Part A - Task 2

TF-IDF vectorization and Evaluation

Group No.- 14

Group Members	**Roll No.**
Ajay Kumar Meena	18CS30006
Kushal Natani	18CS30025
Yash Fulzele	18CS10058
Mudit Gupta	18CY20019

Task 2A (TF-IDF Vectorization)

In this part we loaded the Inverted Index built in Part A - Task in invertedIndex dictionary in which key is term and value is a list of docs with the frequency of term in the doc.
Now, preprocess the data and parse the preprocessed queries using the query_14.txt file.
The following steps were followed to find top 50 docs for each query in all 3 ddd.qqq schemes:

- Precompute normalization coefficients for all docs.
- Iterate over all queries for each of them first find TFIDF vector for the query.
- Now, iterate over all the docs and find score(cosine similarity) for (doc,query) pair and store (score,docno) in a list.
- Now sort the doc score and return top50 docs.
- Finally save the top50 docs for each query as {queryID : docno} in a csv file.

Task 2B:

The highly relevant documents are more useful than moderately relevant documents, which are in turn more useful than irrelevant documents. Every recommendation has a relevance score associated with it. Cumulative Gain is the sum of all the relevance scores in a recommendation set. Discounted Cumulative Gain (DCG) is the metric of measuring ranking quality. It is mostly used in information retrieval problems such as measuring the effectiveness of the search engine algorithm by ranking the articles it displays according to their relevance in terms of the search keyword.
The same logic is implemented in “PAT2_14_evaluator.py” as taught in the class for both the average precision and NDCG.

Note: For task 2A, we also need to use the path to the pre-processed query file <path_to_queries_14.txt> as 3rd argument while running PAT2_14_ranker.py. For more details check running steps.

Python version: Python 3.8.10
Libraries Required:
1. nltk: pip3 install nltk

Running Steps:
$>> python3 PAT2_14_ranker.py <path_to_the_en_BDNews24 folder> <path_to_model_queries_14.pth> <path_to_queries_14.txt>
$>> python3 PAT2_14_evaluator.py <path_to_gold_standard_ranked_list.csv> <path_to_PAT2_14_ranked_list_A.csv>
$>> python3 PAT2_14_evaluator.py <path_to_gold_standard_ranked_list.csv> <path_to_PAT2_14_ranked_list_B.csv>
$>> python3 PAT2_14_evaluator.py <path_to_gold_standard_ranked_list.csv> <path_to_PAT2_14_ranked_list_C.csv>