CS533 Information Retrieval Systems HW#6

Sefa Kılıç

May 7, 2010

[Q1] Consider the following D matrix.

$$D = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

What are the term discrimination values of the terms of the D matrix? Find the values using the

(a) cover coefficient concept (for calculating no. of clusters use the approximate formula implied by the indexing-clustering relationships: $(m \times n)/t$).

The number of clusters, $n_c = \frac{m \times n}{t} = 2$, where m is the number of documents, n is the number of terms and t is the number of non-zero elements in D.
Without t_1 (or t_2), corresponding D matrix is

$$D = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

So, the number of clusters without t_1 (or t_2), $n_{cl} = \frac{m \times (n-1)}{t-2} = 2$.
Without t_3 (or t_4), corresponding D matrix is

$$D = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

So, the number of clusters without t_3 (or t_4), $n_{cl} = \frac{m \times (n-1)}{t-2} = 2$.
So, $TDV_l = n_c - n_{cl} = 0$ for $1 \leq l \leq 4$.

(b) similarity concept where similarity is obtained using the Dice coefficient (use the approximate method based on the collection centroid approach).

Corresponding S matrix is

$$S = \begin{bmatrix} 1 & 1 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 1 \\ & & & 1 \end{bmatrix}$$

Average similarity value $= Q = (S_{12} + S_{13} + S_{14} + S_{23} + S_{24} + S_{34})/6 = 1/3$.
Without t_1 (or t_2), corresponding D and S matrices are

$$D = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad S = \begin{bmatrix} 1 & 1 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 1 \\ & & & 1 \end{bmatrix}$$

So, average similarity value without t_1 (or t_2) $= Q_1 = Q_2 = 1/3$.
Without t_3 (or t_4), corresponding D and S matrices are

$$D = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \qquad S = \begin{bmatrix} 1 & 1 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 1 \\ & & & 1 \end{bmatrix}$$

So, average similarity value without t_3 (or t_4) $= Q_3 = Q_4 = 1/3$.
$TDV_j = Q_j - Q = 0$ for $1 \leq j \leq 4$.

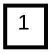
[Q3] PAT tree questions.

(a) Create the PAT tree for the following bit string: `011011100010111000`. What is the associated PAT array? (Create the tree for the first 12 sistrings.)

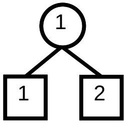
```
Si(1): 011011100010111000
Si(2): 11011100010111000
Si(3): 1011100010111000
Si(4): 011100010111000
Si(5): 11100010111000
Si(6): 1100010111000
Si(7): 100010111000
Si(8): 00010111000
Si(9): 0010111000
Si(10): 010111000
Si(11): 10111000
Si(12): 0111000
```

PAT tree after insertion of each sistring is given below.

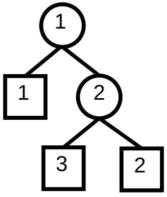
Si(1):



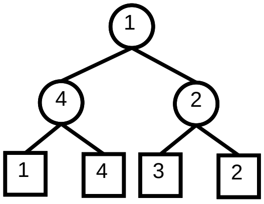
Si(2):



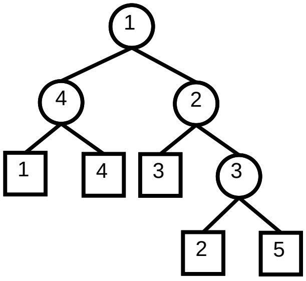
Si(3):



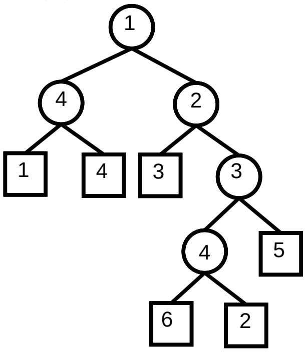
Si(4):



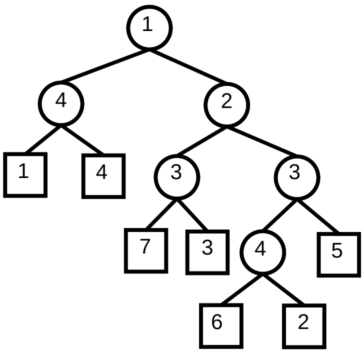
Si(5):



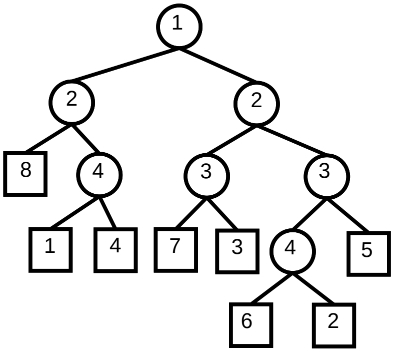
Si(6):



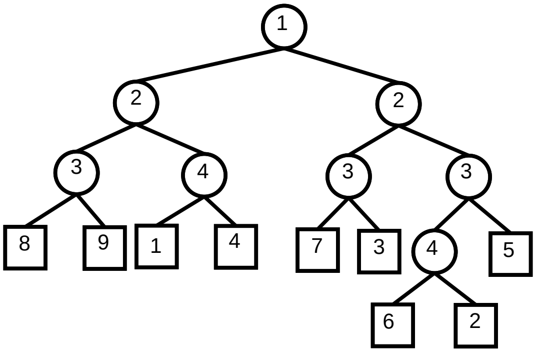
Si(7):



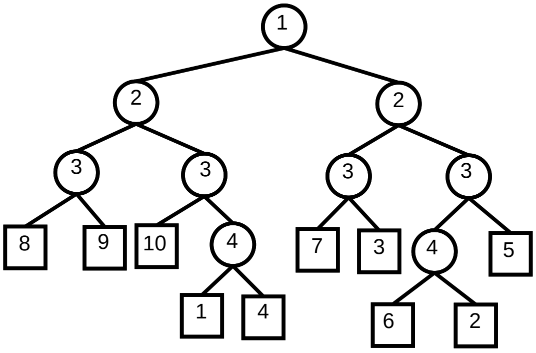
Si(8):



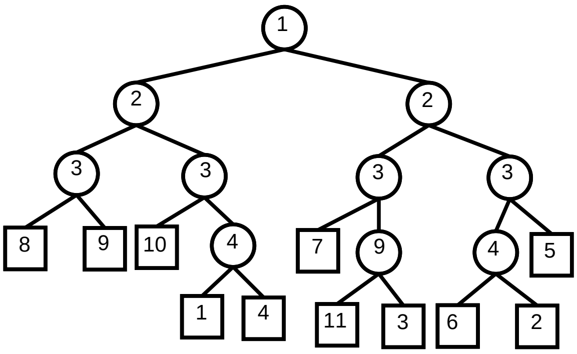
Si(9):



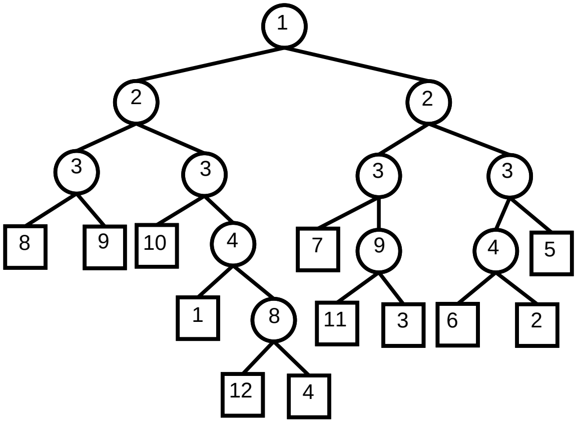
Si(10):



Si(11):



Si(12):



(b) Explain how to use the PAT tree concept to answer a query such as the following: `A <max 20> B`. Here `A` and `B` represent two different words and `<max 20>` indicates the condition that between `A` and `B` there can be at most 20 characters.

To answer such a query, proximity searching algorithm on the PAT tree can be used [1]. First, search for `A` and `B`. Then, sort by position the smaller of the two answers. Finally, traverse the unsorted answer set, searching every position in the sorted set, and check if the distance between positions and order satisfies the condition.

[Q5] In the database environment (a database containing 10240 objects (e.g., documents)) consider a query with 5 bit positions equal to one. These bit positions are 1, 2, 50, 51, 140. (The leftmost position of a signature is bit position 1.) For filtering (i.e., for query signature - document signatures matching) how many pages (disk blocks) need to be accessed in the case of SSF (sequential signature file) and BSF (bit sliced signature file)? Use the following assumption: in SSF the document signatures are placed in the pages one after the other (at the beginning we have the signature of the first document and immediately after that we have the document signature of the second document); in BSF a similar signature placement technique is used for the bit slices (first bit slice is immediately followed by the second bit slice, etc.). Note that signature of a document (or a bit slice) may cross the page boundaries. (Page size is 0.25 Kbytes.)

In SSF method, we need to compare query signature with each document signature and access the documents of the matching signatures. Database contains 10240 documents and each document signature requires 192 bits. Page size is 0.25 Kbytes. So, total number of pages to be accessed $\lceil 10240 \times 192 \ / \ 2^{11} \rceil = 960$.

In BSF, signature matrix is stored column-wise. For query processing, unlike SSF method, only a subset of signature matrix is to be accessed. Columns to be accessed are C_1, C_2, C_{50}, C_{51}, C_{140}. Each column C_i consists of 10240 bits ($i = 1 \ldots 192$). Every five pages store one column C_i. 10240 / $2^{11} = 5$. So, total number of pages to be accessed is 25.

[Q7] Consider the following signatures.

S_1: `0110 1100`

S_2: `1010 0011`

S_3: `0011 1100`

S_4: `0000 1111`

S_5: `1011 0100`

S_6: `0100 1011`

(a) Use the fixed prefix method to partition the above signatures. Take k (key length) as 2.

00: S_3, S_4.
01: S_1, S_6.
10: S_2, S_5.
11: –

(b) Now consider the following queries.

Q_1: `0001 1110`

Q_2: `1001 1100`

Q_3: `0011 0011`

Q_4: `1100 1100`

For each query indicate the partitions selected and also give the PAR (partition activation ratio: no. of partitions activated / total no. of partitions) and SAR (signature activation ratio: no. of signatures activated / total no. of signatures).

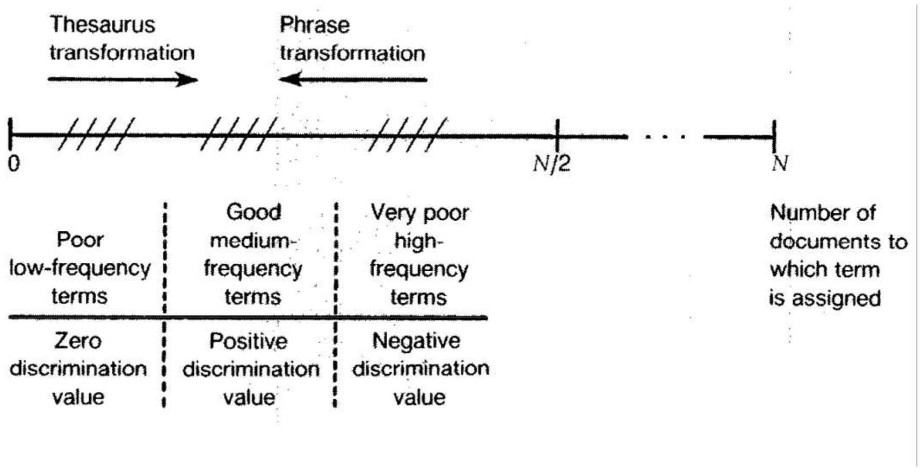
Working sets of query: a set of partitions $\{P_i\}$ such that $P_{key} \cap Q_k = Q_k$.
PAR = number of partitions activated / total number of partitions
SAR = number of signatures activated / total number of signatures

	Query signature	# partitions	PAR	SAR
Q_1	`0001 1110`	4 (all)	4/4	6/6
Q_2	`1001 1100`	2 (10, 11)	2/4	2/6
Q_3	`0011 0011`	4 (all)	4/4	6/6
Q_4	`1100 1100`	1 (11)	1/4	0/6

[Q11] Consider the paper [2]. Briefly explain the thesaurus and phrase transformations explained in the paper. Explain their purpose.

The best content identifiers are terms occurring neither too rarely nor too frequently. Document terms with very high frequency should be moved to the left on the document frequency spectrum by transforming them into phrases. A phrase such as "information retrieval" has lower frequency value than "information" or "retrieval" alone.

Terms with very low frequencies should be moved to the right on the document frequency spectrum. This can be done by transforming semantically similar terms into a common term. So, the frequency of this common term will be more than every transformed term.


Thesaurus transformation
Phrase transformation
0
N/2
N
Poor low-frequency terms
Good medium-frequency terms
Very poor high-frequency terms
Number of documents to which term is assigned
Zero discrimination value
Positive discrimination value
Negative discrimination value

References

[1] Gaston H. Gonnet, Ricardo A. Baeza-Yates, and Tim Snider. New indices for text: Pat trees and pat arrays. pages 66–82, 1992.

[2] Gerard Salton. Another look at automatic text-retrieval systems. *Commun. ACM*, 29(7):648–656, 1986.