1. Perform pre-processing (use nltk library)
    a. Remove stop-words
    b. Remove any hashtag and user account
    c. Remove noise, such that, noise is any word that is less than three letters
2. Text Processing
    a. Find the top 50 keywords from the dataset by tf-idf
    b. Keep only the nouns from tweet text and then, find the top 50 nouns from the dataset by tf-idf
3. Network Science and Pretrained Language Model
    a. Create a graph, G where nodes are tweets and the edges represent cosine similarity between the tweet text. Basically, two tweets are connected by an edge if their cosine similarity is greater than 0.5 else there is no edge. For graph creation, use networkx and for tweets, use the tweet text after performing all the steps of (1) pre-processing
    b. From this graph,
        i. Calculate the degree distribution of the graph
        ii. Calculate degree centrality of all tweets and print the top 10 tweets
        iii. Calculate page rank centrality of all the tweets and print the top 10 tweets
        iv. Share your comments on which centrality is better in this scenario with examples.
        v. Calculate the clustering co-efficient of the whole graph, top 10 tweets and bottom 10 tweets. You can consider degree centrality for understanding the top 10 tweets
    c. Repeat step (a) with cosine similarity calculated based on pre-trained BERT uncased embedding and therefore, generate a graph G1 where two tweets are connected by an edge if their cosine similarity between bert embedding is greater than 0.5 else there is no edge.
        i. Repeat step b (i to v)
            1. Any reasoning which one is better? BERT or text based? Provide examples for each (i to v)
4. Sentiment Calculation
    a. Calculate the sentiment of each tweet by Vader sentiment analyzer and SentiNet. Provide three examples of where Vader fails and 3 examples where SentiNet fails