Exploring NLTK

CS 4395 - Assignment 3

By Hamna Mustafa - hbm170002

```
import nltk
nltk.download("stopwords")
nltk.download("wordnet")
nltk.download("punkt")
nltk.download("omw-1.4")
nltk.download("book")
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt 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!
[nltk_data] Downloading collection 'book'
[nltk_data]    |
[nltk_data]    | Downloading package abc to /root/nltk_data...
[nltk_data]    |   Package abc is already up-to-date!
[nltk_data]    | Downloading package brown to /root/nltk_data...
[nltk_data]    |   Package brown is already up-to-date!
[nltk_data]    | Downloading package chat80 to /root/nltk_data...
[nltk_data]    |   Package chat80 is already up-to-date!
[nltk_data]    | Downloading package cmudict to /root/nltk_data...
[nltk_data]    |   Package cmudict is already up-to-date!
[nltk_data]    | Downloading package conll2000 to /root/nltk_data...
[nltk_data]    |   Package conll2000 is already up-to-date!
[nltk_data]    | Downloading package conll2002 to /root/nltk_data...
[nltk_data]    |   Package conll2002 is already up-to-date!
[nltk_data]    | Downloading package dependency_treebank to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Package dependency_treebank is already up-to-date!
[nltk_data]    | Downloading package genesis to /root/nltk_data...
[nltk_data]    |   Package genesis is already up-to-date!
[nltk_data]    | Downloading package gutenberg to /root/nltk_data...
[nltk_data]    |   Package gutenberg is already up-to-date!
[nltk_data]    | Downloading package ieer to /root/nltk_data...
[nltk_data]    |   Package ieer is already up-to-date!
[nltk_data]    | Downloading package inaugural to /root/nltk_data...
[nltk_data]    |   Package inaugural is already up-to-date!
[nltk_data]    | Downloading package movie_reviews to
[nltk_data]    |     /root/nltk_data...
[nltk_data]    |   Package movie_reviews is already up-to-date!
[nltk_data]    | Downloading package nps_chat to /root/nltk_data...
[nltk_data]    |   Package nps_chat is already up-to-date!
```

```
[nltk_data]     | Downloading package names to /root/nltk_data...
[nltk_data]     |   Package names is already up-to-date!
[nltk_data]     | Downloading package ppattach to /root/nltk_data...
[nltk_data]     |   Package ppattach is already up-to-date!
[nltk_data]     | Downloading package reuters to /root/nltk_data...
[nltk_data]     |   Package reuters is already up-to-date!
[nltk_data]     | Downloading package senseval to /root/nltk_data...
[nltk_data]     |   Package senseval is already up-to-date!
[nltk_data]     | Downloading package state_union to /root/nltk_data...
[nltk_data]     |   Package state_union is already up-to-date!
[nltk_data]     | Downloading package stopwords to /root/nltk_data...
[nltk_data]     |   Package stopwords is already up-to-date!
[nltk_data]     | Downloading package swadesh to /root/nltk_data...
[nltk_data]     |   Package swadesh is already up-to-date!
[nltk_data]     | Downloading package timit to /root/nltk_data...
[nltk_data]     |   Package timit is already up-to-date!
[nltk_data]     | Downloading package treebank to /root/nltk_data...
[nltk_data]     |   Package treebank is already up-to-date!
[nltk_data]     | Downloading package toolbox to /root/nltk_data...
[nltk_data]     |   Package toolbox is already up-to-date!
```

TOKENS

List two things you learned about the tokens() method or Text objects

- I learnt that the tokens method converts a Text object to a list of strings, with whitespace as a delimiter
- I learnt that the tokens() method returns a list of strings that can then be manipulated using string handling such as slicing

```
from nltk.book import text1
text1.tokens[:20]
```

```
['[',
 'Moby',
 'Dick',
 'by',
 'Herman',
 'Melville',
 '1851',
 ']',
 'ETYMOLOGY',
 '.',
 '(',
 'Supplied',
 'by',
 'a',
 'Late',
 'Consumptive',
 'Usher',
 'to',
 'a',
 'Grammar']
```

```
text1.concordance("sea",lines=5)
```

```
Displaying 5 of 455 matches:
 shall slay the dragon that is in the sea ." -- ISAIAH " And what thing soever
 S PLUTARCH ' S MORALS . " The Indian Sea breedeth the most and the biggest fis
cely had we proceeded two days on the sea , when about sunrise a great many Wha
many Whales and other monsters of the sea , appeared . Among the former , one w
 waves on all sides , and beating the sea before him into a foam ." -- TOOKE '
```

COUNT

How does this work, and how is it different or the same as Python's count method?

The count() method in the API takes in a Text object and a word as its parameters and counts the number of times that word appears in that Text object. Python's count() method counts elements in a list as well as a string. The main difference between the two is that the Text count() method only works on a Text object, Python's count() works on lists and strings as well.

```
from nltk.text import Text


print(Text.count(text1,"sea"))
print(text1.count("sea"))

print("Hello from the other side".count("side"))

lst = ["she","sells", "seashells", "seafloor"]

print(lst.count("sea"))
```

```
433
433
1
0
```

RAW TEXT

This text was taken from the song 'The Room Where it Happens' in the Hamilton Musical, written by Lin Manuel Miranda. It is my favorite musical.

```
raw_text = "The art of the compromise, hold your nose and close your eyes. We want ou
from nltk import word_tokenize
tokens = word_tokenize(raw_text)
print(tokens[:10])
```

```
['The', 'art', 'of', 'the', 'compromise', ',', 'hold', 'your', 'nose', 'and']
```

```
from nltk import sent_tokenize
sentences = sent_tokenize(raw_text)
for sentence in sentences:
  print(sentence)
```

```
The art of the compromise, hold your nose and close your eyes.
We want our leaders to save the day, but we don't get a say in what they trade a
We dream of a brand new start, but we dream in the dark for the most part.
Dark as a tomb where it happens.
I've got to be in the room where it happens.
```

Differences between Stemmer and Lemmatizer

1. The stemmer made everything lowercase whereas the lemmatizer didn’t
2. In the 4th sentence, the stemmer removed the 's' in happens but the lemmatizer didn't
3. In the 4th sentence, the lemmatizer changed 'as' to 'a' whereas the stemmer didn't
4. In the first sentence, the stemmer changed the word 'compromise' to 'compromis' whereas the lemmatizer didn't
5. Just like in the 4th sentence, the stemmer also removed the 's' in 'happens' in this sentence as well whereas the lemmatizer didn't

```
from nltk.stem.porter import *
stemmer = PorterStemmer()
stemmed = [stemmer.stem(t) for t in tokens]
print(stemmed)
```

```
['the', 'art', 'of', 'the', 'compromis', ',', 'hold', 'your', 'nose', 'and', 'cl
```

```
from nltk.stem import WordNetLemmatizer
wnl = WordNetLemmatizer()
lemmatized = [wnl.lemmatize(t) for t in tokens]
print(lemmatized)
```

```
['The', 'art', 'of', 'the', 'compromise', ',', 'hold', 'your', 'nose', 'and', 'c
```

Summary

a. your opinion of the functionality of the NLTK library I think the NLTK library is a great library for processing Natural Language. It is very functional in trimming the unnecessary parts in words and sentences so that the computer eventually only has to deal with a much more simplified form of human language

b. your opinion of the code quality of the NLTK library

I appreciate that a lot of the code in the NLTK library is clean and simple. The methods are easy to understand. A little amount of code can do powerful things in this library

c. a list of ways you may use NLTK in future projects

I can use NLTK for a lot of future projects that involve NLP. I can use it for standardizing, simplifying and parsing input.