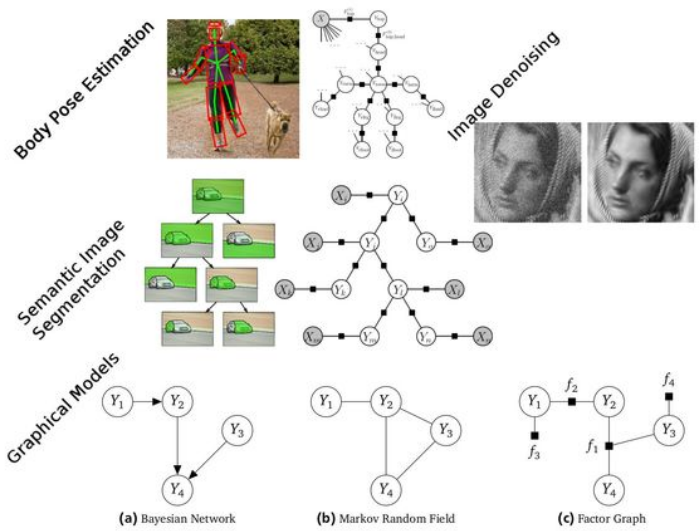
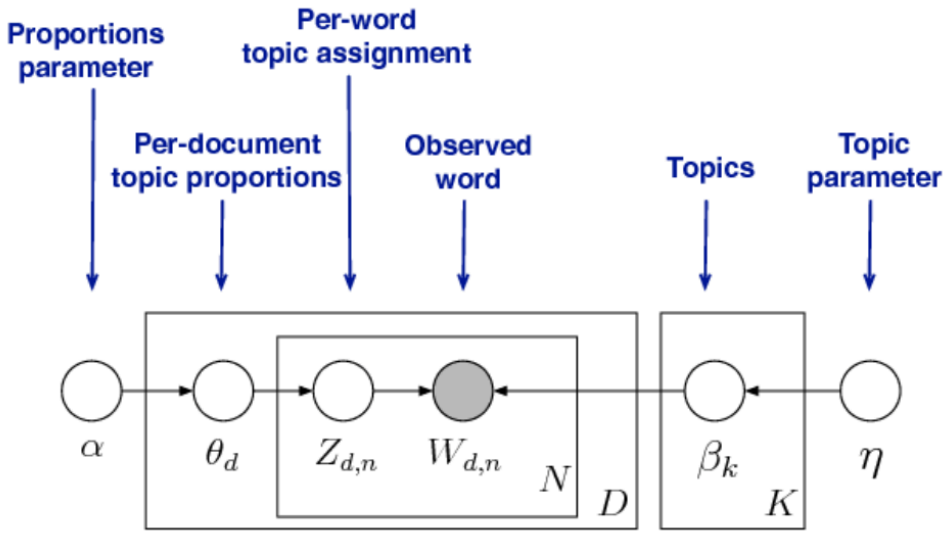
# Probabilistic Graphical Models : Hidden Markov Models and Conditional Random Fields

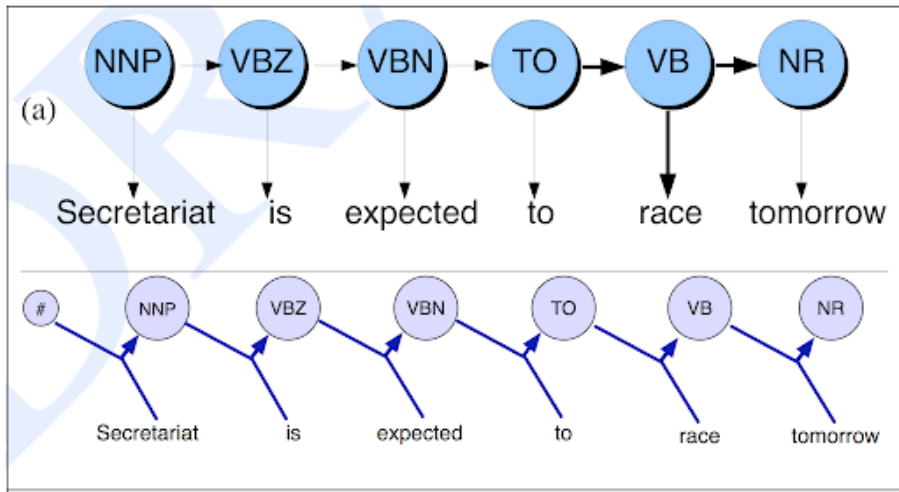
आई आई टी हैदराबाद
IIT Hyderabad

# Probabilistic Graphical Models

- Used in wide range of applications : natural language processing (POS tagging, named entity recognition, automatic speech recognition)
  - They provide a simple way to visualize the structure of a probabilistic model and can be used to design and motivate new models.
  - Insights into the properties of the model, including conditional independence properties, can be obtained by inspection of the graph.

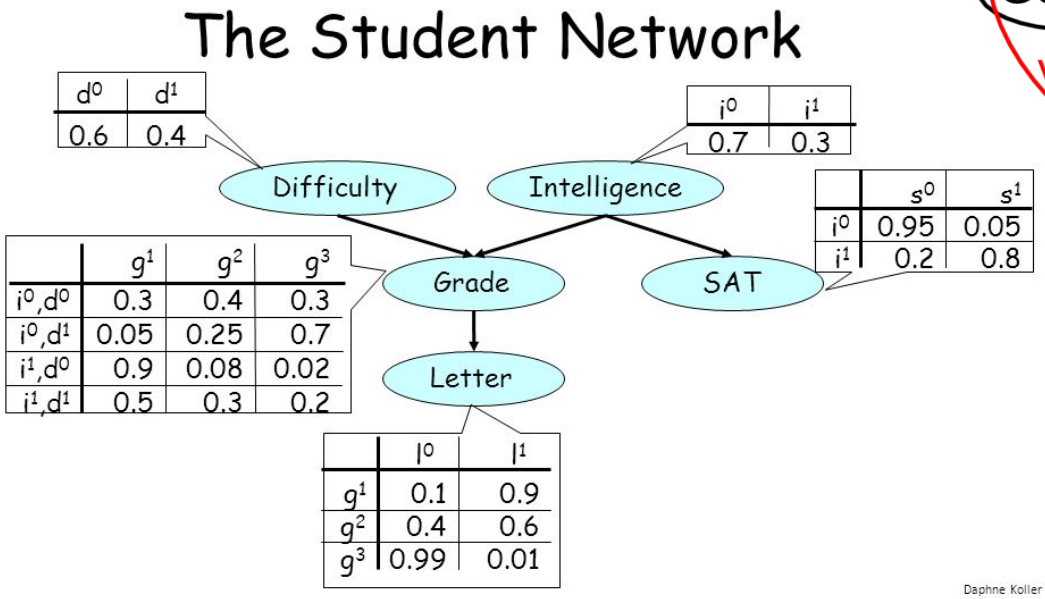


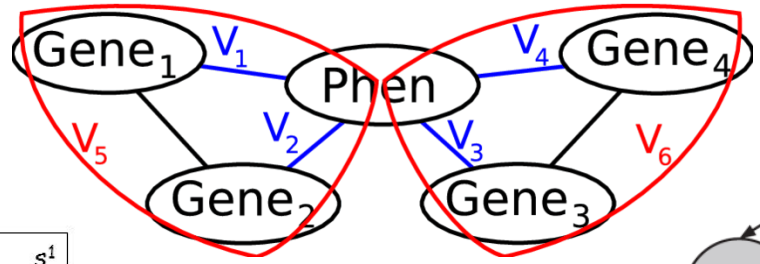



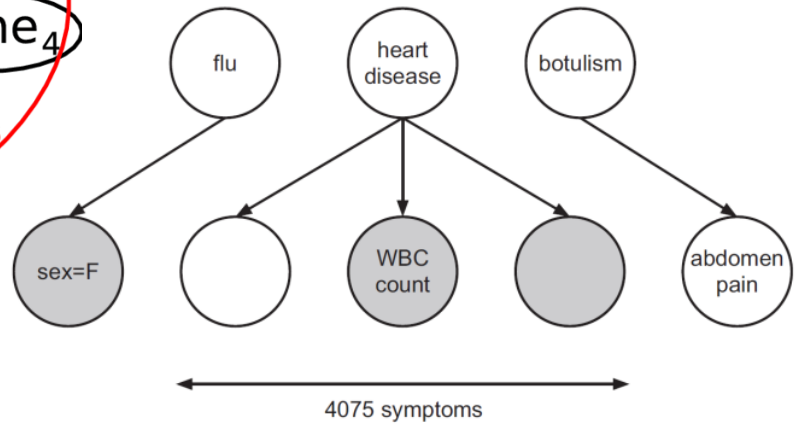



# Probabilistic Graphical Models

- A graph comprises nodes (also called vertices) connected by links (also known as edges or arcs). In a probabilistic graphical model, each node represents a random variable (or group of random variables), and the links express probabilistic relationships between these variables.
- Bayesian networks, also known as directed graphical models
- Markov random fields, also known as undirected graphical models

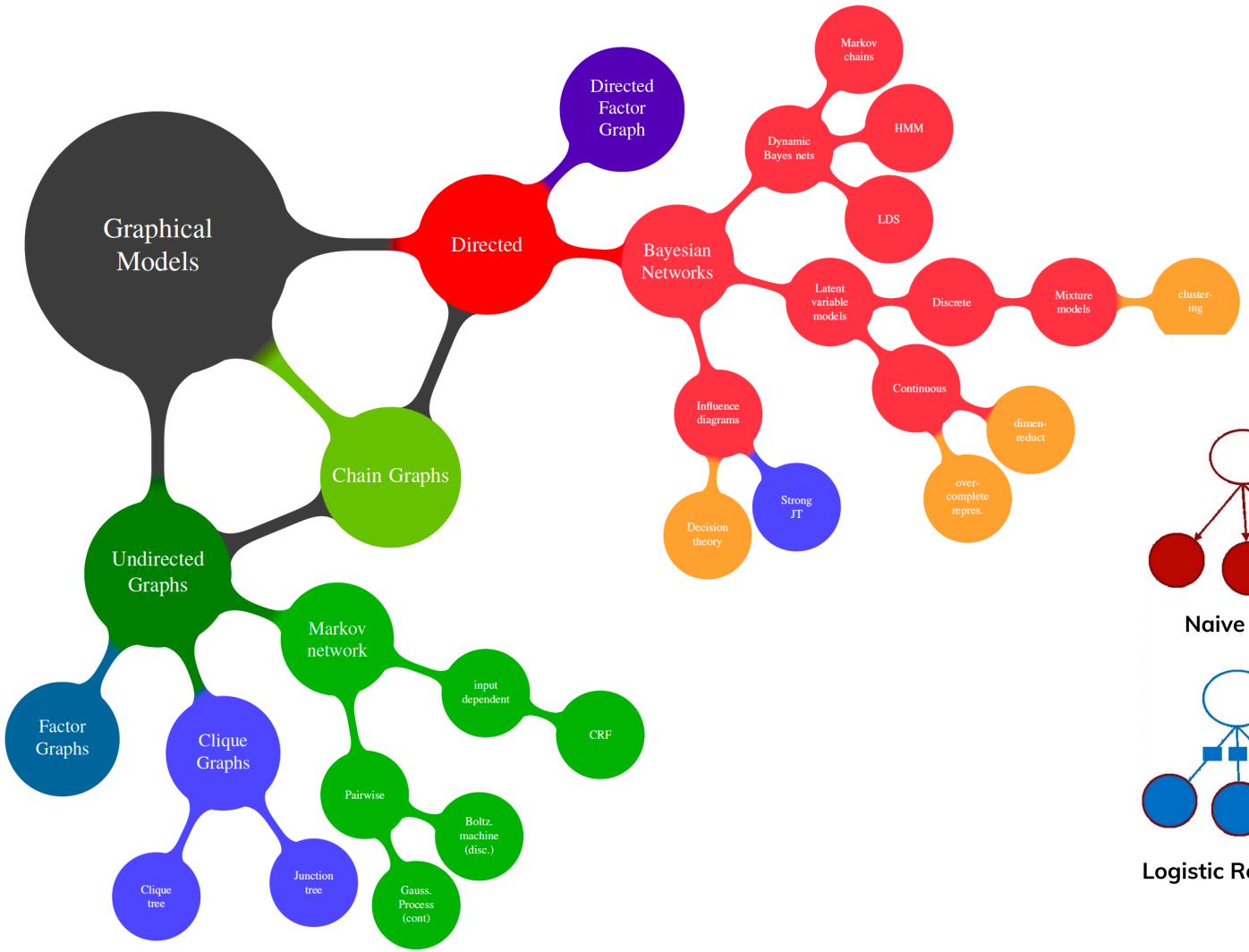


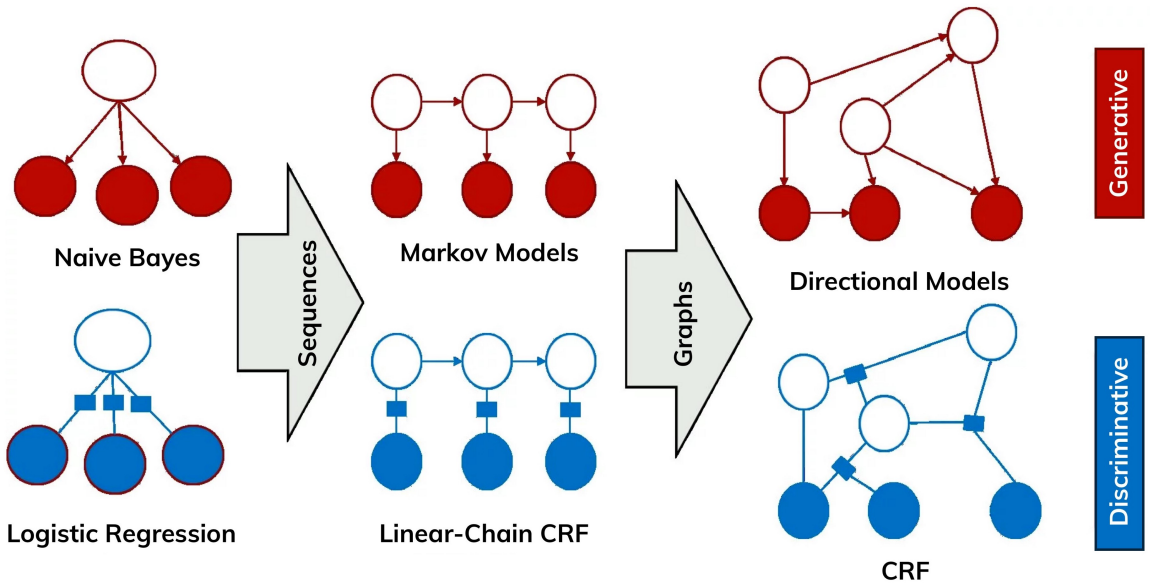




https://www.researchgate.net/publication/47498960_Markov_Logic_Networks_in_the_Analysis_of_Genetic_Data

# Probabilistic Graphical Models



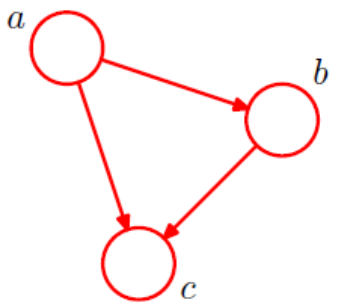




Adapted from C.Sutton, A.McCallum, "An introduction to Conditional Random Fields"

# Probabilistic Graphical Models

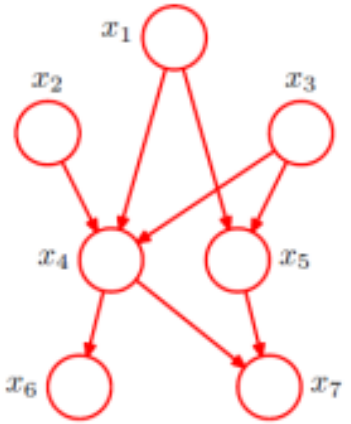
- Independently specifying all the entries of a table p(x1; : : : ; xN) over binary variables xi takes O(2^N) space
- Structure is also important for computational tractability of inferring quantities of interest.
- Given a distribution on N binary variables, p(x1; : : : ; xN), computing a marginal such as p(x1) requires summing over the 2^(N-1) states of the other variables.
- Belief networks (also called Bayes' networks or Bayesian belief networks) are a way to depict the independence
- assumptions made in a distribution

$$p(a, b, c) = p(c|a, b)p(b|a)p(a).$$



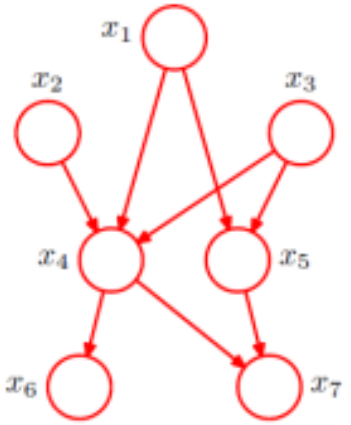
# Probabilistic Graphical Models

- Independently specifying all the entries of a table p(x1; : : : ; xN) over binary variables xi takes O(2^N) space
- Structure is also important for computational tractability of inferring quantities of interest.
- Given a distribution on N binary variables, p(x1; : : : ; xN), computing a marginal such as p(x1) requires summing over the 2^(N-1) states of the other variables.
- Belief networks (also called Bayes' networks or Bayesian belief networks) are a way to depict the independence
- assumptions made in a distribution



# Probabilistic Graphical Models

- Independently specifying all the entries of a table p(x1; …; xN) over binary variables xi takes O(2^N) space
- Structure is also important for computational tractability of inferring quantities of interest.
- Given a distribution on N binary variables, p(x1; … ; xN), computing a marginal such as p(x1) requires summing over the 2^(N-1) states of the other variables.
- Belief networks (also called Bayes' networks or Bayesian belief networks) are a way to depict the independence
- assumptions made in a distribution



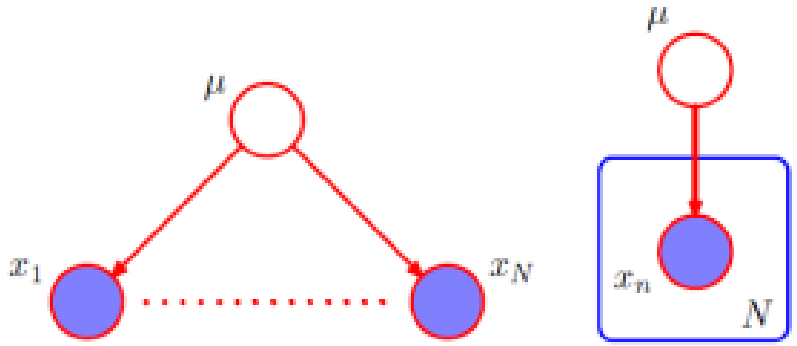

$$p(x_1)p(x_2)p(x_3)p(x_4|x_1,x_2,x_3)p(x_5|x_1,x_3)p(x_6|x_4)p(x_7|x_4,x_5).$$

$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k|\mathrm{pa}_k)$$

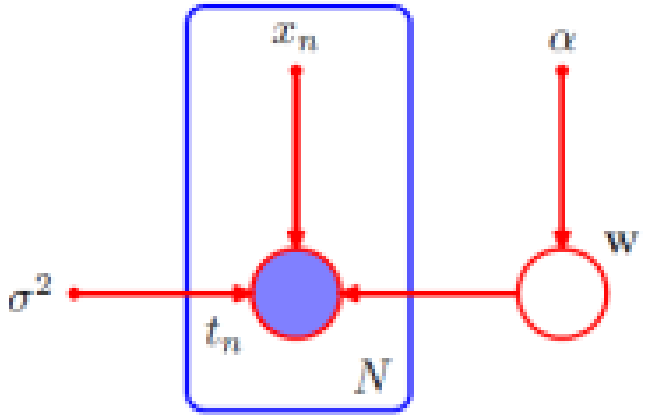
# Probabilistic Graphical Models

- **Plate Notation**
- Bayesian Linear Regression

$$p(\mathcal{D}|\mu) = \prod_{n=1}^{N} p(x_n|\mu).$$




$$p(\mathbf{t}, \mathbf{w}|\mathbf{x}, \alpha, \sigma^2) = p(\mathbf{w}|\alpha) \prod_{n=1}^{N} p(t_n|\mathbf{w}, x_n, \sigma^2).$$



# Probabilistic Graphical Models

- **Conditional Independence**
- *a* is conditionally independent of *b* given *c*.

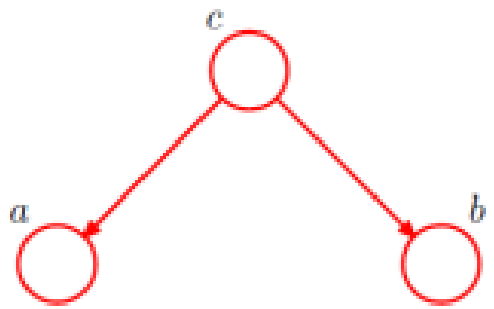
$$a \perp\!\!\!\perp b \mid c \qquad p(a|b,c) = p(a|c).$$

$$\begin{aligned} p(a,b|c) &= p(a|b,c)p(b|c) \\ &= p(a|c)p(b|c). \end{aligned}$$

joint distribution of *a* and *b* factorizes into the product of the marginal distribution of *a* and the marginal distribution of *b* (again both conditioned on *c*).

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

$$p(a,b,c) = p(a|c)p(b|c)p(c).$$

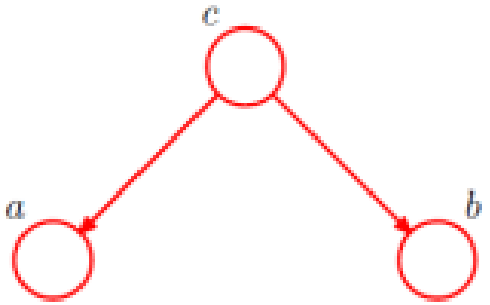


# Probabilistic Graphical Models

- **Conditional Independence**

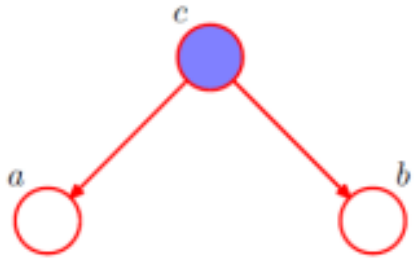
**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

$$p(a, b, c) = p(a|c)p(b|c)p(c).$$




$$p(a, b) = \sum_c p(a|c)p(b|c)p(c). \qquad a \not\perp\!\!\!\perp b \mid \emptyset$$

$$\begin{aligned} p(a, b|c) &= \frac{p(a, b, c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned} \qquad a \perp\!\!\!\perp b \mid c.$$



# Probabilistic Graphical Models

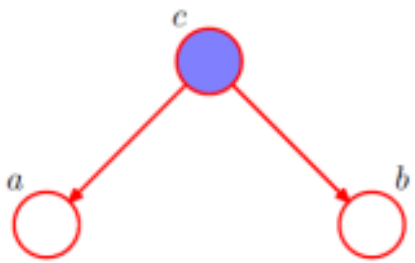
- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

$$p(a,b) = \sum_c p(a|c)p(b|c)p(c). \qquad a \not\perp\!\!\!\perp b \mid \emptyset$$

$$p(a,b|c) = \frac{p(a,b,c)}{p(c)} = p(a|c)p(b|c) \qquad a \perp\!\!\!\perp b \mid c.$$

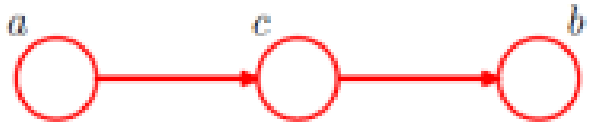
Node $c$ is said to be *tail-to-tail* with respect to this path because the node is connected to the tails of the two arrows, and the presence of such a path connecting nodes $a$ and $b$ causes these nodes to be dependent.



# Probabilistic Graphical Models

- **Conditional Independence**

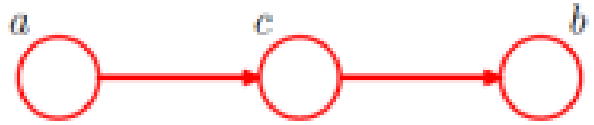
**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph



# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

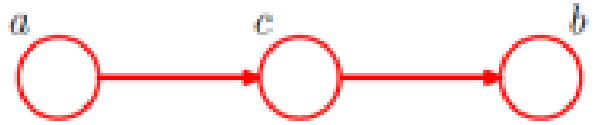



$$p(a, b, c) = p(a)p(c|a)p(b|c).$$

# Probabilistic Graphical Models

- **Conditional Independence**

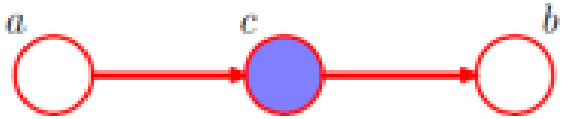
**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph




$$p(a, b, c) = p(a)p(c|a)p(b|c).$$

$$p(a, b) = p(a) \sum_c p(c|a)p(b|c) = p(a)p(b|a).$$

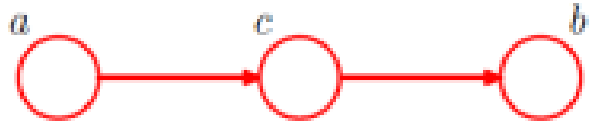
$$a \not\perp b \mid \emptyset$$



# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph



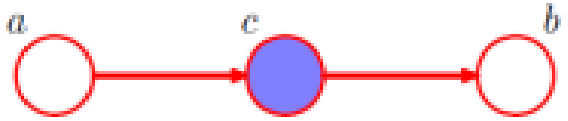

$$p(a, b, c) = p(a)p(c|a)p(b|c).$$

$$p(a, b) = p(a) \sum_c p(c|a)p(b|c) = p(a)p(b|a).$$

$$a \not\perp\!\!\!\perp b \mid \emptyset$$

$$\begin{aligned} p(a, b|c) &= \frac{p(a, b, c)}{p(c)} \\ &= \frac{p(a)p(c|a)p(b|c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned}$$

$$a \perp\!\!\!\perp b \mid c.$$



# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

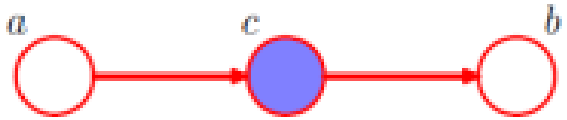
$$p(a,b) = p(a)\sum_c p(c|a)p(b|c) = p(a)p(b|a).$$

$$a \not\perp\!\!\!\perp b \mid \emptyset$$

$$\begin{aligned} p(a,b|c) &= \frac{p(a,b,c)}{p(c)} \\ &= \frac{p(a)p(c|a)p(b|c)}{p(c)} \\ &= p(a|c)p(b|c) \end{aligned}$$

$$a \perp\!\!\!\perp b \mid c.$$

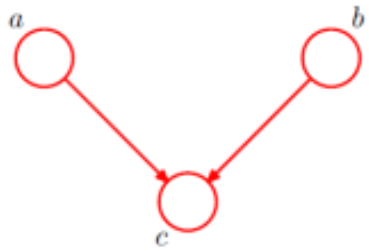
The node *c* is said to be *head-to-tail* with respect to the path from node *a* to node *b*. Such a path connects nodes *a* and *b* and c blocks them renders them dependent.



# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

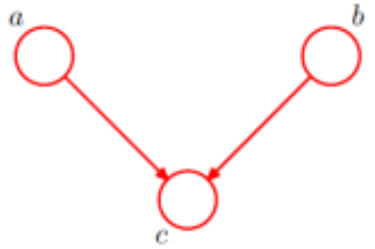


# Probabilistic Graphical Models

- **Conditional Independence**

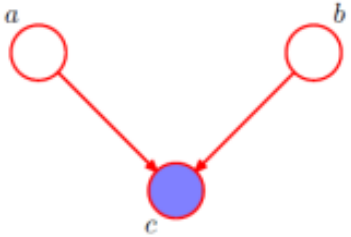
**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

$$p(a, b, c) = p(a)p(b)p(c|a, b).$$




$$p(a, b) = p(a)p(b)$$

$$a \perp\!\!\!\perp b \mid \emptyset.$$

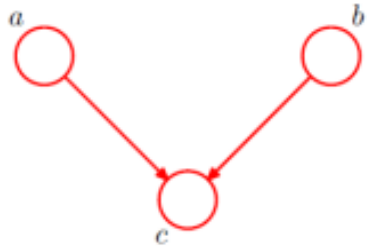


# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

$$p(a, b, c) = p(a)p(b)p(c|a, b).$$

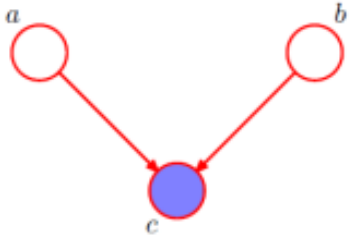



$$p(a, b) = p(a)p(b)$$

$$a \perp\!\!\!\perp b \mid \emptyset.$$

$$p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(b)p(c|a, b)}{p(c)}$$

$$a \not\perp\!\!\!\perp b \mid c.$$



# Probabilistic Graphical Models

- **Conditional Independence**

**D-separation** : An important and elegant feature of graphical models is that conditional independence properties of the joint distribution can be read directly from the graph

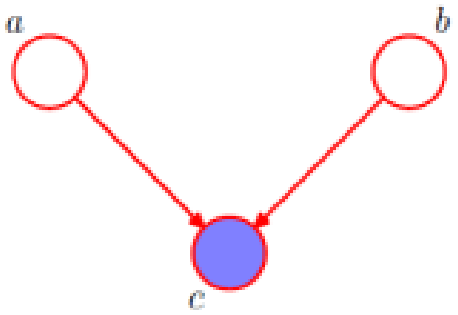
node *c* is *head-to-head* with respect to the path from *a* to *b* because it connects to the heads of the two arrows. When node *c* is unobserved, it 'blocks' the path, and the variables *a* and *b* are independent. However, conditioning on *c* 'unblocks' the path and renders *a* and *b* dependent.

$$p(a, b) = p(a)p(b)$$

$$a \perp\!\!\!\perp b \mid \emptyset.$$

$$p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(b)p(c|a, b)}{p(c)}$$

$$a \not\perp\!\!\!\perp b \mid c.$$




explaining away'.

# Probabilistic Graphical Models

- Inference in Graphical Models
  - Message passing algorithms
  - Sum product algorithm
  - Max-sum algorithm

$$p(x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} p(\mathbf{x}).$$

$$p(\mathbf{x}) = \prod_{s \in \mathrm{ne}(x)} F_s(x, X_s)$$

$$\mathbf{x}^{\max} = \arg\max_{\mathbf{x}} p(\mathbf{x})$$

$$\max_{\mathbf{x}} p(\mathbf{x}) = \max_{x_1} \ldots \max_{x_M} p(\mathbf{x})$$

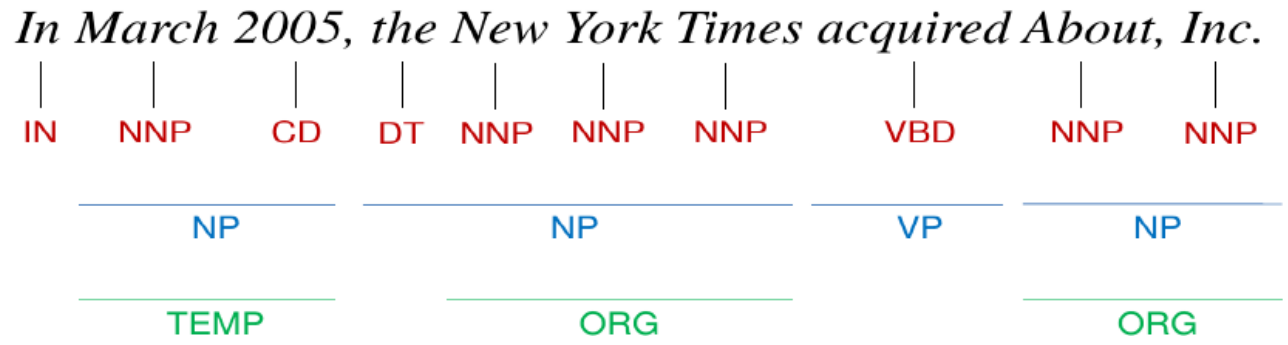
$$p(x) = \prod_{s \in \mathrm{ne}(x)} \left[ \sum_{X_s} F_s(x, X_s) \right]$$

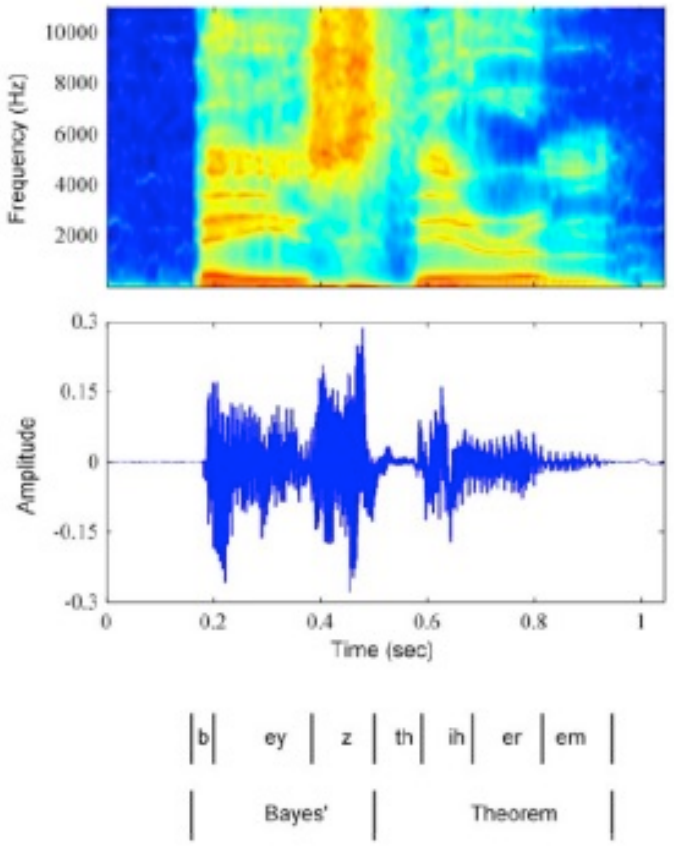
$$\max_{\mathbf{x}} p(\mathbf{x}) = \frac{1}{Z} \max_{x_1} \cdots \max_{x_N} [\psi_{1,2}(x_1, x_2) \cdots \psi_{N-1,N}(x_{N-1}, x_N)]$$

$$= \frac{1}{Z} \max_{x_1} \left[ \psi_{1,2}(x_1, x_2) \left[ \cdots \max_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \right] \right].$$

# Sequential Data

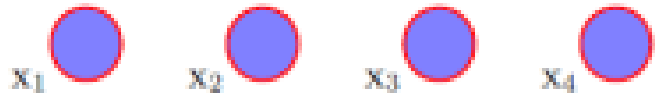
- Sequential data : rainfall measurements on successive days at a particular location, or the daily values of a currency exchange rate (time series data) , sequence of nucleotide base pairs along a strand of DNA or the sequence of characters in an English sentence



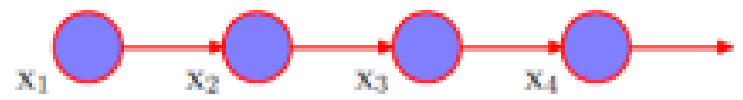
# Markov Model

- binary variable denoting whether on a particular day it rained or not.




$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}).$$

- Markov Model (first Order) : conditional distributions on the right-hand side is independent of all previous observations except the most recent

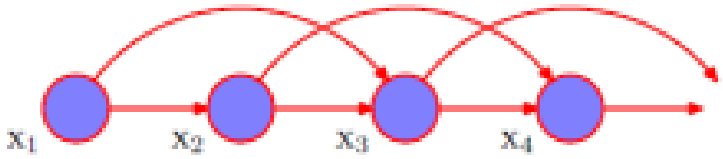



$K(K-1)$ parameters.

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) \prod_{n=2}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}).$$

$$p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) = p(\mathbf{x}_n | \mathbf{x}_{n-1})$$

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) p(\mathbf{x}_2 | \mathbf{x}_1) \prod_{n=3}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}, \mathbf{x}_{n-2}).$$



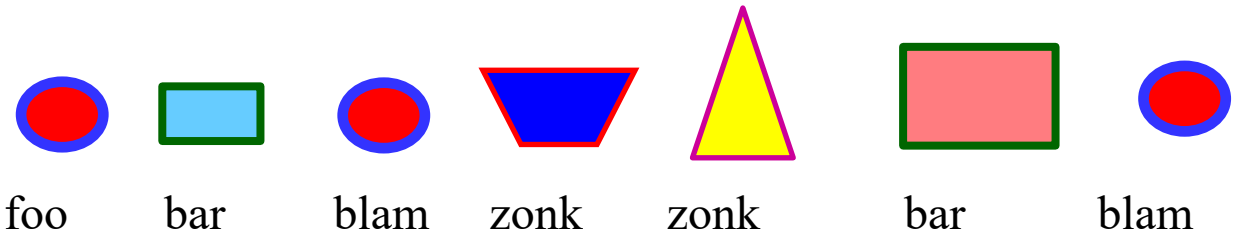

$M^{\text{th}}$ order Markov chain,
$p(\mathbf{x}_n | \mathbf{x}_{n-M}, \ldots, \mathbf{x}_{n-1})$.

$K^M(K-1)$ parameters.

# Hidden Markov Model

- HMM is widely used in speech. recognition (Jelinek, 1997; Rabiner and Juang, 1993), natural language modelling (Manning and Sch¨utze, 1999), on-line handwriting recognition (Nag et al., 1986), and for the analysis of biological sequences such as proteins and DNA

- Standard classification problem assumes individual cases are disconnected and independent (i.i.d.: independently and identically distributed).
- Each token in a sequence is assigned a label. Labels of tokens are dependent on the labels of other tokens in the sequence, particularly their neighbors (not i.i.d).



- A given sentence, *"Time flies like an arrow"*
- Represent the input sentence with a token vector $\boldsymbol{x}$

| $t$ | 1 | 2 | 3 | 4 | 5 | |
|---|---|---|---|---|---|---|
| $\boldsymbol{x}$ | $Time$ | $flies$ | $like$ | $an$ | $arrow$ | $(\mathrm{T} = 5)$ |
| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | |

(Bold italic)
(*NOTE: This does not present a feature vector*)

- Predict part-of-speech (a vector $\boldsymbol{y}$) tags for the tokens $\boldsymbol{x}$

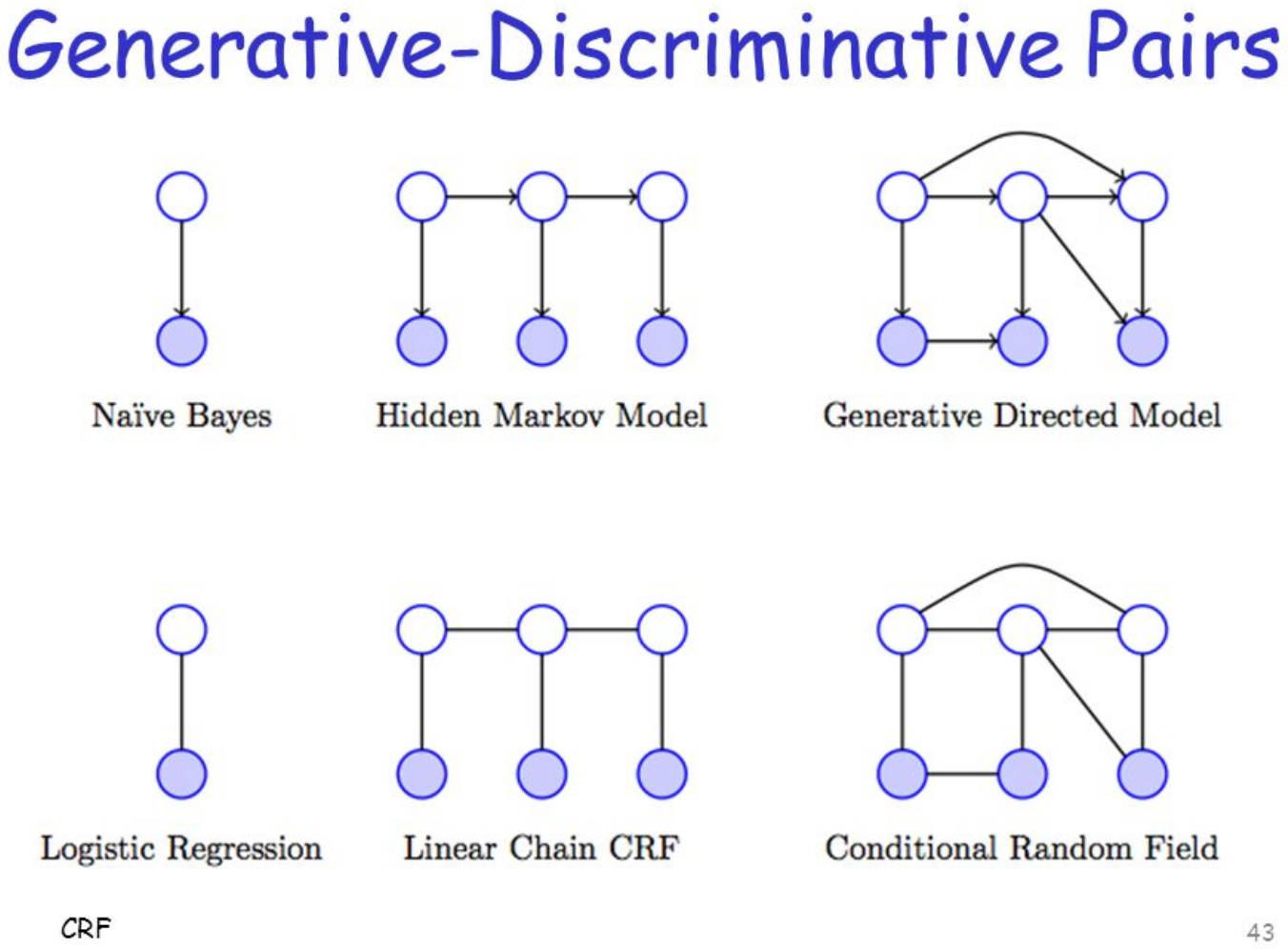
| $t$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $\boldsymbol{x}$ | $Time$ | $flies$ | $like$ | $an$ | $arrow$ |
| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
| $\boldsymbol{y}$ | $NN$ | $VBZ$ | $IN$ | $DT$ | $NN$ |
| | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ |


Predict

- *Modeling*: how to build (assume) $P(\boldsymbol{y}|\boldsymbol{x})$
  - Hidden Markov Model (HMM), Structured Perceptron, Conditional Random Fields (CRFs), etc
- *Training*: how to determine unknown parameters in the model so that they fit to a training data
  - Maximum Likelihood (ML), Maximum a Posteriori (MAP), etc
  - Gradient-based method, Stochastic Gradient Descent (SGD), etc
- *Inference*: how to compute $\mathrm{argmax}\, P(\boldsymbol{y}|\boldsymbol{x})$ efficiently
  - Viterbi algorithm

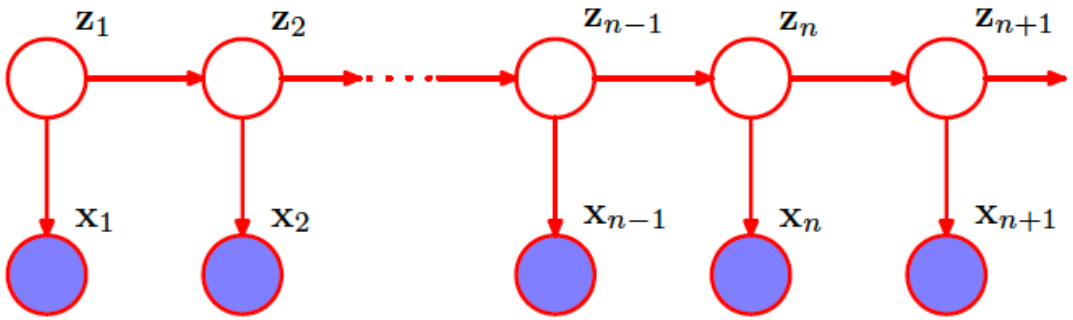
# Probabilistic Sequence Models

- Probabilistic sequence models allow integrating uncertainty over multiple, interdependent classifications and collectively determine the most likely global assignment.
- Two standard models
  - Generative Model : Hidden Markov Model (HMM)
  - Discriminative Model : Conditional Random Field (CRF)



# Hidden Markov Model

- Probabilistic generative model for sequences.
- Assume an underlying set of ***hidden*** (unobserved, latent ) states (**z**) in which the model can be in.
- Assume probabilistic transitions between states (parts of speech) over time as sequence is generated.
- Assume a ***probabilistic*** generation of tokens from states (e.g. words generated for each POS).




$$P(x_1, x_2 \cdots x_{n+1}, z_1, z_2 \cdots z_{n+1})$$
$$= P(z_1)\, P(x_1|z_1)\, P(z_2|z_1) \cdots P(z_{n+1}|z_n)\, P(x_{n+1}|z_{n+1})$$
$$= \prod_t P(x_t|z_t)\, P(z_{t+1}|z_t)$$

# Hidden Markov Model

- $\boldsymbol{x}$: the sequence of tokens (words)
- $\boldsymbol{y}$: the sequence of POS tags

- Bayes' theorem:

$$P(\boldsymbol{y}|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|\boldsymbol{y})P(\boldsymbol{y})}{P(\boldsymbol{x})}$$

- Bayesian inference: decompose $P(\boldsymbol{y}|\boldsymbol{x})$ into two factors, $P(\boldsymbol{x}|\boldsymbol{y})$ and $P(\boldsymbol{y})$, which might be easier to model

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, P(\boldsymbol{y}|\boldsymbol{x}) = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, \frac{P(\boldsymbol{x}|\boldsymbol{y})P(\boldsymbol{y})}{P(\boldsymbol{x})} = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, P(\boldsymbol{x}|\boldsymbol{y})P(\boldsymbol{y})$$

Bayes' theorem

$P(\boldsymbol{x})$ is the same for all $\boldsymbol{y}$

# HMM

- Two Markov assumptions to simplify $P(\boldsymbol{x}|\boldsymbol{y})$ and $P(\boldsymbol{y})$
  - A word appears depending only on its POS tag
    - Independently of other words around the word
    - Generated by emission probability distribution

$$P(\boldsymbol{x}|\boldsymbol{y}) \approx \prod_{t=1}^{T} P(x_t|y_t)$$

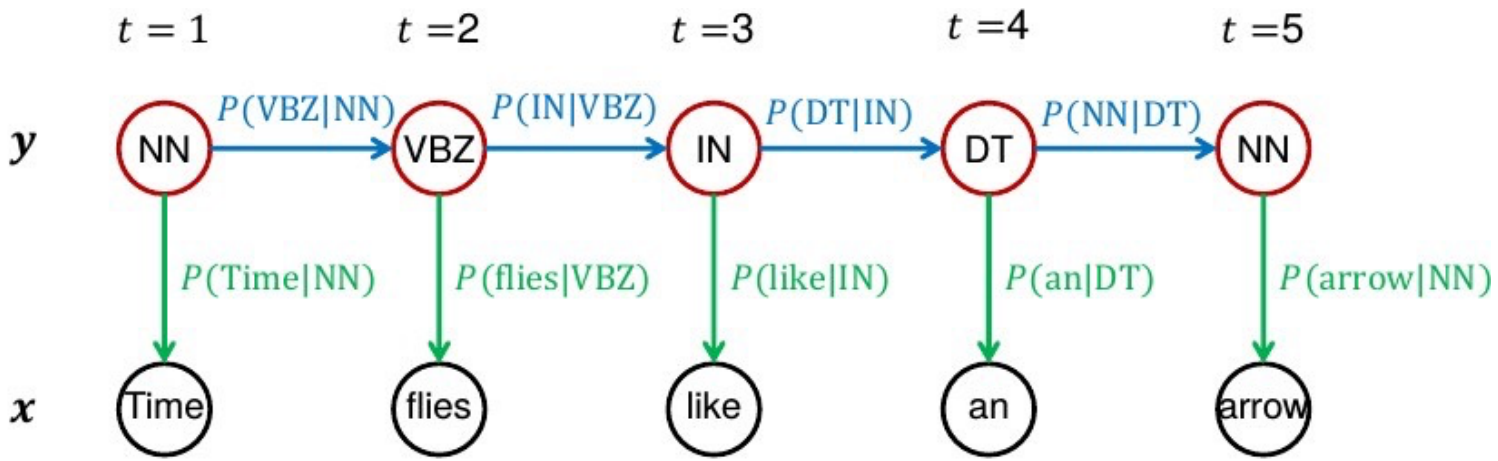
  - A POS tag is dependent only on the previous one
    - Rather than the entire tag sequence
    - Generated by transition probability distribution

$$P(\boldsymbol{y}) \approx \prod_{t=1}^{T} P(y_t|y_{t-1})$$

- Then, the most probable tag sequence $\hat{\boldsymbol{y}}$ is computed by,

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, P(\boldsymbol{y}|\boldsymbol{x}) = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, P(\boldsymbol{x}|\boldsymbol{y})P(\boldsymbol{y}) \approx \underset{\boldsymbol{y}}{\operatorname{argmax}} \prod_{t=1}^{T} P(x_t|y_t)P(y_t|y_{t-1})$$

# POS Tagging




- We can compute $\phi(\boldsymbol{x}, \boldsymbol{y})$ if we decide an assignment of $\boldsymbol{y}$ for a given input $\boldsymbol{x}$: $\prod_{t=1}^{T} P(x_t|y_t)P(y_t|y_{t-1})$

$$P(x_t|y_t) = \frac{C(x_t, y_t)}{C(y_t)} = \frac{(\text{the number of times where } x_t \text{ is annotated as } y_t)}{(\text{the number of occurrences of tag } y_t)}$$

$$P(y_t|y_{t-1}) = \frac{C(y_t, y_{t-1})}{C(y_{t-1})} = \frac{(\text{the number of occurrences of tag } y_t \text{ followd by } y_{t-1})}{(\text{the number of occurrences of tag } y_{t-1})}$$

# Conditional Random Field

- Conditional probability is defined,

$$P(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp((\boldsymbol{w} \cdot \boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}))}{\sum_{\boldsymbol{y}} \exp((\boldsymbol{w} \cdot \boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}))}$$

Normalized by the sum of exp'd scores of all possible paths in the lattice

- The same inference algorithm (Viterbi)
- Input: sequence of tokens $\boldsymbol{x} = (x_1\ x_2\ \ldots\ x_T)$
- Output: sequence of POS tags $\hat{\boldsymbol{y}} = (\widehat{y_1}\ \widehat{y_2}\ \ldots\ \widehat{y_T})$
- Mapping to global feature vector: $\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y})$: $(\boldsymbol{x}, \boldsymbol{y}) \rightarrow \mathcal{R}^m$

$$\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{t=1}^{T} \{\boldsymbol{u}(x_t, y_t) + \boldsymbol{b}(y_{t-1}, y_t)\}$$

Local feature vector (at $t$):
- Unigram feature vector
- Bigram feature vector

  - Each element of feature vector consists of a feature function, e.g.,
    - $u_{109}(x_t, y_t) = \{1\ (\text{if } x_t = \text{Brown } and\ y_t = \text{Noun}); 0\ (\text{otherwise})\}$
    - $b_2(y_{t-1}, y_t) = \{1\ (\text{if } y_{t-1} = \text{Noun } and\ y_t = \text{Verb}); 0\ (\text{otherwise})\}$

# Probabilistic Graphical Models

- A graph comprises nodes (also called vertices) connected by links (also known as edges or arcs). In a probabilistic graphical model, each node represents a random variable (or group of random variables), and the links express probabilistic relationships between these variables.
- Bayesian networks, also known as directed graphical models (e.g HMM)
- Markov random fields, also known as undirected graphical models (e.g CRF)