Name: ________________________________ Std # ________

National University of Computer & Emerging Sciences

(Karachi Campus)

Information Retrieval & Text Mining (CS567)

Final Examination – Fall 2015 (sol)

Course: IR&TM (CS567) Time Allowed: 180 Min.
Date: December 10, 2015 Max. Points: 100

Note: Attempt all questions. *Start each question on a new page of the answer book; answer all queries of the question in consecutive order. Answer to the point. Return this paper along with the answer book.*

Boolean Model+ Term Vocabulary+ Posting List +Tolerant IR

Question No. 1 [Time:25 min] [Points: 20]

a. During the course CS567, we studied many models for information retrieval, you are required to develop a list of desirable features/properties for an ideal model for IR. you need to come up with at least five features/properties, along with the justification for them. [5]

- An IR model should provide partial matching - documents and query cannot be often exactly matched.
- An IR model should provide ranked order - user are interested in best results for their queries and ranked order is better suited for them.
- An IR model should be simple, compact(small in term of space) and easy to compute - it is extensively applied on a large collection.
- An IR model should be adaptive to implicit /explicit feedback - feedback is useful in determining the actual need of a user.
- An IR model should handle contextual and semantics information - the result will be of high semantics quality.

b. Assume a bi-word index is used in an IR system. Give an example of a document which will be returned for a query of "National University FAST" but is actually a false positive which should not be returned. Suggest a solution without a false positive solution. [5]

Consider the following two text document:

D1: The leading computer science institution National University formally called University FAST, was very instrumental in fostering computer culture in Pakistan.

D2: National University FAST is a leading computer science school in Pakistan.

It is clearly evident that D2 contains the query text and is the true positive of this query. If a system uses bi-word index it will hit the D1 as well which is a false positive the answer to this problem is using a positional indexing to support this type of long query.

c. When can a wildcard query useful? Illustrate by giving 3 situational examples from search. [5]

Wildcard queries are used in any of the following situations: (1) the user is uncertain of the spelling of a query term (e.g., Sydney vs. Sidney, (2) the user is aware of multiple variants of spelling a term and (consciously) seeks documents containing any of the variants (e.g., color vs. colour); (3) the user seeks documents containing variants of a term that would be caught by stemming, but is unsure whether the search engine performs stemming (e.g., judicial vs. judiciary, leading to the wildcard query judicia*); (4) the user is uncertain of the correct rendition of a foreign word or phrase (e.g., the query Universit* Stuttgart).

d. What are the general methods of reducing the dictionary size in modern information retrieval system? [5]

The dictionary in modern information retrieval systems offer greater challenges like such as to control the size of the dictionary. Stemming and lemmatization are often applied to reduce the derivational variation of lexemes. Besides these implementers often opt for controlled vocabulary.

Vector Space Model

Question No. 2 [Time:25 min] [Points: 10]

Consider the following document collection consist of the four documents:

D={d1,d2,d3,d4}
d1= mary had a little lamb
d2= little lamb mary had
d3= mary went with lamb
d4= lamb went with mary

The collection vocabulary is given by V ={ a, had, lamb, little, mary, went, with} Assume that we use TF= 1+ log($tf_{t,d}$+1) for computing the term frequency of term t in document d. and IDF is define as log (N/df_t). we can use TF*IDF to represent every term in vector representation of document. **Give the document vectors for all four documents. Given a query vector q=<0,1,1,1,1,1,0> which document is most relevant and why?**

	d1	d2	d3	d4	q
a	1	0	0	0	0
had	1	1	0	0	1
lamb	1	1	1	1	1
little	1	1	0	0	1
mary	1	1	1	1	1
went	0	0	1	1	1
with	0	0	1	1	0

Now, TF and IDF scores

	TF					df	IDF
	d1	d2	d3	d4	q		
a	1.301	1	1	1	1	1	1.386
had	1.301	1.301	1	1	1	2	0.693
lamb	1.301	1.301	1.301	1.301	1.47	4	0
little	1.301	1.301	1	1	1.47	2	0.693
mary	1.301	1.301	1.301	1.301	1.47	4	0
went	1	1	1.301	1.301	1	2	0.693
with	1	1	1.301	1.301	1	2	0.693

	TF*IDF				
	d1	d2	d3	d4	q
a	0.180	1.386	1.386	1.386	1.386
had	0.901	0.901	0.693	0.693	0.693
lamb	0	0	0	0	0
little	0.901	0.901	0.693	0.693	1.018
mary	0	0	0	0	0
went	0.693	0.693	0.901	0.901	0.693
with	0.693	0.693	0.901	0.901	0.693

Sim(d1,q)= 2.751589

Sim(d2,q)= 4.423105
Sim(d3,q)= 4.355505
Sim(d4,q)= 4.355505

d2 is highly relevant.

IR Evaluation & Relevance Feedback

Question No. 3 [Time:25 min] [Points: 5+5]

a. An information system returned 150 documents from a collection of 400 under a given query. There were 70% non-relevant documents in the retrieved result. The collection contains 35% relevant documents. Find the Precision and Recall for this system. [5]

	Rel	Non-Rel
Ret	45	105
Non-Ret	95	155

Total Returned Docs. =150 ; 70% non-relevant that Non-relevant that were returned are 150X0.7= 105
The collection contains 35% relevant that is 400X0.35 =140 relevant documents, the query returned 45 relevant (150-105); 155 Non-relevant were not retrieved.
Precision = 45/150 = 0.3
Recall= 45/140 = 0.321

b. The following list of Rs and Ns represents relevant (R) and non-relevant (N) returned documents in a ranked list of 20 documents retrieved in response to a query from a collection of 10,000 documents. The top of the ranked list (the document the system thinks is most likely to be relevant) is on the left of the list. This list shows 6 relevant documents.

Assume that there are 8 relevant documents in total in the collection.

R R N N N N N N R N R N N N R N N N N R

Assume, now, instead, that the system returned the entire 10,000 documents in a ranked list, and these are the first 20 results returned, which contained only 6 relevant from 8.

1. What is the largest possible MAP that this system could have? [2.5]

Largest possible MAP will be when the system return the two left over relevant documents next to these 20, that is relevant documents at 21st and 22nd positions. Hence the MAP will be

MAP max = 1/8 X [1/1+2/2+3/9+4/11+5/15+6/20+7/21+8/22]

MAP max = 0.5034

2. What is the smallest possible MAP that this system could have? [2.5]

Smallest possible MAP will be when the system return the two left over relevant documents as the last documents from the collection. That is at place 9,999 and 10,000
Hence the MAP will be

MAP max = 1/8 X [1/1+2/2+3/9+4/11+5/15+6/20+7/9999+8/10000]

MAP max = 0.4164

Probabilistic & Language Model

Question No. 4 [Time:25 min] [Points: 15]

Consider the document collection given in Question 2 once again:
D={d1,d2,d3,d4}

d1= mary had a little lamb
d2= little lamb mary had
d3= mary went with lamb
d4= lamb went with mary

Now use probabilistic model of IR to represent these documents in probabilistic vector form. **Using the probabilities term values of each term, find the similarity of each document with the query "mary went". Which document is most similar to the query?**

N= number of documents = 4
$P(N_w) = (N- N_w + 0.5) / (N_w + 0.5)$

	a	had	lamb	little	many	went	with
N_w	1	2	4	2	4	2	2
$P(N_w)$	2.33	1	0.11	1	0.11	1	1

RSV(d1,q) = products of all terms common between document and query = (0.11)
RSV(d2,q) = (0.11)
RSV(d3,q) = (0.11)
RSV(d4,q) = (0.11)

all documents are at the same rank (equally relevant.)

Text Classification

Question No. 5 [Time:25 min] [Points: 15]

Consider the following examples for the task of text classification

Dataset	**DocID**	**Features- Words in documents**	**Class Fruit=Yes/No**
Training set	1	Orange, Orange, Lemon	No
	2	Orange, Red, Blue	No
	3	Apricot, Apple, Mango	Yes
	4	Apple, Banana , Orange	Yes
	5	Apple, Orange, Melon	Yes
Test set	6	Orange, Mango, Melon	?
	7	Orange, Red, Lemon	?

a. Using the training data first calculate the class prior probabilities? [5]

P(Fruit=yes) = (# of instances of Fruit=yes) / total # of instances = 3/5= 0.6

P(Fruit=no) = (# of instances of Fruit=no) / total # of instances = 2/5= 0.4

b. Using Multinomial Naïve Bayes to estimate the probabilities of each term (feature), that you will be using for doing part c? [5]

As there are only 5 terms in the testing documents so we only need to find the class specific probabilities of these five terms:

we have general formula $P(w_i/class) = (count(w_i/class) + 1) / (count (w/class)+ |V|)$

P(orange / F) = 2+1 / 9+9 = 0.167	P(orange/~F) = 3+1 / 6+9 =0.267
P(mango / F) = 0.111	P(mango /~ F) = 0.067
P(melon / F) = 0.111	P(melon / ~F) = 0.067
P(red / F) = 0.056	P(red / ~F) = 0.133
P(lemon/ F) =0.056	P(lemon/~ F) =0.133

c. Apply the Multinomial Naïve Bayes to classify the given test instance(s)? [5]

DocID = 6

P(Fruit/D6) = (0.6) X (0.167) X (0.111) X (0.111) = 0.0012

P(~Fruit/D6) = (0.4) X (0.267) X (0.067) X (0.067) = 0.0004

P(Fruit/D6) > P(~Fruit/D6) => D6 belong to class Fruit.

DocID = 7

P(Fruit/D7) = (0.6) X (0.167) X (0.056) X (0.056) = 0.0003

P(~Fruit/D7) = (0.4) X (0.267) X (0.133) X (0.133) = 0.0018

P(~Fruit/D7) > P(Fruit/D7) => D7 belong to class ~Fruit.

Text Clustering

Question No. 6 [Time:25 min] [Points: 5+5]

a. Consider a fictitious document collection. There are 6 documents in this collection and these are represented as 6 points in two-dimensional vector space as follow:

D1	(2,1)
D2	(1,1)
D3	(4,1)
D4	(1,2)
D5	(2,2)
D6	(4,2)

Suppose that the distance between a pair of documents is measured by the Euclidean distance between their corresponding points. Show how the k-means (single link) algorithm (with k=2) clusters these documents, using D1 and D6 as seeds. [5]

C1 = D1 and C2=D6 are initial seeds for K-mean algorithm, lets calculate distance of each document with these centers.

Distance (C1,D1) =0;
Distance (C1,D2) =1;
Distance (C1,D3) =2;
Distance (C1,D4) =1.41;
Distance (C1,D5) =1;
Distance (C1,D6) =2.23;

Distance (C2,D1) =2.23;
Distance (C2,D2) =3.16;
Distance (C2,D3) =1;
Distance (C2,D4) =3;
Distance (C2,D5) =2;
Distance (C2,D6) =0;

C1 has { D1,D2,D4,D5}, calculating the center we get C1=(1.5,1.5)
C2 has {D3,D6}, calculating the center we get C2=(4,1.5)

C1 =(1.5,1.5) and C2=(4,1.5) now for second iteration of the algorithm, lets calculate the distance from the centers again.

Distance (C1,D1) =0.707;	Distance (C2,D1) =3.041;
Distance (C1,D2) =0.707;	Distance (C2,D2) =2.061;
Distance (C1,D3) =2.549;	Distance (C2,D3) =0.5;
Distance (C1,D4) =0.707;	Distance (C2,D4) =3.041;
Distance (C1,D5) =0.707;	Distance (C2,D5) =2.061;
Distance (C1,D6) =2.549;	Distance (C2,D6) =0.5;

C1 with { D1,D2,D4,D5} and C2 with {D3,D6} have no change in the assigned members. we stopped here.

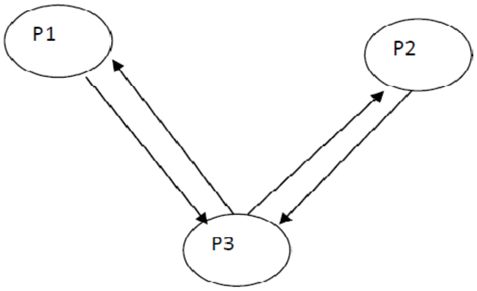
b. Two of the possible termination conditions for K-means were (1) assignment does not change, (2) centroids do not change. Do these two conditions imply each other? Explain. [5]

Yes, the two conditions implicate each other. If there is no change in the assignments of clusters membership for two consecutive iteration of the algorithm, then the average will be calculated from the same group of documents. Hence it also does not change.

Web Search + Web Crawler + Link Analysis

Question No. 7 [Time:30 min] [Points: 10+10]

a. Consider the following graph, which represents different web-pages that are linked together. You are required to calculate the page rank of each page using PageRank algorithm. You can use any method to solve the combination of equations. [5]



Let d=0.8 , we will have the general equation for each page as below

PR(P1) = 0.2 + 0.8 (PR(P3)/2)
PR(P2) = 0.2 + 0.8 (PR(P3) / 2)
PR(P3) = 0.2 + 0.8 (PR(P1) + PR(P2))

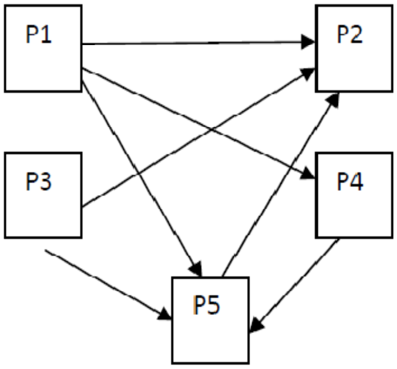
These equations can easily be solved. We get the following PageRank values for the given pages:

PR(P1)= 0.776
PR(P2) = 0.776
PR(P3) = 1.448

b. Outline the key differences between HITS and PageRank algorithms. [5]

HITS	PageRank
It gives 2 scores Hub and Authority for each page.	It gives one score e.g. PageRank.
It is executed at query time	It is executed at indexing time.
Not robust against spams.	Robust against web-spams.
Never favor pages, but can be manipulated.	Favor old pages.
It is query dependent	It is query independent

c. Consider another example of web graph, given below:



using the HITS algorithm calculate the Hub and Authority scores for each page by running the algorithm for one complete iteration. [10]

Consider the adjacency matrix

$$A= \begin{array}{|c|c|c|c|c|} \hline 0 & 1 & 0 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 & 0 \\ \hline \end{array}$$

initially h = [1 1 1 1 1]
a = [1 1 1 1 1]

Iteration 0: updating hub and authority scores

$$h^0 = \begin{array}{|c|c|c|c|c|} \hline 0 & 1 & 0 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 1 & 0 & 0 & 0 \\ \hline \end{array} \begin{array}{|c|} \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline \end{array} = \begin{array}{|c|} \hline 3 \\ \hline 0 \\ \hline 2 \\ \hline 1 \\ \hline 1 \\ \hline \end{array}$$

$$a^0 = \begin{array}{|c|c|c|c|c|} \hline 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 1 & 0 \\ \hline \end{array} \begin{array}{|c|} \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline \end{array} = \begin{array}{|c|} \hline 0 \\ \hline 3 \\ \hline 0 \\ \hline 1 \\ \hline 3 \\ \hline \end{array}$$

normalizing the vectors using Euclidean length normalization, we will get

h^0	[0.774, 0, 0.516, 0.258, 0.258]
a^0	[0, 0.688, 0, 0.229, 0.688]

<The End.>