Experiment No.6
Perform POS tagging on the given English and Indian Language Text
Date of Performance:
Date of Submission:

Aim: Perform POS tagging on the given English and Indian Language Text

Objective: To study POS Tagging and tag the part of speech for given input in english and an Indian Language.

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)

```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
```

```
[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.
True
```

```
from nltk.chunk import RegexpParser
from nltk.tokenize import word_tokenize
```

```
sentence = "Education is the transmission of knowledge, skills, and character traits. "
```

Tokenization

```
tokens = word_tokenize(sentence)
```

```
tokens
```

```
['Education',
 'is',
 'the',
 'transmission',
 'of',
 'knowledge',
 ',',
 'skills',
 ',',
 'and',
 'character',
 'traits',
 '.']
```

POS tagging

```
pos_tags = nltk.pos_tag(tokens)
```

```
pos_tags
```

```
[('Education', 'NN'),
 ('is', 'VBZ'),
 ('the', 'DT'),
 ('transmission', 'NN'),
 ('of', 'IN'),
 ('knowledge', 'NN'),
 (',', ','),
 ('skills', 'NNS'),
 (',', ','),
 ('and', 'CC'),
 ('character', 'NN'),
 ('traits', 'NNS'),
 ('.', '.')]
```

Chunking patterns

```
chunk_patterns = r"""
    NP: {<DT>?<JJ>*<NN>}  # Chunk noun phrases
    VP: {<VB.*><NP|PP>}  # Chunk verb phrases
"""
```

```
chunk_patterns
```

```
'\n    NP: {<DT>?<JJ>*<NN>}  # Chunk noun phrases\n    VP: {<VB.*><NP|PP>}  # Chunk
verb phrases\n'
```

Create a chunk parser

```
chunk_parser = RegexpParser(chunk_patterns)
```

```
chunk_parser
```

```
    <chunk.RegexpParser with 2 stages>
```

▾ Perform chunking

```
result = chunk_parser.parse(pos_tags)
```

```
print(result)
```

```
    (S
      (NP Education/NN)
      (VP is/VBZ (NP the/DT transmission/NN))
      of/IN
      (NP knowledge/NN)
      ,/,
      skills/NNS
      ,/,
      and/CC
      (NP character/NN)
      traits/NNS
      ./.)
```

Conclusion:

POS tagging (Part-of-Speech tagging) involves labeling words in a text with their grammatical categories (e.g., noun, verb, adjective). For English text, widely available libraries like NLTK or spaCy provide accurate tagging due to well-defined grammar. Indian languages pose greater challenges due to their diversity, script variations, and limited resources. Building accurate POS taggers for Indian languages often requires language-specific models and extensive linguistic knowledge.