Ronak Singh
BT21CSE028

Assignment 4

Que 1. a. d1: New york times d2: new york post
d3: los angeles times
Total no. of documents: N = 3, Hence idf values of terms arranged alphabetically are as follows:

angeles :- $\log_2 3/1 = 1.584$
los :- $\log_2 3/2 = 0.584$
new :- $\log_2 3/2 = 0.584$
post :- $\log_2 3/1 = 1.584$
times :- $\log_2 3/2 = 0.584$
york :- $\log_2 3/2 = 0.584$

For all documents, we calculate the tf scores as follows:-

	angeles	los	new	post	Times	york
d1	0	0	1	0	1	1
d2	0	0	1	1	0	1
d3	1	1	0	0	1	0

Now, multiplying tf-idf values we get term vectors for documents

d3 :- $W_{angeles} \rightarrow 1 * 1.584 = 1.584$
$W_{los} \rightarrow 1 * 1.584 = 1.584$
$W_{times} \rightarrow 1 * 0.584 = 0.584$

d2 :- $W_{new} \rightarrow 1 * 0.584 = 0.584$
$W_{york} \rightarrow 1 * 0.584 = 0.584$
$W_{post} \rightarrow 1 * 1.584 = 1.584$

d1 :- $W_{new} \rightarrow 1 * 0.584 = 0.584$
$W_{york} \rightarrow 1 * 0.584 = 0.584$
$W_{times} \rightarrow 1 * 0.584 = 0.584$

d1 = <0.584, 0.584, 0.584>
d2 = <0.584, 0.584, 1.584>
d3 = <1.584, 1.584, 0.584>

Q1. b. When computing tf-idf values for the query terms we divide the frequency by the maximum frequency (2) and multiply with the idf values.

q 0 0 (2/2) * 0.584 = 0.584 0 (1/2) * 0.584 = 0.292

We calculate the length of each document & of the query:-

$d1 = \sqrt{(0.584)^2 + (0.584)^2 + (0.584)^2} = 1.011$

$d2 = \sqrt{(0.584)^2 + (1.584)^2 + (0.584)^2} = 1.786$

$d3 = \sqrt{(1.584)^2 + (1.584)^2 + (1.584)^2} = 2.316$

$q = \sqrt{(0.584)^2 + (0.292)^2} = 0.652$

Then the similarity values are:-

$\text{cosSim}(d1, q) = \frac{[0+0+(0.584)^2+0+0.584*0.292+0]}{(1.011 * 0.652)} = 0.776$

$\text{cosSim}(d2, q) = \frac{[0+0+0.584*0.584+0+0+0]}{1.786 * 0.652} = 0.292$

$\text{cosSim}(d3, q) = \frac{[0+0+0+0+0.584*0.292+0]}{2.316 * 0.652} = 0.112$

	q	d_1	d_2	d_3
ariolimax	0	1	0	0
banana	1	1	1	0
campus	0	0	0	1
columbians	0	1	0	0
cruz	0	0	1	1
mascot	0	0	0	1
mountains	0	0	1	0
santa	0	0	1	1
slug	1	1	1	0

The Rocchio relevance feedback mechanism :- (α, β, γ)

$$\vec{q}_m = \alpha \cdot \vec{q}_0 + \beta \frac{1}{|D_f|} \sum_{\vec{d_j} \in D_f} \vec{d_j} - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d_j} \in D_{nr}} \vec{d_j}$$

$$\alpha = \beta \cdot \gamma = 1$$

Q3. RET(s1) = { $d_2, d_5, d_{150}, d_{250}, d_{11}, d_{33}, d_{50}, d_{600}, d_{600}, d_{520}$ }

RET(s2) = { $d_{250}, d_{400}, d_{150}, d_{210}, d_{999}, d_3, d_{501}, d_{800}, d_{205}, d_{300}$ }

a) Precision = $\frac{\text{No. of relevant documents retrieved}}{\text{No. of documents retrieved}}$

Recall = $\frac{\text{No. of relevant documents retrieved}}{\text{Total no. of relevant documents (REL)}}$

The set REL has 25 relevant documents in this case

REL = { $d_1, d_5, d_6, d_{10}, d_{88}, d_{150}, d_{200}, d_{210}, d_{250}, d_{300}, d_{405}, d_{450}, d_{472}, d_{500}, d_{501}, d_{320}, d_{545}, d_{590}, d_{600}, d_{700}, d_{720}, d_{800}, d_{10}$

For s1, checking for retrieved documents are relevant or not;

{ $d_2, d_{11}, d_{33}, d_{50}, d_{520}$ } = Non relevant

{ $d_5, d_{150}, d_{250}, d_{600}, d_{500}$ } = relevant

retrieved documents	Retrieved docs so far	Precision	Call
d_2	0	0.0	0.0
d_5	1	0.5	0.04
d_{150}	2	0.67	0.08
d_{250}	3	0.75	0.12
d_{11}	3	0.60	0.12
d_{33}	3	0.50	0.12
d_{50}	3	0.43	0.12
d_{600}	4	0.50	0.16
d_{500}	5	0.56	0.20
d_{520}	5	0.50	0.20

For s2, chcking if documents are relevant or not;

{ d_{999}, d_3, d_{205} } = Non-relevant

{ $d_{250}, d_{400}, d_{150}, d_{210}, d_{201}, d_{800}, d_{300}$ } = Relevant

Retrieved document	Relevant Ret. so far	Precision	Recall
d_{250}	1	1.0	0.04
d_{402}	2	1.0	0.08
d_{150}	3	1.0	0.12
d_{210}	4	1.0	0.16
d_{999}	4	0.80	0.16
d_2	4	0.67	0.16
d_{501}	5	0.71	0.20
d_{800}	6	0.75	0.24
d_{205}	6	0.67	0.24
d_{300}	7	0.70	0.28

b) F1 Measure :-

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Computing the F1 measure for each document retrieved for both the systems :-

For system s1, F1 is as follows :-

For $s1_1$

Retrieved documents	Precision	Recall	F1
d_2	0.0	0.0	0.0
d_5	0.5	0.04	0.07
d_{150}	0.67	0.08	0.14
d_{250}	0.75	0.12	0.21
d_{11}	0.60	0.12	0.20
d_{33}	0.50	0.12	0.19
d_{50}	0.43	0.12	0.19
d_{500}	0.50	0.16	0.24
d_{500}	0.56	0.20	0.29
d_{520}	0.50	0.20	0.24

For s2:-

Retrieved Documents	Precision	Recall	F1
d_{250}	1.0	0.04	0.08
d_{400}	1.0	0.08	0.15
d_{150}	1.0	0.12	0.21
d_{210}	1.0	0.16	0.28
d_{999}	0.80	0.16	0.27
d_3	0.67	0.16	0.26
d_{501}	0.71	0.20	0.31
d_{800}	0.75	0.24	0.36
d_{205}	0.67	0.24	0.36
d_{300}	0.70	0.28	0.40

c) To determine which system is better, we can analyze the F1 measures graph. Based on F1 scores, s2 system generally performs better at combining precision & recall, especially as more documents are retrieved.

Q4. a) i) cosine similarity :- Between query Q and a document D_i is calculated as,

$$\text{Cosine similarity} = \frac{Q \cdot D_i}{|Q| * |D_i|}$$

Query Vector $Q = [2, 1, 1, 0, 2, 0, 3, 0]$

$$|Q| = \sqrt{2^2 + 1^2 + 1^2 + 2^2 + 3^2} = \sqrt{19}$$

For Each document ; DOC1 = [0, 3, 4, 0, 0, 2, 4, 0]

DOC2 = [5, 5, 0, 0, 4, 0, 4, 3] DOC3 = [3, 0, 4, 3, 4, 0, 0, 5]

DOC4 = [0, 7, 0, 3, 2, 0, 4, 3] DOC5 = [0, 1, 0, 0, 0, 5, 4, 2]

DOC 6 = [2, 0, 2, 0, 0, 4, 0, 1] DOC7 = [3, 5, 3, 4, 0, 0, 4, 2]

DOC8 = [0, 3, 0, 0, 0, 4, 4, 2] DOC9 = [0, 0, 3, 3, 3, 0, 0, 1]

DOC 10 = [0, 5, 0, 0, 0, 4, 4, 2]

For DOC1 :- $\varphi \cdot D1 \Rightarrow 0+3+4+0+0+0+12+0$

$\Rightarrow 19$

cosine similarity $= \frac{19}{\sqrt{19} \times \sqrt{45}} = \frac{19}{29.14} = 0.746$

For DOC2 :- cosine similarity $= \frac{35}{\sqrt{19}\ \sqrt{91}} = \frac{35}{41.45} = 0.844$

b) Binary term weights with Dot Product :-

DOC1 :- [0, 1, 1, 0, 0, 1, 1, 0] $|D1| = 2$

DOC2 :- [1, 1, 0, 0, 1, 0, 1, 1] $|D2| = \sqrt{5}$

DOC3 :- [1, 0, 1, 1, 1, 0, 0, 1] $|D3| = \sqrt{5}$

DOC4 :- [0, 1, 0, 1, 1, 0, 1, 1] $|D4| = \sqrt{5}$

DOC5 :- [0, 1, 0, 0, 0, 1, 1, 1] $|D5| = 2$

DOC6 :- [1, 0, 1, 0, 0, 1, 0, 1] $|D6| = 2$

DOC7 :- [1, 1, 1, 1, 0, 0, 1, 1] $|D7| = \sqrt{6}$

DOC8 :- [0, 1, 0, 0, 0, 1, 1, 1] $|D8| = 2$

DOC9 :- [0, 0, 1, 1, 1, 0, 0, 1] $|D9| = 2$

DOC10 :- [0, 1, 0, 0, 0, 1, 1, 1] $|D10| = 2$

$\vec{\varphi} = [2, 1, 1, 0, 2, 0, 3, 0]$

Dot Product of binary term weights arranged in decreasing order are as follows :-

DOC2 :- 4 ← DOC7 :- 4 ← DOC1 :- 3 ← DOC3 :- 3 ← DOC4 :- 3 ← DOC5 :- 2 ← DOC5 :- 2 ← DOC6 :- 2 ← DOC8 :- 2 ← DOC9 :- 2 ← BOC10 :- 2

φ_0 = " dog rac " $\varphi_0 = [1, 1, 0, 0, 0, 0, 0, 0, 0, 0]$

Relevant Documents vector sum, computing average

Vector for relevant documents :-

DOC4 = [0,1,0,1,0,0,1,0,0,0]

DOC5 = [1,1,0,0,1,0,0,0,0,0]

DOC6 = [1,1,0,0,0,0,1,0,0,0]

DOC7 = [1,1,0,0,0,0,0,0,0,1]

SUM_R = [3,4,0,1,1,0,2,0,0,1]

$Avg_R = \frac{1}{4}$ [3, 4, 0, 1, 1, 0, 2, 0, 0, 1]

Avg_R = [0.75 1 0 0.25 0.25 0 0.5 0 0 0.25]

Non relevant documents vector sum, computing average vector for non-relevant documents :-

DOC1 = [1, 1, 1, 0, 1, 0, 0, 0, 0, 0]

DOC2 = [1, 1, 0, 0, 1, 0, 0, 0, 0, 0]

DOC3 = [1, 0, 1, 0, 0, 0, 0, 0, 0, 0]

Sum_{NR} = [3, 2, 2, 0, 2, 0, 0, 0, 0, 0]

$Avg_{NR} = \frac{1}{3}$ [3, 2, 2, 0, 2, 0, 0, 0, 0, 0]

Avg_{NR} = [1, 0.66, 0.66, 0, 0.66, 0, 0, 0, 0, 0]

substituting everything in Rocchio's formula :-

$$q_{opt} = 1.0 * q_0 + 1.0 * Avg_R - 0.5 * Avg_{NR}$$

Q_1 = [1,1,0,0,0,0,0,0,0,0] + [0.75, 1, 0, 0.25, 0.25, 0, 0.5, 0, 0, 0.25]
- [0.5, 0.33, 0.33, 0, 0.33, 0, 0, 0, 0, 0]

$\therefore Q_1$ = [1.25, 1.67, -0.33, 0.25, -0.08, 0, 0.5, 0, 0, 0.25]

For DOC3 :- Cosine similarity = $\frac{18}{4.35 * 8.66} = 0.477$

For DOC4 :- 0.5657
For DOC5 :- 0.4397
For DOC6 :- 0.2753
For DOC7 :- 0.1711
For DOC8 :- 0.5130
For DOC9 :- 0.3902
For DOC10 :- 0.4994

For cosine similarity, decreasing order is,
DOC2, DOC7, DOC1, DOC4, DOC8, DOC10, DOC3, DOC5, DOC9, DOC6

ii) Dice's coefficient :- $\frac{2 \times (Q \cdot Di)}{|Q| + |Di|}$

The Dice's coefficient between query & each document is as follows :-

DOC1 :- 1.7273 , DOC2 :- 2.3333 , DOC3 :- 1.2857
DOC4 :- 1.6429 , DOC5 :- 1.2381 , DOC6 :- 0.6667
DOC7 :- 1.7333 , DOC8 :- 1.3626 , DOC9 :- 0.9474
DOC10 :- 1.4167

In decreasing order of dice's coefficient,

DOC2
DOC7
DOC1
DOC4
DOC10
DOC8
DOC3
DOC5
DOC9
DOC6

Q5. Rocchio's formula :-

$$q_1 = \alpha q_0 + \beta \frac{1}{|R|} \sum_{d \in R} D_d - \gamma \frac{1}{|N|} \sum_{d \in N} D_d$$

$$q_{opt} = \alpha q_0 + \beta \frac{1}{|D_r|} \sum_{j \in D_r} \vec{D_j} - \gamma \frac{1}{|D_{nr}|} \sum_{j \in D_{nr}} \vec{D_j}$$

$\beta = 1.0$, $\gamma = 0.5$

Relevant Docs = {Doc4, DOC5, DOC6, DOC7}

Non-Relevant Docs = {DOC1, DOC2, DOC3}

Terms:- dog, race, greyhound, track, betting, iditarod, sled, husky, malamute, alaska.

	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7
dog	1	1	1	1	1	1	1
race	1	1	0	1	1	1	1
greyhound	1	1	1	0	0	0	0
track	1	0	0	0	0	0	0
betting	1	1	0	0	0	1	0
iditarod	0	0	0	1	0	0	1
sled	0	0	0	1	1	1	0
husky	0	0	0	0	1	0	0
malamute	0	0	0	0	1	0	0
alaska	0	0	0	0	0	1	1

The significant increase in weights of terms like 'dog' & 'race' after applying Rocchio's relevance feedback is because they appear frequently in the relevant documents which were positively marked by the users.

The decrease in term weights typically occurs for terms that are more ~~awesome~~ common in non-relevant documents. This reflects the idea that terms frequently appearing in non-relevant documents are less likely to be useful for improving the search results.

Q6. Ranked Document list :-

Total Relevant Docs = 5

1) DOC1
2) DOC2 (Relevant)
3) DOC3 (Relevant)
4) DOC4 (Relevant)
5) DOC5
6) DOC6
7) DOC7
8) DOC8 (Relevant)
9) DOC9
10) DOC10

Calculation precision for each rank position, then use that to find interpolated precision at specified recall

Position 1 :- 0 relevant / 1 retrieved = 0.0
Position 2 :- 1 relevant / 2 retrieved = 0.5
Position 3 :- 2 relevant / 3 retrieved = 0.67
Position 4 :- 3 relevant / 4 retrieved = 0.75
Position 5 :- 3 relevant / 5 retrieved = 0.6
Position 6 :- 3 relevant / 6 retrieved = 0.5
Position 7 :- 3 relevant / 7 retrieved = 0.43
Position 8 :- 4 relevant / 8 retrieved = 0.5
Position 9 :- 4 relevant / 9 retrieved = 0.44

6. Number of documents retrieved = 10
Number of relevant documents = 5
Number of relevant documents retrieved = 4 (positions 2, 3, 4 & 8)

Rank Position	Precision	Recall
1	0/1 = 0	0/5 = 0
2	1/2 = 0.5	1/5 = 0.2
3	2/3 = 0.67	2/5 = 0.4
4	3/4 = 0.75	3/5 = 0.6
5	3/5 = 0.60	3/5 = 0.6
6	3/6 = 0.5	3/5 = 0.6
7	3/7 = 0.43	3/5 = 0.6
8	4/8 = 0.5	4/5 = 0.8
9	4/9 = 0.44	4/5 = 0.8
10	4/10 = 0.4	4/5 = 0.8

Interpreted Precision values:
For each recall level, we take the maximum precision observed recall level greater than or equal to that recall level

Recall level	Interpolated Precision
0.0	0.75
0.1	0.75
0.2	0.75
0.3	0.75
0.4	0.75
0.5	0.75
0.6	0.75
0.7	0.5
0.8	0.5
0.9	0
1.0	0

Rank	Relevant retrieved	Total retrieved	Precision	Recall
1	0	1	0.0	0.0
2	1	2	0.5	0.2
3	2	3	0.67	0.4
4	3	4	0.75	0.6
5	3	5	0.6	0.6
6	3	6	0.5	0.6
7	3	7	0.43	0.6
8	4	8	0.5	0.8
9	4	9	0.44	0.8
10	4	10	0.4	0.8

* Interpreted Precision for specified recalls

Recall level	Interpolated Precision
0.0	0.0
0.1	0.5
0.2	0.5
0.3	0.67
0.4	0.67
0.5	0.75
0.6	0.75
0.7	0.75
0.8	0.5
0.9	0.5
1.0	0.4

Q7] a. Total no. of docs in collection = 10,000
No. of relevant documents = 150
No. of documents retrieved by the system = 250
No. of relevant documents retrieved = 125

Precision = $\frac{\text{Relevant documents retrieved}}{\text{Total documents retrieved}} = \frac{125}{500} = 0.5 \mid 50\%$

Recall = $\frac{\text{Relevant documents retrieved}}{\text{Total relevant documents}} = \frac{125}{250} = \frac{5}{6} = 0.833$

b) Precision ⇒ 50% precision means that half of the documents retrieved by the system are actually relevant. In this scenario, it means that for every 2 patient the system identifies as needing a retest, 1 patient is not actually diabetic or didn't have a mood test on the malfunction day.

Recall ⇒ 83.3% recall indicates that the system was able to identify 83.3% of all diabetic patients who need to repeat their tests. This means the system missed about 16.7% of the diabetic patients who should be contacted for a retest.

The system's recall is quite high, meaning it retrieves most of the relevant patients. The mean precision is lower, suggesting that the hospital would contact many patients unnecessarily. In medical scenario high recall is critical since missing a relevant patient could lead to serious health consequences. However, improving precision would reduce unnecessary communication with patients who don't need retests.

c) If system is tuned for 100% recall, it would attempt to retrieve all relevant documents, this would result in a significant drop in precision. The system would retrieve many more irrelevant documents, meaning the hospital would contact large number of non-diabetic or unaffected patients.

d) In this scenario, recall is more important than precision. That is because :-

1) Missing even a single diabetic patient, who needs a repeat test could lead to health risks. It is better to ensure that as many relevant patients as possible are contacted unnecessarily (low precision)

2) A high recall ensures that most, if not all of the diabetic patients who had a mood test on the malfunction day are identified.

Value of weighing factor α in F-score which is the weighted harmonic mean of precision and recall are as follows:-

$$F_\beta = \frac{(1+\beta)^2 \cdot (\text{Precision} \cdot \text{Recall})}{\beta^2 \cdot (\text{Precision} + \text{Recall})}$$

In this scenario, we would choose higher β. A common value could be $\beta = 2$ which gives more weight to the recall in f-score

In terms of weighing factor α, which assigns a values that prioritates recall, a reasonable choice would be $\alpha = 0.2$ or $\alpha = 0.3$

Q8. a) Precision & recall when both judges must agree for a document to be relevant.

Documents considered relevant by both judges :-

Judge 1 :- {3, 4, 5, 6, 7, 8, 12}

Judge 2 :- {8, 4, 9, 10, 11}

Intersection :- {3, 4}

System returns the set :- {2, 5, 6, 7, 8}

Relevant documents retrieved (common to the system results & the set {3, 4}) = 0

Total retrieved documents = 5

Total relevant documents (Both judges agree) = 2
Precision = $\frac{0}{5} = 0$, Recall = $\frac{0}{2} = 0$

b) A document is considered relevant if either judge considers it relevant. so, we take the union of relevant documents.
Documents considered relevant by either judge :-
Judge 1's relevant documents = { 3, 4, 5, 6, 7, 8, 12 }
Judge 2's relevant documents = { 3, 4, 9, 10, 11 }
Union (Documents either judge considers relevant)
= { 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 }
Now, system returns the set { 2, 5, 6, 7, 8 }
Relevant documents retrieved = { 5, 6, 7, 8 } (4 docs)
Total retrieved documents = 5
Total relevant documents (union of both judges) = 10
Precision = $\frac{4}{5} = 0.8$ Recall = $\frac{4}{10} = 0.4$

c) Average Precision Based on Both Relevance scenario
1) Average

9a. R N R N N N N R R N N N N N R N N N N R

$$MAP(Q) = \frac{1}{8}\left[1 + \frac{2}{3} + \frac{3}{8} + \frac{4}{9} + \frac{5}{15} + \frac{6}{20}\right]$$

$$= 0.389 \approx 0.39$$

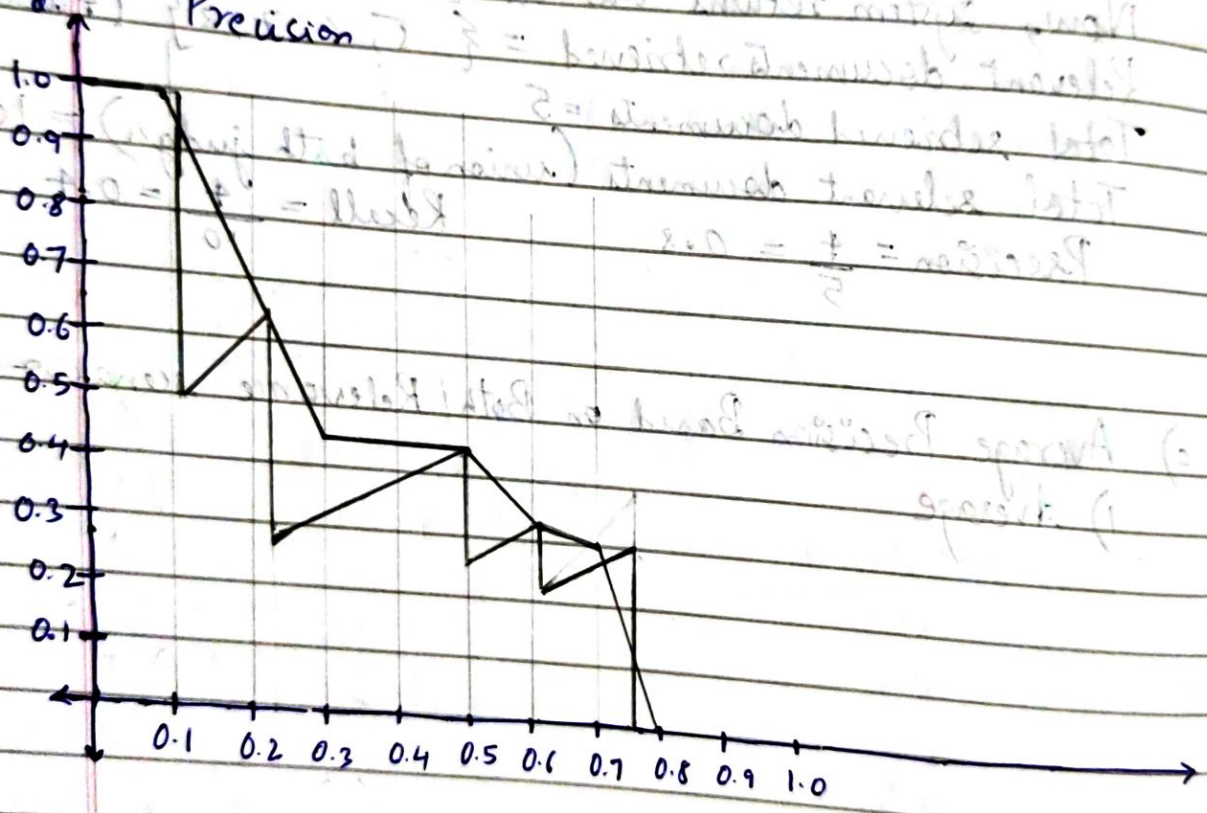
b. For R-precision, we take ratio of relevant docs in the first 8 docs (total relevant docs) to total relevant docs.

∴ R-precision = $\frac{3}{8} = 0.375$ R N R N N N N R

c. Precision at 10

$$\frac{4}{10} = 0.4$$

d. Precision



10a. $\cos\theta = \dfrac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|}$

$$\cos\theta = \frac{x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4}{\sqrt{x_1^2 + x_2^2 + x_3^2 + x_4^2}\sqrt{y_1^2 + y_2^2 + y_3^2 + y_4^2}}$$

b. Report A $\langle 12, 0, 3, 24 \rangle$
Report B $\langle 10, 5, 20, 10 \rangle$
Report C $\langle 0, 12, 9, 8 \rangle$
Query $\langle 1, 1, 1, 1 \rangle$

$$\cos\theta_{A,Q} = \frac{12 + 3 + 24}{\sqrt{12^2+3^2+24^2}\sqrt{1^2+1^2+1^2+1^2}} = \frac{13}{27 \times 2} = \frac{13}{18} = 0.722$$

$$\cos\theta_{B,Q} = \frac{10 + 5 + 20 + 10}{\sqrt{10^2+5^2+20^2+10^2} \times 2} = \frac{45}{25 \times 2} = \frac{9}{10} = 0.9$$

$$\cos\theta_{C,Q} = \frac{12 + 9 + 8}{\sqrt{12^2+9^2+8^2} \times 2} = \frac{29}{17 \times 2} = \frac{29}{34} = 0.853$$

$$\theta_{B,Q} < \theta_{C,Q} < \theta_{A,Q} \quad [\cos\theta_{B,Q} > \cos\theta_{C,Q} > \cos\theta_{A,Q}]$$

B is most likely to be relevant to the query
∴ We choose B.
Since B is the only document containing all the keywords
Report C does not ~~remain~~ mention North Sea.
Report A does not mention oil at all. Superiority of Cosine A seems clear from cosine values, but it is not obvious from simply inspecting the numbers.