Dry Part

Vector Space Model

1.

	df	idf $\ln(\frac{100}{df})$	d1 tf (raw)	d1 tf×idf	d2 tf	d2 tf×idf
a	10%	ln(10) = 2.3	0	0	1	2.3
b	10%	2.3	1	2.3	2	4.6
c	20%	ln(5) = 1.6	2	3.2	0	0
d	5%	ln(20) = 3	0	0	0	0
e	50%	ln(2) = 0.7	1	0.7	0	0
f	90%	ln(1.111) = 0.1	7	0.7	5	0.5

$$Sim(d_1, d_2) = \frac{\vec{d_1} \cdot \vec{d_2}}{|d_1||d_2|}$$

$|d_1| = \sqrt{2.3^2 + 3.2^2 + 0.7^2 + 0.7^2} = 4.06$ $\quad |d_1||d_2| = 20.94$

$|d_2| = \sqrt{2.3^2 + 4.6^2 + 0.5^2} = 5.16$

$\vec{d_1} \cdot \vec{d_2} = 0 + 10.58 + 0 + 0 + 0 + 0.35$

$= 10.93$

$sim(d_1, d_2) = \frac{10.93}{20.94} = 0.52$

After calculating the idf score of each term, and the tf-idf score of each term in d_1, d_2 using raw tf, we are able to calculate the cosine-similarity of d_1, d_2, which is $Sim(d_1, d_2) = 0.52$

2.

	df	idf $\ln(\frac{100}{df})$	d_1 tf (raw)	d_2 tf	d_3 tf	d_4 tf	q tf	q tf×idf
a	10%	ln(10) = 2.3	0	1	1	1	1	2.3
b	10%	2.3	1	2	0	1	1	2.3
c	20%	ln(5) = 1.6	2	0	0	0	0	0
d	5%	ln(20) = 3	0	0	0	0	0	0
e	50%	ln(2) = 0.7	1	0	1	1	0	0
f	90%	ln(1.111) = 0.1	7	5	1	2	1	0.1

$|d_1| = 7.41$ $\quad |d_1||q| = 24.11$

$|d_2| = 5.47$ $\quad |d_2||q| = 17.77$

$|d_3| = 7.14$ $\quad |d_3||q| = 23.2$

$|d_4| = 2.64$ $\quad |d_4||q| = 8.58$

$|q| = 3.25$

$\vec{d_1} \cdot \vec{q} = 0 + 2.3 + 0 + 0 + 0 + 0.7 = 3$

$\vec{d_2} \cdot \vec{q} = 2.3 + 4.6 + 0 + 0 + 0 + 0.5 = 7.4$

$\vec{d_3} \cdot \vec{q} = 2.3 + 0 + 0 + 0 + 0 + 0.7 = 3$

$\vec{d_4} \cdot \vec{q} = 2.3 + 2.3 + 0 + 0 + 0 + 0.2 = 4.8$

$sim(d_1, q) = \frac{3}{24.11} = 0.12$

$sim(d_2, q) = \frac{7.4}{17.77} = 0.416$

$sim(d_3, q) = \frac{3}{23.2} = 0.13$

$sim(d_4, q) = \frac{4.8}{8.58} = 0.56$

Ranking:

d_4, d_2, d_3, d_1

3. Let $\vec{v_1}, \vec{v_2}, \vec{u}$ be normalized vectors, meaning $|v_1|, |v_2|, |u| = 1$. We would like to show that, $d(v_1, u) > d(v_2, u)$ iff $sim(v_1, u) < sim(v_2, u)$, where d is the Euclidean distance and sim is cosine similarity.

For any $v \in \{v_1, v_2\}$:

$$d(v,u) = \sqrt{\sum_i (v_i - u_i)^2} \geq 0,$$

$$d(v,u)^2 = \sum_i (v_i - u_i)^2 = \sum_i v_i^2 - 2 v_i u_i + u_i^2 =$$

$$\sum_i v_i^2 + \sum_i u_i^2 - 2\sum_i v_i u_i = 1 + 1 - 2\, dot(v,u) =$$

$$= 2 - 2\, dot(v,u)$$

normalized vectors

Additionally,

(*) $$sim(v,u) = \frac{dot(v,u)}{|v| \cdot |u|} = dot(v,u)$$

normalized vectors

Therefore, lets assume w.l.o.g, that $d(v_1,u) > d(v_2,u)$:

$$2 - 2\, dot(v_1,u) > 2 - 2\, dot(v_2,u) \quad / -2$$

$$-2\, dot(v_1,u) > -2\, dot(v_2,u) \quad / : -2$$

$$dot(v_1,u) < dot(v_2,u)$$

$\Updownarrow$ (*)

$$sim(v_1,u) < sim(v_2,u)$$

All the performed actions are reversible, which will show that if $sim(v_1,u) < sim(v_2,u)$ then

$d(v_1, u) > d(v_2, u)$. So, we have found that $d(v_1, u) > d(v_2, u)$ iff $sim(v_1, u) < sim(v_2, u)$

Thus, an ordering determined by the Euclidean distance will be the same as an ordering determined by cosine similarity. □

Term weighting

1. cosine similarity is computed as follows:

$$sim(v, u) = \frac{\vec{v} \cdot \vec{u}}{|v| \cdot |u|}$$

The similarity is negatively affected by the length of the document, since longer documents will have a larger magnitude, which means we divide by a greater number, lowering the similarity.
whereas in smaller documents, we will have a smaller magnitud, resulting in higher similarity and a bias.

2.

$$W_{t,d} = \alpha + (1 - \alpha) \frac{tf_{t,d}}{tf_{max(d)}} \qquad \alpha \in [0, 1]$$

The function maps all term frequencies to $[\alpha, 1]$, this is useful to prevent highly frequent terms from having a disproportionally large weight. (similarly to $wf = 1 + \ln(tf)$).

But, a large issue with this function is that any term, even ones with $tf = 0$, will have a $w_{t,d}$ of at least α, so searching for a term that is not present in some document may return that document. Furthermore, querying a very large number of terms will increase the chance of a non-related document to return.

Dry Part

Relevance feedback and evaluation

Question 1

AP doesn't differentiate between gradual graded relevance, therefore a score of 4 is the same as a score of 3 and a score of 1. The relevant documents are doc5, doc2, doc1 and doc3.

$$P_{@5} = \frac{4}{5}$$

$$R_{@5} = \frac{4}{10}$$

$$AP_{@5} = \frac{1}{10} \times \left(1 + \frac{2}{2} + \frac{3}{3} + \frac{4}{4}\right) = \frac{2}{5} = 0.4$$

Question 2

If the graded relevance judgements use a scale from 0 to 4, then the centroid of non-relevant documents is the same since non-relevant documents will get a grade of 0. However, the centroid of relevant documents does change because now the model should move closer to the centroid of relevant documents when the grade is 4 and less close to the centroid of relevant documents when the grade is 1. Therefore, we multiply the centroid of relevant documents by the graded factor: $\frac{\text{grade}}{\text{maximal\_grade}}$. For example, if the document is graded as 4, then the factor will be $\frac{4}{4} = 1$ whereas a document graded as 1 will be $\frac{1}{4}$.

$$\overrightarrow{qm} = \alpha q_0 + \beta \frac{1}{|Dr|} \sum_{\forall dj \in Dr} \overrightarrow{dj} \times \frac{\text{grade}}{\text{maximal\_grade}} - \gamma \frac{1}{|Dnr|} \sum_{\forall dj \in Dnr} \overrightarrow{dj}$$

Hence, the essence of Rocchio's model is maintained, with higher graded documents getting closer to the centroid of relevant documents than lower graded documents.

Question 3

To integrate document ranking in Rocchio's model, we can add a weight variable that represents the document's rank:

$$\overrightarrow{qm} = \alpha q_0 + \beta \frac{1}{|Dr|} \sum_{\forall dj \in Dr} \overrightarrow{dj} \times \frac{1}{r_j} - \gamma \frac{1}{|Dnr|} \sum_{\forall dj \in Dnr} \overrightarrow{dj} \times \frac{1}{r_j}$$

Where r_j is the rank of the j-th document.

By adding weight of the ranking of the documents we can modify the feedback based on the importance of each document. A higher weight (higher rank) on a relevant document will push the new query vector closer to relevant documents whereas a smaller weight (smaller rank) on a relevant document will move the vector closely to the relevant documents but in a smaller measure than if it had been a higher ranked document. The ranking will also affect the centroid of the non-

relevant documents. A higher ranking in a non-relevant document will mean that the new query will be further away from such document.

Evaluation

Question 1

AP doesn't differentiate between gradual relevance scores. The precision parameter inside of AP is binary, so any documents graded between 1 to 4 will all acquire the same value of AP, since they are all consider relevant.

Let the graded relevance of a document be between 0 and 4. Documents graded 0 will always be considered as non-relevant. However, the goal is to obtain a different AP when modifying the relevance criteria (what grade will be considered as relevant and not relevant).

Consider the following algorithm that uses AP for gradual relevance scores:

1. Initialize variable min_score=0 and max_score = 4
2. Retrieve the list of all documents
3. While min_score < max_score:
    a. Grade relevant documents according to relevance criteria: "if grade > min_score then relevant_doc = True".
    b. Calculate AP
    c. min_score = min_score +1
4. Average all obtained AP scores

Consider the following two lists to test the algorithm:

List 1	List 2
4	1
4	1
1	4
1	4
0	0

These are the graded relevance lists obtained for list 1 when changing the relevance criteria (having R: relevant, N: non-relevant):

List 1 score	Min_score = 0	Min_score = 1	Min_score = 2	Min_score = 3
4	R	R	R	R
4	R	R	R	R
1	R	N	N	N
1	R	N	N	N
0	N	N	N	N
AP scores	$\mathrm{AP} = \frac{4}{5}$	$\mathrm{AP} = \frac{2}{5}$	$\mathrm{AP} = \frac{2}{5}$	$\mathrm{AP} = \frac{2}{5}$

Next, when calculating the AP average we obtain:

$$AP\_\text{avg}_{\text{L1}} = \frac{1}{4} \times \left(\frac{4}{5} + \frac{2}{5} + \frac{2}{5} + \frac{2}{5}\right) = \frac{1}{2} = 0.5$$

For list 2, the following are the graded relevance lists obtained when changing the relevance criteria (having R: relevant, N: non-relevant):

List 1 score	Min_score = 0	Min_score = 1	Min_score = 2	Min_score = 3
1	R	N	N	N
1	R	N	N	N
4	R	R	R	R
4	R	R	R	R
0	N	N	N	N
AP scores	$\text{AP} = \frac{4}{5}$	$\text{AP} = \frac{1}{6}$	$\text{AP} = \frac{1}{6}$	$\text{AP} = \frac{1}{6}$

$$AP\_\text{avg}_{\text{L2}} = \frac{1}{4} \times \left(\frac{4}{5} + \frac{1}{6} + \frac{1}{6} + \frac{1}{6}\right) = \frac{13}{40} = 0.325$$

Therefore,

$$AP\_\text{avg}_{L1} > \text{AP\_avg}_{L2}$$

The algorithm can correctly assign a higher AP score for the list that ranked first the higher documents.

Question 2

In general, MAP metric considers whether all relevant documents were ranked highly whereas MRR cares about the single highest-ranked relevant document. Hence, both metrics will return equal results in the following scenarios:

1. There is only one relevant document in the dataset. Both MAP and MRR will be equal to the precision of the system.
2. All documents are relevant, or all documents are non-relevant. Both MAP and MRR will be equal to the precision of the system. For example, if all documents are relevant, the precision of the system is equal to 1, MAP will be equal to 1 and MRR will also be equal to 1 since all documents will be considered as true positives.
3. All queries retrieve the same amount of relevant documents and the relevant documents are ranked at the top of the result list. Both MAP and MRR will yield 1.
4. All relevant documents are retrieved on the same ranking order for each query.

Question 3a

Recall is reduced if the amount of relevant retrieved documents is decreased. There are many stop-words libraries that identify different combinations of words as stop-words. Let's consider the following sentence: "She was dancing and not happy about it." Assuming that there exists a library that would consider the words "she, was, and, not, about, it" as stop-words, the final sentence would be "dancing happy". Hence, the system would retrieve documents were there is positive meaning related to dance even though that was not the intent of the original sentence. The amount of relevant retrieved documents in this case would be lower when removing the stop-words than when

not removing the stop-words, and therefore the recall would decrease. For sentiment analysis, the removal of stop-words can have an impact on recall.

Question 3b

Precision is reduced if the number of retrieved documents is increased. Let's consider that we want to classify all documents related to the "rose" flower. When removing stop-words, the system might retrieve documents where "rose" comes from the verb "rise" or relates to the color "rose". Hence, the number of retrieved documents would increase, and precision would decrease.

True/False questions

1. dft is an inverse measure of the informativeness of term t.

True. The variable dft refers to the document frequency of the term t, which is the number of documents in the collection that contain the term t. Hence, a term that has a high dft appears in many documents and may therefore not be specific to a particular topic. This implies that the term won't be useful for distinguishing relevant documents from irrelevant documents and is therefore less informative than a document with low dft. It is because of this that dft is used as an inverse measure of the informativeness of a term t.

2. Vector space-based retrieval is always more effective than Boolean retrieval.

False. Boolean retrieval might be more effective than vector space-based retrieval when the query requires exact matching of terms and there is a small number of relevant documents. For example, if the query's goal is to find all the relevant documents that match the phrase "World war I". Vector space-based retrieval would be less effective than Boolean retrieval on this scenario because it might not find any documents that meet the criteria because it is a very specific query.

3. In the vector space model, the higher the value of the normalization factor for a document is, the lower are the chances of retrieval for that document.

False. A higher normalization factor for a document might increase the chances of retrieval if such document is relevant to the query. This is due to the fact that the normalization allows the document to be compared more fairly with other documents of different lengths, as its term weights are divided by a larger value.

4. The stemming process increases the number of unique terms in the index.

False. When adding stemming to the indexing parameters, words are reduced to their root form. For example, the word "corporations" will be reduced to the word "corporation". Therefore, in the index there will appear two counts for the word "corporation" which is opposed to one count for "corporation" and one count for "corporations". In this example, the list of unique terms is reduced by one.

5. Values of beta>1 in F-measure emphasize precision.

False. When beta>1, the F-measure emphasizes recall because recall has greater impact on the harmonic mean than precision. The harmonic mean averages values giving more weight to smaller values than to bigger values, so recall receives more weight when beta>1.

6. In Rocchio's model, $q0$ might be closer to the centroid of the relevant documents than qm.

True. It is possible that after receiving the feedback from the user, the modified query vector qm could shift away from the centroid of relevant documents and therefore be further away from it than $q0$. For example, consider the weights β and γ assigned respectively to the centroid of relevant documents and the centroid of non-relevant documents. If the algorithm incorrectly assigns a high γ and a low β, the obtained qm will be biased towards non-relevant documents and therefore further away from the centroid of relevant documents than $q0$.

Wet Part

Part A

Question 1a

The word "corporation" is found in document 2 with a score of 0.76:

Q0 D2 1 0.760648 indri

Question 1b

We expected to retrieve 2 documents (doc2 and doc3) with the words "corporation" and "corporations" respectively. The reason why we didn't retrieve doc3 is because there is no stemming parameter in our indexing that allows our indexing to reduce the word "corporations" to a common base form "corporation". To fix this we create a new parameters file and added a stemmer:

<parameters>

<memory>1G</memory>

<corpus>

<path>HW2/WET_PART_A/docs.txt</path>

<class>trectext</class>

</corpus>

<stemmer> <name>krovetz</name> </stemmer>

<index>a/index2</index>

</parameters>

When running the query for "corporation" we obtain:

Q0 D2 1 0.252117 indri

Q0 D3 2 0.239987 indri

Question 2

The following query returns D2 first:

<parameters>

<query>

<number>1</number>

<text>corporation</text>

</query>

</parameters>

Question 3

The following query returns D1 first:

<parameters>

<query> <number>2</number>

<text>nobel prize</text>

</query>

</parameters>

Question 4a

D4 is not relevant to the information need expressed by the query “Michael Jackson” because the document centers in Lady Gaga and only mentions Michal Jackson once. The user’s information need would not be satisfied because they would not find useful or relevant information regarding Michael Jackson. The following query 3 is run:

<parameters> <query> <number>3</number> <text> Michael Jackson</text> </query> </parameters>

The obtained score is:

3 Q0 D4 1 1.36686 indri

Question 4b

The query ran for which D4 can be marked as relevant document is:

<parameters> <query> <number>4</number> <text>Lady GaGa</text> </query> </parameters>

4 Q0 D4 1 1.85722 indri

For query 4 we get a higher score than for query 3 with a difference of 0.49036. This difference is due to the fact that “Lady Gaga” is mentioned more times in the document than “Michael Jackson”. We can therefore see an improvement in the score, since the information need is better answered with query 4.

Part B

Question 5

The results are:

StopWord Removal	Krovetz Stemmer	MAP	P@5	P@10
With	With	0.2113	0.3919	0.3725
With	Without	0.1860	0.3933	0.3658

The MAP value of the query with stemming is higher than the MAP value of the query without stemming because stemming retrieves more relevant documents than the unstemmed query, and the AP values calculated for each query are therefore higher. When we are not using stemming, we are being more strict with the form of the word and this causes to not retrieve relevant documents. Therefore, the evaluation metric is worse because we are retrieving a smaller amount of relevant documents.