IFN647 ASSIGNMENT 2

Final Report

[Priya Gunda] [N11362189]

[Tahura Naseer] [N11]

[Meetkumar Bhanderi] [N11]

Table of Contents

[Statement of completeness 1](#_Toc165648411)

[User Manual 2](#_Toc165648412)

[• Information on used packages 2](#_Toc165648413)

[• Data folder setup 2](#_Toc165648414)

[• How to execute Python files 2](#_Toc165648415)

[• Expected outputs for each Python file 2](#_Toc165648416)

[• Additional information (if any) 2](#_Toc165648417)

[Design 3](#_Toc165648418)

[o Description of your model for Task 1 3](#_Toc165648419)

[o Assumption (if any) 3](#_Toc165648420)

[o Algorithms (including inputs, outputs, and any other parameters) 3](#_Toc165648421)

[o Description of your model for Task 2 3](#_Toc165648422)

[o Assumption (if any) 3](#_Toc165648423)

[o Algorithms (including inputs, outputs, and any other parameters) 3](#_Toc165648424)

[o Description of your model for Task 3 3](#_Toc165648425)

[o Assumption (if any) 3](#_Toc165648426)

[o Algorithms (including inputs, outputs, and any other parameters) 3](#_Toc165648427)

[Implementation 4](#_Toc165648428)

[Model1: 4](#_Toc165648429)

[o Data structures (used to represent a single document and a set of documents): 4](#_Toc165648430)

[Model2: 4](#_Toc165648431)

[o Data structures (used to represent a single document and a set of documents): 4](#_Toc165648432)

[Model3: 4](#_Toc165648433)

[o Data structures (used to represent a single document and a set of documents): 4](#_Toc165648434)

[Results & Evaluation 5](#_Toc165648435)

[Discussion 6](#_Toc165648436)

[• Analysis about the difference between three models (Significance test) 6](#_Toc165648437)

[• Analysis about your findings and/or Recommendation (A justification must be included on your findings or recommendation) 6](#_Toc165648438)

[References 7](#_Toc165648439)

[Appendix 1 8](#_Toc165648440)

[Appendix 2 9](#_Toc165648441)

[Appendix 3 10](#_Toc165648442)

# Statement of completeness

|  |  |  |
| --- | --- | --- |
| **Statement of completeness:**  **Group Number:** | | |
| The following undersigned members of the group agree to abide by this statement to ensure successful completion of the project (Assignment 2) to meet project requirements and timelines. We declare that each team member has a same or similar contribution to the project. | | |
| **Name, student number & email** | **Signature** | **Date** |
| 1. ***Priya Gunda, N11362189,*** [***n11362189@qut.edu.au***](mailto:n11362189@qut.edu.au) | *Priya Gunda* |  |
|  |  |  |
|  |  |  |
| 1. Only if permitted |  |  |
| ***Task Allocation for each student***   1. *Priya Gunda - Task 1, Task 4* 2. *Tahura Naseer – Task 2* 3. *Meetkumar Bhanderi – Task 3* | ***Allocation Percentage* (%)** | |
| ***Other issues or comments*** | | |

# User Manual

## Information on used packages

## Data folder setup

## How to execute Python files

## Expected outputs for each Python file

## Additional information (if any)

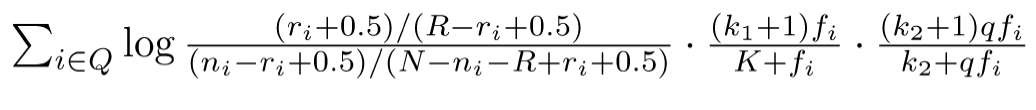
# Design

**Task 1:** Design a BM25-based IR model (***BM25***) that ranks documents in each data collection using the corresponding topic (query) for all 50 data collections.

### Description of your model for Task 1

The BM25-based IR model (***BM25***) is designed to rank documents within each data collection based on their relevance to the corresponding query string. This model uses BM25 scoring function, which considers various factors such as word frequency in the collection, document length, word frequency in the query to calculate a ranking score for each document.

### Assumption (if any)



The above equation is scoring function used to calculate the rank for each document. Here R is the number of relevance documents for a query and ri is the number of relevant documents containing term/word “i”. Since no relevance information is provided, we shall assume values of R, ri to be 0.

### Algorithms (including inputs, outputs, and any other parameters)

Input:

1. 50 long queries(topics) in a text file “the50Queries.txt”
2. 50 Data collections in the folder “Data\_Collection”, and documents for each collection are stored in Data\_C101, Data\_C102… DataC150 folders respectively.

Output:

1. Ranking scores for each collection are stored in a ranking document file named “BM25\_R1\*\*Ranking.dat” file. All the ranking documents are stored in the folder “RankingOutputs”.

Parameters:

1. k1, k2 and K are parameters whose values are set empirically.
2. k1 = 1.2
3. k2 = 500
4. b = 0.75
5. K = k\_1\*((1 - b) + b\*(dl/avdl))
6. dl - doc length
7. avdl - average length of a document in the collection
8. R - total number of relevant documents for the query
9. N - total number of documents in the collection
10. fi - term frequency of term ‘i’ in the document
11. qfi - term frequency of term ‘i’ in the query
12. ni - number of documents containing term
13. N - number of relevant documents containing term

Algorithm:

1. Read the 50 long queries from “the50Queries.txt” file and storing them in a dictionary where key is the data collection number and value is the query string.
2. For each query

Compute the query feature function

For each document in the corresponding data collection

Compute the document feature function

Calculate the BM25 score for document using the provided formula

Store BM25 score in a dictionary bm25\_scores, where key is the docID and value is the score

End For

Sort bm25\_scores dictionary in descending order

Save the ranked document list scored in bm25\_scores in ‘RankingOutputs’ folder with the appropriate filename.

1. Repeat step b for all 50 queries.

**Task 2:** Design a Jelinek-Mercer based Language Model (***JM\_LM***) that ranks documents in each data collection using the corresponding topic (query) for all 50 data collections.

### Description of your model for Task 2

### Assumption (if any)

### Algorithms (including inputs, outputs, and any other parameters)

**Task 3.** Based on the knowledge you gained from this unit, design a pseudo-relevance model (***My\_PRM***) to rank documents in each data collection using the corresponding topic (query) for all 50 data collections.

### Description of your model for Task 3

### Assumption (if any)

### Algorithms (including inputs, outputs, and any other parameters)

# Implementation

**Task 4.** Use Python to implement three models: ***BM25***, ***JM\_LM*** and ***My\_PRM***, and test them on the given 50 data collections for the corresponding 50 queries (topics).

## Model1:

* Python package or module (or any open-source software) you used:

## Data structures (used to represent a single document and a set of documents):

## Model2:

* Python package or module (or any open-source software) you used:

## Data structures (used to represent a single document and a set of documents):

## Model3:

* Python package or module (or any open-source software) you used:

## Data structures (used to represent a single document and a set of documents):

# Results & Evaluation

**Task 5.** Use three effectiveness measures to evaluate the three models.

Table 1: The performance of 3 models on average precison (MAP)

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Model1** | **Model2** | **Model3** |
| R101 |  |  |  |
| R102 |  |  |  |
| R103 |  |  |  |
| … |  |  |  |
| R150 |  |  |  |
| MAP |  |  |  |

Table 2: The performance of 3 models on precision@10

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Model1** | **Model2** | **Model3** |
| R101 |  |  |  |
| R102 |  |  |  |
| R103 |  |  |  |
| … |  |  |  |
| R150 |  |  |  |
| Average |  |  |  |

Table 3: The performance of 3 models on DCG10

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Model1** | **Model2** | **Model3** |
| R101 |  |  |  |
| R102 |  |  |  |
| R103 |  |  |  |
| … |  |  |  |
| R150 |  |  |  |
| Average |  |  |  |

# Discussion

**Task 6.** Recommend a model based on significance test and your analysis.

## Analysis about the difference between three models (Significance test)

## Analysis about your findings and/or Recommendation (A justification must be included on your findings or recommendation)

# References

Appendix 1 (listing the top-15 documents for all queries for Model1)

Appendix 2 (listing the top-15 documents for all queries for Model2)

Appendix 3 (listing the top-15 documents for all queries for Model3)