**Akaike Technologies: NLP Assignment**

**Objective 1:**

**Methodology followed:**

Procedure to extract most common noun phrases:

* Load the json file and form a dataframe containing tweet\_author and tweet\_text
* Pre-process the text and clean it getting rid of punctuations/links/hashtags
* Traverse through the tweet text and pick up entities keeping in mind POS. Add terms that aren’t stopwords; are nouns/noun phrases.
* Making use of Counter from collections, count the frequency of each term

**Limitations:**

* Instead of picking up the same phrases, a lot of phrases which are similar in human knowledge cannot be identified: For example: Chronic lymphocytic leukemia and CLL.

Solution proposed: Making use of sentence transformers/CNN/vectorizers, add most similar phrases as one.

* Making use of POS- Cannot pick-up and identify the same terms. For example:

1. Noun phrases picked up from sentence 1: CLL

2. Noun phrases picked up from sentence 2: CLL mouse mo

Solution proposed: Models can be trained in context of the given text.

* Cannot retrieve original text from the tweet

Note: Instead of picking up the most frequent entities encountered by getting rid of stopwords, most frequent noun-phrases, subjects from the tweets are extracted.

Libraries used: JSON, TextBlob, NLTK

**Objective 2:**

**Proposed approach:**

* Making use of NLTK/Spacy/StanfordNER- pick up NN/Nouns from the sentence by traversing through the tree obtained.
* Along with it, look into associated adjectives for the same- try sentiment analysis on the same using TextBlob.
* To find the topics- LDA can be used as well

Limitations of the proposed approach:

* Adjectives/Nouns aren’t identified for most of the sentences- pre-training is required.
* To find the adjective associated with the given noun- a vague position based aspect and description matching has to be followed- room for error is high
* Various languages are not supported in a single model

**Methodology followed:**

Procedure:

* Pre-process the text in the same way as mentioned above
* Traverse the tweet\_text of the data frame along with the authors; pick up the noun phrases from the given tweet.
* Make use of aspect-based sentiment analysis to find the sentiment associated with a particular term in the entire sentence.
* Form a dataframe of the same

**Limitations:**

* Along with positive-negative, a sentiment ‘neutral’ is present as well which needs to be eliminated
* Time taken for calculating the results could be reduced by fine tuning the absa model
* The original form of the noun-phrases cannot be retrieved.

Libraries used: TextBlob, aspect-based-sentiment-analysis

**Future scope:**

* Google cloud natural language API could be put into use for aspect/entity based sentiment analysis
* Fine-tuning/pre-tuning the model to pick up better sentiments from the given blob of text

**References:**

* Aspect based sentiment analysis using Spacy and TextBlob retrieved from <https://towardsdatascience.com/aspect-based-sentiment-analysis-using-spacy-textblob-4c8de3e0d2b9>
* <https://pypi.org/project/aspect-based-sentiment-analysis/>
* Implementing aspect based sentiment analysis using Python retrieved from <https://medium.com/analytics-vidhya/aspect-based-sentiment-analysis-a-practical-approach-8f51029bbc4a>