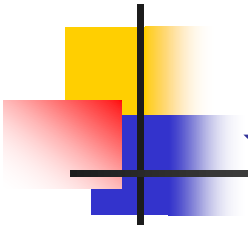
隐马尔可夫模型

刘秉权
哈工大智能技术与自然语言处理研究室
新技术楼612房间
liubq@hit.edu.cn



主要内容

- 马尔可夫模型
- 隐马尔可夫模型
- 隐马尔可夫模型的三个基本问题
- 隐马尔可夫模型的基本算法
- 隐马尔可夫模型的应用

马尔可夫链

一个系统有 N 个状态 $S_1, S_2, \cdots, S_N$，随着时间推移，系统从某一状态转移到另一状态，设 q_t 为时间 t 的状态，系统在时间 t 处于状态 S_j 的概率取决于其在时间 $1,2,\cdots,t-1$ 的状态，该概率为：

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

如果系统在 t 时间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可夫链(马尔可夫过程)：

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

马尔可夫模型(Markov Model)

如果只考虑独立于时间 t 的随机过程：

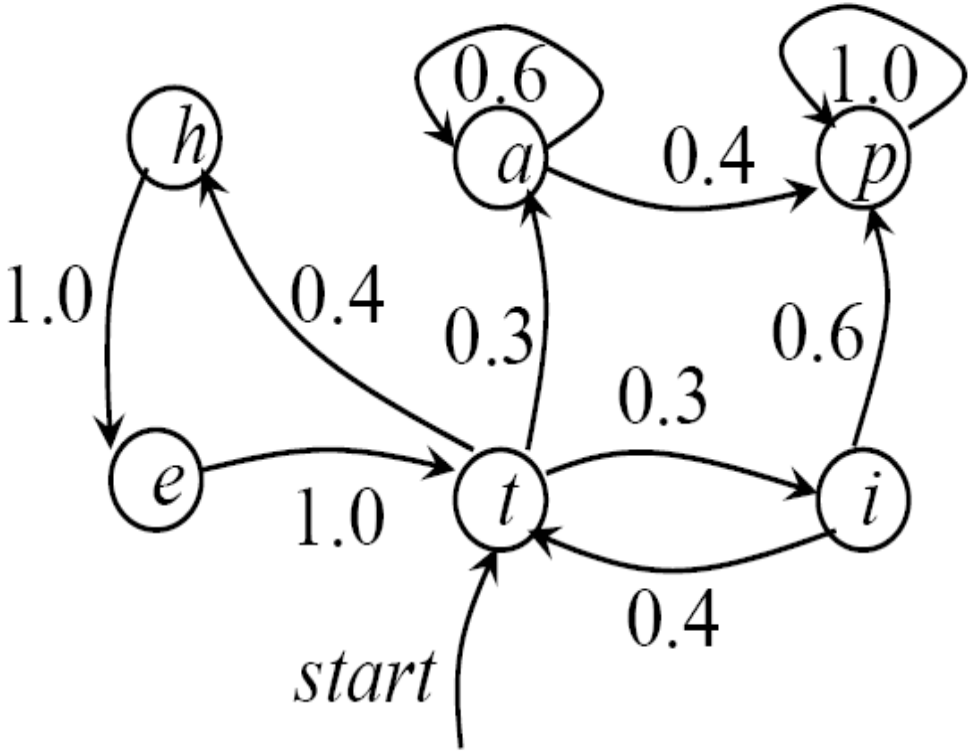
$$P(q_t = S_j \mid q_{t-1} = S_i) = a_{i,j}, 1 \le i, j \le N$$

其中状态转移概率 $a_{i,j}$ 必须满足 $a_{i,j} \ge 0$ 且

$\sum_{j=1}^{N} a_{i,j} = 1$，则该随机过程称为马尔可夫模型。

马尔可夫模型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都有一相应概率，表示自动机采用这一状态转换的可能性



例

假定一段时间内的气象可由一三状态马尔可夫模型 M 描述：S_1：雨，S_2：多云，S_3：晴，转移概率矩阵为：

$$A=[a_{ij}]=\begin{vmatrix}0.4 & 0.3 & 0.3\\ 0.2 & 0.6 & 0.2\\ 0.1 & 0.1 & 0.8\end{vmatrix}$$

每一行都是和为1

例（续）

如果第一天为晴天，根据这一模型，在今后七天中天气为

O ="晴晴雨雨晴云晴"的概率为：

$$P(O \mid M)$$
$$= P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 \mid M)$$
$$= P(S_3) \cdot P(S_3 \mid S_3) \cdot P(S_3 \mid S_3) \cdot P(S_1 \mid S_3) \cdot P(S_1 \mid S_1) \cdot P(S_3 \mid S_1) \cdot P(S_2 \mid S_3) \cdot P(S_3 \mid S_2)$$
$$= 1 \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23}$$
$$= (0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)$$
$$= 1.536 \times 10^{-4}$$

第一天

后面7天

隐马尔可夫模型 (Hidden Markov Model, HMM)

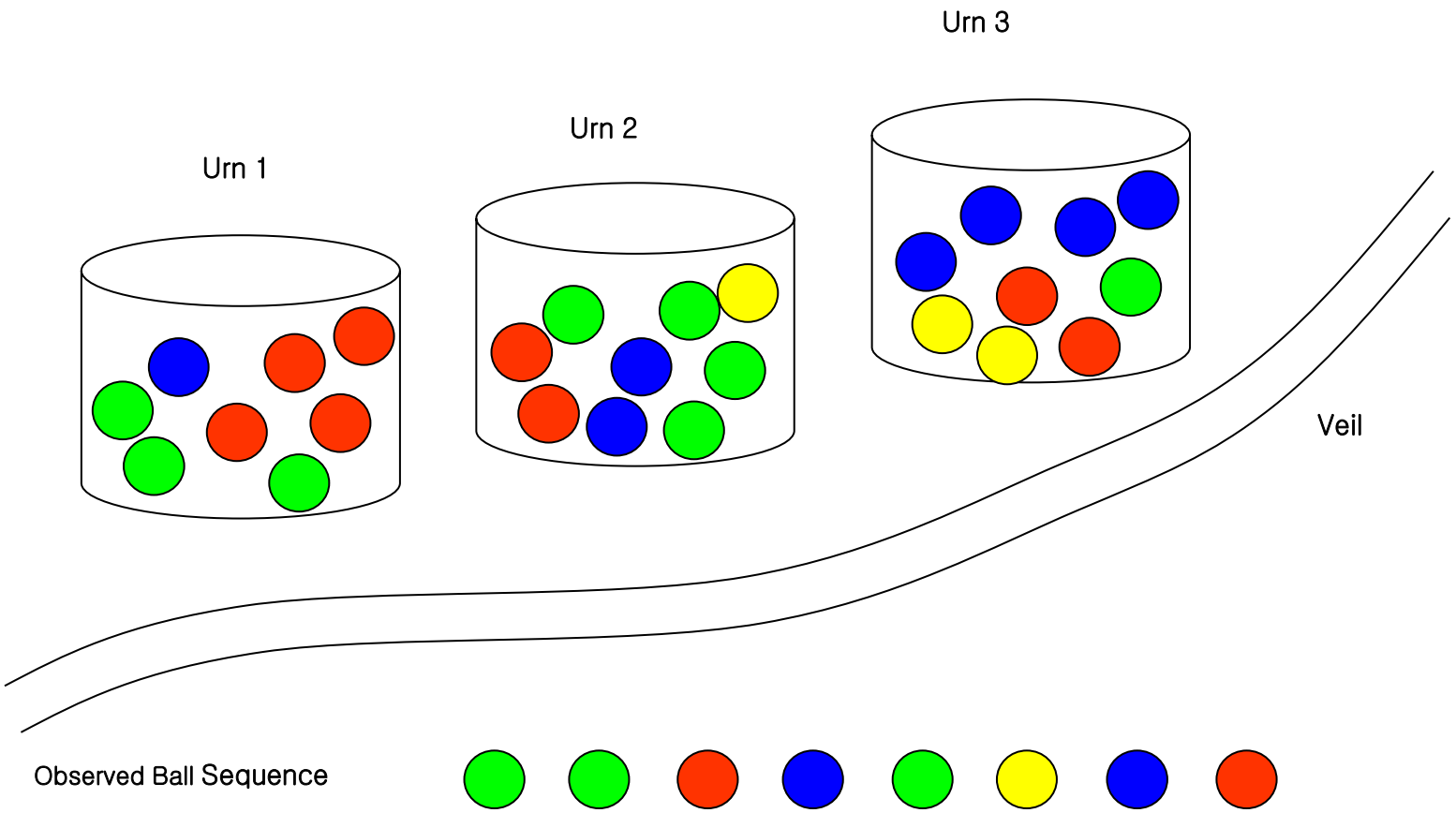
- 在MM中，每一个状态代表一个可观察的事件
- HMM模型是一个双重随机过程
  - 状态转移过程是不可观察(隐蔽)的(马尔可夫链)
  - 可观察的事件的随机过程是隐蔽的状态转换过程的随机函数(一般随机过程)

实例

一房间有 N 只瓮，每只瓮中有 M 种不同颜色的球。根据某一概率分布随机地选择一个初始瓮，根据不同颜色球的概率分布从中随机取出一个球，并报告球的颜色。然后根据某一概率分布随机地选择另一只瓮，再根据不同颜色球的概率分布从中随机取出一个球，并报告球的颜色，…。对房间外的观察者，可观察的过程是不同颜色球的序列，而瓮的序列是不可观察的。

这里每只瓮对应 HMM 模型中的状态，球的颜色对应于状态的输出符号，从一只瓮转向另一只瓮对应于状态转换，从一只瓮中取球对应于从一状态输出观察符号。

实例（续）



实验中的几个要点

- 不能直接观察瓮间的转移
- 从瓮中所选取的球的颜色和瓮并不是一一对应的
- 每次选取哪个瓮由一组转移概率决定

HMM的组成

五元组： $\lambda = (N, M, A, B, \pi)$

简记为： $\lambda = (A, B, \pi)$

N ： 状态数目

M ： 可能的观察值数目

A ： 与时间无关的状态转移概率矩阵 N*N阶

B ： 给定状态下，观察值概率分布 N*M阶

π ： 初始状态空间的概率分布 N维vector

状态转移概率矩阵

$$A = a_{ij}$$

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

$$a_{ij} \ge 0, \quad \sum_{j=1}^{N} a_{ij} = 1$$

观察值概率分布矩阵

从状态 S_j 观察到符号 v_k 的概率分布矩阵：

$$B = b_j(k)$$

$$b_j(k) = P(O_t = v_k \mid q_t = S_j)，\ 1 \le j \le N，\ 1 \le k \le M$$

$$b_j(k) \ge 0，\ \sum_{k=1}^{M} b_j(k) = 1$$

初始状态概率分布

$$\pi = \pi_i$$

$$\pi_i = P(q_1 = S_i)\,,\quad 1 \le i \le N$$

$$\pi_i \ge 0\,,\quad \sum_{i=1}^{N} \pi_i = 1$$

观察序列产生步骤

给定模型 $\lambda = (A, B, \pi)$，观察序列 $O = O_1, O_2, \cdots, O_T$ 可由以下步骤产生：

1.根据初始状态概率分布 $\pi = \pi_i$ 选择一初始状态 $q_1 = S_i$；

2.设 $t = 1$；

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

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

5.设 $t = t + 1$，如果 $t < T$，重复步骤 3、4，否则结束。

HMM中的三个基本问题

问题 1：给定观察序列 $O=O_1,O_2,\cdots,O_T$，以及模型 $\lambda=(A,B,\pi)$，如何计算 $P(O\mid\lambda)$？

问题 2：给定观察序列 $O=O_1,O_2,\cdots,O_T$ 及模型 $\lambda=(A,B,\pi)$，如何选择一个对应的状态序列 $S=q_1,q_2,\cdots,q_T$，使得 S 能够最为合理地解释观察序列 O？

问题 3：如何调整模型参数 $\lambda=(A,B,\pi)$，使得 $P(O\mid\lambda)$ 最大？

解决问题1

直接计算：

$$P(O|\lambda)=\sum_{Q}P(O,Q|\lambda)=\sum_{Q}P(Q|\lambda)P(O|Q,\lambda)$$

其中：$O=O_1,O_2,\cdots,O_T$，$Q=q_1,q_2,\cdots,q_T$

$$P(Q|\lambda)=\pi_{q_1}a_{q_1q_2}a_{q_2q_3}\cdots a_{q_{T-1}q_T}$$

$$P(O|Q,\lambda)=b_{q_1}(O_1)b_{q_2}(O_2)\cdots b_{q_T}(O_T)$$

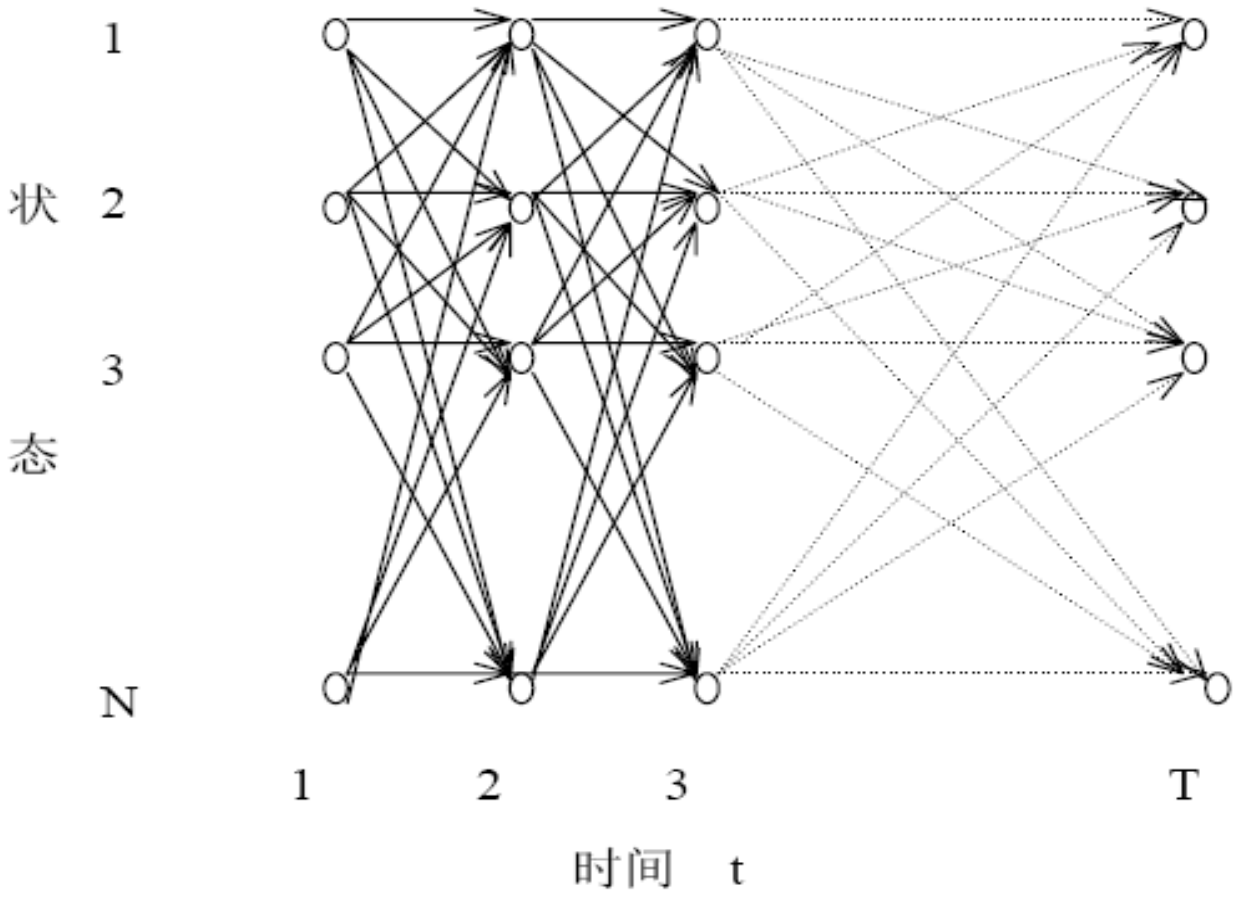
困难：穷尽所有可能的状态序列，复杂度$O(N^T)$，指数爆炸。

有效方法：向前算法，动态规划，复杂性$O(N^2T)$。

动态规划(Dynamic Programming)

- 动态规划是运筹学的一个分支，是求解决策过程(decision process)最优化的数学方法。
- 20世纪50年代初美国数学家R.E.Bellman等人在研究多阶段决策过程(multistep decision process)的优化问题时，提出了著名的最优化原理(principle of optimality)，把多阶段过程转化为一系列单阶段问题，利用各阶段之间的关系，逐个求解，创立了解决这类过程优化问题的新方法——动态规划。
- 1957年出版了他的名著Dynamic Programming，是该领域的第一本著作。

HMM的网格结构



向前算法

思想：高效递归地计算向前变量，以求得最终结果

向前变量：$\alpha_t(i) = P(O_1 O_2 \cdots O_t, q_t = S_i \mid \lambda)$，$1 \le t \le T$

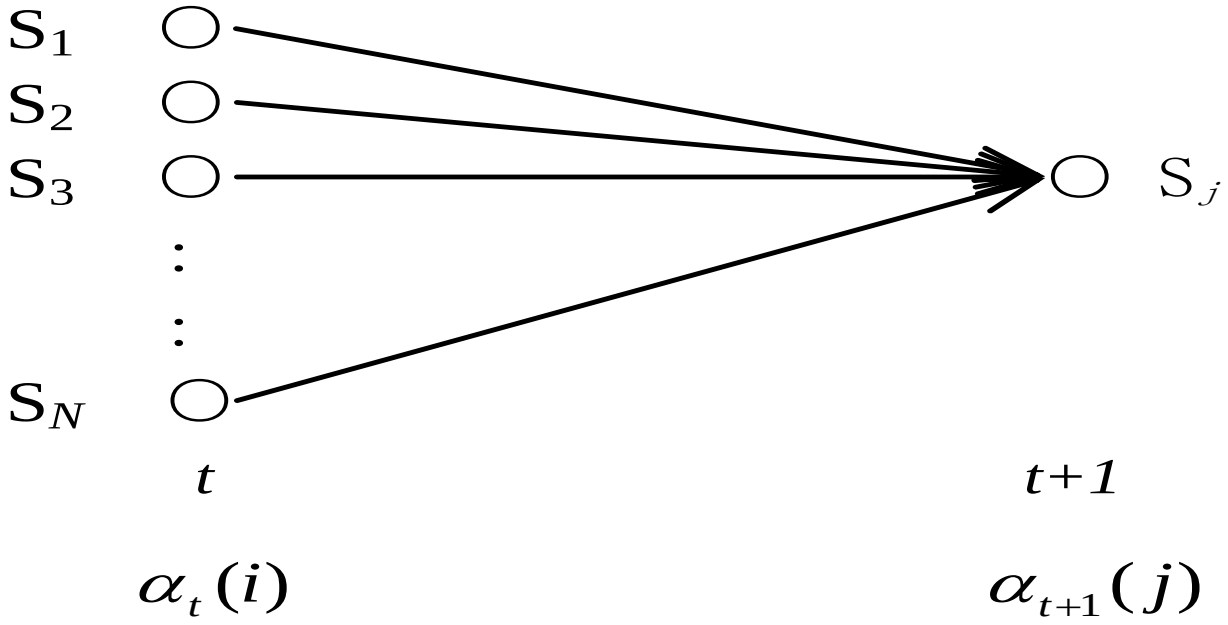
算法：

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

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

3.终结：$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$

向前变量图示



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

状态转变

S_{j}到O_{t+1}

向后算法

思想：与向前算法类似，可用于解决问题 1,3

向后变量：$\beta_t(i) = P(O_{t+1}O_{t+2}\cdots O_T, q_t = S_i \mid \lambda)$，$1 \le t \le T-1$

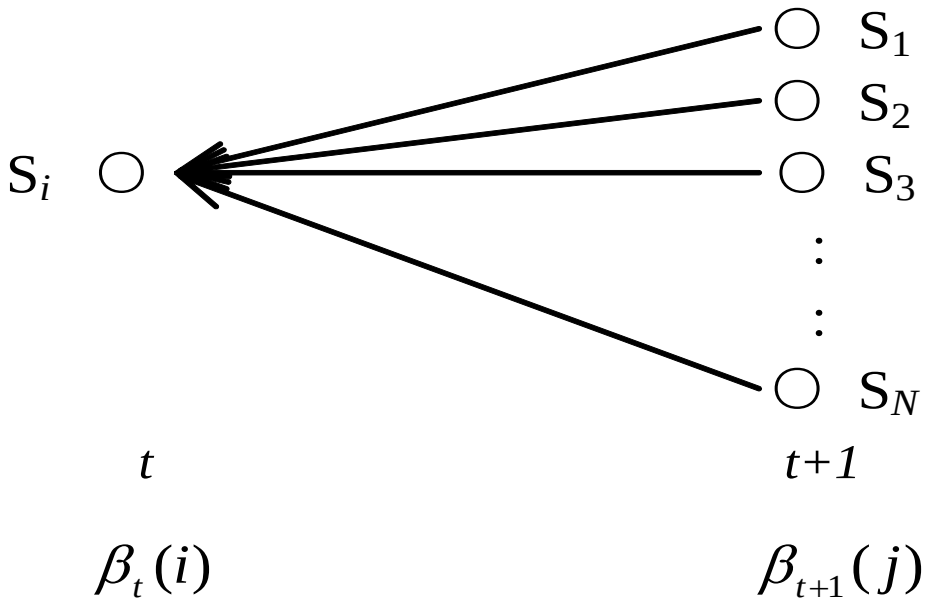
算法：

1.出始化：$\beta_T(i) = 1$，$1 \le i \le N$

2.递归：$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$，$1 \le t \le T-1, 1 \le i \le N$

3.终结：$P(O \mid \lambda) = \sum_{i=1}^{N} \beta_1(i)$

向后变量图示



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

解决问题2：Viterbi算法

目标：给定一个观察序列和 HMM 模型，如何有效选择“最优”状态序列，以“最好地解释”观察序列？

“最优”→概率最大：$Q^* = \arg\max_Q P(Q \mid O, \lambda)$

思想：利用动态规划求解，复杂性$O(N^2T)$

Viterbi 变量：$\delta_t(i) = \max_{q_1,q_2,\cdots,q_{t-1}} P(q_1, q_2, \cdots, q_t = S_i, O_1, O_2 \cdots O_t \mid \lambda)$

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

记忆变量：$\varphi_t(i)$ 纪录概率最大路径上当前状态的前一个状态

Viterbi算法

初始化：$\delta_1(i) = \pi_i b_i(O_1)$，$\varphi_1(i) = 0$，$1 \le i \le N$

递归：$\delta_t(i) = [\max_{1 \le j \le N} \delta_{t-1}(j) a_{ji}] b_i(O_t)$，$2 \le t \le T, 1 \le i \le N$

$$\varphi_t(i) = \arg\max_{1 \le j \le N} [\delta_{t-1}(j) a_{ji}] b_i(O_t)，\quad 2 \le t \le T, 1 \le i \le N$$

终结：$p^* = \max_{1 \le i \le N} [\delta_T(i)])$ ，$q_T^* = \arg\max_{1 \le i \le N} [\delta_T(i)]$

路径回溯：$q_t^* = \varphi_{t+1}(q_{t+1}^*)$，$t = T-1, T-2, \ldots, 1$

解决问题3：HMM参数估计

给定观察序列 $O=O_1,O_2,\cdots,O_T$ 作为训练数据，参数估计的目的是估计模型 λ 中的 π_i，a_{ij}，$b_j(k)$，使得观察序列 O 的概率 $P(O|\lambda)$ 最大。

状态序列已知情况

可以由最大似然估计来估计 HMM 的参数：

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

$$\hat{a}_{ij} = \frac{Q\text{中从状态}S_i\text{转移到}S_j\text{的次数}}{Q\text{中从状态}S_i\text{转移到另一状态(包括}S_i\text{本身)的次数}} = \frac{\sum_{t=1}^{T-1}\delta(q_t, S_i)\times\delta(q_{t+1}, S_j)}{\sum_{t=1}^{T-1}\delta(q_t, S_i)}$$

$$\hat{b}_j(k) = \frac{Q\text{中由状态}S_j\text{输出}v_k\text{的次数}}{Q\text{到达}S_j\text{的次数}} = \frac{\sum_{t=1}^{T}\delta(q_t, S_j)\times\delta(O_t, v_k)}{\sum_{t=1}^{T}\delta(q_t, S_j)}$$

其中，$\delta(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases}$

状态序列未知情况

- 由于HMM中的状态序列是观察不到的(隐变量)，以上的最大似然估计不可行。
- EM(Expectation-Maximization)算法可用于含有隐变量的统计模型的最大似然估计。
- EM算法是一个由交替进行的“期望(E过程)”和“极大似然估计(M过程)”两部分组成的迭代过程：
  - 给定不完全数据和当前的参数值，“E过程”从条件期望中相应地构造完全数据的似然函数值，
  - “M过程”则利用参数的充分统计量，重新估计概率模型的参数，使得训练数据的对数似然最大。
- EM算法的每一次迭代过程必定单调地增加训练数据的对数似然值，于是迭代过程渐进地收敛于一个局部最优值。

向前向后算法(Baum-Welch算法)

1.初始化：随机地给 π_i， a_{ij}， $b_j(k)$ 赋值（满足概率条件），得到模型 λ_0，设 $i=0$。

2.EM 步骤：

E 步骤：由 λ_i 根据公式（1）和（2），计算期望值 $\xi_t(i,j)$ 和 $\gamma_t(i)$。

M 步骤：用 E 步骤所得的期望值，根据公式（3）重新估计 π_i， a_{ij}，$b_j(k)$，得到模型 λ_{i+1}。

3.循环设计：$i=i+1$；重复 EM 步骤，直至 π_i， a_{ij}， $b_j(k)$ 值收敛。

期望值(1)

给定 HMM 和观察序列，在时间 t 位于状态 i，时间 $t+1$ 位于状态 j 的概率：

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

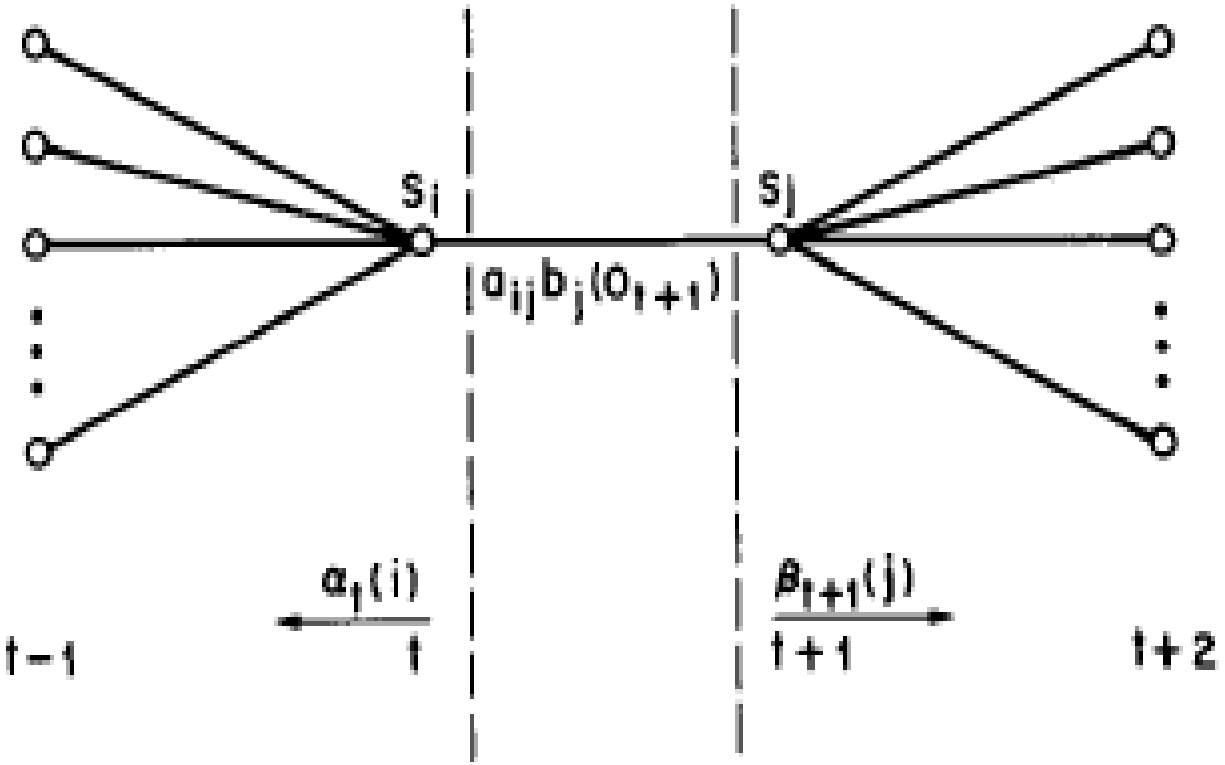
$$= \frac{P(q_t = S_i, q_{t+1} = S_j, O \mid \lambda)}{P(O \mid \lambda)}$$

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

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

图示

此图很重要



期望值(2)

给定 HMM 和观测序列，

在时间 t 位于状态 i 的概率：

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

重估公式(3)

$$\pi_i = q_1 \text{为} S_i \text{的概率} = \gamma_1(i)$$

$$a_{ij} = \frac{Q\text{中从状态}S_i\text{转移到}S_j\text{的期望次数}}{Q\text{中从状态}S_i\text{转移到另一状态(包括}S_i\text{本身)的期望次数}} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}$$

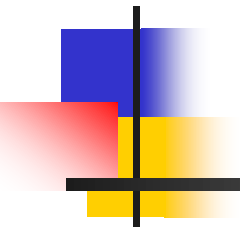
$$b_j(k) = \frac{Q\text{中由状态}S_j\text{输出}v_k\text{的期望次数}}{Q\text{到达}S_j\text{的期望次数}} = \frac{\sum_{t=1}^{T}\gamma_t(j)\times\delta(O_t,v_k)}{\sum_{t=1}^{T}\gamma_t(j)}$$

HMM的应用领域

- 语音识别
- 语言处理
- 机器视觉
  - 人脸检测
  - 机器人足球
- 图像处理
  - 图像去噪
  - 图像识别
- 生物医学分析
  - DNA/蛋白质序列分析

在NLP中的应用

- 词性标注(POS Tagging)
- 名实体识别(NER)
- 基于类的N-gram模型
- 线性插值HMM语言模型
- ……



谢谢！