README FILE

NAME: Sadvik Kondadi

Student ID: 11785837

- directory: dictionary.txt, forward_index.txt, inverted_index.txt, stopwordlist.txt, topic s.txt, main.qrels.
- Then compile the query processor using:
- **javac QueryProcessor.java.**
- Next run the system in title-only mode using:
- **java QueryProcessor title vsm_output_title.txt.**
- Next run the system in title+description mode using:
- **java QueryProcessor titledesc vsm_output_titledesc.txt.**
- Then run the system in title+narrative mode using:
- **java QueryProcessor titlenarr vsm_output_titlenarr.txt.**
- Then read and parse text from the selected query fields (title, description, narrative).
- Next convert query text to lowercase.
- Tokenize query text into individual terms.
- Then we need to remove stopwords using the provided stopword list.
- Apply Porter stemming to all remaining terms.
- For each query term, look up the termID in the dictionary.
- Retrieve the posting list for each termID from the inverted index.
- Then compute document TF values using (1 + log(tf)).
- Compute document IDF using log(N / df).
- Compute document TF–IDF weights using (1 + log(tf)) * log(N/df).
- Compute query TF–IDF weights using tf * log(N/df).
- Calculate cosine similarity by multiplying query weights with document weights.
- Accumulate similarity scores across all query terms for each document.
- Normalize scores using document vector lengths from the forward index.
- Sort documents in descending order of cosine similarity.
- Assign ranks starting from 1 for each query.
- Output each ranked result in the format: <queryID> <DOCNO> <rank> <score>.

- We need to save ranked results into the corresponding file (vsm_output_title.txt, vsm_output_titledesc.txt, or vsm_output_titlenarr.txt).
- Then compare system-retrieved results with relevance judgments in main.qrels.
- Compute precision by dividing relevant-retrieved documents by total retrieved documents.
- Compute recall by dividing relevant-retrieved documents by total relevant documents.
- At last we will evaluate and compare performance across title, title+description, and title+narrative query modes.