第3章 传统自然语言处理分析

六、隐马尔柯夫模型

6.1 马尔柯夫模型

6.1 马尔柯夫模型

❑马尔可夫模型描述

存在一类重要的随机过程：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 表示系统在时间 t 的状态变量，那么，t 时刻的状态取值为 S_j $(1\leq j\leq N)$ 的概率取决于前 t-1 个时刻 (1, 2, …, t-1) 的状态，该概率为：

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

6.1 马尔柯夫模型

◆ **假设1：**

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

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

6.1 马尔柯夫模型

◆ **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：

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

该随机过程称为马尔柯夫模型(Markov Model)。

6.1 马尔柯夫模型

在马尔柯夫模型中，状态转移概率 a_{ij} 必须满足下列条件：

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

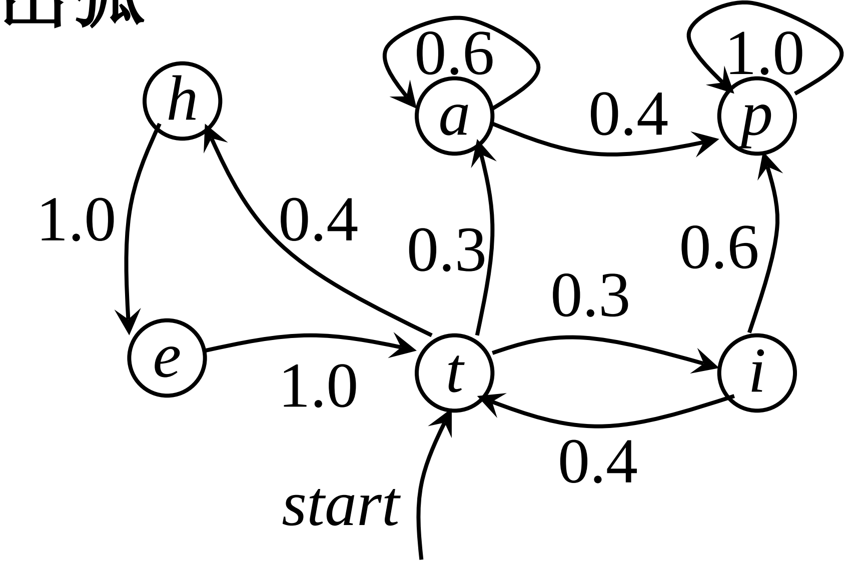
$$\sum_{j=1}^{N} a_{ij} = 1 \qquad \ldots (6.4)$$

马尔柯夫模型又可视为随机有限状态自动机，该有限状态自动机的每一个状态转换过程都有一个相应的概率，该概率表示自动机采用这一状态转换的可能性。

6.1 马尔柯夫模型

◆ 马尔柯夫链可以表示成状态图（转移弧上有概率的非确定的有限状态自动机）

- 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 马尔柯夫模型

状态序列 $S_1, \ldots, S_T$ 的概率：

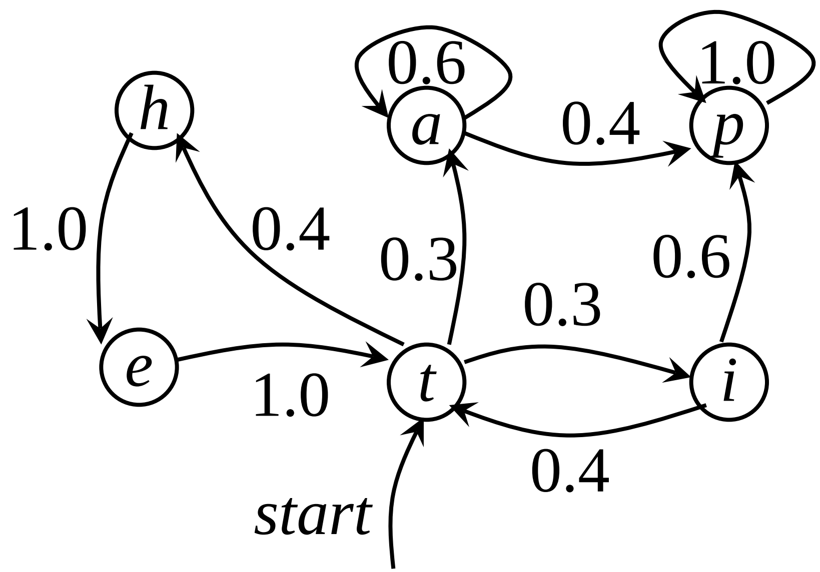
$$P(S_1,\cdots,S_T) = P(S_1)P(S_2 \mid S_1)P(S_3 \mid S_1,S_2)\cdots P(S_T \mid S_1,\cdots,S_{T-1})$$

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

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

其中，$\pi_i=P(q_1=S_i)$，为初始状态的概率。

6.1 马尔柯夫模型



$$P(t,i,p) = P(S_1 = t)P(S_2 = i \mid S_1 = t)P(S_3 = p \mid S_2 = i)$$
$$= 1.0 \times 0.3 \times 0.6$$
$$= 0.18$$

6.2 隐马尔柯夫模型

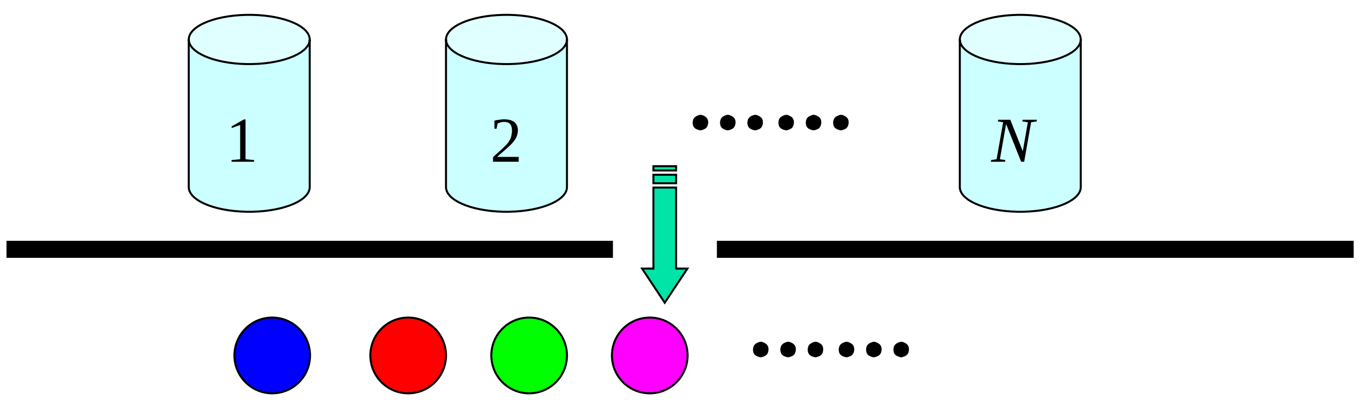
6.2 隐马尔柯夫模型

❑ 隐马尔柯夫模型
（Hidden Markov Model, HMM）

描写：该模型是一个双重随机过程，我们不知道具体的状态序列，只知道状态转移的概率，即模型的状态转换过程是不可观察的（隐蔽的），而可观察事件的随机过程是隐蔽状态转换过程的随机函数。

6.2 隐马尔柯夫模型

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，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。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。每只袋子对应 **HMM** 中的状态；球的颜色对应于 **HMM** 中的状态的输出。



6.2 隐马尔柯夫模型

- 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 M (不同颜色球的数目)

6.2 隐马尔柯夫模型

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 隐马尔柯夫模型

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。那么，

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

6.2 隐马尔柯夫模型

5. 初始状态的概率分布为：$\pi = \pi_i$，其中，

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

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 隐马尔柯夫模型

❑ 给定HMM求观察序列

给定模型 $\mu=(A,B,\pi)$，产生观察序列 $O = O_1 O_2 \cdots O_T$：

(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 隐马尔柯夫模型

- HMM 中的三个问题

(1) 在给定模型 $\mu=(A,B,\pi)$ 和观察序列 $\boldsymbol{O}=\boldsymbol{O}_1\boldsymbol{O}_2\cdots\boldsymbol{O}_T$ 的情况下，怎样快速计算概率 $P(O\mid\mu)$ ？

(2) 在给定模型 $\mu=(A,B,\pi)$ 和观察序列 $\boldsymbol{O}=\boldsymbol{O}_1\boldsymbol{O}_2\cdots\boldsymbol{O}_T$ 的情况下，如何选择在一定意义下“最优”的状态序列 $Q=q_1q_2\cdots q_T$ 使得该状态序列“最好地解释”观察序列？

6.2 隐马尔柯夫模型

(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 $\mu=(A,B,\pi)$ 的参数，使得 $P(O\mid\mu)$ 最大？

6.3 前向算法

6.3 前向算法

- 问题1：快速计算观察序列概率 $P(O \mid \mu)$

给定模型 $\mu = (A, B, \pi)$ 和观察序列 $O = O_1 O_2 \cdots O_T$，快速计算 $P(O \mid \mu)$：

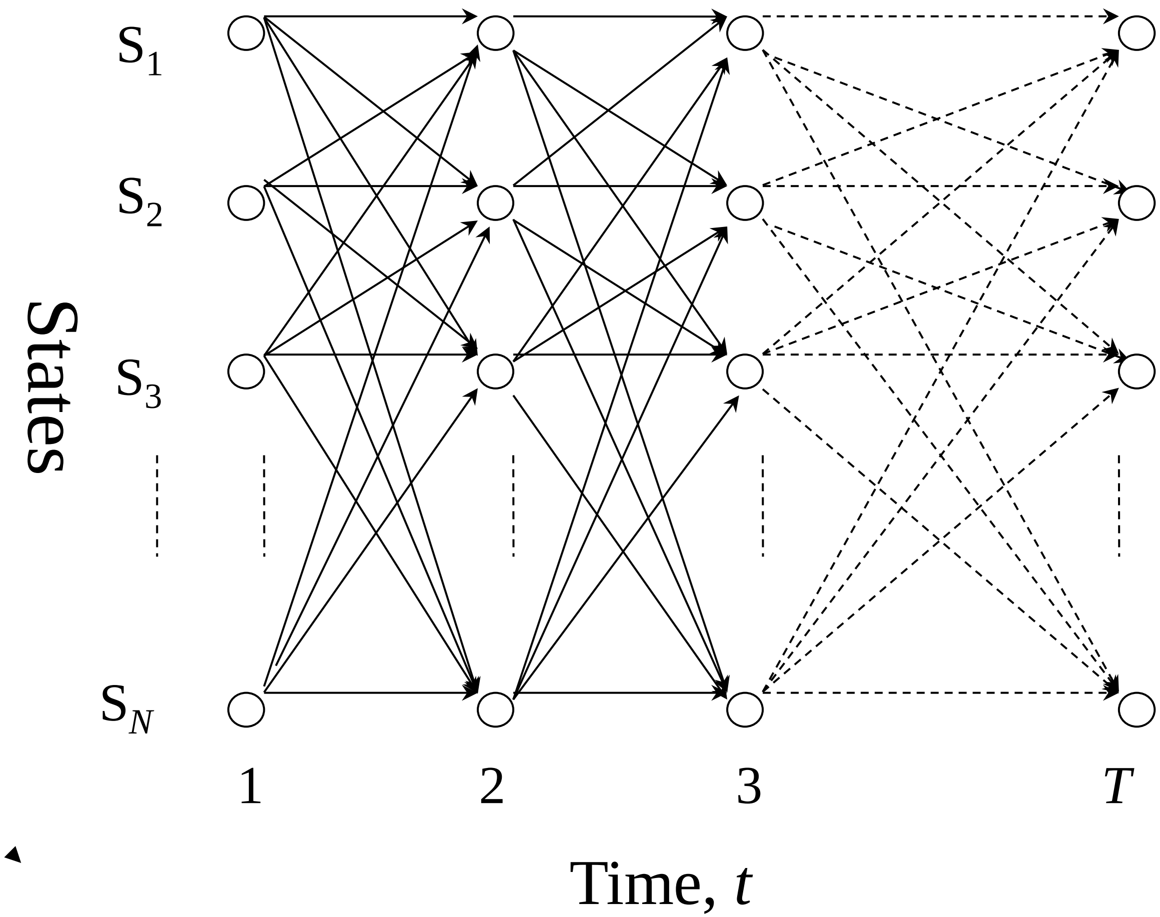
对于给定的状态序列 $Q = q_1 q_2 \cdots q_T$

$$P(O \mid \mu) = \sum_Q P(O, Q \mid \mu) = \sum_Q P(Q \mid \mu) P(O \mid Q, \mu) \quad \ldots (6.9)$$

$$P(Q \mid \mu) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{t-1} q_T} \quad \ldots (6.10)$$

$$P(O \mid Q, \mu) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T) \quad \ldots (6.11)$$

6.3 前向算法



- **困难：** 如果模型 $\mu=(A, B, \pi)$ 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$\alpha_t(i)$：

$$\alpha_t(i) = P(O_1O_2\cdots O_t, q_t = S_i \mid \mu) \qquad \ldots(6.12)$$

如果可以高效地计算$\alpha_t(i)$，就可以高效地求得$P(O \mid \mu)$。

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 $O = O_1 O_2 \ldots O_T$ 的概率(所有可能的概率之和)：

$$P(O|\mu) = \sum_{S_i} P(O_1 O_2 \cdots O_T, q_T = S_i | \mu)$$

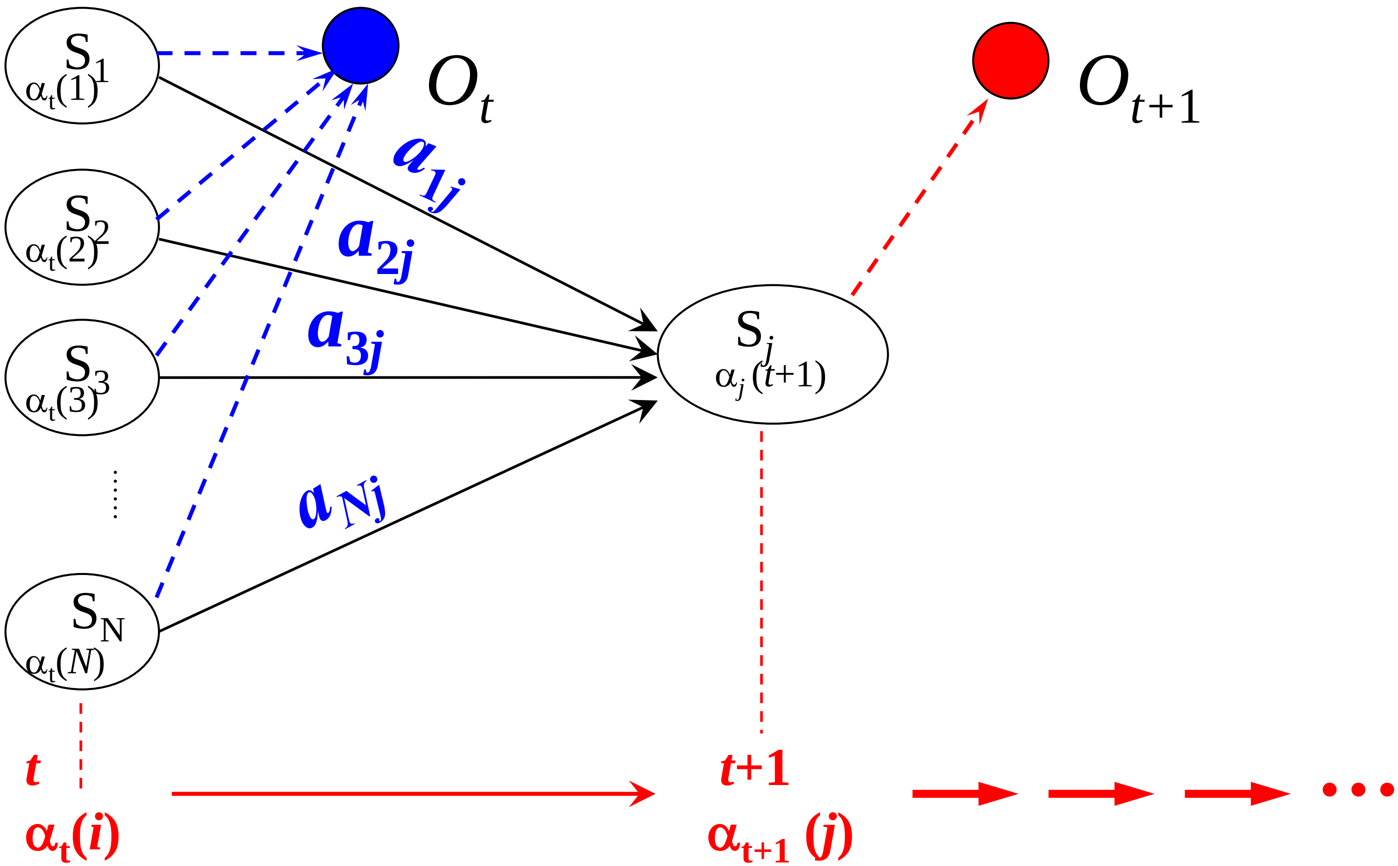
$$= \sum_{i=1}^{N} \alpha_T(i) \qquad \ldots (6.13)$$

6.3 前向算法

动态规划计算 $\alpha_t(i)$：在时间 **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：

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

6.3 前向算法



S1
αt(1)
S2
αt(2)
S3
αt(3)
SN
αt(N)
Ot
Ot+1
a1j
a2j
a3j
aNj
Sj
αj(t+1)
t
αt(i)
t+1
αt+1 (j)
• • •

6.3 前向算法

算法6.1:前向算法

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

(2) 循环计算：

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

(3) 结束，输出：

$$P(O\mid\mu)=\sum\nolimits_{i=1}^{N}\alpha_T(i)$$

6.3 前向算法

- 算法的时间复杂性：

每计算一个 $\alpha_t(i)$ 必须考虑从 t-1 时的所有 N 个状态转移到状态 S_i 的可能性，时间复杂性为 $O(N)$，对应每个时刻 t，要计算 N 个前向变量：$\alpha_t(1), \alpha_t(2), \ldots, \alpha_t(N)$，所以，时间复杂性为：$O(N) \times N = O(N^2)$。又因 $t = 1, 2, \ldots, T$，所以前向算法总的复杂性为：$O(N^2T)$。

6.4 后向算法

6.4 后向算法

◆后向算法 (The backward procedure)

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

$$\beta_t(i)=P(O_{t+1}O_{t+2}\cdots O_T \mid q_t=S_i,\mu) \qquad \ldots (6.15)$$

6.4 后向算法

与前向变量一样，运用动态规划计算后向量：

(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 $O_{t+1}O_{t+2}\cdots O_T$

6.4 后向算法

第一步的概率：$a_{ij} \times b_j(O_{t+1})$

第二步的概率按后向变量的定义为：$\beta_{t+1}(j)$

于是，有归纳关系：

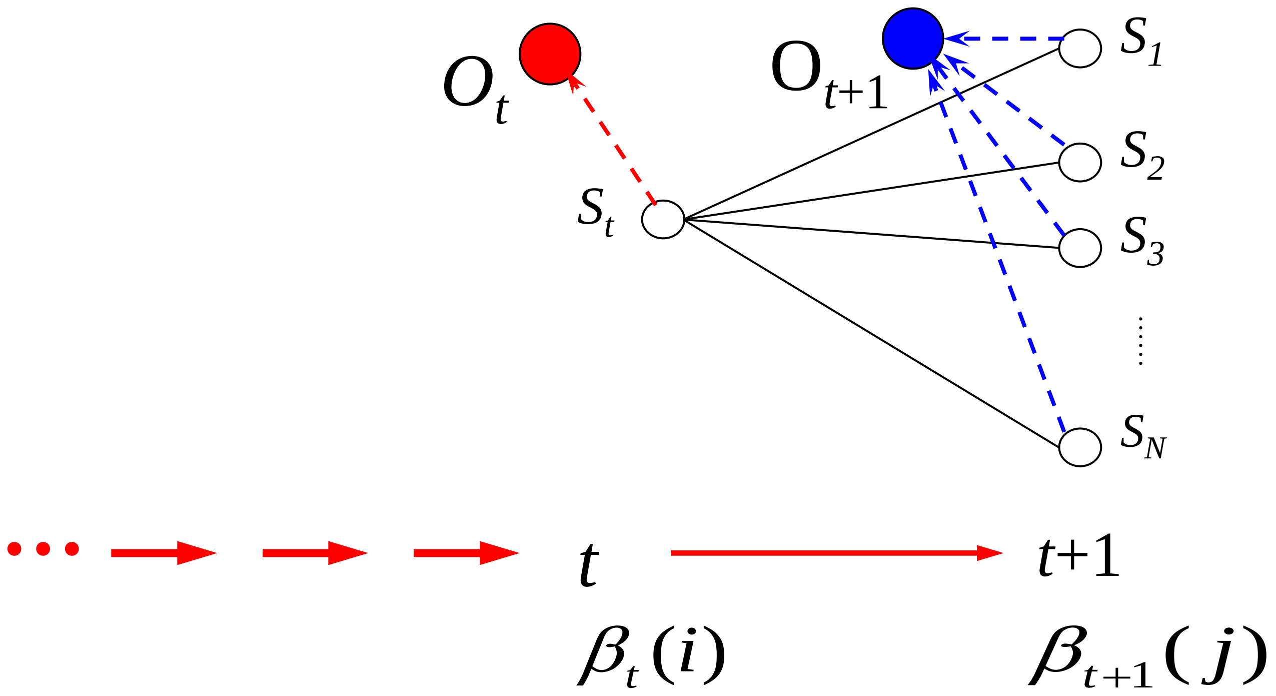
$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \qquad \ldots (6.16)$$

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

(x 为 HMM的状态)

6.4 后向算法

算法的图形解释：



6.4 后向算法

◆ 算法6.2：后向算法

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

(2) 循环计算：

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

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

算法的时间复杂性：$O(N^2 T)$

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

解释不是唯一的，关键在于如何理解“最优”的状态序列？<u>**一种解释是**</u>：状态序列中的每个状态都单独地具有概率，即：对于每个时刻 $t\ (1 \le t \le T)$，寻找 q_t 使得 $\gamma_t(i) = P(q_t = S_i \mid O, \mu)$ 最大。

6.5 Viterbi 搜索算法

$$\gamma_t(i) = P(q_t = S_i \mid O, \mu) = \frac{P(q_t = S_i, O \mid \mu)}{P(O \mid \mu)} \quad \ldots (6.17)$$

HMM 的输出序列 *O*，并且在时间 *t* 到达状态 *i* 的概率。

6.5 Viterbi 搜索算法

分解过程：

（1）**HMM** 在时间 t 到达状态 i, 并且输出

$O_1O_2\cdots O_t$ 根据前向变量的定义，实现

这一步的概率为：$\alpha_t(i)$。

（2）从时间 t，状态 S_i 出发，**HMM** 输出

$O_{t+1}O_{t+2}\cdots O_T$，根据后向变量定义，实现

这一步的概率为 $\beta_t(i)$。

于是：

$$P(q_t = S_i, O \mid \mu) = \alpha_t(i) \times \beta_t(i) \qquad \ldots (6.18)$$

6.5 Viterbi 搜索算法

而 $P(O|\mu)$ 与时间 *t* 的状态无关，因此：

$$P(O|\mu)=\sum_{i=1}^{N}\alpha_t(i)\times\beta_t(i) \qquad \ldots (6.19)$$

将公式(6.18)和(6.19)带入(6.17)式得：

$$\gamma_t(i)=\frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N}\alpha_t(i)\times\beta_t(i)} \qquad \ldots (6.20)$$

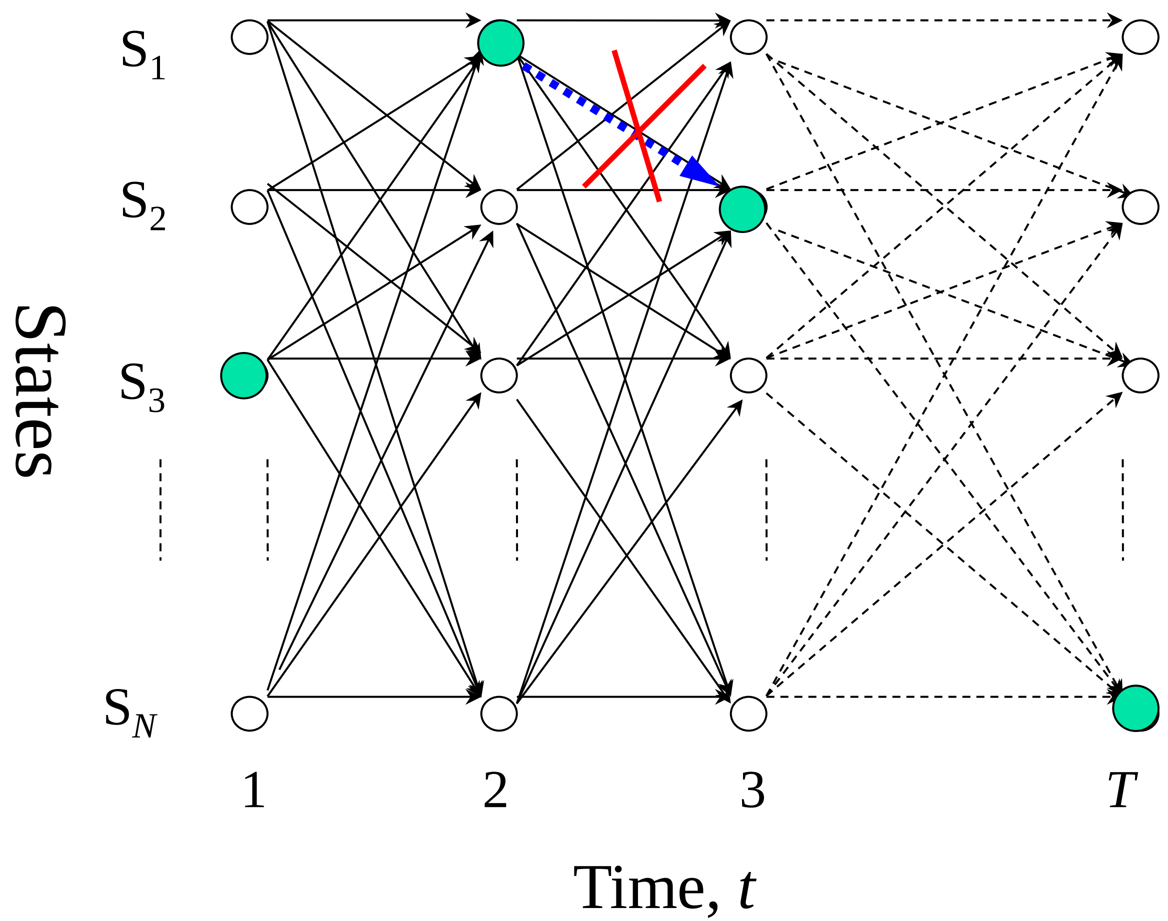
t 时刻的最优状态为： $\hat{q}_t=\underset{1\le i\le N}{\arg\max}(\gamma_t(i))$

6.5 Viterbi 搜索算法

问题：

每一个状态单独最优不一定使整体的状态序列最优，可能两个最优的状态 $\widehat{q}_t$ 和 $\widehat{q}_{t+1}$ 之间的转移概率为0，即 $a_{\widehat{q}_t \widehat{q}_{t+1}} = 0$ 。

6.5 Viterbi 搜索算法



6.5 Viterbi 搜索算法

另一种解释：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：

$$\widehat{Q} = \arg\max_{Q} P(Q \mid O, \mu) \quad \ldots (6.21)$$

Viterbi algorithm: 动态搜索最优状态序列。

定义：**Viterbi** 变量 $\delta_t(i)$是在时间 t 时，**HMM** 沿着某一条路径到达 S_i，并输出观察序列

$O_1 O_2 \cdots O_t$ 的最大概率：

$$\delta_t(i) = \max_{q_1, q_2, \cdots, q_{t-1}} P(q_1, q_2, \cdots, q_t = S_i, O_1 O_2 \cdots O_t \mid \mu) \ldots (6.22)$$

6.5 Viterbi 搜索算法

<u>递归计算</u>： $\delta_{t+1}(i)=\max_{j}[\delta_t(j)\cdot a_{ji}]\cdot b_i(O_{t+1})$ … (6.23)

<u>算法6.3</u>：**Viterbi 算法**

（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：

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

（4）通过回溯得到路径（状态序列）：

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

算法的时间复杂性： $O(N^2 T)$

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 $O = O_1O_2 \ldots O_T$，如何根据最大似然估计来求模型的参数值？即如何调节模型 $\mu=(A, B, \pi)$ 的参数，使得$P(O|\mu)$ 最大？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。

<u>前向后向算法</u>

(Baum-Welch or forward-backward procedure)

6.6 参数学习

如果产生观察序列 O 的状态 $Q = q_1q_2\ldots q_T$ 已知，可以用最大似然估计来计算 HMM 的参数：

$$\bar{\pi}_i = \delta(q_1, S_i)$$

$$\bar{a}_{ij} = \frac{Q□□□□ q_i □□□ q_j □□□}{Q□□□□□□ q_i □□□□□□□ (□□ q_j □□) □□□}$$

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

其中，$\delta(x, y)$ 为克罗奈克(Kronecker)函数，当 $x= y$ 时，$\delta(x, y)$=1, 否则$\delta(x, y)$ = 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

期望值最大化算法 (Expectation-Maximization, EM)

基本思想：初始化时随机地给模型的参数赋值(遵循限制规则，如：从某一状态出发的转移概率总和为 1，得到模型μ_0, 然后可以从μ_0 得到从某一状态转移到另一状态的期望次数，然后以期望次数代替公式中的实际次数，得到模型参数的新估计，由此得到新的模型μ_1，从 μ_1 又可得到模型中隐变量的期望值，由此重新估计模型参数。循环这一过程，参数收敛于最大似然估计值。

6.6 参数学习

给定 HMM 模型 μ 和观察序列 $O = O_1 O_2 \cdots O_T$，那么，在时间 t 位于状态 S_i，时间 $t+1$ 位于状态 S_j 的概率：

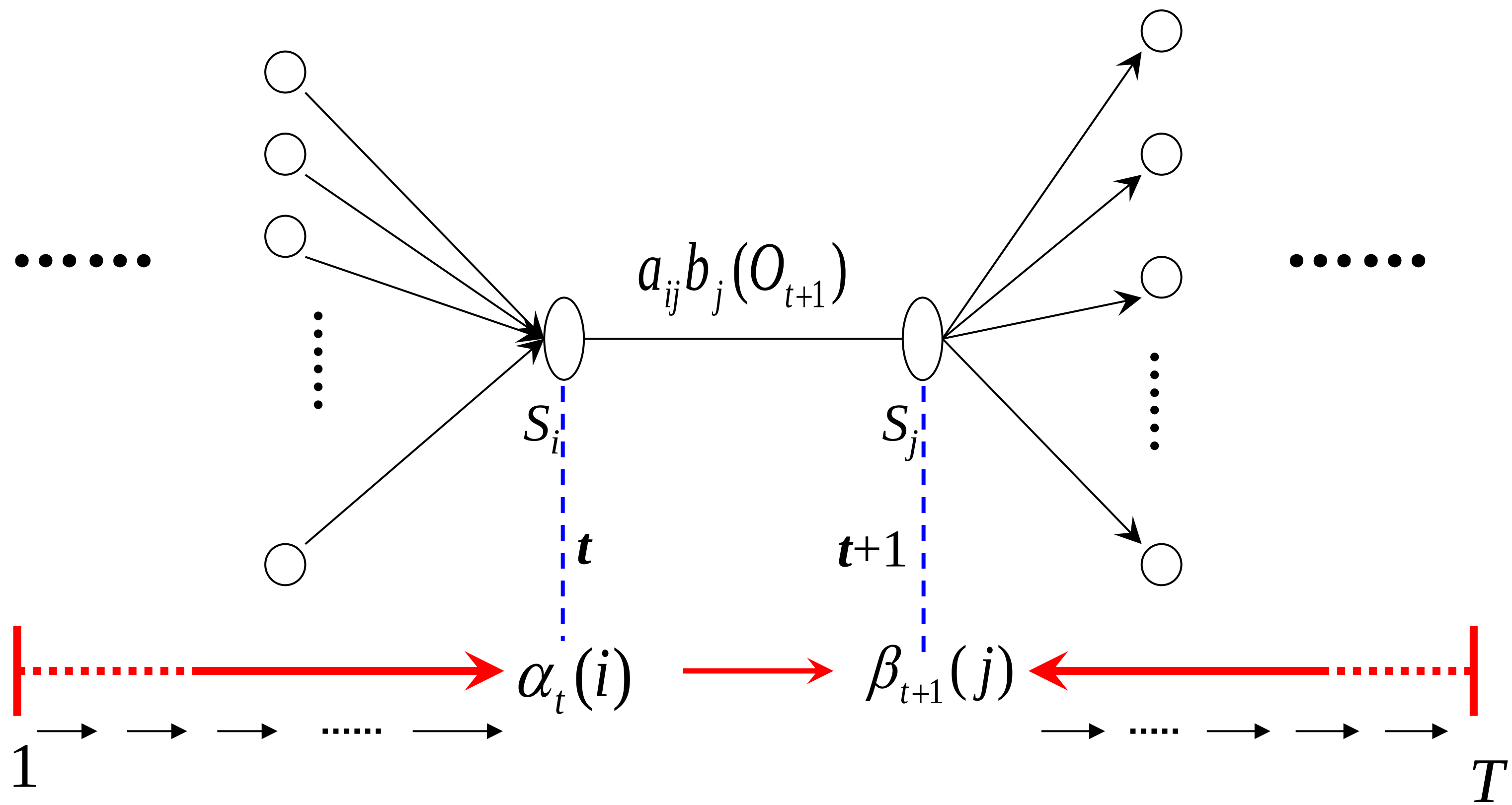
$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \mu) = \frac{P(q_t = S_i, q_{t+1} = S_j, O \mid \mu)}{P(O \mid \mu)}$$

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

$$= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)} \quad \ldots (6.26)$$

6.6 参数学习

图解搜索过程：



6.6 参数学习

那么，给定模型 μ 和观察序列 $O = O_1 O_2 \cdots O_T$，在时间 t 位于状态 S_i 的概率为：

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

由此，模型 μ 的参数可由下面的公式重新估计：

(1) q_1 为 S_i 的概率：

$$\pi_i = \gamma_1(i) \qquad \ldots (6.28)$$

6.6 参数学习

(2)

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

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

(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:

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

使得

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

由此得到模型μ_0，令 $i = 0$。

6.6 参数学习

(2) 执行 EM 算法:

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: $|\log P(O \mid \mu_{i+1}) - \log P(O \mid \mu_i)| < \varepsilon$ 。

(3) 结束算法, 获得相应的参数。

6.6 参数学习

❑ HMM 使用中注意的问题

◆ **Viterbi** 算法运算中的小数连乘，出现溢出

—对数

◆ **Forward-Backward** 算法的小数溢出

—放大系数

—参阅[Rabiner and Juang, 1993: pp. 365-368]

—参阅 **http://htk.eng.cam.ac.uk/**

6.7 应用举例

6.7 应用举例

汉语的自动分词与词性标注问题：

自动化研究所取得的成绩

自动化/N 研究所/N 取得/V 的/X 成绩/N

自动化/N 研究/N 所/P 取得/V 的/X 成绩/N
/V

6.7 应用举例

用 HMM 来解决这一问题：

(1) 状态转移模型

(2) 状态到观察序列的生成模型

思路：

(1) 汉字串(句子) 作为输入；单词串 S_w 为状态的输出，即观察序列 $S_w = w_1 w_2 \cdots w_n \quad (n \geq 1)$；

(2) 词性序列 S_c 为状态序列，每个词性标记 C_i 对应HMM中的一个状态：$S_c = c_1 c_2 \cdots c_n$。

6.7 应用举例

问题：

(1) 估计模型的参数；

(2) 对于一个给定的输入句子及其可能的输出序列S_w和模型 $\mu=(A,B,\pi)$，快速计算 $P(S_w|\mu)$ 。所有可能的S_w中使概率最大的解就是要找的分词结果；

(3) 快速地选择“最优”的状态序列(词性序列)。

6.7 应用举例

假设模型中状态(词性)的数目为词性符号的个数*N*；从每个状态可能输出的不同符号(单词)的数目为词汇的个数*M*。

假设在统计意义上每个词性的概率分布只与上一个词的词性有关(即词性的二元语法)，而每个单词的概率分布只与其词性相关。那么，我们就可以通过对已分词并做了词性标注的训练语料进行统计，得到如下三个矩阵：

6.7 应用举例

(1) 初始状态(词性)的概率分布矩阵；

(2) 状态转移(词性到词性的转移)概率矩阵；

(3) 从状态(词性)观察到输出符号(单词)的概率分布矩阵。

对于任何一个给定的观察值序列(单词串)，总可以通过Viterbi算法很快地得到一个可能性最大的状态值序列(词性串)。

本章小结

- HMM 的构成：
  - 状态数
  - 输出符号数
  - 初始状态的概率分布
  - 状态转移的概率
  - 输出概率

本章小结

- HMM 的三个基本问题：
  1) 快速计算给定模型的观察序列的概率
     －前向算法或后向算法
  2) 求最优状态序列
     －Viterbi 算法
  3) HMM 中的参数估计
     －Baum-Welch (前向-后向)算法
- 模型实现中需要注意的问题：小数溢出

习题

1. 下载 HTK (http://htk.eng.cam.ac.uk/)，了解相应工具的使用方法。
2. 利用HTK工具，实现一个简单的汉语音字转换程序或汉语分词与词性标注程序。

Thanks

谢谢!