Name: _____ Std # _____

National University of Computer & Emerging Sciences

Midterm Examination – Fall 2014(Sol)

Course: IR&TM (CS567) Time Allowed: 1 Hour
Date: October 13, 2014 Max. Marks: 40

Instructions: Attempt all question. Be to the point. Draw neat and clean diagram/code where necessary. Answer each question on the space provided, no marks for junk explanations. You must address all inquires in a question.

Question No. 1 Answer the following short questions. [10] [Time: 15 min.]

a. Enunciate the limitations of Boolean Model for Information Retrieval?[2.5]

-The Boolean model predicts that each document is either relevant or irrelevant. There is no notation for a partial match to the query.
- User must be aware of Boolean model and their connective in Boolean logic sense.
- Exact matching may leads to retrieval of too few or too many documents.
- It is difficult to rank the output, since all matched documents logically satisfy the query.
- It is difficult to perform relevance feedback.

b. What do we mean by tolerant retrieval? Give at least three example cases of tolerant retrieval, which we came across in web searches? [2.5]

- Tolerant retrieval means the information retrieval system deliver you best possible answer on the submission of query by compensating the error in the query both syntax and semantics and guide the users for effectively retrieving information process through information retrieval system.
- Examples of tolerant retrieval from web searches are :
1. Spelling suggestion for query word
2. Wild card support
3. Context sensitive information retrieval
4. Search options based on statistical results of the systems.

c. Differentiate between Lemmatization and Stemming? [2.5]

- Stemming is a heuristic approach in which we chop some of the characters of the token to produced inflectional forms of the lexemes.
- Lemmatization is the process of reducing several inflectional form of the token to the common base token; in the process we used morphological analysis and language understanding to reduce the token.
- When time is a constraint we must use stemming as it does not requires thoroughly language analysis but its results are not very good.

d. In an IR system that uses zone scoring with three zones with weights g1= 0.2, g2=0.3 and g3=0.5, what will be the distinct score values a document may get? [2.5]

Possible values for distinct scores are as below:
Score 0: corresponds to a situation where term is not found in any zones.
Score 1: corresponds to a situation where term is found in all zones.
Besides these values we can have 0.2, 0.3, 0.5, 0.7, and 0.8
There will be 7 distinct scores as 0.5 will be for two combinations g3 or (g1,g2)

Question No. 2 Vector Space Model for IR [20] [Time: 30 min.]

a. Consider a corpus C that consists of the following three documents:

D1: dil dil Pakistan, jan jan Pakistan

D2: Pakistan hum sub ki jan

D3: dil aur jan Pakistan Pakistan

Assuming that the term frequencies are normalized by the maximum frequency in a given document, calculate the TF-IDF weighted term vectors for all documents in C. Assume that the words in the vectors are ordered alphabetically? [8]

N=3	d1	d2	d3	tf	idf
aur	0	0	1	1	Log(3/1)
dil	2	0	1	3	Log(3/2)
hum	0	1	0	1	Log(3/1)
jan	2	1	1	4	Log(3/3)
ki	0	1	0	1	Log(3/1)
pakistan	2	1	2	5	Log(3/3)
sub	0	1	0	1	Log(3/1)

tf*idf vectors for documents are as below:

D1=<0,0.352,0,0,0,0,0>
D2=<0,0,0.477,0,0.477,0,0.477>
D3=<0.477,0.176,0,0,0,0,0>

b. For the above corpus C, consider a query "dil jan Pakistan". Calculate the TF-IDF weighted query vector for this query. [4]

q=<0,0.176,0 ,0,0,0,0>

c. Using the cosine similarity measure, calculate the similarity of the query q with all documents in the collection. Assume that term frequencies are normalized by the maximum frequency in given query. [8]

Sim(D1,q)=cos(D1,q) = Dot-Product (D1,q) / (|D1|) x (|q|) = 0.250
Sim(D2,q)=cos(D2,q) = Dot-Product (D2,q) / (|D2|) x (|q|) = 0.000
Sim(D3,q)=cos(D3,q) = Dot-Product (D3,q) / (|D3|) x (|q|) = 0.347

Question No. 3 Tolerant Retrieval [10] [15 min.]

a. If you wanted to search "L*ve" as a wildcard query, based on permuterm index, what key value(s) you will be searching for this? [3]

We know it is of the form X*Y, so we need to search ve$L* as a key based on permuterm index.

b. What sort of index is good for phrase queries (e.g. information retrieval)? Justify your answer. [3]

Positional index is a good choice for accessing text based on Phrase queries of the form e.g. information retrieval. We can use positional index through which we can access two positions like i and i+1 which contains these two words.

c. What are the problems associated with tokenization of text based on some natural language? [4]

Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. It is very challenging aspect of information retrieval, the tokenization process need to decide about a lot of different aspect of a natural language like:

1. Direction of parsing for tokenization.
2. Should it treat space as separator for token e.g. Les Vegas a single or two tokens.
3. Treatment of punctuation characters like hyphen(-) co-ordinated.
4. If there is no space in between word boundary, how it will decide about tokens like in Japanese.