隐马尔可夫模型（HMM）

作者：Calvin
QQ: 179209347
Mail: 179209347@qq.com

介绍

笔记简介：

- 面向对象：深度学习初学者
- 依赖课程：**线性代数，统计概率**，优化理论，图论，离散数学，微积分，信息论

知乎专栏：

https://zhuanlan.zhihu.com/p/693738275

Github & Gitee 地址：

https://github.com/mymagicpower/AIAS/tree/main/deep_learning

https://gitee.com/mymagicpower/AIAS/tree/main/deep_learning

*** 版权声明：**

- 仅限用于个人学习
- 禁止用于任何商业用途

马尔可夫假设

马尔可夫假设（Markov assumption）是马尔可夫过程（Markov process）和马尔可夫链（Markov chain）的基础。它的核心思想是，未来的状态只依赖于当前的状态，而与过去的状态无关。

定义：马尔可夫假设指出，给定当前状态，未来状态的条件概率分布不依赖于过去的状态。数学上，如果 X_t 表示在时间 t 的状态，那么马尔可夫假设可以表示为：

$$P(X_{t+1}|X_t) = P(X_{t+1}|X_t, X_{t-1}, \dots, X_0)$$

这意味着，状态 X_{t+1} 的概率只取决于当前的状态 X_t ，而与之前的所有状态 $X_{t-1}, \dots, X_0$ 无关。

马尔可夫假设

马尔可夫过程：在**连续**时间马尔可夫过程中，马尔可夫假设也适用，尽管表示形式稍有不同。此时，状态的转移依赖于状态转移率，而非离散的转移概率。

马尔可夫链：马尔可夫链是马尔可夫过程的一种特例，其中时间参数是离散的。在**离散**时间马尔科夫链中，这一假设简化了状态转移的计算。马尔可夫链由状态空间、转移矩阵和初始状态组成：

- **状态空间**：所有可能状态的集合。
- **转移矩阵**：描述从一个状态转移到另一个状态的概率。转移矩阵的每一行表示从当前状态转移到所有其他状态的概率分布。

隐马尔科夫模型（HMM）：在HMM中，马尔可夫假设被用于隐状态的转移，这些隐状态通过观测变量间接观察到。在HMM中，有两个重要的假设：

- **马尔可夫假设**：当前隐状态仅依赖于前一个隐状态。
- **观测独立性假设**：当前观测仅依赖于当前马尔可夫链的状态，与其他隐状态和观测无关。

马尔可夫链 - 状态转移概率矩阵

对于一个马尔可夫状态 s 及其后续状态 s'，状态转移概率 定义为：

$$P_{ss'} = P[X_{t+1} = s' | X_t = s]$$

状态转移概率矩阵 P 定义为从所有的状态 s 到所有的后续状态 s' 的转移概率：

$$\boldsymbol{P} = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1n} \\ P_{21} & P_{22} & \dots & P_{2n} \\ \dots & & & \\ \dots & & & \\ P_{n1} & P_{n2} & \dots & P_{nn} \end{bmatrix}$$

矩阵中行号表示当前状态 s，列号表示到达的后续状态 s'。由概率分布的性质可得，转移矩阵是一个正定矩阵，且每行元素之和等于1。

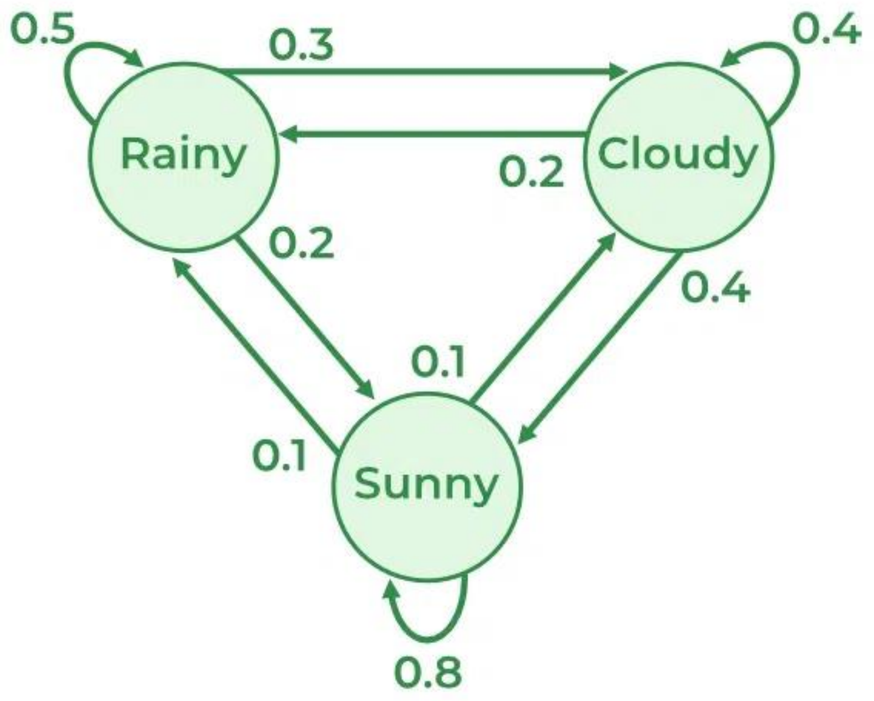
$$\forall i,j: P_{i,j} > 0, \forall i: \sum_j P_{i,j} = 1$$

按相同的方式也可定义n-步转移矩阵： 由n-步转移概率的性质（Chapman–Kolmogorov等式）可知，n-步转移矩阵是其之前所有转移矩阵的连续矩阵乘法：

$$\boldsymbol{P}^{(t)} = \boldsymbol{P}^{(t-1)} \dots \boldsymbol{P}^{(1)} \boldsymbol{I}$$

马尔科夫链 - 状态转移概率矩阵

假设我们有三种状态：晴天、多云和雨天。这个马尔可夫链的转移矩阵可能如下所示：



	sunny	cloudy	rainy
sunny	0.8	0.1	0.1
cloudy	0.4	0.4	0.2
rainy	0.2	0.3	0.5

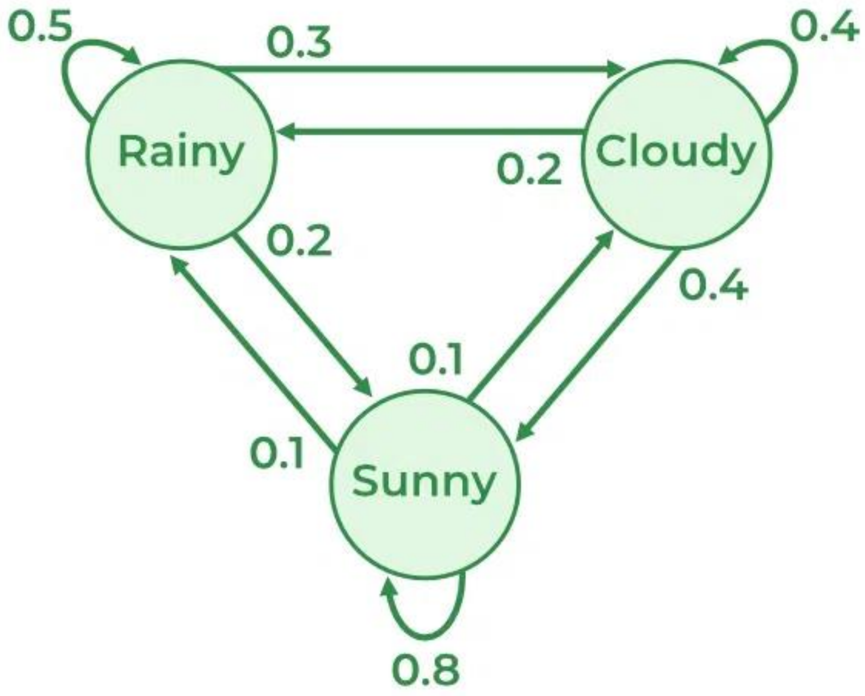
$$P = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}$$

马尔科夫链 - 状态转移概率矩阵

$$\pi(0) = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$

$$\pi(1) = P\pi(0) = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix} = \begin{bmatrix} 0.24 \\ 0.32 \\ 0.36 \end{bmatrix}$$

$$\pi(2) = P^2\pi(0) = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.4 & 0.4 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}^2 \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$



隐马尔可夫模型（Hidden Markov Model，HMM）

隐马尔可夫模型（Hidden Markov Model，HMM）是一种用于处理时间序列数据的统计模型，特别适用于存在潜在（隐藏）状态的情况。它假设系统是一个马尔可夫过程，且观测值是从该过程的隐藏状态生成的。HMM主要用于序列数据分析，如语音识别、生物信息学、金融时间序列分析等。

基本概念

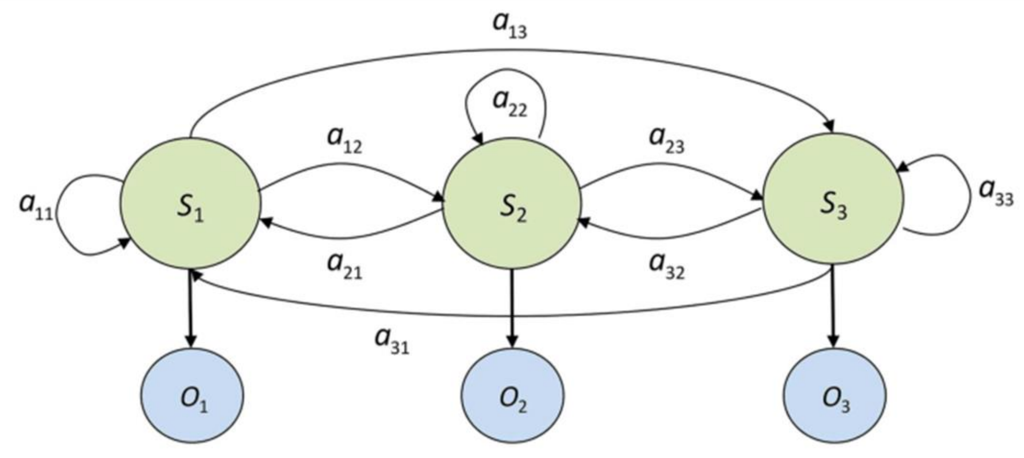
1.隐藏状态（Hidden States）： 系统的实际状态，无法直接观察。假设有N个隐藏状态。

2.观测值（Observations）： 可以直接观察到的输出。假设有M种可能的观测值。

3.状态转移概率（State Transition Probabilities）： 从一个隐藏状态转移到另一个隐藏状态的概率。

4.观测概率（Emission Probabilities）： 在某一隐藏状态下生成某一观测值的概率。

5.初始状态概率（Initial State Probabilities）： 系统在时刻t=1处于某一隐藏状态的概率。



HMM的三要素

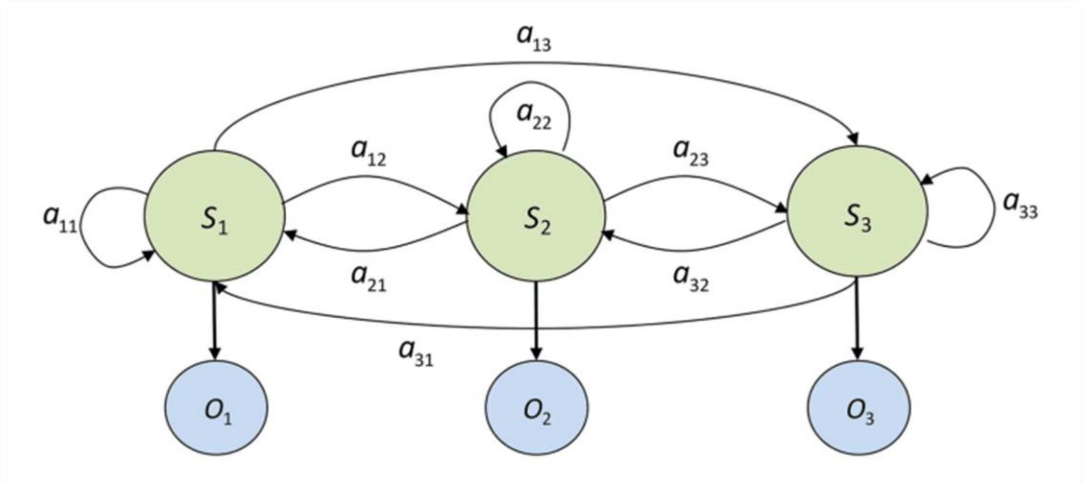
HMM由**三组参数定义：** $\boldsymbol{\lambda = (A, B, \pi)}$。隐马尔可夫模型由初始状态向量π、状态转移矩阵A和观测概率矩阵B决定。π和A决定状态序列，B决定观测序列。

模型表示

- S 是所有可能状态的集合：$S = \{s_1, s_2, \cdots, s_N\}$
- V 是所有可能观测的集合：$V = \{v_1, v_2, \cdots, v_M\}$
- I 是长度为T的状态序列：$I = (i_1, i_2, \cdots i_T)$
- O 是对应的观测序列： $O = (o_1, o_2, \cdots o_T)$

HMM 三组参数： $\lambda = (A, B, \pi)$。

- A 是状态转移矩阵，大小为 $N \times N$。
- B 是观测概率矩阵，大小为 $N \times M$。
- π 是初始状态概率向量，大小为 N 。



- N是可能的状态数
- M是可能的观测数

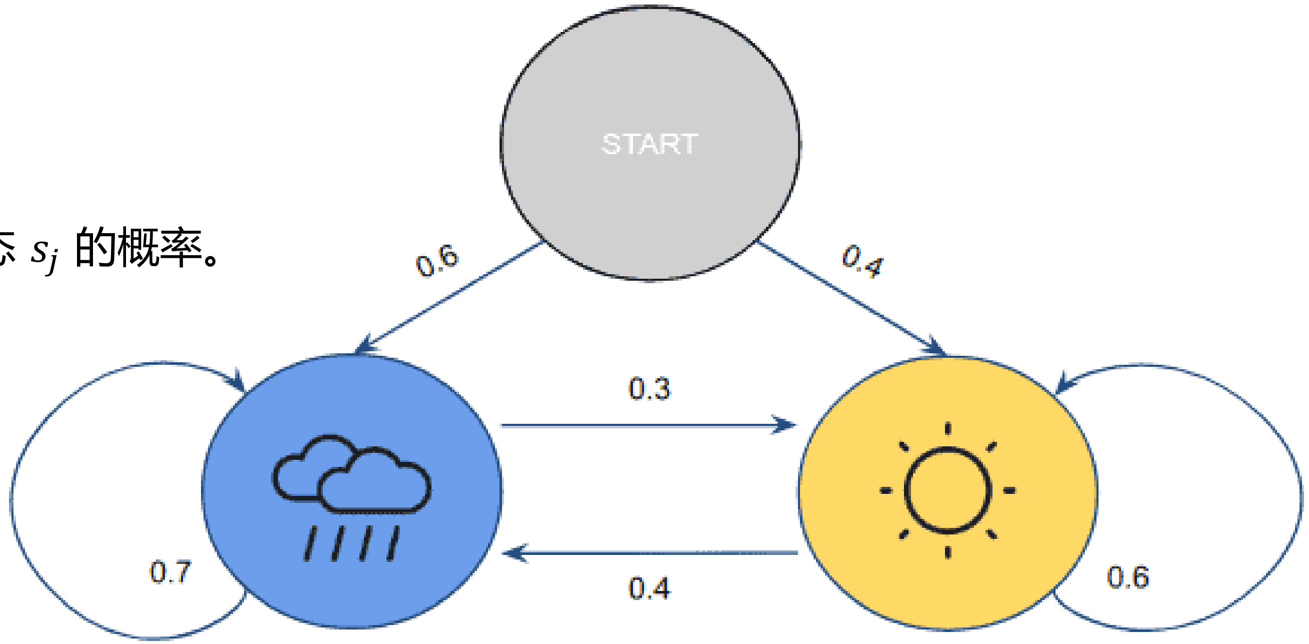
HMM的三要素 - 状态转移概率矩阵 A

状态转移概率：从一个状态到另一个状态的概率被称为转移概率。通常，这些概率使用转移矩阵来定义。

A 是状态转移矩阵，大小为 $N \times N$。

- 通常记为矩阵 A = $[a_{ij}]_{N\times N}$
- 其中 $a_{ij} = P(y_t = s_j \mid y_{t-1} = s_i)$, $1 \le i,j \le N$。
  表示在时刻 (t - 1) 处于状态为 s_i ，则在下一时刻t状态 s_j 的概率。

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$



- N是可能的状态数
- M是可能的观测数

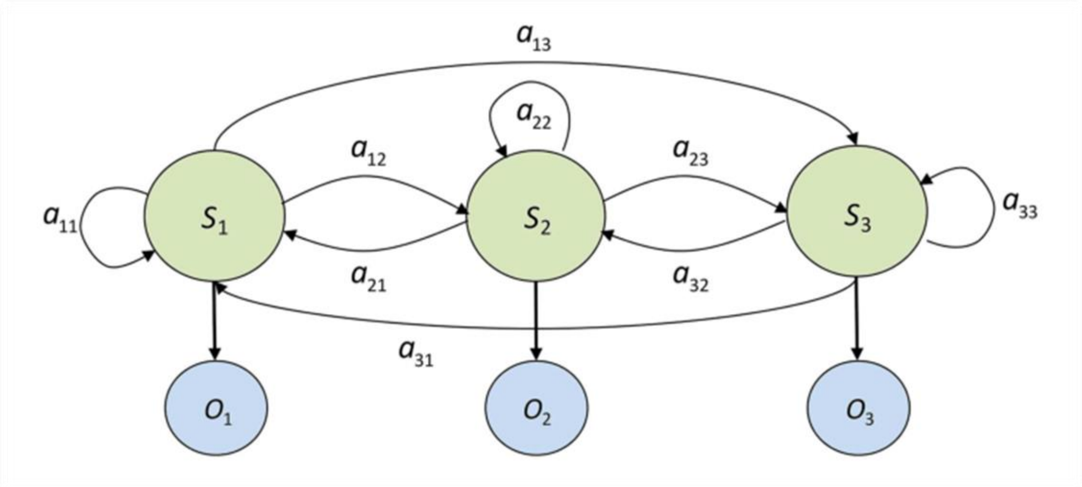
HMM的三要素 – 观测概率矩阵 B

观测概率：即根据当前状态获得各个观测值的概率。

B 是观测概率矩阵，大小为 $N \times M$：

- 通常记为矩阵 $\boldsymbol{B} = \left[b_{ij}\right]_{N\times M}$
- 其中$b_{ij} = P(x_t = o_j \mid y_t = s_i), 1 \le i \le N, 1 \le j \le M$。

  表示在时刻 t 处于状态为 s_i,生成观测值 o_j 的概率（也叫生成概率和发射概率）.



- N是可能的状态数
- M是可能的观测数

HMM的三要素 – 初始状态概率向量 π

π 是初始状态概率向量，大小为 N 。

$$\pi = (\pi_1, \cdots, \pi_N)^{\mathrm{T}}$$

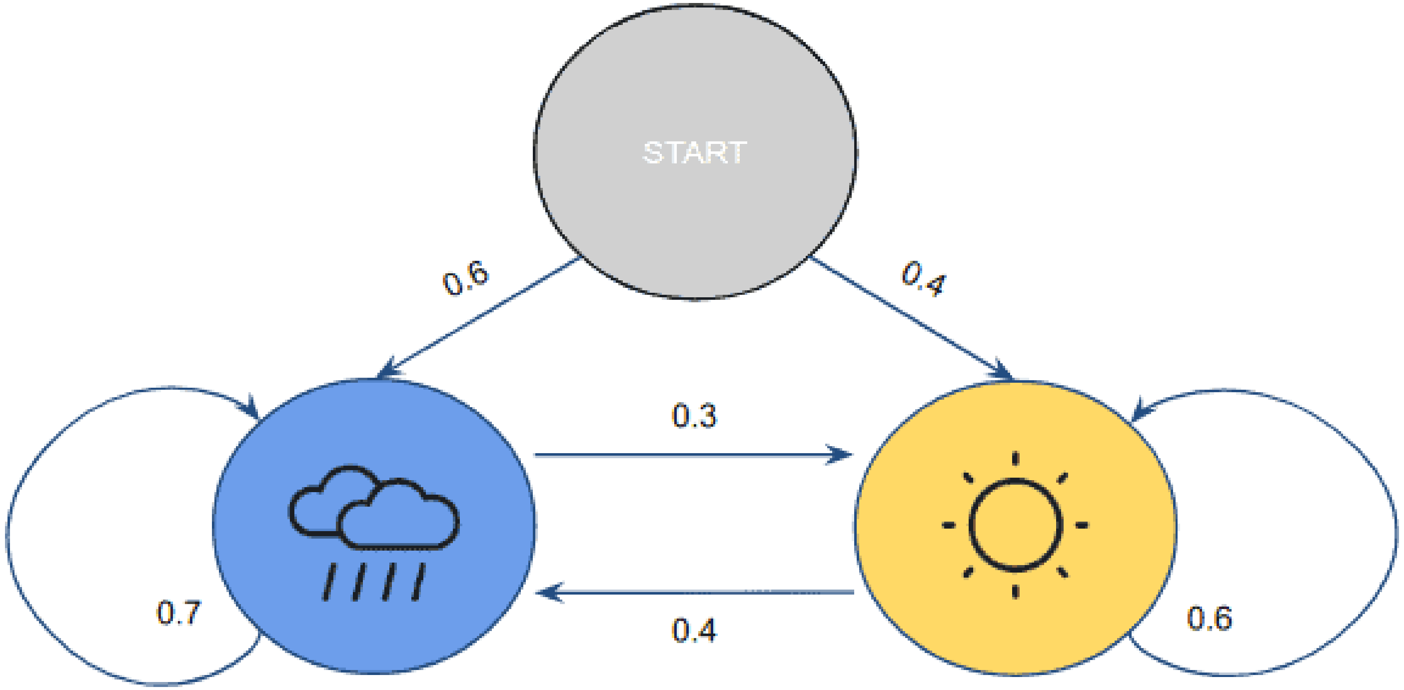
$$\pi_i = P(y_1 = s_i), 1 \leq i \leq N$$

$$0 \leqslant \pi_i \leqslant 1$$

$$\sum_{i=1}^{N} \pi_i = 1$$

π_i 是初始状态s_i 的概率。

$$\pi = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$



联合概率分布

- **马尔可夫假设：** 当前隐状态仅依赖于前一个隐状态。
- **观测独立性假设：** 当前观测仅依赖于当前马尔可夫链的状态，与其他隐状态和观测无关。

我们并不知道给我们观测结果的确切序列。但是，首先让我们回答特定隐藏状态序列发生的概率。由于马尔可夫性质，每个状态转换是独立的，我们可以简单地相乘：

$$p(S \mid \theta) = \prod_{t=1}^{T} p(s_t \mid s_{t-1})$$

因此，得到特定隐藏序列 S 和特定观测序列 O 的联合概率为：

$$p(O, S) = \prod_{t=1}^{T} p(O_t \mid s_t, \theta) \prod_{t=1}^{T} p(s_t \mid s_{t-1})$$

所有变量的联合概率分布更详细的写法：

$$P(o_1, s_1, \cdots, o_T, s_T) = P(s_1) P(o_1 | s_1) \prod_{t=2}^{T} P(o_t | s_t) P(s_t | s_{t-1})$$

联合概率分布

然而，我们需要考虑所有可能生成观测序列的状态序列组合。为此，我们可以对所有可能的隐藏状态序列的联合概率进行边缘化处理，并通过求和得到它们的发生概率：

$$p(O \mid \theta) = \sum_{q=1}^{Q} p(O, S_q \mid \theta)$$

$$= \sum_{q=1}^{Q} \prod_{t=1}^{T} p(O_t \mid s_t, \theta) \prod_{t=1}^{T} p(s_t \mid s_{t-1})$$

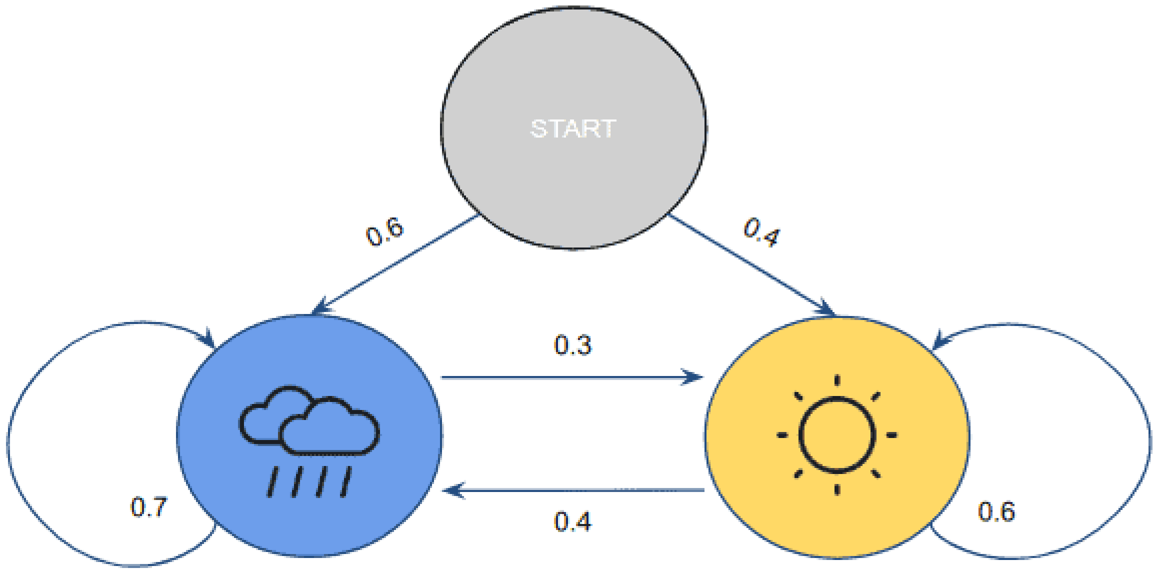
其中 Q 是所有可能的隐藏状态序列的数量。
计算这个结果并不困难，但计算复杂度在实际情况下非常高。如果隐藏状态的数量是 N，则有 N^T个可能的状态序列。对于每个序列，我们需要进行 2T−1 次相乘。最后，我们需要对 $N^T - 1$个操作求和：$(2T-1)N^T + N^T - 1$，或 $O(TN^T)$。

天气例子

由于Alice对Bob所在地区的天气有一个概念，我们可以将她的知识转化为概率：

如果昨天是晴天，那么今天再次晴天的概率是 0.6 ， 下雨的概率是 0.4 。
如果昨天下雨，那么再次下雨的概率为 0.7 ， 晴天的概率为 0.3 。

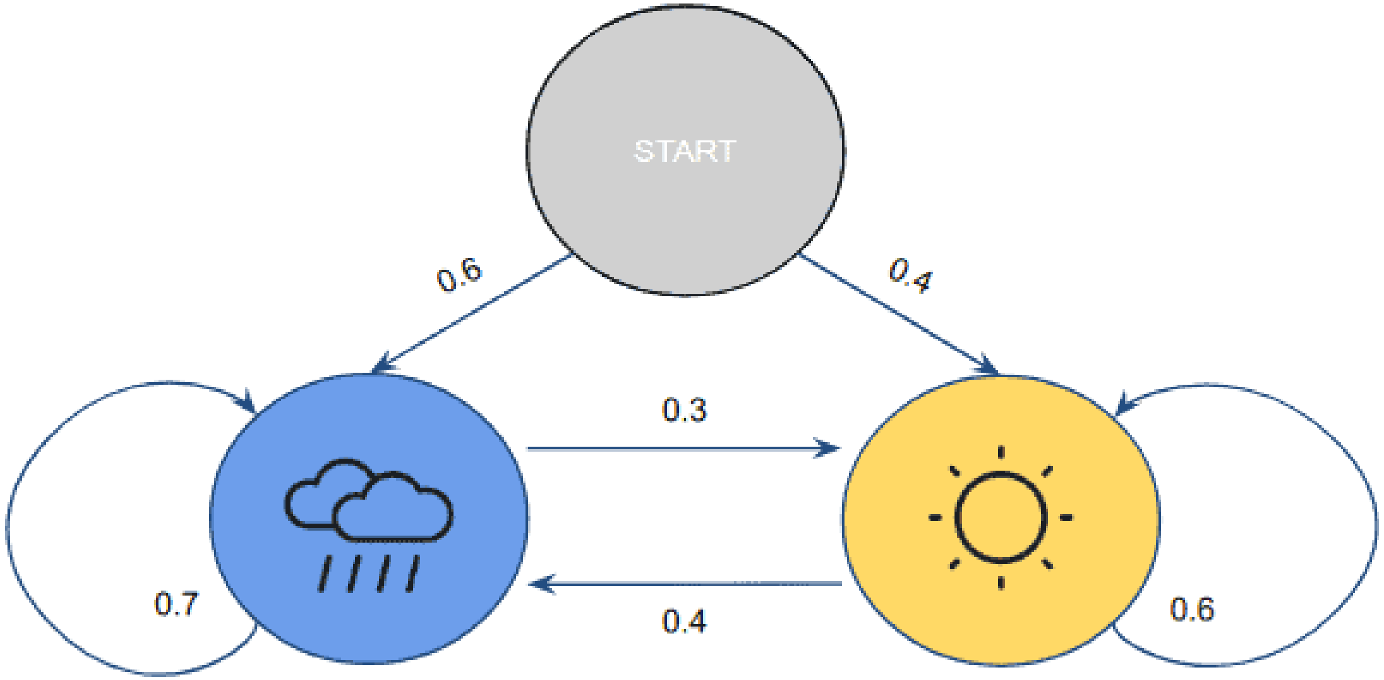
由于序列必须从某个地方开始，假定从雨天开始的概率是 0.6 ， 从晴天开始的概率是 0.4 。则模型看起来如图：



天气例子 – 初始状态概率

由于我们必须从某个地方开始，我们定义另一个包含初始概率的一维向量 π：

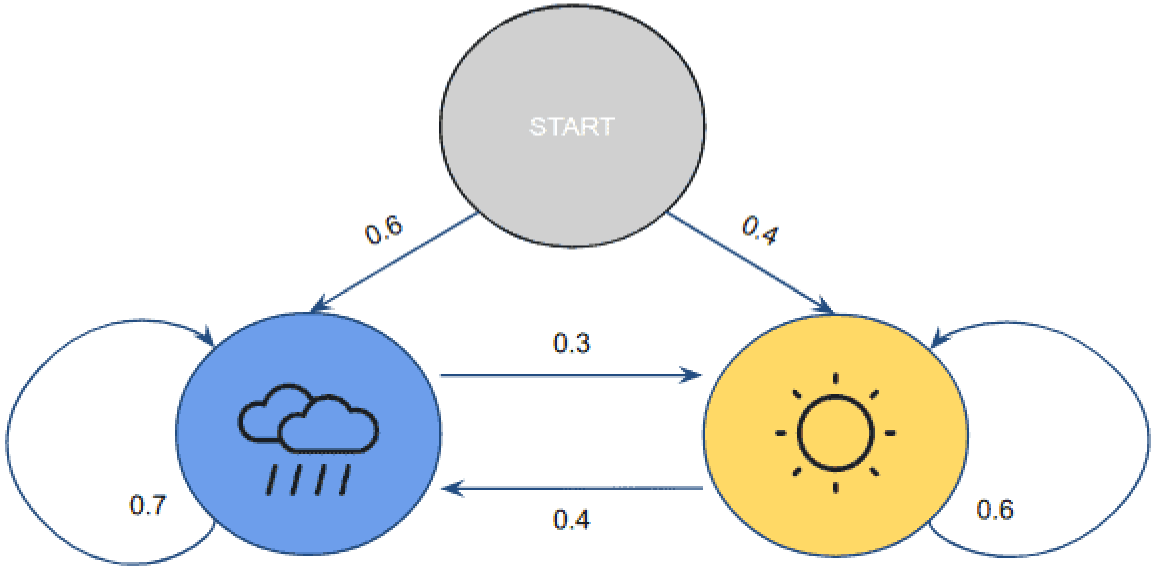
$$\pi = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$



天气例子 – 状态转移矩阵

在我们的例子中，转换矩阵是：

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$



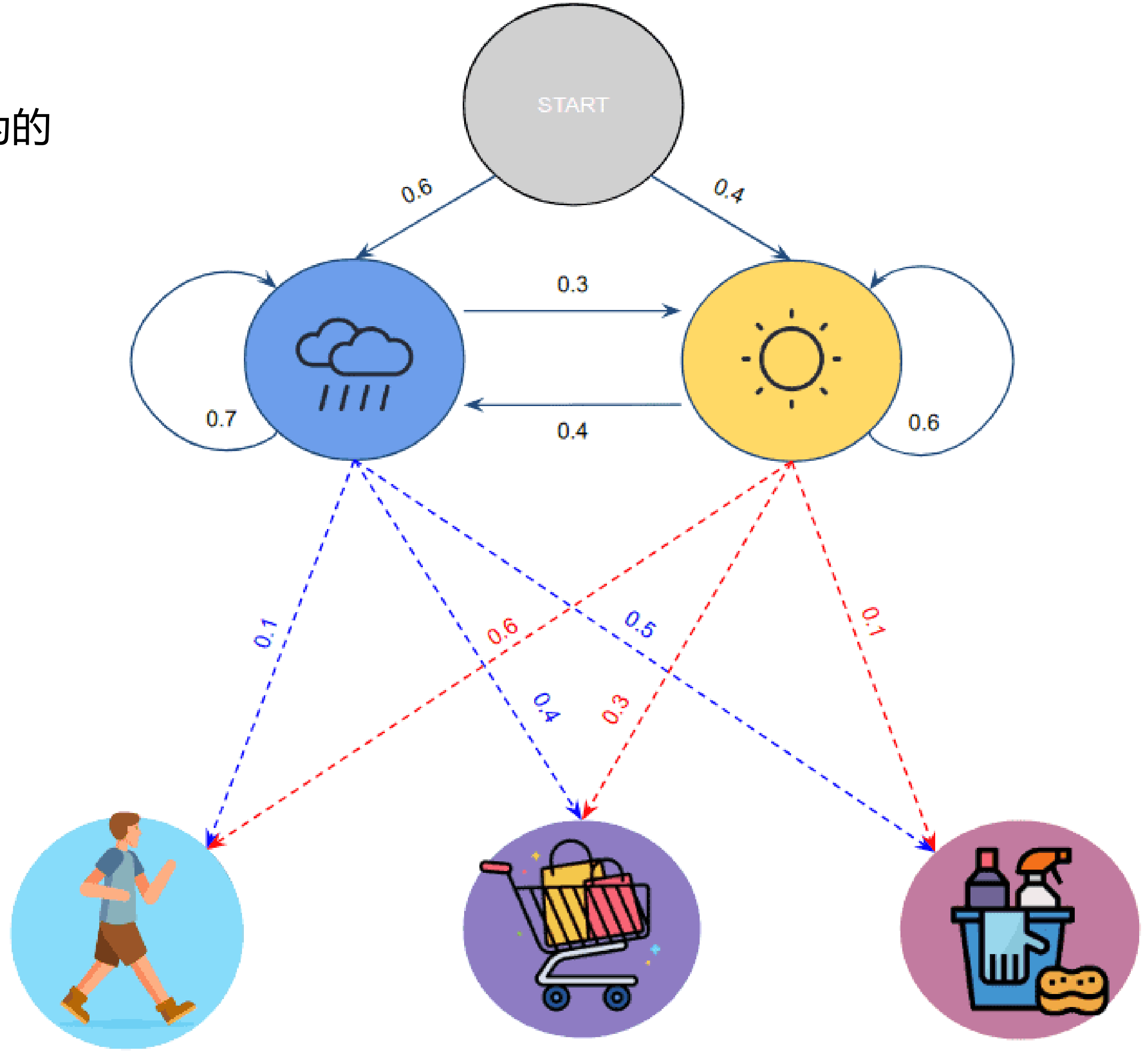
	rainy	Sunny
rainy	0.7	0.3
sunny	0.4	0.6

天气例子 – 发射（观测）概率矩阵

由于 Alice 每天都和 Bob 交谈，她已经收集了一系列对他行为的观察，并且知道得更清楚。根据她的观察，一组概率组合是：

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} = \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

	Walk	*Shopping*	*Cleaning*
rainy	0.1	0.4	0.5
sunny	0.6	0.3	0.1



天气例子总结

天气行为模型 $\lambda = (A, B, \pi)$

状态集合 Q = $(Sunny, Rainy)$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

观测集合 V = $(Walk, Shopping, Cleaning)$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} = \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

$$\pi = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$

状态集合 Q = $(Sunny, Rainy)$

HMM 的三个基本问题

问题1： 评估[Evaluation]（评估观测序列概率）
如何评估模型与观测序列之间的匹配程度?
给定模型 $\lambda=(A,B,\pi)$ 和观测序列 $O=(o_1,o_2,\cdots o_T)$ ，计算在模型 λ 下观测序列 O 出现的概率 $P(O|\lambda)$
解决方法:
- 暴力算法（穷举法）
- 前向算法
- 后向算法

问题2： 学习[learning]（参数估计）
如何训练模型使其能最好地描述观测数据?
给定观测序列 $O=(o_1,o_2,\cdots o_T)$ ，估计模型 $\lambda=(A,B,\pi)$ 的参数，使得模型 λ 下观测序列的条件概率 $P(O|\lambda)$ 最大
解决方法:
- 监督学习（极大似然直接估计）
- 非监督学习（Baum-Welch算法迭代估计）

问题3： 解码[Decoding]（序列可能性问题）
如何根据观测序列推断出隐藏的模型状态?
给定模型 $\lambda=(A,B,\pi)$及观测序列 $O=(o_1,o_2,\cdots o_T)$ ，求最有可能的状态序列。
解决方法:
- 贪心近似算法
- 维特比算法（Viterbi algorithm）

1. 评估[Evaluation] - 暴力算法（穷举法）

给定模型 $\lambda=(A,B,\pi)$ 和观测序列 O 的情况下，求在模型 $\lambda=(A,B,\pi)$ 下观测序列O出现的概率 $P(O|\lambda)$。

暴力算法（穷举法）： 对所有可能的状态序列和观测序列的联合概率求和。
列举所有可能的状态序列，则状态序列的概率为:

$$P(I|\lambda)=\pi_{i_1}a_{i_1i_2}a_{i_2i_3}\cdots a_{i_{T-1}i_T}$$

对于固定的状态序列，观测序列O的概率为:

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

则在模型λ下观测序列O出现的概率 $P(O|\lambda)$ 为:

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

穷举法的时间复杂度过高，可以采用动态规划算法——前向后向算法进行优化。

1. 评估[Evaluation] - 前向算法（Forward Algorithm）

前向算法（Forward Algorithm）： 计算从初始状态到观测序列中每个时刻的部分观测概率。
输入：隐马尔可夫模型入，观测序列 O；
输出：观测序列概率 $P(O|\lambda)$。

1. 初值计算：t=1的观测联合概率

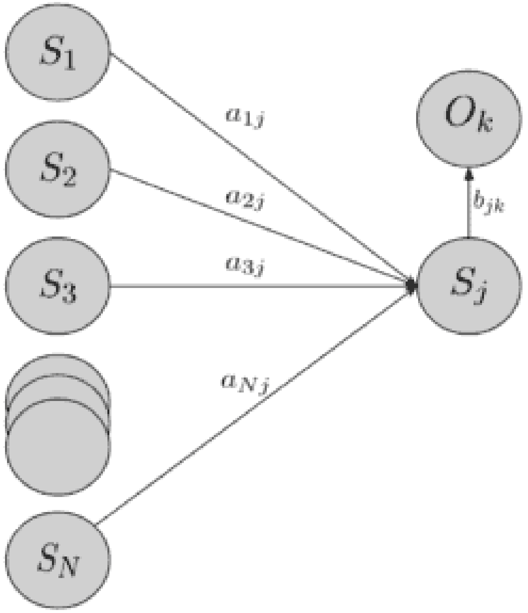
$$\delta_1(i) = \pi_i b_i(o_1) \quad i = 1,2,\cdots,N$$

2. 递推计算：t=2，3，…，T 的观测联合概率

$$\delta_t(i) = \left[\sum_{j=1}^{N} \delta_{t-1}(j) a_{ji}\right] b_i(o_t) i = 1,2,\cdots,N$$

3. 终止计算：累加 t=T 的观测联合概率

$$P(O|\lambda) = \sum_{i=1}^{N} \delta_T(i)$$

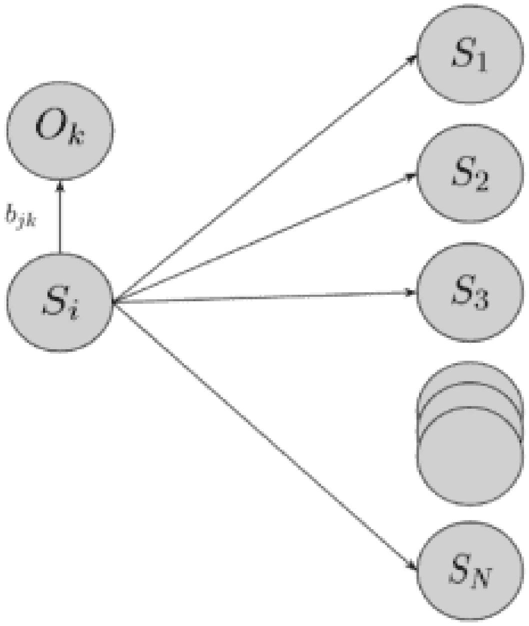


1. 评估[Evaluation] - 后向算法（Backward Algorithm）

后向算法（Backward Algorithm）： 计算从观测序列某个时刻到结束的部分观测概率，递归公式如下：

$$\beta_t(i) = \sum a_{ij}\, b_j(O_{t+1})\beta_{t+1}(j)$$

终止条件为： $\beta_T(i) = 1$



1. 评估[Evaluation] - 前向算法（Forward Algorithm） - 计算例子

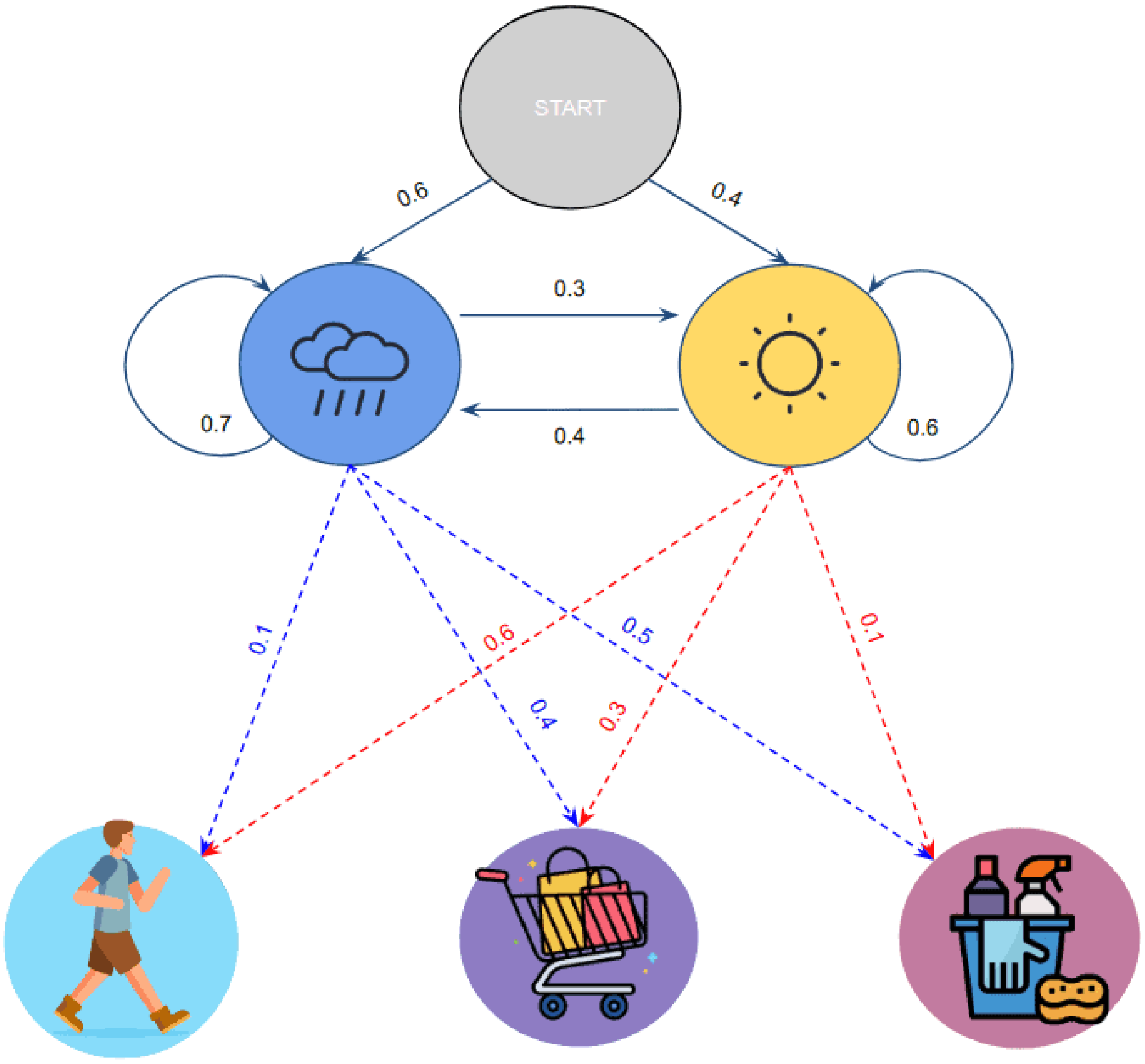
天气行为模型 $\lambda = (A, B, \pi)$　　状态集合 Q = $(Sunny, Rainy)$ $N = 2$　　观测集合 V = $(Walk, Shopping, Cleaning)$

设 T=3，O=$(Walk, Shopping, Cleaning)$，计算$P(O|\lambda)$：

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} = \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

$$\pi = \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$



1. 初值计算：

$$\delta_1(1) = \pi_1 b_1(o_1) = 0.6 * 0.1 = 0.06$$
$$\delta_1(2) = \pi_2 b_2(o_1) = 0.4 * 0.6 = 0.24$$

1. 评估[Evaluation] - 前向算法（Forward Algorithm） - 计算例子

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} = \begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

$$\delta_1(1) = \pi_1 b_1(o_1) = 0.6 * 0.1 = 0.06$$
$$\delta_1(2) = \pi_2 b_2(o_1) = 0.4 * 0.6 = 0.24$$

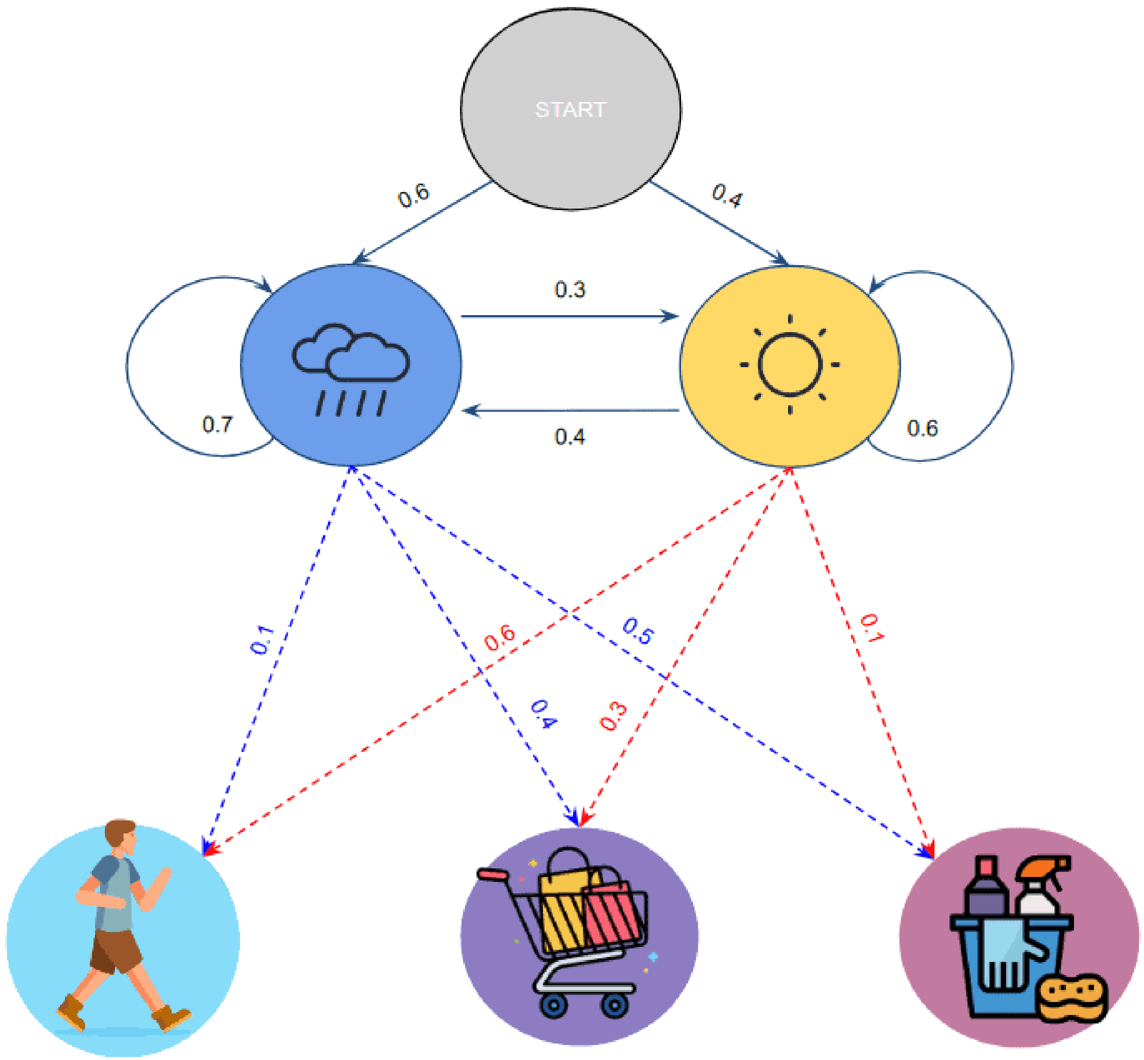
2. 递推计算： t=2，3，…，T 的观测联合概率

$$\delta_2(1) = \left[\sum_{i=1}^{2} \delta_1(i) a_{i1}\right] b_1(o_2) = 0.138 * 0.4 = 0.0552$$

$$0.06 * 0.7 + 0.24 * 0.4 = 0.138$$

$$\delta_2(2) = \left[\sum_{i=1}^{2} \delta_1(i) a_{i2}\right] b_2(o_2) = 0.192 * 0.3 = 0.0576$$

$$0.06 * 0.3 + 0.24 * 0.6 = 0.162$$

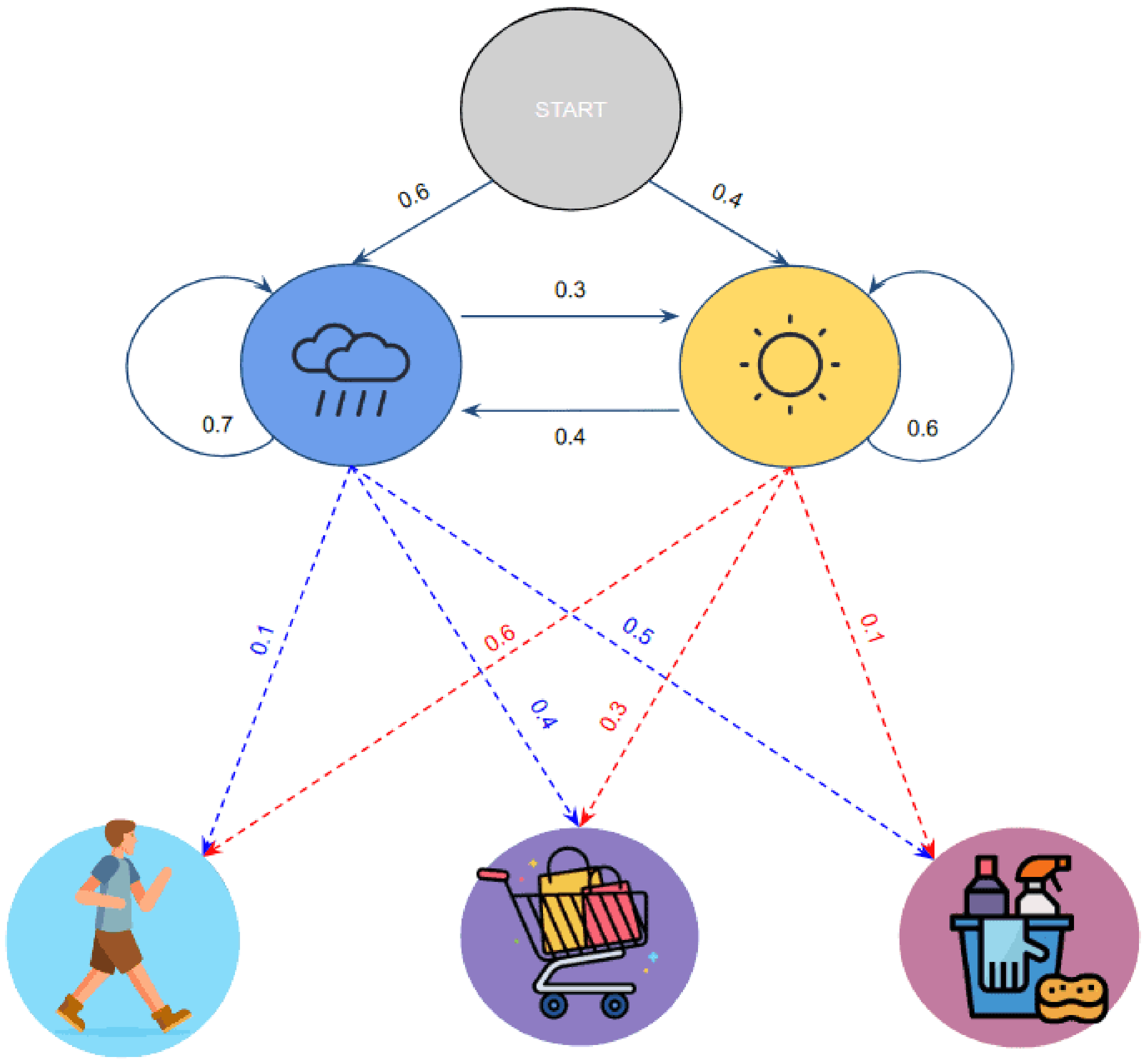


1. 评估[Evaluation] - 前向算法（Forward Algorithm） - 计算例子

$$B=\begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix}=\begin{bmatrix} 0.1 & 0.4 & 0.5 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

$$A=\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}=\begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}$$

$$\delta_2(1)=0.138*0.4=0.0552$$
$$\delta_2(2)=0.192*0.3=0.0576$$



2. 递推计算：t=2，3，…，T 的观测联合概率

$$\delta_3(1)=\left[\sum_{i=1}^{2}\delta_2(i)a_{i1}\right]b_1(o_3)=0.06168*0.5=0.03084$$

$$0.0552*0.7+0.0576*0.4=0.06168$$

$$\delta_3(2)=\left[\sum_{i=1}^{2}\delta_2(i)a_{i2}\right]b_2(o_3)=0.05112*0.1=0.005112$$

$$0.0552*0.3+0.0576*0.6=0.05112$$

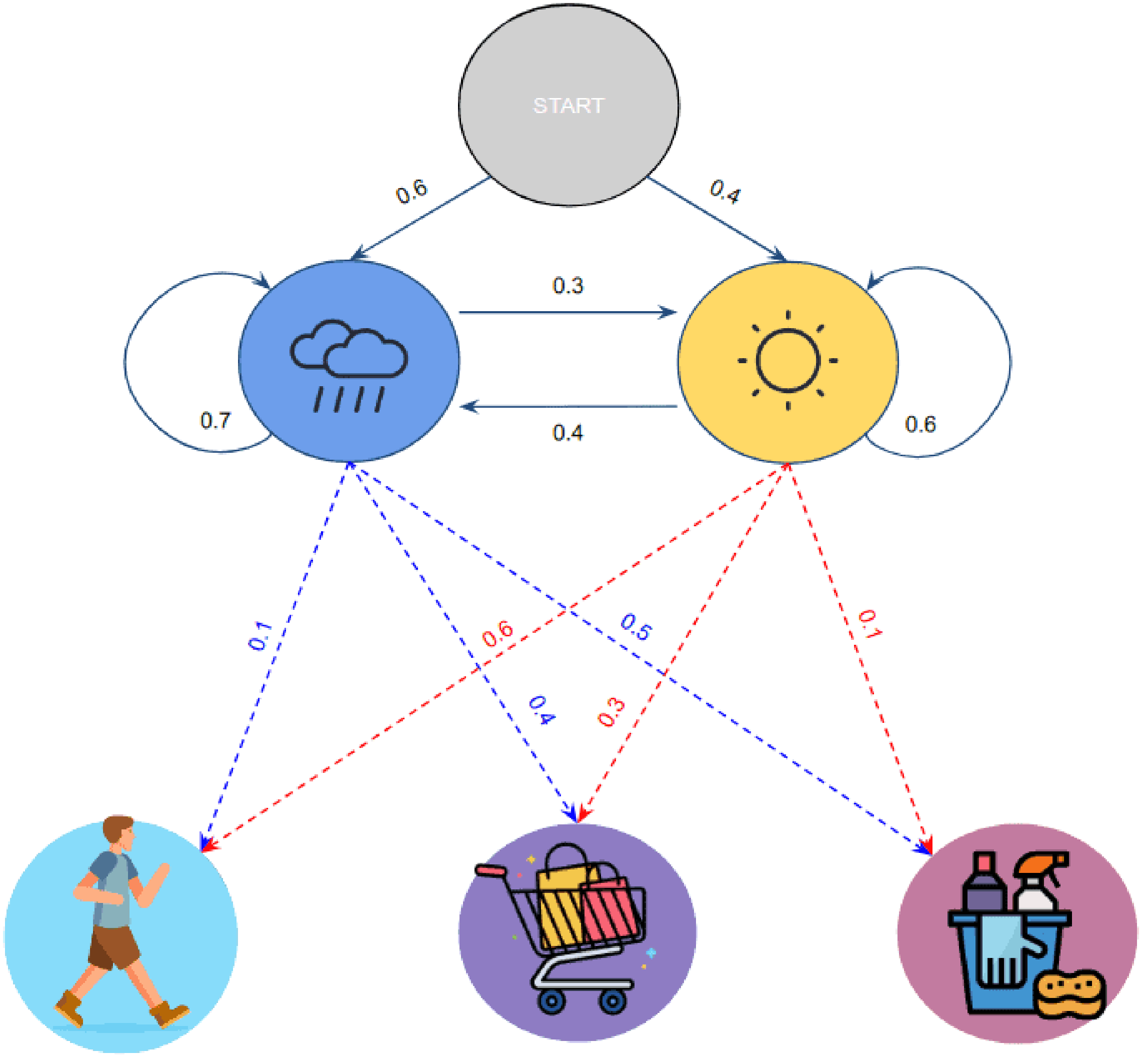
1. 评估[Evaluation] - 前向算法（Forward Algorithm） - 计算例子

$$\delta_3(1) = \left[\sum_{i=1}^{2} \delta_2(i)a_{i1}\right] b_1(o_3) = 0.06168 * 0.5 = 0.03084$$

$$0.0552 * 0.7 + 0.0576 * 0.4 = 0.06168$$

$$\delta_3(2) = \left[\sum_{i=1}^{2} \delta_2(i)a_{i2}\right] b_2(o_3) = 0.05112 * 0.1 = 0.005112$$

$$0.0552 * 0.3 + 0.0576 * 0.6 = 0.05112$$



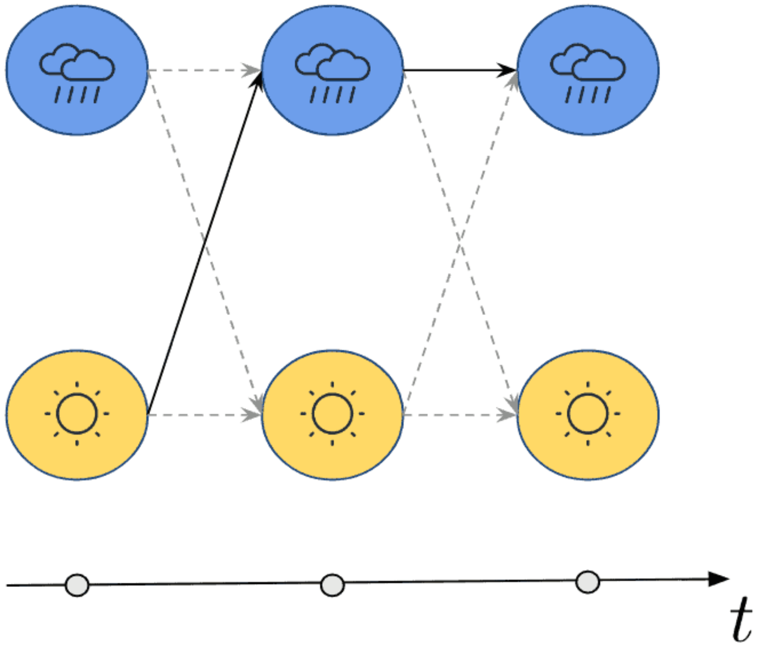
3. 终止计算：累加 t=T 的观测联合概率

$$P(O|\lambda) = \sum_{i=1}^{2} \delta_3(i) = 0.03084 + 0.005112 = 0.035952$$

2. 学习[learning]（参数估计）- Baum-Welch算法

Baum-Welch算法： 迭代算法，用于估计模型参数。基于EM算法，通过不断调整模型参数，使观测序列的概率最大化。包括E步（计算期望值）和M步（最大化参数）。

输入： 观测序列 $O=(o_1,o_2,\cdots o_T)$；
输出： 隐马尔可夫模型参数。



3. 解码[Decoding]（序列可能性问题）- 维特比算法（Viterbi Algorithm）

维特比算法（Viterbi Algorithm）： 动态规划算法，用于找到最可能的隐藏状态序列。
输入： 模型 $\lambda = (A, B, \pi)$，观测序列 $O = (o_1, o_2, \cdots o_T)$；
输出： 最优路径 。

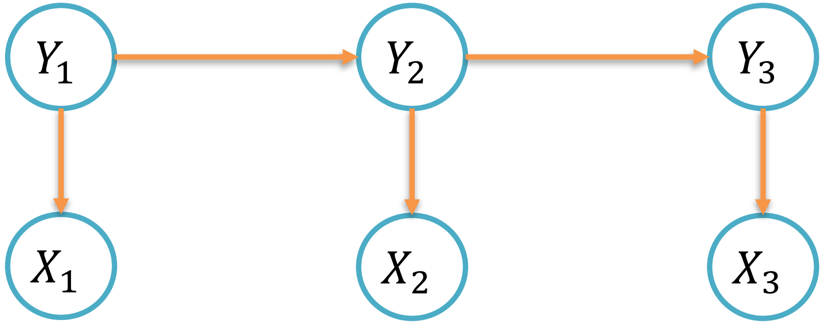
递归公式如下:

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

初始条件为: $\delta_1(j) = \pi_j b_j(O_1)$

Thank You

HMM的三要素及两个假设



联合概率可以分解为:

$$p(x,y;\theta)=\prod_{t=1}^{T}\underbrace{p(y_t|y_{t-1},\theta_s)}_{\text{转移概率}}\underbrace{p(x_t|y_t,\theta_t)}_{\text{输出概率}}$$

天气例子 – 推断天气

但爱丽丝如何利用它来推断天气呢？她需要回答以下问题：给定我们的模型，哪一系列隐藏状态最能解释给定的观测序列？

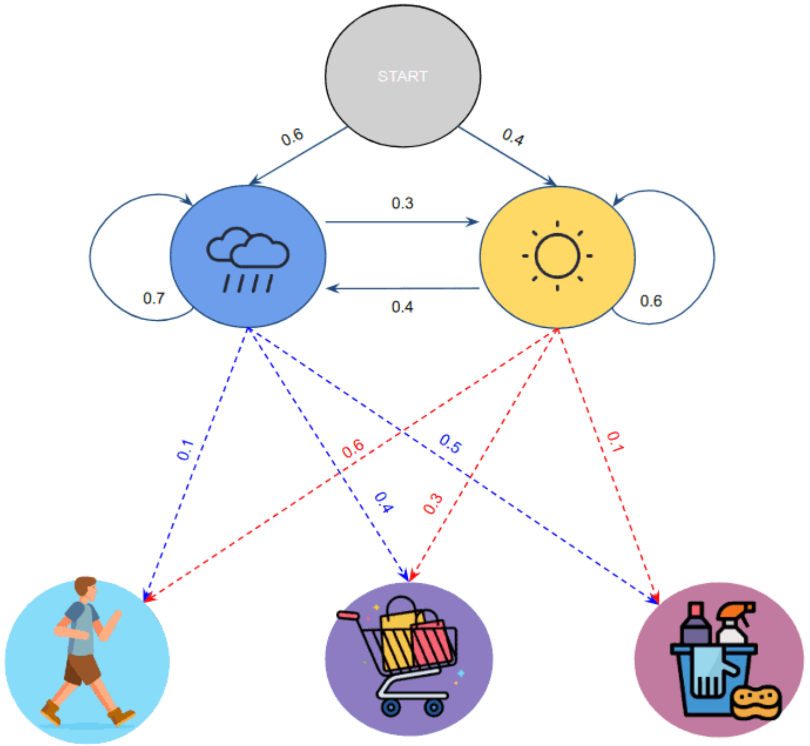
为了更好地理解上述问题，我们先尝试回答它的逆问题：这个特定模型生成给定观测序列的概率是多少？

我们加入一些数学符号。我们有一个观测序列 $O = \{O_1, O_2, ..., O_T\}$，表示 T 个时间步骤，或者在我们的例子中与鲍勃交谈的天数。我们也有以参数 A, B, π 表示的模型，可以简写为 θ。因此，“这个特定模型生成给定观测序列的概率是多少？”这个问题可以表示为：

$$p(O_T \mid \theta) = ?$$

天气例子 – 推断天气

要回答这个问题，首先想象一个生成观测序列的特定隐藏状态序列 S。例如：
如果我们有 T=3 天的观测 O=$(Walk, Shopping, Cleaning)$,
一个可能的隐藏状态序列是 S = $(Sunny, Sunny, Rainy)$。
在给定 S 的情况下，看到这个观测序列的概率是多少?



我们只需要将给定时间 s_t 的隐藏状态下看到特定观测 O 的概率相乘：

$$p(O_T \mid S_T, \theta) = \prod_{t=1}^{T} p(O_t \mid s_t, \theta)$$

这可以很容易地通过查找序列中每个隐藏状态的 B 中的特定发射概率来计算。对于示例 $S_3 = (Sunny, Sunny, Rainy)$：

$$p(Walk, Shopping, Cleaning \mid Sunny, Sunny, Rainy) = p(Walk \mid Sunny) * p(Shopping \mid Sunny) * p(Cleaning \mid Rainy) = 0.$$