Modules Used

1. **requests**
2. **BeautifulSoup (from bs4)**
3. **nltk (Natural Language Toolkit)**

1. `requests`

The `requests` module is a simple and elegant HTTP library for Python. It is used for making HTTP requests to communicate with web servers and APIs. In your script, it is used to:

- Make requests to the Google Custom Search API to retrieve search results.
- Fetch the content of webpages to extract text for comparison.

```
import requests

response = requests.get(url)  # Making an HTTP GET request
```

2. `BeautifulSoup (from bs4)`

`BeautifulSoup` is a library used for parsing HTML and XML documents. It creates parse trees from page source codes that can be used to extract data easily. In your script, it is used to:

- Parse the HTML content of webpages.
- Extract text from `<p>` tags to gather the main content of the webpage.

```
from bs4 import BeautifulSoup

soup = BeautifulSoup(response.text, 'html.parser')
paragraphs = soup.find_all('p')  # Finding all paragraph tags
```

3. `nltk (Natural Language Toolkit)`

`nltk` is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources. In your script, various parts of `nltk` are used for:

- Tokenizing text into sentences and words.
- Removing stopwords (common words that are usually filtered out in natural language processing).
- Stemming words to their root form.

- Part-of-speech tagging to identify nouns in the text.
- Using WordNet to find synonyms and related words.

Here’s a breakdown of the specific parts of `nltk` used in your script:

`sent_tokenize` and `word_tokenize`: Tokenize text into sentences and words.

```
from nltk.tokenize import sent_tokenize, word_tokenize

sentences = sent_tokenize(text)  # Splitting text into sentences
words = word_tokenize(sentence)  # Splitting sentences into words
```

-

`stopwords`: Provides a list of common stopwords to be filtered out.

```
from nltk.corpus import stopwords

stop_words = set(stopwords.words('english'))
```

-

`PorterStemmer`: Stems words to their base or root form.

```
from nltk.stem import PorterStemmer

porter = PorterStemmer()
words = [porter.stem(word) for word in words]  # Stemming words
```

-

`pos_tag`: Tags words with their parts of speech.

```
from nltk import pos_tag

pos_tag(words)  # Tagging words with their parts of speech
```

-

`wordnet`: A lexical database for the English language, used for finding synonyms.

```
from nltk.corpus import wordnet

synsets = wordnet.synsets(noun)
```

Summary of Modules' Roles

- `requests`: Handles HTTP requests to interact with web APIs and fetch webpage content.
- `BeautifulSoup`: Parses HTML content to extract text from specific HTML tags.
- `nltk`: Processes and analyzes text data, including tokenization, stopword removal, stemming, part-of-speech tagging, and synonym finding.

These modules work together to:

1. Retrieve and parse web content.
2. Preprocess the text to focus on meaningful words and concepts.
3. Compare the input text with online content to detect potential plagiarism based on textual similarity.