Queensland University of Technology

CAB431 - Assignment 2

Report

Name : Jamie Martin (N10212361)

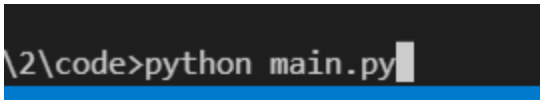
Due Date : 4th June 2021 [11.59 p.m]

Table of Contents

Statement of Completeness

Question	Complete (Y/N)
Q1	Y
Q2	Y
Q3	Y
Q4	Y
Q5	Y
Q6	Y
Q7	N

How to run



Run the `main.py` inside the `code` directory.

1.1 Algorithm

Inputs

Q

A set of documents *U* in the training set folder

Output

A text file *BLresultXX.dat*, where each row includes the document number and the corresponding relevance degree or ranking in descending order.

Method

Step 1

Let *docs* be an empty dictionary.

For each *file* in *U*:

Get the *ID*

Find the contents in `<*text*>`

Calculate the terms and their frequencies

Append *ID* and *term:freq* to *docs*

Step 2

Calculate each terms document frequency and save the results in a dictionary *df*

Step 3

Use query *Q*, *docs* and *df* to to calculate BM25 for each document in *U*

Step 4

Sort the document based on their BM25 score and save the result into a file *BLresultXX.dat*

2.4 Algorithm

Inputs

Outputs

Relevant features in the collection of documents

Methods

Step 1

Parse *benchmark* files into list

Step 2

For each *collection* in *collections*:

> Calculate w4 weights with *collection*, *benchmark* and *theta*
>
> Calculate rankings with *collection*, *weights*
>
> Sort rankings

Read Me

The file structure contains three directories that house files based on their purpose.

Code

The `code` directory holds the code. The entry point is `main.py`. This needs to be run inside the `code` directory.

stemming

The `stemming` package is utilised for word stemming.

benchmark.py

This file holds logic for accessing the `input` benchmarks.

BowDoc.py

This file holds logic for a single BowDoc.

BowDocCollection.py

This file holds logic for a BowDoc collection.

main.py

This file is the entrypoint for the application running the questions answered.

testing.py

This file holds logic for question 6 testing.

topic.py

This file holds logic for getting and parsing the topic_statements and collections from the `input` directory.

training.py

This file holds logic for calculating the Baseline Model and Learning for Ranking Model, bm25 and w4 respectively. It also holds logic for writing these to files.

utils.py

This file holds logic for getting the common english words which are used across the application.

XMLParser.py

This file holds logic for parsing XML documents.

Input

The `input` directory holds the input files for the calculations. These files are from Blackboard.

dataset101-150

Dataset files from Blackboard.

topicassignment

Training files from Blackboard.

common-english-words.txt

Common English stopping words from Blackboard.

TopicStatement101-150.txt

Topic statements file from Blackboard.

Output

The `output` directory holds the output files from the calculations. These files are calculated and overridden when the code is run.

BLresults

BL results from question 3.

LRresults

LR results from question 5.

Training

Benchmark results from question 3, for comparison.

Eresult1.dat

Evaluation results from LR results.

Eresult2.dat

Evaluation results from BL results.