**USER MANUAL FOR SARCASM DETECTION**

The **BinarySarcasmDetection.ipynb** and **BinarySarcasmDetection2.ipynb** consists of the code to classify the headline into sarcasm or not.

**Python libraries to be installed:**

pip install pandas

pip install numpy

pip install nltk

pip install scikit-learn

pip install textblob

**NLTK data download:**

import nltk

nltk.download('vader\_lexicon')

nltk.download('averaged\_perceptron\_tagger')

nltk.download('wordnet')

nltk.download('punkt')

nltk.download('stopwords')

**External Libraries:**

re (Regular Expressions)

string

**Libraries for pre-processing:**

from nltk.tokenize import word\_tokenize

from nltk.stem import PorterStemmer

**Sentiment analyzer:**

from nltk.sentiment.vader import SentimentIntensityAnalyzer

**Vectorization:**

from sklearn.feature\_extraction.text import TfidfVectorizer

**Train-test split:**

from sklearn.model\_selection import train\_test\_split

**Classifiers:**

from sklearn.naive\_bayes import MultinomialNB

from sklearn.linear\_model import LogisticRegression

from sklearn.ensemble import RandomForestClassifier

**Metrics evaluation libraries:**

from sklearn.metrics import accuracy\_score, precision\_score, recall\_score

The **AutomatedClassification.ipynb** is used to annotate the binary labelled data into various types of sarcasms (deadpan, obnoxious, Self-deprecating, maniac, brooding, polite and raging)

Libraries:

!pip install openai

Process:  
Load the dataset and select the binary labeled column and select the sarcastic headlines for further classification. Use the OpenAI API key and provide the prompt for classification using the ‘text-davinci-003’ model of GPT.

Process to get the API key:

Login to the OpenAI website and go to API section. Within first three month of signing-up the account, maximum of 5$ free usage will be provided. Get the API key and use it for classification.

The datasets **Sarcasm\_\_dataet.csv** and **deadpan\_sarcasm.csv** are the datasets for binary classification and deadpan sarcasm classification.

The **DeadpanSarcasm (1).ipynb** is the code file for deadpan sarcasm classification.

Additional libraries required are:

1. Afinn for sentiment analysis.
2. Random Under Sampler to handle data imbalance from the ‘imblearn.under\_sampling’ module.
3. Sklearn.pipeline is used to create data processing and modeling pipeline.

All the codes are implemented on the python environment (google colab).