



SNS COLLEGE OF ENGINEERING



Kurumbapalayam(Po), Coimbatore – 641 007

Accredited by NAAC-UGC with 'A' Grade

Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

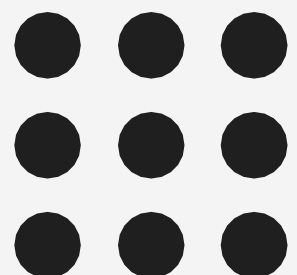
Department of Artificial Intelligence and Data Science

**Course Name – 16AD601 – Natural Language
Processing**

III Year / VI Semester

Unit 1 – Introduction

Topic 5- Sentence Segmentation





Sentence Segmentation



Sentence Segmentation

- Sentence segmentation is another important step in text processing.
- The most use ful cues for segmenting a text into sentences are punctuation, like periods, question marks, and exclamation points.
- Question marks and exclamation points are relatively unambiguous markers of sentence boundaries. Periods, on the other hand, are more ambiguous.
- In general, sentence tokenization methods work by first deciding (based on rules or machine learning) whether a period is part of the word or is a sentence-boundary marker.
- An abbreviation dictionary can help determine whether the period is part of a commonly used abbreviation;



Tokenization



Sentence Tokenization

Sentence tokenization is the process of splitting text into individual sentences. Similar to word tokenization, sentence tokenization can be performed by simple python library function split, NLTK sent_tokenize() module and Regular Expression.

Example

Simple Sentence Tokenization using split

```
text = """Characters like periods, exclamation point and newline char are used to separate the sentences.
But one drawback with split() method, that we can only use one separator at a time! So sentence
tokenization wont be foolproof with split() method."""
```

```
text.split(". ")
```



Tokenization



Sentence Tokenization using NLTK

sent_tokenize() module is used for sentence tokenization.

Example

```
from nltk.tokenize import sent_tokenize
```

```
text = """Characters like periods, exclamation point and newline char are used to separate the sentences.  
But one drawback with split() method, that we can only use one separator at a time! So sentence  
tokenization wont be foolproof with split() method."""
```

```
sent_tokenize(text)
```



Tokenization



Sentence Tokenization using RegEx

Example

```
import re
```

```
text = """Characters like periods, exclamation point and newline char are used to separate the sentences.  
But one drawback with split() method, that we can only use one separator at a time! So sentence  
tonenization wont be foolproof with split() method."""
```

```
tokens_sent = re.compile('[.!?] ').split(text)
```

```
tokens_sent
```

Output

```
['Characters like periods, exclamation point and newline char are used to separate the sentences',  
'But one drawback with split() method, that we can only use one separator at a time',  
'So sentence tonenization wont be foolproof with split() method.']
```



THANK YOU