In [1]:
```
pip install nltk
```

```
Requirement already satisfied: nltk in /home/rmdstic/anaconda3/lib/
python3.7/site-packages (3.4)
Requirement already satisfied: six in /home/rmdstic/anaconda3/lib/p
ython3.7/site-packages (from nltk) (1.12.0)
Requirement already satisfied: singledispatch in /home/rmdstic/anac
onda3/lib/python3.7/site-packages (from nltk) (3.4.0.3)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
import nltk
```

In [3]:
```
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package punkt to /home/rmdstic/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     /home/rmdstic/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /home/rmdstic/nltk_dat
a...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /home/rmdstic/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

Out[3]:
```
True
```

In [4]:
```
text="Tokenization is the first step in the analytics. This process
```

In [9]:
```
from nltk.tokenize import word_tokenize
tokenized_word= word_tokenize(text)
print(tokenized_word)
```

```
['Tokenization', 'is', 'the', 'first', 'step', 'in', 'the', 'analyt
ics', '.', 'This', 'process', 'of', 'breaking', 'down', 'text', 'pa
ragraph', 'into', 'smaller', 'chunks', 'suh', 'as', 'word', 'of', '
sentences', 'is', 'Tokenization', '.']
```

In [11]:
```
from nltk.corpus import stopwords
stop_word= set(stopwords.words("english"))
print(stop_word)
```

```
{'there', 'wouldn', 'the', 'here', 'needn', "you've", "it's", 'been
', 'be', 'you', 'then', "isn't", 'all', 'ain', 'themselves', 'a', "
aren't", 'above', 'own', 'doesn', "haven't", 'ma', "you'd", 'your',
'm', 'y', 'hasn', "that'll", 'her', 'couldn', 'd', "wasn't", 'haven
', 'very', 'or', "didn't", 'being', 'both', 'in', 'yourself', 'each
', 'myself', 'with', "shan't", 'as', 'which', 'after', 'these', 'ca
n', 'aren', 'yourselves', 'ourselves', 'won', 'doing', 'few', 'that
', 'if', 've', 'those', 'and', "don't", 'shan', 'through', 'some',
'who', 'down', 'didn', 'any', "mustn't", 'our', 'them', 'by', 'duri
ng', "won't", "wouldn't", 'mustn', 'will', 'their', 'were', 'until'
, 'into', 'should', "weren't", 'we', 'having', 'an', 'so', "shouldn
't", 'it', 'against', 'my', 'for', "mightn't", 'to', 'most', 'but',
'only', 'nor', 'him', 'too', 'no', "needn't", 'o', 'theirs', "doesn
't", "hadn't", 'had', 'now', "you'll", 'did', 'his', 'itself', 'is'
, 'other', 'not', 'wasn', 'because', 'just', 'at', 'up', 'how', 'me
', "you're", 'why', 'than', 'have', 'its', 'such', "hasn't", 'shoul
dn', 're', "should've", 'himself', 'while', 'they', 'll', 'has', 'o
ver', 'between', 'am', 'yours', 'was', 'again', 'off', 't', 'when',
"she's", 'of', 'about', 'below', 'do', 'she', "couldn't", 'out', 'h
adn', 'from', 'weren', 'what', 'hers', 'once', 'before', 'he', 'fur
ther', 'mightn', 'under', 'don', 'where', 'herself', 'on', 'more',
'same', 'isn', 's', 'ours', 'whom', 'i', 'this', 'are', 'does'}
```

In [12]:
```
import re
```

In [17]:
```
text="The process of breaking down text paragraph into smaller chunks
text=re.sub('[^a-zA-Z]',' ',text)
tokens=word_tokenize(text.lower())
filtered_text = []
for w in tokens:
    if w not in stop_word:
        filtered_text.append(w)
print("Tokenized sentence : ",tokens)
print("filtered sentence : ",filtered_text)
```

```
Tokenized sentence :  ['the', 'process', 'of', 'breaking', 'down',
'text', 'paragraph', 'into', 'smaller', 'chunks', 'such', 'as', 'wo
rd', 'of', 'sentence', 'is', 'tokenization']
filtered sentence :  ['process', 'breaking', 'text', 'paragraph', '
smaller', 'chunks', 'word', 'sentence', 'tokenization']
```

In [19]:
```
from nltk.stem import PorterStemmer
e_words = ["wait","waiting","waited","waits"]
ps=PorterStemmer()
for w in e_words:
    rootword = ps.stem(w)
print(rootword)
```

```
wait
```

In [27]:
```
from nltk.stem import WordNetLemmatizer
wordnet_lemmatizer = WordNetLemmatizer()
text =" studies studing cries cry"
tokenization = nltk.word_tokenize(text)
for w in tokenization:
    print(w,wordnet_lemmatizer.lemmatize(w))
```

```
studies study
studing studing
cries cry
cry cry
```

In [28]:
```
import nltk
from nltk.tokenize import word_tokenize
data = "The black shirt fit him perfectly"
words = word_tokenize(data)
for word in words:
    print(nltk.pos_tag([word]))
```

```
[('The', 'DT')]
[('black', 'JJ')]
[('shirt', 'NN')]
[('fit', 'NN')]
[('him', 'PRP')]
[('perfectly', 'RB')]
```

In [ ]:
```

```