Natural Language Processing: Syntax and Semantic Analysis
Part-of-Speech Tagging and Lexical Semantics

Outline:

1. Syntax Analysis: POS Tagging
2. Semantic Analysis
3. Conclusion

Why do you need Syntax Analyser?

- Check if the code is valid grammatically.
- The syntactical analyzer helps you to apply rules to the code.
- Helps you to make sure that each opening brace has a corresponding closing balance.
- Each declaration has a type and that the type must be exists.

POS tagging

POS tagging is the process of assigning a part of speech (like noun, verb, adjective) to each word in a sentence, based on its **definition** and **context**.

Example:
Input sentence:
The dog barks loudly.
POS Tags:
The → Determiner (DT)
dog → Noun (NN)
barks → Verb (VBZ)
loudly → Adverb (RB)

Penn Treebank Tag Set

Standardized tag set for English POS tagging.

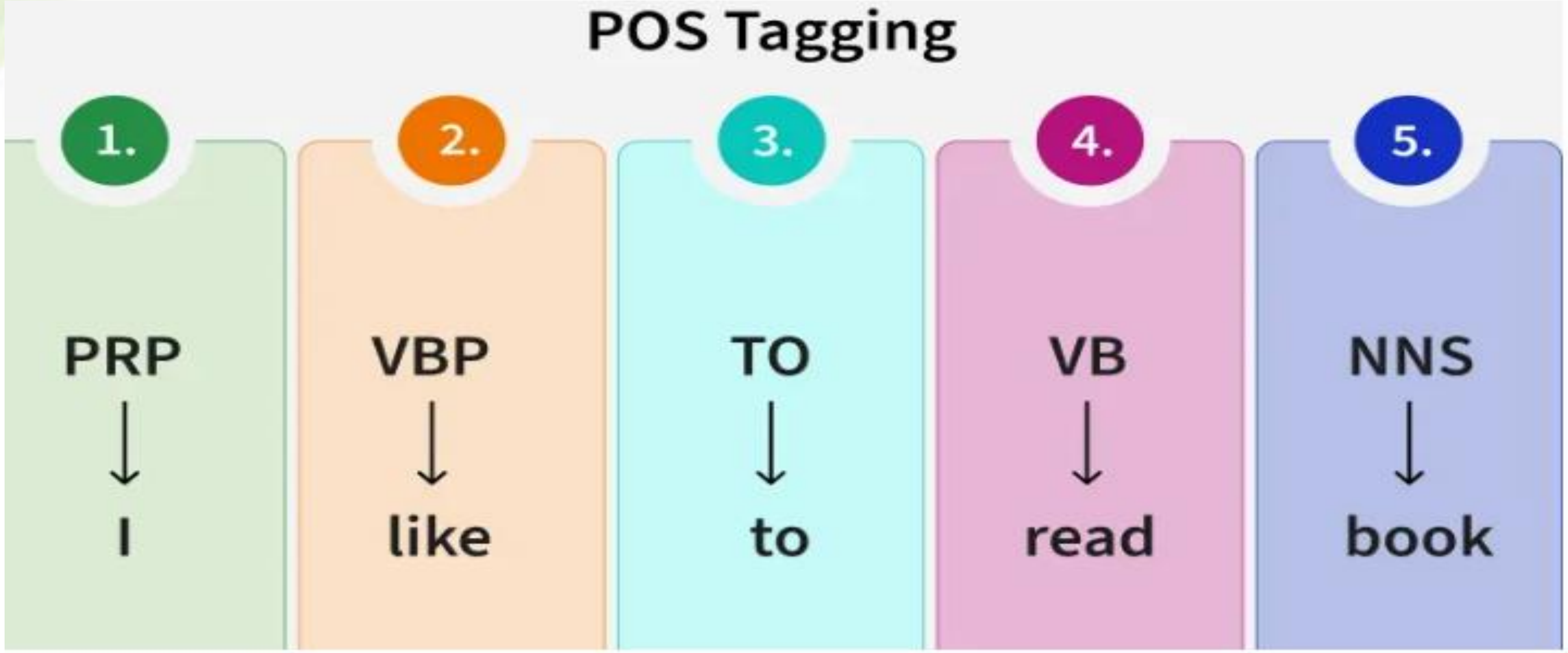
Examples: NN: Noun, singular (e.g., cat) VBZ: Verb, 3rd person singular present (e.g., runs) DT: Determiner (e.g., the)

Contains 36 main tags for parts of speech

Common POS Tags (Penn Treebank Tagset):

Tag	Description	Example
NN	Noun, singular	dog, house
NNS	Noun, plural	dogs, houses
VB	Verb, base form	run, eat
VBZ	Verb, 3rd person	runs, eats
VBD	Verb, past tense	ran, ate
JJ	Adjective	big, blue
RB	Adverb	quickly, very
DT	Determiner	the, a

POS TAGGING:



Rule-Based POS Tagging

Uses predefined grammatical rules to assign tags.

Example: If a word ends in "-ing" and follows a verb, tag as VBG (gerund).

Advantages: Interpretable, works well for regular patterns.
Disadvantages: Limited scalability, struggles with ambiguity

Stochastic POS Tagging

Uses probabilistic models to assign tags based on word and context probabilities.

Common models: Hidden Markov Models (HMM), Maximum Entropy.

Example: $P(NN|cat) \cdot P(VBZ|NN,runs)$

Advantages: Handles ambiguity, data-driven

Issues in POS Tagging

Multiple Tags for Words: Words like “run” (NN or VB).

Unknown Words: New or rare words not in training data.

Solutions: Contextual analysis for disambiguation.
Morphological clues or fallback tags for unknown words.

Context-Free Grammar (CFG)

Formal grammar for syntactic structure.

Rules: S → NP V P, NP → DT NN, etc. Used for parsing sentences into phrase structures.

Example: “The cat runs” → [S [NP The cat] [VP runs]]

Sequence Labeling: Hidden Markov Model (HMM)

Models sequence of words and tags as a Markov process.

States: POS tags; Observations: Words.
Uses Viterbi algorithm for optimal tag sequence.

Example: $P(tag_t \mid tag_{t-1}) \cdot P(word_t \mid tag_t)$

Lexical Semantics

Study of word meanings and their relationships.

Key concepts:

Homonymy: Same form, different meanings (e.g., bank: river vs. financial).

Polysemy: Related meanings (e.g., book: physical vs. content).

Synonymy: Similar meanings (e.g., big, large).

Hyponymy: Hierarchical relations (e.g., dog → animal).

Attachment for English Fragments

Assigning syntactic structure to sentence fragments

Examples: Noun phrases: “The big dog” → [NP DT JJ NN] Verb phrases: “Runs quickly” → [VP VB RB]

Prepositional phrases: “On the table” → [PP IN NP]
Challenges: Ambiguity in attachment (e.g., PP attachment).

Robust Word Sense Disambiguation (WSD)

Assigning correct meaning to a word in context.

Example: "bank" in "river bank" vs. "bank account".

Approaches:
Dictionary-based: Use lexical resources like WordNet.
Supervised: Train classifiers on annotated corpora.

Dictionary-Based WSD

Relies on lexical databases like WordNet.
Process: Identify word senses from dictionary.

Use context clues to select the appropriate sense.

Limitations: Coverage, context ambiguity.