one

March 28, 2025

[2]:

```
!pip install nltk
```

```
Requirement already satisfied: nltk in /usr/local/lib/python3.11/dist-packages
(3.9.1)
Requirement already satisfied: click in /usr/local/lib/python3.11/dist-packages
(from nltk) (8.1.8)
Requirement already satisfied: joblib in /usr/local/lib/python3.11/dist-packages
(from nltk) (1.4.2)
Requirement already satisfied: regex>=2021.8.3 in
/usr/local/lib/python3.11/dist-packages (from nltk) (2024.11.6)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages
(from nltk) (4.67.1)
```

[3]:

```
#  Import necessary libraries from NLTK and others
import nltk
import string
```

[9]:

```
# Download required NLTK data (if not already downloaded)
nltk.download('punkt')
nltk.download('punkt_tab')
nltk.download('wordnet')
nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data…
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data…
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to /root/nltk_data…
[nltk_data]   Package omw-1.4 is already up-to-date!
```

[9]:

```
True
```

[5]:

```
# Tokenizers from NLTK
from nltk.tokenize import WhitespaceTokenizer, word_tokenize,␣
 ↪TreebankWordTokenizer, TweetTokenizer, MWETokenizer
```

```
# Stemmers
from nltk.stem import PorterStemmer, SnowballStemmer

# Lemmatizer
from nltk.stem import WordNetLemmatizer
```

[6]:

```
#  Sample text for processing
text = "Hello, world! This is an example sentence for tokenization. Let's see␣
 ↪how it performs?  "
```

[7]:

```
#  Whitespace Tokenization
whitespace_tok = WhitespaceTokenizer().tokenize(text)
print("Whitespace Tokenization:")
print(whitespace_tok, "\n")
```

```
Whitespace Tokenization:
['Hello,', 'world!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization.', "Let's", 'see', 'how', 'it', 'performs?', ' ']
```

[10]:

```
#  Punctuation-based Tokenization using word_tokenize (splits punctuation)
punctuation_tok = word_tokenize(text)
print("Punctuation-based Tokenization (word_tokenize):")
print(punctuation_tok, "\n")
```

```
Punctuation-based Tokenization (word_tokenize):
['Hello', ',', 'world', '!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization', '.', 'Let', "'s", 'see', 'how', 'it', 'performs', '?', ' ']
```

[12]:

```
#  Treebank Tokenizer
treebank_tok = TreebankWordTokenizer().tokenize(text)
print("Treebank Tokenization:")
print(treebank_tok, "\n")
```

```
Treebank Tokenization:
['Hello', ',', 'world', '!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization.', 'Let', "'s", 'see', 'how', 'it', 'performs', '?', ' ']
```

[13]:

```
#  Tweet Tokenizer (handles emojis, hashtags, etc.)
tweet_tok = TweetTokenizer().tokenize(text)
print("Tweet Tokenization:")
print(tweet_tok, "\n")
```

```
Tweet Tokenization:
['Hello', ',', 'world', '!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization', '.', "Let's", 'see', 'how', 'it', 'performs', '?', ' ']
```

[14]:

```python
#  Multi-Word Expression (MWE) Tokenizer
# For demonstration, we define an MWE for "New York" (if it existed in text)
mwe_tok = MWETokenizer([("New", "York")])
# Tokenize first using word_tokenize then apply MWETokenizer
mwe_tokens = mwe_tok.tokenize(word_tokenize(text))
print("MWE Tokenization (for phrases like 'New York'):")
print(mwe_tokens, "\n")
```

```
MWE Tokenization (for phrases like 'New York'):
['Hello', ',', 'world', '!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization', '.', 'Let', "'s", 'see', 'how', 'it', 'performs', '?', ' ']
```

[15]:

```python
#  Using Porter Stemmer
porter = PorterStemmer()
porter_stems = [porter.stem(word) for word in punctuation_tok]
print("Porter Stemmer:")
print(porter_stems, "\n")
```

```
Porter Stemmer:
['hello', ',', 'world', '!', 'thi', 'is', 'an', 'exampl', 'sentenc', 'for',
'token', '.', 'let', "'s", 'see', 'how', 'it', 'perform', '?', ' ']
```

[16]:

```python
#  Using Snowball Stemmer (for English)
snowball = SnowballStemmer("english")
snowball_stems = [snowball.stem(word) for word in punctuation_tok]
print("Snowball Stemmer:")
print(snowball_stems, "\n")
```

```
Snowball Stemmer:
['hello', ',', 'world', '!', 'this', 'is', 'an', 'exampl', 'sentenc', 'for',
'token', '.', 'let', "'s", 'see', 'how', 'it', 'perform', '?', ' ']
```

[11]:

```python
#  Using WordNet Lemmatizer
lemmatizer = WordNetLemmatizer()
# For lemmatization, it is usually more effective if you provide POS tags.
# For simplicity, we're using the default which assumes nouns.
lemmatized_tokens = [lemmatizer.lemmatize(word) for word in punctuation_tok]
print("Lemmatization (default as nouns):")
print(lemmatized_tokens, "\n")
```

```
Lemmatization (default as nouns):
['Hello', ',', 'world', '!', 'This', 'is', 'an', 'example', 'sentence', 'for',
'tokenization', '.', 'Let', "'s", 'see', 'how', 'it', 'performs', '?', ' ']
```