Midsem solution, NLP 2018

Q1)

I. No
II. Yes
III. No
IV. No
V. No

Q2)

P(icecream|love) =0+1/0+5=1/5
P(icecream|love) =1+1/2+5=2/7

Binary marking.Take assumptions like change in vocab mentioned,etc.

Q3)

.5 marks for each probability calculated correctly. = 0.5*6=3
1 mark for perplexity calculation = 1

P(Sachin|<s>)=0.4/19
P(is|Sachin)=0.4/19
P(a|is)=1
P(great|a)=1/3
P(guy|great)=0.4/19
P(</s>|guy)=1

Vocab size = 19

Take assumptions like change in vocab mentioned,etc.

Question 4. A- Independent assumption- features must be independent of each other.

B- P(x1x2x3---xn/C)= P(x1/C)*P(x2/C)*P(x3/C)------*P(xn/C)*P(C)

Question 5)

1) **No, can not accept epsilon empty string**
   Solution: 0*|1* **(1.5 marks)**

2) **No, can not accept string like 1001**
   Solution(1*| 1*01*01*) **(1.5 marks)**

(0.5) for every part

Question 6)

1:c , 2:a, 3:d, 4:b
.5 for each correct option

Question 7)
Note P(s)=0 . 5 % and not .5

Answer: Let $T \equiv$ "Test positive", $S \equiv$ "Sufferer", $M \equiv$ "Misclassified." Then $\mathbf{P}(T|S) = 0.95$, $\mathbf{P}(T|S') = 0.10$, $\mathbf{P}(S) = 0.005$. Hence

(a) $\mathbf{P}(T) = \mathbf{P}(T|S)\mathbf{P}(S) + \mathbf{P}(T|S')\mathbf{P}(S') = (0.95 \times 0.005) + (0.1 \times 0.995) = 0.10425.$

(b) $$\mathbf{P}(S|T) = \frac{\mathbf{P}(T|S)\mathbf{P}(S)}{\mathbf{P}(T|S)\mathbf{P}(S) + \mathbf{P}(T|S')\mathbf{P}(S')} = \frac{0.95 \times 0.005}{(0.95 \times 0.005) + (0.1 \times 0.995)} = 0.0455.$$

© (05*.005)/(1-.10425)=.000279

(d) $\mathbf{P}(M) = \mathbf{P}(T \cap S') + \mathbf{P}(T' \cap S) = \mathbf{P}(T|S')\mathbf{P}(S') + \mathbf{P}(T'|S)\mathbf{P}(S) = 0.09975.$

Q8)

Word	P(word | Sports)	P(word | Not Sports)
a	$\frac{2+1}{11+14}$	$\frac{1+1}{9+14}$
very	$\frac{1+1}{11+14}$	$\frac{0+1}{9+14}$
close	$\frac{0+1}{11+14}$	$\frac{1+1}{9+14}$
game	$\frac{2+1}{11+14}$	$\frac{0+1}{9+14}$

$$P(a|Sports) \times P(very|Sports) \times P(close|Sports) \times P(game|Sports) \times P(Sports)$$
$$= 4.61 \times 10^{-5}$$
$$= 0.0000461$$

$$\text{P(a — Not Sports)} \times P(very|Not\,Sports) \times P(close|Not\,Sports) \times P(game|Not\,Sports) \times P(Not\,Sports)$$
$$= 1.43 \times 10^{-5}$$
$$= 0.0000143$$

If token size is wrong: 1 marks is given
If vocab is wrong: 2 marks is given
If prior is not taken : 2 marks is given
Else if answer is correct: 4

Question 9)
You need to calculate class specific confusion matrix measures then combine using macro or micro averaging:

Continue	**Actual T**	**Actual F**
Predict T	**3**	**2**
Predict F	**3**	**3**

Precision: 3/5
Recall: 3/6

Not Continue	**Actual T**	**Actual F**
Predict T	**3**	**3**
Predict F	**2**	**3**

Precision: 3/6
Recall: 3/5

Micro	**Actual T**	**Actual F**
Predict T	**6**	**5**
Predict F	**5**	**6**

Precision: 6/11
Recall: 6/11

Macro precision: ½*(3 / 5+ 3 / 6)
Macro Recall: ½*(3 / 6+ 3 / 5)

A. Full marks 3 for all above being correct (any one micro or macro)
B. 1 mark if only one class specific confusion metric measure is given with bellow correct measures: **precision , recall, accuracy , F score**
C. 0.5 for partial correct

Bonus

1A. If (ii) as changed

P (H|A), P(I|H), P(D|I), P(J|E), P(G|J) = 5 [+ 1]

P(b|H), P(c|I), P(f|J) = 3 [+ 1]

Total = 8

Or if (i) as changed

P(B|A), P(C|B), P(D|C), P(F|E), P(G|F) [+ 1]

P(b|B),P(c|C), P(f|F) [+ 1]

Total = 8

1B. Any 3 correct differences (+1 marks for each correct answer)

(https://www.usebackpack.com/resources/19584/download?1536899882)

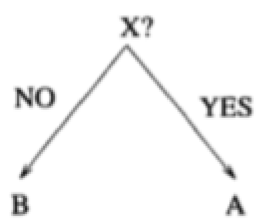
1C.

We prove here that for a fixed set of primitive queries, any binary decision tree can be converted into a transformation list. Extending the proof beyond binary trees is straightforward.

Proof (by induction)

Base Case:

Given the following primitive decision tree, where the classification is **A** if the answer to the query **X?** is yes, and the classification is **B** if the answer is no:



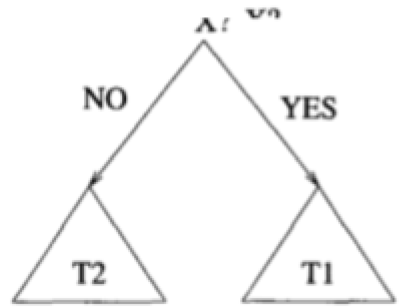
1 marks

this tree can be converted into the following transformation list:

1. Label with S /* Start State Annotation */
2. If X, then S → A
3. S → B /* Empty Tagging Environment—Always Applies To Entities Currently Labeled With S */

Induction:

Assume that two decision trees T_1 and T_2 have corresponding transformation lists L_1 and L_2. Assume that the arbitrary label names chosen in constructing L_1 are not used in L_2, and that those in L_2 are not used in L_1. Given a new decision tree T_3 constructed from T_1 and T_2 as follows:



2 marks

we construct a new transformation list L_3. Assume the first transformation in L_1 is:

Label with S′

and the first transformation in L_2 is:

Label with S″

The first three transformations in L_3 will then be:

1. Label with S
2. If X then S → S′
3. S → S″

2 marks

followed by all of the rules in L_1 other than the first rule, followed by all of the rules in L_2 other than the first rule. The resulting transformation list will first label an item as S′ if X is true, or as S″ if X is false. Next, the tranformations from L_1 will be applied if X is true, since S′ is the initial-state label for L_1. If X is false, the transformations from L_2 will be applied, because S″ is the initial-state label for L_2. □

Source : (https://www.usebackpack.com/resources/19584/download?1536899882)