SATHYABAMA INSTITUTE OF SCIENCE & TECHNOLOGY

SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCSA 2604 NATURAL LANGUAGE PROCESSING LAB

LAB 6: PARTS OF SPEECH TAGGING

AIM: To perform Parts of Speech (POS) tagging program using NLTK

PROCEDURE:

Library Installation and Import: Ensures the NLTK library is available for use and imports the necessary modules for text processing.

Download NLTK Resources: Downloads essential resources (punkt for tokenization, averaged_perceptron_tagger for POS tagging) required by NLTK.

Sample Text: Defines a piece of text to demonstrate POS tagging.

Tokenization: Divides the text into individual words or tokens, making it suitable for further analysis.

POS Tagging: Assigns each word in the text its respective grammatical category or POS tag using NLTK's POS tagging functionality.

Display POS Tags: Prints or displays the words along with their associated POS tags obtained from the tagging process.

The following algorithm outlines the steps involved in performing Parts of Speech (POS) tagging using NLTK in Python. It demonstrates how to tokenize a text and assign grammatical categories to individual words, providing insight into the linguistic structure of the text.

ALGORITHM:

1. Library Installation and Import: Install NLTK library if not already installed. Import the necessary NLTK library for text processing and POS tagging.
2. Download NLTK Resources: Download NLTK resources required for tokenization and POS tagging (punkt for tokenization, averaged_perceptron_tagger for POS tagging).
3. Sample Text: Define a sample text for POS tagging.
4. Tokenization: Break down the provided text into individual words (tokens) using NLTK's word_tokenize() method.

5. POS Tagging: Perform POS tagging on the tokens obtained from the text using NLTK's pos_tag() method.
   Assign POS tags to each word in the text based on its grammatical category (noun, verb, adjective, etc.).
6. Display POS Tags: Print or display the words along with their respective POS tags generated by the POS tagging process.

PROGRAM:

```
# Install NLTK (if not already installed)

!pip install nltk


# Import necessary libraries

import nltk

nltk.download('punkt')

nltk.download('averaged_perceptron_tagger')


# Sample text for POS tagging

text = "Parts of speech tagging helps to understand the function of each word in a sentence."


# Tokenize the text into words

tokens = nltk.word_tokenize(text)


# Perform POS tagging

pos_tags = nltk.pos_tag(tokens)


# Display the POS tags

print("POS tags:", pos_tags)
```

OUTPUT:

```
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)

Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.7)
```

Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.3.2)

Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2023.12.25)

Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.2)

[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.

POS tags: [('Parts', 'NNS'), ('of', 'IN'), ('speech', 'NN'), ('tagging', 'VBG'), ('helps', 'NNS'), ('to', 'TO'), ('understand', 'VB'), ('the', 'DT'), ('function', 'NN'), ('of', 'IN'), ('each', 'DT'), ('word', 'NN'), ('in', 'IN'), ('a', 'DT'), ('sentence', 'NN'), ('.', '.')]