SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC ACCREDITED with "A" GRADE (CGPA : 3.18)

DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE CODE: DJ19ITL504 **DATE: 04/12/23**

COURSE NAME: Artificial Intelligence Laboratory **CLASS: I2-1**

EXPERIMENT NO.08

CO/LO: CO2

AIM / OBJECTIVE: To perform Text Processing on a particular dataset using NLTK

DESCRIPTION OF EXPERIMENT:

Text Processing using NLTK (Natural Language Toolkit):

NLTK is a Python library widely used for Natural Language Processing (NLP) tasks, providing a comprehensive set of tools and resources for working with human language data. Here's a short overview of common text processing tasks using NLTK:

1. Installation:
   - Install NLTK using pip install nltk.
2. Tokenization:
   - Break text into words or sentences using word_tokenize and sent_tokenize.
3. Stopword Removal:
   - Remove common words (stopwords) that do not contribute much to the meaning of the text.
4. Stemming:
   - Reduce words to their base or root form using stemming algorithms like Porter or Lancaster.
5. Part-of-Speech Tagging:
   - Identify the grammatical parts of words in a sentence using pos_tag.
6. Named Entity Recognition (NER):
   - Identify and classify entities (e.g., names, locations) in text using tools like ne_chunk.
7. Frequency Distribution:
   - Analyze word frequency in a text using FreqDist to gain insights into key terms.

8. Concordance:

    - Find occurrences of a word within a specific context using concordance.

9. Similarity Measures:

    - Calculate similarity between words or documents using various metrics.

10. Corpus and Resources:

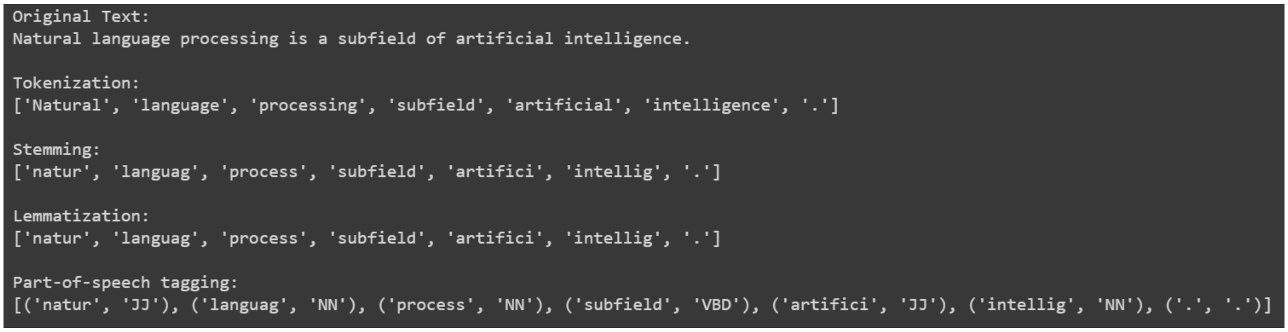
    - Access a wide range of corpora and lexical resources for research and analysis.

CODE:

```
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer, WordNetLemmatizer
from nltk import pos_tag

nltk.download('punkt')
nltk.download('stopwords')
nltk.download('averaged_perceptron_tagger')
nltk.download('wordnet')

def preprocess_text(text):
    # Tokenization
    tokens = word_tokenize(text)

    # Remove stopwords
    stop_words = set(stopwords.words('english'))
    filtered_tokens = [word for word in tokens if word.lower() not in
stop_words]

    # Stemming
    stemmer = PorterStemmer()
    stemmed_tokens = [stemmer.stem(word) for word in filtered_tokens]

    # Lemmatization
    lemmatizer = WordNetLemmatizer()
    lemmatized_tokens = [lemmatizer.lemmatize(word) for word in
stemmed_tokens]

    # Part-of-speech tagging
    pos_tags = pos_tag(lemmatized_tokens)

    return filtered_tokens, stemmed_tokens, lemmatized_tokens, pos_tags

# Example text
```

```
text = "Natural language processing is a subfield of artificial
intelligence."

# Preprocess the text
filtered_tokens, stemmed_tokens, lemmatized_tokens, pos_tags =
preprocess_text(text)

# Display the results
print("Original Text:")
print(text)
print("\nTokenization:")
print(filtered_tokens)
print("\nStemming:")
print(stemmed_tokens)
print("\nLemmatization:")
print(lemmatized_tokens)
print("\nPart-of-speech tagging:")
print(pos_tags)
```

OUTPUT:



```
Original Text:
Natural language processing is a subfield of artificial intelligence.

Tokenization:
['Natural', 'language', 'processing', 'subfield', 'artificial', 'intelligence', '.']

Stemming:
['natur', 'languag', 'process', 'subfield', 'artifici', 'intellig', '.']

Lemmatization:
['natur', 'languag', 'process', 'subfield', 'artifici', 'intellig', '.']

Part-of-speech tagging:
[('natur', 'JJ'), ('languag', 'NN'), ('process', 'NN'), ('subfield', 'VBD'), ('artifici', 'JJ'), ('intellig', 'NN'), ('.', '.')]
```

CONCLUSION:

In this experiment we learnt about text preprocessing in artificial intelligence using NLTK and carried out Tokenization , Stemming , Lemmatization and POS tagging.

REFERENCES:

[1] Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach”, 2nd Edition, Pearson Education, 2010