Relevance Feedback & Query Expansion

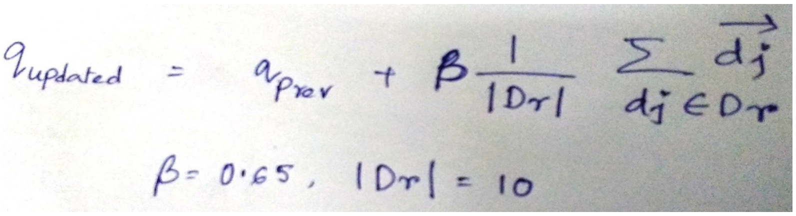
Dataset Description:

Use same dataset given to you for Assignment-1 where,
1. query.txt contains total 82 queries, which has 2 columns query id and query.
2. alldocs.rar contains documents file named with doc id. Each document has set of sentences.
3. output.txt contains 50 relevant documents (doc id) for each query

The link is here- https://drive.google.com/open?id=1l4gZR7f7GpffEPXqafabkrEGn512F7IJ

Task- 1 (Pseudo Relevance Feedback) :

1. Represent each query and document as tf-idf vector where the corpus will be all the documents in alldocs.rar merged.
2. For each query first retrieve top 50 documents using your pylucene/elasticsearch code implemented in Assignment 1. Report precision, recall, f-measure for each query (in a table format) as well as the average.
3. Now apply Rocchio' algorithm to update each query vector by considering top 10 retrieved documents as relevant ones. Follow the given equation only.



$$q_{updated} = q_{Prev} + \beta \frac{1}{|D_r|} \sum_{d_j \in D_r} \vec{d_j}$$

$$\beta = 0.65, \quad |D_r| = 10$$

4. Now from the updated query vector , pick up the top 10 term to obtain the updated query.
5. For updated query again retrieve top 50 documents using your pylucene/elasticsearch code implemented in Assignment 1. Report precision, recall, f-measure for each query (in a table format) as well as the average.

Task-2 (Query Expansion) :

1. A GloVe vector file (consider this as global knowledge) is provided where each line contains a word along with a 300 dimension vector. (https://drive.google.com/open?id=1FICPL4UzoeWJQimPUoS9qXoAtjrQIAEW)
2. Represent each query as a vector by adding the word vectors of the words present in the query
3. Now find top 5 similar (use cosine similarity) words with query vector from GloVe vector file. Use these 5 words to expand the already existing query.

4. For each expanded query retrieve top 50 documents using your pylucene/elasticsearch code implemented in Assignment 1. Report precision, recall, f-measure for each query (in a table format) as well as the average.

Deliverables:

Task 1:

1. Code file for step 1 which computes tf-idf vector representation of a given query/document
2. Text file named “Performance_before_relevance_feedback” containing the precision/recall/f-measure computed in step 2
3. Code file for implementing Rocchio’ algorithm which produces updated query vector given initial query vector and 10 relevant document vectors
4. Text file named “Performance_after_relevance_feedback” containing the precision/recall/f-measure computed in step 5

   NOTE: All these 4 files should be kept in a directory named “Task1_deliverables”

Task 2:

1. Text file named “query_vector” containing the query vector computed in step 2
2. Text file named “Expanded query” containing both the query before expansion and after expansion in a table format.
3. Text file named “Performance_after_query_expansion” containing the precision/recall/f-measure computed in step 4

   NOTE: All these 3 files should be kept in a directory named “Task2_deliverables”

NOTE: Keep both the “Task1_deliverables” and “Task2_deliverables” directories under “IR_Assignment_3_<Roll Number>” directory; create a zip file; upload

Warning: Strictly follow the naming conventions for the deliverables, if specified.