Morphology

December 10, 2023

Date : 10/12/2023

0.1 Morphology

1. Morphology refers to the process of stripping the word to basic unit of meaning.
2. The basic unit of word is known as **morphemes**
3. Types of morphemes are 1) **Free morphemes**, 2) **Bound morphemes**
4. **Free morphemes** = free morphemes are the one which form a word on thier own.
5. **Bound morpheme** = multiple morphemes come togther to form a word are known as Bound morphemes
6. Libraries that facilitate morphology analysis are polyglot, morfessor, pyICU and pycld2.

1 Code - Morhphological analysis

[25]:
```
### ------------------
### Imporing libraries
### ------------------

from polyglot.text import Word, Text

import regex as re

from nltk.tokenize import word_tokenize

import string
```

[14]:
```
### ------------------
### Input text data
### ------------------

text = 'I went to the store, but they were closed, so I had to go to another␣
 ↪store.'
```

1.1 Process Flow

1. Clean input text data
2. Convert the clean dataset into word token
3. Remove punctuations from clean tokens dataset

4. Apply mophological analysis

1.1.1 1. Convert the text dataset into word token

[24]:
```
# Word tokenization is performed using nltk library (word_tokenize)

word_tokens = word_tokenize(text)

word_tokens
```

[24]:
```
['I',
 'went',
 'to',
 'the',
 'store',
 ',',
 'but',
 'they',
 'were',
 'closed',
 ',',
 'so',
 'I',
 'had',
 'to',
 'go',
 'to',
 'another',
 'store',
 '.']
```

The elements of the "text" has been tokenized, as can be seen above

1.1.2 3. Remove punctuations from clean tokens dataset

[26]:
```
'''
To remove the punctuations form the tokens list,

we will iterate over the list and remove elements which match with elements in␣
 ↪the punctuation string
'''

punc = string.punctuation  # String of all punctuations

punc = punc + "'''``"  # adding elements that were not a part of punctuation,␣
 ↪but were in text data

clean_text = [word for word in word_tokens if word not in punc]
```

```
clean_text
```

[26]:
```
['I',
 'went',
 'to',
 'the',
 'store',
 'but',
 'they',
 'were',
 'closed',
 'so',
 'I',
 'had',
 'to',
 'go',
 'to',
 'another',
 'store']
```

Data has been cleaned of any punctuations

1.1.3 4. Apply mophological analysis

[27]:
```
# Now we will apply morphological analysis to clean word tokens dataset

# To apply morphlogical analysis, we will iterate over the list of words and␣
 ↪apply 'Word' method to each

for element in clean_text:

    element = Word(element, language = 'en')

    print(element, '---->', element.morphemes)
```

```
I ----> ['I']
went ----> ['went']
to ----> ['to']
the ----> ['the']
store ----> ['store']
but ----> ['but']
they ----> ['the', 'y']
were ----> ['were']
closed ----> ['close', 'd']
so ----> ['s', 'o']
I ----> ['I']
```

```
had ----> ['had']
to ----> ['to']
go ----> ['go']
to ----> ['to']
another ----> ['an', 'other']
store ----> ['store']
```

[ ]: