# PART 1

## Queries

The listed queries are the ones that have been utilized throughout the assignment. The judgments provided by three different judges for each query are presented in tabular format.

| Query | Judge 1 | Judge 2 | Judge 3 |
|---|---|---|---|
| constipation | 203,251,372 | 203,372 | 203,251,372,442 |
| influenza | 59 | 59,469 | 59,71,469 |
| 'cough' | 372,421 | 372,83,126 | 372,83,126,421 |
| muscle ache | 37,65,76,126,343 | 37,65,76,117,126,387 | 37,65,76,83,100,117,126,206,277,343 |
| vomiting | 203,442 | 203,372,442 | 203,372,442 |
| hepatitis | 16,376 | 16,376 | 16,368,376 |
| diarrhea | 372,442 | 372,203,372,442 | 372,203,372,442 |
| cold | 83,126,421 | 82,83,126,421 | 41,82,83,126 |
| paracetamol ibuprofen | 216,200 | 216,200,365,356 | 216,200,205,356,214 |
| common cold | 83,126,421,372,28,365 | 82,83,126,421,372,28 | 41,82,83,126 |
| paracetamol | 28,214,364,83,126,421,372,28,365 | 28,214,364,421,372,28 | 28,214,364,41,82,83,126 |
| ibuprofen | 200,139,216 | 200,356,139,216 | 200,205,216 |
| fever | 28,214,365,251 | 28,214,365,251 | 28,214,365 |
| virus | 275,372,399,402,403,59,71,371 | 275,372,399,400,402,403,475,59,71 | 275,372,400,403,475,71,371 |

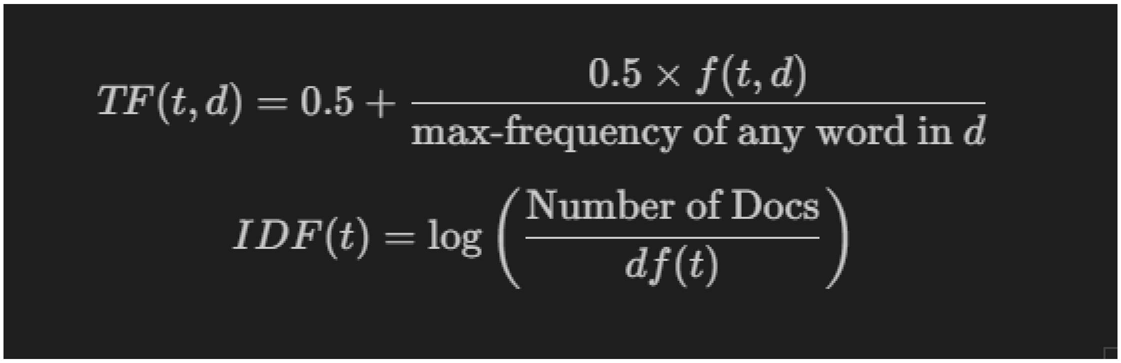
| | | | |
|---|---|---|---|
| hepatitis virus | 368,376,16,401 | 368,376,16,399, 401,402 | 368,376,16,372,399, 402 |
| sore throat | 477,483,71,177,214, 252 | 477,483,71,177,198, 214 | 477,71,177,198 |
| headache | 216,421,214,28,83, 126,421,372,28,365, 61,69,71 | 216,421,214,365,28, 82,83,126,421,372, 28,61,62,69,71 | 216,421,365,28,41, 82,83,126,61,62,71 |
| back pain | 378,162,301,235,413 ,356 | 378,162,235,301,235 ,348,356 | 378,235,301,235,356 |
| stomach ache | 391,372,442 | 391,200,372,203,372 ,442 | 391,200,372,203,372 ,442 |
| nausea | 368,442,203,442 | 368,442,203,372,442 | 368,442,203,372,442 |

# PART 2

## Assumptions

## TF-IDF

The Augmented TF-IDF model has been employed for the given task, which is a variant of the standard TF-IDF model and takes into account the document length. The formula for the Augmented TF-IDF model is given below.

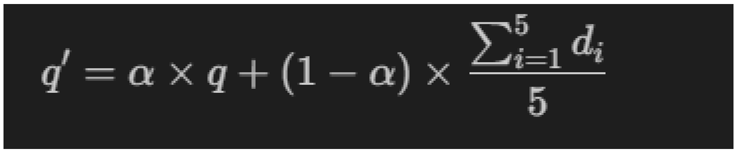


$$TF(t,d) = 0.5 + \frac{0.5 \times f(t,d)}{\text{max-frequency of any word in } d}$$

$$IDF(t) = \log\left(\frac{\text{Number of Docs}}{df(t)}\right)$$

## Pseudo Relevence Feedback

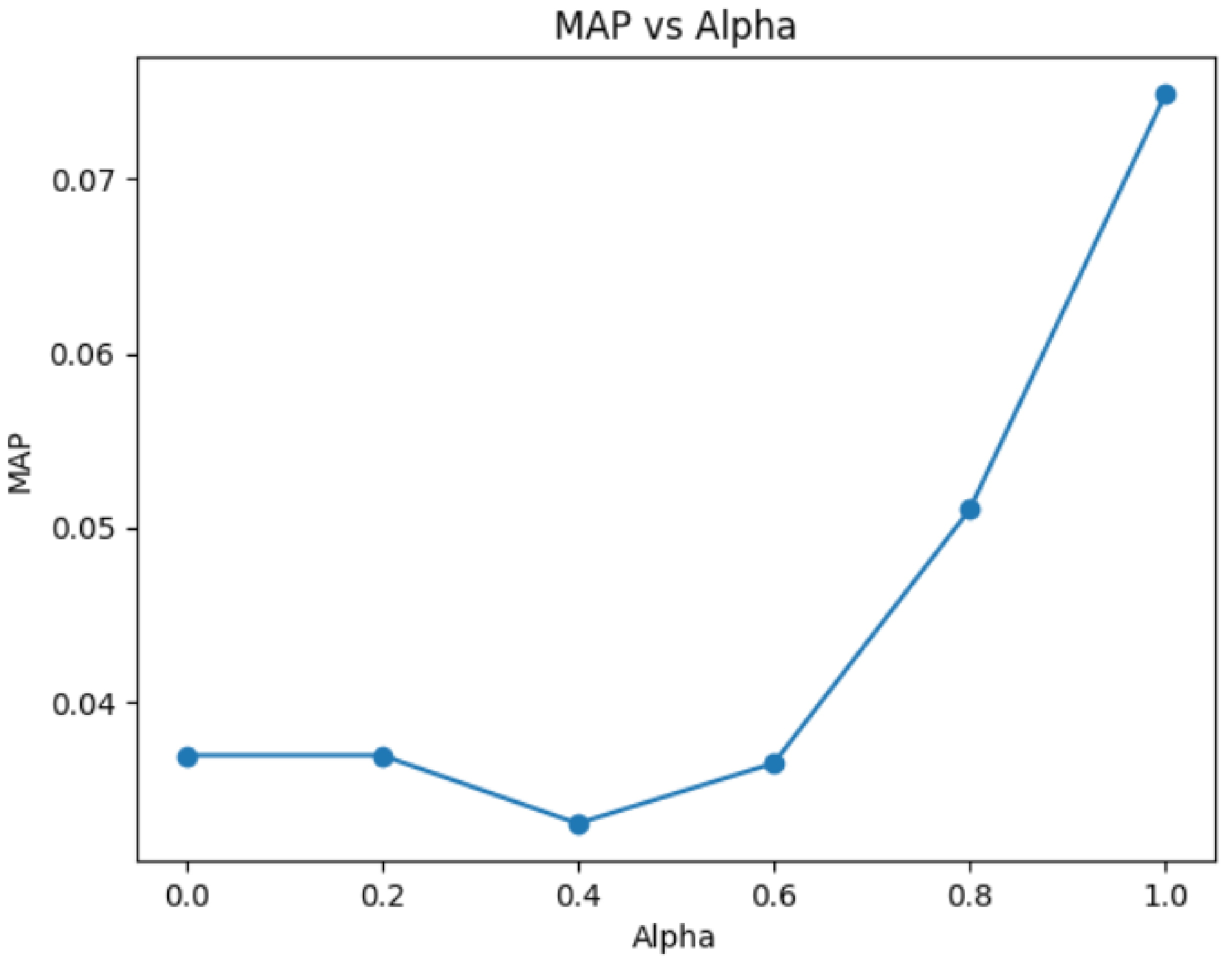
In the pseudo relevance feedback technique, it is assumed that the top 5 documents retrieved are relevant to the query. The query is then updated using the provided equation.



$$q' = \alpha \times q + (1 - \alpha) \times \frac{\sum_{i=1}^{5} d_i}{5}$$

### Result

After comparing between the Mean Average Precision (MAP) and the Model in Part-1b, it can be stated that the performance of the Information Retrieval (IR) engine has been enhanced. It has been observed that the value of the parameter 'α' that maximizes MAP is α = 1.



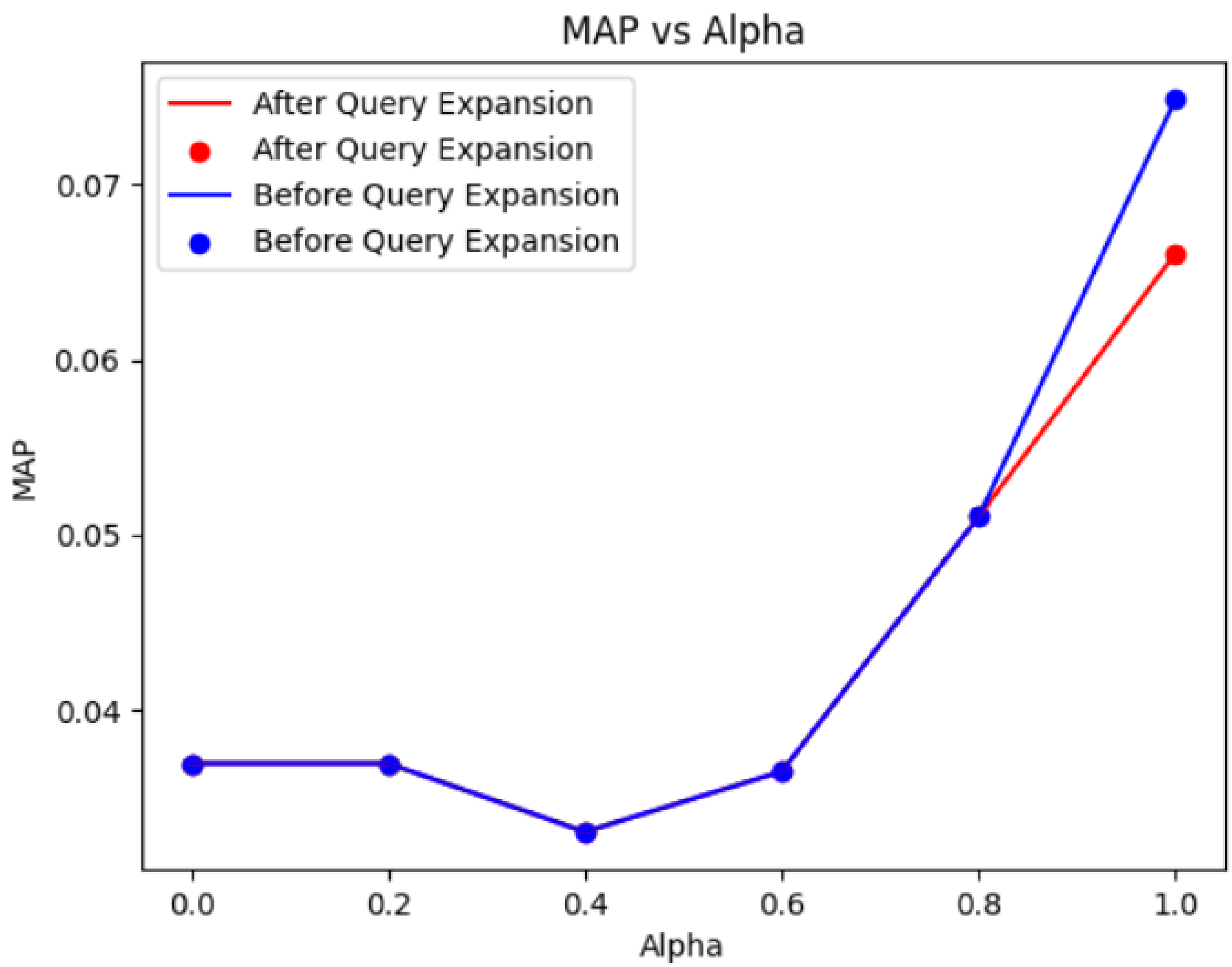
# PART 3

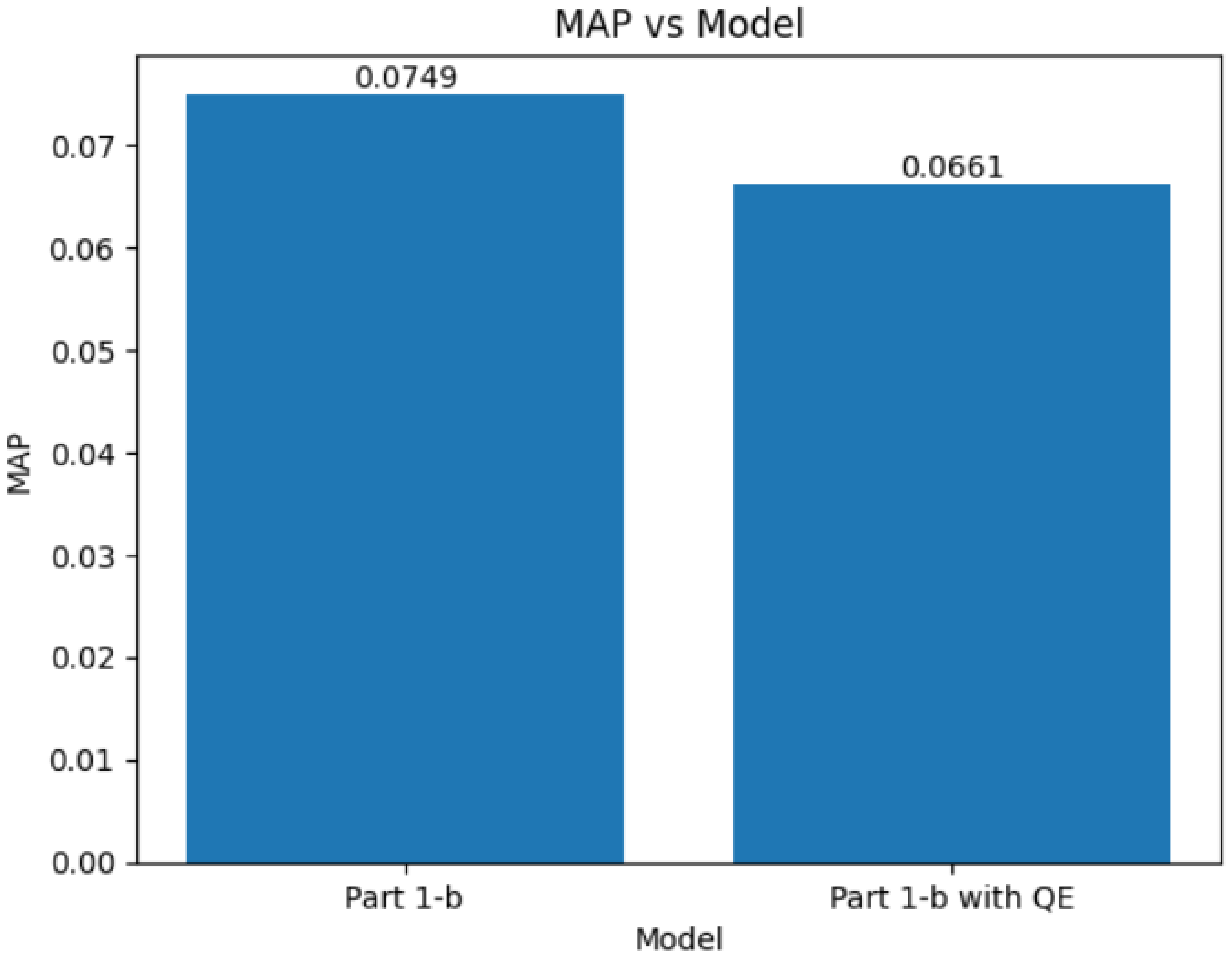
## Assumptions

(I)

Use query expansion technique by including up to two synonyms for the nouns and verbs contained within the query. Acquire the synonyms from the Natural Language Toolkit (NLTK) Wordnet database Compare the performance with the models in Part 1-b and Part 2.

## Results

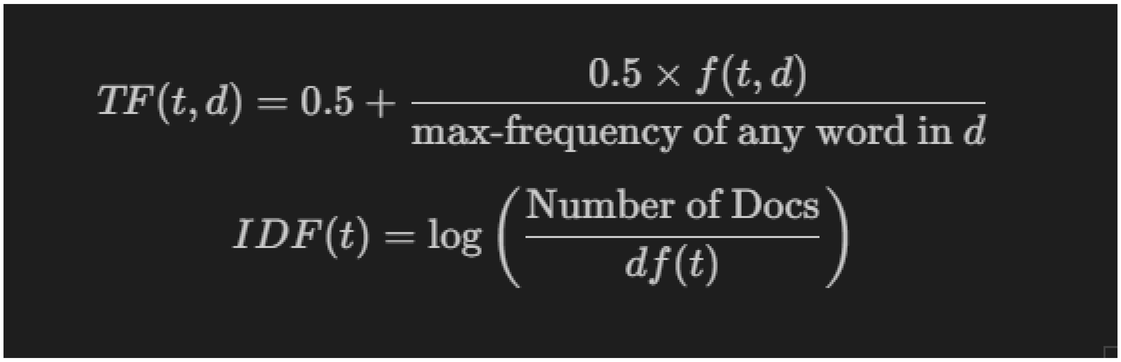




(II)

## TF-IDF

The Augmented TF-IDF model has been employed for the given task, which is a variant of the standard TF-IDF model and takes into account the document length. The formula for the Augmented TF-IDF model is given below.



$$TF(t,d) = 0.5 + \frac{0.5 \times f(t,d)}{\text{max-frequency of any word in } d}$$

$$IDF(t) = \log\left(\frac{\text{Number of Docs}}{df(t)}\right)$$

(III)

## BM25 - Okapi

Used a library for this.

(IV)

## Language Model

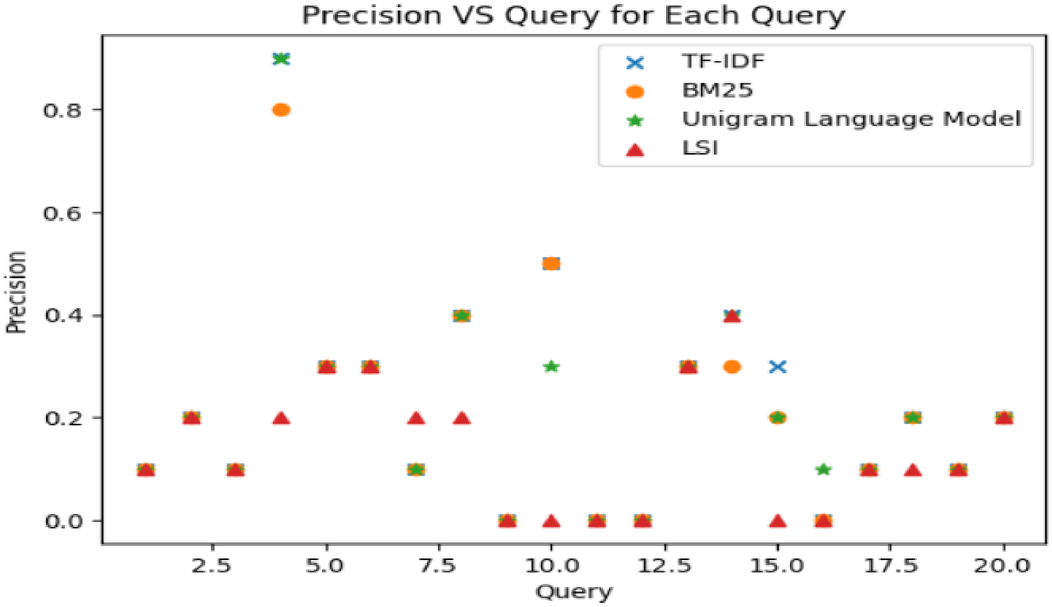
For the Language Model, we will use the Unigram Language Model.

(V)

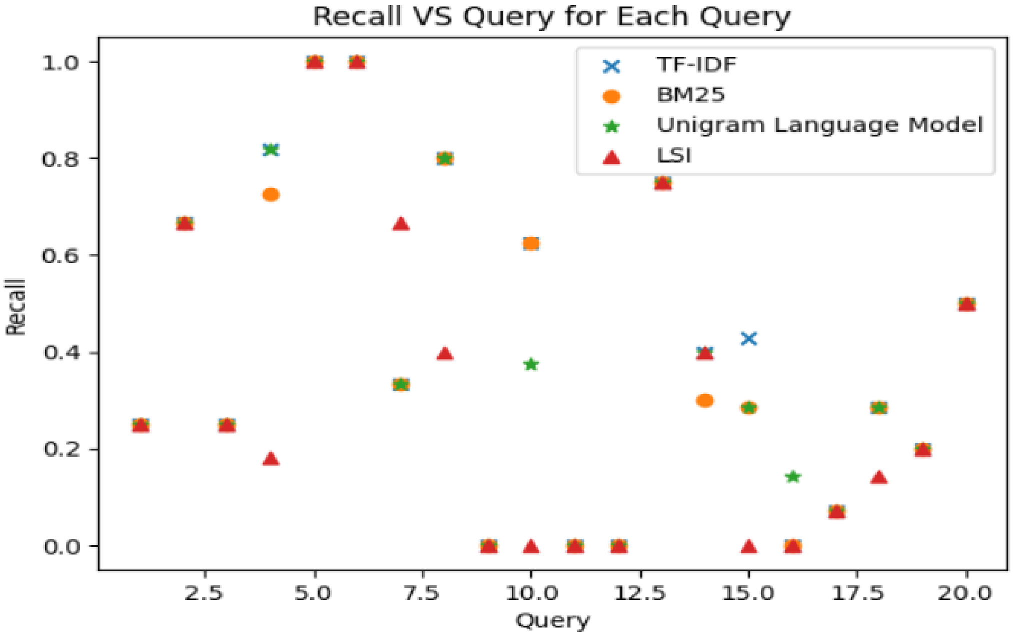
## LSI

In LSI we will use 50 latent dimensions for terms and documents.

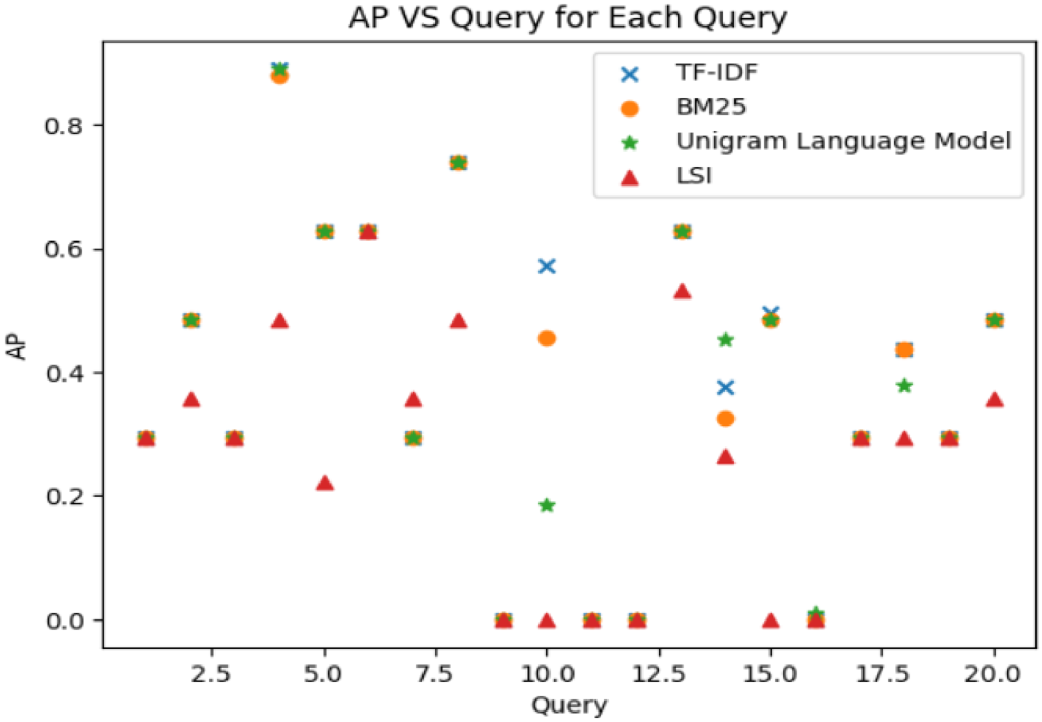
Precision vs Query



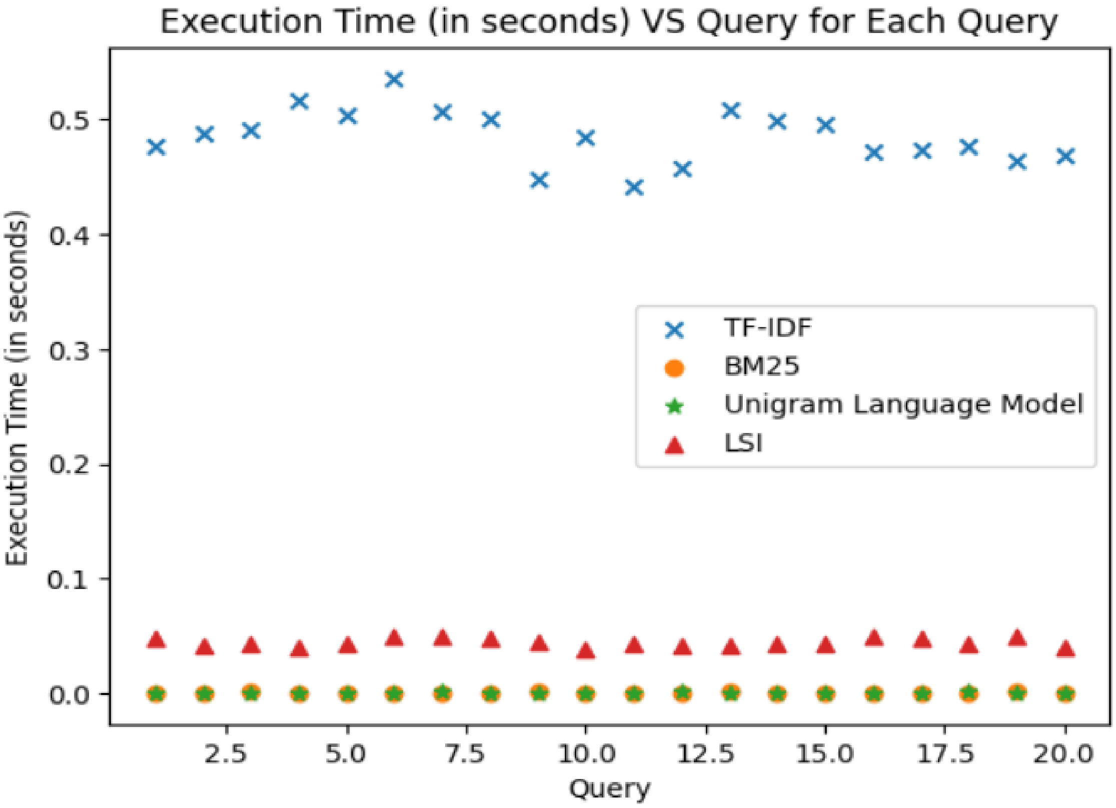
Recall vs Query



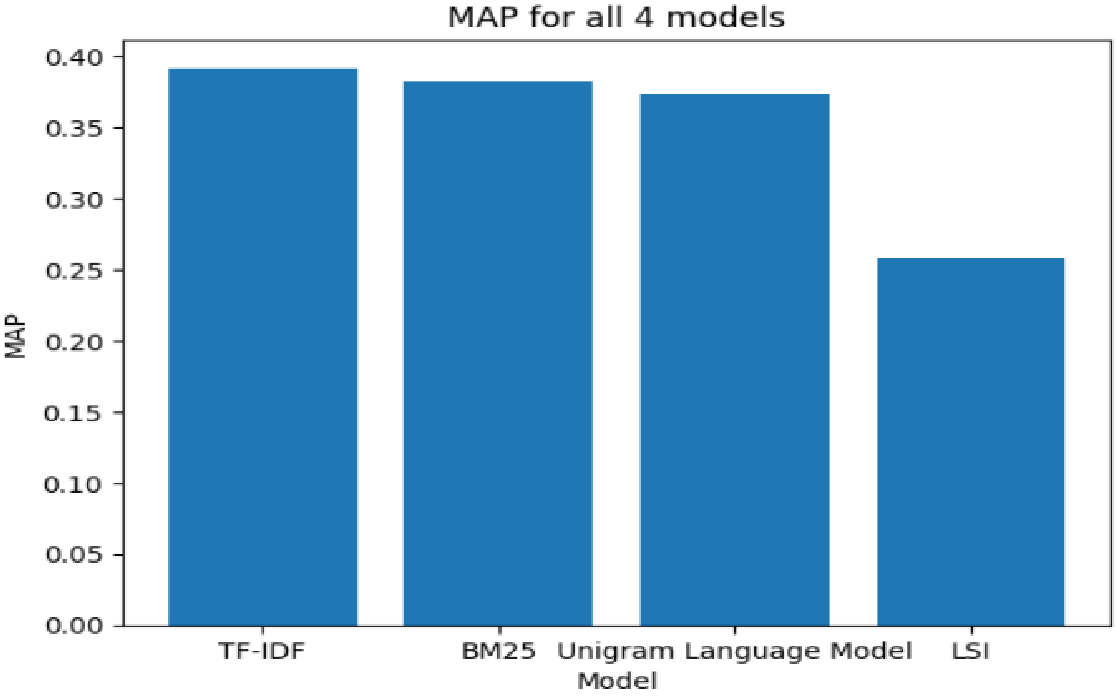
AP vs Query



Execution Time (in seconds) vs Query


Execution Time (in seconds) VS Query for Each Query
Execution Time (in seconds)
0.0
0.1
0.2
0.3
0.4
0.5
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Query
TF-IDF
BM25
Unigram Language Model
LSI

MAP for all 4 models



# Part 4

## Assumptions

The metric for similarity and distance being used in the clustering algorithms are as follows:

Jaccard similarity coefficient :

$$J(A,B) = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Jaccard similarity coefficient.

Jaccard distance :

$$d_j(A,B) = 1 - J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

Jaccard distance.

Assumption applied for K-means clustering:

After random allocation of seeds, the clusters are formed by comparing an elements jaccard similarity with the seed and accordingly the cluster is formed.

The metric being used to find new centroid is such that the sum of the jaccard distances of all other elements in the cluster is the least one.

This is repeated for multiple rounds.

Additional point : Along with the calculation of RSS, we are also penalising when number of clusters using the factor of ‘ lambda*k ’. This gives a more refined measure.

When k = 4, first major elbow is seen in the plot, thus we choose k=4 as the ideal value of k.

# Part - 5

## Assumptions

All the 3 basis of HAC uses the same jaccard coefficient and jaccard distance metric as a measure for similarity and distance respectively.

Since the ideal value of k was found to be four, dendogram was cut in order to have 4 clusters each.

Purity for each cluster was found using the formulae :

$$Purity(\omega_i) = \frac{1}{n_i} \max{}_j (n_{ij}) \quad j \in C$$

NMI was found as follows :

First the entropy of Y (class labels ) and cluster is calculated using the entropy formula :

$$H(p) = -\sum_i p_i \log_2 (p_i)$$

For Y, p_i  is the probability that a class label is ‘i’ and for C it is the probability that element is in C_i.

Then for Mutual Information:

$$I(Y;C) = H(Y) - H(Y|C)$$

H(Y|C) was calculated by iterating over every cluster in the clustering, for example for cluster 1 :

$$H(Y|C=1) = -P(C=1) \sum_{y \in \{1,2,3\}} P(Y=y|C=1) \log(P(Y=y|C=1))$$

After which all of them were added up to find H(Y|C).

Finally NMI is calculated as :

$$NMI(Y,C) = \frac{2 \times I(Y;C)}{[H(Y) + H(C)]}$$

For 5.‘b’ it is mentioned that documents relevant to each query should ideally belong to a separate cluster.To achieve this first pairs of such different documents were formed and then it was checked in each of the 4 clusterings whether these pairs were in the same cluster or not. If they were in the same cluster, score was incremented, thus higher the score worse the models as per the annotated documents. According to our analysis, complete link and centroid based HAC performed the best.