**1. Create a TF IDF matrix from the dataset. Some steps make take several minutes to process in R.**

**TF is the Term frequency--How many times does a word appear in a document:**

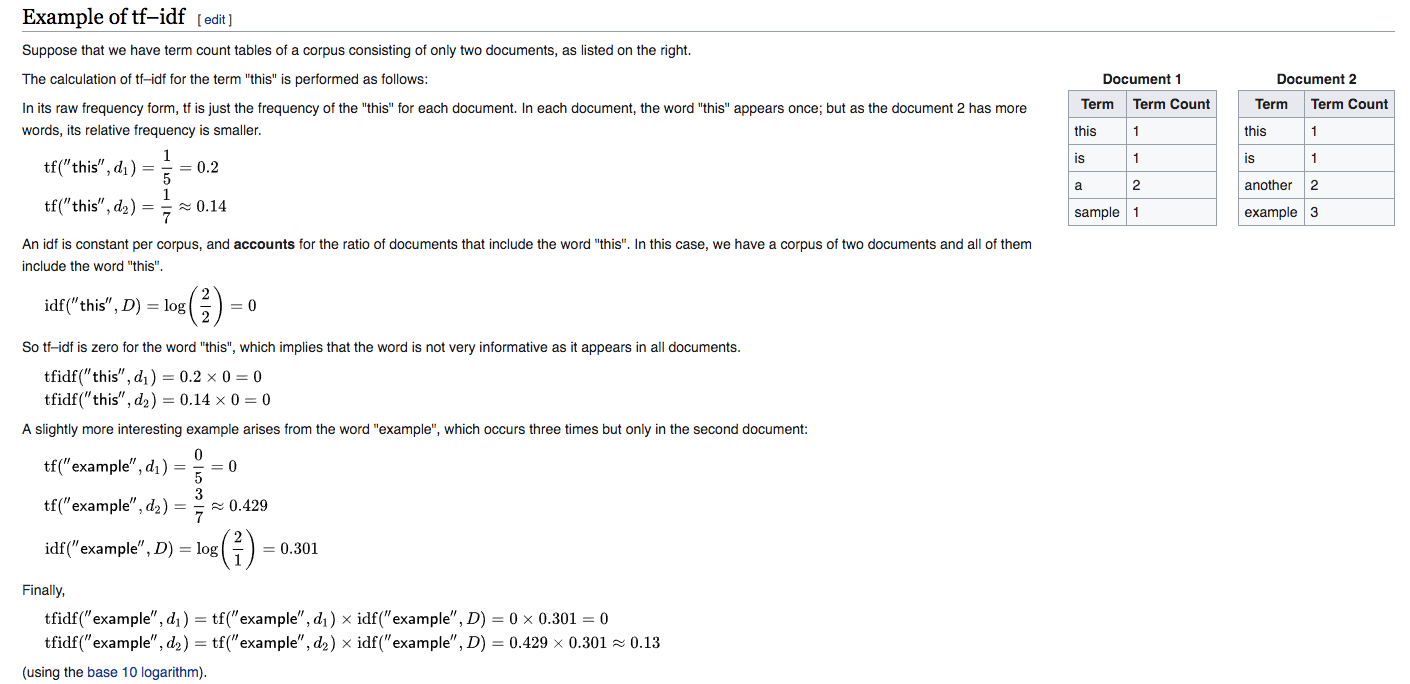
**IDF is the inverse document frequency-- the inverse proportion of documents the term appears in:**

**Where N is the total number of documents and n is the number of documents a term appears in.**

**Hint--create a matrix counts first. You can then count the number of non-zero entries in a column to get "n" for each word.**

**ANS:**

**TF is calculated per word per document; idf is constant for a word across corpus(per corpus - group of documents).**



Reference: <https://en.wikipedia.org/wiki/Tf%E2%80%93idf>

2. Do a K-means clustering for 5,10,20 clusters. What do you observer about the information R returns you? What do each of the parameters represent when you run the kmeans() function.

(Hint assign the result to a variable so you can inspect it!)

3 Run a DBScan on the data. How are the results different from your kmeans? What is the eps? ( hint: help(dbscan) ) What is the meaning of cluster 0?

4. These papers are for a conference submission. When would you use kmeans vs. DBScan in this instance.

5. Using a cluster size of 10 and kmeans, create a word cloud of the different clusters most popular words. Are you able to determine a difference among the papers in different clusters?