Tsinghua University

第十章
隐马尔科夫模型

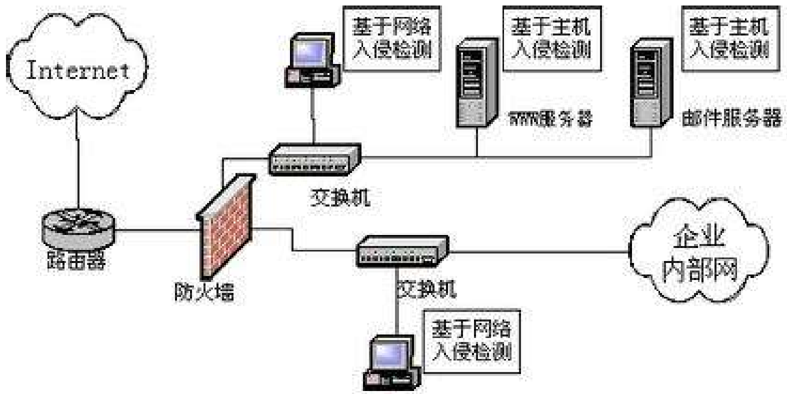
Tsinghua University

Andrey Markov

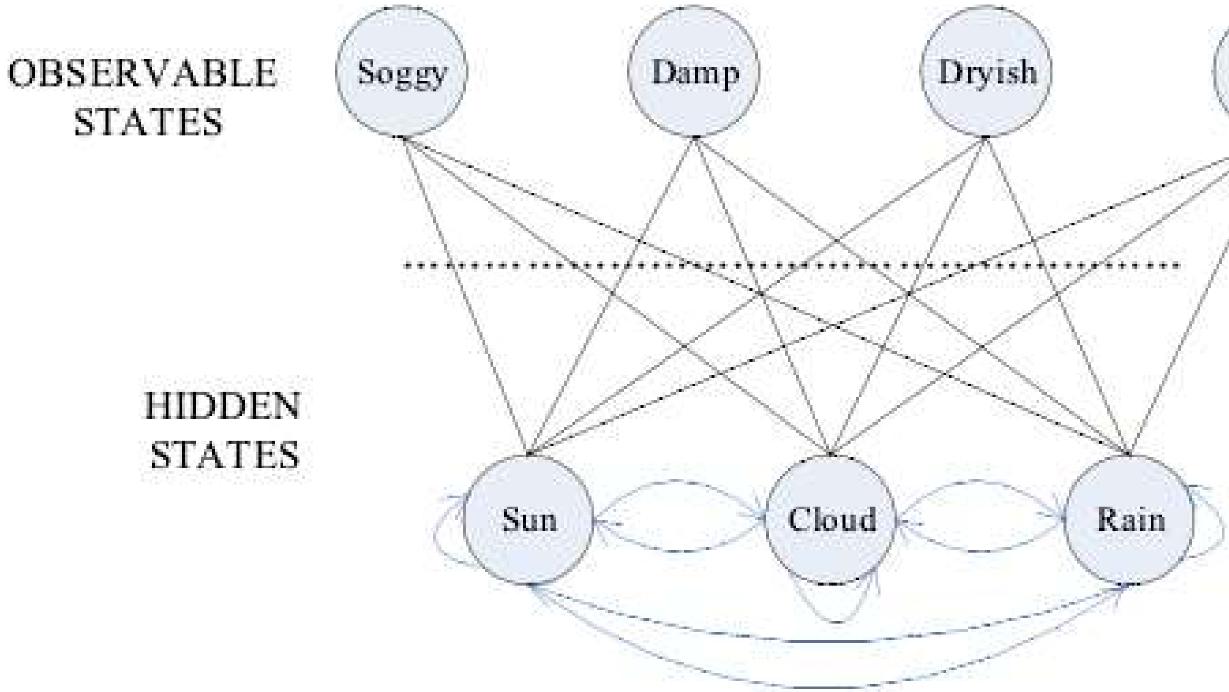
- 中文名 马尔科夫
- 国　籍 俄国
- 出生地 梁赞
- 出生日期 1856年6月14日
- 逝世日期 1922年7月20日
- 主要成就 开创了随机过程这个新领域

HMM应用

- 人脸识别
- 语音识别
- 入侵检测



例:


OBSERVABLE STATES
Soggy
Damp
Dryish
Dry
HIDDEN STATES
Sun
Cloud
Rain

隐马尔科夫模型的定义

- 隐马尔可夫模型是关于时序的概率模型;
- 描述由一个隐藏的马尔可夫链随机生成不可观测的状态随机序列(state sequence)，再由各个状态生成一个观测而产生观测随机序列(observation sequence)的过程,序列的每一个位置又可以看作是一个时刻。

Tsinghua University

隐马尔科夫模型

- 组成
  - 初始概率分布
  - 状态转移概率分布
  - 观测概率分布
  - Q：所有可能状态的集合
  - V：所有可能观测的集合

$$Q=\{q_1,q_2,\cdots,q_N\}, \quad V=\{v_1,v_2,\cdots,v_M\}$$

  - I: 长度为T的状态序列
  - O：对应的观测序列

$$I=(i_1,i_2,\cdots,i_T), \quad O=(o_1,o_2,\cdots,o_T)$$

隐马尔科夫模型

- 组成
  - A：状态转移概率矩阵

$$A=\left[a_{ij}\right]_{N\times N}$$

$$a_{ij}=P(i_{t+1}=q_j \mid i_t=q_i)，\quad i=1,2,\cdots,N;\ j=1,2,\cdots,N$$

时刻t处于状态q_i的条件下在时刻$t+1$转移到状态q_j的概率

隐马尔科夫模型

- 组成
  - B： 观测概率矩阵

$$B=\left[b_j(k)\right]_{N\times M}$$

$$b_j(k)=P(o_t=v_k \mid i_t=q_j)\,,\qquad k=1,2,\cdots,M;\ \ j=1,2,\cdots,N$$

在时刻 t 处于状态 q_j 的条件下生成观测 v_k 的概率

  - $\boldsymbol{\pi}$： 初始状态概率向量

$$\pi_i=P(i_1=q_i)\,,\quad i=1,2,\cdots,N$$

时刻 $t=1$ 处于状态 q_i 的概率

隐马尔科夫模型

- 三要素

$$\lambda=(A,B,\pi)$$

- 两个基本假设
  - 齐次马尔科夫性假设，隐马尔可分链t的状态只和t-1状态有关：

$$P(i_t \mid i_{t-1},o_{t-1},\cdots,i_1,o_1)=P(i_t \mid i_{t-1}), \quad t=1,2,\cdots,T$$

  - 观测独立性假设，观测只和当前时刻状态有关；

$$P(o_t \mid i_T,o_T,i_{T-1},o_{T-1},\cdots,i_{t+1},o_{t+1},i_t,i_{t-1},o_{t-1},\cdots,i_1,o_1)=P(o_t \mid i_t)$$

例：盒子和球模型

• 盒子：	1	2	3	4
• 红球：	5	3	6	8
• 白球：	5	7	4	2

- 转移规则：
  - 盒子1 下一个 盒子2
  - 盒子2或3 下一个 0.4 左，0.6右
  - 盒子4 下一个 0.5 自身，0.5盒子3
- 重复5次： O={ 红，红，白，白，红}

清華大學 Tsinghua University

例：盒子和球模型

- 状态集合：Q={盒子1，盒子2，盒子3，盒子4}，N=4
- 观测集合：V={红球，白球} M=2
- 初始化概率分布：

$$\pi = (0.25, 0.25, 0.25, 0.25)^{\mathrm{T}}$$

- 状态转移矩阵：

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0.6 & 0 \\ 0 & 0.4 & 0 & 0.6 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix}$$

观测矩阵：

$$B = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ 0.8 & 0.2 \end{bmatrix}$$

Tsinghua University

观测序列的生成过程

算法 10.1（观测序列的生成）

输入：隐马尔可夫模型 $\lambda=(A,B,\pi)$，观测序列长度 T；

输出：观测序列 $O=(o_1,o_2,\cdots,o_T)$.

（1）按照初始状态分布 π 产生状态 i_1

（2）令 $t=1$

（3）按照状态 i_t 的观测概率分布 $b_{i_t}(k)$ 生成 o_t

（4）按照状态 i_t 的状态转移概率分布 $\{a_{i_t i_{t+1}}\}$ 产生状态 i_{t+1}， $i_{t+1}=1,2,\cdots,N$

（5）令 $t=t+1$；如果 $t<T$，转步 (3)；否则，终止

Tsinghua University

隐马尔科夫模型的三个基本问题

- 1、概率计算问题
- 给定: $\lambda=(A,B,\pi)$ $O=(o_1,o_2,\cdots,o_T)$
- 计算: $P(O|\lambda)$
- 2、学习问题
- 已知: $O=(o_1,o_2,\cdots,o_T)$
- 估计: $\lambda=(A,B,\pi)$, 使 $P(O|\lambda)$ 最大
- 3、预测问题（解码）
- 已知: $\lambda=(A,B,\pi)$ $O=(o_1,o_2,\cdots,o_T)$
- 求: 使 $P(I|O)$ 最大的状态序列 $I=(i_1,i_2,\cdots,i_T)$

概率计算方法

- 直接计算法
  - 给定模型：$\lambda=(A,B,\pi)$ 和观测概率：$O=(o_1,o_2,\cdots,o_T)$
  - 计算：$P(O|\lambda)$
- 最直接的方法：
  - 列举所有可能的长度为T状态序列 $I=(i_1,i_2,\cdots,i_T)$
  - 求各个状态序列I与观测序列 $O=(o_1,o_2,\cdots,o_T)$ 的联合概率 $P(O,I|\lambda)$
  - 然后对所有可能的状态序列求和，得到 $P(O|\lambda)$

概率计算方法

- 直接计算法
  - 状态序列 $I=(i_1,i_2,\cdots,i_T)$概率：$P(I|\lambda)=\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\cdots a_{i_{T-1}i_T}$
  - 对固定的状态序列I，观测序列O的概率：$P(O|I,\lambda)$

$$P(O|I,\lambda)=b_{i_1}(o_1)b_{i_2}(o_2)\cdots b_{i_T}(o_T)$$

  - O和I同时出现的联合概率为：

$$\begin{aligned}P(O,I|\lambda)&=P(O|I,\lambda)P(I|\lambda)\\&=\pi_{i_1}b_{i_1}(o_1)a_{i_1i_2}b_{i_2}(o_2)\cdots a_{i_{T-1}i_T}b_{i_T}(o_T)\end{aligned}$$

  - 对所有可能的状态序列I求和，得到观测O的概率：

$$\begin{aligned}P(O|\lambda)&=\sum_I P(O|I,\lambda)P(I|\lambda)\\&=\sum_{i_1,i_2,\cdots,i_T}\pi_{i_1}b_{i_1}(o_1)a_{i_1i_2}b_{i_2}(o_2)\cdots a_{i_{T-1}i_T}b_{i_T}(o_T)\end{aligned}$$

复杂度

$O(TN^T)$

前向算法

- 前向概率定义：给定隐马尔科夫模型λ，定义到时刻t部分观测序列为：$o_1, o_2, \cdots, o_t$，且状态为qi的概率为前向概率，记作：$$\alpha_t(i) = P(o_1, o_2, \cdots, o_t, i_t = q_i \mid \lambda)$$

算法 10.2（观测序列概率的前向算法）

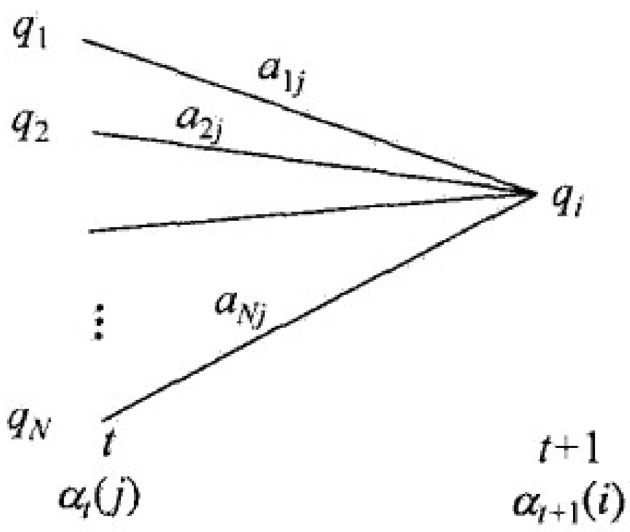
输入：隐马尔可夫模型 λ，观测序列 O；

输出：观测序列概率 $P(O \mid \lambda)$.

- 初值：$\alpha_1(i) = \pi_i b_i(o_1)$，$\quad i = 1, 2, \cdots, N$
- 递推：$$\alpha_{t+1}(i) = \left[\sum_{j=1}^{N} \alpha_t(j) a_{ji}\right] b_i(o_{t+1}), \quad i = 1, 2, \cdots, N$$
- 终止：$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

前向算法

- 因为：
$$\alpha_T(i) = P(o_1, o_2, \cdots, o_T, i_T = q_i \mid \lambda)$$
- 所以：
$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$
- 递推：



复杂度

$O(N^2T)$

前向算法

- 减少计算量的原因在于每一次计算，直接引用前一个时刻的计算结果，避免重复计算。

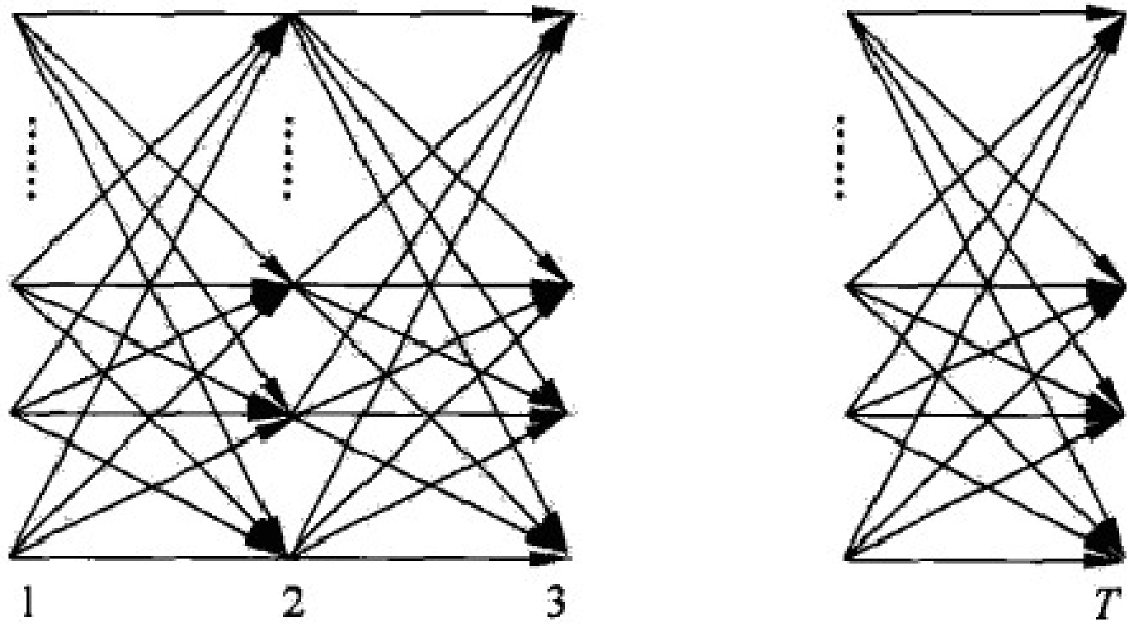


图 10.2　观测序列路径结构

复杂度

$O(N^2T)$

例:

$$A=\begin{bmatrix}0.5 & 0.2 & 0.3\\0.3 & 0.5 & 0.2\\0.2 & 0.3 & 0.5\end{bmatrix},\quad B=\begin{bmatrix}0.5 & 0.5\\0.4 & 0.6\\0.7 & 0.3\end{bmatrix},\quad \pi=(0.2,0.4,0.4)^{\mathrm{T}}$$

设 $T=3$， $O=$(红, 白, 红)，试用前向算法计算 $P(O|\lambda)$

解 按照算法 10.2

（1）计算初值

$$\alpha_1(1)=\pi_1 b_1(o_1)=0.10$$
$$\alpha_1(2)=\pi_2 b_2(o_1)=0.16$$
$$\alpha_1(3)=\pi_3 b_3(o_1)=0.28$$

（2）递推计算

例：

$$\alpha_2(1)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i1}\right]b_1(o_2)=0.154\times0.5=0.077$$

$$\alpha_2(2)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i2}\right]b_2(o_2)=0.184\times0.6=0.1104$$

$$\alpha_2(3)=\left[\sum_{i=1}^{3}\alpha_1(i)a_{i3}\right]b_3(o_2)=0.202\times0.3=0.0606$$

$$\alpha_3(1)=\left[\sum_{i=1}^{3}\alpha_2(i)a_{i1}\right]b_1(o_3)=0.04187$$

$$\alpha_3(2)=\left[\sum_{i=1}^{3}\alpha_2(i)a_{i2}\right]b_2(o_3)=0.03551$$

$$\alpha_3(3)=\left[\sum_{i=1}^{3}\alpha_2(i)a_{i3}\right]b_3(o_3)=0.05284$$

例：

（3）终止

$$P(O|\lambda)=\sum_{i=1}^{3}\alpha_3(i)=0.13022$$

后向算法

- 定义10.3 后向概率：给定隐马尔科夫模型λ，定义在时刻t状态为qi的条件下，从t+1到T的部分观测序列为：$o_{t+1}, o_{t+2}, \cdots, o_T$ 的概率为后向概率，记作：

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \cdots, o_T \mid i_t = q_i, \lambda)$$

可以用递推的方法求得后向概率 $\beta_t(i)$ 及观测序列概率 $P(O \mid \lambda)$

后向算法

算法 10.3 (观测序列概率的后向算法)

输入：隐马尔可夫模型 λ，观测序列 O；

输出：观测序列概率 $P(O|\lambda)$．

（1）

$$\beta_T(i)=1, \quad i=1,2,\cdots,N$$

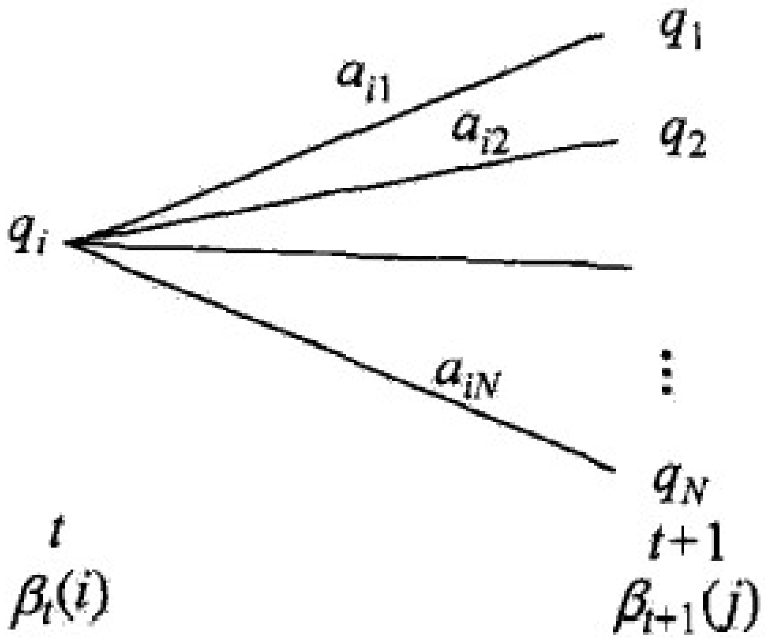
（2）对 $t=T-1,T-2,\cdots,1$

$$\beta_t(i)=\sum_{j=1}^{N}a_{ij}b_j(o_{t+1})\beta_{t+1}(j), \quad i=1,2,\cdots,N$$

（3）

$$P(O|\lambda)=\sum_{i=1}^{N}\pi_i b_i(o_1)\beta_1(i)$$

后向算法



- 前向后向统一写为：（t=1 和t=T-1分别对应）

$$P(O|\lambda)=\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j),\quad t=1,2,\cdots,T-1$$

一些概率和期望值的计算

1．给定模型 λ 和观测 O，在时刻 t 处于状态 q_i 的概率．

记 $$\gamma_t(i) = P(i_t = q_i \mid O, \lambda)$$

$$\gamma_t(i) = P(i_t = q_i \mid O, \lambda) = \frac{P(i_t = q_i, O \mid \lambda)}{P(O \mid \lambda)}$$

$$\alpha_t(i)\beta_t(i) = P(i_t = q_i, O \mid \lambda)$$

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}$$

Tsinghua University

一些概率和期望值的计算

2. 给定模型 λ 和观测 O，在时刻 t 处于状态 q_i 且在时刻 $t+1$ 处于状态 q_j 的概率. 记

$$\xi_t(i,j) = P(i_t = q_i, i_{t+1} = q_j \mid O, \lambda)$$

通过前向后向概率计算:

$$\xi_t(i,j) = \frac{P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}{P(O \mid \lambda)} = \frac{P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}{\sum_{i=1}^{N}\sum_{j=1}^{N} P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}$$

$$P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda) = \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$$

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

一些概率和期望值的计算

3．将$\gamma_t(i)$和$\xi_t(i,j)$对各个时刻t求和，可以得到一些有用的期望值：

（1）在观测O下状态i出现的期望值

$$\sum_{t=1}^{T}\gamma_t(i)$$

（2）在观测O下由状态i转移的期望值

$$\sum_{t=1}^{T-1}\gamma_t(i)$$

（3）在观测O下由状态i转移到状态j的期望值

$$\sum_{t=1}^{T-1}\xi_t(i,j)$$

学习算法

- 监督学习方法:
  - 假设训练数据是包括观测序列O和对应的状态序列I

$$\{(O_1,I_1),(O_2,I_2),\cdots,(O_S,I_S)\}$$

  - 可以利用极大似然估计法来估计隐马尔可夫模型参数。
- 非监督学习方法:
  - 假设训练数据只有S个长度为T的观测序{O1,O2,...Os},
  - 采用Baum-Welch算法

监督学习方法

- 已知： $\{(O_1,I_1),(O_2,I_2),\cdots,(O_S,I_S)\}$
- 1、转移概率a_{ij}的估计：
- 设样本中时刻t处于状态i，时刻t+1转移到状态j的频数为A_{ij}，那么状态转移概率a_{ij}的估计是：

$$\hat{a}_{ij}=\frac{A_{ij}}{\sum_{j=1}^{N}A_{ij}},\quad i=1,2,\cdots,N;\ j=1,2,\cdots,N$$

监督学习方法

- 已知：$\{(O_1,I_1),(O_2,I_2),\cdots,(O_S,I_S)\}$
- 2、观测概率$b_j(k)$的估计：设样本中状态为j并观测为k的频数是$B_j(k)$，那么状态为j观测为k的概率

$$\hat{b}_j(k)=\frac{B_{jk}}{\sum_{k=1}^{M}B_{jk}},\quad j=1,2,\cdots,N;\ k=1,2,\cdots,M$$

- 3、初始状态概率 π_i 的估计 $\hat{\pi}_i$ 为S个样本中初始状态为qi的频率。
- 往往人工标注数据很贵

Baum-Welch算法

- 假定训练数据只包括{O1,O2,...Os},
- 求模型参数λ=（A,B,π)
- 实质上是有隐变量的概率模型：EM算法

$$P(O|\lambda)=\sum_I P(O|I,\lambda)P(I|\lambda)$$

- 1、确定完全数据的对数似然函数
- 完全数据 $(O,I)=(o_1,o_2,\cdots,o_T,i_1,i_2,\cdots,i_T)$
- 完全数据的对数似然函数 $\log P(O,I|\lambda)$

Baum Welch算法

- 2、EM的E步 求 Q 函数 $Q(\lambda,\bar{\lambda})$

$$Q(\lambda,\bar{\lambda})=\sum_I \log P(O,I\mid\lambda)P(O,I\mid\bar{\lambda})$$

$$P(O,I\mid\lambda)=\pi_{i_1}b_{i_1}(o_1)a_{i_1i_2}b_{i_2}(o_2)\cdots a_{i_{T-1}i_T}b_{i_T}(o_T)$$

- 则:

$$Q(\lambda,\bar{\lambda})=\sum_I \log\pi_{i_1}P(O,I\mid\bar{\lambda}) + \sum_I\left(\sum_{t=1}^{T-1}\log a_{i_t i_{t+1}}\right)P(O,I\mid\bar{\lambda})+\sum_I\left(\sum_{t=1}^{T}\log b_{i_t}(o_t)\right)P(O,I\mid\bar{\lambda})$$

- 对序列总长度T进行

Baum Welch算法

- 3、EM算法的M 步，极大化 $Q(\lambda,\bar{\lambda})$ 求模型参数A,B,π

第一项：$\sum_{I}\log\pi_{i_0}P(O,I|\bar{\lambda})=\sum_{i=1}^{N}\log\pi_i P(O,i_1=i|\bar{\lambda})$

由约束条件：$\sum_{i=1}^{N}\pi_i=1$ 利用拉格朗日乘子：

$$\sum_{i=1}^{N}\log\pi_i P(O,i_1=i|\bar{\lambda})+\gamma\left(\sum_{i=1}^{N}\pi_i-1\right)$$

求偏导数，并结果为0

- $$\frac{\partial}{\partial\pi_i}\left[\sum_{i=1}^{N}\log\pi_i P(O,i_1=i|\bar{\lambda})+\gamma\left(\sum_{i=1}^{N}\pi_i-1\right)\right]=0$$

- 得：$P(O,i_1=i|\bar{\lambda})+\gamma\pi_i=0$ $\quad\gamma=-P(O|\bar{\lambda})$ $\quad\pi_i=\dfrac{P(O,i_1=i|\bar{\lambda})}{P(O|\bar{\lambda})}$

学习算法 Baum Welch算法

- 3、EM算法的M 步，极大化$Q(\lambda,\bar{\lambda})$ 求A,B,π

第二项可写成：

$$\sum_{I}\left(\sum_{t=1}^{T-1}\log a_{i_t i_{t+1}}\right)P(O,I\mid\bar{\lambda})=\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{T-1}\log a_{ij}P(O,i_t=i,i_{t+1}=j\mid\bar{\lambda})$$

由约束条件 $\sum_{j=1}^{N}a_{ij}=1$，拉格朗日乘子法：

- 得：

$$a_{ij}=\frac{\sum_{t=1}^{T-1}P(O,i_t=i,i_{t+1}=j\mid\bar{\lambda})}{\sum_{t=1}^{T-1}P(O,i_t=i\mid\bar{\lambda})}$$

Baum Welch算法

- 3、EM算法的M 步，极大化 $Q(\lambda,\bar{\lambda})$ 求A,B,π

第三项：

$$\sum_{I}\left(\sum_{t=1}^{T}\log b_{i_t}(o_t)\right)P(O,I\mid\bar{\lambda})=\sum_{j=1}^{N}\sum_{t=1}^{T}\log b_j(o_t)P(O,i_t=j\mid\bar{\lambda})$$

由约束条件： $\sum_{k=1}^{M}b_j(k)=1$

注意，只有在 $o_t=v_k$ 时 $b_j(o_t)$ 对 $b_j(k)$ 的偏导数才不为 0，

以 $I(o_t=v_k)$ 表示．求得

$$b_j(k)=\frac{\sum_{t=1}^{T}P(O,i_t=j\mid\bar{\lambda})I(o_t=v_k)}{\sum_{t=1}^{T}P(O,i_t=j\mid\bar{\lambda})}$$

学习算法 Baum Welch算法

- 将已上得到的概率分别用 $\gamma_t(i)$， $\xi_t(i,j)$ 表示：

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

$$b_j(k)=\frac{\sum_{t=1,o_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}$$

$$\pi_i=\gamma_1(i)$$

学习算法 Baum Welch算法

算法 10.4（Baum-Welch 算法）

输入：观测数据 $O=(o_1,o_2,\cdots,o_T)$；

输出：隐马尔可夫模型参数.

（1）初始化

对 $n=0$，选取 $a_{ij}^{(0)}$，$b_j(k)^{(0)}$，$\pi_i^{(0)}$，得到模型 $\lambda^{(0)}=(A^{(0)},B^{(0)},\pi^{(0)})$

（2）递推．对 $n=1,2,\cdots$，

$$a_{ij}^{(n+1)}=\frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} \qquad b_j(k)^{(n+1)}=\frac{\sum_{t=1,o_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}$$

$$\pi_i^{(n+1)}=\gamma_1(i)$$

右端各值按观测 $O=(o_1,o_2,\cdots,o_T)$ 和模型 $\lambda^{(n)}=(A^{(n)},B^{(n)},\pi^{(n)})$ 计算

（3）终止．得到模型参数 $\lambda^{(n+1)}=(A^{(n+1)},B^{(n+1)},\pi^{(n+1)})$

预测算法

- 近似算法
- 想法：在每个时刻t选择在该时刻最有可能出现的状态 i_t^*，从而得到一个状态序列 $I^*=(i_1^*,i_2^*,\cdots,i_T^*)$ 将它作为预测的结果，在时刻t处于状态qi的概率：
$$\gamma_t(i)=\frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)}=\frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}$$
- 在每一时刻t最有可能的状态是：$i_t^*=\arg\max_{1\leqslant i\leqslant N}[\gamma_t(i)]$， $t=1,2,\cdots,T$

从而得到状态序列：$I^*=(i_1^*,i_2^*,\cdots,i_T^*)$

得到的状态有可能实际不发生

维特比算法

- Viterbi 方法
- 用动态规划解概率最大路径，一个路径对应一个状态序列。
- 最优路径具有这样的特性：如果最优路径在时刻t通过结点 i_t^* ，那么这一路径从结点 i_t^* 到终点 i_T^* 的部分路径，对于从 i_t^* 到 i_T^* 的所有可能的部分路径来说，必须是最优的。
- 只需从时刻t=1开始，递推地计算在时刻t状态为i的各条部分路径的最大概率，直至得到时刻t=T状态为i的各条路径的最大概率，时刻t=T的最大概率即为最优路径的概率P*，最优路径的终结点 i_T^* 也同时得到。
- 之后，为了找出最优路径的各个结点，从终结点开始，由后向前逐步求得结点 $i_{T-1}^*,\cdots,i_1^*$，得到最优路径

$$I^* = (i_1^*, i_2^*, \cdots, i_T^*)$$

维特比算法

- 导入两个变量δ和ψ，定义在时刻t状态为i的所有单个路径 $(i_1,i_2,\cdots,i_t)$中概率最大值为：

$$\delta_t(i)=\max_{i_1,i_2,\cdots,i_{t-1}} P(i_t=i,i_{t-1},\cdots,i_1,o_t,\cdots,o_1\mid\lambda),\quad i=1,2,\cdots,N$$

- 由定义可得变量δ的递推公式：

$$\begin{aligned}\delta_{t+1}(i)&=\max_{i_1,i_2,\cdots,i_t} P(i_{t+1}=i,i_t,\cdots,i_1,o_{t+1},\cdots,o_1\mid\lambda)\\&=\max_{1\leqslant j\leqslant N}[\delta_t(j)a_{ji}]b_i(o_{t+1}),\qquad i=1,2,\cdots,N;\ t=1,2,\cdots,T-1\end{aligned}$$

- 定义在时刻t状态为i的所有单个路径中概率最大的路径的第t-1个结点为 $(i_1,i_2,\cdots,i_{t-1},i)$

$$\psi_t(i)=\arg\max_{1\leqslant j\leqslant N}[\delta_{t-1}(j)a_{ji}],\qquad i=1,2,\cdots,N$$

Tsinghua University

Viterbi 方法

算法 10.5（维特比算法）

输入：模型 $\lambda=(A,B,\pi)$ 和观测 $O=(o_1,o_2,\cdots,o_T)$；

输出：最优路径 $I^*=(i_1^*,i_2^*,\cdots,i_T^*)$.

（1）初始化

$$\delta_1(i)=\pi_i b_i(o_1), \qquad i=1,2,\cdots,N$$

$$\psi_1(i)=0, \qquad i=1,2,\cdots,N$$

（2）递推．对 $t=2,3,\cdots,T$

$$\delta_t(i)=\max_{1\leqslant j\leqslant N}[\delta_{t-1}(j)a_{ji}]b_i(o_t), \qquad i=1,2,\cdots,N$$

$$\psi_t(i)=\arg\max_{1\leqslant j\leqslant N}[\delta_{t-1}(j)a_{ji}], \qquad i=1,2,\cdots,N$$

Viterbi 方法

（3）终止

$$P^* = \max_{1\leqslant i\leqslant N} \delta_T(i)$$

$$i_T^* = \arg\max_{1\leqslant i\leqslant N}[\delta_T(i)]$$

（4）最优路径回溯．对 $t = T-1, T-2, \cdots, 1$

$$i_t^* = \psi_{t+1}(i_{t+1}^*)$$

求得最优路径 $I^* = (i_1^*, i_2^*, \cdots, i_T^*)$

例

$$A=\begin{bmatrix}0.5 & 0.2 & 0.3\\ 0.3 & 0.5 & 0.2\\ 0.2 & 0.3 & 0.5\end{bmatrix},\quad B=\begin{bmatrix}0.5 & 0.5\\ 0.4 & 0.6\\ 0.7 & 0.3\end{bmatrix},\quad \pi=(0.2,0.4,0.4)^{\mathrm{T}}$$

$O=$(红, 白, 红)，试求最优状态序列，即最优路径 $I^*=(i_1^*,i_2^*,i_3^*)$

- 1、初始化：在t=1时，对每一个状态i，i=1,2,3，求状态i观测O1为红的概率，记为：$\delta_1(i)$

$$\delta_1(i)=\pi_i b_i(o_1)=\pi_i b_i(\text{红}),\quad i=1,2,3$$

- 代入实际数据： $\delta_1(1)=0.10$， $\delta_1(2)=0.16$， $\delta_1(3)=0.28$

记 $\psi_1(i)=0$， $i=1,2,3$

例

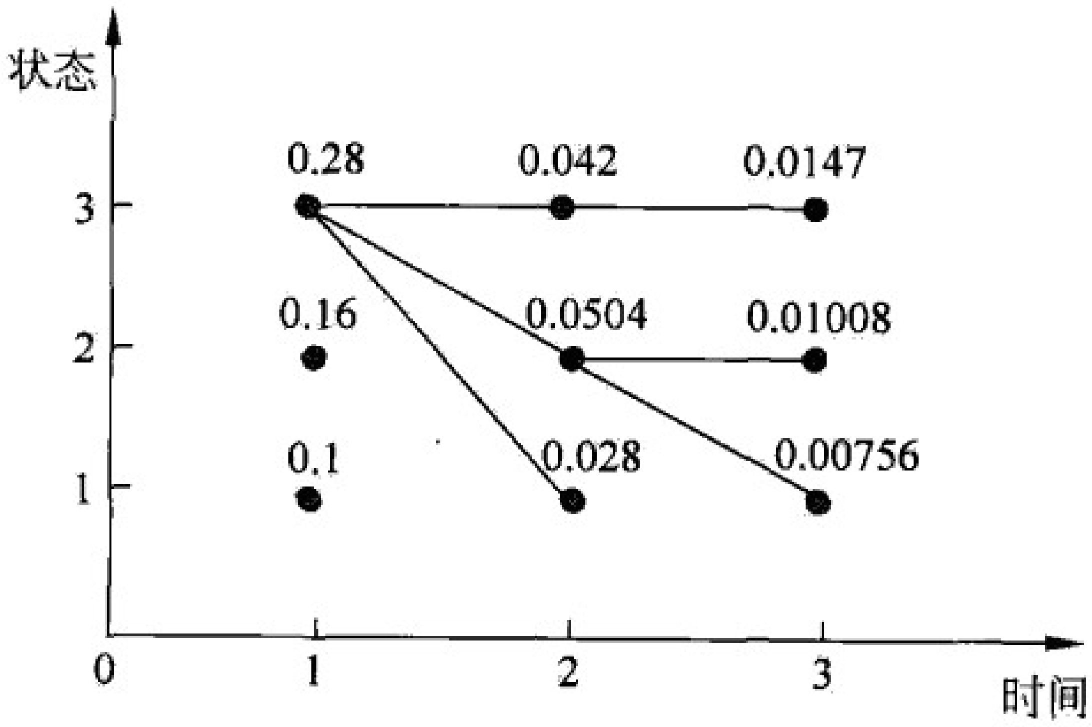


图 10.4　求最优路径

例

- 2、在t=2时，对每一个状态i，i=1,2,3，求在t=1时状态为j观测O1为红并在t=2时状态为i观测O2位白的路径的最大概率，记为 $\delta_2(i)$

$$\delta_2(i) = \max_{1 \leqslant j \leqslant 3} [\delta_1(j) a_{ji}] b_i(o_2)$$

- 同时，对每个状态i，记录概率最大路径的前一个状态j

$$\psi_2(i) = \arg\max_{1 \leqslant j \leqslant 3} [\delta_1(j) a_{ji}], \qquad i = 1,2,3$$

例

计算：

$$\delta_2(1)=\max_{1\leqslant j\leqslant 3}[\delta_1(j)a_{j1}]b_1(o_2)$$
$$=\max_j\{0.10\times0.5,0.16\times0.3,0.28\times0.2\}\times0.5$$
$$=0.028$$

$$\psi_2(1)=3$$

$$\delta_2(2)=0.0504,\quad \psi_2(2)=3$$

$$\delta_2(3)=0.042,\quad \psi_2(3)=3$$

同样，在$t=3$时

$$\delta_3(i)=\max_{1\leqslant j\leqslant 3}[\delta_2(j)a_{ji}]b_i(o_3)$$

$$\psi_3(i)=\arg\max_{1\leqslant j\leqslant 3}[\delta_2(j)a_{ji}]$$

$$\delta_3(1)=0.00756,\quad \psi_3(1)=2$$

$$\delta_3(2)=0.01008,\quad \psi_3(2)=2$$

$$\delta_3(3)=0.0147,\quad \psi_3(3)=3$$

例

- 3、以P*表示最优路径的概率:

$$P^* = \max_{1 \leq i \leq 3} \delta_3(i) = 0.0147$$

- 最优路径的终点是:

$$i_3^* = \arg\max_i [\delta_3(i)] = 3$$

- 4、由最优路径的终点i_3^*， 逆向找到 i_2^*, i_1^*

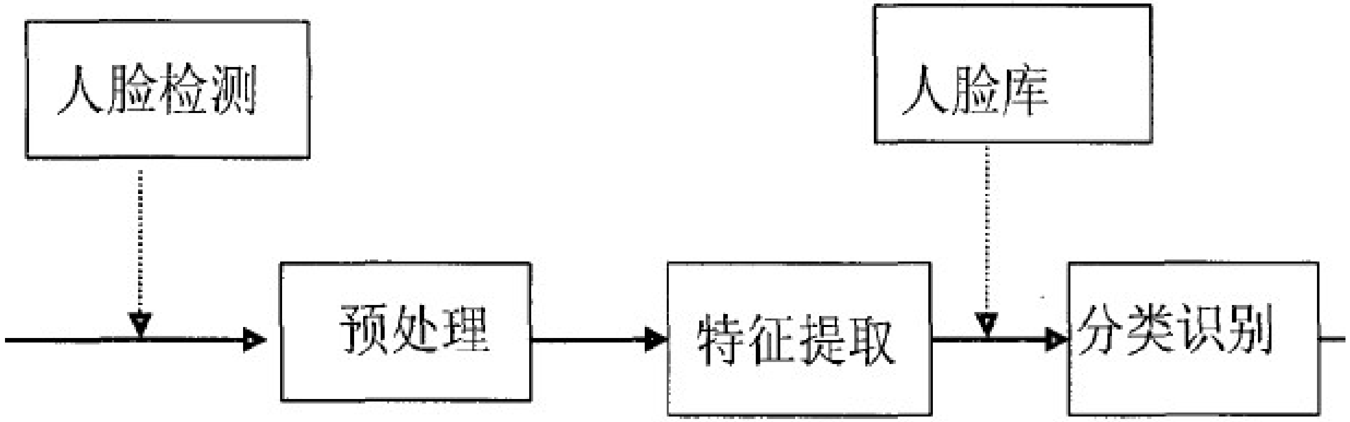
在$t=2$时， $i_2^* = \psi_3(i_3^*) = \psi_3(3) = 3$

在$t=1$时， $i_1^* = \psi_2(i_2^*) = \psi_2(3) = 3$

- 于是求得最优路径，即最优状态序列:

$$I^* = (i_1^*, i_2^*, i_3^*) = (3,3,3)$$

人脸检测



人脸检测

- 人脸图像预处理
  - 光线补偿

$$I_{ave}=\frac{1}{N}\sum_{i=1}^{N}I_i \qquad I_{modify}=\left(127-I_{ave}\right)+I_i$$

  - 中值滤波

  - 归一化处理

$$G\left(i,j\right)=\frac{I\left(i,j\right)-m}{v}\times sv+sm$$

人脸识别

- HMM训练步骤： 对每个人脸建立一个HMM
- 1、人脸特征向量提取
- 2、建立公用的HMM模型
- 3、HMM初始参数确定
- 4、初始模型参数训练，主要是运用Viterbi算法训练均匀分割得到参数，求得最佳分割点，然后重新计算模型初始参数，直到模型收敛为止。
- 5、人脸模型训练过程，将(1)中得到的观测向量代入(4)中得到的模型参数进行训练，使用迭代方法调整模型参数达到最优。

人脸检测

- 1、人脸特征向量提取：
  - 基于奇异值分解的特征提取
  - 离散余弦变换
  - 多维尺度分析(MDS)
  - 人脸等密度线分析匹配方法、
  - 弹性图匹配方法
  - 。。。

人脸检测

- SVD

奇异矩阵 A 可以分解为以下形式：

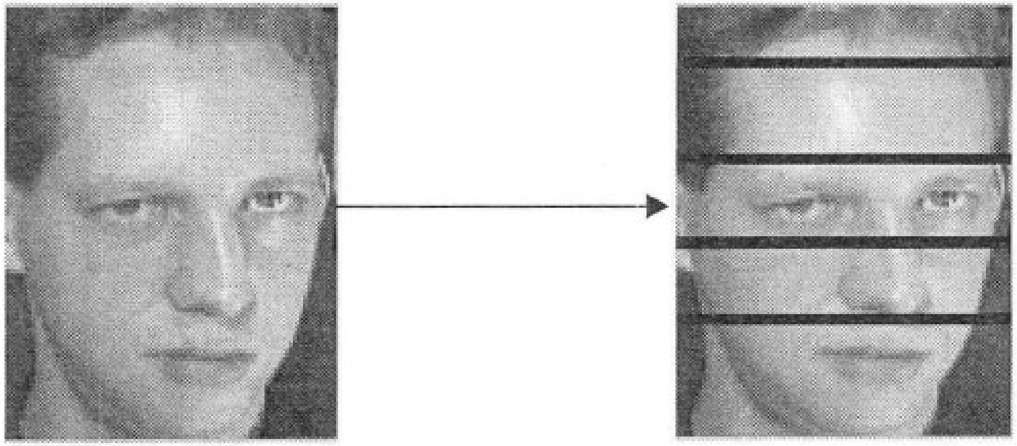
$$A = U\begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix}V^T$$

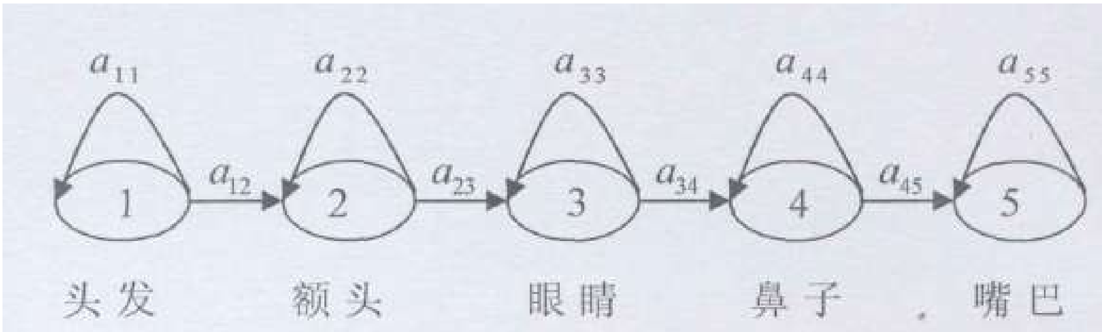
- 稳定性：由于奇异值特征向量具有良好的稳定性，所以它对图像噪音、图像光照条件引起的灰度变化具有不敏感的特性;
- 转置不变性：A和A转置具有相同的奇异值;
- 旋转不变性：图像A和旋转后的图像有相同的特征向量;
- 唯一不变性：对矩阵A换两行或者两列仍然具有相同的特征向量;
- 镜像变换不变形：

人脸检测

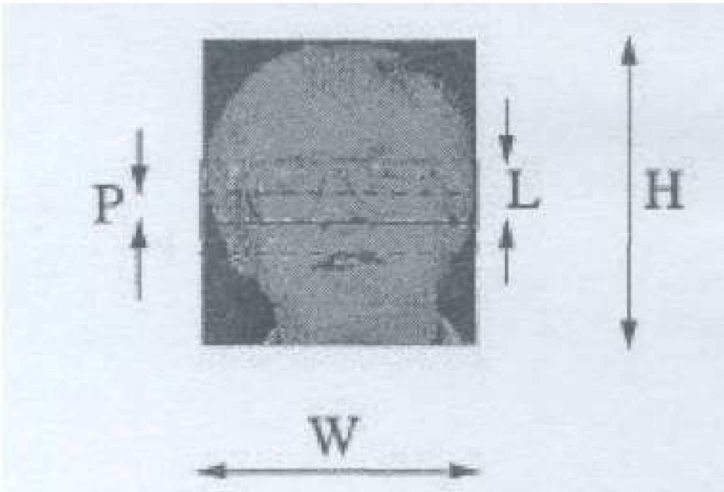
- HMM模型

  状态





人脸检测



$$T=\frac{H-L}{L-P}+1$$

人脸检测

- 3、初始参数确定:
- $\pi = (1,0,0,0,0)$
- A矩阵:

$$A = \begin{bmatrix} \frac{19}{20} & \frac{1}{20} & 0 & 0 & 0 \\ 0 & \frac{19}{20} & \frac{1}{20} & 0 & 0 \\ 0 & 0 & \frac{19}{20} & \frac{1}{20} & 0 \\ 0 & 0 & 0 & \frac{19}{20} & \frac{1}{20} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

- B矩阵：用混合高斯模型，则是将均匀分割后得到的五个部分中的每个部分，使用K均值聚类，将每个状态聚成M类，然后分别计算每一类的均值和方差，将这两个值分别赋给高斯模型。

人脸检测

- 初始模型
- 参数训练：

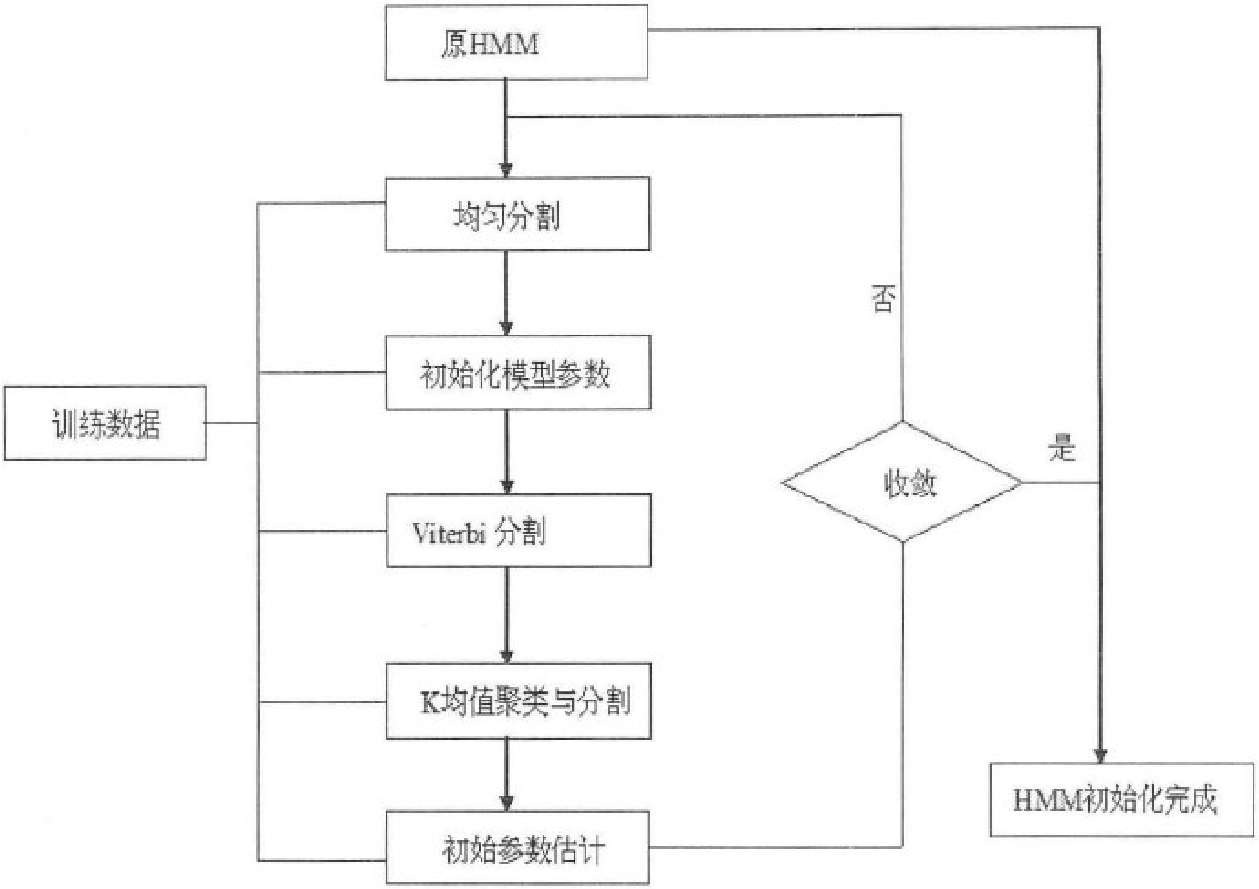


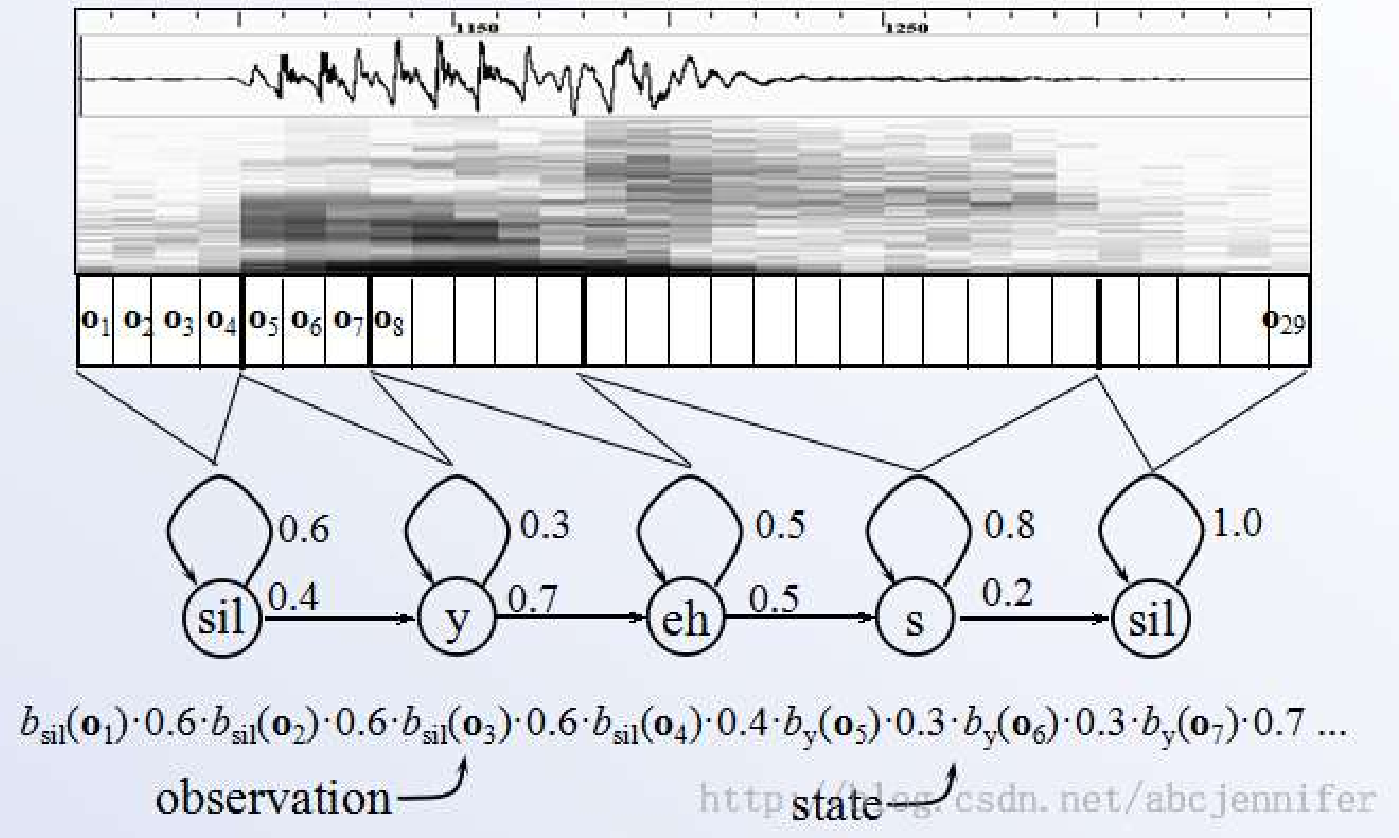
图4.2 HMM的初始化流程图

人脸检测

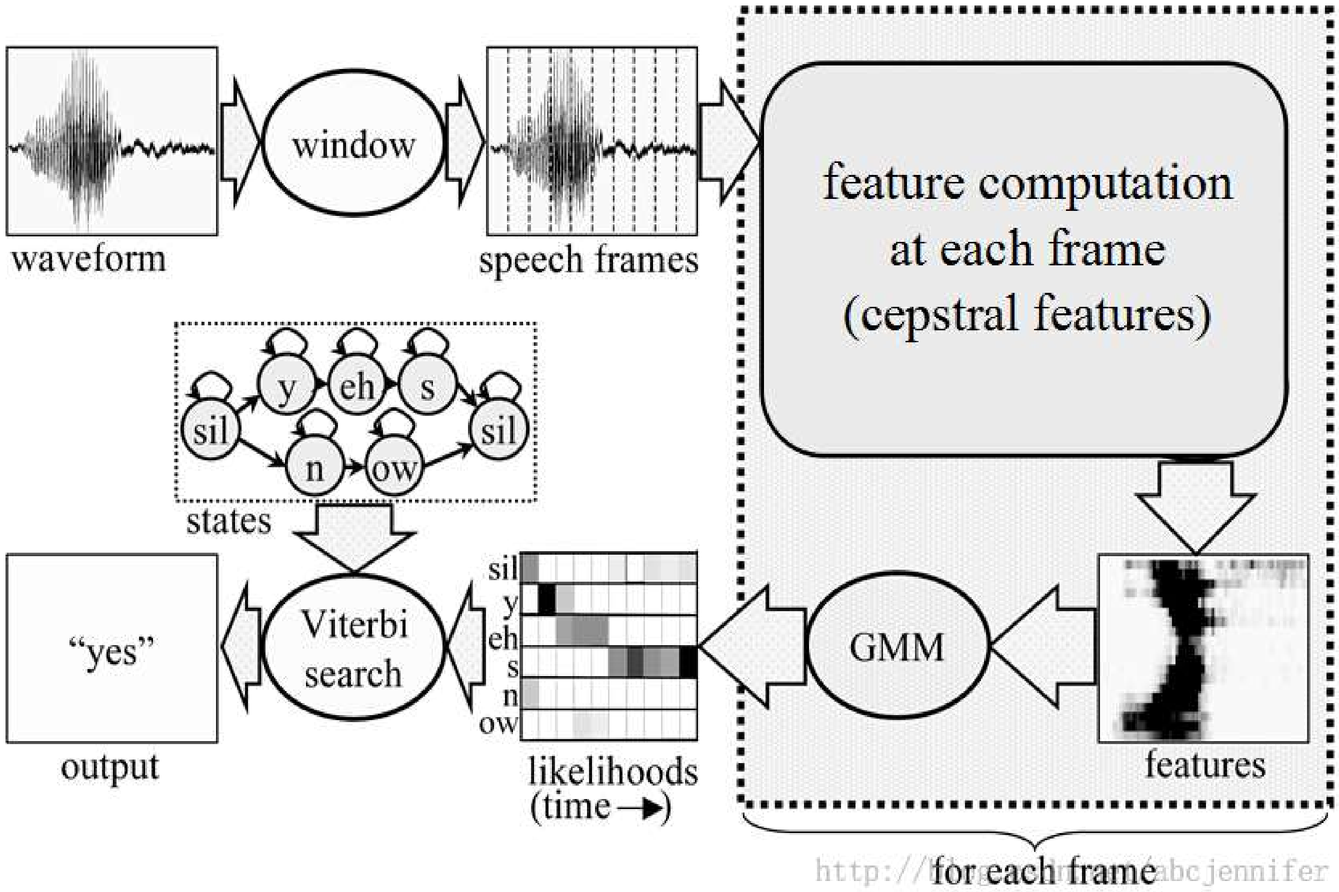
表 4.1 HMM 实验结果（ORL）

	N=2	N=3	N=4
基于像素值特征的 HMM[46-47]	84%	84%	84%
基于 SVD 的 HMM	82.5%	86%	85.5%

语音识别


1150
1250
$\mathbf{o}_1$ $\mathbf{o}_2$ $\mathbf{o}_3$ $\mathbf{o}_4$ $\mathbf{o}_5$ $\mathbf{o}_6$ $\mathbf{o}_7$ $\mathbf{o}_8$
$\mathbf{o}_{29}$
0.6
0.3
0.5
0.8
1.0
sil
y
eh
s
sil
0.4
0.7
0.5
0.2
$b_{\mathrm{sil}}(\mathbf{o}_1)\cdot 0.6\cdot b_{\mathrm{sil}}(\mathbf{o}_2)\cdot 0.6\cdot b_{\mathrm{sil}}(\mathbf{o}_3)\cdot 0.6\cdot b_{\mathrm{sil}}(\mathbf{o}_4)\cdot 0.4\cdot b_{\mathrm{y}}(\mathbf{o}_5)\cdot 0.3\cdot b_{\mathrm{y}}(\mathbf{o}_6)\cdot 0.3\cdot b_{\mathrm{y}}(\mathbf{o}_7)\cdot 0.7\ \ldots$
observation
state

语音识别



waveform
window
speech frames
feature computation at each frame (cepstral features)
y
eh
s
sil
n
ow
sil
states
features
"yes"
output
Viterbi search
sil
y
eh
s
n
ow
likelihoods (time→)
GMM
for each frame
http://blog.csdn.net/abcjennifer

隐马尔科夫模型在入侵检测中的应用

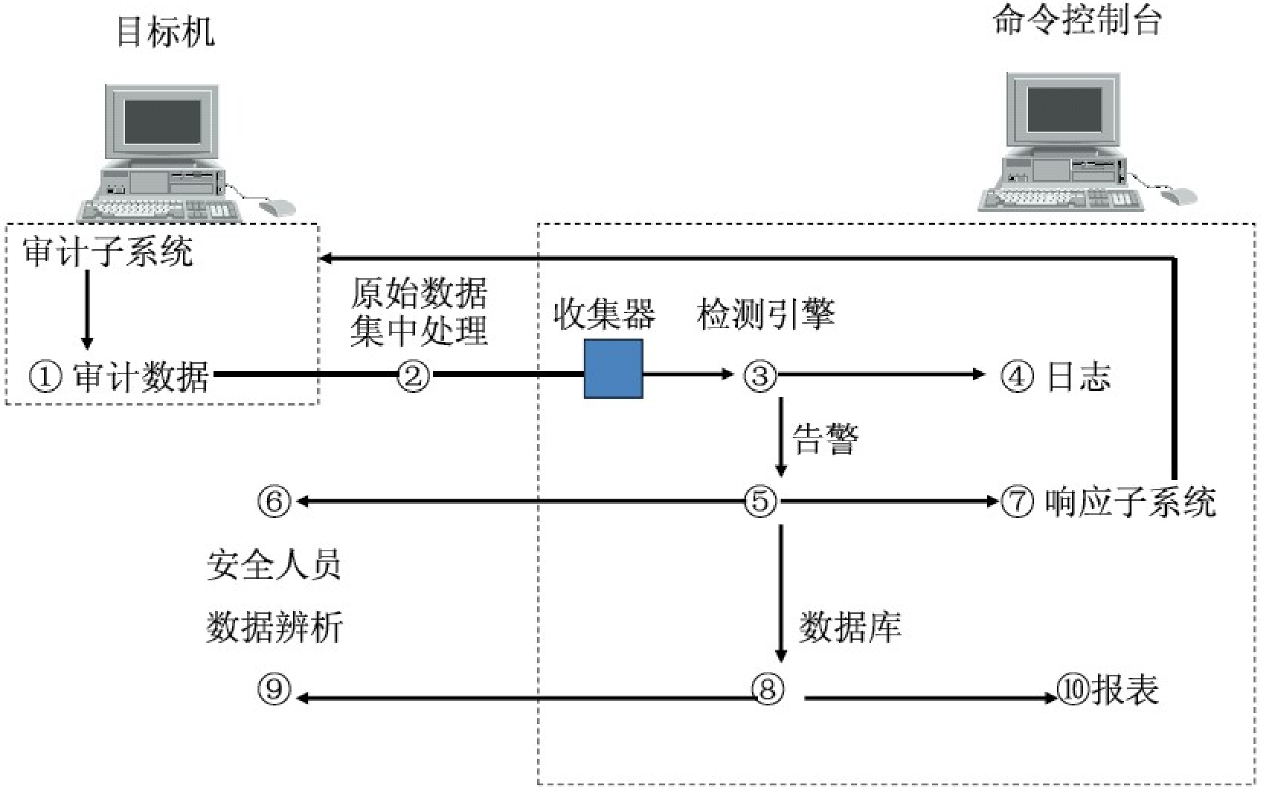
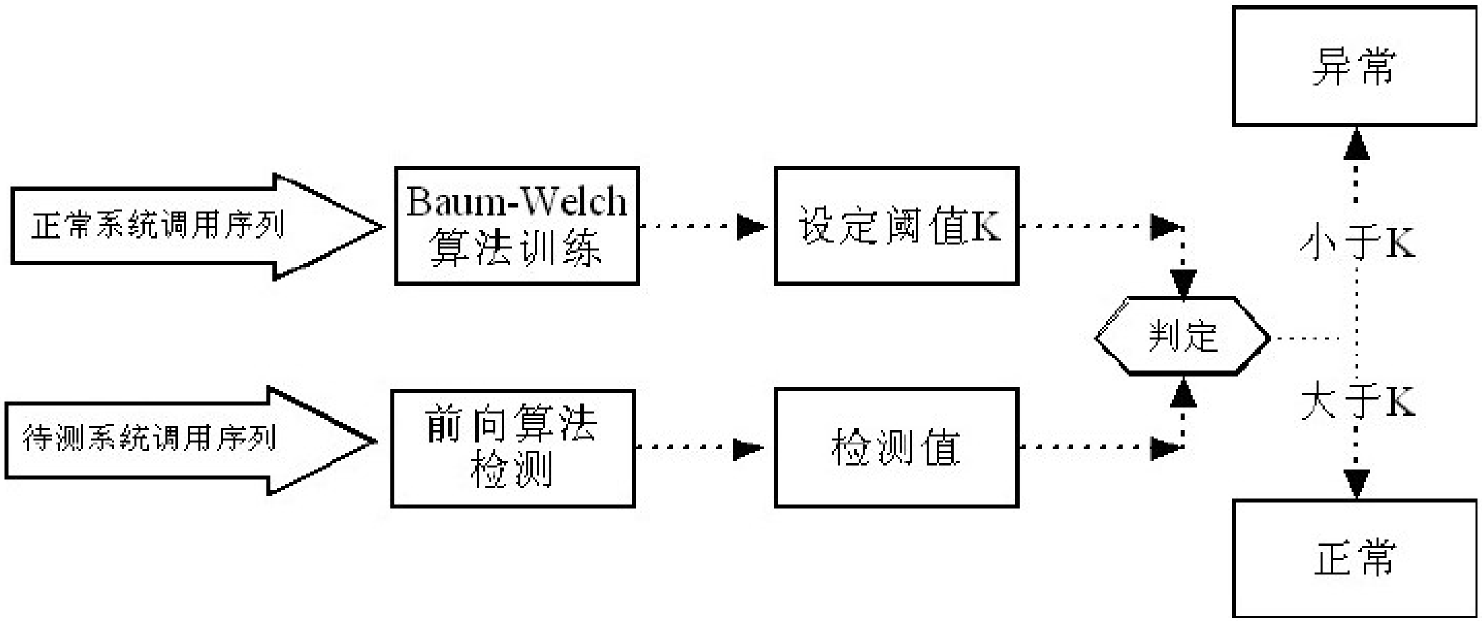


图15.2 集中式基于主机的入侵检测系统结构

隐马尔科夫模型在入侵检测中的应用



数据集

- 两种正常系统调用序列数据集。
- Sendmail正常系统调用序列数据集
- Sendmail的正常训练数据集是由1571583个系统调用构成的序列,这些系统调用分属147个不同的进程(在本实验中只考虑系统调用,不考虑进程)。
- Sendmail正常系统调用序列中包含48个唯一系统调用,这些系统调用的调用号为(按照序列中出现的顺序):4,2,66, 138, 5,23,45,27,167, 85,59,105, 104, 106, 56, 19, 155, 83, 93, 94, 112, 100, 50, 128, 89, 121, 11, 1, 40,38,18, 78,101,102,88,95,6,108, 32,1, 8,9, 14,17,3,124,41,61。

数据集

- lpr的正常系统调用序列数据集
- lpr的正常系统调用序列由2398个系统调用构成,分属9个进程,长度大约是sendmail 正常系统调用序列的六百五十五分之一。
- Lpr正常系统调用序列包含包含37个唯一系统调用,系统调用号分别为(按照序列中出现的顺序):4, 2, 66, 138, 5, 23, 45, 27, 105, 104, 106, 83, 59, 50, 88, 167, 17, 18, 155,19, 127, 93, 100, 112, 143, 128, 85, 89, 121, 3, 56, 7, 119, 32, 9, 8, 94。

- END
- Q&R