S. B. JAIN INSTITUTE OF TECHNOLOGY,
MANAGEMENT & RESEARCH, NAGPUR.

Practical No. 4

Aim: Understanding and applying NLTK functions by importing corpus and stop words.

Name of Student: ______________

Roll No.: ______________

Semester/Year: **IV/VII**

Academic Session: 2025-2026

Date of Performance: __________

Date of Submission: __________

AIM: Understanding and applying NLTK functions by importing corpus and stop words

OBJECTIVE/EXPECTED LEARNING OUTCOME:

- Understanding natural language toolkit
- Various functions in NLTK
- Analyzing stopwords and corpus

HARDWARE AND SOFTWARE REQUIRMENTS:

Hardware Requirement:

Software Requirement:

THEORY:

NLTK is a toolkit build for working with NLP in Python. It provides us various text processing libraries with a lot of test datasets. A variety of tasks can be performed using NLTK such as tokenizing, parse tree visualization, etc

Use the pip install method to install NLTK in your system: *pip install nltk*

In computing, **stop words** are words that are filtered out before or after the natural language data (text) are processed. While “stop words” typically refers to the most common words in a language, all-natural language processing tools don’t use a single universal list of stop words.

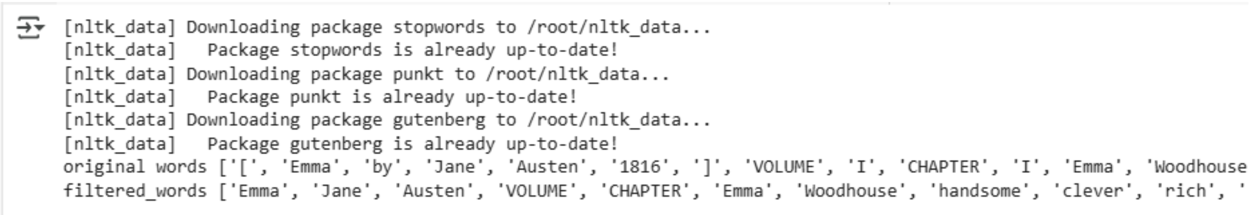
Stopwords are the **words** in any language which does not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence. For some search engines, these are some of the most common, short function words, such as the, is, at, which, and on. In this case, stop words can cause problems when searching for phrases that include them, particularly in names such as “The Who” or “Take That”.

A corpus refers to a large and structured collection of text or spoken language data that is used for linguistic analysis, machine learning, and various NLP tasks. Corpora (plural of corpus) are essential resources for developing and evaluating NLP models and algorithms. They are used to study language patterns, extract linguistic information, train machine learning models, and perform a wide range of language-related research.

CODE:

```
import nltk
nltk.download('stopwords')
nltk.download('punkt')
nltk.download('gutenberg')  # Sample corpus
from nltk.corpus import stopwords, gutenberg
from nltk.tokenize import word_tokenize
sample_text = gutenberg.raw('austen-emma.txt')  # Jane Austen's "Emma"
words = word_tokenize(sample_text)
stop_words = set(stopwords.words('english'))
filtered_words = [word for word in words if word.lower() not in stop_words and word.isalpha()]
print("original words", words[:20])
print("filtered_words",filtered_words[:20])
```

OUTPUT (SCREENSHOT):



```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package gutenberg to /root/nltk_data...
[nltk_data]   Package gutenberg is already up-to-date!
original words ['[', 'Emma', 'by', 'Jane', 'Austen', '1816', ']', 'VOLUME', 'I', 'CHAPTER', 'I', 'Emma', 'Woodhouse
filtered_words ['Emma', 'Jane', 'Austen', 'VOLUME', 'CHAPTER', 'Emma', 'Woodhouse', 'handsome', 'clever', 'rich', '
```

[nltk_data] Downloading package stopwords to /root/nltk_data...

[nltk_data]   Package stopwords is already up-to-date!

[nltk_data] Downloading package punkt to /root/nltk_data...

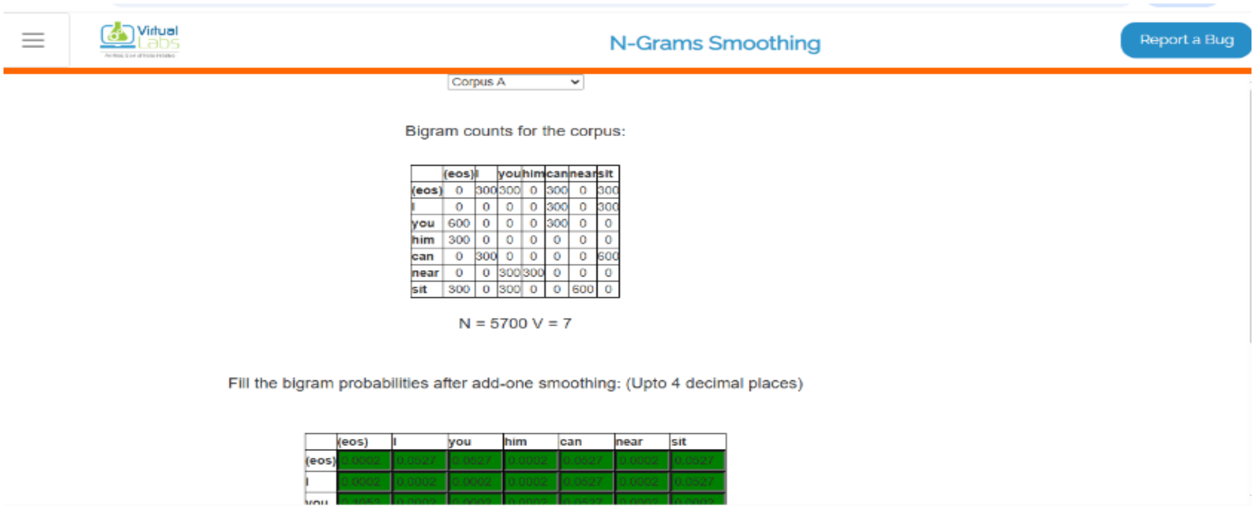
[nltk_data]   Package punkt is already up-to-date!

[nltk_data] Downloading package gutenberg to /root/nltk_data...

[nltk_data]   Package gutenberg is already up-to-date!

original words ['[', 'Emma', 'by', 'Jane', 'Austen', '1816', ']', 'VOLUME', 'I', 'CHAPTER', 'I', 'Emma', 'Woodhouse', ',', 'handsome', ',', 'clever', ',', 'and', 'rich']

filtered_words ['Emma', 'Jane', 'Austen', 'VOLUME', 'CHAPTER', 'Emma', 'Woodhouse', 'handsome', 'clever', 'rich', 'comfortable', 'home', 'happy', 'disposition', 'seemed', 'unite', 'best', 'blessings', 'existence', 'lived'



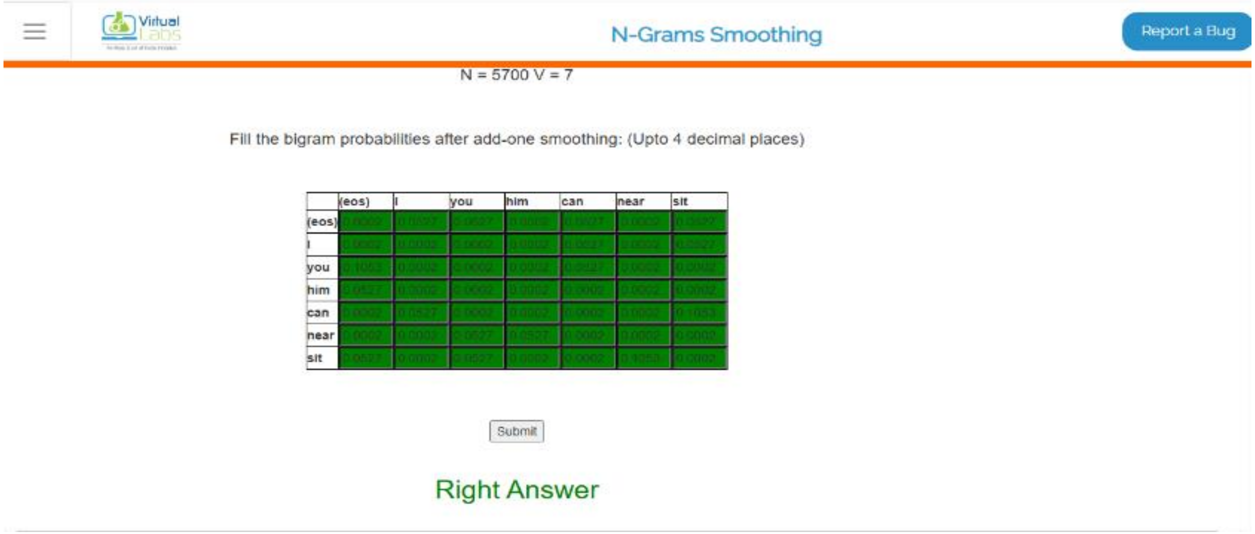
Bigram counts for the corpus:

	(eos)	I	you	him	can	near	sit
(eos)	0	300	300	0	300	0	300
I	0	0	0	0	300	0	300
you	600	0	0	0	300	0	0
him	300	0	0	0	0	0	0
can	0	300	0	0	0	0	600
near	0	0	300	300	0	0	0
sit	300	0	300	0	0	600	0

N = 5700 V = 7

Fill the bigram probabilities after add-one smoothing: (Upto 4 decimal places)

	(eos)	I	you	him	can	near	sit
(eos)	0.0002	0.0527	0.0527	0.0002	0.0527	0.0002	0.0527
I	0.0002	0.0002	0.0002	0.0002	0.0527	0.0002	0.0527
you	0.1053	0.0002	0.0002	0.0002	0.0527	0.0002	0.0002



N = 5700 V = 7

Fill the bigram probabilities after add-one smoothing: (Upto 4 decimal places)

	(eos)	I	you	him	can	near	sit
(eos)	0.0002	0.0527	0.0527	0.0002	0.0527	0.0002	0.0527
I	0.0002	0.0002	0.0002	0.0002	0.0527	0.0002	0.0527
you	0.1053	0.0002	0.0002	0.0002	0.0527	0.0002	0.0002
him	0.0527	0.0002	0.0002	0.0002	0.0002	0.0002	0.0002
can	0.0002	0.0527	0.0002	0.0002	0.0002	0.0002	0.1053
near	0.0002	0.0002	0.0527	0.0527	0.0002	0.0002	0.0002
sit	0.0527	0.0002	0.0527	0.0002	0.0002	0.1053	0.0002

Right Answer

CONCLUSION:

DISCUSSION AND VIVA VOCE:

- What is NLTK, why it is used?
- What are stopwords, how they are removed?
- What is corpus, why it is required?

REFERENCE:

- www.w3schools.com
- www.tutorialsmade.com
- www.towardsdatascience.com