| Experiment No. 5 |
| --- |
| Implement POS Tagging for any given text. |
| Date of Performance: |
| Date of Submission: |

**Exp. No.:** 5

**Title:** Implement POS Tagging for any given text.

**Theory:**

The primary target of Part-of-Speech (POS) tagging is to identify the grammatical group of a given word. Whether it is a NOUN, PRONOUN, ADJECTIVE, VERB, ADVERBS, etc. based on the context. POS Tagging looks for relationships within the sentence and assigns a corresponding tag to the word.

**POS Tagging** (Parts of Speech Tagging) is a process to mark up the words in text format for a particular part of a speech based on its definition and context. It is responsible for text reading in a language and assigning some specific token (Parts of Speech) to each word. It is also called grammatical tagging.

**Steps Involved in the POS tagging example:**

- Tokenize text (word_tokenize)
- apply pos_tag to above step that is nltk.pos_tag(tokenize_text)

**Example:**

**Sentence:** Albert Einstein was born in Ulm, Germany in 1879.

After POS Tagging:

[('Albert', 'NNP'),

 ('Einstein', 'NNP'),

 ('was', 'VBD'),

 ('born', 'VBN'),

 ('in', 'IN'),

 ('Ulm', 'NNP'),

(',', ','),

('Germany', 'NNP'),

('in', 'IN'),

('1879', 'CD'),

('.', '.')]

**Code:**

```python
# Importing the NLTK library
import nltk
from nltk.tokenize import word_tokenize
from nltk import pos_tag

# Download the necessary resources
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')

# Sample text
text = "NLTK is a powerful library for natural language processing."

# Tokenizing the text into words
words = word_tokenize(text)

# Performing PoS tagging
pos_tags = pos_tag(words)

# Displaying the PoS tagged result in separate lines
print("Original Text:")
print(text)

print("\nPoS Tagging Result:")
for word, pos in pos_tags:
    print(f"{word}: {pos}")
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Unzipping taggers/averaged_perceptron_tagger.zip.
Original Text:
NLTK is a powerful library for natural language processing.

PoS Tagging Result:
NLTK: NNP
is: VBZ
a: DT
powerful: JJ
library: NN
for: IN
natural: JJ
language: NN
processing: NN
.: .
```

```python
#importing libraries
import spacy

# Load the English language model
nlp = spacy.load("en_core_web_sm")

# Sample text
text = "SpaCy is a popular natural language processing library."

# Process the text with SpaCy
doc = nlp(text)

# Display the PoS tagged result
print("Original Text: ", text)
print("PoS Tagging Result:")
for token in doc:
  print(f"{token.text}: {token.pos_}")
```

```
Original Text:  SpaCy is a popular natural language processing library.
PoS Tagging Result:
SpaCy: PROPN
is: AUX
a: DET
popular: ADJ
natural: ADJ
language: NOUN
processing: NOUN
library: NOUN
.: PUNCT
```

**Conclusion:**

In conclusion, Part-of-Speech (POS) tagging is a fundamental process in Natural Language Processing (NLP) that identifies the grammatical category of each word in a given text, such as noun, verb, adjective, or adverb, based on its context. By analyzing relationships within a sentence, POS tagging assigns the appropriate grammatical tag to each word, enabling better understanding of the text structure. This process involves tokenizing the text and applying a POS tagging algorithm, such as the `nltk.pos_tag` function, which aids in various NLP tasks like syntactic parsing, information extraction, and text analysis.