Text Analytics: Practical 4
(for Lecture 4: Beyond Frequencies)

1) Find or make up 10 short text-items (10-20 words in each item); they could be supposed/actual emails, short docs, tweets or whatever…

   Make sure they all deal with some common topic of interest (i.e., that they mention the same words):

   a. Remove the standard stop-words from them using some standard list, use nltk, so that you now have the remaining words (the R-words) for each short-text-item
   b. Compute the TF scores for these R-words across all the text-items and use R to show the wordcloud for these words. In your answer provide the matix of TF scores and the wordcloud.
   c. Now, compute the TF-IDF scores for these R-words across all the text-items. Also, provide the matrix of TF-IDF scores.
   d. Discuss the changes that occur in the relative ranking of the top-ranked words between the TF-scoring and the TF-IDF scoring and explain why these changes occur (if any)

2) Using Python or R, compute the PMI scores for all adjacent pairs of words in your 10-doc corpus (i.e., the texts after stop-word removal). [Hint: You will need to write/find a program that generates all bi-grams for a given text or sentence]

   List the top-5 pairs based on the PMI scores found each pairs.

   Do the results make sense? If not, then introduce a minimal cut-off frequency and re-compute the top-5 until they seem sensible.

3) Entropy has been used to determine whether tweet set is interesting (contains variety) or repetitive (spam).

   Create two sets of 10 made-up tweets:

   a. **spam-set**: where the 10 tweets are very similar containing an advert for a product; they way spam tweets look
   b. **random-set**: where the 10 tweets are all different, chosen at random from Twitter.

   Now, find a Python/R program or package that computes entropy and find the entropy values for (i) spam-set, (ii) random-set, (iii) the two sets combined.

   Report the program you used and its source, the tweet data and the entropy values found.