National University of Computer and Emerging Sciences

School of Computing | Fall 2024 | Islamabad Campus

AI4001/CS4063
NLP DS(A,B)
AI(A)

Serial No:

Sessional II
Total Time: 1 Hour
Total Marks: 50

Signature of Invigilator

Tuesday, November 5, 2024
Course Instructor
Mirza Omer Beg

Student Name | Roll No | Section | Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. Verify at the start of the exam that you have a total of three (3) questions printed on four (4) pages including this title page.
2. Attempt all questions on the question-book and in the given order.
3. *This exam is* **open book, open notes.** Mobiles, Internet and note-sharing is not allowed. *Please see that the area in your threshold is free of any material classified as useful in the paper, i.e. mobile/internet or else there may be a charge of cheating.*
4. *Read the questions carefully for clarity of context and understanding of meaning and make assumptions wherever required, for neither the invigilator will address your queries, nor the teacher/examiner will come to the examination hall for any assistance.*
5. Fit in all your answers in the provided space. You may use extra space on the last page if required. If you do so, clearly mark question/part number on that page to avoid confusion.
6. Use only your own stationery and calculator. If you do not have your own calculator, do manual calculations.
7. Use only permanent ink-pens. Only the questions attempted with permanent ink-pens will be considered. Any part of paper done in lead pencil cannot be claimed for checking/rechecking.

	Q1	Q2	Q3	Total
Marks Obtained	05	15	15	35+10
Total Marks	20	15	15	50

Q1. Language Modeling (20 Marks) [10+10]

(a) How many trigrams can be generated from the following sentence, after performing *case normalization* and *replacing punctuation by a single space*. List them and calculate their probabilities as $P(w_i|w_{i-2}\ w_{i-1})$ in the trigram language model that uses *add-one smoothing*. (Assume that the model does NOT use start<s> and end of sentence</s> tags!)

s = #Small fish eats fish, eats fish, eats fish eats fish eats fish.

V =

After case normalization and replacing punctuation = small fish eats fish eats fish eats fish eats fish eats fish

Trigram models that can be generated:

- small fish eats
- fish eats fish
- eats fish eats
- fish eats fish
- eats fish eats
- fish eats fish

7. eats fish eats
8. fish eats fish
9. eats fish eats
10. fish eats fish

V = { small, fish, eats, }

10 trigrams can be generated.

P(eats | small fish) = $\frac{1+1}{1+3} = 2/4 = \frac{1}{2}$

P(fish | fish eats) = $\frac{5+1}{5+3} = 6/8 = 3/4$

P(eats | eats fish) = $\frac{4+1}{5+3} = 5/8$

10

(b) *Perplexity* measures how good a model is at predicting an unseen test set. Given that N is the number of words in the corpus, train two trigram models on the two corpora given below using the following formula, (and no extra tokens):

$$PP(W) = [\prod_{i=1}^{N} P(w_i|w_{i-2}\ w_{i-1})]^{-\frac{1}{N}}$$

c_1 = *oysters eats oysters eats oysters eats oysters eats small oysters*

c_2 = *oysters eats oysters eats*

V = { oysters, eats, small } 5

Calculate the perplexity for the two models on the test sentence:

s = oysters eats oysters eats oysters

c_1 = P(oysters | ~~eat~~ oysters eats) = $\frac{3+1}{4+3} = 4/7$

= P(eats | eats oysters) = $\frac{3+1}{3+3} = 4/6 = 2/3$

c_2

P(oysters | ~~eat~~ oyster eats) = $\frac{1+1}{2+3} = 2/5$

P(eats | eats oysters) = $\frac{1+1}{1+3} = 2/4 = 1/2$

Perplexity = $\sqrt[5]{\frac{1}{\frac{4}{7} \times \frac{2}{3}}}$ = $\sqrt[5]{2.625}$ = 1.21

Perplexity = $\sqrt[5]{\frac{1}{\frac{2}{5} \times \frac{1}{2}}}$ = $\sqrt[5]{5}$ = 1.37

1.21 < 1.37 less perplexity means better accuracy so c_1 is better

National University of Computer and Emerging Sciences

School of Computing Islamabad Campus

Q2. $tf - idf$ (15 Marks) [5+5+5]

Term frequency - Inverse document frequency (tf—idf), is a numerical statistic that is intended to reflect how important a word is to a document in a corpus. Assuming that $tf(t,d) = log(1 + f_{t,d})$ where $f_{t,d}$ is the raw count of a term t in a document d and $idf(t,D) = log\frac{N}{n_t}$ where N is the total number of documents in the corpus D and n_t is the number of documents containing the term t, for the subsequent questions consider the following documents:

ID	Document Text
d_1	annoyed by oysters eating oysters
d_2	happy at big fish eating small fish
d_3	what is the fish eating again?
d_4	like crabs eating oysters and fish

N = 4

Given the set of terms $\mathcal{T} = \{crabs, fish\ eating, oysters, fish\}$ answer the following:

(a) Compute the tf for the terms in $\mathcal{T}$ for each document.

	crabs	fish eating	oysters	fish
d_1	0	0	0.47	0
d_2	0	0.30	0	0.47
d_3	0	0.30	0	0.30
d_4	0.30	0	0.30	0.30

6

(b) Compute idf for the terms in $\mathcal{T}$ for the corpus.

crabs	fish eating	oysters	fish
0.60	0.30	0.30	0.12

(c) Compute $tf\text{-}idf(t,D,d)$ for the terms in $\mathcal{T}$ for each document in the corpus.

	crabs	fish eating	oysters	fish
d_1	0	0	0.14	0
d_2	0	0.09	0	0.05
d_3	0	0.09	0	0.03
d_4	0.18	0	0.09	0.03

15

National University of Computer and Emerging Sciences
School of Computing Islamabad Campus

Q3. Text Classification (15 Marks) [7+8]

You are developing an emotion detection classifier that classifies sentence *affect* as *angry*(-), *calm*(=) or *cool*(+). Consider the following training corpus for the Multinomial Naive Bayes classifier with the given labels.

Training

Sentence	Label
annoyed by his rage	-
very very annoyed at it	-
what does she do in her rage	-
he is cool	+
cool and cool is refreshing	+
calm and calm it is	=

1. Considering that your classifier disregards stopwords={*what, very, and, he, his, her, she, by, is, at, it, in, do, does*}, compute priors and likelihood probabilities for the given classes.

V = { annoyed, rage, cool, refreshing, calm }

Prior(+) = 2/6
Prior(-) = 3/6
Prior(=) = 1/6

P(annoyed|+) = (0+1)/(4+5) = 1/9
P(annoyed|-) = (2+1)/(4+5) = 3/9
P(annoyed|=) = (0+1)/(2+5) = 1/7

P(rage|+) = (0+1)/(4+5) = 1/9
P(rage|-) = (2+1)/(4+5) = 3/9
P(rage|=) = (0+1)/(2+5) = 1/7

P(cool|+) = (2+1)/(4+5) = 4/9
P(cool|-) = (0+1)/(4+5) = 1/9
P(cool|=) = (0+1)/(2+5) = 1/7

P(refreshing|+) = 2/9
P(refreshing|-) = 1/9
P(refreshing|=) = -

P(calm|+) = 1/9
P(calm|-) = 1/9
P(calm|=) = 3/7

2. Classify the following test sentence. Show your work.

Test	Label
in her rage she is calm and cool	?

AfterRemoval = rage calm cool

P = argmax{ 2/6 × 1/9 × 1/9 × 4/9 , 3/6 × 3/9 × 1/9 × 1/9 , 1/6 × 1/7 × 3/7 × 1/7 }

= argmax{ 1.82 × 10^-3 , 2.05 × 10^-3 , 1.45 × 10^-3 }

= 2.05 > 1.82 > 1.45

Classified as negative