CMR
INSTITUTE OF
TECHNOLOGY

USN □□□□□□□□□□

Third Internal Test- Nov 2019

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Sub: | **Natural Language Processing** | | | | | | | Code: | 15CS741 |
| Date: | 18/ 11 / 2019 | Duration: | 90 mins | Max Marks: | 50 | Sem: | VII | Branch: | **ISE (A&B)** |
| Answer Any **FIVE FULL** Questions | | | | | | | | | |

| | | Marks | OBE CO | OBE RBT |
|---|---|---|---|---|
| 1 (a) | Define precision and recall. Write a note on trade-off between precision and recall in evaluation of Information retrieval system | [10] | CO5 | L3 |

IR systems are evaluated by six metrics:

1. Coverage of the collection
2. Time lag
3. Presentation format
4. User effort
5. Precision
6. Recall

$$Precision=\frac{Number\ of\ relevant\ documents\ retrieved\ (NRret)}{Total\ number\ of\ documents\ retrieved(Nret)}$$

$$Recall=\frac{Number\ of\ relevant\ documents\ retrieved\ (NR\boldsymbol{ret})}{Total\ number\ of\ relevant\ documents\ in\ the\ collection\ (NRrel)}$$

*A= Set of relevant documents*

*B=Set of retrieved documents*

Trade-off between Precision and recall:

High value of both at the same time is desirable but precision is high when recall is low and as recall increases, precision decreases. The ideal case of perfect retrieval requires that all relevant documents be retrieved before the first non-relevant document is retrieved.

| | | Marks | OBE CO | OBE RBT |
|---|---|---|---|---|
| 2 (a) | What is WORDNET? What are its application in natural Language Processing? | [10] | CO5 | L2 |

WORDNET is a large database for English language created by Princeton University. Three databases: Noun, Verb, Adjectives and Adverbs

Information organized into sets of synonymous words called Synsets. Synsets are linked to each other by means of lexical and semantic relations. Relation includes synonymy, hyponymy, antonymy, meronymy/holonymy, troponymy.

WordNet for other languages : Hindi WordNet, EuroWordNet

**Example: A WORDNET entry for word READ is as below:**

**Noun**

- S: (n) **read** (something that is read) *"the article was a very good read"*

**Verb**

- S: (v) **read** (interpret something that is written or printed) *"read the advertisement"; "Have you read Salman Rushdie?"*
- S: (v) **read**, say (have or contain a certain wording or form) *"The passage reads as follows"; "What does the law say?"*
- S: (v) **read** (look at, interpret, and say out loud something that is written or printed) *"The King will read the proclamation at noon"*
- S: (v) **read**, scan (obtain data from magnetic tapes or other digital sources) *"This dictionary can be read by the computer"*

WORDNET Applications:

- Concept identifications in Natural language
- Word sense disambiguation
- Automatic Query Expansion
- Document structuring and categorization
- Document summarization

| | | | | |
|---|---|---|---|---|
| 3 (a) | List and explain the patterns used in extracting relationships between entities in string kernels. | [10] | CO4 | L2 |

Blaschke and ELCS methods do relation extraction based on a limited set of matching rules, where a rule is simply a sparse (gappy) subsequence of words or POS tags anchored on the two entities. when a sentence asserts a relationship between two entity mentions, it generally does this using one of the following four patterns:

1. **[FB] F**ore–**B**etween: Words before and between the two entity mentions are simultaneously used to express the relationship.

   Ex: **'interaction of *P*1 with *P*2,' 'activation of *P*1 by *P*2.'**
2. **[B] B**etween: Only words between the two entities are essential for asserting the relationship.

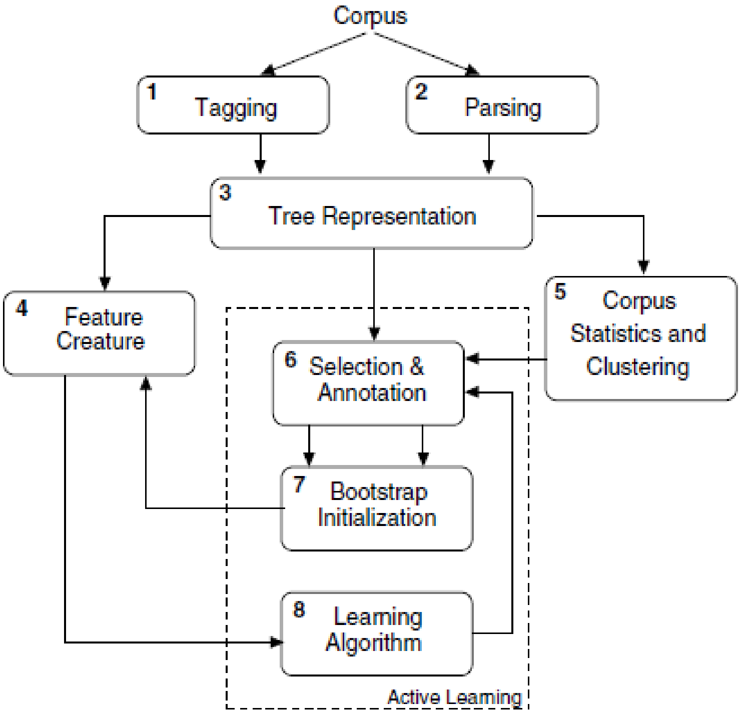
   Ex: **'*P*1 interacts with *P*2,' '*P*1 is activated by *P*2.'**
3. **[BA] B**etween–**A**fter: Words between and after the two entity mentions are simultaneously used to express the relationship.

   Examples: **'*P*1 – *P*2 complex,' '*P*1 and *P*2 interact.'**
4. **[M]M**odifier: The two entity mentions have no words between them.

   Ex: ***U.S. troops*** **(a Role:Staff relation),** ***Serbian general*** **(Role:Citizen).**

While the first three patterns are sufficient to capture most cases of interactions, last pattern is needed to account for various relationships expressed through noun-noun or adjective-noun compounds.

| | | | | |
|---|---|---|---|---|
| 4 (a) | With a neat diagram explain the architecture used in the task of learning to annotate cases with knowledge roles. | [10] | CO4 | L2 |



Learning framework architecture involves 8 different steps in it:

1. **Tagging :** Primary interest in using the tagger is not the POS tagging itself but also getting stem information  and dividing the paragraphs in to sentences.

2. **Parsing:** Syntactical parsing is one of the most important steps in the learning framework, since the produced parse trees serve as input for the creation of features used for learning. Three different parsers are used in learning framework are three different parsers: The Stanford parser, The BitPar parser, and the Sleepy parser.

3. **Tree representation:** The bracketed parse tree and the stem information of tagging serve as input for the step of creating a tree data structure. The tree is composed of terminals (leaf nodes) and non-terminals (internal nodes), all of them known as constituents of the tree. For export purposes as well as for performing exploration or annotation of the corpus, the tree data structures are stored in XML format, according to a schema defined in the TigerSearch tool.

4. **Feature creature**: Features are created from the parse tree of a sentence. A feature vector is created for every constituent of the tree, containing some features unique to the constituent, some features common to all constituents of the sentence, and some others calculated with respect to the target constituent.

5. **Selection and Annotation:**  The Salsa annotation tool reads the XML representation of a parse tree and displays it. The user has the opportunity to add frames and roles as well as to attach them to a desired target verb.

**Learning Algorithm:** Using an active learning approach for acquiring labels from a human annotator has advantages over other approaches of selecting instances for labeling. Active learning is designed using a committee-based classification scheme that is steered by **corpus statistics.**

5 (a) Define Shortest path hypothesis. With an example explain the relation extraction using dependency-graph. [10] CO4 L3

**Shortest path hypothesis**: If *e*1 and *e*2 are two entities mentioned in the same sentence such that they are observed to be in a relationship *R*, then the contribution of the sentence dependency graph to establishing the relationship *R*(*e*1*, e*2) is almost exclusively concentrated in the shortest path between *e*1 and *e*2 in the undirected version of the dependency graph.

**Relation Extraction using dependency path:**

- The shortest path between two entities in a dependency graph offers a very condensed representation of the information needed to assess their relationship.
- A dependency path is represented as a sequence of words interspersed with arrows that indicate the orientation of each dependency, dependency paths use both words and their word classes.

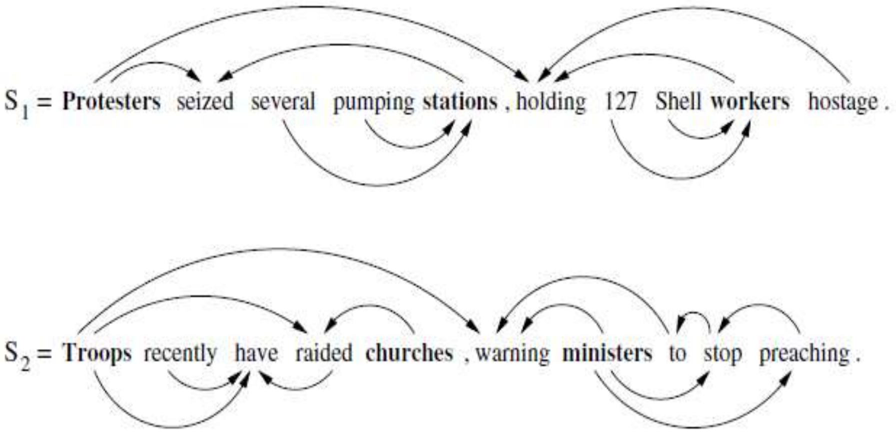
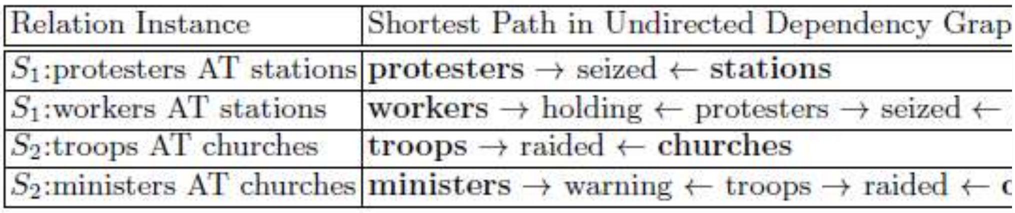

S$_1$ = Protesters seized several pumping stations , holding 127 Shell workers hostage .
S$_2$ = Troops recently have raided churches , warning ministers to stop preaching .

Fig. 3.4. Sentences as dependency graphs.

Table 3.1. Shortest Path representation of relations.

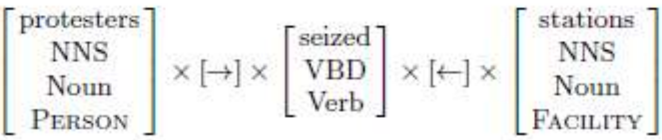


| Relation Instance | Shortest Path in Undirected Dependency Grap |
|---|---|
| $S_1$:protesters AT stations | **protesters** → seized ← **stations** |
| $S_1$:workers AT stations | **workers** → holding ← protesters → seized ← |
| $S_2$:troops AT churches | **troops** → raided ← **churches** |
| $S_2$:ministers AT churches | **ministers** → warning ← troops → raided ← c |

The set of features can be defined as a Cartesian product over words and word classes.

- Features generated by Figure are
  - "protesters → seized ← stations,"
  - "Noun → Verb ← Noun,"
  - "Person → seized ← Facility,"
  - "Person→ Verb ← Facility."
- The total number of features generated by this dependency path is

4 × 1 × 3 × 1 × 4



$$\begin{bmatrix} \text{protesters} \\ \text{NNS} \\ \text{Noun} \\ \text{PERSON} \end{bmatrix} \times [\rightarrow] \times \begin{bmatrix} \text{seized} \\ \text{VBD} \\ \text{Verb} \end{bmatrix} \times [\leftarrow] \times \begin{bmatrix} \text{stations} \\ \text{NNS} \\ \text{Noun} \\ \text{FACILITY} \end{bmatrix}$$

**Fig. 3.6.** Feature generation from dependency path.

| | | | | |
|---|---|---|---|---|
| 6 (a) | Explain the functioning of word matching feedback system used in iSTART system. | [10] | CO4 | L2 |

iSTART (Interactive Strategy Trainer for Active Reading and Thinking) is a webbased, automated tutor designed to help students become better readers via multimedia technologies. It provides young adolescent to college-aged students with a program of self-explanation and reading strategy training called Self-Explanation Reading Training, or SERT.

Word Matching: Word matching is a very simple and intuitive way to estimate the nature of a self-explanation. A trainee's explanation is analyzed by matching the words in the explanation against the words in the target sentence and words in the corresponding association lists. This was accomplished in two ways:
(1) Literal word matching and
(2) Soundex matching.

**Literal word matching** - Words are compared character by character and if there is a match of the first 75% of the characters in a word in the target sentence (or its association list) then we call this a literal match. This also includes removing suffix -s, -d, -ed, -ing, and -ion at the end of each words.

**Soundex matching -** This algorithm compensates for misspellings by mapping similar characters to the same Soundex symbol. Words are transformed to their Soundex code by retaining the first character, dropping the vowels, and then converting other characters into Soundex symbols. If the same symbol occurs more than once consecutively, only one occurrence is retained. For example, 'thunderstorm' will be transformed to 't8693698'; 'communication' to 'c8368.' If the trainee's self-explanation contains 'thonderstorm' or 'tonderstorm,' both will be matched with 'thunderstorm' and this is called a soundex match.

| | | | | |
|---|---|---|---|---|
| 7 (a) | Explain Cluster and Fuzzy models of Information retrieval system | [10] | CO5 | L2 |

Alternative models are neither classical nor non-classical models, they are Enhancements of classical models making use of specific techniques from other fields.
Ex: Cluster model, fuzzy model and latent semantic indexing (LSI) models

- **Cluster Model:**

  Attempts to reduce the number of matches during retrieval based on cluster hypothesis:
  "Closely associated documents tend to be relevant to the same clusters"
  And hence instead of matching a query with individual documents, it is matched with representatives of the cluster(class), and only documents from a class whose representative is close to query, are considered for individual match. Clustering can also be applied on terms based on co-occurrence.

  - Let D={ d1,d2,d3,.....dm} be set of documents.

  - Let $(e_{ij})_{n,n}$ be the similarity matrix, element $E_{i,j}$ denotes a similarity between document $d_i$ and $d_j$ if similarity measure exceed threshold value then are grouped to form a cluster.

- **Fuzzy Model:**

  Document is represented as a fuzzy set of terms $[t_i ,\mu(t_i )]$

  Where μ is a member function that assigns to each term of the document a numeric membership degree.

  Ex: **d1={information, retrieval, query}**

  **d2={retrieval, query, model}**

  **d3={information, retrieval}**

  **T={information, model, query, retrieval}**

  Fuzzy set for terms

  **f1={(d1,1/3),(d2,0),(d3,1/2)}**

  **f2={(d1,0),(d2,1/3),(d3,0)}**

  **f3={(d1,1/3),(d2,1/3),(d3,0)}**

  **f4={(d1,1/3),(d2,1/3),(d3,1/2)}**

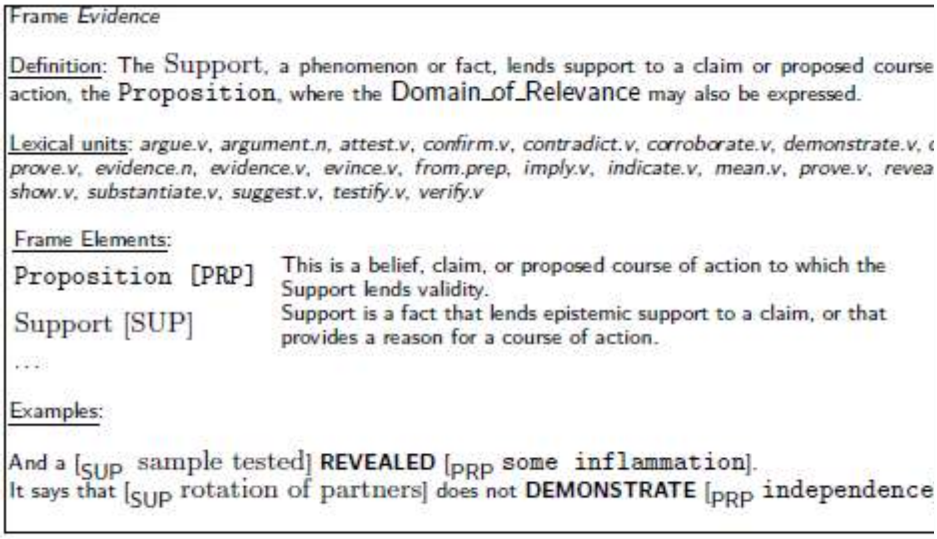
If the query is $t2 \wedge t4$, then document d2 is returned.

| | | | | |
|---|---|---|---|---|
| 8 (a) | Write a note on Frame Semantics and Semantic Role Labeling. | [10] | CO4 | L1 |

**Frame Semantics:** frame is a "script-like conceptual structure that describes a particular type of situation, object, or event and the participants involved in it" . Based on this theory, the Berkeley FrameNet Project1 is creating an online lexical resource for the English language by annotating text from the 100 million words British National Corpus.

The structure of a frame contains lexical units (pairs of a word with its meaning), frame elements (semantic roles played by different syntactic dependents), as well as annotated sentences for all lexical units that evoke the frame.

A frame for Evidence is as below:



Frame *Evidence*

Definition: The Support, a phenomenon or fact, lends support to a claim or proposed course action, the Proposition, where the Domain_of_Relevance may also be expressed.

Lexical units: *argue.v, argument.n, attest.v, confirm.v, contradict.v, corroborate.v, demonstrate.v, prove.v, evidence.n, evidence.v, evince.v, from.prep, imply.v, indicate.v, mean.v, prove.v, revea show.v, substantiate.v, suggest.v, testify.v, verify.v*

Frame Elements:

| | |
|---|---|
| Proposition [PRP] | This is a belief, claim, or proposed course of action to which the Support lends validity. |
| Support [SUP] | Support is a fact that lends epistemic support to a claim, or that provides a reason for a course of action. |
| . . . | |

Examples:

And a [$_{\text{SUP}}$ sample tested] **REVEALED** [$_{\text{PRP}}$ some inflammation].
It says that [$_{\text{SUP}}$ rotation of partners] does not **DEMONSTRATE** [$_{\text{PRP}}$ independence

**Semantic Role Labeling:** Annotation of text with frames and roles in FrameNet has been performed manually by trained linguists. Semantic interpretation of text in terms of frames and roles would contribute to many applications, like question answering, information extraction, semantic dialogue systems, as well as statistical machine translation or automatic text summarization, and finally also to text mining. Research on semantic role labeling (SRL) has grown steadily, and in the years 2004 and 2005 a shared task at the CoNLL2 was defined, in which several research institutions compared their systems. In the meantime, besides FrameNet, another corpus with manually annotated semantic roles has been prepared, PropNet , which differs from FrameNet in the fact that it has general semantic roles not related to semantic frames.

SRL is approached as a learning task. For a given target verb in a sentence, the syntactic constituents expressing semantic roles associated to this verb need to be identified and labeled with the right roles. SRL systems usually divide sentences word-by-word or phrase-by-phrase and for each of these instances calculate many features creating a feature vector. The feature vectors are then fed to supervised classifiers, such as support vector machines, maximum entropy, or memory-based learners.