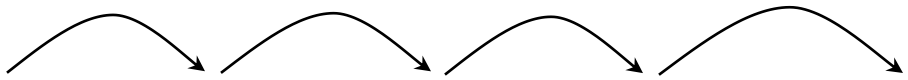
Information Retrieval (CS60092)
Computer Science and Engineering, Indian Institute of Technology Kharagpur

Supplementary End-Semester Examination for Session 2012 - 2013, July 2013 SOLUTIONS

Answer as many questions as you can. *Time: 3 hours*
Use of calculator is allowed. *Maximum Marks: 100*
State any assumptions made clearly.

Q. 1> Consider the following postings lists p1 and p2. p1 has skip pointers. p2 does not have any skip pointer.



p1: 3 5 9 15 24 39 60 68 75 81 84 89 92 96 97 100 115

p2: 3 5 89 95 97 99 100 101

(a) Intersect the postings lists WITHOUT USING the skip pointers. Write down the comparisons (x,y) made while doing the intersection, where x is a docID from p1 and y is a docID from p2. How many comparisons are required?

Answer: The comparisons are: (3,3), (5,5), (9,89), (15,89), (24,89), (39,89), (60,89), (68,89), (75,89), (81,89), (84,89), (89,89), (92,95), (96,95), (96,97), (97,97), (100,99), (100,100), (115,101). **19** comparisons are required.

(b) Intersect the postings lists USING the skip pointers. Write down the comparisons (x,y) made while doing the intersection, where x is a docID from p1 and y is a docID from p2. How many comparisons are required?

Answer: The comparisons are: (3,3), (5,5), (9,89), (15,89), (24,89), (75,89), (75,89), (92,89), (81,89), (84,89), (89,89), (92,95), (115,95), (96,95), (96,97), (97,97), (100,99), (100,100), (115,101). **19** comparisons are required.

(c) Do skip pointers help in processing AND queries? Justify your answer.

Answer: Yes. They are useful in finding intersections of lists in less time.

(d) Do skip pointers help in processing OR queries? Justify your answer.

Answer: No. While processing queries of the form "q1 OR q2", it is essential to visit every docID in the posting lists of both the terms. Thus skipping part of either list will result in incorrect answer. **[4 + 4 + 1 + 1 = 10]**

Q.2> Suppose that a document collection consists of following two documents

d1: *free eBooks free software eBooks*

d2: *hundred free pdfs*

User's initial query is **q:** *free eBooks free pdfs free computer eBooks*

The user judges **d1** relevant and **d2** non-relevant. Assume that we are using direct term frequency (with no scaling and no document frequency). There is no need to length-normalize vectors. Using Rocchio relevance feedback, what would the revised query vector be after relevance feedback? Assume $\alpha = 1$, $\beta = 0.75$, $\gamma = 0.25$.

[6]

Answer: The formula for Rocchio Algorithm is

$$\vec{q}_m = \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$

where q_m is the modified query, q_0 is the original query vector, D_r and D_{nr} are the set of known relevant and non-relevant documents respectively, and α, β, and γ are weights attached to each term.

Here the numbers of relevant and non-relevant documents are both 1.

So, the modified query would be

1(3 * free +2* eBooks +pdfs +computer)* + 0.75*(2* *free+ 2*eBooks+ software)* - *0.25*(hundred+ free+ pdfs)*

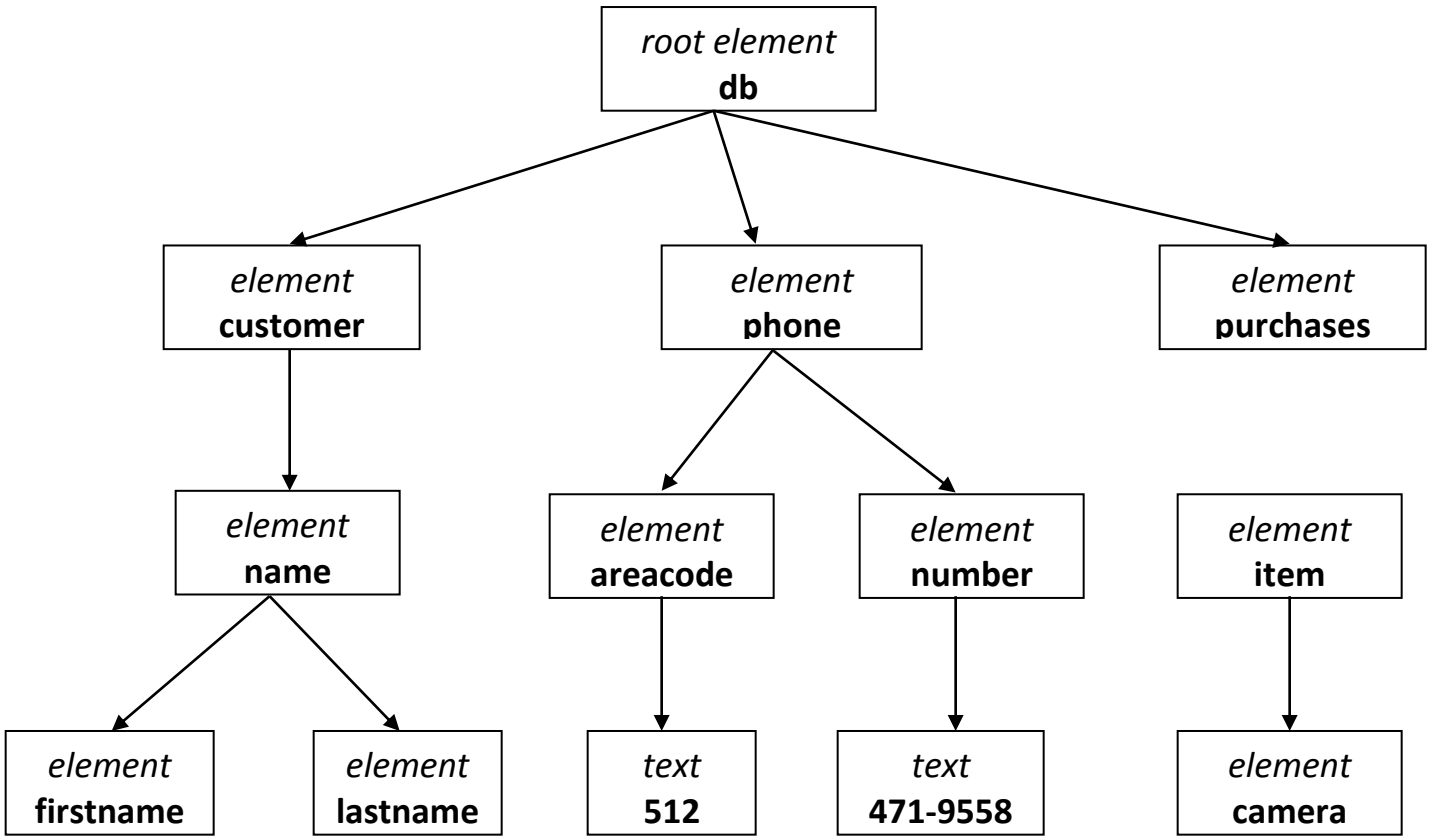
*= (3+0.75*2-0.25) * free+ (2+0.75*2) *eBooks+ computer+ (1-0.25)*pdfs+0.75*software*

= 4.25 free +3.5* eBooks+ computer+ 0.75*pdfs+ 0.75*software*

Q.3> Draw the DOM tree for the following XML document. **[4]**

```
<db>
 <customer>
   <name>
    <firstname>John</firstname> <lastname>Doe</lastname>
   </name>
   <phone>
    <areacode>512</areacode> <number>471-9558</number>
   </phone>
   <purchases>
    <item>
     <camera>
       <type>Canon digital</type> <price>200</price>
     </camera>
    </item>
   </purchases>
 </customer>
</db>
```

Answer:



Q. 4> Consider the following matrix representing **distance** between six documents:

Document	A	B	C	D	E	F
A	0	662	877	255	412	996
B	662	0	295	468	268	400
C	877	295	0	754	564	138
D	255	468	754	0	219	869
E	412	268	564	219	0	669
F	996	400	138	869	669	0

Compute hierarchical single-linkage clustering of these six documents. Clearly show the matrices at each step of building the dendrogram. (No marks will be given for showing only the final dendrogram.) **[10]**

Answer:

The nearest pair of document is C and F, at distance 138. These are merged into a single cluster called "C/F".
Then we compute the distance from this new compound object to all other objects. In single link clustering the rule is that the distance from the compound object to another object is equal to the shortest distance from any member of the cluster to the outside object.
So the distance from "C/F" to E is chosen to be 564, which is the distance from C to E, and so on.

After merging C with F, we obtain the following matrix:

Document	A	B	C/F	D	E
A	0	662	877	255	412
B	662	0	295	468	268
C/F	877	295	0	754	564
D	255	468	754	0	219
E	412	268	564	219	0

min d(i,j) = d(D,E) = 219 => merge D and E into a new cluster called "D/E"

After merging D with E, we obtain the following matrix:

Document	A	B	C/F	D/E
A	0	662	877	255
B	662	0	295	268
C/F	877	295	0	564
D/E	255	268	564	0

min d(I,j) = d(A,D/E) = 255 => merge A and D/E into a new cluster called A/D/E

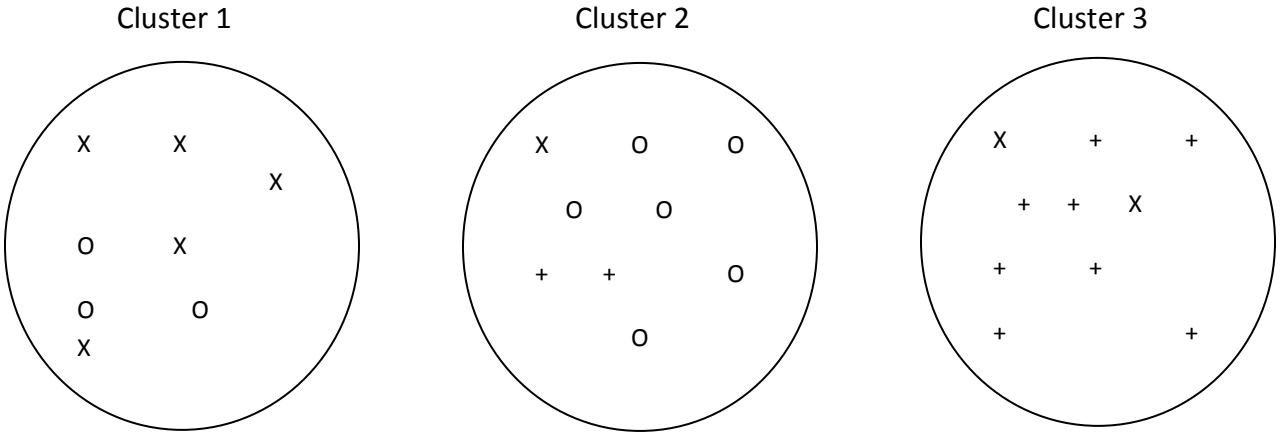
Document	A/D/E	B	C/F
A/D/E	0	268	564
B	268	0	295
C/F	564	295	0

min d(I,j) = d(A/D/E,B) = 268 => merge A/D/E and B into a new cluster called A/D/E/B

Document	A/D/E/B	C/F
A/D/E/B	0	295
C/F	295	0

Finally, we merge the last two clusters at distance 295.

Q. 5> Consider the following figure for clusters found after performing flat clustering (K-Means) on a set of documents. The gold standard for each document is produced by human judges. Each document belongs to one of the three gold standard classes (x, o and +)



Calculate the following quality measures for the above clustering
(a) Purity
(b) NMI
(c) Rand Index
(d) F-Measure **[2 + 4 + 2 + 2 = 10]**

Answer:

(a) To compute purity, each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and dividing by N.

$$\text{purity}(\Omega, \mathbb{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|$$

Purity = (1/27) * (5+6+8) = 19/27 = **0.704**

(b) NMI is based on defining a confusion matrix N, where the rows correspond to the gold standard classes and the columns correspond to the clusters found.
The member of N, N_{ij} is simply the number of nodes in the class i that appear in the found cluster j. The number of classes is denoted C_A and the number of found clusters is denoted by C_B. The sum over row i of matrix N_{ij} is denoted N_i; and the sum over column j is denoted N_j.

$$NMI(A,B) = \frac{-2\sum_{i=1}^{c_A}\sum_{j=1}^{c_B} N_{ij} log\left(\frac{N_{ij}N}{N_{i.}N_{.j}}\right)}{\sum_{i=1}^{c_A} N_{i.} log\left(\frac{N_{i.}}{N}\right) + \sum_{j=1}^{c_B} N_{.j} log\left(\frac{N_{.j}}{N}\right)}$$

NMI = [-2*[5*log(5*27/8*8)+1*log(1*27/8*9)+2*log(2*27/8*10)
+3*log(3*27/9*8)+6*log(6*27/9*6)+2*log(2*27/10*9)
+8*log(8*27/10*10)]] / [8*log(8/27)+9*log(9/27) +10*log(10/27)
+9*log(9/27)+9*log(9/27)+10*log(10/27)]
= 6.1015*2/25.736 = **0.475**

(c) A true positive (TP) decision assigns two similar documents to the same cluster; a true negative (TN) decision assigns two dissimilar documents to different clusters.

There are two types of errors we can commit. A false positive (FP) decision assigns two dissimilar documents to the same cluster. A false negative (FN) decision assigns two similar documents to different clusters. The Rand Index (RI) measures the percentage of decisions that are correct.

$$RI = \frac{TP + TN}{TP + FP + FN + TN}$$

We first compute TP +FP.
The three clusters contain 8, 9 and 10 points, respectively, so the total number of "positives" or pairs of documents that are in the same cluster is:

TP + FP = 8C2 + 9C2 + 10C2 = 109

Of these, the x pairs in cluster 1 and 3, the o pairs in cluster 1 and 2, the + pairs in cluster 2 and 3 are true positives.

TP = 5C2 + 2C2 + 3C2 + 6C2 + 8C2 + 2C2 = 58

Thus, FP = 109 – 58 = 51

FN = 5*2 + 5*1 +2*1 + 6*3 + 8*2 = 51

TN = Total – (TP+FP+FN) = 27C2 – (58+51+51) = 351 – 160 = 191

Thus, RI = (58+191)/351 = 249/351 = **0.709**

(d) P = TP/ (TP+FP) = 58/ (58+51) = 0.53
R =TP/ (TP+FN) = 58/ (58+51) = 0.53

F Measure = 2PR/(P+R) = **0.530**

Q. 6> Consider the problem of learning to classify a name as being Food or Beverage.
Assume the following training set:

Document	**Class**
Cherry Pie Chocolate	Food
Chicken Wings Crispy	Food
Cream Soda Water	Beverage
Orange Soda	Beverage

(a) Train a multinomial naïve Bayes classifier on the above data. Calculate the multinomial parameters (priors and conditional probabilities). Use *Laplace Smoothing* for the calculation of conditional probabilities.

(b) What does this classifier predict about the class of the following test document: ***chocolate cream soda***? Assume *positional independence* of terms. **[7 + 3 = 10]**

Answer: (a) We denote the two classes Food and Beverages by F and B respectively. There are 10 distinct terms. We use unigram based naïve bayes classifier (multinomial model) with laplace smoothing for classification.

Term (t)	P(t|F) [Raw]	P(t|B) [Raw]	P(t|F) [Smoothed]	P(t|B) [Smoothed]
Chocolate	1/6	0/5	(1+1)/(6+10) = 2/16	(0+1)/(5+10) = 1/15
Cream	0/6	1/5	(0+1)/(6+10) = 1/16	(1+1)/(5+10) = 2/15
Soda	0/6	2/5	(0+1)/(6+10) = 1/16	(2+1)/(5+10) = 3/15

(b) P(F|chocolate cream soda) α P(F) * P(Chocolate|F) * P(Cream|F) *P(soda|F)

= (2/4) *(2/16) * (1/16) * (1/16)

= 4/(4 * 16 * 16 * 16).

P(B|chocolate cream soda) α P(B) * P(Chocolate|B) * P(Cream|B) *P(soda|B)

= (2/4) *(1/15) * (2/15) * (3/15)

= 12/(4 * 15 * 15 * 15).

As P(B|*chocolate cream soda*) > P(F|*chocolate cream soda*), the predicted class for the test document is "Beverage".

Q. 7> Consider a document collection that contains the following documents:
d_1: *tick goes the clock goes tick tick tick*
d_2: *tick tock big time*
d_3: *clock tower*
d_4: *big tower of clock*
Let a query be *"clock tick".* Compute the tf-idf scores of each document with respect to this query and provide the resultant document ranking. **[10]**

Answer: $idf_{clock} = \log_{10}(N/df_t) = \log_{10}(4/3) = 0.12$
$idf_{tick} = \log_{10}(N/df_t) = \log_{10}(4/2) = 0.30$
For d_1, $tf_{clock} = 1$, $idf_{clock} = 0.12 \rightarrow$ tf-$idf_{clock} = 1 \times 0.12 = 0.12$

tf_{tick} = 4, idf_{tick} = 0.30 → $tf\text{-}idf_{tick}$ = 4 x 0.30 = 1.20
Score of d_1 = 0.12 + 1.20 = **1.32 Ans.**

For d_2, tf_{clock} = 0, idf_{clock} = 0.12 → $tf\text{-}idf_{clock}$ = 0 x 0.12 = 0.00
tf_{tick} = 1, idf_{tick} = 0.30 → $tf\text{-}idf_{tick}$ = 1 x 0.30 = 0.30
Score of d_2 = 0.00 + 0.30 = **0.30 Ans.**

For d_3, tf_{clock} = 1, idf_{clock} = 0.12 → $tf\text{-}idf_{clock}$ = 1 x 0.12 = 0.12
tf_{tick} = 0, idf_{tick} = 0.30 → $tf\text{-}idf_{tick}$ = 0 x 0.30 = 0.00
Score of d_1 = 0.12 + 0.00 = **0.12 Ans.**

For d_4, tf_{clock} = 1, idf_{clock} = 0.12 → $tf\text{-}idf_{clock}$ = 1 x 0.12 = 0.12
tf_{tick} = 0, idf_{tick} = 0.30 → $tf\text{-}idf_{tick}$ = 0 x 0.30 = 0.00
Score of d_1 = 0.12 + 0.00 = **0.12 Ans.**

Resultant document ranking: ***d*₁, *d*₂, *d*₃, *d*₄** OR ***d*₁, *d*₂, *d*₄, *d*₃.**

Q. 8> Consider two queries for which there are 4 and 6 relevant documents in the collection respectively. Contrast two systems run on this collection. Their top 10 results are judged for relevance as follows (the leftmost item is the top ranked search result):

System 1, Query 1: R N R R R N N N N N

System 1, Query 2: R R R R N N N R N R

System 2, Query 1: N N N N R R R N N R

System 2, Query 2: N N N N R R R R R R

(a) What is the MAP of each system? Which system has a higher MAP?

Answer: AP for System 1, Query 1 = (1 + 2/3 + 3/4 + 4/5)/4 = 0.804

AP for System 1, Query 2 = (1 + 1 + 1 + 1 + 5/8 + 6/10)/6 = 0.871

MAP for System 1 = (0.804 + 0.871)/2 = **0.838**

AP for System 2, Query 1 = (1/5 + 2/6 + 3/7 + 4/10)/4 = 0.340

AP for System 2, Query 2 = (1/5 + 2/6 + 3/7 + 4/8 + 5/9 + 6/10)/6 = 0.436

MAP for System 2 = (0.340 + 0.436)/2 = **0.388**

System 1 has a higher map.

(b) What does the result say about what is important in getting a good MAP score?

Answer: For getting a good MAP score, a system must retrieve relevant pages higher up in the ranked list of retrieved documents.

(c) How is R-precision of a system defined? What is the R-precision of each system here? Does it rank the systems in the same order as MAP? [**6 + 1 + 3 = 10]**

Answer: The precision at the top *Rel* documents returned, where *Rel* is the number of relevant documents for the query, is known as the R-precision of the system.

R-precision for System 1 = (3/4 + 4/6)/2 = **0.708**

R-precision for System 2 = (0 + 2/6)/2 = **0.167**

Yes, R-precision ranks the systems in the same order as MAP.

Q. 9> Consider the following documents:

D1: *english channel atlantic*

D2: *national geography channel english*

D3: *doordarshan national english news*

Using unigram language model, rank the above documents for the query

national news channel english.

To compute the model probabilities, combine MLE estimates from documents and the collection giving equal importance to both. **[10]**

Answer:

Using linear interpolation smoothing with alpha = 0.5.

Term	**P(t|D1)**	**P(t|D2)**	**P(t|D3)**	**P(t|C)**
National	0	1/4	¼	2/11
News	0	0	¼	1/11
Channel	1/3	1/4	0	2/11
English	1/3	1/4	¼	3/11

Query *q* = *national news channel english*

P(q|D1) = (0 + 2/11) / 2 * (0 + 1/11) / 2 * (1/3 + 2/11) / 2 * (1/3 + 3/11) / 2

= $3.225 * 10^{-4}$

P(q|D2) = (1/4 + 2/11) / 2 * (0 + 1/11) / 2 * (1/4 + 2/11) / 2 * (1/4 + 3/11) / 2

= $5.538 * 10^{-4}$

P(q|D3) = (1/4 + 2/11) / 2 * (1/4 + 1/11) / 2 * (0 + 2/11) / 2 * (1/4 + 3/11) / 2

= $8.744 * 10^{-4}$

Hence, the ranking is: **D3 > D2 > D1.**

Q. 10> Consider the query *obama health plan*. The document collection consists of six documents only, which are marked as relevant (R) or non-relevant (NR):

d_1: *president rejects rumors about his own bad health* (NR)
d_2: *the plan is to visit obama* (NR)
d_3: *obama raises concerns with us medical reforms* (R)
d_4: *president states a health vision* (R)
d_5: *romney states a health issue* (NR)
d_6: *obama states a health plan* (R)

Assume a binary independence model (BIM) of retrieval. Rank the documents in descending order of their retrieval status value (RSV). Use contingency tables to show intermediate steps. Do not use any smoothing. The RSV for a BIM model is given by

$$RSV_d = \sum_{t:x_t = q_t = 1} \log_{10} \frac{p_t(1-u_t)}{u_t(1-p_t)}$$

where, for each term *t*, the probabilities of occurrence p_t and u_t can be represented in the form of the following contingency table:

	Document	R	NR
Term present	$x_t = 1$	p_t	u_t
Term absent	$x_t = 0$	$1 - p_t$	$1 - u_t$

[10]

Answer: For term *obama*,

	Document	R	NR
Term present	$x_t = 1$	0.67	0.33
Term absent	$x_t = 0$	0.33	0.67

For term *health*,

	Document	R	NR
Term present	$x_t = 1$	0.67	0.67
Term absent	$x_t = 0$	0.33	0.33

For term *plan*,

	Document	R	NR
Term present	$x_t = 1$	0.33	0.33
Term absent	$x_t = 0$	0.67	0.67

$RSV_1(for\ health) = \log_{10}\frac{0.67*0.33}{0.67*0.33} = 0.00$

$RSV_2(for\ obama\ and\ plan) = \log_{10}\frac{0.67*0.67}{0.33*0.33} + \log_{10}\frac{0.33*0.67}{0.33*0.67} = 0.60$

$RSV_3(for\ obama) = \log_{10}\frac{0.67*0.67}{0.33*0.33} = 0.60$

$RSV_4(for\ health) = \log_{10}\frac{0.67*0.33}{0.67*0.33} = 0.00$

$RSV_5(for\ health) = \log_{10}\frac{0.67*0.33}{0.67*0.33} = 0.00$

$RSV_6(for\ obama, health\ and\ plan) = \log_{10}\frac{0.67*0.67}{0.33*0.33} + \log_{10}\frac{0.67*0.33}{0.67*0.33} + \log_{10}\frac{0.33*0.67}{0.33*0.67} = 0.60$

Ranking: $d_2 = d_3 = d_6 > d_1 = d_4 = d_5$

P. T. O.

Q. 11> Consider the following term-document matrix *C*.

Terms	D1	D2	D3	D4	D5	D6
Ship	1	0	1	0	0	0
Boat	0	1	0	0	0	0
Ocean	1	1	0	0	0	0
Voyage	1	0	0	1	1	0
Trip	0	0	0	1	0	1

(a) Suppose vector space model is used to represent the documents. Vector dimensions are filled with raw frequency counts of the corresponding terms. According to this representation, what is the similarity between the documents D2 and D3?

(b) C is decomposed as $C = U\sum V^T$. The matrices U, ∑ and V are given below.

U =

	1	2	3	4	5
ship	−0.44	−0.30	0.57	0.58	0.25
boat	−0.13	−0.33	−0.59	0	0.73
ocean	−0.48	−0.51	−0.37	0	−0.61
voyage	−0.70	0.35	0.15	−0.58	0.16
trip	−0.26	0.65	−0.41	0.58	−0.09

∑ =

2.16	0	0	0	0
0	1.59	0	0	0
0	0	1.28	0	0
0	0	0	1	0
0	0	0	0	0.39

V^T =

	d1	d2	d3	d4	d5	d6
1	−0.75	−0.28	−0.20	−0.45	−0.33	−0.12
2	−0.29	−0.53	−0.19	0.63	0.22	0.41
3	0.28	−0.75	0.45	−0.20	0.12	−0.33
4	0	0	0.58	0	−0.58	0.58
5	−0.53	0.29	0.63	0.19	0.41	−0.22

<i> Suppose a low rank approximation of C is obtained as C_2 by keeping the *most* important two terms. According to C_2, what is the cosine similarity between documents D2 and D3?

C_2 =

[0.8511 0.5189 0.2807 0.1272 0.2087 -0.0815
0.3628 0.3567 0.1559 -0.2042 -0.0228 -0.1814
1.0128 0.7201 0.3614 -0.0443 0.1637 -0.2081
0.9726 0.1284 0.1967 1.0310 0.6214 0.4096
0.1215 -0.3905 -0.0840 0.9038 0.4127 0.4911]

Sim(D2,D3)
= < [0.52 0.36 0.72 0.13 -0.39], [0.28 0.16 0.36 0.20 -0.08]>
= .52 (Similarity is calculated using inner product)
= **0.9417.**

<ii> Suppose another low rank approximation of C is obtained as C'_2 by keeping the *least* important two terms. According to C'_2, what is the cosine similarity between documents D2 and D3?

C'_2 =

[0.8511 0.5189 0.2807 0.1272 0.2087 -0.0815
0.3628 0.3567 0.1559 -0.2042 -0.0228 -0.1814
1.0128 0.7201 0.3614 -0.0443 0.1637 -0.2081
0.9726 0.1284 0.1967 1.0310 0.6214 0.4096
0.1215 -0.3905 -0.0840 0.9038 0.4127 0.4911]

Sim(D2,D3) =
= < [0.03 0.08 -0.07 0.02 -0.01], [0.40 0.18 -0.15 -0.30 0.31]>
= .028 (Similarity is calculated using inner product)
= **0.3896.**

(c) Find out the Eigen Values of the matrix CC^T.

Answer: 4.68, 2.54, 1.63, 1.00, 0.16

[2 + 2 + 2 + 4 = 10]
