CS 6300

Viterbi Algorithm Practice

April 12, 2022

You are trying to diagnose whether your computer is broken or not. On a given day, your computer's hidden state is either *broken* or *working*. Each day you make one of the following observations: *blue-screen*, *slow*, or *snappy*, depending on the state of your computer. You decide to use the following HMM to model your daily observations:

Initial Distribution

State	$P(X_\bullet)$
working	0.9
broken	0.1

Transition Distribution

State	Next State	$P(X_{t+1}\|X_t)$
working	*working*	0.9
working	*broken*	0.1
broken	*broken*	1.0
broken	*working*	0.0

Emission Distribution

State	Observation	$P(O_t\|X_t)$
working	*snappy*	0.7
working	*slow*	0.2
working	*blue-screen*	0.1
broken	*snappy*	0.1
broken	*slow*	0.4
broken	*blue-screen*	0.5

What is the most likely sequence of hidden states X_1, X_2, X_3 given the observation sequence (*snappy, slow, blue-screen*)?

The Viterbi algorithm from the course notes has the recursive relationship:

$$\begin{aligned} m_1[x_1] &= P(e_1|x_1)P(x_1) \\ m_t[x_t] &= P(e_t|x_t)\max_{x_{t-1}} P(x_t|x_{t-1})m_{t-1}[x_{t-1}] \end{aligned}$$

Day 1:

$$\begin{aligned} m_1[working] &= P(snappy|working)P(working) = 0.7 * 0.9 = 0.63 \\ m_1[broken] &= P(snappy|broken)P(broken) = 0.1 * 0.1 = 0.01 \end{aligned}$$

Day 2:

$$\begin{aligned} m_2[working] &= P(slow|working)\max\begin{cases} P(working|working)m_1[working] \\ P(working|broken)m_1[broken] \end{cases} \\ &= 0.2\max\{0.9 * 0.63, 0.0 * 0.01\} = 0.113 \\ m_2[broken] &= P(slow|broken)\max\begin{cases} P(broken|working)m_1[working] \\ P(broken|broken)m_1[broken] \end{cases} \\ &= 0.4\max\{0.1 * 0.63, 1.0 * 0.01\} = 0.0252 \end{aligned}$$

Day 3:

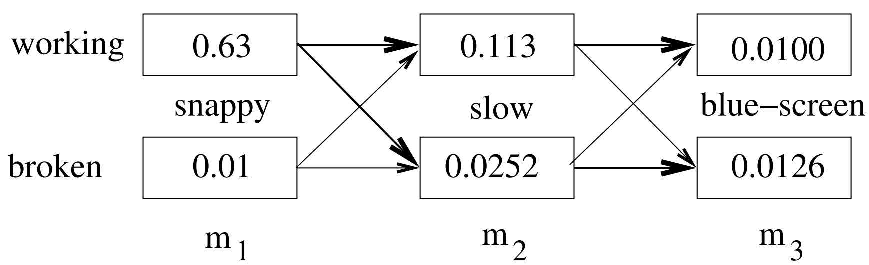
$$m_3[working] = P(blue-screen|working) \max \begin{cases} P(working|working)m_2[working] \\ P(working|broken)m_2[broken] \end{cases}$$

$$= 0.1 * \max\{0.9 * 0.113, 0.0 * 0.0252\} = 0.0100$$

$$m_3[broken] = P(blue-screen|broken) \max \begin{cases} P(broken|working)m_2[working] \\ P(broken|broken)m_2[broken] \end{cases}$$

$$= 0.5 * \max\{0.1 * 0.113, 1.0 * 0.0252\} = 0.0126$$

Fill in the appropriate m_i values in the trellis below. Emphasize the back pointers by thickening the edges in the trellis from the final m_3 values for both states *working* and *broken*.



working, broken, broken is the optimal sequence.