USN | | | | | | | | | | |

Internal Assessment Test 1 – October 2023

Sub:	Natural Language Processing					Sub Code:	18CS743	Branch:	ISE	
Date:	31/10/2023	Duration:	90 min's	Max Marks:	50	Sem/Sec:	VII A, B & C		OBE	

Answer any FIVE FULL Questions | MARKS | CO | RBT

1. Define NLP. Outline challenges of NLP [10] CO1 L2

Applications:

The application utilizing NLP includes the following.

Machine Translation
- Automatic translation of text from one human language to another.
- It requires to understand the words and phrases, grammars, semantics and world knowledge.

Speech Recognition
- Mapping acoustic speech signals to a set of words.
- Difficulties arises due to
- Variations in the pronunciation of words, homonym
- Acoustic ambiguities

Speech Synthesis
- Automatic production of speech(utterance of sentences)
- Can read your mails on telephone, or story book for you.
- To generate utterances, text has to be processed.
- NLP remains an important component of any speech synthesis system.
- Natural language Interfaces to Databases
- Allow querying a structured database using natural language sentences.

Information Retrieval
- Identifying documents relevant to a user's query.
- Indexing, word sense disambiguation, query modification have also been used in IR system to enhance performance.
- WordNet, LDOCE and Roget's Thesaurus are some of the useful resources for IR research.

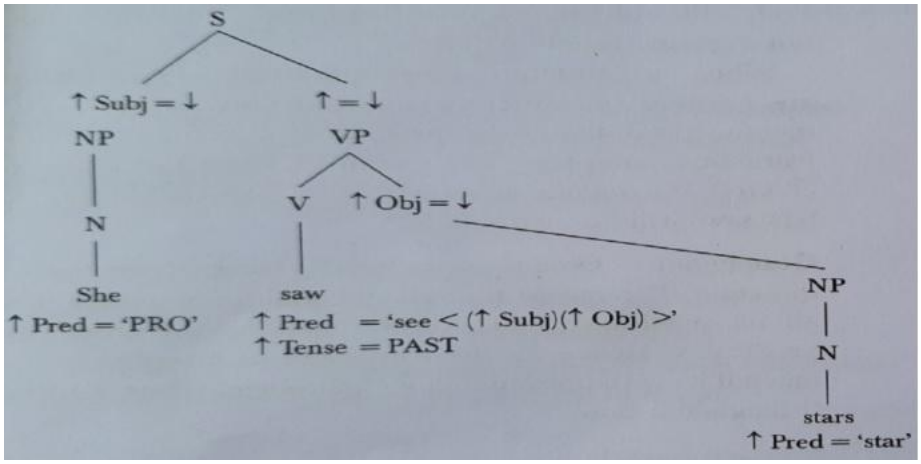
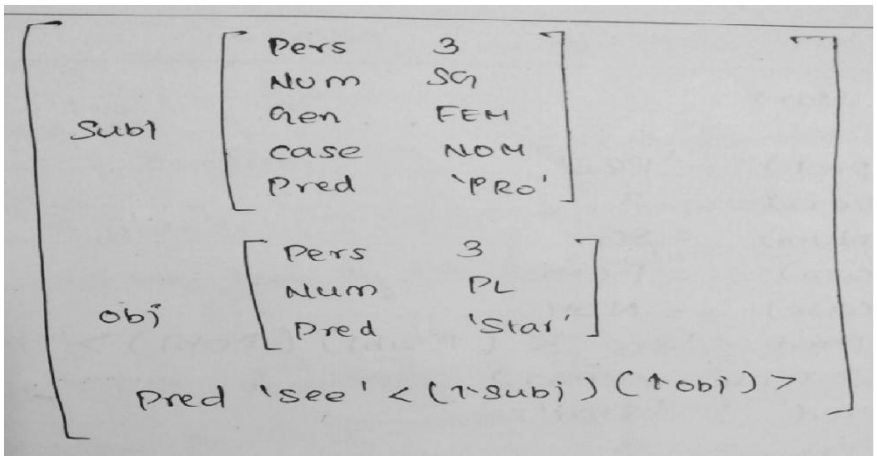
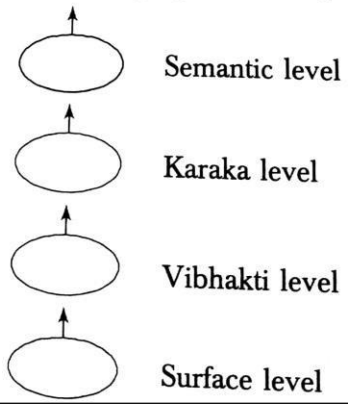
Information Extraction
- Captures and outputs factual information contained within a document.
- Responds to user's information.
- Specified as pre-defined database schemes or templates.
- It Identifies a subset of information within a document that fits the pre-defined template.

Question Answering
- Attempts to find precise answer, or precise portion of text in which the answer appears.
- Returns whole document that seems relevant to the user's query.
- Requires precise analysis of questions and portions of texts, as well as background knowledge to answer certain type of questions.

Text Summarization
- Deals with the creation of summaries of documents and involves syntactic and semantic of text.

Challenges:

- Representing & intrepreting NL is a challenging task.
- Natural languages are highly ambiguous and vague, achieving such representation can be difficult.
- It is almost impossible to embody all sources of knowledge that humans use to process language.
- Identifying the semantics in natural language is difficult.
- Words alone do not make sentence. It is their syntactic and semantic relation that give meaning to a sentence.
- A language keeps on evolving.

2.	Construct the C- structure and f-structure for the following sentence "**She saw stars**". Consider LFG rule.	[10]	CO1	L3
	LFG provides well-defined objects called ● constituent structure : It is derived from the usual phrase and sentence structure syntax. ● functional structure : when Functional specifications are applied to c-structure it results in f-structure. up arrow: refers to f-structure of mother node that is on left hand side of the rule. down arrow: refers to f-structure of node under which it is denoted.  Ex- She saw stars- C- Structure  Ex- She saw stars- F- Structure ● f-structure is the set of attribute-value pairs, represented as above ● final f-structure is obtained through unification of various f-structures for subject, object , verb, complement etc...			
3.	Identify different Karaka's in following sentence in hindi language (any sentence could be given) **maan Bachche ko aangan mein haath se rotii khilaatii hei**	[10]	CO1	L3
	Levels of Paninian Grammar:  			

	Karaka literally means CASE, these case relations are based on the way the word group participates in the activity denoted by the verb group. Karaka relations are assigned based on the roles players by various participants in main activity. Various karaka's are (case marker in hindi) 1. Karta (subject) - maan 2. Karma (Object) - rotii 3. Karana (instrument)- haath 4. Sampradana (beneficiary)- bachche 5. Apadan (separation)- ko, se/dwara, ke (Case marker)			
4.	Solve to find the probability of test sentence S2 in the following training set S1: **The Arabian Knights** S2: **These are the fairy tales of the east** S3: **The stories of the Arabian knights are translated in many languages**	[10]	CO1	L3
	Bi-gram model: P(the/<s>)=0.67 P(Arabian/the)=0.4 P(knights/Arabian)=1.0 P(are/these)=1.0 P(the/are)=0.5 P(fairy/the)=0.2 P(tales/fairy)=1.0 P(of/tales)=1.0 P(the/of)=1.0 P(east/the)=0.2 P(stories/the)=0.2 P(of/stories)=1.0 P(are/knights)=1.0 P(translated/are)=0.5 P(in/translated)=1.0 P(many/in)=1.0 P(languages/many)=1.0 Test sentence(s): The Arabian knights are the fairy tales of the east. P(The/<s>) x P(Arabian/the) x P(Knights/Arabian) x P(are/knights) x P(the/are) x P(fairy/the) x P(tales/fairy) x P(of/tales) x P(the/of) x P(east/the) **= 0.67 x 0.4 x 1.0 x 1.0 x 0.5 x 0.2 x 1.0 x 1.0 x 1.0 x 0.2** **=0.0067**			
6.	Explain Character classes with examples.	[10]	CO2	L2
	● Characters are grouped by putting them between square brackets. /[abcd]/ ● Any character in the class will match one character in the input. Example: the pattern/[abcd]/ will match a, b, c, d. ● Use of brackets specifies a disjunction of characters. ● The character classes are important building blocks in expressions. Example: It is inconvenient to write the regular expression /[abcdefghijklmnopqrstuvwxyz]/ to specify 'any lowercase letter'. ● A dash is used to specify a range. Example: /[5-9]/ specifies any one of the characters 5, 6, 7, 8, or 9. ● Regular expressions can also specify what a single character cannot be. it uses caret at the beginning. Example:			

/[^x]/ matches any single character except x.

- Regular expressions are case sensitive.

Example:

/s/ matches lower case ‘s’ but not uppercase ‘S’.

- The pattern /[sS]/ will match the string containing either s or S.
- A solution is needed to specify both ‘supernova’ and ‘supernovas’.
- The pattern /[sS]upernova[sS]/ does not match with the string ‘supernova’.
- This is achieved with the use of a question mark /?/.
- A question mark makes the preceding character optional, i.e., zero or one occurrence of the previous character.
- The regular expression /supernovas?/ specifies both ‘supernova’ and ‘supernovas’.
- The * operator called Kleene * specify repeated occurrences of a character.
- The * operator specifies zero or more occurrences of a preceding character or regular expression.
- The regular expression /b*/ will match any string containing zero or more occurrences of ‘b’.
- It will also match ‘aaa’, since that string contains zero occurrences of ‘b’.
- To match a string containing one or more ‘b’s the regular expression is /bb*/.
- This means ‘b’ followed by zero or more ‘b’s.
- The regular expression /[ab]*/ specifies zero or more ‘a’s or ‘b’s.
- This will match strings like ‘aa’, ‘bb’, or ‘abab’.
- The Kleene+ provides one or more occurrence of a character.
- Using Kleene+, we can specify a sequence of digits by the regular expression /[0-9]+/.
- The caret (^) is used to specify a match at the beginning of a line.
- The dollar sign($) is used to specify a match at the end of the line.
- If you want to search the line containing only the phrase ‘the nature’

/^the nature\.$/ --> this expression will search exactly only this line.

To check if string is an email address or not

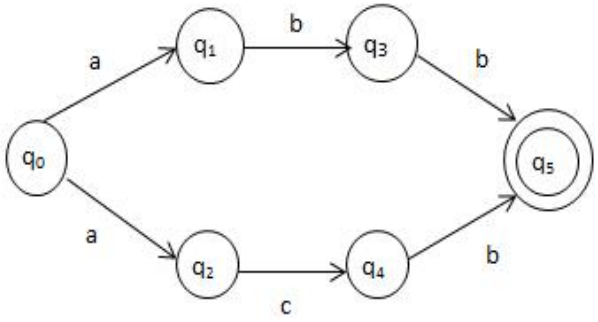
^[A-Za-z0-9_\.-]+ @[A-Za-z0-9_\.-]+[A-Za-z0-9_][A-Za-z0-9_]$

^[A-Za-z0-9_\.-]+ Match a positive number of acceptable characters at beginning of the string.

@ Match the @ sign

[A-Za-z0-9_\.-]+ Match any domain name, including a dot

[A-Za-z0-9_][A-Za-z0-9_]$ Match two acceptable characters but not a dot.

	This ensures that the email address ends with .xx, .xxx, .xxxx, etc.			
5	Construct NFA if **Σ={a,b,c}**, the set of **states={q0,q1,q2,q3,q4,q5}**,q0 being the start state and q5 the final state.	[10]	CO2	L3
	The below figure shows the 2 possible transitions from state q0 on input symbol a.  A path leading to 1 of the final states is a successful path. The FSAs encode regular languages. For automata with cycles, these sets are not finite. Set of all strings that lead to final state is the language accepted by the FA. We represent an automaton as state-transition table.			

Faculty Signature CCI Signature HOD Signature