**PRACTICAL NO 10**

**AIM :** Advanced Topics in Information Retrieval

* Implement a text summarization algorithm (e.g., extractive or abstractive).
* Build a question-answering system using techniques such as information extraction

**INPUT:**

import nltk

nltk.download('punkt')

nltk.download('stopwords')

from nltk.tokenize import sent\_tokenize, word\_tokenize

from nltk.corpus import stopwords

from collections import Counter

def generate\_summary(text, num\_sentences=2):

sentences = sent\_tokenize(text)

words = [word.lower() for word in word\_tokenize(text) if word.isalnum()]

words = [word for word in words if word not in stopwords.words("english")]

word\_freq = Counter(words)

sentence\_scores = {sent: sum(word\_freq[word] for word in word\_tokenize(sent.lower()) if word in word\_freq) for sent in sentences}

summary = ' '.join(sorted(sentence\_scores, key=sentence\_scores.get, reverse=True)[:num\_sentences])

return summary

# Example usage

text = """

Natural language processing (NLP) is a field of computer science, artificial intelligence, and computational linguistics concerned with the interactions between computers and human (natural) languages.

As such, NLP is related to the area of human–computer interaction.

Many challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input, and others involve natural language generation.

"""

print(generate\_summary(text))

**OUTPUT:**

