Information Retrieval Systems Class Notes (May 7, 2014)

Prepared by: Hamed Rezanejad

Signature file:

- All the signatures that represent the documents in the collection are kept in a file called "signature file".
- Each document is divided into blocks containing an equal number of distinct words (possibly not the last one).
- If we consider each document like below (each block have same number of terms):

Block 1
Block 2
Block 3
…
Block n

✓ For last block we may have less number of terms.

Example:

Term	*Term signature*
Object	1000 1000
Signature	0010 0100
Generation	1000 0100
	1010 1100

✓ The last row is block signature which we consider the number of 1's in the block.

How we can make sure that 50%=1 and 50%=0?

$$Number\ of\ bits\ to\ be\ set\ in\ term\ signature = \frac{F\ Ln2}{\mathrm{D}}$$

- F: signature size in bits
- D: no. of unique terms in a block

Query	*Query signature*	*Result*
Database	1100 0000	No match
generation	1000 0100	True match
information	1010 0000	False match

If ([(Q_s) and (block signature)] = Q_s)

Then we consider the block as a possible match.

False drop resolution: After each match we have to eliminate possible false matches

Sequential signature file:

S1	0001 1110	Doc 1
S2	1101 0001	Doc 2
S3	0011 1100	Doc 3
S4	1100 0011	Doc 4
S5	0011 0110	Doc 5
S6	1100 1001	Doc 6

Bit-sliced signature:

C1	010 101
C2	010 101
C3	001 010
C4	111 010
C5	101 001
C6	101 010
C7	100 110
C8	010 101

Storage sample:

010 101	010 101	001 010	…

- ✓ If we have additions to signature we have to shift the bit arrays in storage which takes huge amount of process.

Assume we have a query like "generation" => Q_s: **1**000 0**1**00

We got bit-sliced signature for all 1's of query and AND them bit by bit:

C1	010 101
C6	101 010
AND's result	**000 000**

- ✓ After processing a number of bit slices it may be more efficient to do false drop resolution rather than accessing another bit slice.
- ✓ Remember that a bit slice may occupy several disk pages
- ✓ Perform partial evaluation (i.e. do not process all 1 bits of query signature)

Let: **op** = $\frac{Number\ of\ 1's\ in\ a\ signature}{signature\ length}$ it should be approximately 0.5

fd: False Drop probability = $\frac{number\ of\ matches}{number\ of\ signatures\ (number\ of\ blocks)}$

Note that we assume that all of matches are false drops, since number of true matches is very small.

Response time after processing i number of bits: RT(i)

$$\mathbf{RT(i)} = i.\ T_{slice} + N.\ op^{i}\ .\ T_{resolve}$$

T_{slice}: time required to access a bit slice

$T_{resolve}$: time required to resolve (eliminate) a false drop

To find i value for the minimum response time we take the derivate of RT(i) with respect to i.

$$\frac{d(RT(i))}{di} = T_{slice} + N.T_{resolve}.op^{i}.\ln(op)$$

To find the optimum value of i let the above equation equal to 0 and solve it for i:

$$Op^{i} = \frac{Tslice}{N.Tresolve.(\ln(op))}$$

$$\text{Ln } op^{i} = i\ \ln(op) = \ln(\frac{Tslice}{N.Tresolve(\ln(op))})$$

$$i = \ln\left(\frac{Tslice}{N.Tresolve\ (\ln op)}\right) / \ln(op)$$

if (i > weight of query(number of 1's in query))

then (use only the bits available in the query)

Reference: Seyit Kocberber, Fazli Can: Partial Evaluation of Queries for Bit-Sliced Signature Files. *Inf. Process. Lett.* 60(6): 305-311 (1996)

Signature file partitioning

S1	0111 1000
S2	1000 1011
S3	0011 1100
S4	1100 0011
S5	0110 1100
S6	1001 0011
S7	0000 1111

Fixed prefix partitioning:

- Use 1 bit: k=1 =>

0	1
S1	S2
S3	S4
S5	S6
S7	

- Use 2 bit: k=2 =>

00	01	10	11
S3	S1	S2	S4
S7	S5	S6	

- Use 3 bit: k=3 =>

000	001	010	011	100	101	110	111
S7	S3		S1	S2		S4	
			S5	S6			

Q1: 1110 0001

Q2: 0000 1111

Q3: 0110 0011

If (Q_s & B_s = Q_s)

Then consider the block and match Q_s with individual signature of the block for matching block signature perform false drop resolution.

$\{P_i \mid P_{key} \cap Q_{key} = Q\}$

PAR: Partition Activation Ratio = $\frac{Number\ of\ partition\ activated\ (selected)}{total\ number\ of\ partitions}$

SAR: Signature Activation Ratio = $\frac{Number\ of\ signature\ in\ the\ activation\ partition}{total\ number\ of\ signatures}$

	K=1	K=2	K=3	PAR (k=1)	SAR (K=1)	PAR (k=2)	SAR (K=2)	PAR (k=3)	SAR (K=3)
Q1: 100 …	1(1)*	2 (10, 11)	4(100, 101, 110, 111)	1/2	3/7	2/4	3/7	4/8	3/7
Q2: 000 …	2(0, 1)	4 (00, 01, 10, 11)	8(all)	2/2	7/7	4/4	7/7	8/8	7/7
Q3: 011 …	2(0, 1)	2 (01, 11)	2(011, 111)	2/2	7/7	2/4	3/7	2/8	2/7
* No of partitions (matching partition query)									

- ✓ Signature partition with a key that contains all 1's matching all queries.
  Number of partitions matching the query signature = 2.

Reference: Dik Lun Lee, Chun-Wu Roger Leng: Partitioned Signature Files: Design Issues and Performance Evaluation. *ACM Trans. Inf. Syst.* 7(2): 158-180 (1989).

Questions:

1. If we have 3 unique terms in our block and we have 12 bits for each term's signature, what would be the number of 1s in each term signature?

- ✓ We know that:

$$Number\ of\ bits\ to\ be\ set\ in\ term\ signature = \frac{F\ Ln2}{\mathrm{D}}$$

So we can say that: 12*ln2/3 = 2.77 => we have 3

2. Consider below signature:

Term	*Term signature*
Free	0001 0100 0010
Text	0100 0010 0001
Data	1000 0010 0010
Block signature	**1101 0110 0011**

False drop happens generally in which situations? What are the possible reasons of false drop?

- ✓ False drop occurs when a document's signature matches a query's signature but the query's word does not match any word in the document.
- ✓ It is possible because 2 distinct blocks may have the same signatures due to:
    - the hashing algorithm
    - superimposed coding

3. Is it possible to have different PAR and SAR values? Explain your answer.

- ✓ Yes: Different blocks may have different number of signatures.