SI 650/EECS 549 - Information Retrieval
Assignment 2

Due Thursday, Oct 11th, 23:59 EDT .
Please submit attachments via Canvas.
General discussion encouraged, but everyone should come up with the solution independently.
If you received help from anyone, you should list the name(s) on the top of your submission.

The goal of this assignment is to understand how a real retrieval system works. The major task is to implement a basic retrieval algorithm with the metapy package, which is introduced in the textbook, and to experiment with the algorithms on some sample text data. Please download the ipython notebook in the attachment. You can load it locally or upload to google colab (`https://colab.research.google.com`), which is recommended.

1 Build Retrieval Function

In general, you need a function that computes a score of every document by aggregating the weight of every word that occurs in both the document and the query:

$$s(q,d) = g[f(w_1,q,d) + ... + f(w_N,q,d), q, d], \tag{1}$$

where $\{w_1, w_2, ..., w_N\}$ are terms in the query q that appeared in the documents d. That is, the score of a document d given a query q is a function g of the accumulated weight f for *each matched term*.

We have implemented a simple retrieval function called **SimpleRanker** in one of the cell in the ipython notebook.

Task (40 points) Implement the Pivoted Normalization and BM25. You can find the formulas in the lecture slides. In the ipython notebook, you can find a cell to define class named **Pivoted** and another class named **BM25**. You need to change the methods in the class to implement them. **Please include your code for this retrieval function in your submission.**
Note: If you want to check whether your retrieval function runs normally, you can test the code in the end of the notebook to illustrate search results for your queries.

2 Evaluate your Retrieval Function

You can evaluate the performance of the retrieval function by some metrics, such as MAP and Precision@30, which is implemented in the notebook already. You need to compare and choose retrieval function using these metrics.

Task (20 points) Specify 9 parameter values for your Pivoted Normalization function (0.1, 0.2, $\cdots$, 0.9). Plot the performance curve using MAP (y-axis), against the parameter value (from 0.1 to 0.9, on x-axis) on the cranfield data. Report the best performance of your function (including the parameter value). Specify 9 parameter values for b in BM25 (0.1, 0.2, $\cdots$, 0.9). Plot the performance curve using MAP(y-axis), against the parameter values(x-axis). Report the best performance of your function (including the parameter value).