Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

Course:	**Natural Language Processing**	**Course Code:**	**CS 535**
Program:	**MS(Computer Science)**	**Semester:**	**Spring 2020**
Duration:	**20 Minutes**	**Total Marks:**	**12**
Paper Date:	**11-Feb-20**	**Weight**	**5%**
Section:	**CS**	**Page(s):**	**2**
Exam:	**Quiz 1**		

Q1) You are given the following training corpus: [1 + 1 + 1 + 3 + 2 = 8 Marks]

<s> I like cars </s>
<s> cars like I </s>
<s> We like bikes </s>
<s> I do not like bikes and cars </s>

a) Calculate the probability of following test sentence. Include </s> in your counts just like any other token. λ_1 = trigram weight, λ_2 = bigram weight, λ_3 = unigram weight, λ_1 = 0.5, λ_2 = 0.3, λ_3 = 0.2

<s> I like bikes </s>

i. Unigram Model

ii. Bigram Model

iii. Trigram Model

iv. Trigram language model with linear interpolation.

Name: _____ Reg #: _____ Section: _____

b) Calculate perplexity of test sentence using trigram model with linear interpolation

Q2) Give some example of how language modeling helps in task in speech. [2 Marks]

Q3) A dictionary can be used to identify spelling errors. How does language modeling help in task of spelling correction? [2 Marks]

Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

	Course:	**Natural Language Processing**	**Course Code:**	**CS 535**
	Program:	**MS(Computer Science)**	**Semester:**	**Spring 2019**
	Duration:	**30 Minutes**	**Total Marks:**	**10**
	Paper Date:	**15-Feb-19**	**Weight**	**5%**
	Section:	**CS**	**Page(s):**	**2**
	Exam:	**Quiz 1**		

Q1) For the following questions, assume we are using a corpus completely summarized by the unigram counts below (thus V = 11):

Unigram counts:

red 29	retrieval 41	tree 33
apple 34	leaf 1	singing 12
orange 18	skin 4	table 9
animal 1	material 49	

N = Total words = 231

a) What are the following probabilities:

P_{MLE}(orange) = 18/231 = 0.077

P_{MLE}(skin) = 4/231 = 0.017

P_{MLE}(boot) = 0

b) Now assume we are using Laplace smoothing. What are the following probabilities?

$P_{Laplace}$(boot) = 1/242 = 0.00413

$P_{Laplace}$(skin) = 5/242 = 0.0206

Name: _____ Reg #: _____ Section: _____

Q2) You are given the following training corpus: [5 + 5 = 10 Marks]

<s> I like cars </s>
<s> cars like I </s>
<s> We like bikes </s>
<s> I do not like bikes and cars </s>

a) Calculate the probability of following test sentence. Include <s> and </s> in your counts just like any other token. λ_1 = trigram weight, λ_2 = bigram weight, λ_3 = unigram weight, λ_1 = 0.5, λ_2 = 0.3, λ_3 = 0.2

<s> I like bikes </s>

i. Unigram Model

P (<s> I like bikes </s>) = P(I) * P(like) * P(bikes) = 4/24*3/24 *4/24*2/24*4/24
= 0.000048

ii. Bigram Model

P (<s> I like bikes </s>) = P(I | <s>) * P(like | I) * P(bikes | like) * P(</s> | bikes)
= 2/4*1/3 *2/4*1/2 =0.0412

iii. Trigram Model

P (<s> I like bikes </s>) = P(I | <s> <s>)* P(like | <s> I) * P(bikes |I like) * P(</s> | like bikes)
= 2/4*1/2 *0/1*1/2 = 0

iv. **Trigram language model with linear interpolation**.

P (<s> I like bikes </s>) = $P_{interpolated}$(I | <s> <s>)* $P_{interpolated}$(like | <s> I) * $P_{interpolated}$ (bikes |I like) * $P_{interpolated}$ (</s> | like bikes)

$P_{interpolated}$(I | <s> <s>) = 0.5 * P(I | <s> <s>) + 0.3* P(I | <s>) + 0.2* P(I) = 0.5*(2/4) + 0.3*(2/4) + 0.2*(4/24)=0.43
$P_{interpolated}$(like |<s> I)= 0.5 * P(like | <s> I) + 0.3* P(like | I) + 0.2* P(like) = 0.5*(1/2) + 0.3*(1/3) + 0.2*(4/24)=0.38

Name: _____ Reg #: _____ Section: _____

$P_{interpolated}$(bikes |I like)= 0.5 * P(bikes |I like) + 0.3* P(bikes | like) + 0.2* P(bikes) = 0.5*(0/1) + 0.3*(1/4) + 0.2*(2/24)=0.092

$P_{interpolated}$(</s> | like bikes)= 0.5 * P(</s> | like bikes) + 0.3* P(</s> | bikes) + 0.2* P(</s>) = 0.5*(1/2) + 0.3*(1/2) + 0.2*(4/24)=0.43

P (<s> I like bikes </s>) = $P_{interpolated}$(I | <s> <s>)* $P_{interpolated}$(like | <s> I) * $P_{interpolated}$ (bikes |I like) * $P_{interpolated}$ (</s> | like bikes) = 0.43*0.38*0.092*0.43 = 0.0064

b) Calculate the probability of P(cars | like) using Kneser Ney smoothing from the corpus given above. d = discounting factor = 0.5

P(cars | like) = (1 – 0.5) / 4 + 3*(0.5/4)* (3/18) = 0.187

Continuation count:
<s> 0
I 2
like 4
cars 3
we 1
</s> 4
Bikes 1
Do 1
Not 1
and 1

Continuation count of all words = 18

Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

Course:	Natural Language Processing	Course Code:	CS 535
Program:	MS(Computer Science)	Semester:	Spring 2020
Duration:	20 Minutes	Total Marks:	10
Paper Date:	5-March-20	Weight	5%
Section:	CS	Page(s):	2
Exam:	Quiz 2		

Q1) Given following test collection, compute the probability of test document for positive and negative class using normal Naïve Bayes classifier with Laplace smoothing. What will be class prediction of Naïve Bayes for doc 5? (6 Marks)

Training	**Doc**	**Words**	**Class**
	1	I like this movie	**Positive**
	2	ordinary cast but great script	**Positive**
	3	interesting plot average film	**Negative**
	4	movie is interesting but long and slow paced	**Negative**
Test	**5**	great cast but average movie	**?**

Name: _____ Reg #: _____ Section: _____

Q2) Following is Tf.Idf based vector representation of words “exam” and “pass”. What is cosine similarity of the two words? [4 Marks]

Words	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6
Exam	0	2	0	2	1.2	2.6
Pass	1	2.5	0	0	3.4	0.5

Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

Course:	Natural Language Processing	Course Code:	CS 535
Program:	MS(Computer Science)	Semester:	Spring 2020
Duration:	20 Minutes	Total Marks:	12
Paper Date:	5-March-20	Weight	5%
Section:	CS	Page(s):	2
Exam:	Quiz 2		

Q1) Given following test collection, compute the probability of test document for positive and negative class using normal Naïve Bayes classifier with Laplace smoothing. What will be class prediction of Naïve Bayes for doc 5?

Training	Doc	Words	Class
	1	I like this movie	**Positive**
	2	ordinary cast but great script	**Positive**
	3	interesting plot average film	**Negative**
	4	movie is interesting but long and slow paced	**Negative**
Test	5	great cast but average movie	**?**

Solution:

Vocabulary = V = 18
P(Positive) = 2/4 = 0.5
P(Negative) = 2/4 = 0.5

P (great | Positive) = (1+1)/(9 + 18) = 2/27
P (cast | Positive) = (1+1)/(9 + 18) = 2/27
P (but | Positive) = (1+1)/(9 + 18) = 2/27
P (average | Positive) = (0+1)/(9 + 18) = 1/27
P (movie | Positive) = (1+1)/(9 + 18) = 2/27

P (great | Negative) = (0+1)/(12 + 18) = 1/30
P (cast | Negative) = (0+1)/(12 + 18) = 1/30
P (but | Negative) = (1+1)/(12 + 18) = 2/30
P (average | Negative) = (1+1)/(12 + 18) = 2/30
P (movie | Negative) = (1+1)/(12 + 18) = 2/30

P (Positive | d5) = P(d5 | Positive) * P(Positive) = 2/27*2/27*2/27*1/27*2/27*1/2 = 5.575 e-7

P (Negative | d5) = P(d5 | Negative) * P(Negative) = 1/30*1/30*2/30*2/30*2/30*1/2 = 1.646 e-7

D5 belongs to Positive class

Name: _____ Reg #: _____ Section: _____

Q2) Following is Tf.Idf based vector representation of words “exam” and “pass”. What is cosine similarity of the two words?

Words	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6
Exam	0	2	0	2	1.2	2.6
Pass	1	2.5	0	0	3.4	0.5

Solution:

Do Product (Exam, Pass) = 0*1 + 2*2.5 + 0*0 + 2*0 + 1.2*3.4 + 2.6*0.5 = 10.38
Length (Exam) = sqrt (2*2 + 2*2 + 1.2*1.2 + 2.6*2.6) = 4
Length (Pass) = sqrt(1*1 + 2.5*2.5 + 3.4*3.4 + 0.5*0.5) = 4.36

Cosine (Exam, Pass) = 10.38 / (4*4.36) = 0.595

Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

	Course:	**Natural Language Processing**	**Course Code:**	**CS 535**
	Program:	**MS(Computer Science)**	**Semester:**	**Spring 2019**
	Duration:	**30 Minutes**	**Total Marks:**	**12**
	Paper Date:	**16-April-19**	**Weight**	**5%**
	Section:	**CS**	**Page(s):**	**2**
	Exam:	**Quiz 3**		

Q1) Which of the following is the definition for "polysemy"? [2 Marks]

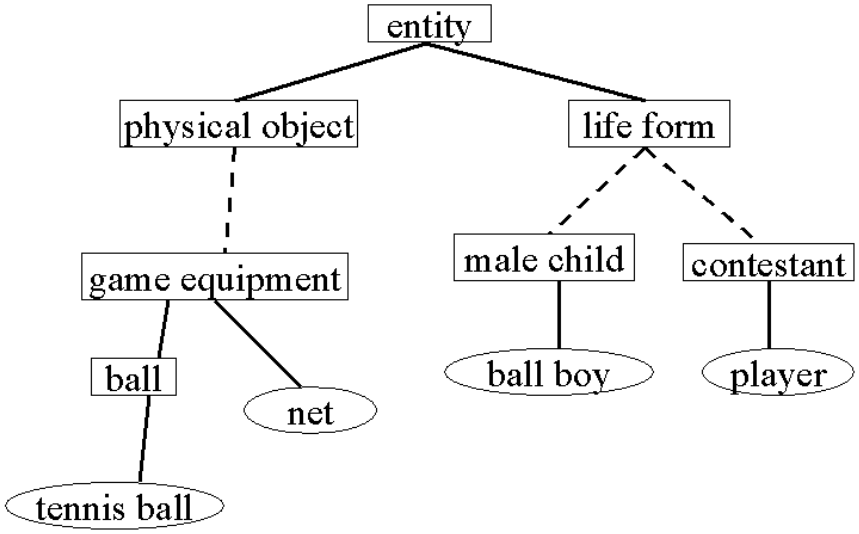
a) Two senses of a word that do not have particular relations between them, for example, the "financial institution" and "sloping mound senses" of "bank."
b) Two senses of a word that are related semantically, for example, the "financial institute" and "the building belonging to a financial institution" senses of "bank."
c) Two senses of two different words that are (nearly) identical.
d) Two senses of two different words that are opposite to each other.
e) None of the above

Q2) What is boundary error problem in evaluation of NER systems. Explain with example. **[3 Marks]**

Q3) Give examples of some features that are used for word sense disambiguation. [3 Marks]

Name: _____ Reg #: _____ Section: _____

Q4) Following is a WordNet hierarchy. The probabilities of words are given in table below: **[4 Marks]**



Word	**Probability**
entity	0.395
Physical object	0.167
Life form	0.0231
Game equipment	0.00453
Male child	0.00153
contestant	0.00743
Ball	0.000343
Net	0.00054
Ball boy	0.000113
Player	0.000445
Tennise ball	0.000189

a) Compute path based similarity between “tennis ball” and “net”

b) Compute information content based similarity proposed by Lin (Lin Similarity function) between “ball” and “player”

Name:_____ **Roll Number:**_____

Date: 09-09-2024 **Quiz -1** **Time allowed: 25 mins**

NLP-B **Total Marks: 18**

Q1. Identify the different types of morphemes (bound & free, or reduplication etc.) in the following words: (7)

1) Unhappiness
2) chit chat
3) overwhelm
4) بےخوابی
5) reconsider
6) beautiful
7) پرانا

Q2. What do the following regex return. (3)

- /geor?ge*?/
- /\$[0-9]+\.[0-9][0-9]/
- /a\.{24}z/

Q3. State the difference between stemming and lemmatization. Provide three examples where they provide different results. (3)

Q4. Assume you have the following training corpus, include </s> in your corpus like any other token. Perform all steps. (5)

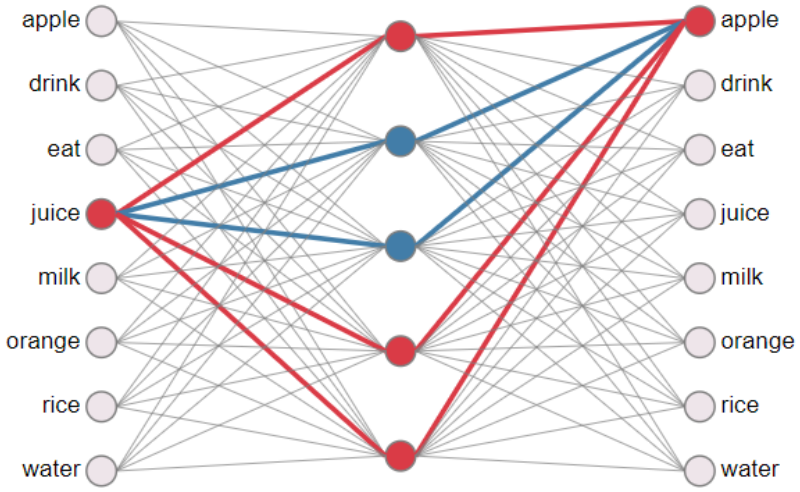
<s> i want to eat thai food </s>
<s> we ate pakistani food </s>
<s> i ate apples </s>
<s> they ate thai food </s>
Calculate the probability of the following *test sentence*. Include </s> in your counts just like any other token.

<s> i ate thai food </s>

Find the ***Unigram and Bigram*** probability of the given test sentence.
Calculate the perplexity of these both as well.

Quiz 2 NLP Roll #:_____

Question: Word to Vector Approach (Skip Gram with Negative Sampling)



We have a training corpus in which we have multiple input words and output as we your task is to define embedding of 'drink' , 'milk' , 'orange' , 'juice' given below weight matrices. Order of vocabulary is define in above architecture. We have obtained these weight matrixes after 500 training samples.

Weight Matrixes:

Input Vector

-1	1	1	-1	-1
1	1	-1	-1	-1
-0.3	0.1	-0.3	0.1	0.1
-1	1	1	-1	-1
-1	-1	1	1	-1
0.1	-1	-1	-1	-1
-1	1	1	-1	-1
-1	1	1	-1	-1

Output Vector

-1	1	1	-1	-1
-1	1	1	-1	-1
0.6	-1	-1	-1	-1
-1	1	1	-1	-1
-1	1	1	-1	-1
-1	1	1	-1	-1
-1	-1	1	1	-1
1	1	-1	-1	-1

Note:

Input Vector is the weight matrix between Input to hidden while Output Vector is the weight matrix between hidden to output.

First define a one hot encoding vector using above architecture and then process for embedding computations.

Tasks:

1. Length of the embedding _____ and how you find it out _____?
2. In case of skip grams embedding vector is just a dot product between

   _____?
3. Can we use analogies on embedding vectors? IF we yes then can we say

milk ➔ drink is same like **orange ➔ juice** ?

Justify your answer by obtaining embedding vectors from above weight matrix. Also define one hot encoding vector.

Solution:

Name: _____ Quiz 2 Roll # : _____

Question: Calculate the PMI of the words **"cat"** and **"mat".** Apply Preprocessing and write all steps. Clearly write assumptions you make. Consider sliding window size = ± 1. First create a co-occurrence matrix with respect to window size.

CORPUS:

D1: THE CAT SAT ON THE MAT.
D2: THE DOG SAT ON THE MAT.
D3: THE CAT AND DOG SAT BESIDE EACH OTHER ON THE MAT.
D4: CATS AND DOGS ARE OFTEN CONSIDERED AS PETS BY HUMANS

Name: Arooj Quiz 1(NLP) 25/25 Roll Number: 21L-5619

Q1. Write down the Vocabulary, number of tokens and number of types in the following text: 23.5+1.5 = 25

Solution: [2+2+1 marks]

Vocabulary:

{

بلی کو سمجھانے آئے، چوہے کئی ہزار
نے اک بات نہ مانی روئے
زار و زار سنو گپ شپ ناؤ میں
ندی ڈوب چلی شیر اور بکری
مل کر بیٹھے گھوڑا گھاس کھائے تینوں
اپنی ضد کے پورے کون کسے سمجھائے

Tokens: 42

Types: 38

بلی کو سمجھانے آنے
چوہے کئی ہزار
بلی نے اک بات نہ مانی
روئے زار و زار
سنو گپ شپ سنو گپ شپ
ناؤ میں ندی ڈوب چلی
شیر اور بکری مل کر بیٹھے
گھوڑا گھاس نہ کھائے
تینوں اپنی ضد کے پورے
کون کسے سمجھائے

consider as 1 token and type

Q2. Assume you have the following training corpus: [10+5 Marks]

<s> I am from Vellore </s>
<s> I am a teacher </s>
<s> Students are good and are from various cities</s>
<s> Students from Vellore do engineering</s>

i) Find the Bigram probability of the given test sentence, including <s> & </s> as a token. ii) Compute the perplexity of this bigram.

Test data:
<s> Students are from Vellore cities</s>

Note: Use Laplace (Add-1) Smoothing if needed. Write down vocabulary as well if you use it.

Vocabulary: <s>, I, am, from, Vellore, a, teacher, Students, are, good, and, various, cities, do, engineering, </s>

|V| = 16

Name: Arooj Quiz 1(NLP) Roll Number: 21L-5619

(a)

Solution: <s> Students are from Vellore cities </s>

We use Add-1 smoothing as.

$P(\text{cities}|\text{Vellore}) = 0$

Formula: $P(w_i|w_{i-1}) = \frac{c(w_{i-1}, w_i) + 1}{c(w_{i-1}) + |V|}$

$\Rightarrow P(\text{students}|\langle s\rangle) = \frac{2+1}{4+16} = \frac{3}{20}$

$\Rightarrow P(\text{are}|\text{students}) = \frac{1+1}{2+16} = \frac{2}{18}$

$\Rightarrow P(\text{from}|\text{are}) = \frac{1+1}{2+16} = \frac{2}{18}$

$\Rightarrow P(\text{Vellore}|\text{from}) = \frac{2+1}{3+16} = \frac{3}{19}$

$\Rightarrow P(\text{cities}|\text{Vellore}) = \frac{0+1}{2+16} = \frac{1}{18}$

$\Rightarrow P(\langle /s\rangle|\text{cities}) = \frac{1+1}{1+16} = \frac{2}{17}$

$P(\langle s\rangle \text{ Students are from Vellore cities } \langle /s\rangle) =$

$\left(\frac{3}{20}\right)\left(\frac{2}{18}\right)\left(\frac{2}{18}\right)\left(\frac{3}{19}\right)\left(\frac{1}{18}\right)\left(\frac{2}{17}\right)$

$= 1.91 \times 10^{-6}$

(b) perplexity $= P(\text{sentence})^{-1/N}$

$= (1.91 \times 10^{-6})^{-1/N}$

$\therefore N = 6+1$

$= 8.97$

<s> </s>

18 + 9.5

13.5 + 18

Name: Avooj Quiz 1(NLP) Roll Number: 21L-5619

Q3: Use byte pair encoding on training corpus to generate suitable vocabulary and apply it on our testing corpus and describe how well it segments our testing data. [3+2 Marks]

Training Corpus:
bat cat mat rat hat pat sat fat bat cat

Testing Set:
bat cat mat lat dat

5

Solution:

Vocabulary :- b, a, t, c, m, r, h, p, s, f, at, at_, bat_, cat_

2 bat_
2 cat_
1 mat_
1 rat_
1 hat_
1 pat_
1 sat_
1 fat_

By merging a and t

By merging at and _

By merging b and at_

By merging c and at_

Testing set: let's assume we have:

bat_, cat_, mat_, lat_, dat_

It segment our testing data as:

bat_ as bat_ , cat_ as cat_ , m at_ > mat_ ,

we don't have lat_ and dat_ in our vocabulary (even not by segmenting)

we don't have l and d in our vocabulary only at_.

Name: Fatima Usman Quiz 2 Roll #: 21L-5605

Question: Calculate the PMI of the words "cat" and "mat". Apply Preprocessing and write all steps. Clearly write assumptions you make. Consider sliding window size = ±1. First create a co-occurrence matrix with respect to window size.

CORPUS:
D1: THE CAT SAT ON THE MAT.
D2: THE DOG SAT ON THE MAT.
D3: THE CAT AND DOG SAT BESIDE EACH OTHER ON THE MAT.
D4: CATS AND DOGS ARE OFTEN CONSIDERED AS PETS BY HUMANS

PMI(cat, mat)

5 / 5

P(w|c)

Adding due to required values

	THE	SAT	AND	CAT	MAT	
CAT	2	1	2	0	0	5
MAT	3	0	0	0	0	3
	5	1	2	0	0	8

C = 33

~~P(M~~

$$PMI(cat, mat) = \log_2\left(\frac{P(cat, mat)}{P(cat) \cdot P(mat)}\right) = \frac{0/8}{(5/8)(8/8)}$$

$\log_2(0)$ = very minor value.

is very low,

PMI(cat, mat) = 0 As they ~~both are~~ target words they are not in the content of each other, (according to given corpus corpus and sliding window size.

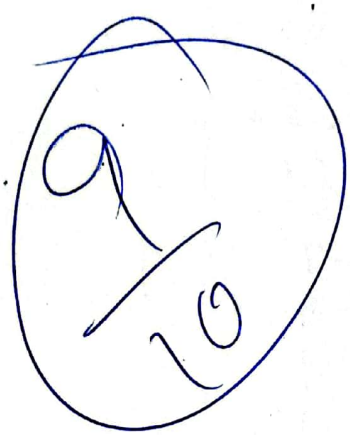
use max(formula, 0)

Name: Fiza Shahzad Quiz 3 (RNN) Roll#: 21L-7693

Q: In this task, you are required to compute the forward pass for the subsequent time step of a Recurrent Neural Network (RNN), given the provided details.

1. Draw Architecture of RNN for this simple scenario where t=0 information is given and you are asked to compute for next time stamp i.e., t=1. Also mention dimensions of each component.
2. Compute Hidden State (h_t) for the next time Stamp, Use Tanh activation function?
3. Compute Output ($y^$) for the next time Stamp, Use Sigmoid activation function?

```
Weight for Input:
 [[4]
 [1]]
Weight for Hidden State:
 [[6 6]
 [1 4]]
Weight for Output:
 [[4 3]]
Bais for Input:
 [[6]
 [4]]
Bais for Ouput:
 [[4]]
Input: [[-0.56843908]]
Previous Context:  [[ 0.2357065 ]
 [-2.06228849]]
```

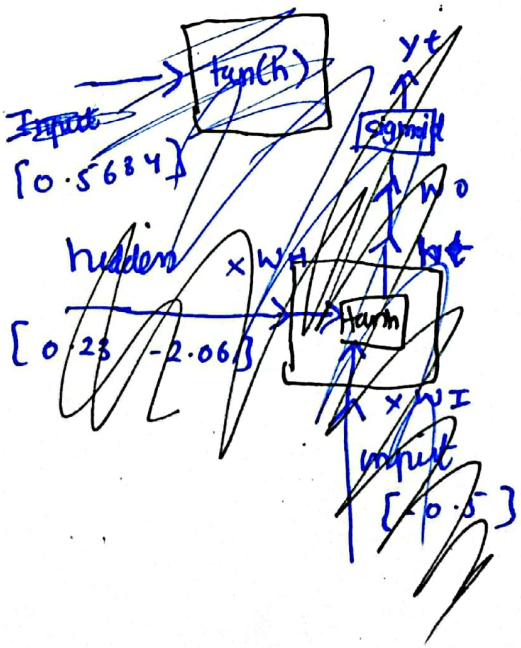


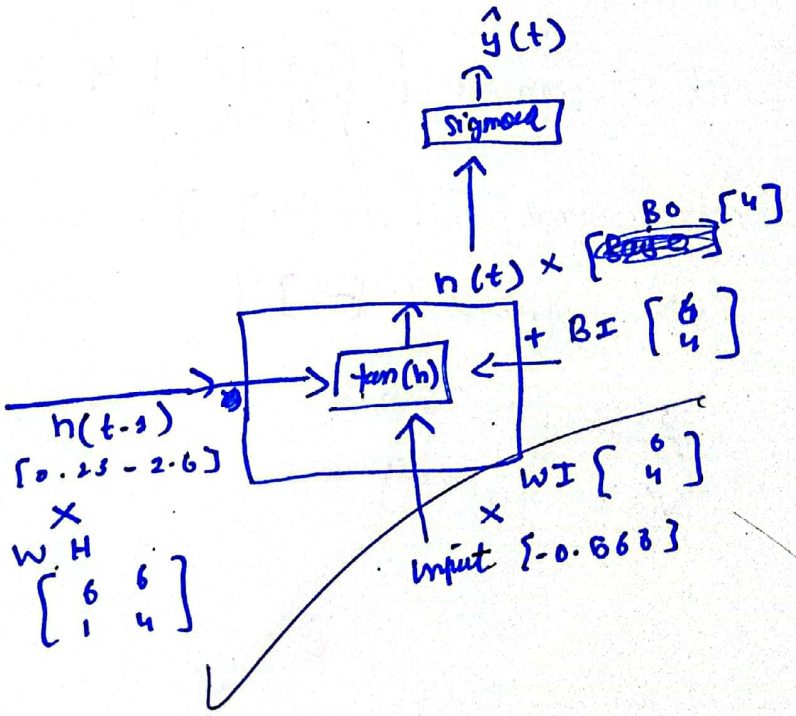
Important Formulas:

$$\tanh x = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

Sigmoid / Logistic

$$f(x) = \frac{1}{1+e^{-x}}$$





2 Hidden State :

$$a_t = [0.2357065 \quad -2.062288 \quad -0.5684] \times \begin{bmatrix} 6 & 6 & 4 \\ 1 & 4 & 1 \end{bmatrix} + \begin{bmatrix} 6 \\ 4 \end{bmatrix}$$

$$= \begin{bmatrix} -13.232 \\ -8.58 \end{bmatrix} + \begin{bmatrix} 6 \\ 4 \end{bmatrix}$$

$$= \begin{bmatrix} -7.232 \\ -4.58 \end{bmatrix}$$

$$h_t = \tanh \begin{pmatrix} -7.232 \\ -4.58 \end{pmatrix}$$

$$h_t = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

3) output state : $[-1 \quad -1]$

$$o_t = \text{sigmoid}\left(\begin{bmatrix} \\ \\ -1 \end{bmatrix}^T \begin{bmatrix} 4 & 3 \end{bmatrix} + [4] \right)$$

$$o_t = \text{sigmoid}[[-7] + [4]]$$

$$o_t = \text{sigmoid}[[-3]]$$

~~$y_t = [0.95]$~~

$$y(t) = [0.95]$$

Name: M. Ahmad Quiz Roll#: 21L-5695

Q: We've thoroughly practiced employing LSTM (Long Short-Term Memory) in our previous assignment to forecast forthcoming work tasks. The curfent objective involves computing values for the below given tasks.

1. Compute embedding from the given target weight matrix based on One Hot vector: [1 0 0 0]. 4×1
2. Define Stacked Input.
3. Compute value for forget gate from the data given below.
4. Compute C_t & h_t value from all supporting values given below.
5. Write Equations for finding C_t & h_t.

8+2 / 10

Target Weight Matrix:

4	1	3	4
2	3	3	4
4	1	1	0
2	0	2	4

Weight Matrix for Forget Gate:

6	2	4	6	6	4	4	5
5	5	1	1	0	5	6	4
2	4	2	0	1	5	5	5
6	4	2	3	1	6	3	6

4×8

Bias for Forget Gate:

0
0
2
0

Forget Gate	Input Gáte	Output Gate	h_{t-1}	C_{t-1}	$\tilde{C}$
1	1	1	0	0	4
1	1	1	0	0	2
1	1	0.99	0	0	4
1	0.99	1	0	0	2

Told by Sir

Solution: (Show Steps)

(1) Embedding = [4 2 4 2] 2

(2) Stacked Input = [0 0 0 0 4 2 4 2] 2

(3) $f = W_f \cdot I + b_i$

$$= \begin{bmatrix} 6 & 2 & 4 & 6 & 6 & 4 & 4 & 5 \\ 5 & 5 & 1 & 1 & 0 & 5 & 6 & 4 \\ 2 & 4 & 2 & 0 & 1 & 5 & 5 & 5 \\ 6 & 4 & 2 & 3 & 1 & 6 & 3 & 6 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 4 \\ 2 \\ 4 \\ 2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 50 \\ 42 \\ 44 \\ 40 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix} = 6\left(\begin{bmatrix} 50 \\ 42 \\ 46 \\ 40 \end{bmatrix} \right) \quad \text{Sigmoid}$$

$$f = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$ 2

(4) $C_t = f * C_{t-1} + i * \tilde{C}$

$$= \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0.99 \end{bmatrix} * \begin{bmatrix} 4 \\ 2 \\ 4 \\ 2 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 4 \\ 2 \\ 4 \\ 1.98 \end{bmatrix}$$

$$C_t = \begin{bmatrix} 4 \\ 2 \\ 4 \\ 1.98 \end{bmatrix}$$ ✓ (2)

$h_t = O * \tanh(C_t)$

$$= \begin{bmatrix} 1 \\ 1 \\ 0.99 \\ 1 \end{bmatrix} * \tanh\left(\begin{bmatrix} 4 \\ 2 \\ 4 \\ 1.98 \end{bmatrix}\right)$$

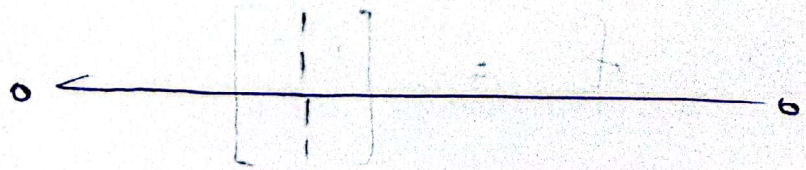
$$= \begin{bmatrix} 1 \\ 1 \\ 0.99 \\ 1 \end{bmatrix} * \begin{bmatrix} 0.99 \\ 0.96 \\ 0.99 \\ 0.96 \end{bmatrix}$$ (1+1)

$$h_t = \begin{bmatrix} 0.99 \\ 0.96 \\ 0.98 \\ 0.96 \end{bmatrix}$$

---

(5) $C_t = f * C_{t-1} + i * \tilde{C}$

$h_t = O * \tanh(C_t)$ ✓ (1+1)



Name:_____ **Roll Number:**_____

Date: 09-10-2024 **Quiz -2** **Time allowed: 20 mins**

NLP-A **Total Marks: 11**

Q1 a) How does a neural language model differ from a traditional n-gram language model? (2)

Sparsity problem and storage problem is reduced

b) In the Skip-Gram model, if a target word has 4 context words, how many different pairs of input-output examples will be generated from a single target word in the training set? and how? (2)

for a target word with 4 context words, the Skip-Gram model will generate **4 different input-output pairs**. Each pair consists of the target word as the input and one of the context words as the output.

c) What is the role of word embeddings in neural language models? (2)

- Dimensionality Reduction
- Capturing Semantic Relationships
- Handling Sparsity
- Facilitating Transfer Learning
- Improving Model Performance
- Contextual Representations

d) If two words have very similar context words across the corpus, how will their embeddings be affected in the Skip-Gram model? (2)

In the Skip-Gram model, if two words share very similar context words, their embeddings will be closely positioned in the vector space, reflecting similar meanings or usages. This leads to high cosine similarity between their embeddings, indicating frequent co-occurrence in similar contexts. Additionally, the embeddings will capture shared semantic features, enhancing their similarity.

Q2. Draw a CBOW architecture with vocabulary 10,000, embedding dimension to be 100. What will be the weight matrix for the context and target words. Mention the dimensions at each step of the network. (4)

- **Input Layer: 1x10,000 (representing the one-hot encoded vector of the current word)**
- **W1 (Input to Hidden): 10,000x100 (mapping words to their 100-dimensional embeddings)**
- **Hidden Layer (H): 1x100 (where N is the number of hidden units)**
- **W2 (Hidden to Output): 100 x 10,000**
- **Output Layer: 1 x 10,000 (predicting the probability distribution over the vocabulary)**

Name:_____ **Roll Number:**_____

1. What is the concept of static and customized embeddings in NLM?
   Static such as word to vec...which are not learned when used for any problem such as in NLM.... customized embeddings are contextualized on your corpus and backpropoagated during the training process.

2. Consider a text corpus with the following attributes: (3+2+2)
   - Total words: 100,000
   - Vocabulary size: 4,000
   - Embedding dimension for each word: 200
   - Context window: 5 words
   - Hidden layer: 50 units

   You are tasked with training a **Neural Language Model (NLM)** on this corpus.
   - Draw the architecture of this NLM, specifying the dimensionality of each and weight matrices.

   d= 200, N=50
   1xV Vxd 1xd dxN 1xN Nx V 1xV (change this acc to the context size)

   - How would the dimensions of the embedding layer differ if the vocabulary size were 10,000 instead of 4,000, and embedding dimension is changed to 100?
     - **d= 100, N=50 and V=10000**
     - 1xV Vxd 1xd dxN 1xN Nx V 1xV

   - Discuss how the *context window size* affects the input layer dimensions and the number of parameters (that is weights) in the model.

   In neural language models (NLMs), the context window size determines the number of words considered at a time to predict the next word. A larger context window allows the model to capture longer-range dependencies between words, while a smaller context window focuses on more local relationships. But it will be computationally expensive...so num of weights will increase in the input layer since more words used in this layer..and cost too.

Name:_____ **Roll Number:**_____

3. In skipgram model, how are the negative samples decided for words with a very low frequency?

Could pick w according to their unigram frequency P(w)

More common to use $p_\alpha(w)$

$$P_\alpha(w) = \frac{count(w)^\alpha}{\sum_w count(w)^\alpha}$$

α= ¾ works well because it gives rare noise words slightly higher probability

4. **Given a target word "dog" and its context words ["barks", "loud", "runs"], explain how the Skip-Gram model would update the embeddings for the word "dog" and its context words using negative sampling.**

In the Skip-Gram model with negative sampling, the target word "dog" and its context words ("barks", "loud", "runs") are updated by:

1. Computing the dot product between "dog" and each context word (positive samples).
2. Selecting random words as negative samples and computing their dot products with "dog".
3. Applying the sigmoid function to both positive and negative pairs.
4. Using gradient descent to update the embeddings, making "dog" closer to its true context words and farther from negative samples.
5. The result is that "dog" and its context words are closer in the vector space, improving the word embeddings.

5. Consider a text corpus with the following attributes: (5)

- **Total words: 500,000**
- **Vocabulary size: 5,000**
- **Embedding dimension for each word: 100**
- **Context window: 4 words**
- **Hidden layer: 50 units**

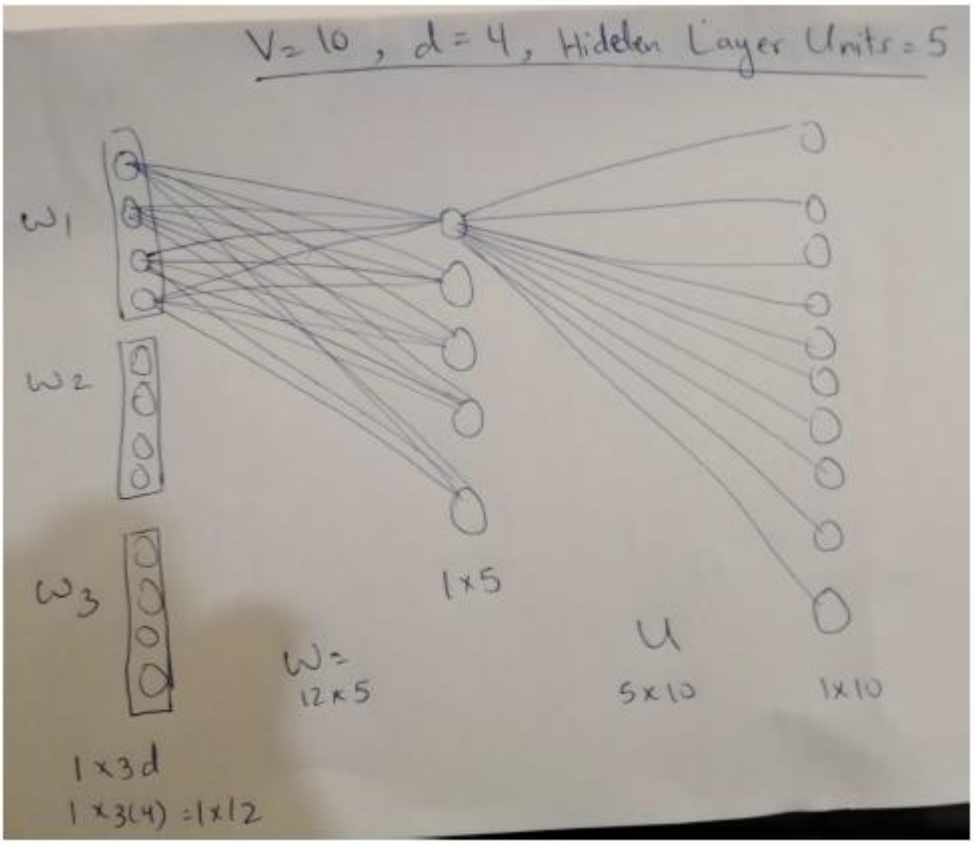
You are tasked with training a Neural Language Model (NLM) on this corpus.

- Draw the architecture of this NLM, specifying the dimensionality of each layer (input, hidden, output) and weight matrices.

Name:_____ **Roll Number:_____**

4 words will be in the input layer.... V=5000, d=100, hidden =50

1x4(d) 4(d)x50 1x50 50x5000 1x5000



"Solution"

Roll No: _____ Quiz 5 (NLP) Name: _____

Q1: If our concern is to save Training Time with infinite memory in hand then which classifier you will pick between Transformer & LSTM. Briefly explain your answer? (No more than 40 words)

Transformers as Training its Encoder & Decoder will take only Time stamp = 1 but at cost of a lot of storage while LSTM takes Time stamp = 'n' & here 'n' is the sentence length.

Q2: Write Equation and compute Masked Attention on the given (Pre-Trained) embedding of words?

Note: Consider Q, K, and V = X

① مکھی [0.5, -0.7]
شہ [-0.7, 0.5]
$X = \begin{bmatrix} 0.5 & -0.7 \\ -0.7 & 0.5 \end{bmatrix}$
Maskeel Attention

② $\Rightarrow \text{Softmax}\left(\frac{QK^T}{\sqrt{2}}\right)V$

③ $QK^T \Rightarrow \begin{bmatrix} 0.74 & -0.7 \\ -0.7 & 0.74 \end{bmatrix}$

④ $\frac{QK^T}{\sqrt{2}} = \begin{bmatrix} 0.52 & -0.49 \\ -0.49 & 0.52 \end{bmatrix}$

⑤ Masking $\Rightarrow \begin{bmatrix} 0.52 & -\infty \\ -0.49 & 0.52 \end{bmatrix}$

⑥ Applying Softmax $\begin{bmatrix} 1 & 0 \\ 0.27 & 0.73 \end{bmatrix}$

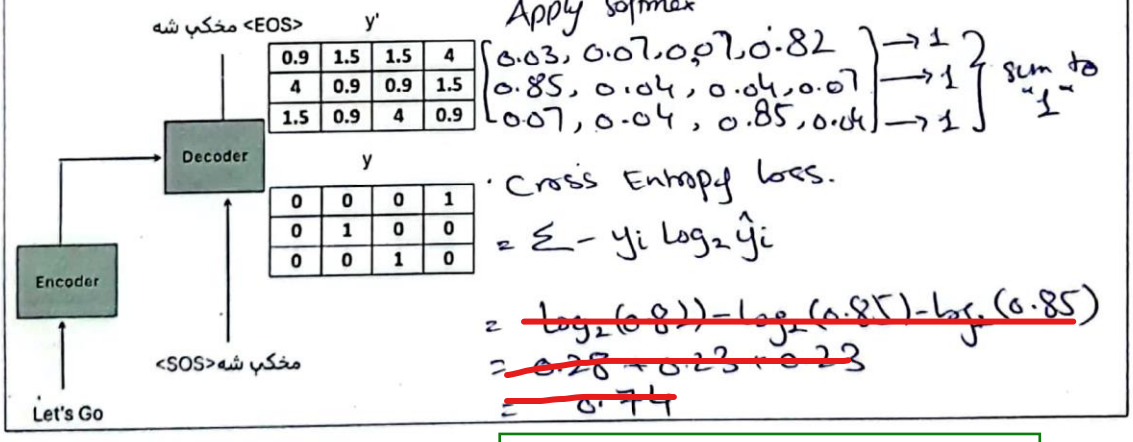
Maskeel Attention (Q, K, V)
$\Rightarrow \text{Softmax}\left(\frac{QK^T}{\sqrt{\text{Dimension}}}\right) \cdot V$

⑦ $\begin{bmatrix} 1 & 0 \\ 0.27 & 0.73 \end{bmatrix} \begin{bmatrix} 0.5 & -0.7 \\ -0.7 & 0.5 \end{bmatrix} \Rightarrow \begin{bmatrix} 0.5 & -0.7 \\ -0.37 & 0.17 \end{bmatrix}$

Q3: Compute BLEU Score for the following Machine Translation Sentence?

Example: Reference 1: The cat is on the mat.
Reference 2: There is a cat on the mat.
MT output: The cat the cat on the mat.

Q4: Compute Loss function on the following given scenario?



0.9	1.5	1.5	4
4	0.9	0.9	1.5
1.5	0.9	4	0.9

0	0	0	1
0	1	0	0
0	0	1	0

Apply "Softmax"

$\begin{bmatrix} 0.03, 0.07, 0.07, 0.82 \\ 0.85, 0.04, 0.04, 0.07 \\ 0.07, 0.04, 0.85, 0.04 \end{bmatrix} \rightarrow 1$ (sum to "1")

· Cross Entropy loss.

$= \Sigma - y_i \log_2 \hat{y}_i$

$= -\log_2(0.82) - \log_2(0.85) - \log_2(0.85)$

$= 0.28 + 0.23 + 0.23$

$= 0.74$

-(-log2(0.82) - log2(0.04) - log2(0.85))

Name: _____ Quiz 3 (RNN) Roll#: _____

Q: In this task, you are required to compute the forward pass for the subsequent time step of a Recurrent Neural Network (RNN), given the provided details.

1. Draw Architecture of RNN for this simple scenario where t=0 information is given and you are asked to compute for next time stamp i.e., t=1. Also mention dimensions of each component.
2. Compute Hidden State (h_t) for the next time Stamp, Use Tanh activation function?
3. Compute Output ($\hat{y}$) for the next time Stamp, Use Sigmoid activation function?

```
Weight for Input:
 [[4]
 [1]]
Weight for Hidden State:
 [[6 6]
 [1 4]]
Weight for Output:
 [[4 3]]
Bais for Input:
 [[6]
 [4]]
Bais for Ouput:
 [[4]]
Input: [[-0.56843908]]
Previous Context:  [[ 0.2357065 ]
 [-2.06228849]]
```

Important Formulas:

$$\tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Sigmoid / Logistic

$$f(x) = \frac{1}{1 + e^{-x}}$$

Name: _____ Reg #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

Course:	Natural Language Processing	Course Code:	CS 535
Program:	MS(Computer Science)	Semester:	Spring 2019
Duration:	30 Minutes	Total Marks:	12
Paper Date:	8 May-19	Weight	5%
Section:	CS	Page(s):	2
Exam:	Quiz 4		

Q1) Compute value of ROUGE-2 score for following summary. **[4 Marks]**

System Generated Summary: The quake had a preliminary magnitude of 6.9. in an area so isolated there are no roads connecting it to the outside world.

Reference Summary (Human Generated Summary): The quake had a preliminary magnitude of 6.9. An earthquake in the same region in February killed 2300 people and left thousands homeless.

Solution:

7/21

Q2) The first step in query focused multi document summarization is to simplify the sentences. Simplify following sentences using simple rules discussed in class. **[4 Marks]**

a) Genette's bedroom desk, the biggest disaster area in the house, is a collection of overdue library books, dirty plates, computer components, old mail, cat hair, and empty potato chip bags.
b) Robbie, a hot-tempered tennis player, charged the umpire and tried to crack the poor man's skull with a racket.
c) The car began sliding sideways, and then it hit the tree," she said
d) He died in France, as a matter of fact, and wated to be buried there.

Solution:

e) Genette's bedroom desk, is a collection of overdue library books, dirty plates, computer components, old mail, cat hair, and empty potato chip bags.
f) Robbie, charged the umpire and tried to crack the poor man's skull with a racket.
g) The car began sliding sideways, and then it hit the tree,"
h) He died in France, and wated to be buried there.

Q3) Explain Bootstrapping method for relation extraction. **[4 Marks]**

Solution:

- Gather a set of seed pairs that have relation R
- Iterate:
    1. Find sentences with these pairs
    2. Look at the context between or around the pair and generalize the context to create patterns
    3. Use the patterns for grep for more pairs

Name:_____ **Roll Number:**_____

Date: 09-09-2024 **Quiz -1** **Time allowed: 25 mins**

NLP-A **Total Marks: 18**

Q1. Identify the different types of morphemes (bound & free, or reduplication etc.) in the following words: (6)

1) sunflower

2) Discussion

3) windmill

4) dilly-dally

5) Beautify

6) womanly

Q2. Using the standard MaxMatch algorithm segment the following unsegmented sentence. Does it have any unknown words? (5)
Unsegmented sentence: thesistermisjudgedatortoiseagain
Dictionary:
the, gain, misjudged, or, at, sister, an, thesis, to, a, judged, term, is
Segmented Sentence:

Q3. How do BPE and maxmatch differ? and which algorithm is a better tokenizer than the other? Explain with an example. (2)

Q4. Assume you have the following training corpus, include </s> in your corpus like any other token. (5)

<s> I love eating pizza and ice cream </s>
<s> My favorite food is sushi and ramen </s>
<s> Eating out is fun and exciting </s>
<s> Pizza is my go to food </s>
Find the ***Unigram and Bigram*** probability of the given test sentence.
Calculate the perplexity of these both as well.

Test Sentence:
<s> I love eating sushi </s>

Name:_____ **Roll Number:**_____

Date: 09-09-2024 **Quiz -1** **Time allowed: 25 mins**

NLP-A **Total Marks: 18**

Q1. Identify the different types of morphemes (bound & free, or reduplication etc.) in the following words: (6)

1) Pharmaceuticals

2) outrageous

3) sunscreen

4) کتابدار

5) محافظت

6) postpone

Q2. Mention a few techniques to handle OOV words. Explain with reference to few examples. (5)

Q3. How do BPE and maxmatch differ? and which algorithm is a better tokenizer than the other? Explain with an example. (2)

Q4. Assume you have the following training corpus, include </s> in your corpus like any other token. (5)

<s> I love eating thai and asian </s>
<s> My favorite food is sushi and ramen </s>
<s> Eating out is fun and exciting </s>
<s> Fast food is my go to meal </s>
Find the ***Unigram and Bigram*** probability of the given test sentence.
Calculate the perplexity of these both as well.

Test Sentence:
<s> I love eating ramen </s>

Name:_____ **Roll Number:**_____

Date: 09-09-2024 **Quiz -1** **Time allowed: 25 mins**

NLP-B **Total Marks: 18**

Q1. Identify the root word and inflectional/derivational affixes in the following words. (mention each root word, and its inflectional or derivational affix, mention the one that is applicable) (9)

1. Boxes (root:_____ ; inflectional/derivational: _____)
2. terrorize
3. Unaccountability
4. Unjustifiably
5. Spoken
6. Misfortune
7. بے پناہ
8. لا حاصل
9. لاپرواہی

Q2. Suppose you have a language model trained on a corpus of English text using unigram probabilities. How would you expect the perplexity of this model to differ between two sentences: "the cat chased the mouse" and "the mouse chased the cat"? (1)

a. The perplexity on "the cat chased the mouse" is greater.
b. The perplexity on "the mouse chased the cat" is greater.
c. The perplexities are equal.

Q3. Can stemming or lemmatization help reduce the impact of OOV words? Explain by giving 3 examples. (3)

Q4. Assume you have the following training corpus, include </s> in your corpus like any other token. Perform all steps. (5)

<s> i am from Vellore </s>
<s> i am a teacher </s>
<s> students are good and are from various cities</s>
<s> students from Vellore do engineering and medical</s>

Find the unigram and Bigram probability of the given test sentence, including </s> as a token. Calculate the perplexity of these both as well.

Test data:
<s> students are from Vellore </s>

Name: _____ Quiz Roll#: _____

Q: We've thoroughly practiced employing LSTM (Long Short-Term Memory) in our previous assignment to forecast forthcoming work tasks. The current objective involves computing values for the below given tasks.

1. Compute embedding from the given target weight matrix based on One Hot vector: [1 0 0 0].
2. Define Stacked Input.
3. Compute value for forget gate from the data given below.
4. Compute C_t & h_t value from all supporting values given below.
5. Write Equations for finding C_t & h_t.

Target Weight Matrix:

4	1	3	4
2	3	3	4
4	1	1	0
2	0	2	4

Weight Matrix for Forget Gate:

6	2	4	6	6	4	4	5
5	5	1	1	0	5	6	4
2	4	2	0	1	5	5	5
6	4	2	3	1	6	3	6

Bias for Forget Gate:

0
0
2
0

Forget Gate	Input Gate	Output Gate	h_{t-1}	c_{t-1}
	1	1	0	0
	1	1	0	0
	1	0.99	0	0
	0.99	1	0	0

Solution: (Show Steps)

Name: _____ Quiz Roll#: _____

Q: We've thoroughly practiced employing LSTM (Long Short-Term Memory) in our previous assignment to forecast forthcoming work tasks. The current objective involves computing values for the below given tasks.

1. Compute embedding from the given target weight matrix based on One Hot vector: [0 1 0 0]
2. Compute value for forget gate from the data given below.
3. Compute C_t & h_t value from all supporting values given below.
4. Write Equations for finding C_t & h_t.

Target Weight Matrix:

4	1	3	4
2	3	3	4
4	1	1	0
2	0	2	4

Weight Matrix for Input Gate:

6	4	3	1	5	6	2	0
0	0	6	6	1	3	3	5
6	4	2	3	4	5	1	2
6	6	2	6	1	4	0	0

Bias for Input Gate:

4
4
2
1

Forget Gate	Input Gate	Output Gate	h_{t-1}	c_{t-1}
1		1	0.76	1
1		1	0.76	1
1		1	0.76	1
1		1	0.76	0.99

<u>Solution:</u> (Show Steps)

Name: _____ Quiz Roll#: _____

Q: We've thoroughly practiced employing LSTM (Long Short-Term Memory) in our previous assignment to forecast forthcoming work tasks. The current objective involves computing values for the below given tasks.

1. Compute embedding from the given target weight matrix based on One Hot vector: [0 0 1 0]
2. Compute value for forget gate from the data given below.
3. Compute C_t & h_t value from all supporting values given below.
4. Write Equations for finding C_t & h_t.

Target Weight Matrix:

4	1	3	4
2	3	3	4
4	1	1	0
2	0	2	4

Weight Matrix for Output Gate:

1	3	0	2	2	4	2	3
0	2	0	2	0	3	6	0
0	4	0	2	0	0	1	3
3	4	1	0	5	1	1	0

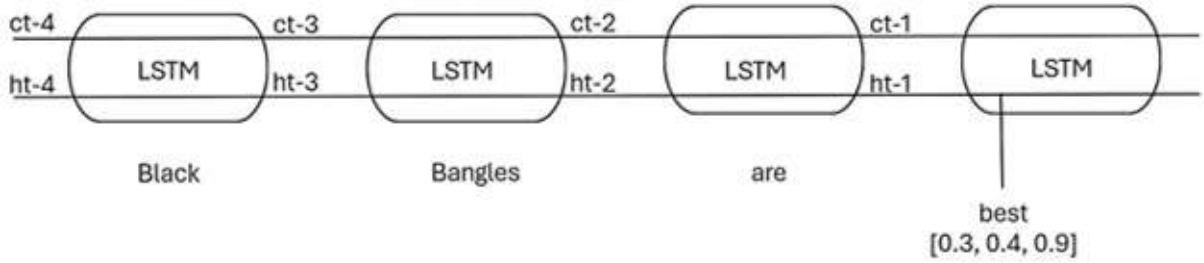
Bias for Output Gate:

5
2
3
8

Forget Gate	Input Gate	Output Gate	h_{t-1}	c_{t-1}
1	1		0.964	2
1	1		0.964	2
1	1		0.964	2
1	1		0.964	1.99

Solution: (Show Steps)

You are required to infer a LSTM for sentiment analysis task, your customer has bought a product from your ecommerce side. Given weight matrixes compute inference over trained LSTM to check the response of our customer as "Positive", "Negative" or "Neutral". [8 Marks]



As in above figure first 3 words are already processed via LSTM cells, your task is to compute value of c_t and $h_t$based on given weight matrix, h_{t-1} = 0.5, c_{t-1} = 0.75 and also to predict the sentiment of whole sentence? **Note:** Last cell is the weight value of h_{t-1}.

W^{forget}

0.50
0
0.50
0

W^{input}

0
0.75
0
0.75

$W^{candidate}$

0
0.75
0
0.75

W^{output}

1
1
0
0

$W^{softmax}$

0.8	0.1	0.1

In a RNN setup we are assuming a very basic structure where we have input and hidden context as 1x1 dimensional vector containing just one value. If we unroll our RNN 4 timestamps and every single time input is 0.9, h_0=0.75 and weight vector for input is 2 weight vector for hidden state is 1. Compute h_t for the last timestamp including all intermediate calculations. [5 Marks]

How attention gives you alignment for the target language?

In a RNN setup we are assuming a very basic structure where we have input and hidden context as 1x1 dimensional vector containing just one value. If we unroll our RNN 5 timestamps and every single time input is 0.8, h_0=0.58 and weight vector for input is 3, weight vector for hidden state is 2. Compute h_t for the last timestamp including all intermediate calculations. [5 Marks]

What was the problem in the encoder-decoder based seq-to-seq models that was catered/resolved through attention mechanisms. [3marks]

$$(h_{t-1}, x) = [0.5 \quad 0.3 \quad 0.4 \quad 0.9]$$

$$f = \sigma\left([0.5 \quad 0.3 \quad 0.4 \quad 0.9] \begin{bmatrix} 0 \\ 0.50 \\ 0 \\ 0.50 \end{bmatrix}\right)$$

$$f = \sigma(0.6)$$

$$\boxed{f = 0.645}$$

$$i = \sigma\left([0.5 \quad 0.3 \quad 0.4 \quad 0.9] \begin{bmatrix} 0.75 \\ 0 \\ 0.75 \\ 0 \end{bmatrix}\right)$$

$$= \sigma(0.675)$$

$$\boxed{i = 0.6626}$$

$$O = \sigma\left([0.5 \quad 0.3 \quad 0.4 \quad 0.9] \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}\right)$$

$$O = \sigma(0.7)$$

$$\boxed{O = 0.668}$$

$$\tilde{C}_t = \tanh\left(\begin{bmatrix} 0.5 & 0.3 & 0.4 & 0.9 \end{bmatrix} \begin{bmatrix} 0.8 \\ 0 \\ 0.75 \\ 0 \end{bmatrix}\right)$$

$$= \tanh(0.675)$$

$$\boxed{\tilde{C}_t = 0.588}$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t$$

$$C_t = 0.645 \times 0.75 + 0.6626 \times 0.588$$

$$\boxed{C_t = 0.873}$$

8

$$h_t = o_t \circ \tanh(C_t)$$

$$h_t = 0.668 \circ \tanh(0.873)$$

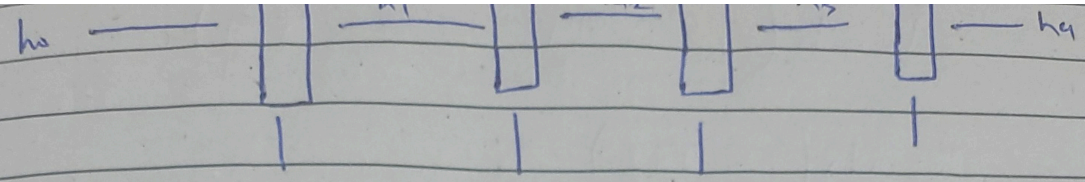
$$\boxed{h_t = 0.469}$$

$$\hat{y} = W_{\text{softmax output}} \, h_t$$

$$\hat{y} = \begin{bmatrix} 0.8 & 0.1 & 0.1 \end{bmatrix} \begin{bmatrix} 0.469 \end{bmatrix}$$

$$\boxed{\hat{y} = \begin{bmatrix} 0.375 & 0.0469 & 0.0469 \end{bmatrix}}$$

Response will be positive as it has highest value



$$h_t = \tanh(W_1 h_{t-1} + W_2 x_t)$$

$$h_1 = \tanh(W_1 h_0 + W_2 x_1)$$

$$= \tanh(1 \times 0.75 + 2 \times 0.9)$$

$$= \tanh(2.55)$$

$$h_1 = 0.987$$

$$h_2 = \tanh(W_1 h_1 + W_2 x_2)$$

$$h_2 = \tanh(1 \times 0.987 + 2 \times 0.9)$$

$$= \tanh(2.787)$$

$$h_2 = 0.992$$

$$h_3 = \tanh(W_1 h_2 + W_2 X_3)$$

$$= \tanh(1 \times 0.992 + 2 \times 0.9)$$

$$= \tanh(2.792)$$

$$h_3 = 0.992$$ ✓

6

$$h_4 = \tanh(W_1 h_3 + W_2 X_4)$$

$$= \tanh(1 \times 0.992 + 2 \times 0.9)$$

$$= \tanh(2.792)$$

$$\boxed{h_4 = 0.992}$$ at last timestamp

SecA- batch1

Q1. If a transformer layer has 8 attention heads, and each head has a weight matrix W of size 64x64, what is the total number of learnable parameters in the self-attention mechanism? How? (3)

Given that each weight matrix W^Q, W^K, and W^V has a size of 64×64, and there are 8 attention heads, we need to calculate the total number of parameters as follows:

Each attention head has:

- W^Q matrix: 64×64
- W^K matrix: 64×64
- W^V matrix: 64×64

So, for each attention head, there are $3 \times (64 \times 64)$ parameters.

Since there are 8 attention heads, the total number of parameters for the self-attention mechanism is:
$8 \times 3 \times (64 \times 64) = 12,288$

Q2. If a transformer model has 6 encoder layers and 6 decoder layers, with 8 attention heads each, how many total attention mechanisms are used in the entire model? How? (3)

2. For the decoder:

Each decoder layer has self-attention and encoder-decoder attention.

- Self-attention: 6 decoder layers * 8 attention heads per layer = 48 attention mechanisms
- Encoder-decoder attention: 6 decoder layers * 8 attention heads per layer = 48 attention mechanisms

Total attention mechanisms in the decoder = Self-attention + Encoder-decoder attention = 48 + 48 = 96 attention mechanisms

SecA- batch 2

4) Suppose, you have the following sequence: "This is my first embedding computation". You have to compute multihead self-attention on this sequence, that has an embedding of 4 dimensions, and divide into heads of 2. (10)

Assume the token embeddings to be the following:

- This: 0.31,0.22, 0.99, 0.04
- is: 0.21, 0.42, 0.09, 0.06
- my: 0.72, 0.41, 0.30, 0.39
- first: 0.51, 0.31, 0.87, 0.78
- embedding: 0.62, 0.73,0.11, 0.12
- computation:0.72, 0.11, 0.14, 0.15

Assume the weights to be

[0.8 0.3

0.9 0.6]

Compute the multihead attention for this sequence, given the above weight matrix.

SecB batch 1- **6 marks**

Given the following embeddings for three tokens ($d_{\text{model}} = 3$):

$$\text{Token 1: } [1, 0, 1], \quad \text{Token 2: } [0, 1, 0], \quad \text{Token 3: } [1, 1, 0]$$

and the following weight matrices for query, key, and value (W_Q, W_K, W_V):

$$W_Q = \begin{bmatrix} 0.2 & 0.1 & 0.4 \\ 0.3 & 0.5 & 0.1 \\ 0.7 & 0.2 & 0.6 \end{bmatrix}, \quad W_K = \begin{bmatrix} 0.3 & 0.8 & 0.2 \\ 0.5 & 0.1 & 0.4 \\ 0.6 & 0.7 & 0.3 \end{bmatrix}, \quad W_V = \begin{bmatrix} 0.1 & 0.4 & 0.7 \\ 0.3 & 0.6 & 0.2 \\ 0.5 & 0.8 & 0.9 \end{bmatrix}$$

Compute the self-attention scores for Token 1 as the query. Normalize the scores and calculate the weighted value output.

1. Calculate Query, Key, and Value for each token:

Since *d_model* = 3, the embeddings and weight matrices have compatible dimensions for multiplication.

- **Token 1:**
  - Query (Q1): [1, 0, 1] * W_Q = [1, 0, 1] * [[0.2, 0.1, 0.4], [0.3, 0.5, 0.1], [0.7, 0.2, 0.6]] = [0.9, 0.3, 1]
  - Key (K1): [1, 0, 1] * W_K = [1, 0, 1] * [[0.3, 0.8, 0.2], [0.5, 0.1, 0.4], [0.6, 0.7, 0.3]] = [0.9, 1.5, 0.5]
  - Value (V1): [1, 0, 1] * W_V = [1, 0, 1] * [[0.1, 0.4, 0.7], [0.3, 0.6, 0.2], [0.5, 0.8, 0.9]] = [0.6, 1.2, 1.6]
- **Token 2:**
  - Query (Q2): [0, 1, 0] * W_Q = [0.3, 0.5, 0.1]
  - Key (K2): [0, 1, 0] * W_K = [0.5, 0.1, 0.4]
  - Value (V2): [0, 1, 0] * W_V = [0.3, 0.6, 0.2]
- **Token 3:**
  - Query (Q3): [1, 1, 0] * W_Q = [0.5, 0.6, 0.5]
  - Key (K3): [1, 1, 0] * W_K = [0.8, 0.9, 0.6]
  - Value (V3): [1, 1, 0] * W_V = [0.4, 1, 0.9]

2. Compute Self-Attention Scores for Token 1:

We will calculate the attention scores for Token 1 by taking the dot product of its query (Q1) with the keys of all tokens (K1, K2, K3), and then dividing by the square root of *d_model* (which is √3 ≈ 1.732).

- Score(Token 1, Token 1) = Q1 • K1 / √3 = (0.9 * 0.9 + 0.3 * 1.5 + 1 * 0.5) / 1.732 = (0.81 + 0.45 + 0.5) / 1.732 ≈ 1.016
- Score(Token 1, Token 2) = Q1 • K2 / √3 = (0.9 * 0.5 + 0.3 * 0.1 + 1 * 0.4) / 1.732 = (0.45 + 0.03 + 0.4) / 1.732 ≈ 0.508
- Score(Token 1, Token 3) = Q1 • K3 / √3 = (0.9 * 0.8 + 0.3 * 0.9 + 1 * 0.6) / 1.732 = (0.72 + 0.27 + 0.6) / 1.732 ≈ 0.918

3. Normalize the Scores (Softmax):

- Softmax(1.016, 0.508, 0.918) ≈ (0.38, 0.19, 0.34) (These values don't add up to exactly 1 due to rounding, but they should be very close).

4. Calculate the Weighted Value Output:

The weighted value output for Token 1 is the weighted sum of the value vectors, using the normalized attention scores as weights:

- Output(Token 1) = 0.38 * V1 + 0.19 * V2 + 0.34 * V3
  ≈ 0.38 * [0.6, 1.2, 1.6] + 0.19 * [0.3, 0.6, 0.2] + 0.34 * [0.4, 1, 0.9]
  ≈ [0.228, 0.456, 0.608] + [0.057, 0.114, 0.038] + [0.136, 0.34, 0.306]
  ≈ [0.421, 0.91, 0.952]

SecB – Batch 2

Q. A transformer model uses positional encodings to incorporate positional information into the word embeddings. The model has an embedding dimension *d_model* = 4. Calculate the positional encodings for the first two positions (pos = 0 and pos = 1) using the standard sine and cosine formulas.

Show your calculations for each dimension of the positional encodings. What is the purpose of using both sine and cosine functions, and how does the 2i/d_model term contribute to the effectiveness of positional encoding?

(3 + 3)

Here's how to calculate the positional encodings:

Position 0 (pos = 0):

- **i = 0:**
    - PE(0, 0) = sin(0 / 10000^(0)) = sin(0) = 0
    - PE(0, 1) = cos(0 / 10000^(0)) = cos(0) = 1
- **i = 1:**
    - PE(0, 2) = sin(0 / 10000^(2/4)) = sin(0) = 0
    - PE(0, 3) = cos(0 / 10000^(2/4)) = cos(0) = 1

Therefore, the positional encoding for position 0 is [0, 1, 0, 1].

Position 1 (pos = 1):

- **i = 0:**
    - PE(1, 0) = sin(1 / 10000^(0)) = sin(1) ≈ 0.8415
    - PE(1, 1) = cos(1 / 10000^(0)) = cos(1) ≈ 0.5403
- **i = 1:**
    - PE(1, 2) = sin(1 / 10000^(2/4)) = sin(1 / 100) ≈ 0.01
    - PE(1, 3) = cos(1 / 10000^(2/4)) = cos(1 / 100) ≈ 0.99995

Therefore, the positional encoding for position 1 is approximately [0.8415, 0.5403, 0.01, 0.99995].

Purpose of Sine and Cosine:

Using both sine and cosine functions allows the model to represent relative positional relationships. Because sine and cosine are related through a phase shift, the model can easily learn to attend to relative positions by learning

simple linear transformations. This allows the model to generalize to longer sequences than those seen during training.

Contribution of 2i/d_model:

The 2i/d_model term creates different frequencies for different dimensions of the positional encoding. This allows the model to capture both broad and fine-grained positional information. Lower dimensions (smaller i) capture broader relationships (lower frequencies), while higher dimensions (larger i) capture finer-grained relationships (higher frequencies). This range of frequencies provides a richer representation of position and allows the model to learn more complex positional dependencies.