ASSIGNMENT 2

Data Set: http://archives.textfiles.com/stories.zip dataset

Pre-processing Steps:

1) Have used porter stemmer to perform stemming of the documents and the query.
2) Have removed all the commas across all the docs and replaced it with ‘’ in order to handle cases for numbers like 50,000 .
3) Have performed num2words() to convert the digits to numbers.
4) Stop words were not given separate treatment but were taken care while handling titles.

Technique 1: Jaccard Score

Steps:

- Preprocess the query and all the docs.
- Calculate length of the intersection and union between query terms and a doc.
- Calculate Jaccard score = len(intersection)/math.sqrt(len(union))
- Sort the jaccard score of all the docs in decreasing order.
- Return top k results.

Technique 2 : TF-IDF score

Steps:

- Preprocess the query and all the docs.
- Calculate summation of tf-idf score for each query terms in one doc.
- Perform previous step for all the docs.
- Sort the tf-idf score of all the docs in decreasing order.
- Return top k results.

Variations of TF-IDF schemes

TF score	IDF Score
n (natural Term frequency)	Log(N/df)
1+log(1+tf) (logarithmic term frequency)	Log(N/df)
{0,1}(Boolean)	Log(N/df)

Comparison among these variants is performed in analysis.pdf file.

Handling Titles

Whenever any query term *which is not a stop word is* coming in the title of the doc then I have considered the weighing factor 0f 0.7 otherwise weighing factor is taken as 0.3.

Technique 3: Cosine similarity

- Dot product of Normalized query vector and document vector is taken and sorted in the decreasing order of the similarity.
- Return top k results.