

```
import nltk
nltk.download('punkt', download_dir='./nltk_data')
```

```
↗ [nltk_data] Downloading package punkt to ./nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.
True
```

```
#Give Input as ant test
```

```
text = "It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife."
text = text.lower()
print(text)
```

```
↗ it is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife.
```

```
text = "It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife."
text = text.lower()
print(text)
```

```
↗ it is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife.
```

```
import string
print(string.punctuation)
```

```
↗ !"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~
```

```
text_p = "".join([char for char in text if char not in string.punctuation]); print(text_p)
```

```
↗ it is a truth universally acknowledged that a single man in possession of a good fortune must be in want of a wife
```

```
import nltk
nltk.download('punkt')
from nltk import word_tokenize
words = word_tokenize(text_p)
words1=sent_tokenize(text_p)
print(words)
print(words1)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.

-----
LookupError                                Traceback (most recent call last)
<ipython-input-6-8a0fa413e99f> in <cell line: 0>()
      2 nltk.download('punkt')
      3 from nltk import word_tokenize
----> 4 words = word_tokenize(text_p)
      5 words1=sent_tokenize(text_p)
      6 print(words)
```

↕ 5 frames

```
/usr/local/lib/python3.11/dist-packages/nltk/data.py in find(resource_name, paths)
    577     sep = "*" * 70
    578     resource_not_found = f"\n{sep}\n{msg}\n{sep}\n"
--> 579     raise LookupError(resource_not_found)
    580
    581
```

LookupError:

Resource **punkt_tab** not found.
Please use the NLTK Downloader to obtain the resource:

```
>>> import nltk
>>> nltk.download('punkt_tab')
```

For more information see: <https://www.nltk.org/data.html>

Attempted to load **tokenizers/punkt_tab/english/**

Searched in:

- '/root/nltk_data'
- '/usr/nltk_data'
- '/usr/share/nltk_data'
- '/usr/lib/nltk_data'
- '/usr/share/nltk_data'
- '/usr/local/share/nltk_data'
- '/usr/lib/nltk_data'
- '/usr/local/lib/nltk_data'

Next steps: [Explain error](#)

```
import nltk
from nltk.tokenize import word_tokenize, sent_tokenize
from nltk.corpus import stopwords
import string

# Download necessary NLTK data files
nltk.download('punkt')
nltk.download('stopwords')

# Sample text
text = "It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife."

# Remove punctuation
text_p = "".join([char for char in text if char not in string.punctuation])

# Tokenize
words = word_tokenize(text_p)
sentences = sent_tokenize(text_p)

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

# Print results
print("Words:", words)
print("Filtered Words:", filtered_words)
```



```

LookupError                                Traceback (most recent call last)
Cell In[7], line 17
    14 text_p = "".join([char for char in text if char not in string.punctuation])
    16 # Tokenize
--> 17 words = word_tokenize(text_p)
    18 sentences = sent_tokenize(text_p)
    20 # Remove stopwords

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\tokenize\__init__.py:142, in
word_tokenize(text, language, preserve_line)
    127 def word_tokenize(text, language="english", preserve_line=False):
    128     """
    129     Return a tokenized copy of *text*,
    130     using NLTK's recommended word tokenizer
    (...)
    140     :type preserve_line: bool
    141     """
--> 142     sentences = [text] if preserve_line else sent_tokenize(text, language)
    143     return [
    144         token for sent in sentences for token in _treebank_word_tokenizer.tokenize(sent)
    145     ]

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\tokenize\__init__.py:119, in
sent_tokenize(text, language)
    109 def sent_tokenize(text, language="english"):
    110     """
    111     Return a sentence-tokenized copy of *text*,
    112     using NLTK's recommended sentence tokenizer
    (...)
    117     :param language: the model name in the Punkt corpus
    118     """
--> 119     tokenizer = _get_punkt_tokenizer(language)
    120     return tokenizer.tokenize(text)

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\tokenize\__init__.py:105, in
_get_punkt_tokenizer(language)
    96 @functools.lru_cache
    97 def _get_punkt_tokenizer(language="english"):
    98     """
    99     A constructor for the PunktTokenizer that utilizes
    100     a lru cache for performance.
    (...)
    103     :type language: str
    104     """
--> 105     return PunktTokenizer(language)

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\tokenize\punkt.py:1744, in
PunktTokenizer.__init__(self, lang)
    1742 def __init__(self, lang="english"):
    1743     PunktSentenceTokenizer.__init__(self)
--> 1744     self.load_lang(lang)

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\tokenize\punkt.py:1749, in
PunktTokenizer.load_lang(self, lang)
    1746 def load_lang(self, lang="english"):
    1747     from nltk.data import find
--> 1749     lang_dir = find(f"tokenizers/punkt_tab/{lang}/")
    1750     self._params = load_punkt_params(lang_dir)
    1751     self._lang = lang

File c:\Users\Sudarshan Ahire\AppData\Local\Programs\Python\Python313\Lib\site-packages\nltk\data.py:579, in find(resource_name,
paths)
    577 sep = "*" * 70
    578 resource_not_found = f"\n{sep}\n{msg}\n{sep}\n"
--> 579 raise LookupError(resource_not_found)

LookupError:
*****
Resource punkt_tab not found.
Please use the NLTK Downloader to obtain the resource:

>>> import nltk
>>> nltk.download('punkt_tab')

For more information see: https://www.nltk.org/data.html

Attempted to load tokenizers/punkt_tab/english/

```

Searched in:

- 'C:\\Users\\Sudarshan Ahire\\nltk_data'
- 'c:\\Users\\Sudarshan Ahire\\AppData\\Local\\Programs\\Python\\Python313\\nltk_data'
- 'c:\\Users\\Sudarshan Ahire\\AppData\\Local\\Programs\\Python\\Python313\\share\\nltk_data'
- 'c:\\Users\\Sudarshan Ahire\\AppData\\Local\\Programs\\Python\\Python313\\lib\\nltk_data'
- 'C:\\Users\\Sudarshan Ahire\\AppData\\Roaming\\nltk_data'

```
filtered_words = [word for word in words if word not in stop_words]
print(filtered_words)
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[8], line 1
----> 1 filtered_words = [word for word in words if word not in stop_words]
      2 print(filtered_words)

NameError: name 'words' is not defined
```

```
from nltk.stem.porter import PorterStemmer
porter = PorterStemmer()
stemmed = [porter.stem(word) for word in filtered_words]
print(stemmed)
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[9], line 3
      1 from nltk.stem.porter import PorterStemmer
      2 porter = PorterStemmer()
----> 3 stemmed = [porter.stem(word) for word in filtered_words]
      4 print(stemmed)

NameError: name 'filtered_words' is not defined
```

```
import nltk
nltk.download('averaged_perceptron_tagger')
```