



Vidyavardhini's College of Engineering & Technology  
Department of Computer Science and Engineering (Data Science)

---

Experiment No.2
Apply Tokenization on given English and Indian Language Text
Date of Performance:
Date of Submission:



## Vidyavardhini's College of Engineering & Technology

### Department of Computer Science and Engineering (Data Science)

---

**Aim:** Apply Tokenization on given English and Indian Language Text

**Objective:** Able to perform sentence and word tokenization for the given input text for English and Indian Language.

#### **Theory:**

Tokenization is one of the first step in any NLP pipeline. Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens. If the text is split into words, then its called as 'Word Tokenization' and if it's split into sentences then its called as 'Sentence Tokenization'. Generally 'space' is used to perform the word tokenization and characters like 'periods, exclamation point and newline char are used for Sentence Tokenization. We have to choose the appropriate method as per the task in hand. While performing the tokenization few characters like spaces, punctuations are ignored and will not be the part of final list of tokens.

#### **Why Tokenization is Required?**

Every sentence gets its meaning by the words present in it. So by analyzing the words present in the text we can easily interpret the meaning of the text. Once we have a list of words we can also use statistical tools and methods to get more insights into the text. For example, we can use word count and word frequency to find out important of word in that sentence or document.



## Vidyavardhini's College of Engineering & Technology

### Department of Computer Science and Engineering (Data Science)

#### Input Text

Tokenization is one of the first step in any NLP pipeline. Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens.

#### Word Tokenization

Tokenization	is	one	of
the	first	step	in
any	NLP	pipeline	Tokenization
is	nothing	but	splitting
the	raw	text	into
small	chunks	of	words
or	sentences	called	tokens

#### Sentence Tokenization

Tokenization is one of the first step in any NLP pipeline

Tokenization is nothing but splitting the raw text into small chunks of words or sentences, called tokens



## Vidyavardhini's College of Engineering & Technology

### Department of Computer Science and Engineering (Data Science)

#### Library required for Preprocessing

```
!pip install nltk

Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.6)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.3.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2023.6.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.1)

import nltk

nltk.download('punkt')

[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Package punkt is already up-to-date!
True
```

#### Sentence Tokenization

```
from nltk.tokenize import sent_tokenize

text = '''Stephenson 2-18 is now known as being one of the largest, if not the current largest star ever discovered, surpassing other stars like VY Canis Majoris and UY Scuti.
Stephenson 2-18 has a radius of 2,150 solar radii, being larger than almost the entire orbit of Saturn (1,940 - 2,169 solar radii).'''

sentences = sent_tokenize(text)

sentences

['Stephenson 2-18 is now known as being one of the largest, if not the current largest star ever discovered, surpassing other stars like VY Canis Majoris and UY Scuti.',
'Stephenson 2-18 has a radius of 2,150 solar radii, being larger than almost the entire orbit of Saturn (1,940 - 2,169 solar radii).']
```

#### Word Tokenization

```
from nltk.tokenize import word_tokenize

words = word_tokenize(text)

words

['Stephenson',
'2-18',
'is',
'now',
'known',
'as',
'being',
'one',
'of',
'the',
'largest',
',',
',',
'if',
'not',
'the',
'current',
'largest',
'star',
']
```

<https://colab.research.google.com/drive/1u8fS2xbvZQr5uwcSISGbXktcof-f0FgU#scrollTo=juvenile-powder&printMode=true>

1/5



# Vidyavardhini's College of Engineering & Technology

## Department of Computer Science and Engineering (Data Science)

16/08/2023, 14:34

textPreprocessingTechniques.ipynb - Colaboratory

```
'ever',  
'discovered',  
,  
'surpassing',  
'other',  
'stars',  
'like',  
'VY',  
'Canis',  
'Majoris',  
'and',  
'UY',  
'Scuti',  
,  
'Stephenson',  
'2-18',  
'has',  
'a',  
'radius',  
'of',  
'2,150',  
'solar',  
'radii',  
,  
'being',  
'larger',  
'than',  
'almost',  
'the',  
'entire',  
'orbit',  
'of',  
'Saturn',  
(,  
'1,940',  
'-',  
'2,169',  
'solar',  
'radii',  
)',
```

```
for w in words:  
    print (w)
```

```
2-18  
is  
now  
known  
as  
being  
one  
of  
the  
largest  
,  
if  
not  
the  
current  
largest  
star  
ever  
discovered  
,  
surpassing  
other  
stars  
like  
VY  
Canis  
Majoris  
and  
UY  
Scuti  
,  
Stephenson  
2-18  
has  
a  
radius  
of  
2,150  
solar
```

<https://colab.research.google.com/drive/1u8fS2xbvZQr5uwcSISGbXktcof-HoFgU#scrollTo=juvenile-powder&printMode=true>

2/5



# Vidyavardhini's College of Engineering & Technology

## Department of Computer Science and Engineering (Data Science)

16/08/2023, 14:34

textPreprocessingTechniques.ipynb - Colaboratory

```
,  
being  
larger  
than  
almost  
the  
entire  
orbit  
of  
Saturn  
(  
1,940  
-  
2,169  
solar  
radii  
)  
.
```

### ▼ Levels of Sentences Tokenization using Comprehension

```
sent_tokenize(text)
```

```
['Stephenson 2-18 is now known as being one of the largest, if not the current largest star ever discovered, surpassing other stars  
like VY Canis Majoris and UY Scuti.',  
'Stephenson 2-18 has a radius of 2,150 solar radii, being larger than almost the entire orbit of Saturn (1,940 - 2,169 solar radii).']
```

```
[word_tokenize(text) for t in sent_tokenize(text)]
```

```
[['Stephenson',  
'2-18',  
'is',  
'now',  
'known',  
'as',  
'being',  
'one',  
'of',  
'the',  
'largest',  
',',  
'if',  
'not',  
'the',  
'current',  
'largest',  
'star',  
'ever',  
'discovered',  
',',  
'surpassing',  
'other',  
'stars',  
'like',  
'VY',  
'Canis',  
'Majoris',  
'and',  
'UY',  
'Scuti',  
',',  
'Stephenson',  
'2-18',  
'has',  
'a',  
'radius',  
'of',  
'2,150',  
'solar',  
'radii',  
',',  
'being',  
'larger',  
'than',  
'almost',  
'the',  
'entire',  
'orbit',  
'of',  
'Saturn',  
'(']
```

<https://colab.research.google.com/drive/1u8fS2xbvZQr5uwcSISGbXktcoFfoFgU#scrollTo=juvenile-powder&printMode=true>

3/5



# Vidyavardhini's College of Engineering & Technology

## Department of Computer Science and Engineering (Data Science)

16/08/2023, 14:34

textPreprocessingTechniques.ipynb - Colaboratory

```
'1,948',  
'_',  
'2,169',  
'solar',  
'radii',  
'\')
```

```
from nltk.tokenize import wordpunct_tokenize
```

```
wordpunct_tokenize(text)
```

```
['Stephenson',  
'2',  
'_',  
'18',  
'is',  
'now',  
'known',  
'as',  
'being',  
'one',  
'of',  
'the',  
'largest',  
',',  
'if',  
'not',  
'the',  
'current',  
'largest',  
'star',  
'ever',  
'discovered',  
',',  
'surpassing',  
'other',  
'stars',  
'like',  
'VY',  
'Canis',  
'Majoris',  
'and',  
'UY',  
'Scuti',  
',',  
'Stephenson',  
'2',  
'_',  
'18',  
'has',  
'a',  
'radius',  
'of',  
'2',  
',',  
'150',  
'solar',  
'radii',  
',',  
'being',  
'larger',  
'than',  
'almost',  
'the',  
'entire',  
'orbit',  
'of',  
'Saturn',  
'(']
```

### ▼ Filtration of Text by converting into lower case

```
text.lower()
```

```
'stephenson 2-18 is now known as being one of the largest, if not the current lar  
gest star ever discovered, surpassing other stars like vy canis majoris and uy sc  
uti.\n      stephenson 2-18 has a radius of 2.150 solar radii. being larger tha
```

```
text.upper()
```

<https://colab.research.google.com/drive/1u8fS2xbvZQr5uwcSISGbXktcof-f0FgU#scrollTo=juvenile-powder&printMode=true>

4/5



# Vidyavardhini's College of Engineering & Technology

## Department of Computer Science and Engineering (Data Science)

---

16/08/2023, 14:34

textPreprocessingTechniques.ipynb - Colaboratory

\*STEPHENSON 2-38 IS NOW KNOWN AS BEING ONE OF THE LARGEST, IF NOT THE CURRENT LARGEST STAR EVER DISCOVERED, SURPASSING OTHER STARS LIKE VY CANIS MAJORIS AND UY SCUTILLI. STEPHENSON 2-38 HAS A RADIUS OF 2.158 SOLAR RADIUS, BEING LARGER THAN

[Colab paid products](#) - [Cancel contracts here](#)

✓ 0s completed at 14:33







### **Conclusion:**

Tools used for tokenization of Indian language input are:

1. IndicNLP Library: IndicNLP is an open-source Python library that provides tokenization tools for several Indian languages. It includes support for various scripts like Devanagari (used for Hindi, Marathi, Sanskrit, etc.), Tamil, Bengali, and more.
2. Stanford NLP: The Stanford NLP library provides support for tokenizing Indian languages, including Hindi and Telugu, using their pre-trained models.
3. Multilingual BERT Models: Multilingual BERT models, such as mBERT and IndicBERT, have been used to tokenize text in various Indian languages. These models can handle multiple languages and have shown good performance for tokenization in Indian scripts.
4. Pynini for Sanskrit: Pynini is a library for working with finite-state transducers and grammars. It has been used to create tokenization and morphological analysis tools specifically for Sanskrit, which has a rich linguistic tradition.
5. ILMT Tokenizer for Tamil: The Indian Language Toolkit (ILMT) provides a language-specific tokenizer for Tamil. It has been developed as part of a larger effort to promote Indian language processing.
6. Malayalam Morphological Analyzer: For Malayalam, in addition to Malaya, there are morphological analyzers and tokenization tools that have been developed to handle the complex morphological structure of the language.
7. Bengali Tokenizers: Several language-specific tokenization tools are available for Bengali, catering to the unique characteristics of the Bengali script.
8. Gujarati Tokenization Tools: There are specific tokenization tools and resources available for the Gujarati language.
9. Punjabi Tokenizers: Language-specific tools are also available for tokenizing Punjabi text, recognizing the script's unique features.



Vidyavardhini's College of Engineering & Technology  
Department of Computer Science and Engineering (Data Science)

---

10. Kannada Tokenization Libraries: Kannada, like other Indian languages, has libraries and resources specifically designed for tokenization.