Natural Language Processing

Assignment 4

Type of Question: MCQ

Number of Questions: 7 Total Marks: (4×1)+(3×2)=10

===

1. Baum-Welch algorithm is an example of - **[Marks 1]**
A) Forward-backward algorithm
B) Special case of the Expectation-maximisation algorithm
C) Both A and B
D) None

Answer: C

Solution: Theory.

===

2. Once a day (e.g. at noon), the weather is observed as one of state 1: rainy state 2: cloudy state 3: sunny The state transition probabilities are :

0.4	0.3	0.3
0.2	0.6	0.2
0.1	0.1	0.8

Given that the weather on day 1 (t = 1) is sunny (state 3), what is the probability that the weather for the next 7 days will be "sun-sun-rain-rain-sun-cloudy-sun"? **[Marks 2]**

A) $1.54 * 10^{-4}$
B) $8.9 * 10^{-2}$
C) $7.1 * 10^{-7}$
D) $2.5 * 10^{-10}$

Answer: A

Solution:

O = {S3, S3, S3, S1, S1, S3, S2, S3}

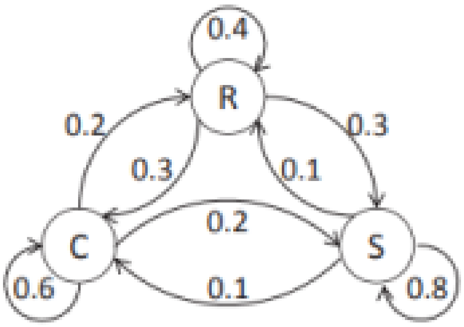
P(O | Model)

= P(S3, S3, S3, S1, S1, S3, S2, S3 | Model)

= P(S3) P(S3|S3) P(S3|S3) P(S1|S3) P(S1|S1) P(S3|S1) P(S2| S3) P(S3|S2) = Q3 · a33 · a33 · a31 · a11 · a13 · a32 · a23

= (1)(0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)

= 1.536 × 10-4



==

3. In the question 2, the expected number of consecutive days of sunny weather is:

A) 2
B) 3
C) 4
D) 5 **[Marks 1]**

Answer: D

Solution:

Exp(i) = 1/(1-p_{ii}) So for sunny the exp = 1/(1-0.8) = 5

==

4. You are building a model distribution for an infinite stream of word tokens. You know that the source of this stream has a vocabulary of size 1200. Out of these 1200 words you know of 200 words to be stop words each of which has a probability of 0.001. With only this knowledge what is the maximum possible entropy of the modelled distribution. (Use log base 10 for entropy calculation) **[Marks 2]**

A) 2.079
B) 4.5084
C) 2.984
D) 3.0775

Answer: D

Solution: There are 200 stopwords with each having an occurrence probability of 0.001. Hence,
$P(\text{Stopwords}) = 200 * 0.001 = 0.2$
$P(\text{non} - \text{stopwords}) = 1 - 0.2 = 0.8$

For maximum entropy, the remaining probability should be uniformly distributed.
For every non-stopword w, $P(w) = 0.8/(1200 - 200) = 0.8/1000 = 0.0008$. Finally, the value of the entropy would be,

$$
\begin{aligned}
H &= E(\log(1/p)) \\
&= -200(0.001 * \log(0.001)) - 1000(0.0008 \log(0.0008)) \\
&= -200(0.001 * (-3)) - 1000(0.0008 * (-3.0969)) \\
&= 0.6 + 2.4775 \\
&= 3.0775
\end{aligned}
$$

==

5. Suppose you have the input sentence “Sachin Tendulkar is a great player”. And you know the possible tags each of the words in the sentence can take.

• Sachin: NN, NNS, NNP, NNPS
• Tendulkar: NN, NNS, NNP, NNPS
• is: VB
• a: DT
• great: ADJ
• player: NN, NNS, NNP

How many possible hidden state sequences are possible for the above sentence and States? **[Marks 1]**

A) $4 \times 3 \times 3$
B) $4^{3\wedge 3}$
C) $2^4 \times 2^3 \times 2^3$
D) 3×4^2

Answer: D

Solution: Each possible hidden sequence can take only one POS tag for each of the words. Hence the total possibility will be a product of the number of candidates for each word.

==

6. What are the space and time complexity order of the Viterbi algorithm? K is the number of states and N number of time steps.

[Marks 1]

A) KN, K^2N
B) K^2N, KN
C) K^2N, K^2N
D) KN, KN

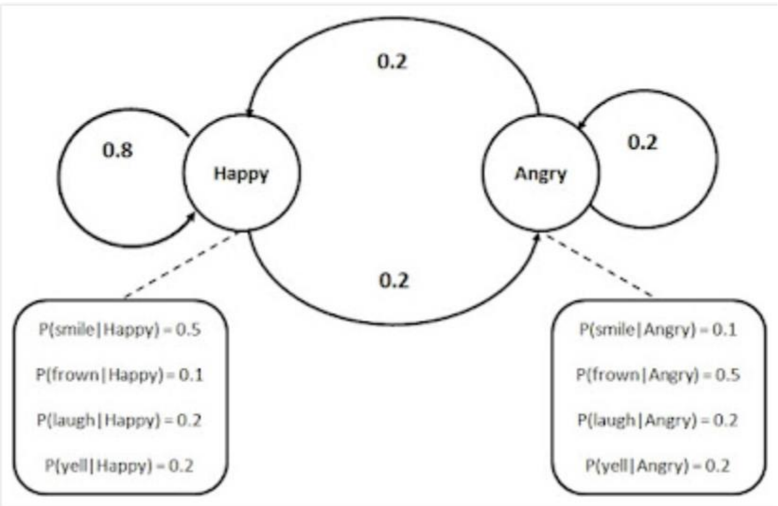
Answer: A

Solution: The sum-product algorithm is polynomial. The time complexity is $O(K^2N)$, the space complexity is O(KN), where K is the number of states and N number of time steps.

===

7. Mr. X is happy someday and angry on other days. We can only observe when he smiles, frowns, laughs, or yells but not his actual emotional state. Let us start on day 1 in a happy state. There can be only one state transition per day. It can be either a happy state or an angry state. The HMM is shown below-

Assume that q_t is the state on day t and o_t is the observation on day t. Answer the following questions;



What is $P(o_2 = frown)$? **[Marks 2]**

A) 0.56
B) 0.18
C) 0.03
D) 0.78

Answer: B

Solution: We need to find the probability of observation ***frown*** on day 2. But we don't know whether he is happy or not on day 2 (we know he was happy on day 1). Hence, the probability of the observation is the sum of products of observation probabilities and all possible hidden state transitions.

$P(o_2 = \text{frown}) = P(o_2 = \text{frown} \mid q_2 = \text{Happy}) + P(o_2 = \text{frown} \mid q_2 = \text{Angry})$

$= P(\text{Happy} \mid \text{Happy})^* \ P(\text{frown} \mid \text{Happy}) + P(\text{Angry} \mid \text{Happy})^* \ P(\text{frown} \mid \text{Angry})$

$= (0.8 * 0.1) + (0.2 * 0.5) = 0.08 + 0.1 = 0.18$