Name suliman khan
Id 14282
Subject Nlp

Question No. 2: (5)

Specify the text strings using the below regular expressions:

a. /a(bc)
   - Given string: ab abc ac acb a0b a2b a42c A87d

b. /[abc]
   - Given string: ab abc ac acb a0b a2b a42c A87d

c. /abc+
   - Given string: ab abc abcc babc

d. /abc*
   - Given string: ab abc abcc babc

e. /[^a-z A-Z 0-9]
   - Given string: a89 opx cfff $1!

Ans:

a. /a(bc)
   - Given string: ab abc ac acb a0b a2b a42c A87d

Ans ab **abc** ac acb a0b a2b a42c A87d

b./[abc]
   - Given string: ab abc ac acb a0b a2b a42c A87d

Ans ab abc ac acb a0b a2b a42c A87d

c. /abc+
   - Given string: ab abc abcc babc

ab abc abcc babc

d. /abc*
   - Given string: ab abc abcc babc

Ans ab abc abcc babc

e./[^a-z A-Z 0-9]

- Given string: a89 opx cfff $1!

Ans a89 opx cfff $1!

Q no 4:

a) Explain Part of Speech Tagging (POS) and explain POS tag ambiguity with two examples

Ans :

Pos tagging *it is a process of converting a sentence to form –list of words list of tuples where each tuple is having a form word tag.the tag in case of is a part of speech tag and signifies whether the word is a noun adjective verb and so on*

Default tag is a basic step for the part of speech tagging it is performed the is using the default tagger class.the default tagger class takes tag as a single argument.nn is the tag for a singular noun .defualt tagger is most useful when it gets to work with most common part of specch tag that's why a noun tag is recommended.

POS Tagging is a process that attaches each word in a sentence with a suitable tag from a given set of tags. The set of tags is called the Tag-set. Standard Tag-set : Penn Treebank (for English)

Pos tag ambiguity:

Common parts of speech in English are noun,veerb adjective adverb etc the pos tagging problem is to determine the pos tag for a particular instance of a word.the main problem with pos tagging is ambiguity.

For example

Pos tag ambiguity

In English 1 bank on the bank on the river bank for my transactions.

Bank1 is verb the other two banks are noun

b)State difference between open vs. closed classes in POS tagging

Ans:

POS Tagging • Words often have more than one POS: back • The back door = JJ (adjective)

• On my back = NN (noun) • Win the voters back = RB (adverb)

• Promised to back the bill = VB (verb)

• The POS tagging problem is to determine the POS tag for a particular instance of a word

open classes (like nouns, verbs **and** adjectives) acquire new members constantly,

closed classes (such as pronouns **and** conjunctions) acquire new members infrequently, if at all

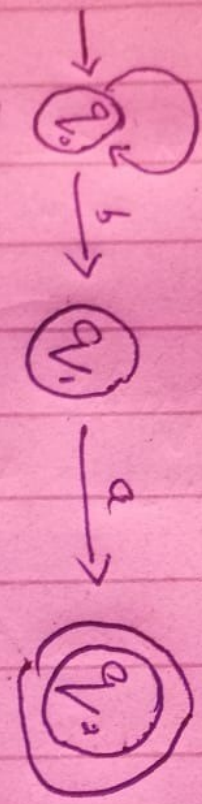
Q no 3 part (b)

Design an NFA for the regular expression : $a^* b(a+b)^*$

Ans

Qno 3 part (B)

NFA:



State	a	b
q_0	q_0	q_2, q_1
q_1	q_2	—
q_2	—	—

a) Design an NFA over an alphabet ∑= {a, b} such that every string accepted must end with a string --ba. Identify its tuples and also convert it into DFA

Ans

Here as we can see that each string of the above language ends with ‘ab’ but the below language is not accepted by this NFA because some of the string of below language does not end with ‘ab

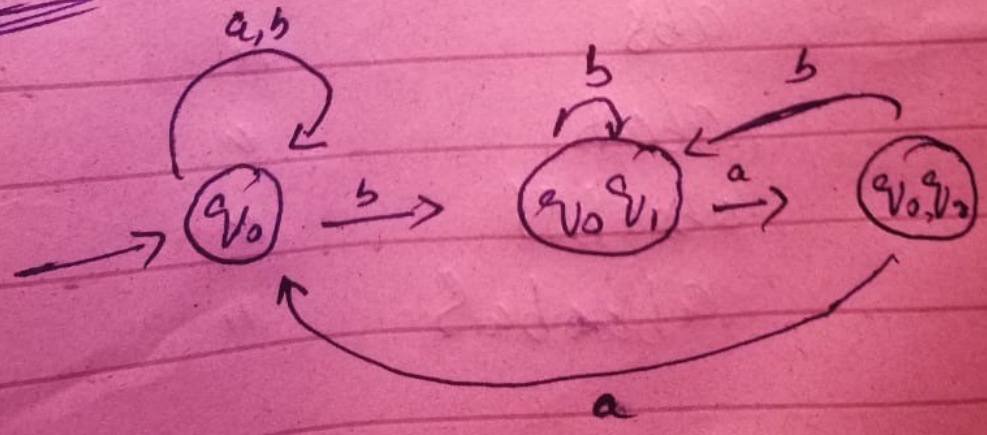
In the above NFA, the initial state ‘X’ on getting ‘a’ as the input it either remains in the state of itself or transit to a state ‘Y’ and on getting ‘b’ as the input it remains in the state of itself. The state ‘Y’ on getting ‘b’ as the input it transmits to a final state ‘Z’.

Problem-2:
Construction of a minimal NFA accepting a set of strings over {a, b} in which each string of the language is not ending with ‘ab’.

Qno 3 part (a) :

State	a	b
→ q_0	$\{q_0\}$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_2\}$	$\{q_0, q_1\}$
$\{q_0, q_2\}$	$\{q_0\}$	$\{q_0, q_1\}$

DFA



: **What is the minimum number of states in deterministic finite automata** (**DFA) for string starting with ba^2 and ending with 'a' over alphabet** {a, **b}** Explanation: In the **above DFA**, **minimum number of states** required is six.

Question No. 1: (5)

Explain objectives of NLP? Write the name of 2 Applications of NLP with example? Write the name of 2 Challenges of NLP with example

Ans :

Natural Language Processing*, usually shortened as* ***NLP****, is a branch of artificial intelligence that deals with the interaction between computers and humans using the natural language. The ultimate* ***objective of NLP*** *is to read, decipher, understand, and make sense of the human languages in a manner that is valuable*

The goal of natural language processing is to specify a language comprehension and production theory to such a level of detail that a person is able to write a computer program which can understand and produce natural language. The basic goal of NLP is to accomplish human like language processing. The choice of word "processing" is very deliberate and should not be replaced with "understanding". For although the field of NLP was originally referred to as Natural Language Understanding (NLU), that goal has not yet been accomplished. A full NLU system would be able

2 Applications of NLP with example

Machine Translation. Everyone knows what is a manual translation — we translate information from one language into another.

1. Market Intelligence.
2. *Text categorization & summarization*

2 Challenges of NLP with example

Below are the steps involved and some challenges that are faced in the machine learning process for NLP:

- *Breaking the sentence. ...*
- *Tagging the parts of speech (POS) and generating dependency graphs*.

Question No. 5:

a) Apply Bayesian theorem over the below given string:

^John got many NLP books. ^He found them all very interesting.

Where for lexical probabilities assume John=0.5, got=0.3, many=0.2, NLP=0.1 and books=0

Bayes' theorem, named after 18th-century British mathematician Thomas Bayes, is a mathematical formula for determining conditional probability. The theorem provides a way to revise existing predictions or theories (update probabilities) given new or additional evidence

Ans:

P(T)=PWIT)=tt(ti/ti-1)-P(wi)tiP1(ti/ti-1)=P(Wi/ti)
Le corpus :^ John got many NLP books found all very interesting
POS tagged
^N V N N.^N V N A R A
Recording numbers

	^	N	V	A	R	.
^	0	2	0	0	0	0
N	0	1	2	1	0	1
V	0	1	0	1	0	0
A	0	1	0	0	1	1
R	0	0	0	1	0	0
.	1	0	0	0	0	0

Bigram Probabilties **P(N/V)=# (^=N)/#^**

	^	N	V	A	R	.
^	0	1	0	0	0	0
N	0	1/5	2/5	1/5	0	1/5
V	0	1/2	0	1/5	0	0
A	0	1/3	0	0	1/3	1/3
R	0	0	0	1	0	0
.	1	0	0	0	0	0

Lexical probability

	John	Got	Many	NLP	books
^	0.5	0.3	0.2	0.1	0
N	0.5	0.3	0.2	-	-
V	0.5	0.3	0.2	-	-
A	0.5	-	-	-	-
R	0.5	-	-	-	-
.					

P(John/^) = P(Wi=John/ti=^)
*=# (John,^)/ #^*
*# (m y,^)/#^* *= 0.5x2/2 = 0.5*

Q no 5

b) Find the CFG of the string “abaabaa” using the production rules

S→a, S→aAS, A→bS

Qno 5:

(b) Find the CFG of the string "abaabaa" using the production rules.

$S \rightarrow a, S \rightarrow aAS, A \rightarrow bS$

Ans:

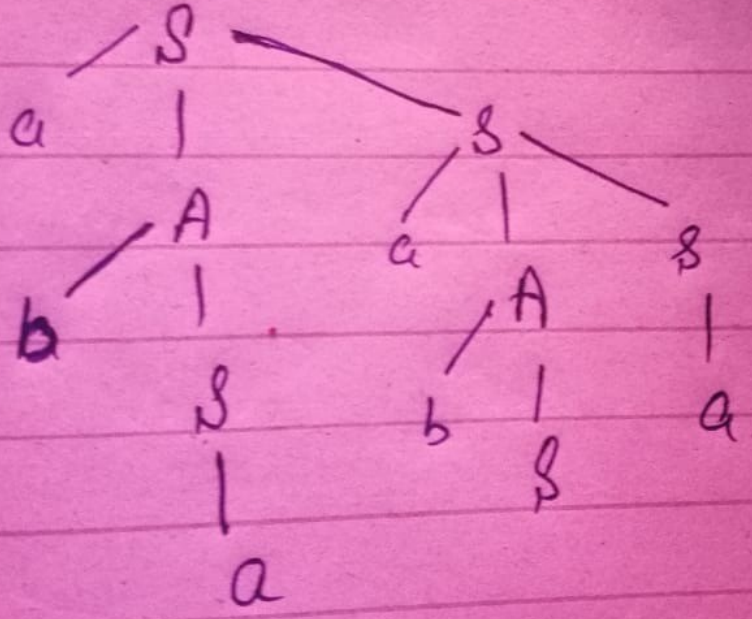
Compute the string W = 'abaabaa' with left most derivation.

$S \Rightarrow aAS$ (Rule 2)

abS	"	3)
abaAS	"	1
abaabSS	"	2
abaabaS	"	3

abaabaa

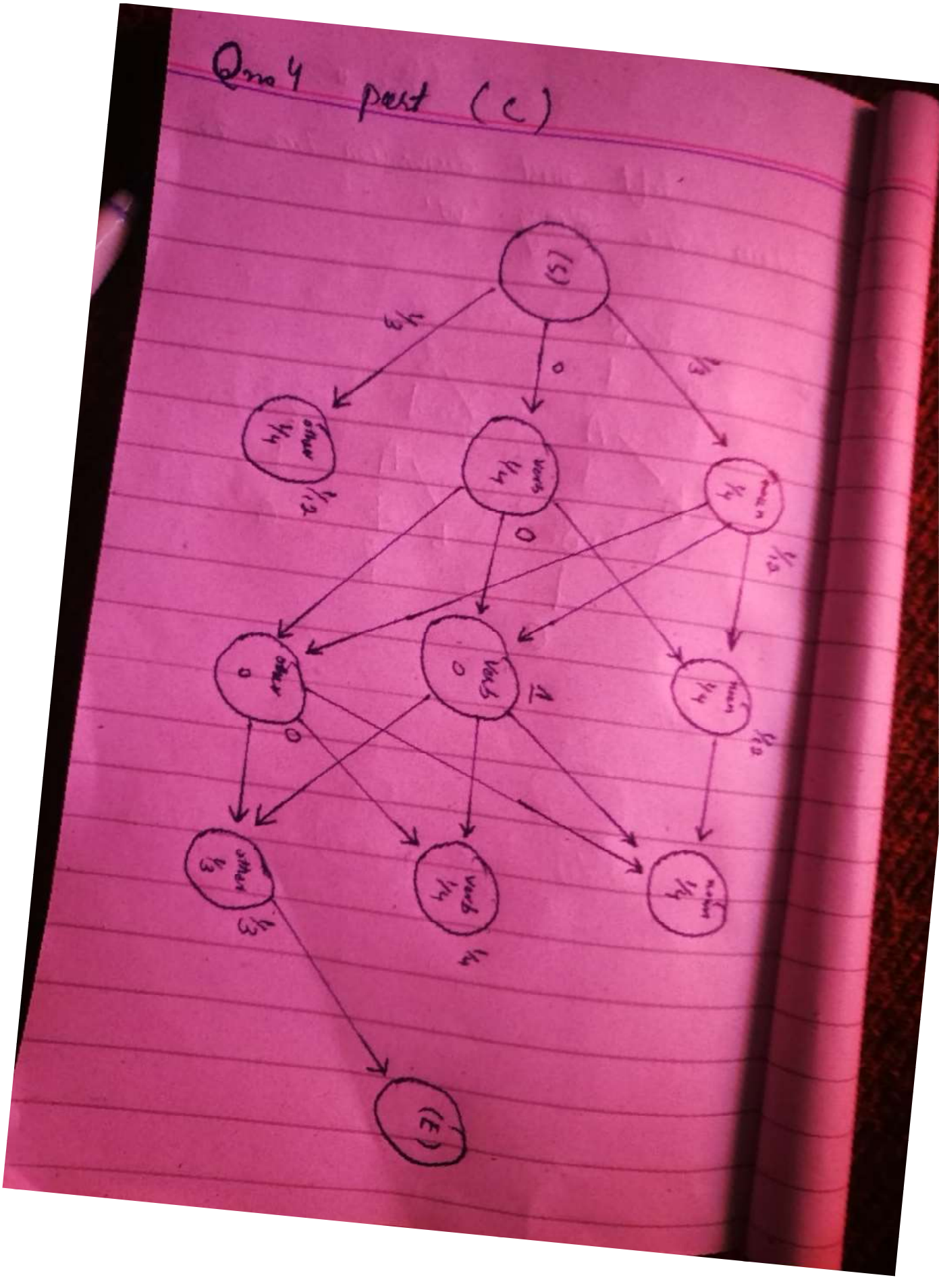
left most derivation Tree to obtain the strong 'w'



Q no #4

b) Apply Viterbi Algorithm on the below given bigram and lexical probabilities

Qno 4 part (c)


(S)
4/4
0
1/3
other
1/4
1/12
words
1/4
0
noun
1/2
1/12
other
0
0
verbs
0
1/7
noun
1/4
1/10
other
1/3
1/3
verb
1/4
1/4
noun
1/4
(E)