PVP 23

Code: 23CS4602C

III B.Tech - II Semester - Regular Examinations – APRIL 2026

NATURAL LANGUAGE PROCESSING
(COMPUTER SCIENCE & ENGINEERING)

Duration: 3 hours Max. Marks: 70

Note: 1. This question paper contains two Parts A and B.
2. Part-A contains 10 short answer questions. Each Question carries 2 Marks.
3. Part-B contains 5 essay questions with an internal choice from each unit. Each Question carries 10 marks.
4. All parts of Question paper must be answered in one place.

BL – Blooms Level CO – Course Outcome

PART – A

		BL	CO
1.a)	Define unigram and bigram models.	L1	CO1
1.b)	Describe perplexity in language model evaluation.	L1	CO1
1.c)	Define Maximum Likelihood Estimation (MLE) in n-gram models.	L1	CO1
1.d)	What is data sparsity problem?	L1	CO1
1.e)	Define treebank.	L1	CO1
1.f)	What is Chomsky Normal Form?	L1	CO1
1.g)	Define Word Sense Disambiguation (WSD).	L1	CO1
1.h)	Define First Order Logic.	L1	CO1
1.i)	Define anaphora with example.	L1	CO1
1.j)	Tell the use of stemming and lemmatization.	L1	CO1

PART – B

			BL	CO	Max. Marks
		UNIT-I			
2	a)	Explain the origins and challenges of Natural Language Processing.	L2	CO1	5 M
	b)	Differentiate grammar-based and statistical language models with examples.	L2	CO1	5 M
		OR			
3	a)	Explain a multilingual tokenization strategy for NLP systems.	L2	CO1	5 M
	b)	Discuss types of language models: Statistical, Neural, and Transformer-based.	L2	CO1	5 M
		UNIT-II			
4	a)	Demonstrate Hidden Markov Model for POS tagging with Viterbi algorithm.	L3	CO2	5 M
	b)	Apply Add-One smoothing technique to a given dataset.	L3	CO2	5 M
		OR			
5	a)	Illustrate rule-based, stochastic, and transformation-based POS tagging methods.	L3	CO2	5 M
	b)	Identify the benefits of word classes in language modeling and illustrate the open	L3	CO2	5 M

		class and closed class words.			
		UNIT-III			
6	a)	Apply structural and attachment ambiguity in sentences.	L3	CO2	5 M
	b)	Demonstrate CYK algorithm with suitable example.	L3	CO2	5 M
		OR			
7	a)	Relate constituency parsing vs dependency parsing.	L3	CO2	5 M
	b)	Illustrate a shallow parsing system for information extraction.	L3	CO2	5 M
		UNIT-IV			
8	a)	Explain requirements of meaning representation in NLP.	L4	CO4	5 M
	b)	Explain natural language sentences using First Order Logic.	L4	CO4	5 M
		OR			
9	a)	Differentiate polysemy and homonymy with examples.	L4	CO4	5 M
	b)	Compare and contrast supervised and knowledge-based WSD techniques.	L4	CO4	5 M
		UNIT-V			
10	a)	Illustrate discourse segmentation techniques.	L3	CO3	5 M
	b)	Apply coherence relations using Rhetorical Structure Theory (RST).	L3	CO3	5 M

		OR			
11	a)	Demonstrate Brown Corpus and British National Corpus in NLP research.	L3	CO3	5 M
	b)	Use of discourse segmentation framework for dialogue systems.	L3	CO3	5 M

Code: 20CS4502C

PVP-23

PRASAD V POTLURI SIDDHARTHA INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

III B. Tech- II Semester- Regular Examinations- APRIL 2026

NATURAL LANGUAGE PROCESSING

Duration: 3 Hours **Max. Marks** : 70

PART-A

1 a) Define Unigram and Bigram Models. **[CO1-L1] [2M]**

Ans:

Unigram Model:

A language model that assumes each word occurs independently of other words. It calculates the probability of a sentence as the product of individual word probabilities.

Example: $P(w_1, w_2, ..., w_n) = P(w_1) \times P(w_2) \times ... \times P(w_n)$

Bigram Model:

A language model that considers the probability of a word based on the previous word. It captures dependencies between consecutive words.

Example: $P(w_1, w_2, ..., w_n) = P(w_1) \times P(w_2|w_1) \times P(w_3|w_2) \times ... \times P(w_n|w_{n-1})$

b) Define Perplexity in Language Model Evaluation **[CO1-L1] [2M]**

Ans: **Perplexity** is a measure used to evaluate how well a language model predicts a sequence of words. It indicates how "confused" the model is when predicting the next word.

- Lower perplexity → better model (more accurate predictions)
- Higher perplexity → worse model (more uncertainty)

c) Define Maximum Likelihood Estimation(MLE) in n-gram Models. **[CO1-L1] [2M]**

Ans: MLE is a method used to estimate the probability of word sequences by using their frequency counts in a training corpus. In n-gram models, the probability of a word given its previous n−1 words is calculated as:

$$P(w_n \mid w_1^{n-1}) = \frac{\text{Count}(w_1, w_2, ..., w_n)}{\text{Count}(w_1, w_2, ..., w_{n-1})}$$

d) What is Data Sparsity Problem? **[CO1-L1] [2M]**

Ans: It occurs when many possible word sequences (n-grams) are missing or rarely seen in the training data, leading to zero or unreliable probability estimates for unseen words or sequences.

e) Define Treebank? **[CO1-L1] [2M]**

Ans: A treebank is a linguistically annotated corpus in which sentences are labeled with syntactic structure (parse trees), showing grammatical relationships between words.

f) What is Chomsky Normal Form? **[CO1-L1] [2M]**

Ans: A form of context-free grammar where each production rule is either A→BC or A→a, i.e., two non-terminals or one terminal.

g) Define Word Sense Disambiguation? **[CO1-L1] [2M]**

Ans: It is the process of identifying the correct meaning of a word when it has multiple meanings, based on the context in which it is used.

Example:

The word *"bank"* can mean a financial institution or the side of a river. In the sentence *"He sat near the river bank,"* WSD helps identify that *"bank"* refers to the river side.

h) Define First Order Logic? **[CO1-L1] [2M]**

Ans: It is a formal system used to represent knowledge about objects and their relationships using predicates, constants, variables, functions, and quantifiers such as ∀ (for all) and ∃ (there exists).It allows expressing facts and rules logically for reasoning.

Example:

$\forall x\,(Human(x) \rightarrow Mortal(x))$ means "All humans are mortal."

i) Define anaphora with example **[CO1-L1] [2M]**

Ans: Anaphora is a linguistic phenomenon where a word (usually a pronoun) refers back to another word or phrase mentioned earlier in the sentence or discourse (called the antecedent). It helps in maintaining coherence and avoiding repetition.

Example:

Ravi went to school. He was late.

Here, **"He"** refers to **"Ravi."**

j) Tell the use of Stemming and Lemmatization **[CO1-L1] [2M]**

Ans: They are techniques used in NLP to reduce words to their base or root form.

- **Stemming:** removes suffixes to get a root form (e.g., *playing → play*).
- **Lemmatization:** converts a word to its meaningful base form using vocabulary (e.g., *better → good*).

Use: Helps in improving accuracy in tasks like search engines, text classification, and information retrieval by treating similar words as the same.

PART-B

UNIT-I

2 a) Explain the origins and challenges of Natural Language Processing **[CO1-L2] [5M]**

Ans:

Origins of NLP:

The origins of NLP date back to the **1950s**, when researchers began exploring machine translation. Initially, NLP was based on **rule-based approaches**, where linguists created grammar rules manually. In the **1980s–1990s**, NLP shifted towards **statistical methods**, using probability and large corpora. In recent years, NLP has advanced with **machine learning and deep learning techniques**, such as neural networks and transformers, leading to applications like chatbots, speech recognition, and language translation systems.

Challenges of NLP:

1. **Ambiguity:**
   Words and sentences can have multiple meanings (e.g., *bank* as river side or financial institution).
2. **Context Understanding:**
   Understanding meaning depends on context, which is difficult for machines.
3. **Syntax and Grammar Complexity:**
   Human languages have complex and flexible grammatical structures.
4. **Semantic Understanding:**
   Machines struggle to capture the exact meaning and relationships between words.
5. **Data Sparsity:**
   Limited data for rare words or phrases affects model performance.
6. **Multilingual and Cultural Variations:**
   Different languages and dialects make processing more complex.

2 b) Differentiate grammar-based and statistical language models with example **[CO1-L2] [5M]**

Ans:

Aspect	Grammar-Based Models	Statistical Language Models
Approach	Use predefined linguistic rules	Use probabilities from data
Basis	Syntax and grammar rules	Frequency of word sequences
Flexibility	Less flexible	More flexible and adaptive
Data Requirement	Do not require large datasets	Require large corpora
Example	Rule: $S \rightarrow NP + VP$	Bigram: $P(w_n \mid w_{n-1})$

Example:

- **Grammar-Based Models**: *"The boy eats food"* is valid based on grammar rules.
- **Statistical Language Models:** Probability of *"eats food"* is calculated based on how often it appears in data.

(OR)

3 a) Explain the Multilingual tokenization strategy for NLP systems [CO1-L2] [5M]

Ans: Tokenization is the process of breaking text into smaller units called tokens. In multilingual NLP, it must handle different languages, scripts, and structures.

Strategy:

1. **Unicode Representation:**
   All languages are represented using a common encoding (UTF-8), enabling processing of multiple scripts.
2. **Word-Level Tokenization:**
   Splitting text based on spaces and punctuation:
   $$\text{Text} \rightarrow \{w1, w2, w3, \ldots, wn\}$$
3. **Subword Tokenization (BPE / WordPiece):**
   Words are split into smaller units to handle rare words.
   $$w \rightarrow \{s1, s2, \ldots, sk\}$$
   Example: *unhappiness* → *un* + *happi* + *ness*
4. **Probability-Based Segmentation:**
   Tokenization can be based on maximizing likelihood:
   $$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i \mid w_1^{i-1})$$
5. **Vocabulary Reduction:**
   Subword models reduce vocabulary size:
   $$|V_{subword}| < |V_{word}|$$
6. **Handling Multilingual Data:**
   Shared vocabulary across languages improves learning and transfer.

Advantages:

- Handles multiple languages
- Reduces unknown words (OOV problem)
- Improves model efficiency

3 b) Discuss types of Language models: Statistical, Neural, and Transformer based.

[CO1-L2] [5M]

Ans: Language models predict the probability of word sequences and are widely used in NLP applications like machine translation, chatbots, and text generation.

1. **Statistical Language Models (SLMs) :** Statistical Language Models estimate the probability of a sequence of words using statistical techniques based on observed frequencies in a corpus. They assume that the probability of a word depends on a limited number of previous words (Markov assumption).

Mathematical Notation

For a sentence $W = (w_1, w_2, ..., w_n)$:

$$P(W) = P(w_1, w_2, ..., w_n) = \prod_{i=1}^{n} P(w_i \mid w_1, ..., w_{i-1})$$

Using n-gram approximation:

$$P(w_i \mid w_1, ..., w_{i-1}) \approx P(w_i \mid w_{i-(n-1)}, ..., w_{i-1})$$

Example (Bigram model):

$$P(w_i \mid w_{i-1}) = \frac{\text{Count}(w_{i-1}, w_i)}{\text{Count}(w_{i-1})}$$

Example

Sentence: *"I love AI"*

Bigram probability:

$$P(I, love, AI) = P(I) \cdot P(love \mid I) \cdot P(AI \mid love)$$

2. Neural Language Models (NLMs) : Neural Language Models use neural networks to learn word representations (embeddings) and predict the next word in a sequence. They overcome sparsity by learning distributed representations.

Mathematical Notation

$$P(w_t \mid w_1, ..., w_{t-1}) = \text{Softmax}(f(w_1, ..., w_{t-1}))$$

Where:

- f = Neural network (Feedforward / RNN / LSTM)
- Softmax:

$$P(w_t = w \mid context) = \frac{e^{z_w}}{\sum_{w'} e^{z_{w'}}}$$

Example

Input: *"I love"*
Model predicts next word:

$$P(AI \mid I, love) = 0.7$$

2. **Transformer-based Language Models:** Transformer models use self-attention mechanisms to capture relationships between all words in a sequence simultaneously, without recurrence or convolution.

Mathematical Notation

Self-Attention Mechanism

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Where:

- Q = Query
- K = Key
- V = Value
- d_k = dimension

UNIT-II

4 a) Demonstrate Hidden Markov Model for POS tagging with Viterbi algorithm. (CO2-L3) [5M]

Ans: A **Hidden Markov Model (HMM)** is a probabilistic model used for sequence labeling tasks like **Part-of-Speech (POS) tagging**, where:

- **States** = POS tags (hidden)
- **Observations** = Words (visible)

The goal is to find the most probable sequence of POS tags for a given sentence.

2. HMM Components

- **States (S):** POS tags (Noun, Verb, Adj, etc.)
- **Observations (O):** Words in sentence
- **Transition Probability (A):**

$$P(t_i \mid t_{i-1})$$

- **Emission Probability (B):**

$$P(w_i \mid t_i)$$

- **Initial Probability (π):**

$$P(t_1)$$

Example

Sentence: **"Time flies"**

Possible tags:

- Time → Noun (N), Verb (V)
- flies → Noun (N), Verb (V)

Given Probabilities:

- Transition:
  - $P(N|START) = 0.6, P(V|START) = 0.4$
  - $P(V|N) = 0.7, P(N|N) = 0.3$
- Emission:
  - $P(Time|N) = 0.8, P(Time|V) = 0.2$
  - $P(flies|V) = 0.6, P(flies|N) = 0.4$

Step 1: Initialization

$$V_1(N) = 0.6 \times 0.8 = 0.48$$

$$V_1(V) = 0.4 \times 0.2 = 0.08$$

Step 2: Recursion

$$V_2(V) = \max[(0.48 \times 0.7), (0.08 \times 0.4)] \times 0.6 = 0.2016$$

$$V_2(N) = \max[(0.48 \times 0.3), (0.08 \times 0.6)] \times 0.4 = 0.0576$$

Step 3: Result

Best tag sequence:

Time (Noun), flies (Verb)

4 b) Apply Add-One smoothing technique to a given dataset. (CO2-L3) [5M]

Ans: Add-One smoothing (Laplace smoothing) is used in language models to handle **zero probabilities** by adding 1 to all frequency counts.This ensures that unseen events get a small non-zero probability.

2. Formula

For probability estimation:

$$P(w_i \mid w_{i-1}) = \frac{\text{Count}(w_{i-1}, w_i) + 1}{\text{Count}(w_{i-1}) + V}$$

Where:

- V = Vocabulary size
- $\text{Count}(w_{i-1}, w_i)$ = Bigram count
- $\text{Count}(w_{i-1})$ = Unigram count

4. Apply Add-One Smoothing

Example 1:

$$P(like \mid I) = \frac{2+1}{2+4} = \frac{3}{6} = 0.5$$

(OR)

5 a) Illustrate rule-based, stochastic, and transformation-based POS tagging methods. (CO2-L3) [5M]

Ans:

1. Rule-Based POS Tagging: Uses hand-crafted linguistic rules and dictionaries to assign tags based on grammar and context.

How it Works:

- Uses lexicon (word → possible tags)
- Applies rules like:
  - If word ends with "-ing" → Verb
  - If previous word is "the" → likely Noun

Example : Sentence: *"The cat runs":*

- "The" → Determiner
- Rule: Determiner + Noun → "cat" = Noun
- "runs" → Verb

2. Stochastic (Probabilistic) POS Tagging: Uses **probability models** (like HMM) to assign tags based on likelihood from training data.

Formula

$$T^* = \arg\max_T P(W|T) \cdot P(T)$$

(Bigram assumption)

$$P(t_i \mid t_{i-1}), \quad P(w_i \mid t_i)$$

Example

Sentence: *"Time flies"*

Possible tags:

- Time → Noun / Verb
- flies → Noun / Verb

Model chooses:

Time (Noun), flies (Verb)

based on higher probability.

3. Transformation-Based Tagging (Brill Tagger): Combines rule-based and statistical methods by **learning rules automatically** from data.

How it Works

1. Assign initial tags (e.g., most frequent tag)
2. Apply transformation rules to correct errors

Example

Sentence: *"She can swim"*

Initial tagging:

- can → Noun

Transformation rule:

- If previous word is pronoun → change Noun → Verb

Final:

- can → Verb

5 b) Identify the benefits of word classes in language modeling and illustrate the open classes and closed class words **(CO2-L3) [5M]**

Ans:

1. Definition

Word classes (or categories) group words based on their **syntactic and semantic roles** (e.g., Noun, Verb, Adjective).

In language modeling, grouping words into classes helps improve probability estimation.

2. Benefits of Word Classes in Language Modeling

1. Reduces Data Sparsity

- Instead of learning probabilities for each word, we learn for **classes**.
- Example:

$$P(dog \mid the) \approx P(\text{Noun} \mid the)$$

3. Reduces Model Complexity

- Vocabulary size V is large → classes C are smaller
- Computation becomes efficient

---

4. Better Probability Estimation

Class-based model:

$$P(w_i \mid w_{i-1}) = P(w_i \mid c_i) \cdot P(c_i \mid c_{i-1})$$

Where:

- c_i = class of word w_i

2. Improves Generalization

- Words in the same class behave similarly.
- Example: "dog", "cat", "car" → all nouns

3. Open Class Words

Definition

Open class words are categories that **frequently accept new words**.

Types & Examples

- **Nouns** → dog, computer, AI
- **Verbs** → run, learn, compute
- **Adjectives** → big, intelligent
- **Adverbs** → quickly, softly

4. Closed Class Words

Definition

Closed class words have a **fixed set** and rarely accept new words.

Types & Examples

- **Pronouns** → he, she, it
- **Prepositions** → in, on, at
- **Conjunctions** → and, but
- **Determiners** → the, a, an

5. Example:

Word	Class Type	Category
The	Closed	Determiner
cat	Open	Noun
runs	Open	Verb
quickly	Open	Adverb

UNIT-III

6 a) Apply structural and attachment ambiguity in sentences. (CO2-L3) [5M]

Ans: Ambiguity in Natural Language Processing (NLP) occurs when a sentence has **more than one possible interpretation**.
Two important types are:

- **Structural (Syntactic) Ambiguity**
- **Attachment Ambiguity**

These arise due to different possible **parse trees** of a sentence.

a) **Structural Ambiguity:** Structural ambiguity occurs when a sentence can be parsed in multiple grammatical structures, leading to different meanings.

Example 1

Sentence:

"I saw the man with a telescope."

Interpretation 1

- *I used a telescope to see the man.*

Interpretation 2

- *The man I saw had a telescope.*

Parse Representation (Simplified)

1. **[VP saw [NP the man] [PP with a telescope]]**
   → Telescope used by the observer
2. **[VP saw [NP the man [PP with a telescope]]]**
   → Telescope belongs to the man

b) **Attachment Ambiguity:**

Attachment ambiguity occurs when it is unclear **which part of the sentence a phrase (usually a prepositional phrase) attaches to.**

Example

Sentence:

"She saw the boy with a dog."

Interpretation 1

- She saw [the boy who has a dog]

Interpretation 2

- She used a dog to see the boy (less common, but grammatically valid)

6 b) Demonstrate CYK algorithm with suitable example (CO2-L3) [5M]

Ans:

CYK Algorithm (Cocke–Younger–Kasami Algorithm)

1. Definition

The **CYK algorithm** is a **dynamic programming parsing algorithm** used to determine whether a given sentence can be generated by a **Context-Free Grammar (CFG)** in **Chomsky Normal Form (CNF).**

2. Key Requirements

- Grammar must be in **CNF**:
  - $A \to BC$ (two non-terminals)
  - $A \to a$ (single terminal)

---

3. Idea of CYK

- Uses a **triangular table (matrix)**
- Bottom-up parsing
- Each cell stores possible non-terminals that can generate a substring

4. Algorithm Steps

1. Convert CFG → CNF
2. Create table $T[i, j]$
   - i = start position
   - j = length of substring
3. Fill first row using terminal rules
4. Fill remaining cells using combinations
5. If start symbol S is in top cell → sentence is accepted

5. Example

Grammar (CNF)

$$S \to AB \mid BC$$

$$A \to BA \mid a$$

$$B \to CC \mid b$$

$$C \to AB \mid a$$

Input Sentence

$$w = "baaba"$$

6. **Step 1: Initialization (Length = 1)**

Position	Word	Possible Variables
1	b	B
2	a	A, C
3	a	A, C
4	b	B
5	a	A, C

7. **CYK Table Filling**

Length = 2

- Combine adjacent cells using rules

Example:

- (1,2) = combine B & (A,C) → possible: A, S
- (2,3) = combine (A,C) & (A,C) → B
- (3,4) = combine (A,C) & B → S, C
- (4,5) = combine B & (A,C) → A, S

Length = 3

- (1,3), (2,4), (3,5) computed similarly

Final Table (Top Cell)

$$T[1,5] = \{S, A, C\}$$

Result

Since **Start symbol** S is present in the top cell:

The string "baaba" is accepted by the grammar

(OR)

7 a) Relate constituency parsing vs dependency parsing. (CO2-L3) [5M]

Ans:

Constituency Parsing vs Dependency Parsing

1. Introduction

Parsing is the process of analyzing the **syntactic structure** of a sentence.
Two major approaches are:

- **Constituency Parsing (Phrase Structure Parsing)**
- **Dependency Parsing**

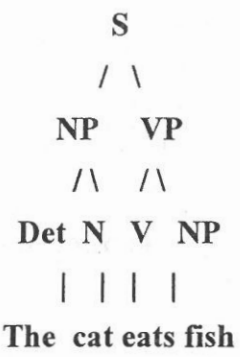
2. Constituency Parsing

Definition

Constituency parsing represents a sentence as a **hierarchical tree of phrases (constituents)** such as NP (Noun Phrase) and VP (Verb Phrase).

Example

Sentence: *"The cat eats fish"*



Characteristics

- Focus on **phrase structure**
- Uses grammar rules like:

$$S \rightarrow NP\ VP$$

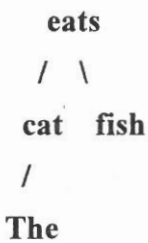
3. Dependency Parsing

Definition

Dependency parsing represents a sentence as a **set of binary relations** between words, showing how each word depends on another.

Example

Sentence: *"The cat eats fish"*



Relations:

- eats → cat (subject)
- eats → fish (object)
- cat → The (determiner)

Comparision:

Feature	Constituency Parsing	Dependency Parsing
Structure	Phrase tree	Dependency graph
Focus	Phrases (NP, VP)	Word relations
Grammar	CFG-based	Dependency grammar
Representation	Hierarchical	Head-dependent
Complexity	More complex	Simpler
Example	NP, VP nodes	subject, object links

7 b) Illustrate a shallow parsing system for information extraction. (CO2-L3) [5M]

Ans:

Shallow parsing (chunking) is a technique in NLP that identifies **basic syntactic units (chunks)** such as noun phrases (NP), verb phrases (VP), without building a full parse tree.
It is widely used in **Information Extraction (IE)** to quickly identify relevant information from text.

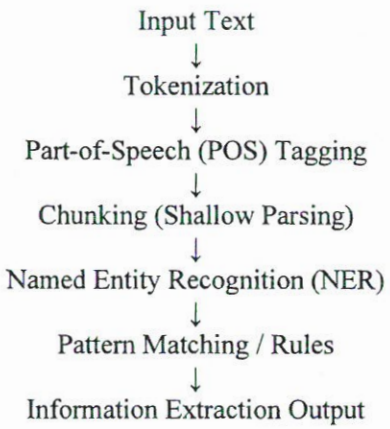
2. What is Information Extraction (IE)?

Information Extraction is the process of extracting **structured information** (entities, relationships, events) from unstructured text.

Example output:

- Person: *Ravi*
- Organization: *Google*
- Action: *joined*

3. Architecture of Shallow Parsing System



1. Tokenization

- Splits text into words
  Example: "Ravi joined Google" → [Ravi, joined, Google]

2. POS Tagging

- Assigns grammatical tags
  Example:
- Ravi → NNP
- joined → VBD
- Google → NNP

---

3. Chunking (Shallow Parsing)

- Groups words into phrases

Example:

- [Ravi] → NP
- [joined] → VP
- [Google] → NP

4. Named Entity Recognition (NER)

- Identifies entities like:
  - Person → Ravi
  - Organization → Google

5. Pattern Matching

- Uses rules/templates to extract information

Example Rule:

- NP + VP + NP → (Person Action Organization)

Example: Sentence:
"Ravi joined Google in 2022."

Stage	Output
Tokens	Ravi, joined, Google, in, 2022
POS Tags	NNP VBD NNP IN CD
Chunks	[Ravi]NP [joined]VP [Google]NP
NER	Ravi(Person), Google(Org), 2022(Date)
Extraction	(Ravi, joined, Google, 2022)

UNIT-IV

8 a) Explain requirements of meaning representation in NLP. (CO4-L4) [5M]

Ans: Meaning representation in Natural Language Processing (NLP) is the process of converting natural language sentences into a formal, machine-understandable structure that captures the semantics (meaning) of the sentence.

Requirements of Meaning Representation

A good meaning representation system should satisfy the following:

1. **Unambiguity**
   Each sentence should have a clear and unique interpretation.
   - Example: "I saw a man with a telescope" should resolve who has the telescope.
2. **Expressiveness**
   It should represent all kinds of meanings including:
   - Entities, actions, events, time, negation, quantifiers
   - Example: "Every student passed" vs "Some students passed"
3. **Canonical Form**
   Different sentences with the same meaning should have the same representation.
   - "John loves Mary" = "Mary is loved by John"
4. **Inference Capability**
   The representation should support logical reasoning.
   - From: "All humans are mortal" and "Socrates is human"
   - Infer: "Socrates is mortal"
5. **Compositionality**
   The meaning of a sentence should be built from meanings of its parts.
   - Sentence meaning = combination of word meanings
6. **Handling Ambiguity**
   Should deal with lexical and structural ambiguities.

Mathematical Notation (Using First Order Logic)

Meaning is often represented using **First Order Predicate Logic (FOPL)**.

General Form:

- Predicate(argument1, argument2, ...)

Example Symbols:

- $\forall$: "for all"
- $\exists$: "there exists"
- $\rightarrow$: implies
- $\wedge$: and
- $\neg$: not

Example

Sentence:
"All humans are mortal."

Representation:

$$\forall x\ (Human(x) \rightarrow Mortal(x))$$

Another Sentence:
"John is a human."

$$Human(John)$$

Inference:

$$Mortal(John)$$

8 b) Explain natural language sentences using First Order Logic. (CO4-L4) [5M]

Ans: First Order Logic (FOL), also called Predicate Logic, is a formal system used to represent the meaning of natural language sentences using predicates, variables, constants, functions, and quantifiers. It helps convert human language into a machine-understandable logical form for reasoning.

Basic Components of FOL

1. **Constants** – Specific objects
   - Example: John, Mary
2. **Variables** – General objects
   - Example: x, y
3. **Predicates** – Properties or relations
   - Example: Human(x), Loves(John, Mary)
4. **Quantifiers**
   - ∀ (Universal): "for all"
   - ∃ (Existential): "there exists"
5. **Logical Connectives**
   - ∧ (AND), ∨ (OR), ¬ (NOT), → (IMPLIES)

Steps to Represent Sentences in FOL

1. Identify **objects (nouns)**
2. Identify **relations (verbs)**
3. Identify **quantifiers (all, some, a)**
4. Construct logical expression

Examples

1. Simple Sentence
Sentence: John is a student
FOL:

$$Student(John)$$

---

2. Universal Statement
Sentence: All humans are mortal

$$\forall x \, (Human(x) \rightarrow Mortal(x))$$

---

3. Existential Statement
Sentence: Some students are intelligent

$$\exists x \, (Student(x) \land Intelligent(x))$$

---

4. Relation Between Objects
Sentence: John loves Mary

$$Loves(John, Mary)$$

5. Sentence with Two Variables
Sentence: Every student reads a book

$$\forall x \, (Student(x) \rightarrow \exists y \, (Book(y) \land Reads(x, y)))$$

---

6. Negation
Sentence: John is not a teacher

$$\neg Teacher(John)$$

---

7. Conditional Statement
Sentence: If it rains, the ground is wet

$$Rain \rightarrow Wet(Ground)$$

(OR)

9 a) Differentiate polysemy and homonymy with examples (CO4-L4) [5M]

Ans:

Definition

- **Polysemy**: A single word having **multiple related meanings**.
- **Homonymy**: Different words having the **same spelling or pronunciation but unrelated meanings**.

Feature	Polysemy	Homonymy
Meaning	Related meanings	Unrelated meanings
Origin	Same origin	Different origins
Sense Connection	Semantically connected	No connection
Dictionary Entry	Usually one entry with multiple senses	Separate entries
Example Nature	Extension of meaning	Coincidental similarity

Homonymy Examples

1. Word: "Bank"
   - Financial institution
   - River bank

   → Completely unrelated meanings
2. Word: "Bat"
   - Flying animal
   - Cricket bat

   → No semantic connection

Polysemy Examples

1. Word: "Head"
   - Head of a person
   - Head of a company

   → Both relate to "top or leader"
2. Word: "Foot"
   - Foot of a person
   - Foot of a mountain

   → Both indicate "bottom part"

9 b) Compare and contrast supervised and knowledge-based WSD techniques. (CO4-L4) [5M]

Ans:

1. Supervised WSD

Definition

Supervised WSD uses **labeled training data** where words are already tagged with correct senses. A machine learning model is trained to predict senses.

Working

- Input: Sense-annotated corpus
- Features: context words, POS tags, surrounding words
- Model: Naïve Bayes, Decision Trees, SVM, Neural Networks

Example

Sentence: *"I deposited money in the bank."*
Model learns from training data → predicts **bank = financial institution**

2. Knowledge-Based WSD

Definition

Knowledge-based WSD uses **external lexical resources** like dictionaries, WordNet, or ontologies without labeled training data.

Working

- Uses definitions (glosses), synonyms, semantic relations
- Algorithms: Lesk algorithm, semantic similarity

Example

Sentence: *"He sat on the bank of the river."*
Using WordNet → context words like "river" → select **bank = river side**

Comparision Table:

Feature	Supervised WSD	Knowledge-Based WSD
Data Requirement	Labeled data required	No labeled data
Approach	Machine learning	Lexical resources
Accuracy	High	Moderate
Scalability	Limited	High
Domain Adaptation	Poor	Better
Example Method	SVM, Neural Nets	Lesk Algorithm

UNIT-V

10 a) Illustrate discourse segmentation techniques. (CO3-L3) [5M]

Ans:

Definition

Discourse segmentation is the process of dividing a text into smaller meaningful units called **Elementary Discourse Units (EDUs)**. These units typically correspond to clauses or simple sentences and serve as the basic building blocks for discourse analysis.

Formally, for a text T, segmentation produces:

$$T = \{EDU_1, EDU_2, EDU_3, \ldots, EDU_n\}$$

Discourse Segmentation Techniques:

1. Rule-Based Segmentation

Description

This approach uses **linguistic rules** to identify boundaries between EDUs.

Rules Used

- **Punctuation-based**: comma (,), period (.), semicolon (;)
- **Conjunction-based**: because, although, but, and
- **Clause markers**: subordinate clauses

Example

Sentence:
"She left early because she was tired."

Segmentation:

- EDU1: She left early
- EDU2: because she was tired

2. Cue Phrase-Based Segmentation

Description

Uses **discourse markers (cue phrases)** that signal boundaries.

Common Cue Words

- Contrast: however, but
- Cause: because, therefore
- Condition: if, unless

Example

"He was ill, however he attended the meeting."

Segmentation:

- EDU1: He was ill
- EDU2: however he attended the meeting

3. Syntactic Parsing-Based Segmentation

Description

Uses **syntactic parse trees** to identify clause boundaries.

Approach

- Analyze sentence structure (NP, VP, subordinate clauses)
- Split based on **S (sentence) nodes** or clause boundaries

Example

Sentence:
"Although it was raining, she went outside."

Parse identifies subordinate clause → segmentation:

- EDU1: Although it was raining
- EDU2: she went outside

4. Statistical / Machine Learning-Based Segmentation

Description

Uses **annotated corpora** and machine learning models to predict EDU boundaries.

Mathematical Formulation

Given a sequence of words $W = \{w_1, w_2, ..., w_n\}$, predict boundary labels:

$$Y = \{y_1, y_2, ..., y_n\}, \quad y_i \in \{0, 1\}$$

where:

- $y_i = 1$ → boundary exists after word w_i

Models Used

- Hidden Markov Models (HMM)
- Conditional Random Fields (CRF)
- Support Vector Machines (SVM)
- Neural Networks (BiLSTM, Transformers)

Features

- POS tags
- Word embeddings
- Punctuation
- Context window

Example:

Sentence:

"Although it was raining, she went to college and attended the class because she had an exam."

Segmentation Output

1. Although it was raining
2. she went to college
3. and attended the class
4. because she had an exam

10 b) Apply coherence relations using Rhetorical Structure Theory (RST).

(CO3-L3) [5M]

Ans:

Rhetorical Structure Theory (RST) is a framework used to analyze the coherence of a text by identifying relationships between discourse units (EDUs). It explains how different parts of a text are connected to form a meaningful whole.

Common Coherence Relations in RST

Relation	Description	Example
Elaboration	Adds more details	She bought a car (She chose a red one)
Cause	Shows reason	He left early because he was tired
Result	Shows outcome	It rained heavily so the match was cancelled
Contrast	Shows opposition	He is rich but he is unhappy
Condition	Shows condition	If it rains, we will stay home
Explanation	Clarifies meaning	She failed because she didn't study
Sequence	Shows order	He woke up, then he brushed his teeth

Mathematical Representation:

RST relations can be represented as:

$$Relation(Nucleus, Satellite)$$

Example:

$$Cause(N, S)$$

Or:

$$Contrast(S, N)$$

Example:

Step 1: Segment the Text into EDUs

Example:
"Although it was raining, she went to college because she had an exam."

EDUs:

1. Although it was raining
2. she went to college
3. because she had an exam

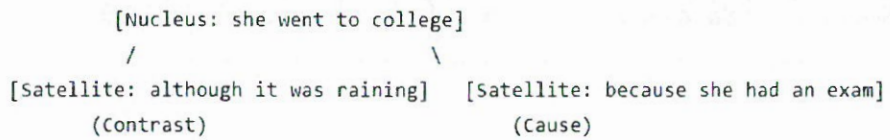
Step 2: Identify Relations

- EDU1 → EDU2 → **Contrast/Condition**
- EDU3 → EDU2 → **Cause**

Step 3: Assign Nucleus and Satellite

- EDU2 (main action) → **Nucleus**
- EDU1 (contrast condition) → **Satellite**
- EDU3 (reason) → **Satellite**

RST Representation (Tree Structure)



(OR)

11 a) Demonstrate Brown Corpus and British National Corpus in NLP research.

(CO3-L3) [5M]

Ans:

A corpus is a large, structured collection of texts used for linguistic analysis and Natural Language Processing (NLP). Corpora help in training, testing, and evaluating NLP models.

1. Brown Corpus:

Definition

The Brown Corpus is the **first electronically available general-purpose corpus** of English, developed at **Brown University (1960s)**.

Key Features

- Size: ~1 million words
- Language: American English
- Texts: 500 samples (each ~2000 words)
- Categories: 15 genres (news, fiction, religion, science, etc.)
- Annotation: Part-of-Speech (POS) tagged

Structure

$$Corpus = \{T_1, T_2, \ldots, T_{500}\}$$

Each text is tagged:

$$Word \rightarrow POS\ Tag$$

Example

Sentence:

"The quick brown fox jumps over the lazy dog"

Tagged:

- The/DT quick/JJ brown/JJ fox/NN jumps/VBZ over/IN the/DT lazy/JJ dog/NN

2. British National Corpus(BNC): The British National Corpus (BNC) is a large, balanced corpus of British English, developed in the 1990s.

Structure

$$BNC = \{W_1, W_2, \ldots, W_{100M}\}$$

Key Features

- Size: ~100 million words
- Language: British English
- Data Types:
  - 90% written text
  - 10% spoken language
- Genres: newspapers, books, academic texts, conversations
- Annotation:
  - POS tagging
  - Syntactic information

Example

Spoken text sample:

"I think it's quite interesting, isn't it?"

Tagged with context and speaker info.

Comparison: Brown Corpus vs BNC

Feature	Brown Corpus	BNC
Year	1960s	1990s
Size	1 million words	100 million words
English Type	American English	British English
Data Type	Written only	Written + Spoken
Annotation	POS tagging	POS + syntactic
Complexity	Simple	Complex
Usage	Basic NLP tasks	Advanced NLP tasks

11 b) Use of discourse segmentation framework for dialogue systems.

(CO3-L3) [5M]

Ans: A discourse segmentation framework divides a conversation into smaller meaningful units called Elementary Discourse Units (EDUs).

In dialogue systems, it helps the system understand user intent, context, and flow of conversation.

Why Discourse Segmentation is Needed in Dialogue Systems

Dialogue is not just isolated sentences—it is a **sequence of related utterances.** Segmentation helps to:

- Identify **intent boundaries**
- Understand **context shifts**
- Maintain **coherence in conversation**
- Improve response generation

Framework Components

1. Input Processing

- User input (text or speech)
- Preprocessing: tokenization, POS tagging

2. Segmentation Module

- Splits dialogue into EDUs
- Based on:
  - Punctuation
  - Pause (in speech)
  - Cue words (because, but, so)

Mathematical Representation

Given a dialogue:

$$D = \{U_1, U_2, ..., U_n\}$$

Segmentation:

$$U_i \rightarrow \{EDU_1, EDU_2, ..., EDU_k\}$$

3. Discourse Analysis

- Identify relations between segments:
  - Question–Answer
  - Cause–Effect
  - Clarification
  - Elaboration

4. Dialogue Management

- Uses segmented units to:
  - Track context
  - Maintain state
  - Select appropriate response

Example

User Input:

"Although I am busy, I need to book a ticket, so help me."

Segmentation:

1. Although I am busy
2. I need to book a ticket
3. so help me

Analysis:

- EDU1 → Background
- EDU2 → Main intent (Nucleus)
- EDU3 → Request

System Understanding:

→ Intent: Ticket booking

Types of Segmentation Used

1. Rule-Based

- Based on punctuation and conjunctions

2. Machine Learning-Based

- Uses models like CRF, LSTM

3. RST-Based

- Identifies relations like cause, contrast

**********THE END**********