Information Retrieval Assignment 1 -

Name - TAPAN AUTI

Class - 1MAI1

Student ID - 20231499

Question 1 -

D1:Shipment of gold damaged in a fire

D2:Delivery of silver arrived in a silver truck

D3:Shipment of gold arrived in a truck

Q:gold silver truck

The total documents are 3 hence d = 3 for every document in the word I have assumed to normalisation of words to lower case and alphabetical order.

The total words in the document are 11 and they are sorted alphabetically as below -

a, arrived, damaged, delivery, fire, gold, in, of, silver, shipment, truck

The idf value for each is calculated by the log (N/Ni) where N - the total no of documents and Ni - documents containing the word.

ex - for 'gold' the idf will be log (3/2) = 0.176 ,

similarly the idf values of all 11 words are -

idf a = 0

idf arrived = 0.176

idf delivery = 0.477

idf fire = 0.477

idf gold = 0.176

idf in = 0

idf of = 0

idf silver = 0.477

idf shipment = 0.176

idf truck = 0.176

Similarly we can calculate the tf values for each word in sentences which is nothing but the no of occurrences of in that sentence

tf for each word in document

Document	a	arrived	damaged	delivery	fire	gold	in	of	shipment	silver	truck
Q	0	0	0	0	0	1	0	0	0	1	1
D1	1	0	1	0	1	1	1	0	1	0	0
D2	0	1	0	1	0	0	0	0	0	2	1
D3	0	1	0	0	0	1	0	0	1	0	1

Similarly we can plot the tf-idf document vector which is given by formula tf * idf for each word.

tf-idf document vector

Document	a	arrived	damaged	delivery	fire	gold	in	of	shipment	silver	truck
Q	0	0	0	0	0	0.176	0	0	0	0.477	0.176
D1	0	0	0.477	0	0.477	0.176	0	0	0.176	0	0
D2	0	0.176	0	0.477	0	0	0	0	0	0.954	0.176
D3	0	0.176	0	0	0	0.176	0	0	0.176	0	0.176

We can use simple matching technique for finding the similarity between query and document.

$\text{sim}(Q,D1) = \sum (Qi * D1i)$

$= (0.176 * 0.176)$

$= 0.03$ ……. did not consider the words whose frequency is 0.

similarly

$\text{sim}(Q,D2) = (0.477 * 0.954) + (0.176 * 0.176)$

$= 0.486$

$\text{sim}(Q,D3) = (0.176 * 0.176) + (0.176 * 0.176)$

$= 0.062$

By the above calculations we can conclude that the ranking of documents base on the query would be like D2—D3—D1

Question 2 -

Let us first consider the (a) - D1 = Shipment of gold damaged in a fire. Fire.

In this document the change we have made is adding the fire after end, by doing this the term frequency of fire will change in turn affecting the tf-idf value for fire. But as we take a look in the query the word 'fire' does not exist so this change won't affect the similarity between the document D1 and the query Q, i.e. the similarity before and after augmentation would remain the same.

Similarly for scenario (b) - D1 = Shipment of gold damaged in a fire. Fire. Fire.

Here we are adding another word fire which is increasing the term frequency of fire in the document making it 3 in total that will definitely increase the tf-idf value in document vector but same as (a) won't make any difference in similarity as 'Fire' is not present in the query Q, i.e. the similarity before and after augmentation would remain the same.

For (c) - Shipment of gold damaged in a fire. Gold.

In this case the term frequency of gold will increase in the document and as the word is present in the query it will in turn increase the similarity quotient between Q and D1. We can say that the similarity weight will increase after augmentation and will be twice as much as the previous one.

For (d) - Shipment of gold damaged in a fire. Gold. Gold

In this case the term frequency of gold will increase and become 3 in the document and as the word is present in the query it will also increase the similarity degree between Q and D1. The similarity quotient will increase after augmentation and will be thrice as much as before augmentation. In this case we are not satisfying the fourth axiom constraint , as the term occurrences increase the factor is not sub-linear. For this we might then have to take into consideration some other weighting scheme such as include some normalization factor in it or vector length but that will change results as the length will increase with each increase in the term.

Question 3 -

As these are scientific documents they all have a title so I think foremost the query should be weighted and checked with the title of the document or article and then after that should be matched with the occurrences of terms from query with the weights of that terms in documents, for this I think just the term frequency will suffice.

As this being a scientific documents we can also take into consideration the author of the document so that we can match the query with that similarity another thing which can be used is the relevance feedback based on ehich we can retrieve the documents based on the information present with us.

All these documents belong to a specific class of documents like magazine, published paper or article. What we can do is that we can use mean of frequent terms like clustering method to classify the documents best suited based on the query and based on that we can add this mean of weighted term to the term factor which can definitely help to add aid in the weighting of documents.

For both the above features we can use tf or tf idf methods for execution.