Logic Programming (ITSE301)

Introduction to
Natural Language Processing (5)

Lexical Analysis

What is Lexical Analysis?

- Lexical analysis in NLP is the process of converting a sequence of characters into meaningful tokens by identifying the structure and components of the text.
- Lexical analysis is the first step in many NLP applications, such as text mining, sentiment analysis, and machine translation.

What is Lexical Analysis?

- Lexical analysis in NLP includes tasks such as:
  - tokenization
  - Morphological Analysis
  - Part-of-Speech (PoS) Tagging

Tokenization

- Tokenization is the initial phase of lexical analysis, where text is divided into smaller units called tokens.
- Tokens can be words, numbers, punctuations, and other symbols.
- The primary objective is to simplify the text and prepare it for more complex NLP tasks such as machine translation.

Tokenization examples

- ?- tokenize(‘Prolog is used for NLP.’, Tokens).

  Tokens = [‘Prolog’, ‘is’, ‘used’, ‘for’, ‘NLP’, ‘.’]
- ?- tokenize(‘Ali’s car is new.’, Tokens).

  Tokens = [‘Ali’, ‘’s’, ‘car’, ‘is’, ‘new’, ‘.’]

Morphological Analysis

- Morphological analysis in NLP involves analyzing the internal structure of words to determine their root form or base form.
- It helps in reducing inflected or derived words to their canonical or dictionary form.
- The two main techniques used for morphological analysis are lemmatizing and stemming.

Morphological Analysis: Lemmatizing

- Lemmatizing is the process of identifying the base or dictionary form of a word, known as its lemma.
- Lemmatizing applies linguistic rules to transform the word to its dictionary or base form.
- For example, the lemma of the words "running," "runs," and "ran" is "run."

Morphological Analysis: Stemming

- Stemming is the process of removing prefixes and affixes from words to obtain their stem. The stem might not be a valid word.
- Stemming relies on pattern matching to strip common prefixes and suffixes from words, without considering the word's meaning or context.
- For example, the stem of the word “flies“ is “fli" and the suffix is “es"

Part-of-Speech (POS) tagging

- POS tagging is a technique used to assign a grammatical category for each token.
- This step is critical for understanding the syntactic structure of sentences.
- POS tagging involves assigning labels to words as noun, verb, adj, etc.
- The accuracy of POS tagging directly affects the accuracy of subsequent NLP tasks such as Named Entity Recognition.

Part-of-Speech (POS) tagging: Examples

- ?- pos(‘Prolog is cool’, POS).

  POS = [‘Prolog/PN’, ‘is/Verb’, ‘cool/Adj’]

Reading from the keyborad

- Prolog has a built in predicate called readln(S).
- It allows you to read a line and put it in a list.
- We can use it to read a sentence:
  - **run :-**

```
readln(S),
tokenize(S, Tokens),
write("The tokens: "),
write(Tokens).
```

Try the tokenizer

- Now load the tokenizer and run the code.

```
| ?-run.
```