Information Retrieval (CS60092)
End-semester examination, Autumn 2013 – 2014

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

Q. 1> We wish to apply the naïve Bayes text classification with a multinomial model with the document class information below.

set	doc-id	document text	class label
train	1	*money invest in bank*	finance
	2	*river bank is green and green*	general
	3	*time invest in exam*	general
	4	*money invest bank for interest*	finance
	5	*paper is money is paper*	general
	6	*bank bank money money*	finance
test	7	*invest bank on river bank*	?

Use the stop list: *a, an, the, how, what, why, when, who, in, is, and, for, to, from, by, with, of*.

a. What are the prior probabilities needed to classify the test document?

Sol. Prior probabilities: $\hat{P}(finance) = \frac{3}{6} = \frac{1}{2}$; $\hat{P}(general) = \frac{3}{6} = \frac{1}{2}$

b. What is the vocabulary size (without the stop words)?

Sol. Vocabulary size $V = 9$ [*money, invest, river, bank, green, time, exam, interest, paper*]

c. What are the conditional probabilities required to classify the test document?

Sol. The conditional probabilities required to classify the test document are as follows:

$$\hat{P}(invest|finance) = \frac{2+1}{11+9} = \frac{3}{20}$$

$$\hat{P}(invest|general) = \frac{1+1}{10+9} = \frac{2}{19}$$

$$\hat{P}(bank|finance) = \frac{4+1}{11+9} = \frac{5}{20} = \frac{1}{4}$$

$$\hat{P}(bank|general) = \frac{1+1}{10+9} = \frac{2}{19}$$

$$\hat{P}(river|finance) = \frac{0+1}{11+9} = \frac{1}{20}$$

$$\hat{P}(river|general) = \frac{1+1}{10+9} = \frac{2}{19}$$

d. What is the class label assigned to the test document? **[2 + 1 + 6 + 1 = 10]**

Sol.

$$\hat{P}(finance|doc\text{-}id\text{-}7) \propto \frac{1}{2} \times \frac{3}{20} \times \left(\frac{1}{4}\right)^2 \times \frac{1}{20} \approx 0.0002343$$

$$\hat{P}(general|doc\text{-}id\text{-}7) \propto \frac{1}{2} \times \frac{2}{19} \times \left(\frac{2}{19}\right)^2 \times \frac{2}{19} \approx 0.0000614$$

Thus, the class label *finance* is assigned to the test document.

Q. 2> We wish to use the following queries and corpus for document ranking using the MLE unigram model. Use the stop list: *a, an, the, how, what, why, when, who, in, is, and, for, to, from, by, with, of*.

doc-id	document text	relevance (*q*1)	relevance (*q*2)
1	*jaguar is a fast animal*	R	NR
2	*jaguar is a fast computer*	NR	R
3	*animal family jaguar leopard panther*	R	NR
4	*animal animation with jaguar computer*	NR	R

We have two queries – *q*1: *animal jaguar* and *q*2: *jaguar computer.*

a. Rank the documents with respect to *q*1 and *q*2 using individual document models only.

Sol. P(q1|d1) = 1/3 x 1/3 = 1/9
P(q1|d2) = 0 x 1/3 = 0
P(q1|d3) = 1/5 x 1/5 = 1/25
P(q1|d4) = 1/4 x 1/4 = 1/16

P(q2|d1) = 1/3 x 0 = 0
P(q2|d2) = 1/3 x 1/3 = 1/9
P(q2|d3) = 1/5 x 0 = 0

P(q2|d4) = 1/4 x 1/4 = 1/16

Thus, ranking for q1 (top rank first): d1, d4, d3, d2
Ranking for q2 (top rank first): d2, d4, d1 = d3

b. Now rank the documents for both queries mixing document and corpus models with the weight ratio 1:3.

Sol. Weight ratio 1:3 implies document weight = 1/4 and corpus weight = 3/4, i.e. λ = ¼
P(q1|d1) = [(1/3 x 1/4) + (3/15 x 3/4)] x [(1/3 x 1/4) + (4/15 x 3/4)] = 0.0661
P(q1|d2) = [(0 x 1/4) + (3/15 x 3/4)] x [(1/3 x 1/4) + (4/15 x 3/4)] = 0.0425
P(q1|d3) = [(1/5 x 1/4) + (3/15 x 3/4)] x [(1/5 x 1/4) + (4/15 x 3/4)] = 0.05
P(q1|d4) = [(1/4 x 1/4) + (3/15 x 3/4)] x [(1/4 x 1/4) x (4/15 x 3/4)] = 0.0027

P(q2|d1) = [(1/3 x 1/4) + (4/15 x 3/4)] x [(0 x 1/4) + (2/15 x 3/4)] = 0.0283
P(q2|d2) = [(1/3 x 1/4) + (4/15 x 3/4)] x [(1/3 x 1/4) + (2/15 x 3/4)] = 0.0519
P(q2|d3) = [(1/5 x 1/4) + (4/15 x 3/4)] x [(0 x 1/4) + (2/15 x 3/4)] = 0.025
P(q2|d4) = [(1/4 x 1/4) + (4/15 x 3/4)] x [(1/4 x 1/4) + (2/15 x 3/4)] = 0.0427

Thus, ranking for q1 (top rank first): d1, d3, d4, d2
Ranking for q2 (top rank first): d2, d4, d1, d3

c. Which of the rankings – part a or part b – has higher MAP? Is corpus modelling advantageous in this context? **[4 + 4 + 2 = 10]**

Sol. Part a. AP for q1 = (1/1 + 2/3)/2 = 0.833
AP for q2 = (1/1 + 2/2)/2 = 1
MAP = (0.833 + 1)/2 = 0.917

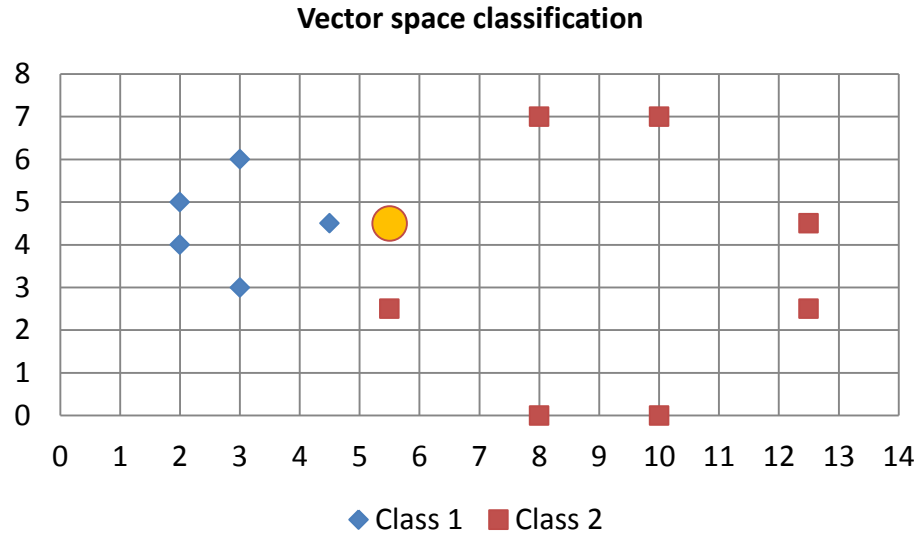
Part b. AP for q1 = (1/1 + 2/2)/2 = 1
AP for q2 = (1/1 + 2/2)/2 = 1
MAP = (1 + 1)/2 = 1.000

Thus, the ranking for part b has higher MAP.
Thus, corpus modeling is advantageous in this context.

Q. 3> Consider the following points P1 through P13 on a 2-D plane: (2, 4), (2, 5), (3, 3), (4.5, 4.5), (3, 6), (5.5, 4.5), (5.5, 2.5), (8, 0), (10, 0), (12.5, 2.5), (12.5, 4.5), (10, 7), and (8, 7). Note that the order and nomenclature of the points are important, i.e. P1 is (2, 4), P2 is (2, 5), P3 is (3, 3), and so on up to P13. Points P1 through P5 belong to class 1 and P7 through P13 belong to class 2.

a. Classify P6 using Rocchio classification.

Sol.



Vector	*x*	*y*
$\overrightarrow{P_1}$	2	4
$\overrightarrow{P_2}$	2	5
$\overrightarrow{P_3}$	3	3
$\overrightarrow{P_4}$	4.5	4.5
$\overrightarrow{P_5}$	3	6
$\overrightarrow{P_7}$	5.5	2.5
$\overrightarrow{P_8}$	8	0
$\overrightarrow{P_9}$	10	0
$\overrightarrow{P_{10}}$	12.5	2.5
$\overrightarrow{P_{11}}$	12.5	4.5
$\overrightarrow{P_{12}}$	10	7
$\overrightarrow{P_{13}}$	8	7
$\overrightarrow{\mu_{C_1}}$	2.9	4.5
$\overrightarrow{\mu_{C_2}}$	9.5	3.3571

Distance of test point P6 from $\overrightarrow{\mu_{C_1}} = |\overrightarrow{\mu_{C_1}} - \overrightarrow{P_6}| = \sqrt[2]{(2.9 - 5.5)^2 + (4.5 - 4.5)^2} = 2.6$

Distance of test point P6 from $\overrightarrow{\mu_{C_2}} = |\overrightarrow{\mu_{C_2}} - \overrightarrow{P_6}| = \sqrt[2]{(9.5 - 5.5)^2 + (3.3571 - 4.5)^2} = \sqrt[2]{16 + 1.3062} = 4.1601$

Thus, Rocchio assigns P6 to Class 1.

b. To which class would 3-NN classify P6?

$|\overrightarrow{P_1} - \overrightarrow{P_6}| = \sqrt[2]{(2 - 5.5)^2 + (4 - 4.5)^2} = \sqrt[2]{12.25 + 0.25} = 3.5355$
$|\overrightarrow{P_2} - \overrightarrow{P_6}| = \sqrt[2]{(2 - 5.5)^2 + (5 - 4.5)^2} = \sqrt[2]{12.25 + 0.25} = 3.5355$
$|\overrightarrow{P_3} - \overrightarrow{P_6}| = \sqrt[2]{(3 - 5.5)^2 + (3 - 4.5)^2} = \sqrt[2]{6.25 + 2.25} = 2.9155$
$|\overrightarrow{P_4} - \overrightarrow{P_6}| = \sqrt[2]{(4.5 - 5.5)^2 + (4.5 - 4.5)^2} = \sqrt[2]{1 + 0} = 1$
$|\overrightarrow{P_5} - \overrightarrow{P_6}| = \sqrt[2]{(3 - 5.5)^2 + (6 - 4.5)^2} = \sqrt[2]{6.25 + 2.25} = 2.9155$
$|\overrightarrow{P_7} - \overrightarrow{P_6}| = \sqrt[2]{(5.5 - 5.5)^2 + (2.5 - 4.5)^2} = \sqrt[2]{0 + 4} = 2$
$|\overrightarrow{P_8} - \overrightarrow{P_6}| = \sqrt[2]{(8 - 5.5)^2 + (0 - 4.5)^2} = \sqrt[2]{6.25 + 20.25} = 5.1478$
$|\overrightarrow{P_9} - \overrightarrow{P_6}| = \sqrt[2]{(10 - 5.5)^2 + (0 - 4.5)^2} = \sqrt[2]{20.25 + 20.25} = 6.3640$
$|\overrightarrow{P_{10}} - \overrightarrow{P_6}| = \sqrt[2]{(12.5 - 5.5)^2 + (2.5 - 4.5)^2} = \sqrt[2]{49 + 4} = 7.2801$
$|\overrightarrow{P_{11}} - \overrightarrow{P_6}| = \sqrt[2]{(12.5 - 5.5)^2 + (4.5 - 4.5)^2} = \sqrt[2]{49 + 0} = 7$
$|\overrightarrow{P_{12}} - \overrightarrow{P_6}| = \sqrt[2]{(10 - 5.5)^2 + (7 - 4.5)^2} = \sqrt[2]{20.25 + 6.25} = 5.1478$
$|\overrightarrow{P_{13}} - \overrightarrow{P_6}| = \sqrt[2]{(8 - 5.5)^2 + (7 - 4.5)^2} = \sqrt[2]{6.25 + 6.25} = 3.5355$

Thus, the three closest points to P6: P4, P7, P5 (or P3)
P4 and P5 (or P3) belong to class 1 and P7 belongs to class 2.
Thus, by majority rule, P6 is assigned to Class 1 by 3-NN.

c. Now assume that it is known that P6 belongs to class 2 while the class information for P7 is missing. Classify P7 using Rocchio. Is P7's class label the same as earlier? **[4 + 3 + 3 = 10]**

Vector	*x*	*y*
$\overrightarrow{P_1}$	2	4
$\overrightarrow{P_2}$	2	5
$\overrightarrow{P_3}$	3	3
$\overrightarrow{P_4}$	4.5	4.5
$\overrightarrow{P_5}$	3	6
$\overrightarrow{P_6}$	5.5	4.5
$\overrightarrow{P_8}$	8	0
$\overrightarrow{P_9}$	10	0
$\overrightarrow{P_{10}}$	12.5	2.5
$\overrightarrow{P_{11}}$	12.5	4.5
$\overrightarrow{P_{12}}$	10	7
$\overrightarrow{P_{13}}$	8	7

$\overrightarrow{\mu_{C_1}}$	2.9	4.5
$\overrightarrow{\mu_{C_2}}$	9.5	3.6429

Distance of test point P7 from $\overrightarrow{\mu_{C_1}}$ = $|\overrightarrow{\mu_{C_1}} - \overrightarrow{P_7}|$ = $\sqrt[2]{(2.9-5.5)^2+(4.5-2.5)^2}$ = $\sqrt[2]{6.76+4} = 3.2802$

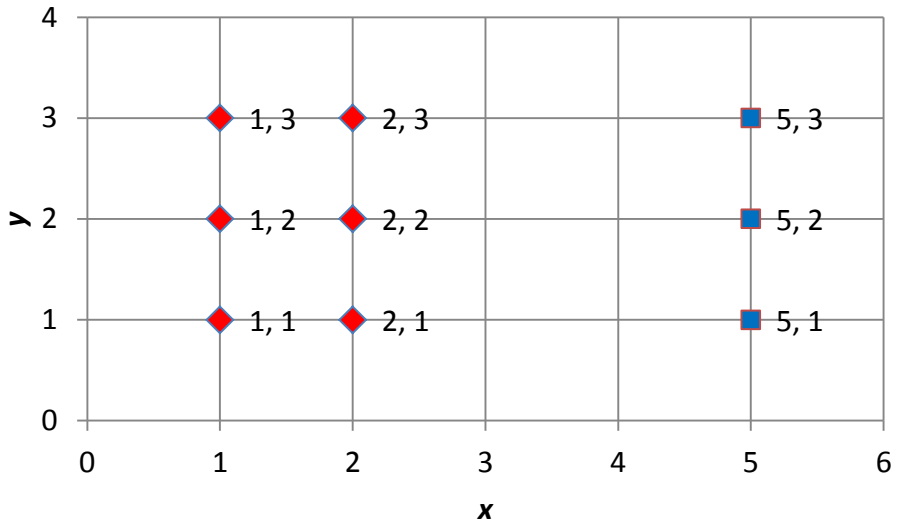
Distance of test point P7 from $\overrightarrow{\mu_{C_2}}$ = $|\overrightarrow{\mu_{C_2}} - \overrightarrow{P_7}|$ = $\sqrt[2]{(9.5-5.5)^2+(3.6429-2.5)^2}$ = $\sqrt[2]{16+1.3062} = 4.1601$

Thus, Rocchio assigns P7 to Class 1.

Thus, P7's class label has changed from Class 2 to Class 1.

You may wish to plot the points on *x*- and *y*-axes to have a better picture of the scenario.

Q. 4> Consider the following points P1 through P9 on a 2-D plane: (1, 3), (2, 3), (1, 2), (2, 2), (1, 1), (2, 1), (5, 3), (5, 2) and (5, 1). Note that the order and nomenclature of the points are important, i.e. P1 is (1, 3), P2 is (2, 3), P3 is (1, 2), and so on up to P9. It is known that points P1 through P6 are red and points P7 through P9 are blue.



a. Break the set of points into two clusters using the *k*-means algorithm, starting at P2 and P6. Stop when there is no change in cluster membership between iterations. In case of tied distances from centroids, select the centroid with lower *y*. If the *y*-coordinates also match, select the centroid with lower *x*.

Sol. Seed centroid vector 1 $\overrightarrow{\mu_{C_1}} = 2\hat{\imath} + 3\hat{\jmath}$

Seed centroid vector 2 $\overrightarrow{\mu_{C_2}} = 2\hat{\imath} + \hat{\jmath}$

For every point P_i (*i* = 1 to 9), let C1 and C2 be the two centroids at a particular iteration and let d(C1) and d(C2) be the Euclidean distances from these centroids respectively.

Point	(*x*, *y*)	Iteration 1					Point	(*x*, *y*)	Iteration 2				
		C1	d(C1)	C2	d(C2)	Assigned Cluster			C1	d(C1)	C2	d(C2)	Assigned Cluster
P1	(1, 3)	(2, 3)	1	(2, 1)	2.236	Cl 1	P1	(1, 3)	(2.667, 3)	1.667	(2.667, 1.5)	2.243	Cl 1
P2	(2, 3)	(2, 3)	0	(2, 1)	2	Cl 1	P2	(2, 3)	(2.667, 3)	0.667	(2.667, 1.5)	1.642	Cl 1
P3	(1, 2)	(2, 3)	1.414	(2, 1)	1.414	Cl 2	P3	(1, 2)	(2.667, 3)	1.944	(2.667, 1.5)	1.740	Cl 2
P4	(2, 2)	(2, 3)	1	(2, 1)	1	Cl 2	P4	(2, 2)	(2.667, 3)	1.202	(2.667, 1.5)	0.834	Cl 2
P5	(1, 1)	(2, 3)	2.236	(2, 1)	1	Cl 2	P5	(1, 1)	(2.667, 3)	2.604	(2.667, 1.5)	1.740	Cl 2
P6	(2, 1)	(2, 3)	2	(2, 1)	0	Cl 2	P6	(2, 1)	(2.667, 3)	2.108	(2.667, 1.5)	0.834	Cl 2
P7	(5, 3)	(2, 3)	3	(2, 1)	3.606	Cl 1	P7	(5, 3)	(2.667, 3)	2.333	(2.667, 1.5)	2.774	Cl 1
P8	(5, 2)	(2, 3)	3.162	(2, 1)	3.162	Cl 2	P8	(5, 2)	(2.667, 3)	2.538	(2.667, 1.5)	2.386	Cl 2
P9	(5, 1)	(2, 3)	3.606	(2, 1)	3	Cl 2	P9	(5, 1)	(2.667, 3)	3.073	(2.667, 1.5)	2.386	Cl 2

Iteration 2 C1 = (1 + 2 + 5)/3 $\hat{\imath}$ + (3 + 3 + 3)/3 $\hat{\jmath}$ = 2.667 $\hat{\imath}$ + 3 $\hat{\jmath}$
Iteration 2 C2 = (1 + 2 + 1 + 2 + 5 + 5)/6 $\hat{\imath}$ + (2 + 2 + 1 + 1 + 2 + 1)/6 $\hat{\jmath}$ = 2.667 $\hat{\imath}$ + 1.5 $\hat{\jmath}$

Cluster membership does not change between iterations 1 and 2 and hence we stop.

Final clusters: Cluster 1: P1, P2, P7; Cluster 2: P3, P4, P5, P6, P8, P9.

b. What is the NMI of this clustering output with respect to red and blue colouring? Show all steps of the computation. Use log base 2. **[5 + 5 = 10]**

You may wish to plot the points on *x*- and *y*-axes to have a better picture of the scenario.

Sol. $NMI(\Omega, C) = \frac{I(\Omega;C)}{[H(\Omega)+H(C)]\ /\ 2}$
Here Ω = $\{\omega_1, \omega_2\}$ are the clusters (cluster 1, cluster 2) and $C = \{c_1, c_2\}$ are the classes (red, blue).

$$I(\Omega; C) = \frac{2}{9}\log_2\frac{9*2}{3*6} + \frac{1}{9}\log_2\frac{9*1}{3*3} + \frac{4}{9}\log_2\frac{9*4}{6*6} + \frac{2}{9}\log_2\frac{9*2}{6*3}$$
$$= \frac{2}{9}\log_2 1 + \frac{1}{9}\log_2 1 + \frac{4}{9}\log_2 1 + \frac{2}{9}\log_2 1 = 0.000$$

Thus, NMI = 0.000/$([\boldsymbol{H(\Omega)} + \boldsymbol{H(C)}]\ /\ \mathbf{2})$ **= 0.000.**

[Optional calculations:

$$H(\Omega) = -\left[\frac{3}{9}\log_2\frac{3}{9} + \frac{6}{9}\log_2\frac{6}{9}\right] = -\left[\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}\right] = -\frac{1}{3}\left[\log_2\frac{1}{3} + 2\log_2\frac{2}{3}\right]$$
$$= -\frac{1}{3}\left[\log_2\frac{1}{3} + \log_2\frac{4}{9}\right] = -\frac{1}{3}\log_2\frac{4}{27} = -\frac{1}{3}\frac{\log_{10}\frac{4}{27}}{\log_{10}2} = \frac{-0.333 * -0.829}{0.301}$$
$$= 0.917$$

Similarly, $H(C) = -\left[\frac{6}{9}\log_2\frac{6}{9} + \frac{3}{9}\log_2\frac{3}{9}\right] = 0.917$]

Q. 5> Consider the set of six documents below to be your corpus.

doc-id	document text
1	*java coffee is the best coffee for programming*
2	*java is a good island*
3	*java programming with java coffee*
4	*programming with java*
5	*best island in java island*
6	*java programming is the best programming*

Use the stop list: *a, an, the, how, what, why, when, who, in, is, and, for, to, from, by, with, of, good, better, best*.

We use a vector space model. Assume that the index of a term is determined by its order of encounter in the log (*java* gets position 1 in the vector, *coffee* gets position 2, *programming* position 3, and so on). Using **simple TF-IDF** (use raw TF, multiplied by IDF(t) = $\log_{10}(N/\text{DF}(t))$) for **term weighting of documents**. Do **not** apply **any** normalization on the document weights. Use **Euclidean distance** as the distance between two vectors.

a. Build a doc-doc unnormalized Euclidean distance matrix and apply HAC using the *complete link* concept.

Sol. tf = term frequency, tw = term weight.

term	DF(t)	IDF(t)	doc-1 -tf	doc-1 -tw	doc-2 -tf	doc-2 -tw	doc-3 -tf	doc-3 -tw	doc-4 -tf	doc-4 -tw	doc-5 -tf	doc-5 -tw	doc-6 -tf	doc-6 -tw
java	6	0.000	1	0.000	1	0.000	2	0.000	1	0.000	1	0.000	1	0.000
coffee	2	0.477	2	0.954	0	0.000	1	0.477	0	0.000	0	0.000	0	0.000
programming	4	0.176	1	0.176	0	0.000	1	0.176	1	0.176	0	0.000	2	0.352
island	2	0.477	0	0.000	1	0.477	0	0.000	0	0.000	2	0.954	0	0.000

Thus, the document vectors can be represented as:
doc-id-1: (0, 0.954, 0.176, 0)
doc-id-2: (0, 0, 0, 0.477)
doc-id-3: (0, 0.477, 0.176, 0)
doc-id-4: (0, 0, 0.176, 0)
doc-id-5: (0, 0, 0, 0.954)
doc-id-6: (0, 0, 0.352, 0)

The (square symmetric) matrix of the pairwise Euclidean distances is given below:

	doc-id-1	doc-id-2	doc-id-3	doc-id-4	doc-id-5	doc-id-6
doc-id-1	0.000	1.081	0.477	0.954	1.361	0.970

doc-id-2		0.000	0.697	0.508	0.477	0.593
doc-id-3			0.000	0.477	1.081	0.508
doc-id-4				0.000	0.970	0.176
doc-id-5					0.000	1.017
doc-id-6						0.000

HAC = Hierarchical Agglomerative Clustering.

Complete link concept => Distance between clusters = minimum similarity between any member pair = Largest distance between any member pair.

The minimum distance in the matrix between different documents is 0.176 (d4, d6). They are agglomerated first. We now have

	doc-id-1	**doc-id-2**	**doc-id-3**	**doc-id-4 & 6**	**doc-id-5**
doc-id-1	0.000	1.081	0.477	0.970	1.361
doc-id-2		0.000	0.697	0.593	0.477
doc-id-3			0.000	0.508	1.081
doc-id-4 & 6				0.000	1.017
doc-id-5					0.000

The minimum distance in the matrix between different clusters is 0.477 (d1, d3) (or (d2, d5)). They are agglomerated second. We now have

	doc-id-1 & 3	**doc-id-2**	**doc-id-4 & 6**	**doc-id-5**
doc-id-1 & 3	0.000	1.081	0.970	1.361
doc-id-2		0.000	0.593	0.477
doc-id-4 & 6			0.000	1.017
doc-id-5				0.000

The minimum distance in the matrix between different clusters is 0.477 (d2, d5). They are agglomerated third. We now have

	doc-id-1 & 3	**doc-id-2 & 5**	**doc-id-4 & 6**
doc-id-1 & 3	0.000	1.361	0.970
doc-id-2 & 5		0.000	1.017
doc-id-4 & 6			0.000

The minimum distance in the matrix between different clusters is 0.970 (d1-d3 and d4-d6). They are agglomerated fourth. We now have

	doc-id-1,3,4,6	**doc-id-2 & 5**
doc-id-1,3,4,6	0.000	1.361
doc-id-2 & 5		0.000

The minimum distance in the matrix between different clusters is 1.361 (d1-d3-d4-d6 and d2-d5). They are agglomerated fifth. We now have

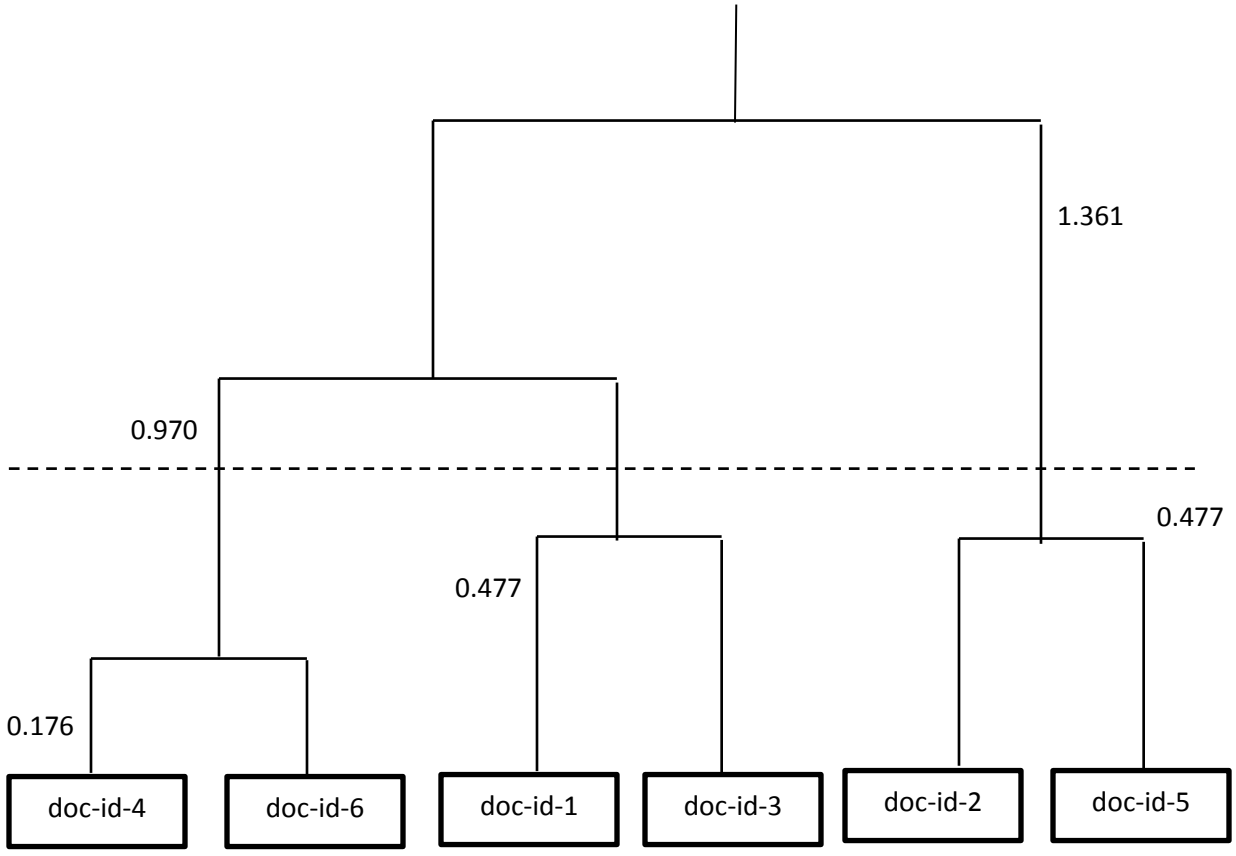
	doc-id-1,3,4,6,2,5
doc-id-1,3,4,6,2,5	0.000

This is the end of the HAC process.

P. T. O.

b. Draw the complete resultant dendrogram and mark the combination similarities at each join.

Sol. The combination similarity is the *unnormalized* Euclidean distance between clusters/documents, computed iteratively using the complete link concept.



c. Cut the dendrogram where the gap between two successive combination similarities is the largest. How many clusters do you get? List the elements of these clusters. **[7 + 1 + 2 = 10]**

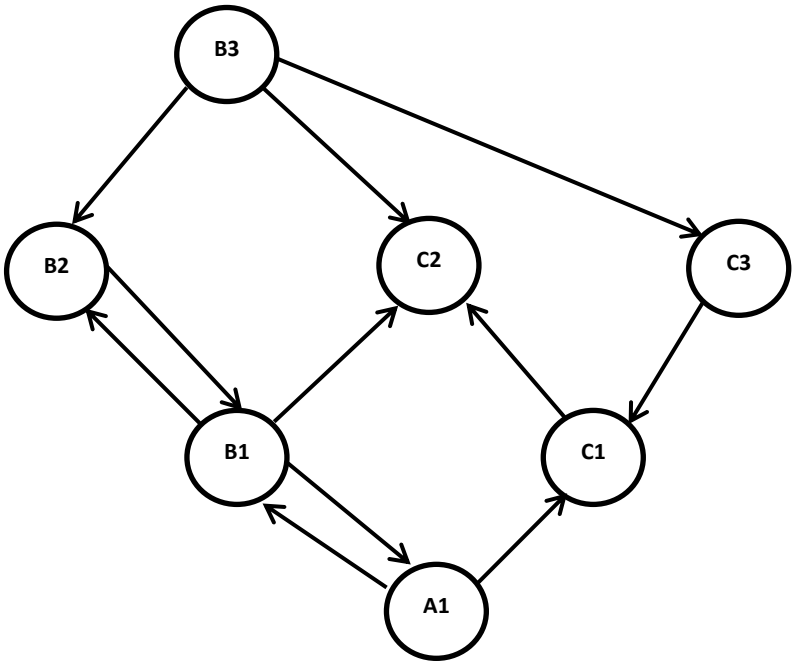
Sol. Gap between step 1 and step 2 = 0.477 – 0.176 = 0.301
Gap between step 2 and step 3 = 0.477 – 0.477 = 0
Gap between step 3 and step 4 = 0.970 – 0.477 = 0.493
Gap between step 4 and step 5 = 1.361 – 0.970 = 0.391

Thus, highest gap = 0.493 (dashed line in figure above)

There are 3 clusters.

Element listing:
Cluster 1: doc-id-4, doc-id-6
Cluster 2: doc-id-1, doc-id-3
Cluster 3: doc-id-2, doc-id-5.

Q. 6>



Consider the following toy Web graph, which shows webpages as labelled circles and their link structure using directed arrows. There are three domains in this graph: A, B and C. A has one webpage A1, B has three webpages B1, B2 and B3, and C has three webpages C1, C2 and C3. Suppose a crawler takes 100 milliseconds to fetch one webpage. *Politeness policy* states that a crawler cannot issue more than one page request to any domain in 50 milliseconds.

a. Crawler X must finish crawling all pages in a domain before switching to a different domain. What is the shortest time in which X can finish crawling all accessible pages if it observes the *Politeness Policy* and starts crawling on page A1?

Sol. Path to be followed: A1 -> B1 -> B2 -> C1 (or C2) -> C2 (B3, C3 inaccessible)
Required shortest time = 100 + 100 + 50 + 100 + 100 + 50 + 100 (wait between B1 and B2, and C1 and C2) = **600 ms.**

b. What is the total crawl time if X starts from B3? What important challenge faced by a crawler is being highlighted in this scenario?

Sol. Path to be followed: B3 -> B2 -> B1 -> C3 -> C2 -> C1 -> A1 (other orderings are also allowed, provided pages of a domain are crawled successively)
Required shortest time = 100 + 50 + 100 + 50 + 100 + 100 + 50 + 100 + 50 + 100 + 100 (wait between B3 and B2, B2 and B1, C3 and C2, C2 and C1) = 100(7) + 50(4) = **900 ms.**

The starting points (Webpages) for a crawler are very important. If the starting point(s) has poor connectivity, then crawl coverage will also be poor.

c. What is a common method for modeling the challenge in the previous problem in mathematical formulations of the Web graph?

Sol. Use of **teleportation** probabilities.

d. Let *A* is the adjacency matrix of any Web graph (or its subset), where *A* is a square matrix with one row and one column for each page. The entry A_{ij} is one if there is a hyperlink from page *i* to page *j*, and zero otherwise. What would the cells of AA^T represent? What about A^TA?

[3 + 4 + 1 + 2 = 10]

Sol. Cells of AA^T represent the number of common out-neighbours, i.e. cell[i][j] contains the number of common out-neighbours for page *i* and page *j*.

Cells of A^TA represent the number of common in-neighbours, i.e. cell[i][j] contains the number of common in-neighbours for page *i* and page *j*.

Q. 7> a. Find the eigenvalues and corresponding eigenvectors for the matrix $S = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$.

Sol.

$$\left|\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right| = 0$$

$$\Rightarrow \left|\begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix}\right| = 0$$

From the characteristic equation $|S - \lambda I| = 0$
we have the quadratic
$(2 - \lambda)^2 - 1 = 0$
$\Rightarrow 4 - 4\lambda + \lambda^2 - 1 = 0$
$\Rightarrow \lambda^2 - 4\lambda + 3 = 0$
$\Rightarrow \lambda^2 - 3\lambda - \lambda + 3 = 0$
$\Rightarrow \lambda(\lambda - 3) - 1(\lambda - 3) = 0$
$\Rightarrow (\lambda - 3)(\lambda - 1) = 0$
whose solutions yield the **eigenvalues 3 and 1.**

Now, for eigenvectors: $S\vec{x} = \lambda\vec{x}$. For λ = 3 case, $\left(\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} - 3\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\vec{x} = \vec{0}$

- ⇨ $\begin{bmatrix}-1 & 1\\ 1 & -1\end{bmatrix}\vec{x} = \begin{bmatrix}0\\0\end{bmatrix}$
- ⇨ Augmented matrix: $\begin{bmatrix}-1 & 1 & 0\\ 1 & -1 & 0\end{bmatrix}$ => R2 -> R2 + R1: $\begin{bmatrix}-1 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}$ => -x1 + x2 = 0 => x2 = x1, where $\vec{x} = \begin{bmatrix}x1\\x2\end{bmatrix}$
- ⇨ The eigenvector is then, $\begin{bmatrix}x1\\x2\end{bmatrix} = \begin{bmatrix}x1\\x1\end{bmatrix}$. Using x1 = 1, we get $\begin{bmatrix}1\\1\end{bmatrix}$

For λ = 1 case,

- ⇨ $\left(\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix} - 1\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\right)\vec{x} = \vec{0}$
- ⇨ $\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}\vec{x} = \begin{bmatrix}0\\0\end{bmatrix}$
- ⇨ Augmented matrix: $\begin{bmatrix}1 & 1 & 0\\ 1 & 1 & 0\end{bmatrix}$ => R2 -> R2 - R1: $\begin{bmatrix}1 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}$ => x1 + x2 = 0 => x2 = -x1, where $\vec{x} = \begin{bmatrix}x1\\x2\end{bmatrix}$
- ⇨ The eigenvector is then, $\begin{bmatrix}x1\\x2\end{bmatrix} = \begin{bmatrix}x1\\-x1\end{bmatrix}$. Using x1 = 1, we get $\begin{bmatrix}1\\-1\end{bmatrix}$.

The corresponding eigenvectors are thus $\begin{bmatrix}\mathbf{1}\\\mathbf{1}\end{bmatrix}$ and $\begin{bmatrix}\mathbf{1}\\\mathbf{-1}\end{bmatrix}$ (any multiple is also fine).

b. Since *S* is symmetric, what special property does its eigenvectors possess?

Sol. Its eigenvectors are orthogonal (dot product is zero).

c. Compute the eigen decomposition for *S.*

Sol. We use the matrix diagonalization theorem here which is applicable for any *MxM* square matrix *S* with *M* linearly independent eigenvectors. $S = U \Lambda U^{-1}$, where the columns of *U* are the eigenvectors of *S* and *Λ* is a diagonal matrix whose diagonal entries are the eigenvalues of *S* in decreasing order.

So $U = \begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$

$\Lambda = \begin{bmatrix}3 & 0\\ 0 & 1\end{bmatrix}$

$U^{-1} = \frac{1}{|U|}Adjoint(U) = \frac{1}{2}Adjoint\left(\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\right) = \frac{1}{2}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \begin{bmatrix}0.5 & 0.5\\ 0.5 & -0.5\end{bmatrix}$

Thus, the required decomposition:

$$\begin{bmatrix}\mathbf{2} & \mathbf{1}\\ \mathbf{1} & \mathbf{2}\end{bmatrix} = \begin{bmatrix}\mathbf{1} & \mathbf{-1}\\ \mathbf{1} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\mathbf{3} & \mathbf{0}\\ \mathbf{0} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\mathbf{0.5} & \mathbf{0.5}\\ \mathbf{0.5} & \mathbf{-0.5}\end{bmatrix}$$

d. Is the decomposition for the given *S* unique? Why or why not? **[3 + 1 + 5 + 1 = 10]**

Sol. Yes, this decomposition for *S* is unique because the eigenvalues of *S* are distinct.

Q. 8> Multiple choice questions on text classification. Provide your option with a brief reason (<= 3 sentences).

a. Which of the following are linear classifiers?
(A) Naïve Bayes and k-NN (B) Rocchio and k-NN (C) Naïve Bayes and Rocchio (D) All
Sol. (C) Naïve Bayes and Rocchio. They are linear classifiers because they classify based on a simple linear combination of the features.

b. What is the testing time complexity for multinomial Naïve Bayes text classifier (|C| is the number of priors and M_a is the number of types in the test document)?
(A) $\Theta(|C|^2 M_a)$ (B) $\Theta(|C| M_a)$ (C) $\Theta(|C|)$ (D) $\Theta(M_a)$

Sol. (B) $\Theta(|C| M_a)$. Assuming that the length of test documents is bounded, $\Theta(L_a + |C| M_a) = \Theta(|C| M_a)$ because $L_a < b|C| M_a$ for a fixed constant b.

c. The number of linear separators for two classes is:
(A) Infinite (B) Zero (C) Infinite or zero (D) Depends on the data points

Sol. (C) Infinite or zero. If there is one linear separator, then there is an epsilon ε such that after moving it by ε in the direction of the closest point, it is still a separator.

d. Which of the following has optimal running time complexity?
(A) Naïve Bayes and k-NN (B) Rocchio and k-NN (C) Naïve Bayes and Rocchio (D) None

Sol. (C) Naïve Bayes and Rocchio. In general, we have $|C||V| < |D|L_{ave}$, so both training and testing complexity are linear in the time it takes to scan the data. Because we have to look at the data at least once, naïve Bayes can be said to have optimal time complexity. Training and test time complexities of Rocchio classification are exactly the same.

e. Is the Naïve Bayes classifier optimal with respect to error rate on test data?
(A) Never (B) Always (C) Under specific assumptions (D) Depends on the data points **[2 x 5 = 10]**

Sol. (C) Under specific assumptions. It can be shown that naïve Bayes is an optimal classifier (in the sense of minimal error rate on new data) for data where independence assumptions hold.

Q. 9> a. What are the two key parameters that determine the rank score of an ad?

Sol. Bid and Click probability (AdQuality score is also correct).

b. What are the three steps involved in delivering an ad?

Sol. The three steps involved in delivering an ad are as follows:

- ⇨ Selecting qualified ads
- ⇨ Sort the selected ads using a quality score based threshold

⇨ Allocate the ads to the available slots and price

c. What are the three components of an ad?

Sol. The three components of an ad are as follows: Ad Title, Display URL, Ad Text (or Ad copy).

d. Which location does the user go to upon clicking an ad?

Sol. Destination URL.

e. Can a given ad have multiple bids by the same advertiser? **[2 + 3 + 3 + 1 + 1 = 10]**

Sol. Yes, since an advertiser can bid on multiple keywords. The selection is based on keywords bid by the advertiser.

Q. 10> While selecting ads beyond exact string matches what are ten different approaches used to expand/rewrite a query? Give an example for each approach. **[10]**

Sol.

1	Stopword removal	(apartments for low income earners -> apartments low income earners; remove stop words from query)
2	Spell correction	(xboyx -> xbox)
3	Singular/plural	(disney cruises -> disney cruise)
4	Stemming	(who has cheapest car insurance -> cheap car insurance: ability to trim ending of keywords, e.g. cheapest -> cheap)
5	*n*-gram based matching	(cheap car insurance -> car insurance or cheap car; though the latter is not a good expansion)
6	Location based expansion/rewrite	(cheap car insurance -> cheap car insurance in new york)
7	Smart expansion	(nokia phone -> samsung phone)
8	Extract related words from advertiser landing page	(countries continent africa -> map continent africa; italian restaurant -> pasta pizza etc.)
9	Query segmentation	(cheap vacations rental in south africa -> (cheap) (vacations rental) (south africa)
10	User preference-based or session-based information to expand	(wedding jewellery ->women jewellery)
