Homework 8

1. Consider the following newspaper headlines

   Headline 1: Their COVID vaccine is 95% effective, says Pfizer.
   Headline 2: Like Pfizer, Moderna's COVID vaccine trial is 95% effective
   Headline 3: AstraZeneca's COVID vaccine trial halted.
   Headline 4: The race for the COVID vaccine is over – Pfizer and Moderna vaccines are 95% effective.

a. Find the inverse document frequency (idf) of each word in this corpus. Ignore the stop words and punctuations. A list of stop words can be found here https://gist.github.com/sebleier/554280. Note that you can lemmatize words like 'says' to 'say' and 'trials' to 'trial'.
b. Show each document as a tf-idf weighted vector.
c. Normalize the document vectors and show each one of them.
d. Given the user query "effective COVID vaccines", find the most similar news headline.
e. Rank the headlines based on their similarity scores w.r.t. the query.

2. A. Consider the table of term frequencies for 3 documents denoted Doc1, Doc2, Doc3 in Table 1(a). Compute the tf-idf weights for the terms car, auto, insurance, best, for each document, using the df values from Table 1(b). Assume a corpus size of 100,000 documents.

	Doc1	Doc2	Doc3
car	27	4	24
auto	3	33	0
insurance	0	33	29
best	14	0	17

Table 1(a) Term Frequency

term	DF
car	18,165
auto	6,723
insurance	19,241
best	25,235

Table 1(b) Document Frequency

B. Compute the normalized document vectors for each of the documents.

C. Compute the cosine similarity between each pair of documents.

3. Let the google query be Q= *TFIDF cosine similarity*.

Consider the top 5 URLs returned by Google for this query. Rank the corresponding web pages by their approximate cosine similarity to the query Q.

The approximate cosine similarity relies only on query terms (3 terms) and3-dimesional TF-IDF vectors for each document using just query terms. In this case the query terms are TFIDF (or TF-IDF), cosine and similarity.

Show the values of cosine similarity to Q for each document as well as their tf-idf vectors and explain how you obtained their values.

Assume total number of Google documents as 100 trillion (N). Assume DF for term t of the query to be approximate number of google results for this term t. Get term frequency (TF) of term t in a document D, using control-F search function.