Birla Institute of Technology & Science, Pilani
Work Integrated Learning Programmes Division
Second Semester 2023-2024

Mid-Semester Test
(EC-2 Regular)

Course No.	: AIMLCZG537/DSECLZG537
Course Title	: Information Retrieval
Nature of Exam	: Closed Book
Weightage	: 30%
Duration	: 2 Hours
Date of Exam	: 20/07/2024 (AN)

No. of Pages = 2
No. of Questions = 2

Note to Students:
1. Please follow all the *Instructions to Candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made, if any, should be stated clearly at the beginning of your answer.

Q.1. (a) What is Term Frequency? Does Boolean retrieval system require Term Frequency? Justify your answer. [2]

Term frequency is the number of occurrences of a term in a document (tf).
Boolean retrieval does not rank the documents. Hence tf is not required.

(b) Compute the minimum edit distance using Levenshtein algorithm, between the terms **HEAP** and **WEEP.** Fill in the table given below by distances between all prefixes as computed by the algorithm. . [3]

Table construction – 2.5 marks

		W	E	E	P
	0	1	2	3	4
H	1	1	2	3	4
E	2	2	1	2	3
A	3	3	2	2	3
P	4	4	3	3	2

Minimum edit distance between Heap and WEEP is 2 --- 0.5 marks

(c) Assume an IR application is designed to answer questions based on the document corpus. Which is better option to include in the preprocessing of the documents, Stemming or Lemmatization? Justify with an example.

[2]

Lemmatization preserves the contextual meaning of words better than stemming, making it more suitable for question-answering systems where understanding the exact meaning of a word is crucial. – 1 mark

Example: In a system designed to answer questions based on a corpus of medical texts, lemmatization ensures that words like "better" and "best" are accurately reduced to "good," preserving the meaning in context. – 1 mark [any justifying example]

crackbitswilp.in

(d) Assume a biword index. Give an example of a document which will be returned for a query of **"artificial intelligence applications"** but is a false positive which should not be returned. [only relevant answers will be awarded mark]. [1]

One possible answer:
……NLP is an application of artificial intelligence. Applications of NLP are……

(e) What is the Soundex code for the following two terms, ***Square*** and **Smoke**? Assume that the alphabets are mapped to numbers as follows: *(B, F, P, V → 1), (C, G, J, K, Q, S, X, Z → 2), (D,T → 3), (L → 4), (M, N → 5)* and *(R → 6)* . **Show the computations clearly** (Mere answers will not fetch marks) . [2]

Square- S20060 ; S and Q belong to same code , hence code is S600
Smoke – S5020; Code is S520

[2+3+2+1+2=10]

Q.2. (a) What is Heap's Law? . How is it useful for Inverted Index compression in Information Retrieval? [2]

Heaps' law: $M = kT^b$
Where M is the size of the vocabulary; T is the number of tokens in the collection.Estimates vocabulary size as a function of the collection size - 1 mark

In inverted index compression it is used for estimating the number of terms. - 1 mark

(b) Given below is a snapshot of dictionary terms; Show how Front coding compression is applied on this with a block size of 4, with proper explanation. [3]

Battalion ;Batten; Batter; Battered; Battery; Batting; Battle; Battleship; Batty
9batt*alion 1;en 2; er 3; ered ; 7batt*ery 1; e 3; ing 2; *le 3; leship; 5 batty

(c) What happens when the block size is increased in dictionary compression? Discuss with an example. [2]

Increasing the block size will increase the search time. It will be lograthimic plus Linear Search -- - 1 mark
Example - - 1 mark

(d) Give the **Variable Byte encoding** for the following postings list of the index term;

random : 100, 109, 130, 345 [3]

11100100 -100
11101101 - 109
00000001 1000010 -130
00000001 11011001-345

crackbitswilp.in

[2+3+2+3=10]

Q.3. (a) What are the issues in using Jaccard coefficient for scoring? [2]

It doesn't consider *term frequency* (how many times a term occurs in a document)

Rare terms in a collection are more informative than frequent terms. Jaccard doesn't consider this information

Length of the document affect the result. No length normalization

All three has to be given

(b) Assume a collection statistic of N (Total No. of Documents)= 200,000, T(number of Tokens) = 500,000. For n = 12 splits, r=5 segments, j= 3 partitions, in a distributed Index creation using MapReduce architecture,

i) What is the minimum number of parser machines needed ?

ii) What is the size of split allotted to the parser? [2]

i) Minimum 5 (no of segments) are needed.

ii) Total tokens/12= 500,000/12= 41667 approx./split

(c) Consider the following documents: [6]

For each document the terms along with their term frequencies are given.

Doc1: Data (40), analysis (10), algorithms (20)

Doc2: Data (55), mining (100), techniques (80)

Doc3: Data (100), warehousing (45), algorithms (45)

Rank the above documents for the query **"Data mining Data warehousing "** using Cosine Similarity between the query and documents. Assume logarithmic term weighting for query and documents, no **idf** weighting for both query and documents and apply cosine normalization for both query and documents.

[2+2+6 =10]

	Doc1			Doc2			Doc3			Query		
Terms	tf-raw	tf-wt	n'lize	tf-raw	tf-wt	n'lize	tf-raw	tf-wt	n'lize	tf-raw	tf-wt	n'lize
Data	40	2.6	0.65	55	2.74	0.55	100	3	0.63	2	1.3	0.68
analysis	10	2	0.5	0	0	0	0	0	0	0	0	0
algorithms	20	2.3	0.58	0	0	0	45	2.65	0.55	0	0	0
mining	0	0	0	100	3	0.6	0	0	0	1	1	0.52
techniques	0	0	0	80	2.9	0.58	0	0	0	0	0	0
warehousing	0	0	0	0	0	0	45	2.65	0.55	1	1	0.52

crackbitswilp.in

Cosine Similarity between (doc1 , q) = 0.65*0.68 = 0.442

Cosine Similarity between (doc2 , q) = 0.55*0.68 + 0.6*0.52 = 0.686

Cosine Similarity between (doc3 , q) = 0.63*0.68 + 0.55*0.52 = 0.714

The ranking of documents is doc3, doc2,doc1.

Tf-idf Computation for each document and query (1* 4 = 4)

Similarity computation between each document and query (0.5*3= 1.5)

Ranking the documents – 0.5

crackbitswilp.in