CSE6060

Statistical Natural Language Processing

NLP Tool

Name : Kavianand G

Reg. No. : 19MAI0050

Date : 22 – June – 2020

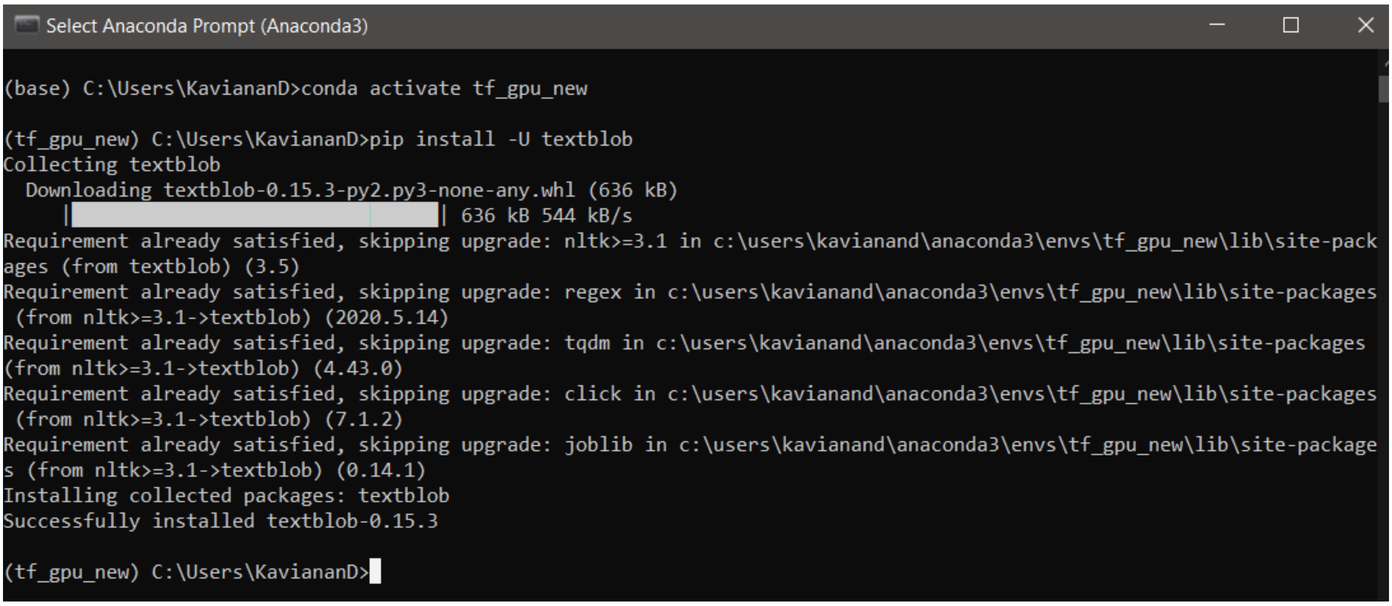
NLP Tool - TextBlob

TextBlob is a Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

Features

Noun phrase extraction
Part-of-speech tagging
Sentiment analysis
Classification (Naive Bayes, Decision Tree)
Tokenization (splitting text into words and sentences)
Word and phrase frequencies
Parsing
n-grams
Word inflection (pluralization and singularization) and lemmatization
Spelling correction
Add new models or languages through extensions
WordNet integration

Installing TextBlob in the Environment



```
(base) C:\Users\KavianandD>conda activate tf_gpu_new

(tf_gpu_new) C:\Users\KavianandD>pip install -U textblob
Collecting textblob
  Downloading textblob-0.15.3-py2.py3-none-any.whl (636 kB)
     |████████████████████████████████| 636 kB 544 kB/s
Requirement already satisfied, skipping upgrade: nltk>=3.1 in c:\users\kavianand\anaconda3\envs\tf_gpu_new\lib\site-pack
ages (from textblob) (3.5)
Requirement already satisfied, skipping upgrade: regex in c:\users\kavianand\anaconda3\envs\tf_gpu_new\lib\site-packages
 (from nltk>=3.1->textblob) (2020.5.14)
Requirement already satisfied, skipping upgrade: tqdm in c:\users\kavianand\anaconda3\envs\tf_gpu_new\lib\site-packages
(from nltk>=3.1->textblob) (4.43.0)
Requirement already satisfied, skipping upgrade: click in c:\users\kavianand\anaconda3\envs\tf_gpu_new\lib\site-packages
 (from nltk>=3.1->textblob) (7.1.2)
Requirement already satisfied, skipping upgrade: joblib in c:\users\kavianand\anaconda3\envs\tf_gpu_new\lib\site-package
s (from nltk>=3.1->textblob) (0.14.1)
Installing collected packages: textblob
Successfully installed textblob-0.15.3

(tf_gpu_new) C:\Users\KavianandD>
```

TextBlob

In [2]:

```
from textblob import TextBlob
```

In [46]:

```
main_text = '''NLP is the practice of understanding how people organise their thinking,
A key element of NLP is that we form our unique internal mental maps of the world as a

```

1.POS Tagging

In [164]:

```python
blob = TextBlob(main_text)
print(blob.tags)
```

```
[('NLP', 'NNP'), ('is', 'VBZ'), ('the', 'DT'), ('practice', 'NN'), ('of',
'IN'), ('understanding', 'VBG'), ('how', 'WRB'), ('people', 'NNS'), ('orga
nise', 'VBP'), ('their', 'PRP$'), ('thinking', 'NN'), ('feeling', 'NN'),
('language', 'NN'), ('and', 'CC'), ('behaviour', 'NN'), ('to', 'TO'), ('pr
oduce', 'VB'), ('the', 'DT'), ('results', 'NNS'), ('they', 'PRP'), ('do',
'VBP'), ('NLP', 'NNP'), ('provides', 'VBZ'), ('people', 'NNS'), ('with',
'IN'), ('a', 'DT'), ('methodology', 'NN'), ('to', 'TO'), ('model', 'VB'),
('outstanding', 'JJ'), ('performances', 'NNS'), ('achieved', 'VBN'), ('b
y', 'IN'), ('geniuses', 'NNS'), ('and', 'CC'), ('leaders', 'NNS'), ('in',
'IN'), ('their', 'PRP$'), ('field', 'NN'), ('NLP', 'NNP'), ('is', 'VBZ'),
('also', 'RB'), ('used', 'VBN'), ('for', 'IN'), ('personal', 'JJ'), ('deve
lopment', 'NN'), ('and', 'CC'), ('for', 'IN'), ('success', 'NN'), ('in',
'IN'), ('business', 'NN'), ('A', 'NNP'), ('key', 'JJ'), ('element', 'NN'),
('of', 'IN'), ('NLP', 'NNP'), ('is', 'VBZ'), ('that', 'IN'), ('we', 'PR
P'), ('form', 'VBP'), ('our', 'PRP$'), ('unique', 'JJ'), ('internal', 'J
J'), ('mental', 'JJ'), ('maps', 'NNS'), ('of', 'IN'), ('the', 'DT'), ('wor
ld', 'NN'), ('as', 'IN'), ('a', 'DT'), ('product', 'NN'), ('of', 'IN'),
('the', 'DT'), ('way', 'NN'), ('we', 'PRP'), ('filter', 'VBP'), ('and', 'C
C'), ('perceive', 'JJ'), ('information', 'NN'), ('absorbed', 'VBN'), ('thr
```

2.Noun Phrase Extraction

In [6]:

```python
blob.noun_phrases
```

Out[6]:

```
WordList(['nlp', 'people organise', 'nlp', 'outstanding performances', 'nl
p', 'personal development', 'key element', 'nlp', 'unique internal mental ma
ps', 'perceive information'])
```

3.Sentiment Analysis

In [123]:

```python
text = TextBlob("NLP class is amazing.")
print(text.sentiment)
text = TextBlob("NLP class is amazing!")
print(text.sentiment)
text = TextBlob("NLP class is amazing!!")
print(text.sentiment)
text = TextBlob("NLP class is amazing!!!")
print(text.sentiment)
text = TextBlob("NLP class is amazing!!!!")
print(text.sentiment)
text = TextBlob("NLP is bad")
print(text.sentiment)
text = TextBlob("NLP is worst")
print(text.sentiment)
```

```
Sentiment(polarity=0.6000000000000001, subjectivity=0.9)
Sentiment(polarity=0.7500000000000001, subjectivity=0.9)
Sentiment(polarity=0.9375000000000001, subjectivity=0.9)
Sentiment(polarity=1.0, subjectivity=0.9)
Sentiment(polarity=1.0, subjectivity=0.9)
Sentiment(polarity=-0.6999999999999998, subjectivity=0.6666666666666666)
Sentiment(polarity=-1.0, subjectivity=1.0)
```

Sentiment Analysis - Observation

The Sentiment polarity and subjectivity mainly depends on the "Punctuation" used in the sentence. The first 5 sentence has positive sentiment polarity while the last two sentence has negative polarity signifies that statement is negative.

In [32]:

```python
text = TextBlob("He died. So sad.")
print(text.sentiment)
text = TextBlob("he is sad.")
print(text.sentiment)
```

```
Sentiment(polarity=-0.5, subjectivity=1.0)
Sentiment(polarity=-0.5, subjectivity=1.0)
```

4.Spelling Correction

In [40]:

```python
text = TextBlob('Computre Sciene is a banch of Enginering')
print(text.correct())
text = TextBlob('Kavi anand')
print(text.correct())
text = TextBlob('Kavi anand is norml persn')
print(text.correct())
```

```
Computer Science is a branch of Engineering
Have and
Have and is normal person
```

Spelling Correction - Observation 1

The noun words in the sentence cannot be subjected to Spelling Correction. They were mis-identified. Before subjecting a Paragraph to spelling correction, Noun words have to be taken care of.

In [138]:

```python
word = TextBlob("scence")
(word.words[0].spellcheck())
```

Out[138]:

```
[('science', 0.5504587155963303), ('scene', 0.44954128440366975)]
```

In [137]:

```python
word = TextBlob("scence")
(word.words[0].spellcheck())
```

Out[137]:

```
[('science', 0.5357142857142857),
 ('scene', 0.4375),
 ('spence', 0.026785714285714284)]
```

Spelling Correction - Observation 2

The probability of words for misspelled word "Science" was illustrated above.

5. Basic Grammatical Operations

In [148]:

```python
text = TextBlob(main_text)
print(text)
```

```
NLP is the practice of understanding how people organise their thinking, fee
ling, language and behaviour to produce the results they do. NLP provides pe
ople with a methodology to model outstanding performances achieved by genius
es and leaders in their field. NLP is also used for personal development and
for success in business
A key element of NLP is that we form our unique internal mental maps of the
world as a product of the way we filter and perceive information absorbed th
rough our five senses from the world around us.
```

In [149]:

```python
# Break into Sentences and words
print("Number of Sentences \t\t\t:", len(text.sentences))
print("Number of Words in First Sentence \t:" , len(text.sentences[1].words))
print("Total number of words \t\t\t:" , len(text.words))
```

```
Number of Sentences                     : 3
Number of Words in First Sentence       : 18
Total number of words                   : 89
```

In [161]:

```python
# can do Words count for a string

print("the \t-> ", text.word_counts["the"])
print("NLP \t-> ", text.word_counts["nlp"])
print("the \t-> ", text.word_counts["we"])
```

```
the     ->   5
NLP     ->   4
the     ->   2
```

In [105]:

```python
from textblob import Word
# singular and plural form
word_1 = Word("tries")
word_2 = Word("try")
print("Y Replacement Rule \t->\t "+ word_1.singularize() +" - " +word_2.pluralize())

word_1 = Word("Watches")
word_2 = Word("watch")
print("E insertion Rule \t->\t "+ word_1.singularize() +" - " +word_2.pluralize())

print(" --> Miscelleneous Try")
words = Word("kavis")
print("\t", words.singularize())
words = Word("kavi")
print("\t",words.pluralize())
```

```
Y Replacement Rule      ->       try - tries
E insertion Rule        ->       Watch - watches
 --> Miscelleneous Try
         kavi
         kavis
```

In [111]:

```
for word,pos in blob.tags:
    if pos == 'NN':
        print (word.singularize() + "\t\t-\t\t" +word.pluralize())
```

```
practice                -               practices
thinking                -               thinkings
feeling         -               feelings
language                -               languages
behavimy                -               behaviours
methodology             -               methodologies
field           -               fields
development             -               developments
succes          -               successes
busines         -               businesses
element         -               elements
world           -               worlds
product         -               products
way             -               ways
information             -               information
world           -               worlds
```

6. Language Detection

In [139]:

```
txt = TextBlob("மகிழ்ச்சி")
```

In [145]:

```
# Can detect language
txt.detect_language()
```

Out[145]:

```
'ta'
```

```
     # Here 'ta' stands for Tamil
```

In [146]:

```
# Language Translation also available in TextBlob
print(txt.translate(from_lang='ta', to ='en'))
print(txt.translate( to ='en'))
```

```
Happiness
Happiness
```

In [ ]:

```

```

Pros:

Since, it is built on the shoulders of NLTK and Pattern, therefore making it simple for beginners by providing an intuitive interface to NLTK.

It provides language translation and detection which is powered by Google Translate (not provided with Spacy).

Cons:

It is little slower in the comparison to spacy but faster than NLTK. (Spacy > TextBlob > NLTK)

It does not provide features like dependency parsing, word vectors etc. which is provided by spacy.

---End of Documentation---

Submitted on 22 - June - 2020

Submitted by Kavianand G