Assignment 1 Report

Our Algorithm Steps:

- We first pre-process the dataset.
    - For a particular dataset file:
        - Read the file and then pre-process (lowercase, removed punctuations, word-tokenize, removed stop words) the content.
        - Extract the tokens from the content and append them in the inverted index dictionary as a key and their document id as values (This is a linked-list of document ids in a sorted manner).
- Take the query as input and pre-process it as well. Then take operations as input from the user.
- Proceed only when, length of query = ((length of operations) + 1). Else raise ValueError.
- Extract the documents list for each token in a query. Then we just have to perform the operations on these documents list.
- We start evaluating the operations from Left to Right.
    - **NOTE**: To optimise the comparisons, we had created the clusters of the same consecutive operations and then evaluating those clusters in such a way that tokens with lesser document list will be evaluated first within the same cluster. This also gave operator precedence as NOT > AND > OR [Removed after the updated instructions for the assignment]
    - If operation == “OR”:
        - We used a two-pointer approach to find the OR between two lists and returned the output list and the total no. of comparisons.
    - If operation == “AND”:
        - We used a two-pointer approach to find the AND between two lists and returned the output list and the total no. of comparisons.
    - If operation == “OR NOT”:
        - To calculate X OR NOT Y, we can even perform the following to reduce the no. of comparisons:

X OR NOT Y = U - (Y - X)
where U is a Universal Set.
- So we calculate the U-(Y-X) instead of the “OR NOT” operation.

- If operation == “AND NOT”:
  - To calculate X AND NOT Y, we can even perform the following to reduce the no. of comparisons:
    X OR NOT Y = X - Y
  - So we calculate X-Y instead of the “AND NOT” operation.
- To perform the - (set difference) operator, we have implemented a two pointer-based system to keep the track of document id in each of the operands.

Our Assumptions:

For every comparison in the OR, AND, and SET DIFFERENCE operator, we have incremented +1; whether the expression is being satisfied or not, we have added +1. If the operation has +1 in comparison and the elements in else, it will have +2 as there is one more if statement inside of it.

The intuition behind that is, whether the condition is satisfied or not, the expression has still been evaluated.