# e-PGPathshala

# Subject : Computer Science

# Paper: Machine Learning

# Module: HMM– Baum Welsh and Viterbi Algorithms

# Module No: CS/ML/35

# Quadrant I – e-text

Welcome to the e-PG Pathshala Lecture Series on Machine Learning.In this module we will discuss the Hidden Markov Model in detail and discuss the three issues associated with HMM

## Learning Objectives:

The learning objectives of this module are as follows:

- To understand the three issues of HMM
- To explain the Baum Welsh algorithm for evaluation using the model
- To discuss Viterbi algorithm for decoding

## 35.1 Recap: Hidden Markov Model

As we have already discussed Hidden Markov Model (HMM) is an extension of a Markov model in which the input symbols are not the same as the states.This means we don't know which state we are in (hence called Hidden State). For example in HMM POS-tagging, the input symbols are the words, states are the part of speech tags. In addition we make two important assumptions namely Markov assumption and Output-Input assumption. **Markov assumption** states that the state transition depends only on the origin and destination and the **Output-independent assumption** which states that all observation frames are dependent on the state that generated them, not on neighbouring observation frames.

### 35.1.1 Parameters of an HMM:

- In Hidden Markov Models, S = $\{s_1,\dots,s_n\}$ is a set of states where 'n' is the number of possible states
- **Transition probabilities is a two dimensional Matrix** *A=* $a_{1,1},a_{1,2},\dots,a_{n,n}$ *where each* $a_{i,j}$ *represents the probability of transitioning from state* $s_i$ *to* $s_j$*.*

- **Emission probabilities**or the observation symbol distribution probabilityis aset B of functions of the form $b_i$ ($o_t$) which is the probability of observation $o_t$being emitted or observed by state $s_i$
- **Initial state distribution**$\pi$is the probability that $s_i$ is a start state

Therefore we have two model parameters n, the number of states of the HMM and m, the number of distinct observation symbols per state. In addition we have three probability measures A-the transition probability, B-the emission probability and $\pi$- the initial probability.

The HMM Formalism is represented as given in Figure 35.1 where {*S, K*, P, *A, B*} is the model. Here S is the state and K is the observation. $\pi$ = {$p_i$} are theinitial state probabilities, *A* = {$a_{ij}$} are the state transition probabilities and *B* = {$b_{ik}$} are the observation or emission state probabilities.

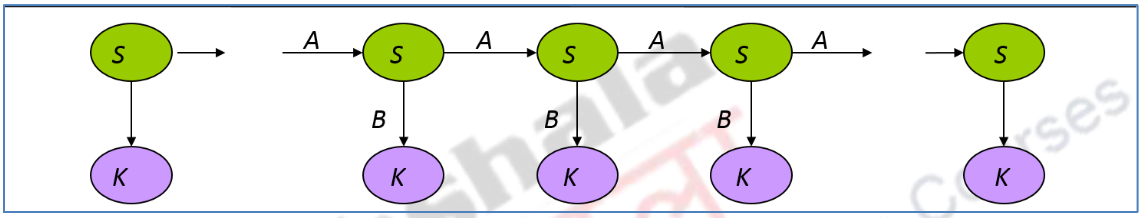


**Figure 35.1 The HMM Formalism**

### 35.1.2 Building the observation sequence

Before we proceed further, let us understand how an observation sequence is generated. Given the above five elements, we can build an observation sequence:

$$O = O_1 O_2 \dots O_T$$

where T is the number of observations. This observation sequence is built as follows:

1. We first set t=1.
2. Then we choose an initial state
   $$q_1 = S_i$$
   according to the initial distribution.
3. Then we choose
   $$O_t = V_k$$
   according to the symbol probability distribution.
4. Then we transit to a new state

$$q_{t+1} = S_j$$

according to the state transition probability distribution.

5. Then we set t  t=t+1, and iterate by returning to step 3 while t<=T

## 35.2 The Three Problems of HMM

The following are the three problems associated with HMM:

Problem 1: **Evaluation**

Here given the observation sequence $O=o_1,\dots,o_T$and an HMM model how do we compute the probability of O given the model?

**Q1**: **How do we compute the probability of a given sequence of observations?**

**A1:** Forward – Backward dynamic programming algorithm – **the Baum Welch algorithm**

Problem 2: **Decoding**

Here given the observation sequence $O=o_1,\dots,o_T$and an HMM model, how do we find the state sequence that best explains the observations?

**Q2:How to compute the most probable sequence of states, given a sequence of observations?**

**A2: Viterbi's** dynamic programming Algorithm

Given an observation sequence, compute the most likely hidden state sequence

Problem 3: **Learning**

Here given an observation sequence and set of possible models, which model most closely fits the data?How do we adjust the model parameters

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

to maximize

$$P(O \mid \lambda)$$

**Q3: Given an observation sequence and set of possible models, which model most closely fits the data?**

**A3: The Expectation Maximization (EM) heuristic.**

## 35.3 Markov Assumption with an Example

Let us discuss the Markov assumptions with an example. Let us assume that we have a sentence with n words. The Markov assumption states that probability of the occurrence of word $w_i$ at time t depends only on occurrence of word $w_{i-1}$ at time t-1.

The normal chain rule would be as follows:

$$P(w_1, \ldots, w_n) = \prod_{i=2}^{n} P(w_i \mid w_1, \ldots, w_{i-1})$$

However the Markov Assumption approximates the probability of the sequence of words as:

$$P(w_1, \ldots, w_n) \approx \prod_{i=2}^{n} P(w_i \mid w_{i-1})$$

A common way of representing the Hidden Markov Model is the Trellis Diagram shown in Figure 35.2

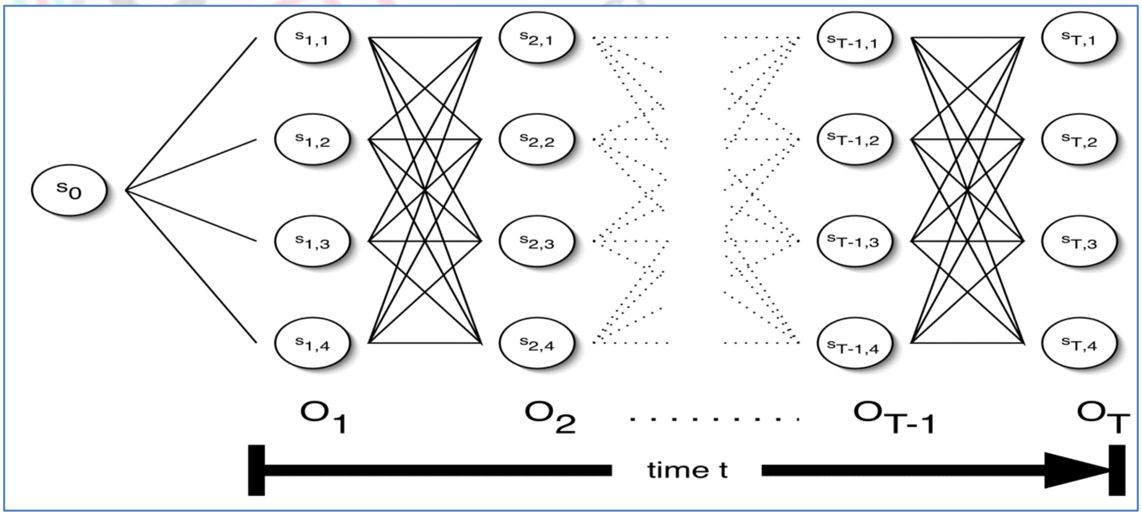


**Figure 35.2 Trellis Diagram**

In this diagram we have the starting state $s_0$ which influences the states $s_{1,1}$,$s_{1,2}$,$s_{1,3}$,$s_{1,4}$ with observation at time $t_1$ being $o_1$ and so on until finally we end up with the states $s_{T,1}$,$s_{T,2}$,$s_{T,3}$,$s_{T,4}$at time $t_T$.

## 35.4 Problem 1 - Evaluation - Forward and Backward Algorithms

Here given the observation sequence $O=o_1,\ldots,o_T$and an HMM model we need to compute the probability of O given the model that is we are given a sequence of observations and we need to compute the probability of that specific sequence of observations. This likelihood of a sequence can be determined by either the forward procedure or the backward procedure. We will discuss the Forward algorithm which is essentially a dynamic programming algorithm. Backward algorithm can be similarly explained. The Forward algorithm can be used with two options, one the “Any Path” methodwhere the likelihood is measured using any sequence of states of length T, and the second option the “Best Path” method where we choose an HMM by the probability generated using the best possible sequence of states. Solving the evaluation problem involves the determination of the probability that a particular sequence of symbols O was generated by that model (Figure 35.3). Here the model starts at state q1 with initial probability $\pi_{q1}$, followed by the transition probabilities $a_{q1,q2}$, …………….$a_{qT-1,qT.}$

$$P(Q|M) = \pi_{q_1} \cdot \prod_{t=1}^{T-1} a_{q_t,q_{t+1}} = \pi_{q_1} \cdot a_{q_1,q_2} \cdot a_{q_1,q_2} \cdots a_{q_{T-1},q_T}$$

$$P(O|Q,M) = \prod_{t=1}^{T} P(o_t | q_t, M) = b_{q_1,o_1} \cdot b_{q_2,o_2} \cdots b_{q_T,o_T}$$

$$\Rightarrow \quad P(O|M) = \sum_{allQ} P(O|Q,M) \cdot P(Q|M)$$

### 35.4.1 Forward Probabilities:

We need to determine the forward probability that, given an HMM λ, at time t the state is i and the partial observation $o_1 \ldots o_t$ has been generated that is

$$\alpha_t(i) = P(o_1 \ldots o_t, q_t = s_i | \lambda)$$

Here we start from the initial state and calculate the probability of each subsequent state in the forward direction, and hence this probability is called the forward probability.

$$\alpha_t(i) = P(o_1 \dots o_t, q_t = s_i \mid \lambda)$$

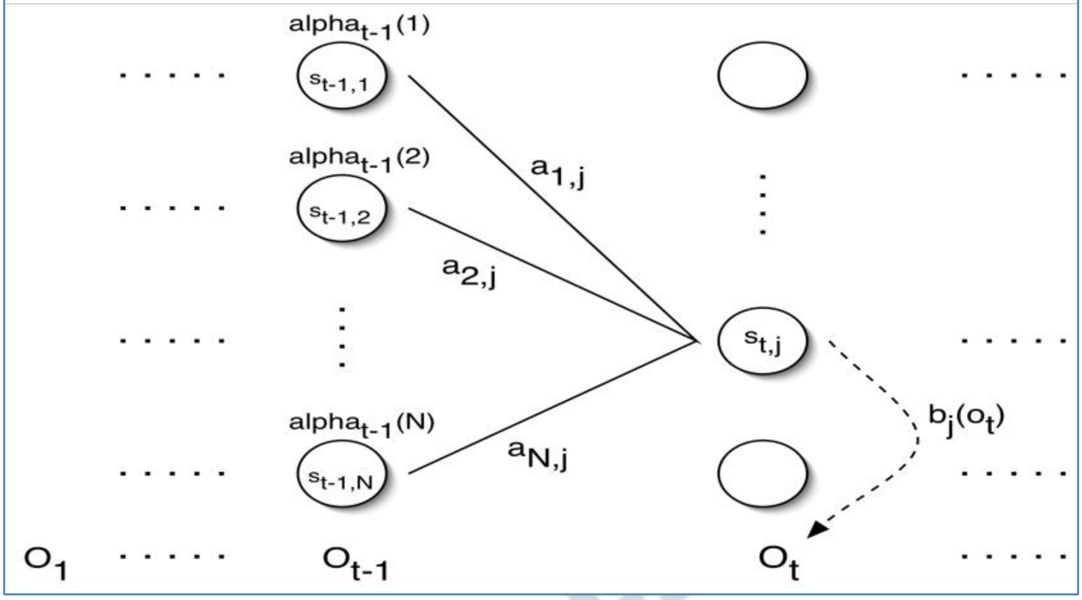


**Figure 35.4 Forward Probability**

The forward probability at a time slice t of a state j is the sum of each the N forward probabilities i at time slice t-1 multiplied by the transition probability of each state i at time slice t-1 to the state j under consideration t time slice t (Figure 35.4). This sum is then multiplied by the emission probability of observing $o_t$ at state j.

$$\alpha_t(j) = \left[ \sum_{i=1}^{N} \alpha_{t-1}(i)\, a_{ij} \right] b_j(o_t)$$

### 35.4.2 Forward recursion:

As we have already discussed forward probability

$$\boxed{\alpha_t(i) = P(o_1, \dots, o_t, q_t = s_i \mid M)}$$

is calculated using forward recursion. The initialization is as follows:

$$\boxed{\alpha_1(i) = \pi_i b_i(o_1)}$$

Here the initial forward probability of state i at time slice 1 is the product of the initial probability of state i and the probability of emitting the observation o1 at state i at time slice 1. The forward recursion is determined as follows:

$$\boxed{\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] b_j(o_{t+1})}$$

Here the forward probability at time slice t+1 is determined by considering the forward probabilities of all states at time slot t, the transition probabilities from state i to state j (the state whose forward probability is to be determined) and the emission probability of the observation at time slice t+1 at state j.

Finally we have the termination as follows:

$$P(O \mid M) = \sum_{i=1}^{N} P(o_1, o_2, \ldots, o_T, q_T = s_i \mid M) = \boxed{\sum_{i=1}^{N} \alpha_T(i)}$$

### 35.4.3 Example for the Calculation of Forward Probability

We show an example for the calculation of forward probabilities (Figure 35.5). Here there are three states which can emit the observations R,G and B. We have the observation sequence R,R,G,B. Now the initial probability shows that the start state is 1. The probability of state 1 emitting R is 0.6, G is 0.2, B is 0.2. The probabilities of state 2 and state 3 being the initial state is 0. The probabilities of state 2 emitting R is 0.2, G is 0.5, B is 0.3.and the probabilities of state 3 emitting R is 0.0, G is 0.3, B is 0.7. The transition probabilities from each state to all the other states are also shown in the Figure. At time slot 1 we have the probability of only state 1 and it emitting R is 0.6. Now let us calculate the forward probabilities of each of the states at the time slice 2 with observation being R.

$$\alpha_2(1) = (\alpha_1(1) X\ a_{11} + \alpha_1(2)\ Xa_{21} + \alpha_1(3)\ X\ a_{31}) Xb_1(R)$$

$$=(0.6X0.5+0X0.4+0X0.1)\ X\ 0.6 = 0.18$$

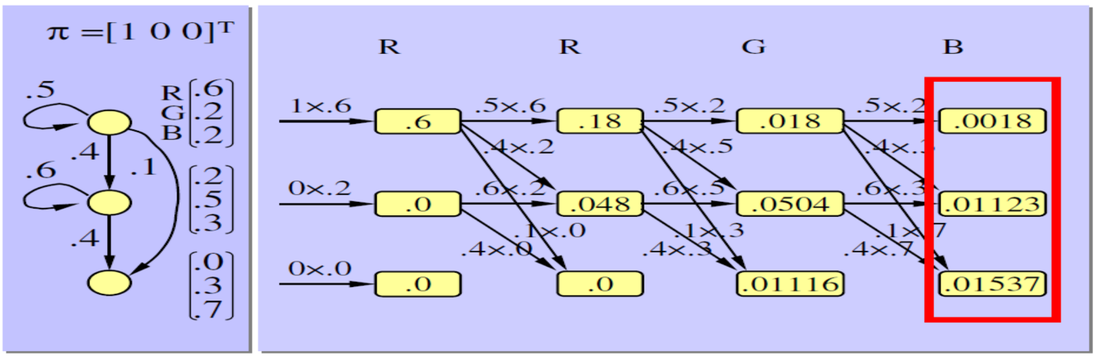
$$\alpha_2(2) = (\alpha_1(1) X\ a_{12} + \alpha_1(2)\ Xa_{22} + \alpha_1(3)\ X\ a_{32}) Xb_2(R)$$

$$= (0.6X0.4+0X0.6+0X0)\ X\ 0.2 = 0.048$$

$$\alpha_2(3) = (\alpha_1(1) X\ a_{13} + \alpha_1(2)\ Xa_{23} + \alpha_1(3)\ X\ a_{33}) Xb_3(R)$$

= )

**Numerical Example:** $P$(RRGB|λ) [신봉기 03]



Example from isoft.postech.ac.kr/Course/CS704/.../**HiddenMarkovModel**.pdf

**Figure 35.5 Example for Calculation of Forward Probabilities**

Now let us calculate the forward probabilities of each of the states at the time slice 3 with observation being G.

$\alpha_3(1) = (\alpha_2(1)X\ a_{11}+\alpha_2(2)\ Xa_{21}+\ \alpha_2(3)\ X\ a_{31})Xb_1(G)$

$= (0.18X0.5+0.048X0+0X0)\ X\ 0.2 = 0.018$

$\alpha_3(2) = (\alpha_2(1)X\ a_{12}+\alpha_2(2)\ Xa_{22}+\ \alpha_2(3)\ X\ a_{32})Xb_2(G)$

$= (0.18X0.4+0.048X0.6+0X0)\ X\ 0.5 = 0.0504$

$\alpha_3(3) = (\alpha_2(1)X\ a_{13}+\alpha_2(2)\ Xa_{23}+\ \alpha_2(3)\ X\ a_{33})Xb_3(G)$

$= (0.18X0.1+0.048X0.4+0X0.1)\ X\ 0.3 = 0.1116$

Now let us calculate the forward probabilities of each of the states at the final time slice 4 with final observation of the sequence being B.

$\alpha_4(1) = (\alpha_3(1)X\ a_{11}+\alpha_3(2)\ Xa_{21}+\ \alpha_3(3)\ X\ a_{31})Xb_1(B)$

$=(0.018X0.5+0.0504X0+0.01116X0)X0.2 = 0.0018$

$\alpha_4(2) = (\alpha_3(1)X\ a_{12}+\alpha_3(2)\ Xa_{22}+\ \alpha_3(3)\ X\ a_{32})Xb_2(B)$

$=(0.018X0.4+0.0504X0.6+0X0.01116)X0.3=$
$0.01123$

$\alpha_4(3) = (\alpha_3(1)X\ a_{13}+\alpha_3(2)\ Xa_{23}+\ \alpha_3(3)\ X\ a_{33})Xb_3(G)$

$=(0.018X0.1+0.0504X0.6+0X0.01116)X0.7=$
$0.01537$

### 35.4.4 Forward Algorithm Complexity

In the naïve approach to solving problem 1 the time taken is of the order of $2T*N^T$ computations where T is the number of time slices in the sequence and N is the number of states in the HMM. However the forward algorithm takes time of the order of $N^2T$ computations.

### 35.4.5 Backward Probabilities

Analogous to the forward probability, but just in the other direction that is in the backward direction starting from the last state and traveling to the initial state.

Now we need to determine the backward probability that given an HMM and given the state at time t is i, the partial observation $o_{t+1} \ldots o_T$ is generated.

$$\beta_t(i) = P(o_{t+1} \ldots o_T \mid q_t = s_i, \lambda)$$

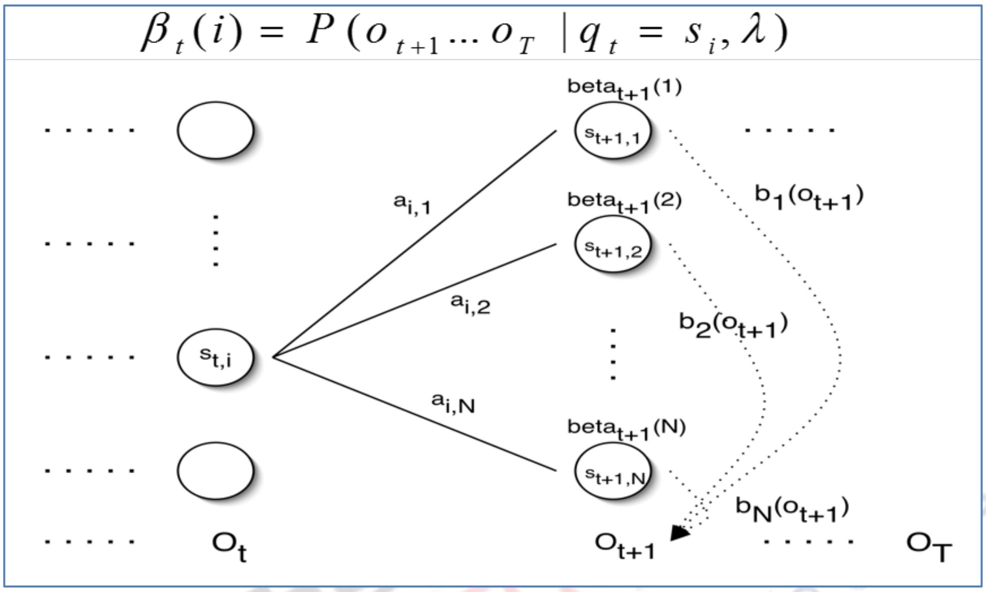


**Figure 35.6 Backward Probability**

Here we start from the final state and calculate the probability of each preceding state in the backward direction, and hence this probability is called the backward probability. The backward probability at a time slice t of a state j is the sum of each the N forward probabilities i at time slice t+1 multiplied by the transition probability of each state j at time slice t+1 to the state i under consideration t time slice t (Figure 35.6). This sum is then multiplied by the emission probability of observing $o_{t+1}$at state j.

$$\beta_t(i) = \left[ \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \right]$$

### 35.4.6 Backward recursion:

As we have already discussed backward probability

$$\beta_t(i) = \left[ \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \right]$$

is calculated using backward recursion. The initialization is as follows:

$$\beta_T(i) = 1, \quad 1 \leq i \leq N$$

Here the initial backward probability of state i at time slice T is 1. The backward recursion is determined as follows:

$$\beta_t(i) = \left[ \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j) \right] \quad t = T-1 \ldots 1, 1 \le i \le N$$

Here the backward probability at time slice t is determined by considering the backward probabilities of all states at time slot t+1, the transition probabilities from state i (the state whose backward probability is to be determined) to state j and the emission probability of the observation at time slice t+1 at state j.
Finally we have the termination happens when the backward probability of state i at time slice 1 multiplied by the initial probability of state i as follows:

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

## 35.5 Problem 2 – Decoding – Viterbi Algorithm

Given the observation sequence *O=o*$_1$*,…,o*$_T$and an HMM model  now we want to find the state sequence that best explains the observations. In other words we need to compute the most probable sequence of states, given a sequence of observations. For this decoding we describe Viterbi's dynamic programming algorithm.

As we discussed for the solution to Problem 1 (Evaluation) was the efficient determination of the sum of all paths through an HMM. For solving the decoding problem we want to find the path with the highest probability. Here given a set of symbols O determine the most likely sequence of hidden states Q that led to the observations. In other words, we want to find the state sequence Q=q$_1$…q$_T$, which maximizesP(Q|o$_1$,o$_2$,...,o$_T$) that is as follows:

$$Q = \arg\max_{Q'} P(Q' \mid O, \lambda)$$

Here we see we need to find the states that maximizes the probability of the sequence given the sequence of observations and the HMM model.

When we want to find the mostprobable state sequence we use the idea that if we know the identity of $Q_i$ , then the most probable sequence on *i+1,…,n* does not depend on observations before time *i*.

### 35.5.1 Viterbi Algorithm

The purpose of the Viterbi algorithm is to carry out an analysis of the internal processing result for finding the best most likely state sequence. It uses the dynamic programming concept to align state and observation transitions. The Viterbi algorithm is similar to computing the forward probabilities, but instead of summing over transitions from incoming states, we compute the maximum at each and every time slice. While in forward algorithm we have

$$\alpha_t(i) = P(o_1,\ldots,o_t,q_t = s_i \mid M)$$

in the case of Viterbi recursion we have

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

As we can see instead of considering the summation of forward probabilities of all the preceding states, in the case of Viterbi recursion we consider only the transition from the state where the product of the state probability and the transition is the maximum. Figure 35.7 shows the HMM.

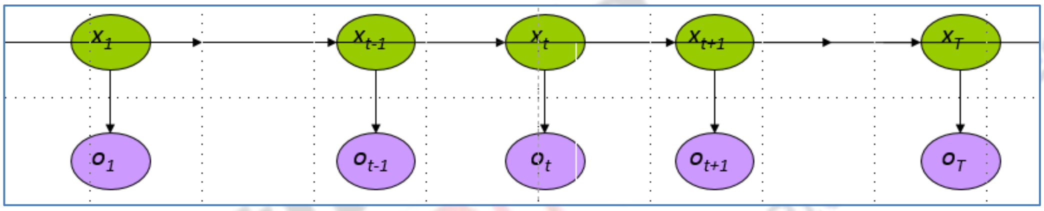


**Figure 35.7 HMM Model**

The forward probability already discussed is as given below:

$$\boxed{P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(T)}$$

Similarly backward probability has already discussed and is given below:

$$\boxed{P(O \mid \mu) = \sum_{i=1}^{N} \pi_i \beta_i(1)}$$

The forward probability and backward probability can be combined:

$$\boxed{P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(t)\beta_i(t)}$$

The initialization of the Viterbi algorithm is at time slice 1 and state i.

$$\boxed{\delta_1(i) = \pi_i b_j(o_1) \qquad 1 \le i \le N}$$

Then at each recursive step we find the maximum probability from one of the N states as shown by the induction.

$$\boxed{\delta_t(j) = \left[\max_{1\le i\le N} \delta_{t-1}(i)\, a_{ij}\right] b_j(o_t)}$$

$$\delta_j(t) = \max_{x_1 \dots x_{t-1}} P(x_1 \dots x_{t-1}, o_1 \dots o_{t-1}, x_t = j, o_t)$$

The state sequence which maximizes the probability of seeing the observations to time t-1, landing in state j, and seeing the observation at time t

Then we find the argument maximum to find the termination condition:

$$\boxed{\psi_t(j) = \left[\arg\max_{1\le i\le N} \delta_{t-1}(i)\, a_{ij}\right] \quad 2 \le t \le T, 1 \le j \le N}$$

$$\boxed{\begin{aligned} \delta_j(t+1) &= \max_i \delta_i(t)\, a_{ij}\, b_{jo_{t+1}} \\ \psi_j(t+1) &= \arg\max_i \delta_i(t)\, a_{ij}\, b_{jo_{t+1}} \end{aligned}}$$

In this way the best sequence of hidden states that gave rise to the given set of observations as given below:

$$\boxed{p^* = \max_{1\le i\le N} \delta_T(i)}$$

$$\boxed{q_T^* = \arg\max_{1\le i\le N} \delta_T(i)}$$

In this way the final sequence of states is computed by working backwards given as below:

$$\boxed{q_t^* = \psi_{t+1}(q_{t+1}^*) \qquad t = T-1, \dots, 1}$$

$$\hat{X}_T = \arg\max_i \delta_i(T)$$

$$\hat{X}_t = \psi_{\hat{X}_{t+1}}(t+1)$$

$$P(\hat{X}) = \arg\max_i \delta_i(T)$$

### 35.5.2 Example for Viterbi Algorithm

This is the same example that we discussed in Section 35.4.3. However here we need to find the maximum values at each step rather than the sum of forward probabilities coming from each step of the induction. Now let us calculate the probabilities of each of the states at the time slice 2 with observation being R.

$\delta_1(1)$ = Maximum (1X0.6, 0X0.2, 0X0 = 0.6

$\delta_2(1)$= Maximum is ($\delta_1(1)$X $a_{11}$)X$b_1$(R)

= (0.6X0.5) X 0.6 = 0.18

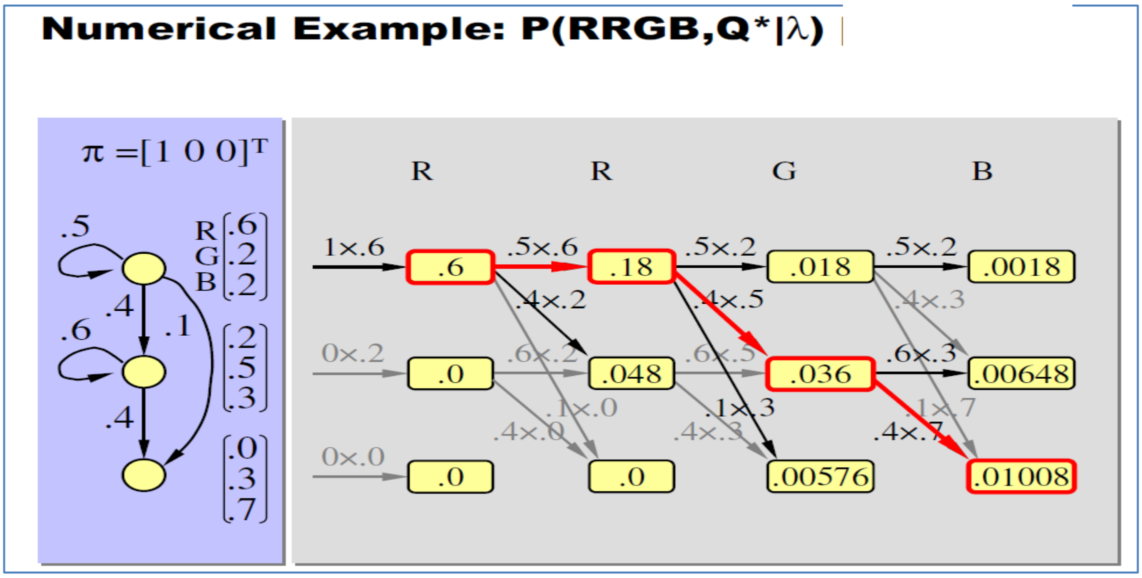
$\delta_3(2)$ = Maximum is ($\delta_2(1)$X $a_{12}$)X$b_2$(G)

= (0.18X0.4) X 0.5= 0.036

$\alpha_2(3)$ = Maximum is ($\delta_3(2)$ X$a_{23}$) X$b_3$(B)

= (0.036X0.4)X 0.7 = 0.01008

The sequence of hidden states are 1,1,2,3 to get observation sequence R,R,G,B



Numerical Example: P(RRGB,Q*|λ)

Example from isoft.postech.ac.kr/Course/CS704/.../**HiddenMarkovModel**.pdf

**Figure 35.8 Example for Viterbi Algorithm**

We will discuss the solution to the third problem associated with HMM that is the learning of the HMM model with the EM algorithm in the next module.

## Summary

- Explained the three issues of HMM
- Discussed the Baum Welsh Forward and Backward algorithms for Evaluation using the model
- Outlined Viterbi algorithm for Decoding