1. Brief description of the tokenizer, spell correctors, stemmer, and lemmatizer you used.

The given data set assignment\_data.txt is subdivided into 3 txt files as student\_data.txt, tweets\_data.txt, research\_data.txt. Because they are totally different formatted data so per-processing techniques should be varied. It had done by nltk.tokenize package.

**Tokenization**

Accuracy was checked by difference between before and after tokenization.

Defect percentage =

(|beforeTokenizedWordCount – afterTokenizedWordCount|) \* 100%/ beforeTokenizedWordCount

if it 0 then accuracy is high, when it increases implices that the low accuracy occurred form the used tokenization method

for student\_data.txt

|  |  |  |
| --- | --- | --- |
| Tokenizer used | word\_tokenize |  |
| Before tokenize word count is | = 401 |  |
| after tokenize word count is | = 434 |  |
| Defect percentage | =(|434 - 401 |)/401 \*100  =8.23% |  |

|  |  |
| --- | --- |
| Tweets\_data.txt | Research\_data.txt |
|  |  |
|  |  |
|  |  |
|  |  |

|  |
| --- |
| spell corrector |
| Stemmer |
| lemmatizer |

2. Discussion as given in the task. ­

1. For each text type in assignment\_data.txt, tokenize the text using an available tokenizer. Discuss

about the accuracy of the tokenizer on each type of text.

2. Carry out isolated word correction and context sensitive word correction on the text. Based on

your observations on the processed text, discuss the impact of each type of spell corrections on

the three types of text.

3. Stem and lemmatize the text using a suitable stemmer and a lemmatizer. Based on your

observations on the processed text, discuss the suitability of stemming and lemmatizing for

retrieving base forms of words.

References

<https://pythonspot.com/tokenizing-words-and-sentences-with-nltk/>

<https://www.nltk.org/book/ch03.html>

<https://medium.com/@datamonsters/text-preprocessing-in-python-steps-tools-and-examples-bf025f872908>

<https://www.datacamp.com/community/tutorials/text-analytics-beginners-nltk>

<https://rustyonrampage.github.io/text-mining/2017/11/28/spelling-correction-with-python-and-nltk.html>

<https://media.readthedocs.org/pdf/pyspellchecker/latest/pyspellchecker.pdf>

<https://www.tutorialspoint.com/python/python_spelling_check.htm>

<https://www.youtube.com/watch?v=2SjQBRagpkg>

<https://pypi.org/project/pyspellchecker/>

<https://nlp.stanford.edu/IR-book/html/htmledition/context-sensitive-spelling-correction-1.html>

<https://norvig.com/spell-correct.html?fbclid=IwAR2-SyWinhXws01i8FldkXImPw6gBhQGwdVg8IDUbVltMWEAwhUvdJE3K5M>

<https://norvig.com/spell-correct.html>

<https://dzone.com/articles/nlp-tutorial-using-python-nltk-simple-examples>