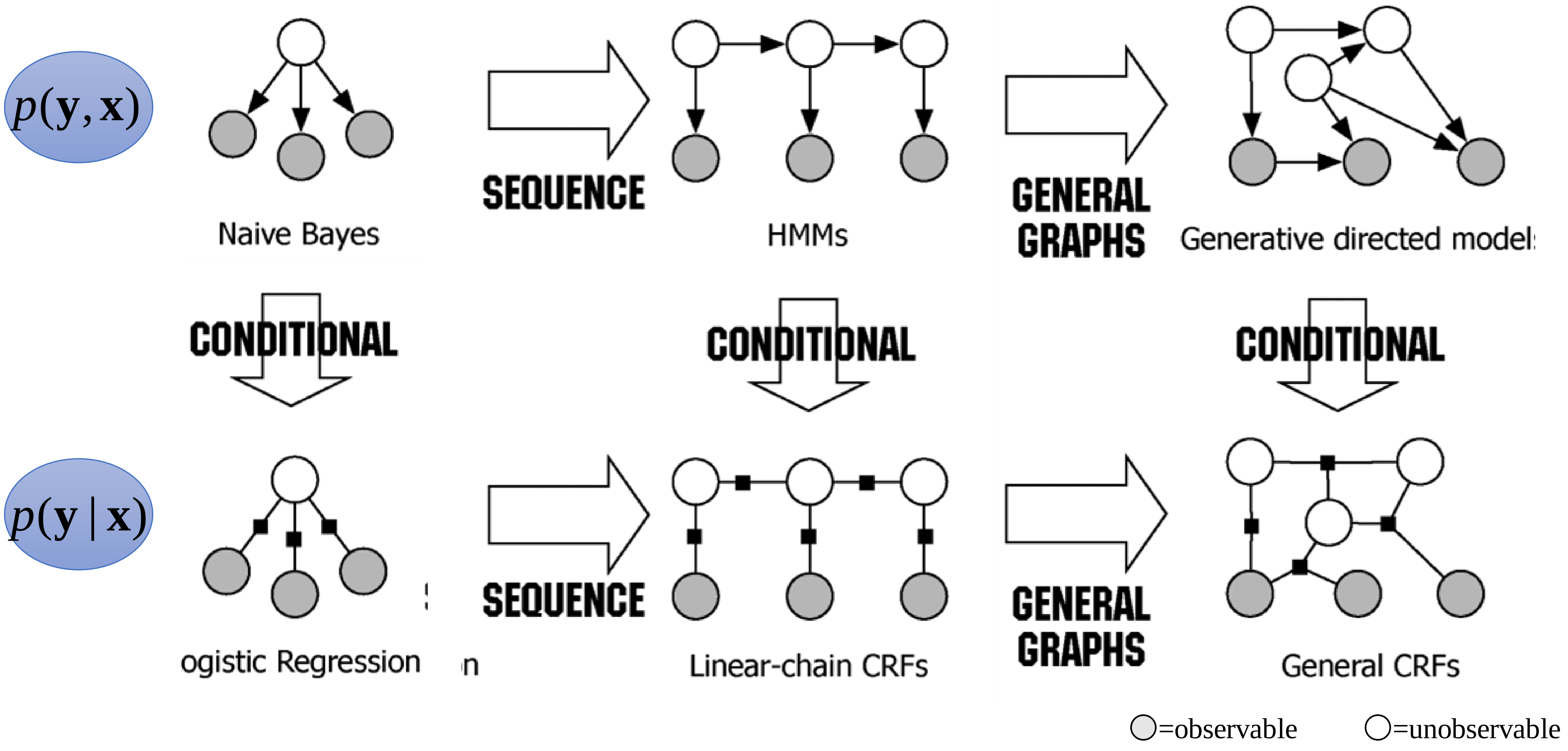
- **Discriminative:** Directly estimate the posterior probability; Aim at modeling the "discrimination" between different outputs
- **MaxEnt** classifier: linear combination of feature function in the exponent,

$$p(y|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp \left\{ \sum_{k=1}^{K} \theta_k f_k(y, \mathbf{x}) \right\}$$



Both generative models and discriminative models describe distributions over (y , x), but they work in different directions.

Discriminative Vs. Generative



Markov Networks

- Undirected graph over a set of random variables, where an edge represents a dependency.
- The **Markov blanket** of a node, *X*, in a Markov Net is the set of its neighbors in the graph (nodes that have an edge connecting to *X*).
- Every node in a Markov Net is conditionally independent of every other node given its Markov blanket.

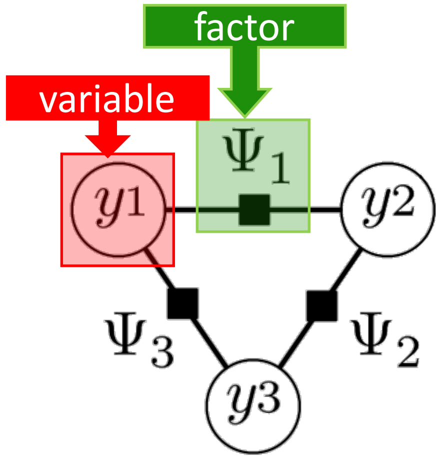
Distribution for a Markov Network

- The distribution of a Markov net is most compactly described in terms of a set of **potential functions**, ψ_k, for each clique, k, in the graph.
- For each joint assignment of values to the variables in clique k, ψ_k assigns a non-negative real value that represents the compatibility of these values.
- The joint distribution of variables $\boldsymbol{y}$:

$$p(\mathbf{y}) = \frac{1}{Z}\prod_{C}\psi_C(\mathbf{y}_C),\ \ Z = \sum_{\mathbf{y}}\prod_{C}\psi_C(\mathbf{y}_C)$$

$$\psi_C(\mathbf{y}_C) \geq 0$$

Typically $\psi_C(\mathbf{y}_C) = \exp\{-E(\mathbf{y}_C)\}$

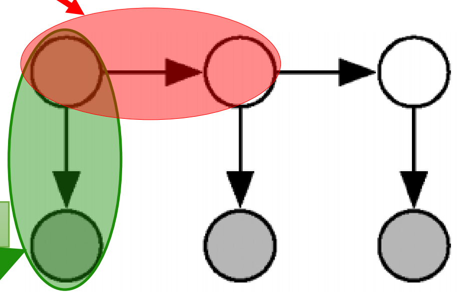


$$\boxed{p(y_1, y_2, y_3) \propto \Psi_1(y_1, y_2)\Psi_2(y_2, y_3)\Psi_3(y_1, y_3)}$$

Sequence prediction

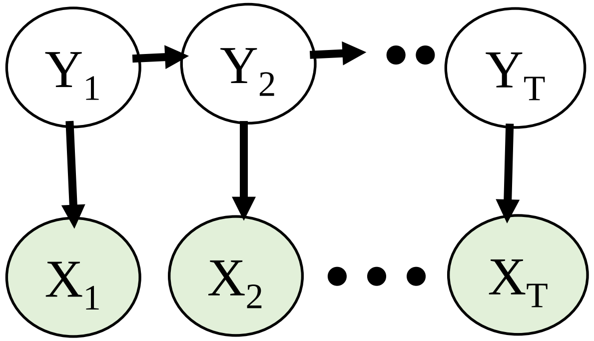
- NER: identifying and classifying proper names in text,
  - Set of observation, $X = \{x_t\}_{t=1}^{T}$
  - Set of underlying sequence of states, $Y = \{y_t\}_{t=1}^{T}$
- HMM is generative:

$$p(\mathbf{y}, \mathbf{x}) = \prod_{t=1} \overbrace{p(y_t|y_{t-1})}^{\text{Transition probability}} \underbrace{p(x_t|y_t)}_{\text{Observation probability}}$$



- Doesn't model long-range dependencies
- Not practical to represent multiple interacting features (hard to model p(x))
- CRFs :
  - conditional nature, resulting in the relaxation of the independence assumptions
  - it can handle overlapping features

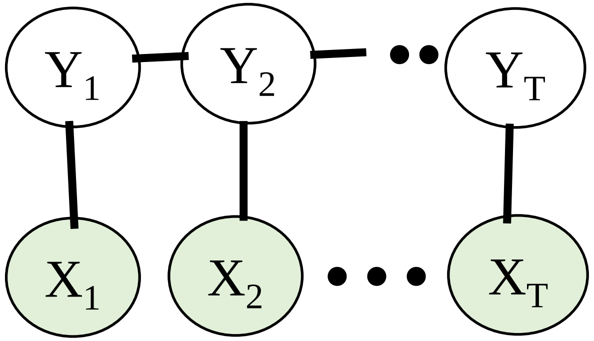
Sequence Labeling



HMM

Generative

Discriminative



Linear-chain CRF

Simple Linear Chain CRF Features

- Models the conditional distribution.
- Create feature functions $f_k(Y_t, Y_{t-1}, X_t)$
  - Feature for each state transition pair *i, j*
    - $f_{i,j}(Y_t, Y_{t-1}, X_t) = 1$ if $Y_t = i$ and $Y_{t-1} = j$ and 0 otherwise
  - Feature for each state observation pair *i, o*
    - $f_{i,o}(Y_t, Y_{t-1}, X_t) = 1$ if $Y_t = i$ and $X_t = o$ and 0 otherwise

- **Note**: number of features grows quadratically in the number of states (i.e. tags)

Conditional Distribution for Linear Chain CRF

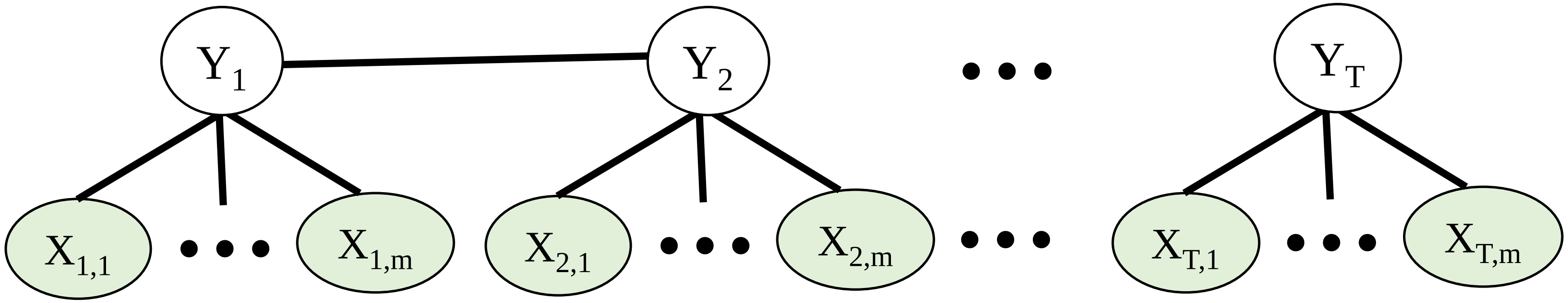
Using these feature functions for a simple linear chain CRF, we can define:

$$P(Y \mid X) = \frac{1}{Z(X)} \exp\left(\sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(Y_t, Y_{t-1}, X_t)\right)$$

$$Z(X) = \sum_{Y} \exp\left(\sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(Y_t, Y_{t-1}, X_t)\right)$$

Adding Token Features to a CRF

- Can add token features $X_{i,j}$



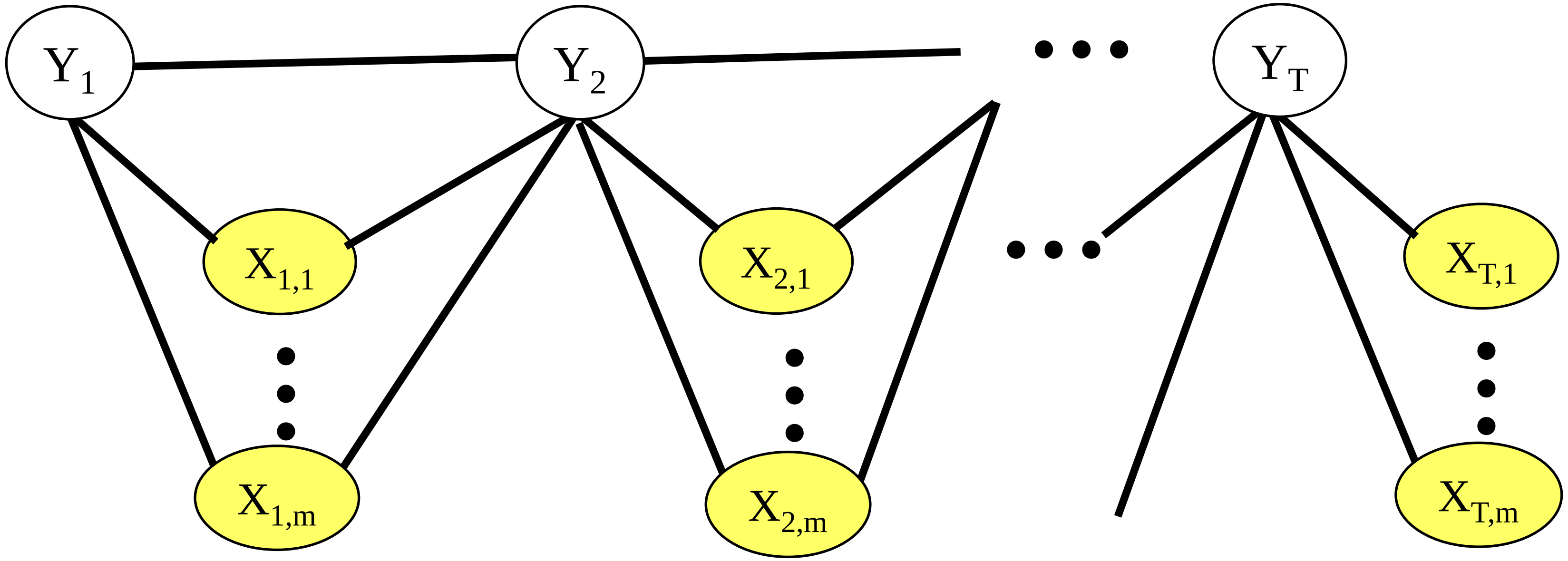
Can add additional feature functions for each token feature to model conditional distribution.

Features in POS Tagging

- For POS Tagging, use lexicographic features of tokens.
  - Capitalized?
  - Start with numeral?
  - Ends in given suffix (e.g. “s”, “ed”, “ly”)?

Enhanced Linear Chain CRF (standard approach)

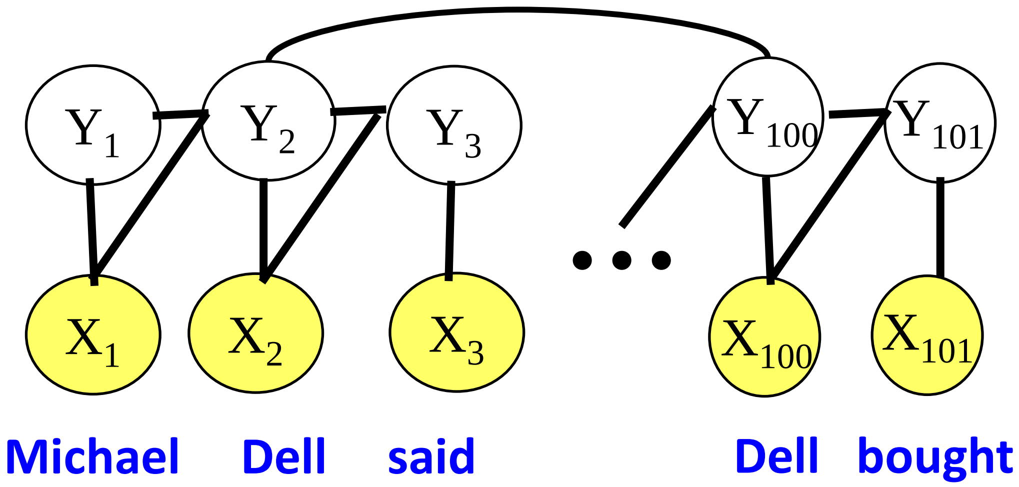
Can also condition transition on the current token features.



$$f_{i,j,k}(Y_t, Y_{t-1}, X) = 1 \text{ if } Y_t = i \text{ and } Y_{t-1} = j \text{ and } X_{t-1,k} = 1$$
$$= 0 \text{ otherwise}$$

Skip-Chain CRFs

Can model some long-distance dependencies (i.e. the same word appearing in different parts of the text) by including long-distance edges in the Markov model.



Additional links make exact inference intractable, so must resort to approximate inference to try to find the most probable labeling.

CRF

- Usually have superior accuracy on various sequence labeling tasks.
  - Part of Speech tagging
  - Noun phrase chunking
  - Named entity recognition
  - Semantic role labeling
- CRFs are much slower to train and do not scale as well to large amounts of training data.
- Skip-chain CRFs improve results on IE.

CRF for NER

Encoding classes for sequence labeling

	IO encoding	IOB encoding
Ram	PER	B-PER
went	O	O
to	O	O
Medica	ORG	B-ORG
Super	ORG	I-ORG
Hospital	ORG	I-ORG
for	O	O
treatment	O	O

Features: Word substrings

Entity Types: Drug Company Movie Place Person

1. Cotrimoxazole
2. Wethersfield
3. Alien Fury: Countdown to Invasion

1. Ajabgar, Ajabpur, Baghberia
2. Ekanjeet, Faiyaz, Meher, Shanaya

Features: Word shapes

- Word Shapes
  - Map words to simplified representation that encodes attributes such as length, capitalization, numerals, Greek letters, internal punctuation, etc.

Varicella-zoster	Xx-xxx
mRNA	xXXX
CPA1	XXXd