第6章 马尔科夫模型与条件随机场

6.1 马尔可夫模型

6.1 马尔可夫模型

◆马尔可夫模型描述

马尔科夫过程：如果一个系统有 N 个状态 $S_1, S_2, \ldots, S_N$, 随着时间的推移，该系统从某一状态转移到另一状态。

如果用q_t表示系统在 t时刻的状态(取值为 S_j)($1\le j\le N$) ,则其概率取决于前 t-1 个时刻 (1, 2, …, t-1) 的状态，该概率为：

$$p(q_t = S_j \mid q_{t-1} = S_i, q_{t-2} = S_k, \cdots)$$

简单说：马尔科夫过程就是指过程中的每个状态的转移只依赖于之前的 n个状态。

6.1 马尔可夫模型

为控制复杂性，我们对其进行简化。

●假设1：

如果在特定情况下，系统在时间 t 的状态只与时间 t-1 的状态相关，则该系统构成一个离散的一阶马尔可夫链：

$$p(q_t = S_j \mid q_{t-1} = S_i, q_{t-2} = S_k, \cdots) = \boxed{p(q_t = S_j \mid q_{t-1} = S_i)} \quad \ldots (6.1)$$

6.1 马尔可夫模型

● 假设2：

如果只考虑公式(6.1)独立于时间 t 的随机过程，即所谓的不动性假设，状态与时间无关，那么：

$$p(q_t = S_j \mid q_{t-1} = S_i) = a_{ij}, \quad 1 \le i, j \le N \qquad \ldots (6.2)$$

$P(S_j|S_i)=a_{ij}$

该随机过程称为(一阶)马尔可夫模型(**Markov Model)**，**或者马尔科夫链**。

6.1 马尔可夫模型

在马尔可夫模型中，$\boxed{\text{状态转移概率 } a_{ij}}$必须满足下列条件：

$$a_{ij} \geq 0 \qquad \ldots (6.3)$$

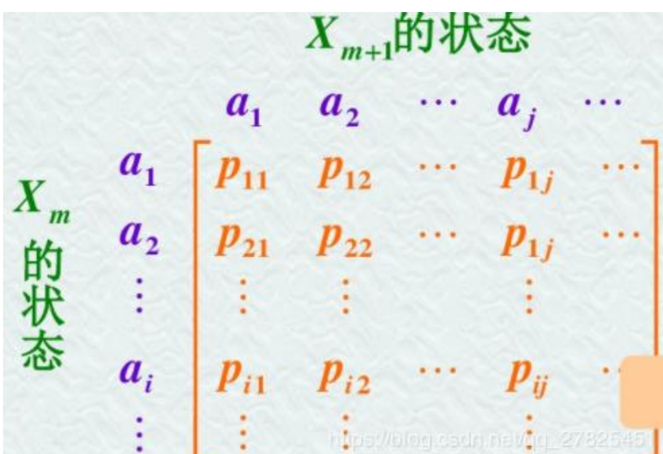
$$\sum_{j=1}^{N} a_{ij} = 1 \qquad \ldots (6.4)$$

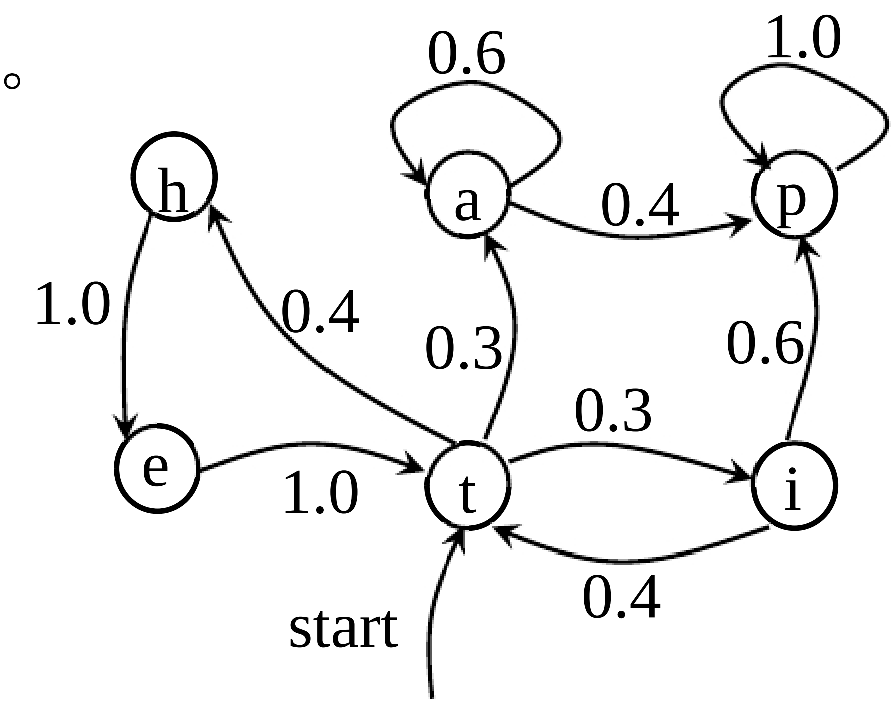
6.1 马尔可夫模型

◆马尔可夫模型可以表示成状态图（转移弧上有概率的非确定的有穷状态自动机）

— 零概率的转移弧省略。

— 每个节点上所有发出弧的概率之和等于1。





6.1 马尔可夫模型

状态序列 $S_1, \ldots, S_T$ 的概率：

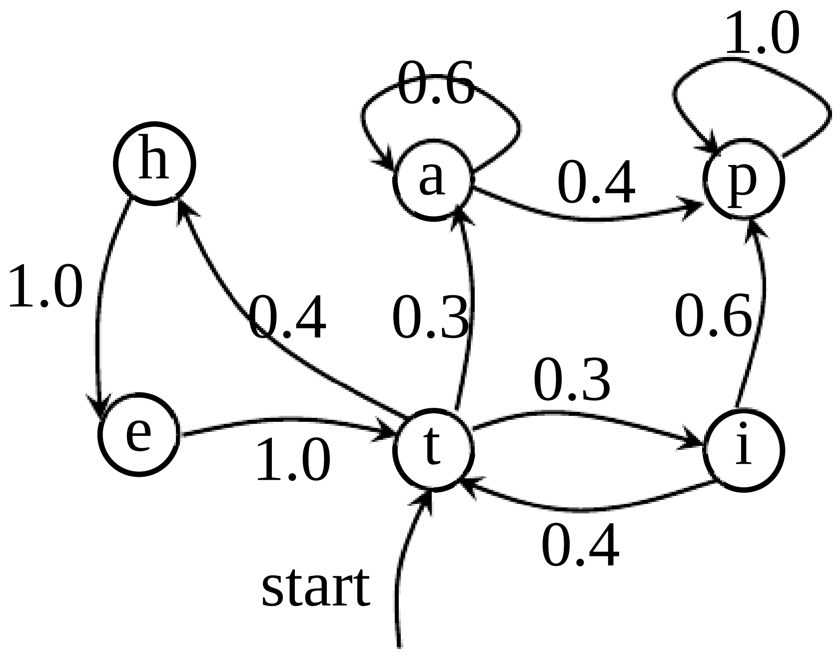
$$p(S_1,\cdots,S_T) = p(S_1)\times p(S_2 \mid S_1)\times p(S_3 \mid S_1,S_2)\times\cdots\times p(S_T \mid S_1,\cdots,S_{T-1})$$

$$= p(S_1)\times p(S_2 \mid S_1)\times p(S_3 \mid S_2)\times\cdots\times p(S_T \mid S_{T-1})$$

$$= \pi_{S_1}\prod_{t=1}^{T-1} a_{S_t S_{t+1}} \qquad \ldots (6.5)$$

其中，$\pi_i = p(q_1 = S_i)$，为初始状态的概率。

6.1 马尔可夫模型



$$p(\mathrm{t, i, p}) = p(S_1 = \mathrm{t}) \times p(S_2 = \mathrm{i} \mid S_1 = \mathrm{t}) \times p(S_3 = \mathrm{p} \mid S_2 = \mathrm{i})$$

$$= 1.0 \times 0.3 \times 0.6$$

$$= 0.18$$

6.2 隐马尔可夫模型

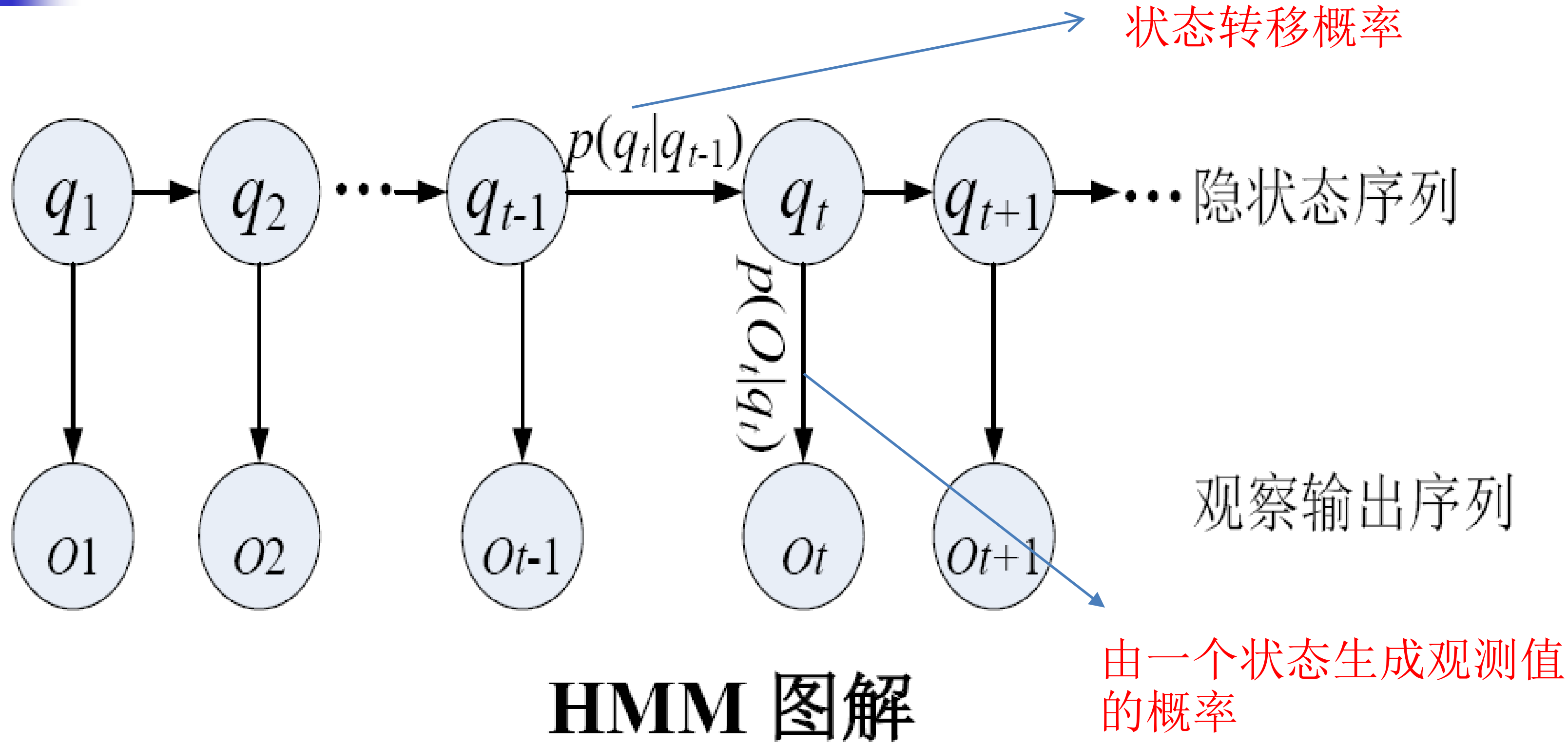
6.2 隐马尔可夫模型

◆隐马尔可夫模型 (**Hidden Markov Model, HMM**)

马尔可夫模型中假定状态值(如天气)是可以直接观察到的。但很多情况下，状态无法直接观察到，但可以通过其它观测值推测得到, 如土壤干燥(观测值)可以推测出天气晴朗(隐状态)。

描述：该模型是一个双重随机过程，其中状态转移是不可观测的（马尔可夫过程），而可观测序列是由隐藏的状态序列以一定的概率随机生成（随机过程）。

6.2 隐马尔可夫模型



HMM 图解

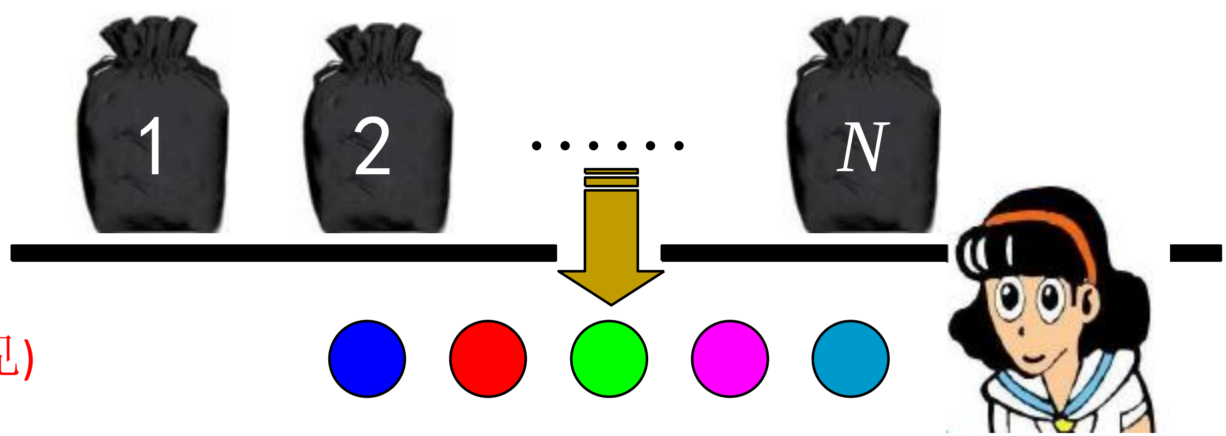
注意：马尔科夫模型和隐马尔科夫模型都是有向图；
除满足马尔科夫过程要求，观测值满足独立假设

6.2 隐马尔可夫模型

例如：N 个袋子，每个袋子中有 M 种不同颜色的球。一实验员根据某一概率分布选择一个袋子(对应HMM中的一个状态)，然后根据袋子中不同颜色球的概率分布随机取出一个球，并报告该球的颜色（球的颜色对应于 HMM 中的观察输出）。

对局外人：可观察的过程是不同颜色球的序列，而袋子的序列是不可观察的。

隐状态序列



观测序列(可见)

6.2 隐马尔可夫模型

◆**HMM** 的组成

1. 模型中的状态数为 N **(**袋子的数量**)**

2. 从每一个状态可能输出的不同的符号数为M **(**不同颜色球的数目**)**

6.2 隐马尔可夫模型

3. 状态转移概率矩阵 $A=a_{ij}$，a_{ij}为实验员从一只袋子(状态 S_i) 转向另一只袋子(状态 S_j) 取球的概率。其中,

$$\begin{cases} a_{ij} = p(q_{t+1} = S_j \mid q_t = S_i), \quad 1 \le i, j \le N \\ a_{ij} \ge 0 \\ \sum_{j=1}^{N} a_{ij} = 1 \end{cases} \qquad \ldots (6.6)$$

6.2 隐马尔可夫模型

4. 从状态 S_j 观察到某一特定符号 v_k的概率分布矩阵为：

$B=b_j(k)$

其中，$b_j(k)$为 实验员从第 j 个袋子中取出第 k 种颜色的球的概率,也称发射概率。那么，

$$
\begin{cases}
b_j(k)=p(O_t=v_k \mid q_t=S_j), \quad 1\le j \le N, \quad 1\le k \le M \\
b_j(k) \ge 0 \\
\sum_{k=1}^{M} b_j(k)=1
\end{cases}
\qquad \ldots (6.7)
$$

6.2 隐马尔可夫模型

5. 初始状态的概率分布为：$\pi = \pi_i$，其中，

$$\begin{cases} \pi_i = p(q_1 = S_i), \quad 1 \le i \le N \\ \pi_i \ge 0 \\ \sum_{i=1}^{N} \pi_i = 1 \end{cases} \qquad \dots (6.8)$$

转移概率　发射概率　初始状态

为了方便，一般将 HMM 记为：$\mu = (A, B, \pi)$

或者 $\mu = (S, O, A, B, \pi)$ 用以指出模型的参数集合。

6.2 隐马尔可夫模型

◆给定HMM求观察序列 -序列生成问题

给定模型 $\mu = (A, B, \pi)$, $\boxed{\text{产生观察序列 } O = O_1 O_2 \ldots O_T;}$

(1) 令 $t = 1$;

(2) 根据初始状态分布 $\pi = \pi_i$ 选择初始状态 $q_1 = S_i$;

(3) 根据状态 S_i 的输出概率分布 $b_i(k)$, 输出 $O_t = v_k$;

(4) 根据状态转移概率 a_{ij}, 转移到新状态 $q_{t+1} = S_j$;

(5) $t = t+1$, 如果 $t < T$, 重复步骤 (3) (4), 否则结束。

6.2 隐马尔可夫模型

◆三个问题：

(1) 在给定模型$\mu=(A, B, \pi)$ 和观察序列 $O＝O_1O_2 \ldots O_T$ 的情况下，怎样快速计算概率 $p(O|\mu)$？

如丢硬币测试(假定3个硬币各不相同, 其序号为隐藏状态)，上述问题对应：

给定HMM模型，观察结果（硬币的正反面）为O={H，T，H}的概率是多少？

6.2 隐马尔可夫模型

◆三个问题：

(2) 在给定模型 $\mu=(A, B, \pi)$ 和观察序列 $O=O_1O_2 \ldots O_T$ 的情况下，如何选择在一定意义下“最优”的状态序列 $Q = q_1q_2\ldots q_T$，使得该状态序列“最好地解释”观察序列？

如对于丢硬币测试(假定其序号为隐藏状态)，上述问题对应：

若给定观察结果O={H，T，H}，那么最可能的状态序列（硬币序号）是什么？

6.2 隐马尔可夫模型

◆三个问题：

(3) 给定一个观察序列 $O = O_1O_2 \ldots O_T$，如何根据最大似然估计来求模型的参数值？即如何调节模型的参数，使得$p(O|\mu)$ 最大？

如对于丢硬币(假定每个硬币均不相同)测试，上述问题对应：

如何根据观察结果 O， 得到模型未知的参数A、B、π？

6.3 前向算法

6.3 前向算法

◆ 求解问题1：

给定模型 $\mu=(A, B, \pi)$ 和观察序列$O=O_1O_2\ldots O_T$，快速计算 $p(O|\mu)$：

对于给定的状态序列 $Q=q_1q_2\ldots q_T$, $p(O|\mu)=?$

$$p(O|\mu)=\sum_{Q}p(O,Q|\mu)=\sum_{Q}\boxed{p(Q|\mu)}\times\boxed{p(O|Q,\mu)} \quad \ldots (6.9)$$

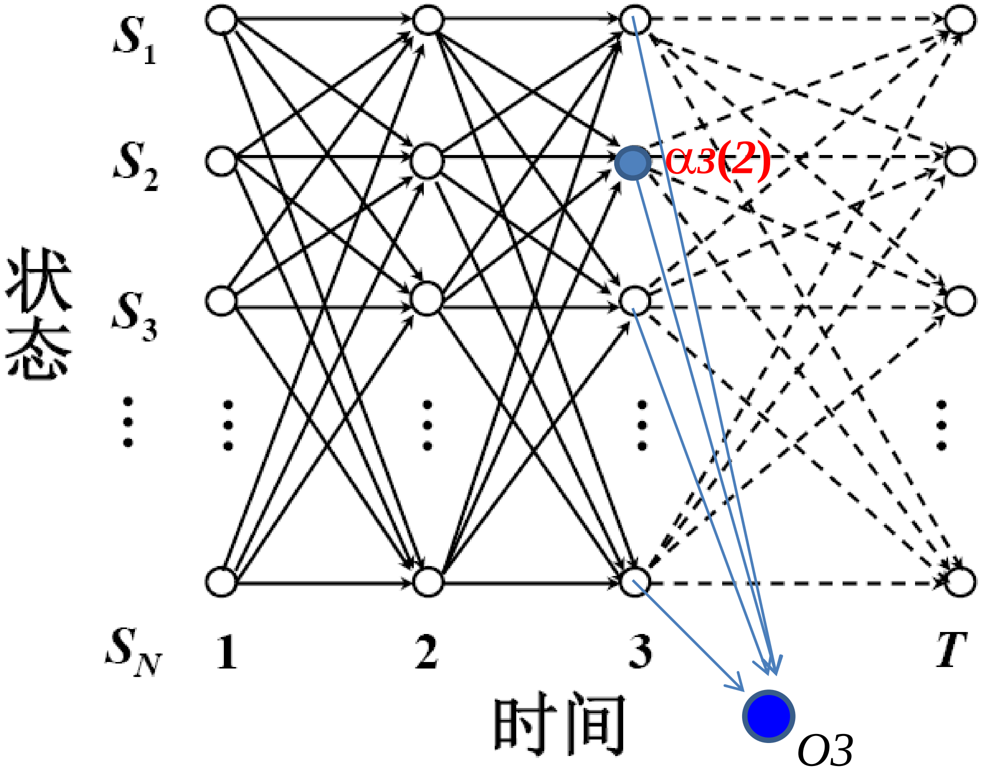
$$p(Q|\mu)=\pi_{q_1}\times a_{q_1q_2}\times a_{q_2q_3}\times\cdots\times a_{q_{t-1}q_T}$$ 转移概率

$$p(O|Q,\mu)=b_{q_1}(O_1)\times b_{q_2}(O_2)\times\cdots\times b_{q_T}(O_T)$$ 发射概率

6.3 前向算法

对每个O_i，需要考虑所有可能路径下的概率 累加

累加：进入状态q_j的概率 乘以 由q_j转到Oi的概率



- **困难：**
如果模型μ有 N 个不同的状态，时间长度为 T，那么有 N^T 个可能的状态序列，搜索路径成指数级组合爆炸。

6.3 前向算法

● 解决办法：动态规划

前向算法(The forward procedure)

● 基本思想：定义前向变量（前向概率）$\alpha_t(i)$：

$$\alpha_t(i) = p(O_1 O_2 \cdots O_t, q_t = S_i \mid \mu) \qquad \ldots(6.12)$$

$\alpha_t(i)$：t时刻时，状态为S_i且观测到序列$O_1, O_2, \ldots, O_t$的概率

前向概率值= “截止t时刻, 进入到状态i 的概率(考虑所有可能状态路径) x 该状态下的发射概率”

6.3 前向算法

因为 $p(O|\mu)$ 是在到达状态 q_T 时观察到序列 $O = O_1 O_2 \dots O_T$ 的概率(所有可能的概率之和):

N为隐状态总数

$$p(O|\mu) = \sum_{S_i} p(O_1 O_2 \cdots O_T, q_T = S_i | \mu) = \sum_{i=1}^{N} \alpha_T(i) \qquad \dots (6.13)$$

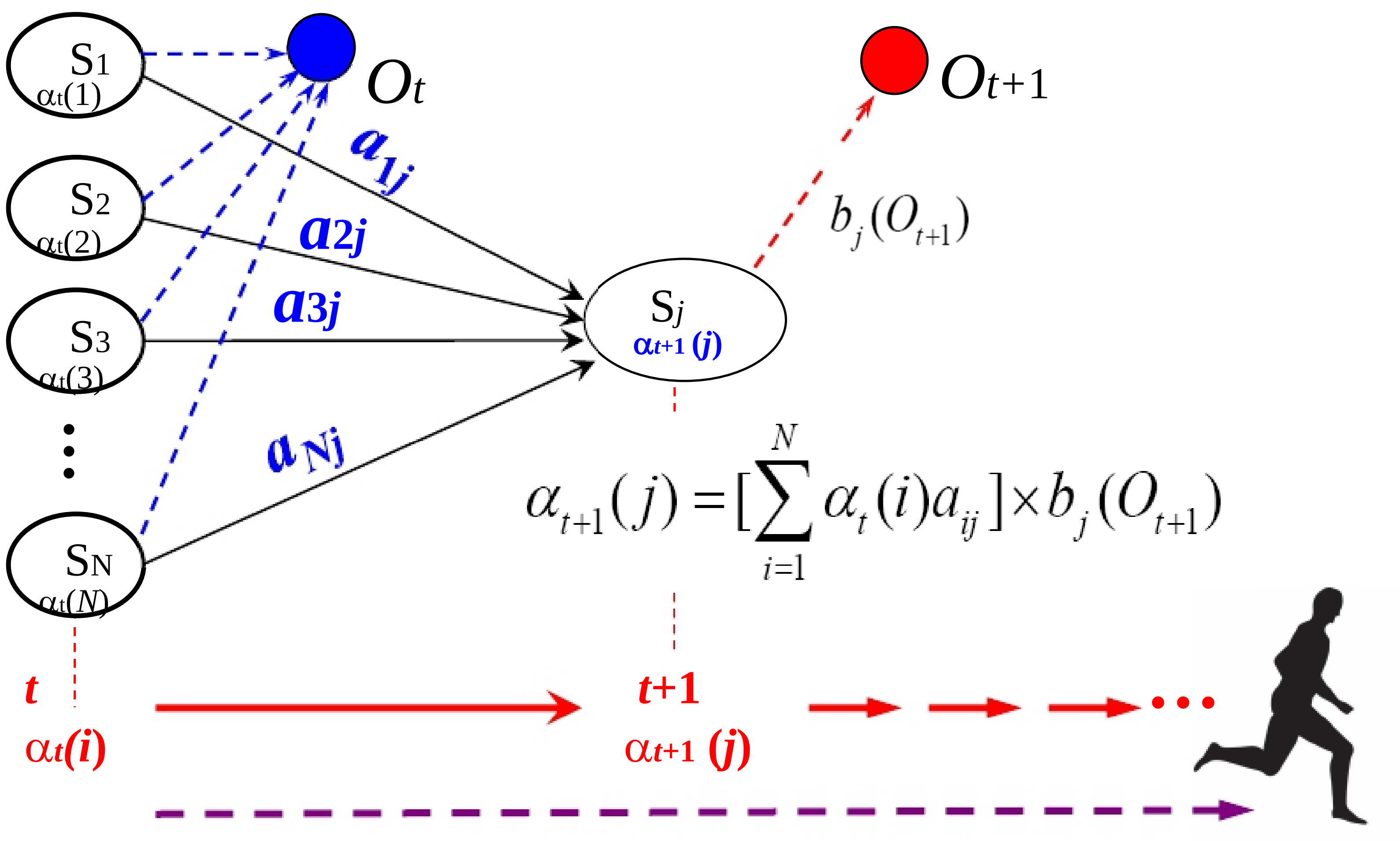
动态规划计算 $\alpha_t(i)$：时间 $t+1$ 的前向变量可以根据时间 t 的前向变量 $\alpha_t(1),\cdots,\alpha_t(N)$ 的值递推计算：

状态j到Ot+1的发射概率

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] \times b_j(O_{t+1}) \qquad \dots (6.14)$$

前一时刻所有可能状态i到下一时刻状态j的连接

6.3 前向算法



$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] \times b_j(O_{t+1})$$

6.3 前向算法

- 算法6.1：前向算法描述

(1) 初始化： $\alpha_1(i) = \pi_i b_i(O_1)$， $1 \le i \le N$

(2) 循环计算：

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] \times b_j(O_{t+1}), \quad 1 \le t \le T-1$$

(3) 结束，输出：

$$p(O \mid \mu) = \sum_{i=1}^{N} \alpha_T(i)$$

6.4 前向算法-实例分析

观察集合是: V={红，白}，M=2

状态集合是: Q={盒子1，盒子2，盒子3}，N=3

球的颜色的观测序列: O={红，白，红}

初始状态分布为: Π=(0.2, 0.4, 0.4)

其它转移概率、发射概率均已知(未列出)。

（1）首先计算时刻1三个状态的前向变量：时刻1是红色球，

隐藏状态是盒子1的概率为：

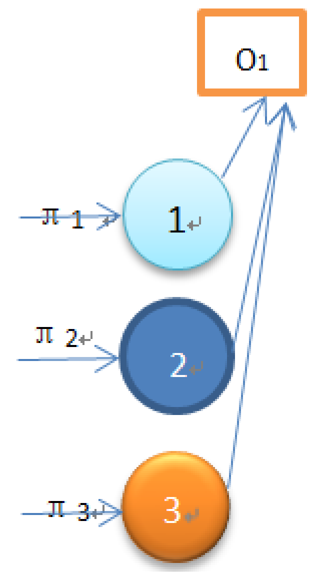
$$\alpha_1(1) = \pi_1 b_1(o_1) = 0.2 \times 0.5 = 0.1$$

隐藏状态是盒子2的概率为：

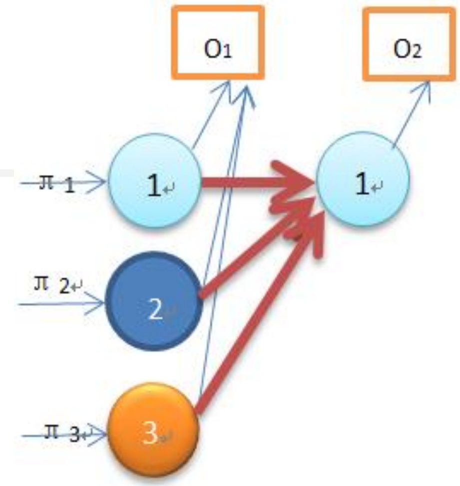
$$\alpha_1(2) = \pi_2 b_2(o_1) = 0.4 \times 0.4 = 0.16$$

隐藏状态是盒子3的概率为：

$$\alpha_1(3) = \pi_3 b_3(o_1) = 0.4 \times 0.7 = 0.28$$



6.4 前向算法–实例分析



球的颜色的观测序列: O={红，白，红}

（2）开始递推，时刻2三个状态的前向概率：时刻2是白色球
隐藏状态是盒子1的概率为：

转移概率　　　　发射概率

$$\alpha_2(1)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i1}\right]b_1(o_2)=[0.1*0.5+0.16*0.3+0.28*0.2]\times 0.5=0.077$$

隐藏状态是盒子2的概率为：

$$\alpha_2(2)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i2}\right]b_2(o_2)=[0.1*0.2+0.16*0.5+0.28*0.3]\times 0.6=0.1104$$

隐藏状态是盒子3的概率为：

$$\alpha_2(3)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i3}\right]b_3(o_2)=[0.1*0.3+0.16*0.2+0.28*0.5]\times 0.3=0.0606$$

6.4 后向算法

6.4 后向算法

- 后向算法 (The backward procedure)

后向变量 $\beta_t(i)$：是在给定了模型$\mu = (A,B,\pi)$和时间 t 时状态为 S_i 的条件下，模型输出观察序$O_{t+1}O_{t+2}\ldots\ldots O_T$ 的概率：

$$\beta_t(i) = p(O_{t+1}O_{t+2}\cdots O_T \mid q_t = S_i, \mu)$$

基于当前状态i预测后续输出$O_{t+1}\ldots O_T$的概率

后向变量存储：“从最后时刻每个可能状态，进入到t时刻状态i的概率(累加) x 发射概率”

6.4 后向算法

与前向变量一样，运用动态规划计算后向变量：

(1)当t=T时 $\beta_T(i)=1,\quad 1\le i\le N$

(2)在时间 t=T-1时

$$\beta_t(i)=\sum_{j=1}^{N}a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j)$$

由 β_{t+1}倒推β_t

归纳顺序： $\beta_T(x),\beta_{T-1}(x),\cdots,\beta_1(x)$

6.4 后向算法

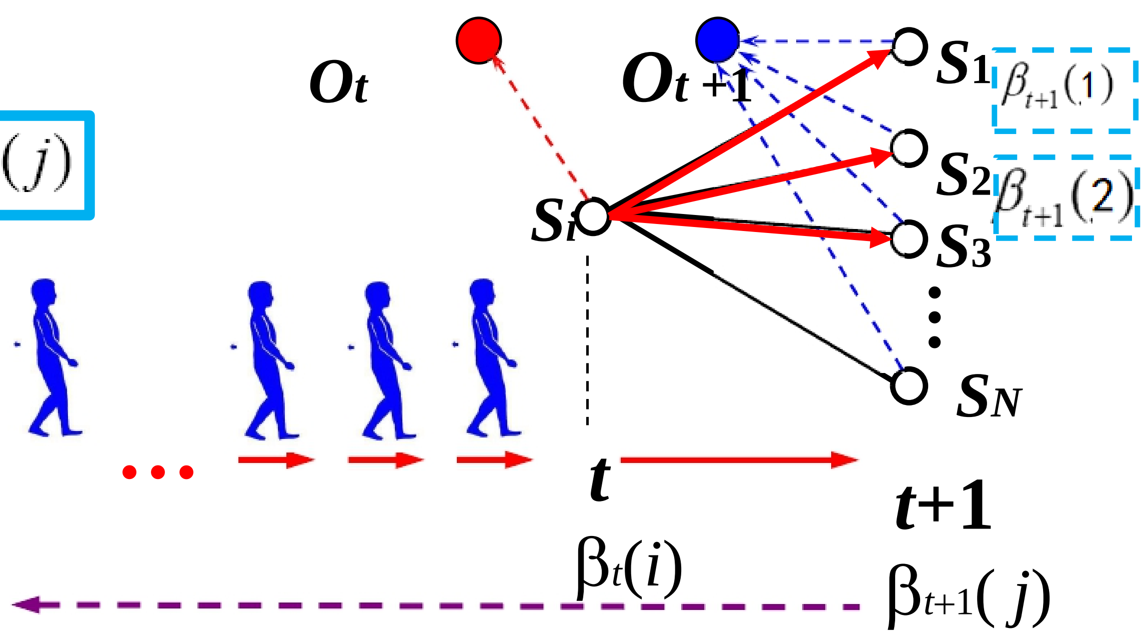
算法图解：

(1) 从时刻 t 到 $t+1$，模型由状态 S_i 转移到状态 S_j，并从 S_j 输出 O_{t+1}；

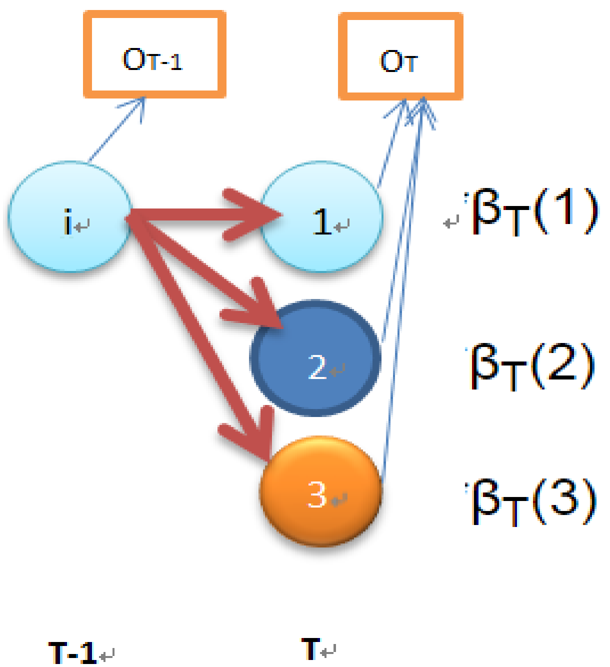
(2) 在时间 $t+1$，状态为 S_j 的条件下，模型输出观察序列 $O_{t+2}O_{t+3}\cdots O_T$ 。

$$\beta_t(i)=\sum_{j=1}^{N}\boxed{a_{ij}b_j(O_{t+1})}\times\boxed{\beta_{t+1}(j)}$$

基于当前状态i预测下一时刻输出O_{t+1}的概率



6.4 后向算法



$$\beta_{T-1}(i) = P(O_T | q_{T-1}=s_i, \mu) =$$

$$a_{i1} * b_1(O_T) * \beta_T(1) + a_{i2} * b_2(O_T) * \beta_T(2)$$

$$+ a_{i3} * b_3(O_T) * \beta_T(3)$$

6.4 后向算法

- 算法6.2：后向算法描述

(1) 初始化：$\beta_T(i)=1,\quad 1\le i\le N$

从最后时刻T开始

(2) 循环计算：

$$\beta_t(i)=\sum_{j=1}^{N}a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j),\quad T-1\ge t\ge 1,\quad 1\le i\le N$$

(3) 输出结果：$p(O\mid\mu)=\sum_{i=1}^{N}\beta_1(i)\times\pi_i\times b_i(O_1)$

6.5 Viterbi搜索算法

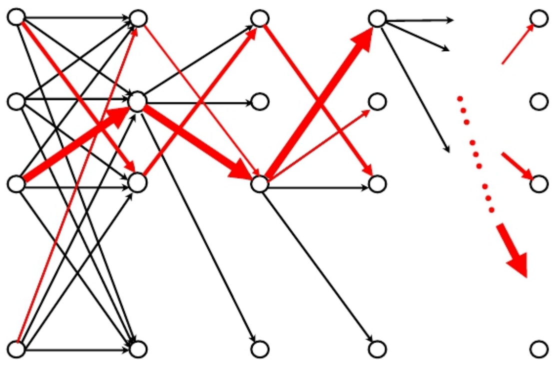
6.5 Viterbi 搜索算法

◆问题2—如何发现“最优”状态序列

能够“最好地解释”观察序列

一种解释：在给定模型μ 和观察序列O的条件下求概率最大的状态序列：

$$\hat{Q} = \underset{Q}{\operatorname{argmax}}\ p(Q|O,\mu) \qquad \dots (6.21)$$



Viterbi 算法： 利用动态规划求解概率最大的路径，该路径对应一个状态序列。

6.5 Viterbi 搜索算法

原理：从t=1时刻开始，不断向后递推到下一个状态经历路径的最大概率，直至最后到达终点。然后从终点回溯到起始点，这样就能得到最优路径。

定义：**Viterbi 变量**$\delta_t(i)$是在时间 t 时，模型沿着某一条路径到达状态 S_i，且输出观察序列 $O=O_1O_2\ldots O_t$ 的最大概率为：

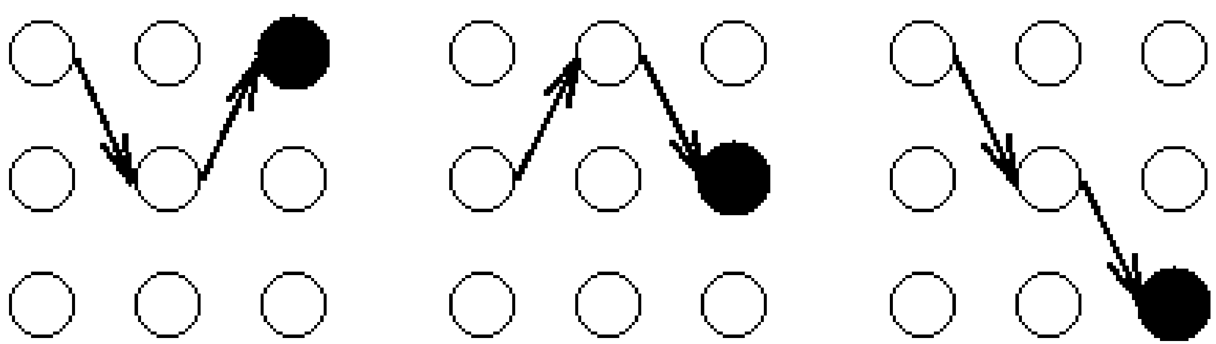
$$\delta_t(i) = \max_{\boxed{q_1,q_2,\cdots,q_{t-1}}} p(q_1,q_2,\cdots,q_t = S_i, O_1O_2\cdots O_t \mid \mu) \quad \ldots (6.22)$$

6.5 Viterbi 搜索算法

$$\delta_t(i) = \max_{q_1,q_2,\cdots,q_{t-1}} p(\underbrace{q_1,q_2,\cdots,q_t}= S_i, O_1O_2\cdots O_t \mid \mu) \quad \ldots (6.22)$$

变量$\delta t(i)$存储了一条到达中间状态Si时的局部最优路径，且通过该路径到达状态Si且观测到Ot的概率为$\delta t(i)$。

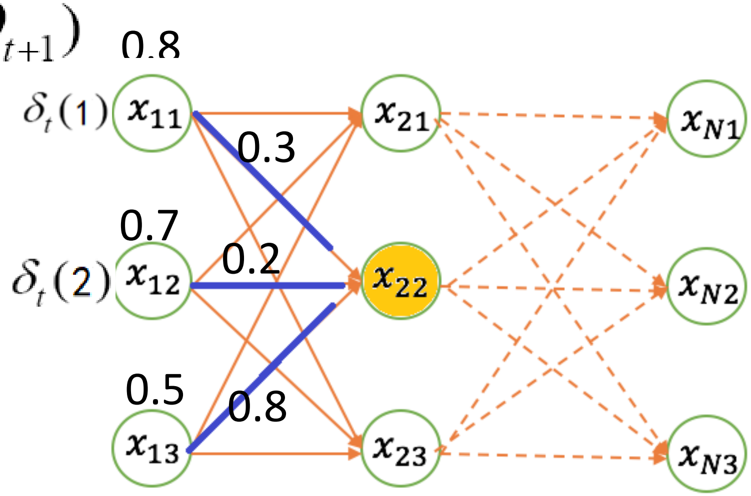
通常时刻t时，到达不同状态Si都有一条可能的最优路径，如第3时刻，每个状态Si 的最优路径如下：



6.5 Viterbi 搜索算法

递归计算：$\delta_{t+1}(i) = \max_j [\delta_t(j) \cdot a_{ji}] \cdot b_i(O_{t+1})$

三选一



t时刻

t+1时刻
观测值Ot+1

- **算法6.3**：**Viterbi 算法描述**

(1) 初始化：$\delta_1(i) = \pi_i b_i(O_1)$，$1 \le i \le N$

概率最大的路径变量：$\psi_1(i) = 0$

(2) 递推计算：

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \quad 2 \le t \le T, \quad 1 \le j \le N$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \ 2 \le t \le T, \ 1 \le i \le N$$

6.5 Viterbi 搜索算法

(3) 结束：

T时刻，所有状态中δ最大的那个状态i

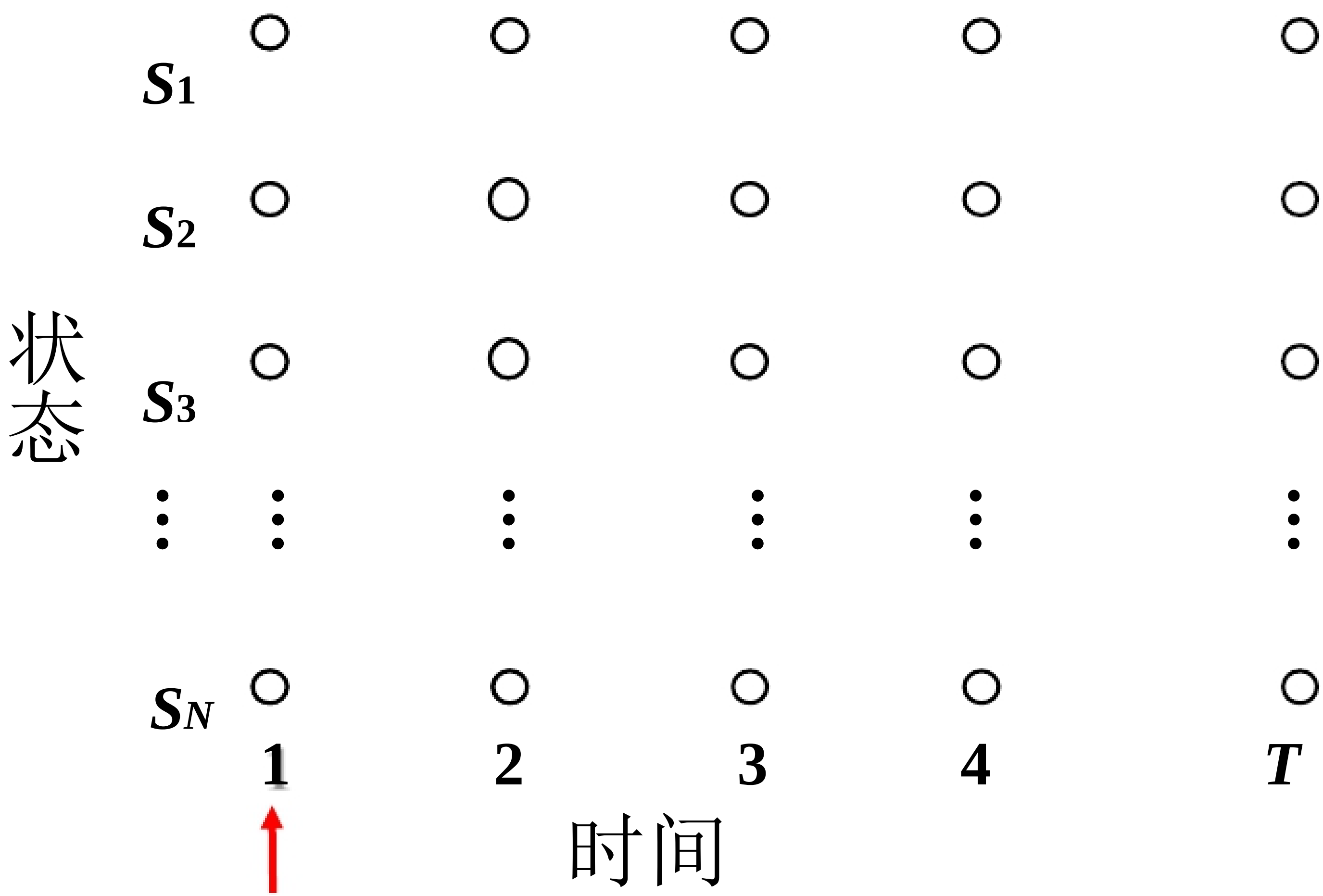
$$\widehat{Q}_T = \arg\max_{1 \le i \le N} [\delta_T(i)], \qquad \widehat{p}(\widehat{Q}_T) = \max_{1 \le i \le N} \delta_T(i)$$

(4) 通过回溯得到路径（状态序列）：

$$\widehat{q}_t = \psi_{t+1}(\widehat{q}_{t+1}), \quad t = T-1, T-2, \cdots, 1$$

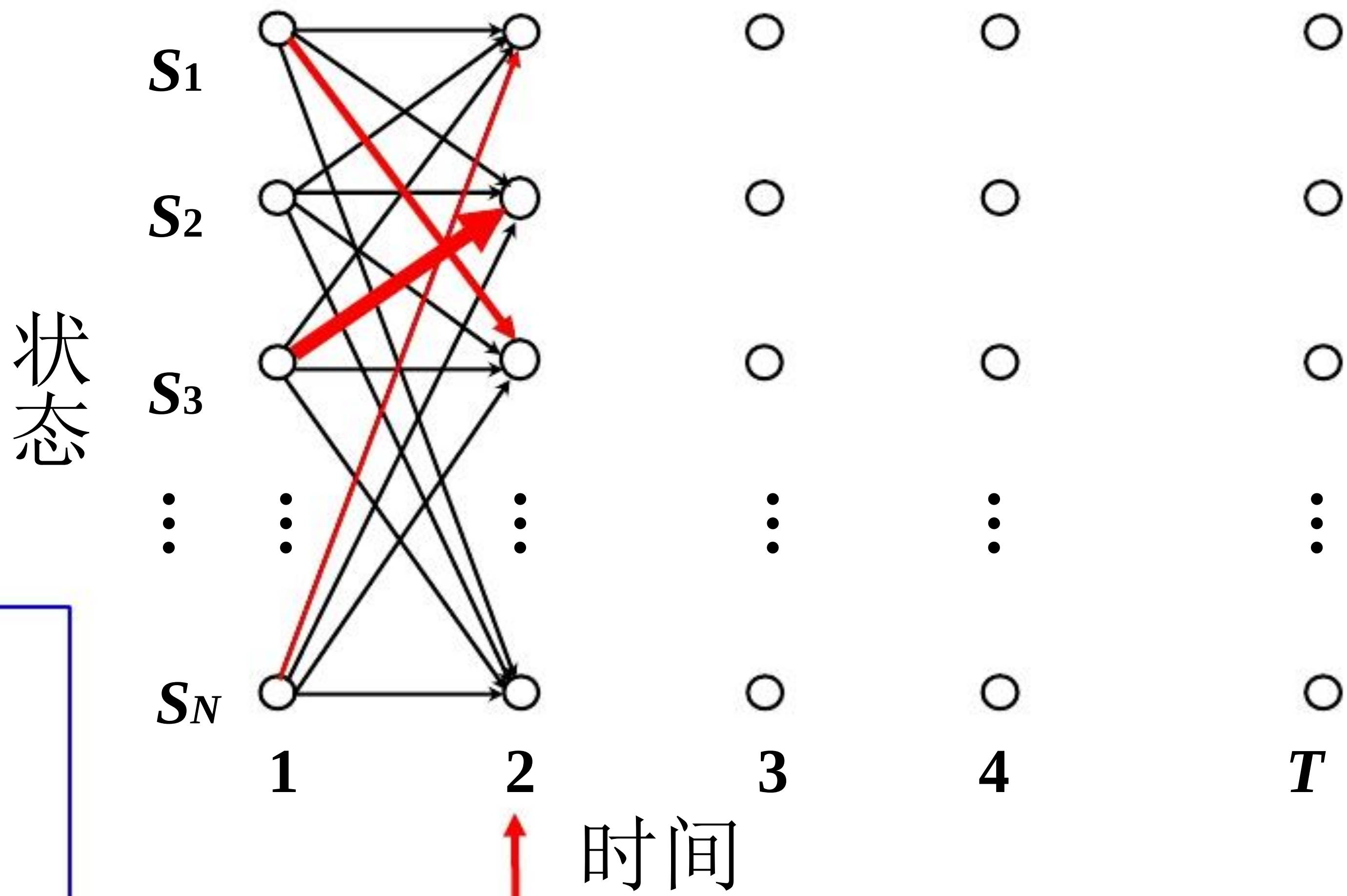
6.5 Viterbi 搜索算法

图解
Viterbi
搜索
过程



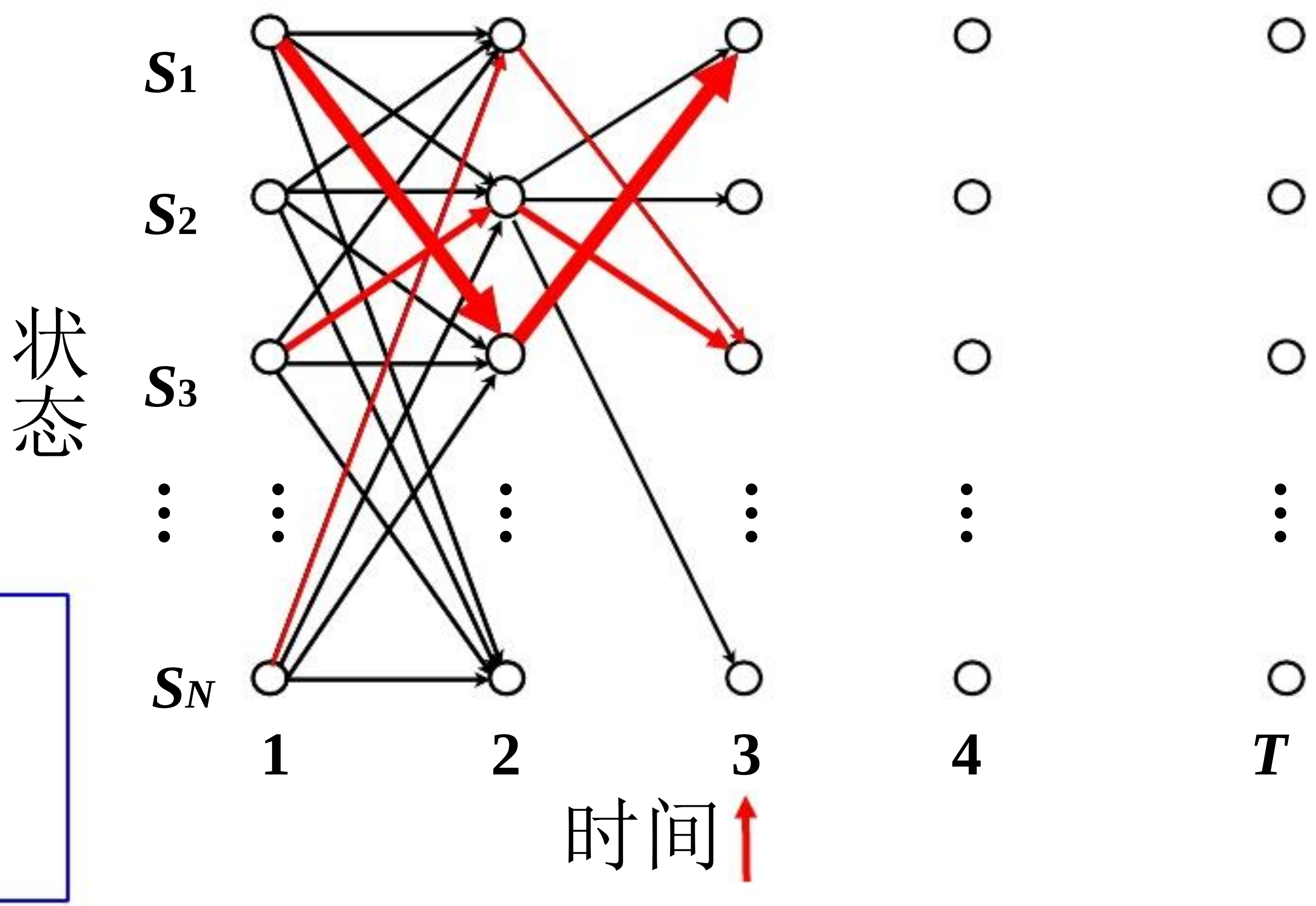
6.5 Viterbi 搜索算法

图解
Viterbi
搜索
过程



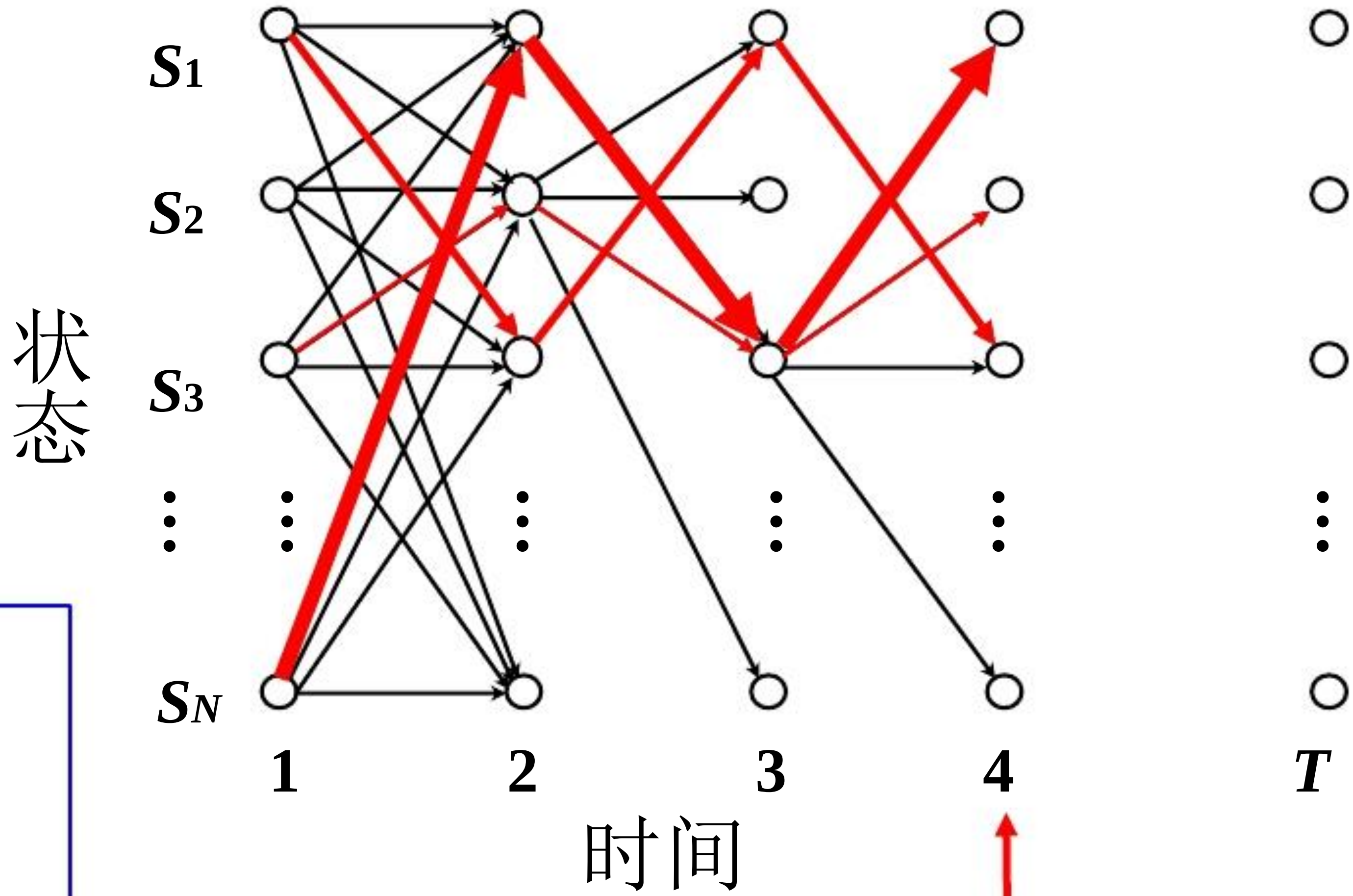
6.5 Viterbi 搜索算法

图解 **Viterbi** 搜索过程



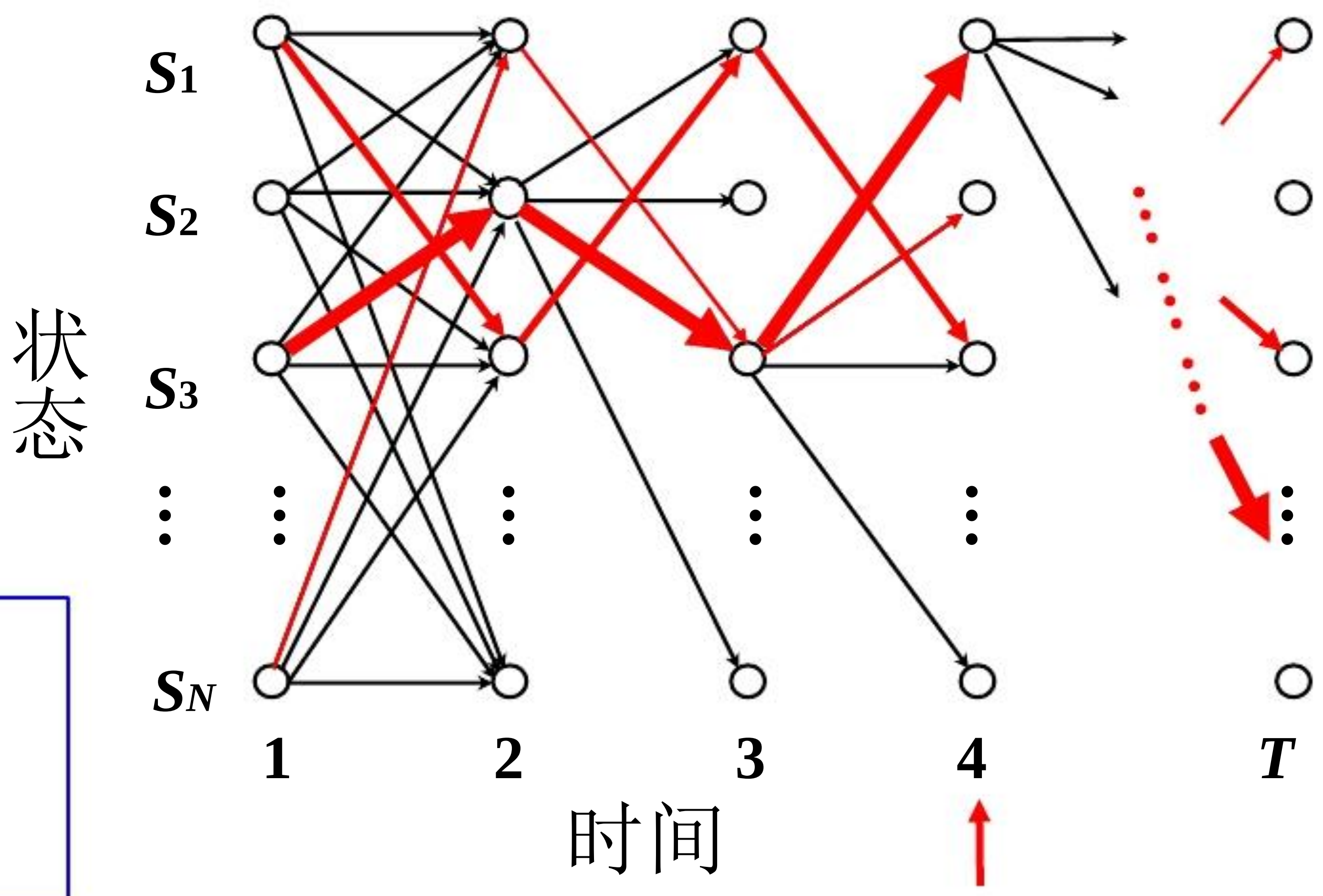
6.5 Viterbi 搜索算法

图解 **Viterbi** 搜索过程



6.5 Viterbi 搜索算法

图解 **Viterbi** 搜索过程



6.6 参数学习

6.6 参数学习

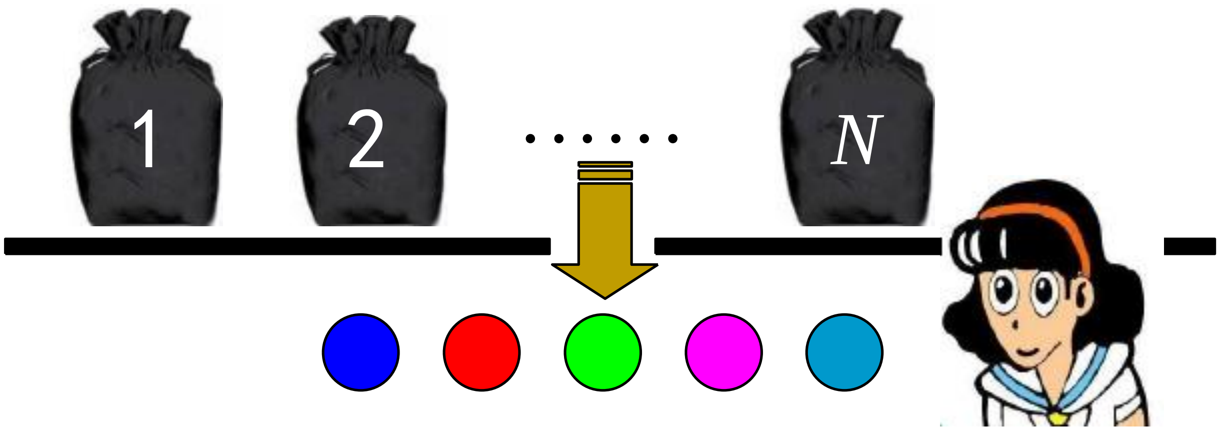
◆问题3－模型参数学习

给定一个观察序列 $O = O_1O_2 \ldots O_T$，如何根据极大似然估计来求模型的参数值？

即估计模型中的π_i, a_{ij}, $b_j(k)$使得观察序列 O 的概率 $p(O|\mu)$ 最大。

6.6 参数学习

(1) 如果产生观察序列 O 时，状态 $Q = q_1q_2\ldots q_T$已知(即存在状态已标注的样本)，可以用极大似然估计来计算 μ 的参数：



相当于，实验员从哪个袋子取球的过程是透明的，我们知道整个过程经历了哪些内部状态改变。

6.6 参数学习

(1)如果产生观察序列 O 的状态 $Q = q_1q_2\ldots q_T$已知, 可以用极大似然估计来计算 μ 的参数:

$$\bar{\pi}_i = \delta(q_1, S_i)$$ 时刻1处于状态Si的次数

$$\bar{a}_{ij} = \frac{Q\text{中从状态}q_i\text{转移到}q_j\text{的次数}}{Q\text{中所有从状态}q_i\text{转移到另一状态(包括}q_j\text{自身)的总数}}$$

$$= \frac{\sum_{t=1}^{T-1} \delta(q_t, S_i) \times \delta(q_{t+1}, S_j)}{\sum_{t=1}^{T-1} \delta(q_t, S_i)} \quad \ldots (6.24)$$

其中，δ(*x*, *y*) 为克罗奈克(Kronecker)函数， 当 *x*= *y* 时, δ(*x*, *y*)=1, 否则δ(*x*, *y*) = 0。

6.6 参数学习

类似地，

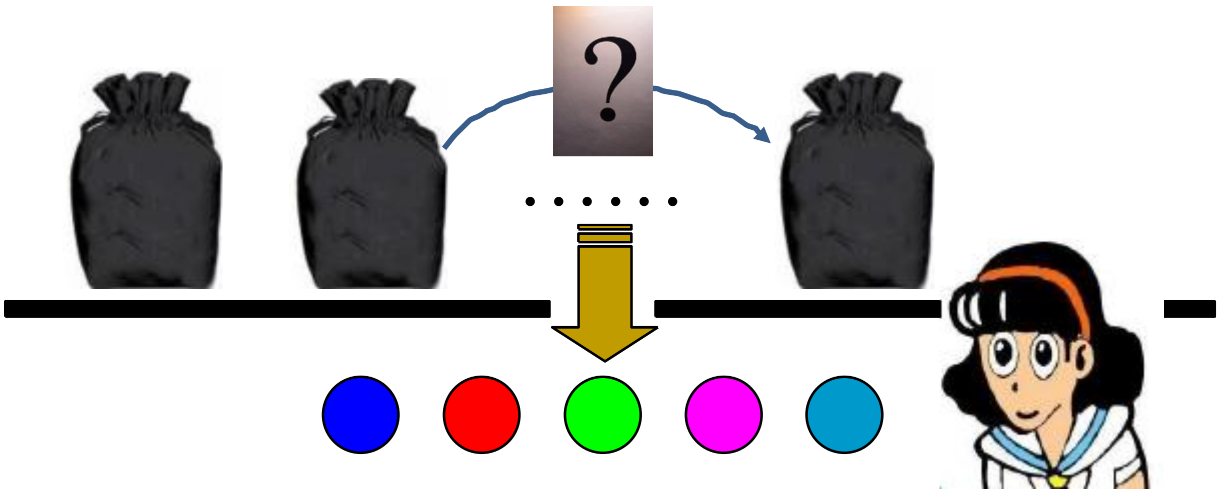
$$\bar{b}_j(k) = \frac{Q\text{中从状态}q_j\text{输出符号}v_k\text{的次数}}{Q\text{到达}q_j\text{的总次数}}$$

$$= \frac{\sum_{t=1}^{T}\delta(q_t, S_j) \times \delta(O_t, v_k)}{\sum_{t=1}^{T}\delta(q_t, S_j)} \qquad \ldots (6.25)$$

其中，v_k 是模型输出符号集中的第 k 个符号。

6.6 参数学习

（2）如果不存在（状态）标注的样本



这时，只能观察到取出球的序列，但整个过程经历了哪些内部状态改变是未知的。

6.6 参数学习

如果不存在状态标注的样本。可以采用期望最大算法 (Expectation-Maximization, EM)，基本思想：

（1）初始化时，随机地给模型的参数赋值，得到模型 μ_0

（2）对每个样本，根据 μ_0 求模型中隐变量的期望值。

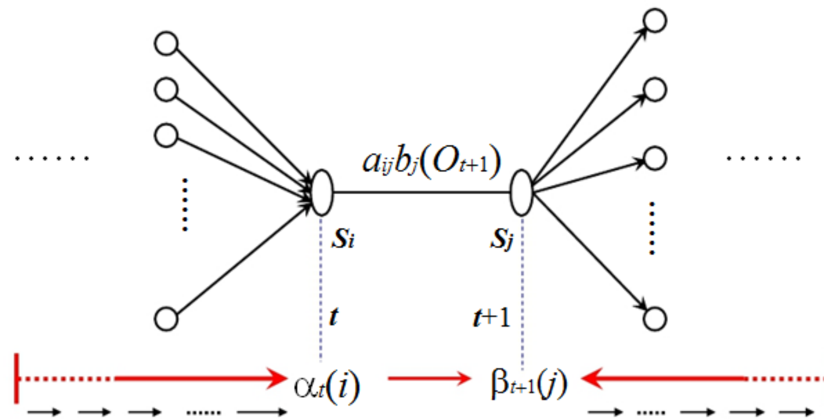
如：根据 μ_0 求得到从某一状态转移到另一状态的期望次数

(3) 然后以期望次数代替公式中的实际次数，更新得到新的模型 μ_1。

循环这一过程，直到参数收敛于最大似然估计值。

6.6 参数学习

$$\alpha_t(i) = p(O_1O_2\cdots O_t, q_t = S_i \mid \mu)$$



推算准备1：给定模型 μ和观察序列 $O=O1O2\ldots OT$，时间(t)位于状态 Si，后一时间(t+1) 位于状态 Sj 的转移概率估计：

$$\xi_t(i,j) = p(q_t = S_i, q_{t+1} = S_j \mid O, \mu) = \frac{p(q_t = S_i, q_{t+1} = S_j, O \mid \mu)}{p(O \mid \mu)}$$

$$= \frac{\alpha_t(i) \times a_{ij} b_j(O_{t+1}) \times \beta_{t+1}(j)}{p(O \mid \mu)}$$

计算中要用到初始模型参数

$$= \frac{\alpha_t(i) \times a_{ij} b_j(O_{t+1}) \times \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i) \times a_{ij} b_j(O_{t+1}) \times \beta_{t+1}(j)}$$

(1) 初始化： $\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \le i \le N$

(2) 循环计算：

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] \times b_j(O_{t+1}), \quad 1 \le t \le T-1$$

6.6 参数学习

推算准备2： 给定模型 μ 和观察序列 $O = O_1 O_2 \ldots\ldots O_T$，在时间 t 位于状态 S_i 的概率为：

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j) \qquad \ldots (6.27)$$

接下来，模型参数μ 可由下面的公式重新估计：

(1) q_1 为 S_i 的初始概率：

$$\pi_i = \gamma_1(\mathrm{i}) \qquad \ldots (6.28)$$

6.6 参数学习

(2)

$$\bar{a}_{ij} = \frac{Q\text{中从状态}q_i\text{ 转移到 }q_j\text{的期望次数}}{Q\text{中所有从状态 }q_i\text{ 转移到下一状态(包括 }q_j\text{自身)的期望次数}}$$

$$= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad \cdots (6.29)$$

见（6. 27）

6.6 参数学习

(3)

$$\bar{b}_j(k) = \frac{Q\text{中从状态}q_j\text{输出符号}v_k\text{的期望次数}}{Q\text{到达}q_j\text{的期望次数}}$$

$$= \frac{\sum_{t=1}^{T} \gamma_t(j) \times \delta(O_t, v_k)}{\sum_{t=1}^{T} \gamma_t(j)} \qquad \ldots (6.30)$$

6.6 参数学习

- **算法6.4**: Baum-Welch 算法(前向后向算法)描述:

(1) 初始化：随机地给 π_i, a_{ij}, $b_j(k)$ 赋值，

使得

$$\begin{cases} \sum_{i=1}^{N} \pi_i = 1 & \\ \sum_{j=1}^{N} a_{ij} = 1 & 1 \le i \le N \\ \sum_{k=1}^{M} b_i(k) = 1 & 1 \le i \le N \end{cases} \quad \ldots (6.31)$$

由此得到模型 μ_0，令 $i = 0$。

6.6 参数学习

(2) 执行 EM 算法:

$$\xi_t(i,j)=\frac{\alpha_t(i)\times a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)\times a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j)}$$

$$\gamma_t(i)=\sum_{j=1}^{N}\xi_t(i,j)$$

E-步: 由模型 μ_i 根据公式 (6.26) 和 (6.27) 计算期望值 $\xi_t(i,j)$和 $\gamma_t(i)$。

M-步: 用E-步中所得到的期望值，根据公式 (6.28-6.30) 重新估计π_i, a_{ij}, $b_j(k)$ 得到模型μ_{i+1}。

$$\bar{a}_{ij}=\frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}$$

$$\bar{b}_j(k)=\frac{\sum_{t=1}^{T}\gamma_t(j)\times\delta(O_t,v_k)}{\sum_{t=1}^{T}\gamma_t(j)}$$

6.6 参数学习

(2) 执行 EM 算法：

$$\xi_t(i,j)=\frac{\alpha_t(i)\times a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)\times a_{ij}b_j(O_{t+1})\times\beta_{t+1}(j)}$$

$$\gamma_t(i)=\sum_{j=1}^{N}\xi_t(i,j)$$

E-步：由模型 μ_i 根据公式 (6.26) 和 (6.27) 计算期望值 $\xi_t(i,j)$和 $\gamma_t(i)$。

M-步：用E-步中所得到的期望值，根据公式 (6.28-6.30) 重新估计 π_i, a_{ij}, $b_j(k)$ 得到模型 μ_{i+1}。

循环：$i=i+1$，重复执行 E-步和M-步，直至 π_i, a_{ij}, $b_j(k)$的值收敛：$|\log p(O|\mu_{i+1})-\log p(O|\mu_i)|<\varepsilon$ 。

(3) 结束算法，获得相应的参数。

6.6 参数学习

假设一个HMM的模型的状态集S={1, 2, 3}, 观测集V={1, 2}, π = (0, 1, 0), 转移概率A, 发射概率B如下：

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0.0193 & 0 & 0.9807 \\ 0.0001 & 0.9999 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0.9858 & 0.0142 \\ 1 & 0 \\ 0.1505 & 0.8495 \end{bmatrix}.$$

使用1000个观测值O=(1, 2, 1, 2, 1, 2, 1, 2, 1, 1……)， 训练后

$$\pi^{**} = (0.0000, 1.0000, 0.0000)$$

$$A^{**} = \begin{bmatrix} 0.0000 & 1.0000 & 0.0000 \\ 0.0565 & 0.0000 & 0.9435 \\ 0.0000 & 1.0000 & 0.0000 \end{bmatrix} \qquad B^{**} = \begin{bmatrix} 0.9369 & 0.0631 \\ 1.0000 & 0.0000 \\ 0.1304 & 0.8696 \end{bmatrix}.$$

HMM应用

中文分词

（1）将状态集Q设为{B, E, M, S}，表示词的开始、结束、中间（begin、end、middle）及字符独立成词（single）；

（2）观测序列即为中文句子。比如，“今天天气不错”；

（3）中文分词的任务对应于前述问题二（解码）：利用维特比算法找到该观测序列下最优的状态序列，如：

“B E B E B E”

则分词结果为“今天/天气/不错”。

即：对观测值$C\{c_1, c_2, \cdots, c_n\}$，求下列最大条件概率

$$max\ P(q_1, q_2, \ldots, q_t \mid c_1, c_2 \ldots c_n)$$ qi表示字符c_i对应的分词状态{B, E, M, S}；

HMM应用

假定训练样本集如下：

北	N	B
京	N	E
欢	V	B
迎	V	M
你	N	E

思考：HMM可以用于解决语音识别、唇语识别问题吗？如何构建隐状态？

6.8 CRFs及其应用

6.8 CRFs及其应用

◆提出

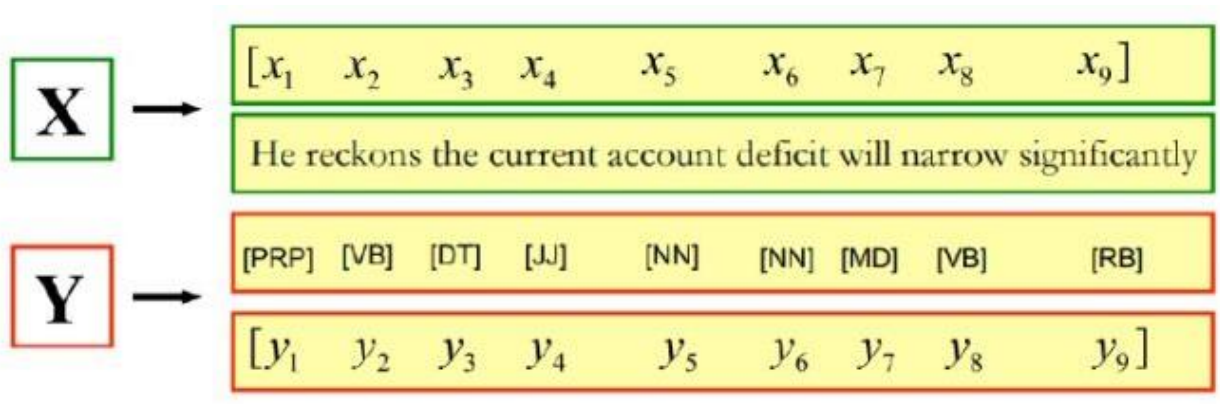
条件随机场(conditional random fields, CRFs)于2001年由 **J. Lafferty** 等人提出，是用于序列标注和结构划分的概率模型，在**NLP**和图像处理中得到了广泛应用。

基本思路：给定观察序列 $\boldsymbol{X}$，输出标识序列 $\boldsymbol{Y}$，通过计算 $\boldsymbol{P(Y|X)}$ 求解最优标注序列。

6.8 CRFs及其应用

◆ 定义

随机场是由若干个位置组成的整体，当给每一个位置中按照某种分布随机赋予一个值之后，其全体就叫做随机场



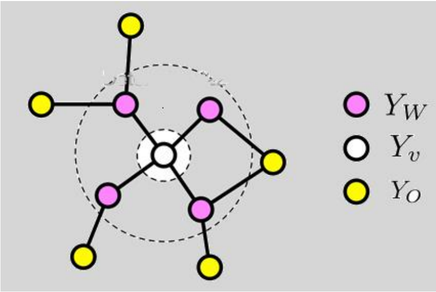
6.8 CRFs及其应用

◆定义

设 $G=(V, E)$ 为一个无向图，V为结点集，E为无向边集，$Y = \{ Y_v \mid v \in V \}$，即每个随机变量 Y_v 对应V中一个结点, 其取值范围为可能的标记集合$\{y\}$。

如果以观察序列X为条件，每个随机变量 Y_v 都满足以下马尔可夫特性：

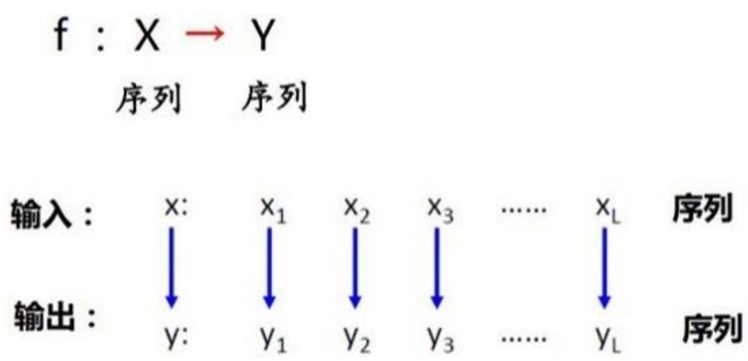
$$p(Y_v \mid X, Y_w, w \neq v) = p(Y_v \mid X, Y_w, w \sim v)$$

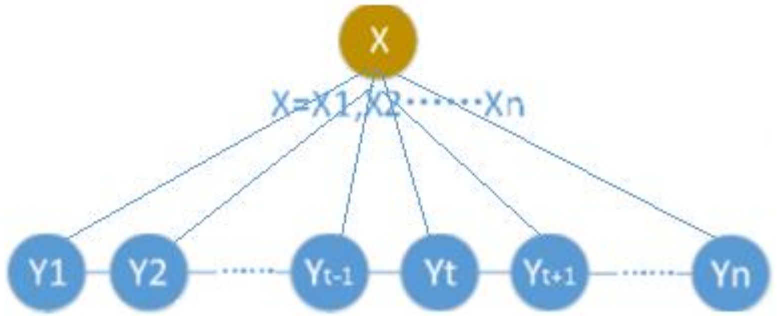


其中，$w \sim v$ 表示两个结点在图中是邻居结点。那么，(X, Y) 为一个条件随机场。

6.8 CRFs及其应用

序列标注问题

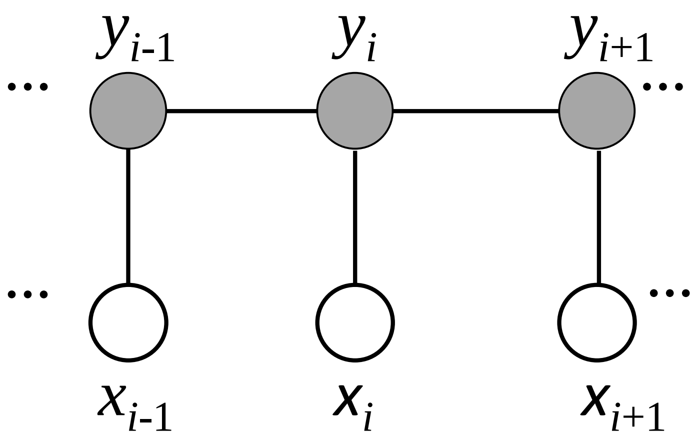




该问题可以用线性链CRF求解。

6.8 CRFs及其应用

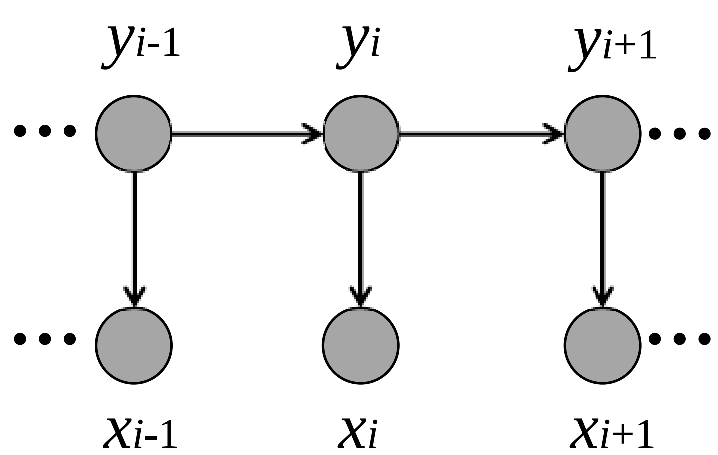
一般地，当X和Y具有相同图结构时，线性链结构就变为如下所示



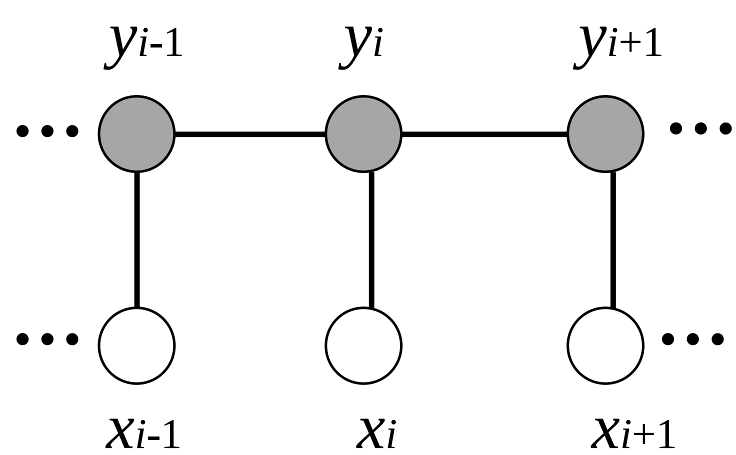
X是给定的（观测序列）,Y才是预测对象（状态序列）。

6.8 CRFs及其应用

HMMs vs. CRFs



HMMs



CRFs

一个是有向图，一个是无向图。

一个是生成式模型，一个是判别式模型

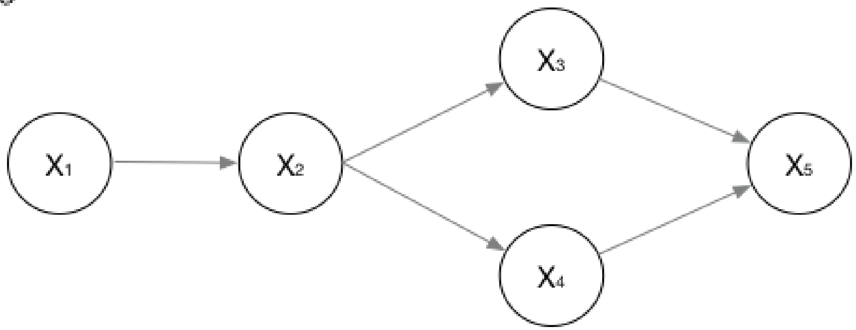
CRFs 中的空心节点 ***x*** 表示该节点并不是由模型生成的。

6.8 CRFs及其应用

概率有向图的联合概率计算

$$P(x_1, \cdots, x_n) = \prod_{i=0} P(x_i | \pi(x_i))$$

如右图的联合概率为：



$$P(x_1, \cdots, x_n) = P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_2) \cdot P(x_4|x_2) \cdot P(x_5|x_3, x_4)$$

6.8 CRFs及其应用

概率无向图的联合概率计算

概率无向图的联合概率多采用因子分解的方式，将其表示为若干个团的联合概率乘积。

$$P(Y) = \frac{1}{Z}\prod_{C}\Psi_C(Y_C) \qquad Z = \sum_{Y}\prod_{C}\Psi_C(Y_C)$$

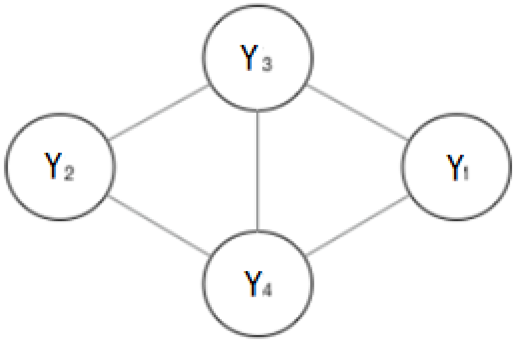
Z为规范化因子，保证P(Y)构成概率分布

团（clique）是指两两之间都有连边的点的集合

6.8 CRFs及其应用

概率有向图的联合概率计算

注意：概率计算中的团必须是“最大团”，即最大连通子图



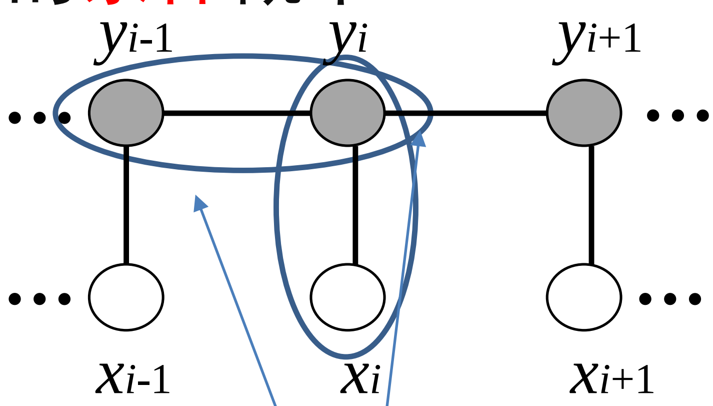
联合概率：$P(Y)=\frac{1}{Z}\prod_{C}\Psi_C(Y_C)$

$$P(Y)=\frac{1}{Z(Y)}(\psi_1(Y_1,Y_3,Y_4)\cdot\psi_2(Y_2,Y_3,Y_4))$$

$\psi_c(Y_c)$ 叫势函数， 多用指数形式：$\psi_c(Y_c)=e^{-E(Y_c)}$

6.8 CRFs及其应用

线性链CRF的条件概率



利用因子分解式，线性链CRF的条件概率P(Y|X)：

$$P(Y|X) = \frac{1}{Z(x)} \prod_c \psi_c(Y_c|X) = \frac{1}{Z(x)} e^{\sum_c \sum_k \lambda_k f_k(y_i, y_{i-1}, x, i)}$$

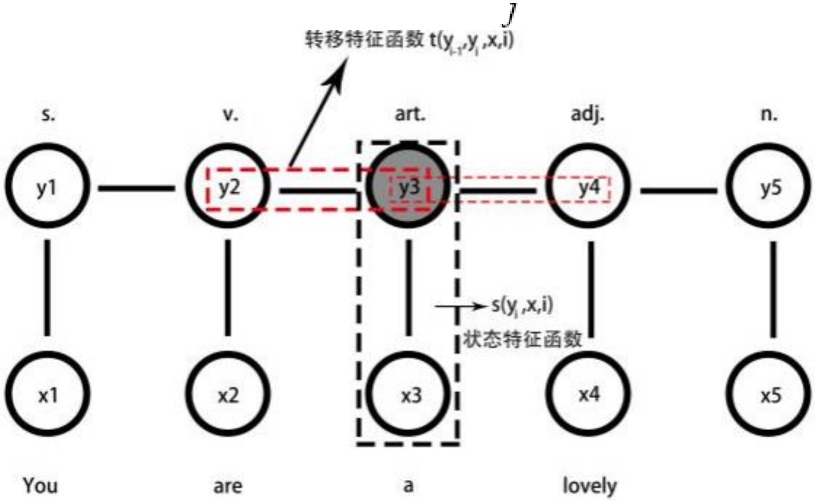
其中，f为团上的特征函数，由它决定团的能量大小。

6.8 CRFs及其应用

CRF特征函数类型

- 状态特征函数$s_k(yi, X, i)$， 表示观察序列X 在 i 位置的标记概率；
- 转移特征函数 $t_j(yi\text{-}1, yi, X, i)$,表示标注序列Y在 i 及 i-1 位置上标记的转移概率；

$$P(Y|X)= \exp\left(\sum_j \lambda_j t_j(y_{i-1}, y_i, X, i) + \sum_k \mu_k s_k(y_i, X, i)\right)$$



λ_j和μ_k分别是 t_j和s_k的权重，这是模型需要从训练样本集中学习的主要参数。

6.8 CRFs及其应用

特征函数理解

$$P(Y|X) = \frac{1}{Z(x)} e^{\sum_c \sum_k \lambda_k f_k(y_i, y_{i-1}, x, i)}$$

这里不区分函数t和s，统一记为f
i表示序列中第i个位置

特征函数定义了团中变量可能取值的组合，如Y=｛男，女｝，X=｛长发，短发｝，则可能形成4个特征函数：

当训练样本中 x=短发　f1:{男，短发}　f2:{女，短发}

当训练样本中 x=长发　f3:{女，长发}　f4:{男，长发}

每个f前的参数λ用于衡量该特征函数的价值，其由训练样本集决定。

特征函数f的取值只有2个：如果特征满足则为1，不满足则为0。

6.8 CRFs及其应用

特征函数理解

序列标注时：

如输入样本x为“短发”，则分别计算：

Y标记为“男”时，此时满足f1为1， P(男|短发)

Y标记为“女”时，此时满足f2为1， P(女|短发)

考虑到，训练时f1的参数应该更大，因此 P(男|短发) 更大

6.8 CRFs及其应用

特征函数理解

例如：对语句x=“我 在 楼上 学习”进行词性标记。可能标记序列包括：

y1={代词 介词 名词 动词}；　　y2={代词 介词 介词 名词}；

y3= …….

求：argmaxP(y|x)　　　　因此，需要计算所有可能标记序列的概率

$$f1=\begin{cases}1 & \text{yi\_1是介词，yi是名词时}\\ 0 & \text{其它}\end{cases}\qquad f2=\begin{cases}1 & \text{yi\_1是介词，yi是介词时}\\ 0 & \text{其它}\end{cases}$$

通过训练样本，可以看到介词后多跟名词，而非介词，因此参数λ1大于λ2。此时，匹配到特征f1时的得分更高，因此P(y1|x)更大

$$P(Y|X)=\frac{1}{Z(x)}e^{\sum_c \sum_k \lambda_k f_k(y_i,y_{i-1},x,i)}$$

6.8 CRFs及其应用

◆ CRF词性标注实例

假设输入的都是三个词的句子，即X=(X1,X2,X3),输出的词性标记为Y=(y1,y2,y3),其中yi∈{1(名词)，2(动词)}

通过训练，假定得到特征函数如下（只列出取值为1的）：

状态特征函数

$$s_1(y_1=1,x,1)\quad \mu_1=1$$
$$s_2(y_i=2,x,i),i=1,2,\quad \mu_2=0.5$$
$$s_3(y_i=1,x,i),i=2,3,\quad \mu_3=0.8$$
$$s_4(y_3=2,x,3)\quad \mu_4=0.5$$

转移特征函数

$$t_1(y_{i-1}=1,y_i=2,x,i),i=2,3,\quad \lambda_1=1$$
$$t_2(y_1=1,y_2=1,x,2)\quad \lambda_2=0.5$$
$$t_3(y_2=2,y_3=1,x,3)\quad \lambda_3=1$$
$$t_4(y_1=2,y_2=1,x,2)\quad \lambda_4=1$$
$$t_5(y_2=2,y_3=2,x,3)\quad \lambda_5=0.2$$

求给定序列标记P(Y|x)=P((1,2,2)|x)的非规范化概率。

6.8 CRFs及其应用

◆ **CRF词性标注实例**

利用linear-CRF的参数化公式，我们有：

$$P(y|x) \propto exp\Big[\sum_{k=1}^{5}\lambda_k\sum_{i=2}^{3}t_k(y_{i-1},y_i,x,i)+\sum_{l=1}^{4}\mu_l\sum_{i=1}^{3}s_l(y_i,x,i)\Big]$$

代入标记序列(1, 2, 2)

$$P(y_1=1,y_2=2,y_3=2|x) \propto exp(3.2)$$

6.8 CRFs及其应用

实现 **CRFs** 需要解决如下三个问题：

①特征函数定义

②模型训练

课后阅读

确定参数λ_j和μ_k

③解码

采用viterbi求解序列标注问题

6.8 CRFs及其应用

①特征函数定义 $f_j(y_{i-1}, y_i, X, i)$

特征函数非常多，直接设计比较麻烦。可以采取先创建特征模版，再根据模版自动创建特征函数的方法。我们以CRF++为例，讲解模板构建方法。

特征模板格式：%x[row,col]

首字母可取U或B，对应两种特征函数类型。U表示生成状态特征函数，B表示生成转移特征函数。

row表示相对当前位置的行，0即是当前行；
col对应训练文件中的列。

```
U00:%x[-2,0]
U01:%x[-1,0]
U02:%x[0,0]
U03:%x[1,0]
U04:%x[2,0]
U05:%x[-2,0]/%x[-1,0]/%x[0,0]
U06:%x[-1,0]/%x[0,0]/%x[1,0]
```

6.8 CRFs及其应用

①特征函数定义

北	N	B
京	N	E
欢	V	B
迎	V	M
你	N	E

如果当前位置是‘京’，那按照下列模版取出的特征分别为：

U00:%x[0,0]=====> 京

U01:%x[-1,1]=====> N

U02:%x[-1,2]=====> B

U03:%x[1,0]/%x[2,0] =====>欢/迎

训练样本(日文分词)

特征模板(日文分词)

6.8 CRFs及其应用

①特征函数定义

这里说明标注的特征依据

北 N B
京 N E
欢 V B
迎 V M
你 N E

如，针对分词标记任务

根据模板U02:%x[0,0]，系统自动创建的特征函数如下：

func1=if(output=E and feature='U02:京') return 1 else return 0

如果当前位置字为京，标记为E，则返回1

该特征模版：将当前位置特征与标记作为特征对。

对每个模版中当前位置，系统会重复L次（L表示标记个数，如BIE）

func2=if(output=I and feature='U02:京') return 1 else return 0

func3=if(output=B and feature='U02:京') return 1 else return 0

合理的“特征函数”在样本中出现的次数较多，对应的权值参数就高，不合理的“特征函数”在样本中出现的少，对应的权重就小。

6.8 CRFs及其应用

①特征函数定义

北	N	B
京	N	E
欢	V	B
迎	V	M
你	N	E

对模板U02:%x[0,0]，然后下移，扫描下一个字'欢'，同样会得到三个特征函数：

func4=if(output=E and feature='U02:欢’) return 1 else return 0
func5=if(output=I and feature='U02:欢’) return 1 else return 0
func6=if(output=B and feature='U02:欢’) return 1 else return 0

对每个模版，每个样本行，重复执行，以生成多个特征函数

6.8 CRFs及其应用

②模型训练

包括梯度下降法、改进的迭代尺度法IIS、拟牛顿法

为了训练特征权重λ_j，需要计算模型的损失和梯度。由梯度更新λ_j，直到λ_j收敛。

- 损失函数定义为负对数似然函数：

$$L(\lambda) = -\log p(Y \mid X, \lambda) + \frac{\varepsilon}{2}\lambda^2 \qquad p(Y \mid X, \lambda) = \frac{1}{Z(X)}\exp\left(\lambda_j \cdot F_j(Y, X)\right)$$

- 损失函数的梯度为：$\dfrac{\partial L(\lambda)}{\partial \lambda_j} = \dfrac{\partial \log Z(X)}{\partial \lambda_j} - F_j(Y, X) + \varepsilon\lambda$

CRFs及其应用

③ 解码

条件随机场解码的过程就是给定条件随机场P(Y|X)和输入序列x，求条件概率最大的标记序列y*，即对观测序列进行标注。

可以由维特比 (**Viterbi**)算法完成。

CRFs及其应用

② 解码

输入：模型特征向量 $F(y,x)$ 和权值向量 w，观测序列 $x=(x_1,x_2,\cdots,x_n)$；

输出：最优路径 $y^*=(y_1^*,y_2^*,\cdots,y_n^*)$．

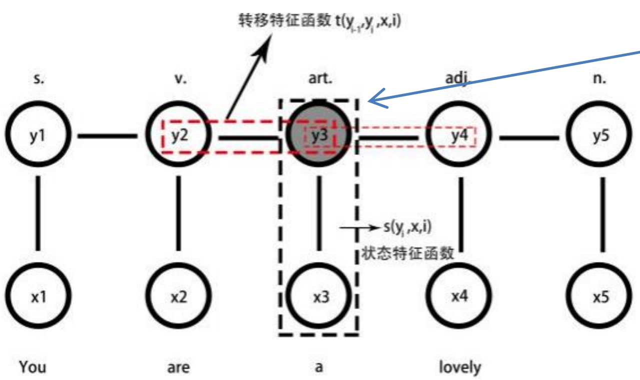
（1）初始化

$$\delta_1(j)=w\cdot F_1(y_0=start,y_1=j,x),\quad j=1,2,\cdots,m$$

（2）递推．对 $i=2,3,\cdots,n$

$$\delta_i(l)=\max_{1\leqslant j\leqslant m}\{\delta_{i-1}(j)+w\cdot F_i(y_{i-1}=j,y_i=l,x)\},\quad l=1,2,\cdots,m$$

$$\Psi_i(l)=\arg\max_{1\leqslant j\leqslant m}\{\delta_{i-1}(j)+w\cdot F_i(y_{i-1}=j,y_i=l,x)\},\quad l=1,2,\cdots,m$$



（3）终止

$$\max_y(w\cdot F(y,x))=\max_{1\leqslant j\leqslant m}\delta_n(j)$$

$$y_n^*=\arg\max_{1\leqslant j\leqslant m}\delta_n(j)$$

（4）返回路径

$$y_i^*=\Psi_{i+1}(y_{i+1}^*),\quad i=n-1,n-2,\cdots,1$$

求得最优路径 $y^*=(y_1^*,y_2^*,\cdots,y_n^*)$．

CRFs及其应用

②解码

以中文分词为例：乒 乓 球 拍 卖 完 了

维特比算法就是在下面由标记组成的矩阵中搜索一条最优的路径。

乒	乓	球	拍	卖	完	了
B	B	B	B	B	B	B
M	M	M	M	M	M	M
E	E	E	E	E	E	E
S	S	S	S	S	S	S

分词结果：乒/**B** 乓/**M** 球/**M** 拍/**E** 卖/**S** 完/**S** 了/**S**

6.8 CRFs及其应用

关于条件随机场模型的实现工具：

- **CRF++** （**C++**版）：
  http://crfpp.googlecode.com/svn/trunk/doc/index.html
- **CRFSuite** （**C**语言版）：
  http://www.chokkan.org/software/crfsuite/
- **MALLET**（**Java**版，通用的自然语言处理工具包，包括分类、序列标注等机器学习算法）：
  http://mallet.cs.umass.edu/
- **NLTK**（**Python**版，通用的自然语言处理工具包，很多工具是从**MALLET**中包装转成的**Python**接口）：
  http://nltk.org/

6.8 CRFs及其应用

CRF与HMM性能比较

- CRF比HMM要强大，它可以解决所有HMM能够解决的问题。
- HMM最大的缺点就是由于其输出独立性假设，导致其不能考虑上下文的特征；其次，HMM中当前状态只考虑与前一状态有关(一阶马尔可夫模型)
- CRF具有表达长距离依赖的能力，可以引入更多的特征函数，因此能够求得全局的最优解。当然，模型也变复杂了。

谢谢！