SUMMARY OF LIBRARIES AND PACKAGES USED IN PROJECT

* NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical sources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum .
* Newspaper 3k - Newspaper is a Python module used for extracting and parsing newspaper articles. Newspaper use advance algorithms with web scrapping to extract all the useful text from a website. It works amazingly well on online newspapers websites. Since it use web scrapping too many request to a newspaper website may lead to blocking, so use it accordingly.
* CountVectorizer- In order to use textual data for predictive modeling, the text must be parsed to remove certain words – this process is called **tokenization**. These words need to then be encoded as integers, or floating-point values, for use as inputs in machine learning algorithms. This process is called feature extraction.

Scikit-learn’s CountVectorizer is used to convert a collection of text documents to a vector of term/token counts. It also enables the ​pre-processing of text data prior to generating the vector representation. This functionality makes it a highly flexible feature representation module for text.

* Cosine\_similarity - Cosine similarity, or the cosine kernel, computes similarity as the normalized dot product of X and Y.

Cosine similarity is one of the most widely used and powerful similarity measure in Data Science. It is used in multiple applications such as finding similar documents in NLP, information retrieval, finding similar sequence to a DNA in bioinformatics, detecting plagiarism and may more.

Cosine similarity is calculated as follows,

* Punkt- This tokenizer divides a text into a list of sentences, by using an unsupervised algorithm to build a model for abbreviation words, collocations, and words that start sentences. It must be trained on a large collection of plaintext in the target language before it can be used.

The NLTK data package includes a pre-trained Punkt tokenizer for English.