Lab 3

November 14, 2022

1. Import the necessary libraries.

[68]:
```
import nltk
from nltk.tokenize import word_tokenize, sent_tokenize
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
```

2. Load the text in data.txt to a variable text. Apply lower casing.

[69]:
```
text = """Natural language processing makes it possible for computers
to understand the human language. In natural language
processing, human language is separated into fragments so that
the grammatical structure of sentences and the meaning of words
can be analysed and understood in context. This helps computers
read and understand spoken or written text in the same way as
humans. I am studying Natural Language Processing at Amrita
University."""
```

[70]:
```
text = text.lower()
```

[71]:
```
wordTokens = word_tokenize(text)
```

[72]:
```
splitTokens = text.split()
```

[73]:
```
print(len(wordTokens))
print(len(splitTokens))
```

```
72
67
```

[74]:
```
missing = [x for x in wordTokens if x not in splitTokens]
missing
```

[74]:
```
['.', ',', 'context', '.', 'humans', '.', 'university', '.']
```

[75]:
```
missing2 = [x for x in splitTokens if x not in wordTokens]
missing2
```

[75]:
```
['language.', 'processing,', 'context.', 'humans.', 'university.']
```

4 a). How is str.split() different from word tokenizer?

str.split() – splits the words by empty spaces and not by special character or anything
word_tokenize – splits the sentence by spaces and by special characters

5. Apply the sentence tokenization process to the text and store in sentTokens.

[76]:
```
sentTokens = sent_tokenize(text)
sentTokens
```

[76]:
```
['natural language processing makes it possible for computers\nto understand the
human language.',
 'in natural language\nprocessing, human language is separated into fragments so
that\nthe grammatical structure of sentences and the meaning of words\ncan be
analysed and understood in context.',
 'this helps computers\nread and understand spoken or written text in the same
way as\nhumans.',
 'i am studying natural language processing at amrita\nuniversity.']
```

[77]:
```
file1 = open('sentence', 'r')
Lines = file1.readlines()
print(Lines)
```

```
['Natural language processing makes it possible for computers\n', 'to understand
the human language. In natural language\n', 'processing, human language is
separated into fragments so that\n', 'the grammatical structure of sentences and
the meaning of words\n', 'can be analysed and understood in context. This helps
computers\n', 'read and understand spoken or written text in the same way as\n',
'humans. I am studying Natural Language Processing at Amrita\n',
'University.\n']
```

a. How is readlines() different from sentence tokenizer.

b. How many tokens are there now?

[78]:
```
print(len(sentTokens))
print(len(Lines))
```

```
4
8
```

c. Print the tokens and compare them with the readlines.

[79]:
```
missing2 = [x for x in sentTokens if x not in Lines]
missing2
```

[79]:
```
['natural language processing makes it possible for computers\nto understand the
human language.',
 'in natural language\nprocessing, human language is separated into fragments so
that\nthe grammatical structure of sentences and the meaning of words\ncan be
```

```
analysed and understood in context.',
 'this helps computers\nread and understand spoken or written text in the same
way as\nhumans.',
 'i am studying natural language processing at amrita\nuniversity.']
```

6. Apply spelling correction on each word tokens and print the initial 15 misspelled tokens as well as the corrected tokens. Keep a count of corrected tokens.

7. Remove stop words and punctation characters from the corrected token list.

[80]:
```
StopWords = stopwords.words("english")
no_stopwords = [x for x in wordTokens if x not in StopWords]
```

8. Stem each tokens.

[81]:
```
stemmer = PorterStemmer()
stemmed_words = [stemmer.stem(i) for i in wordTokens]
```

[ ]:
```

```