Experiment No. 2

Aim: To Study Morphological analysis

Theory:

Morphological Analysis:

While performing the morphological analysis, each particular word is analyzed. Non-word tokens such as punctuation are removed from the words. Hence the remaining words are assigned categories. For instance, Ram's iPhone cannot convert the video from .mkv to .mp4. In Morphological analysis, word by word the sentence is analyzed. So here, Ram is a proper noun, Ram's is assigned as possessive suffix and .mkv and .mp4 is assigned as a file extension.

As shown above, the sentence is analyzed word by word. Each word is assigned a syntactic category. The file extensions are also identified present in the sentence which is behaving as an adjective in the above example. In the above example, the possessive suffix is also identified. This is a very important step as the judgment of prefixes and suffixes will depend on a syntactic category for the word. For example, swims and swims are different. One makes it plural, while the other makes it a third-person singular verb. If the prefix or suffix is incorrectly interpreted then the meaning and understanding of the sentence are completely changed. The interpretation assigns a category to the word. Hence, discard the uncertainty from the word

Regular Expression:

Regular expressions also called regex. It is a very powerful programming tool that is used for a variety of purposes such as feature extraction from text, string replacement and other string manipulations. A regular expression is a set of characters, or a pattern, which is used to find sub strings in a given string. for ex. extracting all hashtags from a tweet, getting email id or phone numbers etc., from a large unstructured text content.

In short, if there's a pattern in any string, you can easily extract, substitute and do variety of other string manipulation operations using regular expressions. Regular expressions are a language in itself since they have their own compilers and almost all popular programming languages support working with regexes.

Stop Word Removal:

The words which are generally filtered out before processing a natural language are called stop words. These are actually the most common words in any language (like articles, prepositions, pronouns, conjunctions, etc) and does not add much information to the text. Examples of a few stop words in English are "the", "a", "an", "so", "what".

Stop words are available in abundance in any human language. By removing these words, we remove the low-level information from our text in order to give more focus to the important information. In order words, we can say that the removal of such words does not show any negative consequences on the model we train for our task.

Removal of stop words definitely reduces the dataset size and thus reduces the training time due to the fewer number of tokens involved in the training.

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

Synonym:

The word synonym defines the relationship between different words that have a similar meaning. A simple way to decide whether two words are synonymous is to check for substitutability. Two Words are synonyms in a context if they can be substituted for each for each other without changing the meaning of the sentence.

Stemming:

Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words known as a lemma. Stemming is important in natural language understanding (NLU) and natural language processing (NLP).

Code

Regular Expression:

```
import re

input="The 5 biggest animals are 1. Elephant,2 Rhino and 3 dinasaur"

input=input.lower()

print(input)

result= re.sub(r'\d+','',input)

print(result)
```

Output:

the 5 biggest animals are 1. elephant,2 rhino and 3 dinasaur

the biggest animals are . elephant, rhino and dinasaur

Stop word removal:

```
def punctuations(raw_review):
 text = raw_review
 text = text.replace("n't", ' not')
 text = text.replace("'s", ' is')
 text = text.replace("'re", ' are')
 text = text.replace("'ve", ' have')
 text = text.replace("'m", ' am')
 text = text.replace("'d", ' would')
 text = text.replace("'ll", ' will')
 text = text.replace("in", 'ing')
 import re
 letters_only = re.sub("[^a-zA-Z]"," ",text)
 return(''.join(letters_only))
t="Hows's my team doin, you're supposed to be not loosin"
p=punctuations(t)
print(p)
```

Output

Hows is my team doing you are supposed to be not loosing

Synonym:

```
import nltk
nltk.download('wordnet')
from nltk.corpus import wordnet
```

```
synonyms = []
for syn in wordnet.synsets(Machine'):
for lemma in syn.lemmas():
synonyms.append(lemma.name())
print(synonyms)
```

Output:

['machine', 'machine', 'machine', 'machine', 'simple_machine', 'machine', 'political_machine', 'car',

'auto', 'automobile', 'machine', 'motorcar', 'machine', 'machine']

Stemming:

```
from nltk.stem import PorterStemmer
stemmer = PorterStemmer()
print(stemmer.stem('eating'))
print(stemmer.stem('ate'))
```

Output:

eat

ate

Conclusion:

Thus, in the above experiment we have studied regarding morphological analysis in detail with stemming, synonym, stop word removal, regular expression and tried to implement the code and got proper output.