

CMU SCS

## 15-826: Multimedia Databases and Data Mining

*Text - part II*  
C. Faloutsos

CMU SCS

## Outline

Goal: ‘Find similar / interesting things’

- Intro to DB
- Indexing - similarity search
- Data Mining

15-826 Copyright: C. Faloutsos (2006) 2

CMU SCS

### Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- text
- multimedia
- ...

15-826 Copyright: C. Faloutsos (2006) 3

CMU SCS

### Text - Detailed outline

- text
  - problem
  - full text scanning
  - inversion
  - signature files
  - clustering
  - information filtering and LSI

15-826 Copyright: C. Faloutsos (2006) 4

CMU SCS

### Text - Inversion

15-826 Copyright: C. Faloutsos (2006) 5

CMU SCS

### Text - Inversion

Q: space overhead?

15-826 Copyright: C. Faloutsos (2006) 6

**Text - Inversion**

A: mainly, the postings lists

15-826      Copyright: C. Faloutsos (2006)      7

**Text - Inversion**

- how to organize dictionary?
- stemming – Y/N?
- insertions?

15-826      Copyright: C. Faloutsos (2006)      8

**Text - Inversion**

- how to organize dictionary?
  - B-tree, hashing, TRIEs, PATRICIA trees, ...
- stemming – Y/N?
- insertions?

15-826      Copyright: C. Faloutsos (2006)      9

**Text – Inversion**

- variations:
- Parallelism [Tomasic+93]
- Insertions [Tomasic+94], [Brown+]
  - ‘zipf’ distributions
- Approximate searching ('glimpse' [Wu+])

15-826      Copyright: C. Faloutsos (2006)      10

**Text - Inversion**

- postings list – more Zipf distr.: eg., rank-frequency plot of ‘Bible’

15-826      Copyright: C. Faloutsos (2006)      11

**Text - Inversion**

- postings lists
  - Cutting+Pedersen
    - (keep first 4 in B-tree leaves)
  - how to allocate space: [Faloutsos+92]
    - geometric progression
  - compression (Elias codes) [Zobel+] – down to 2% overhead!

15-826      Copyright: C. Faloutsos (2006)      12



## Conclusions

- Conclusions: needs space overhead (2%-300%), but it is the fastest

15-826

Copyright: C. Faloutsos (2006)

13



## Text - Detailed outline

- text
  - problem
  - full text scanning
  - inversion
  - signature files
  - clustering
  - information filtering and LSI

15-826

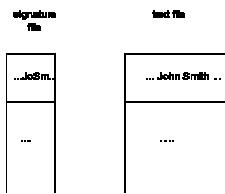
Copyright: C. Faloutsos (2006)

14



## Signature files

- idea: ‘quick & dirty’ filter



15-826

Copyright: C. Faloutsos (2006)

15



## Signature files

- idea: ‘quick & dirty’ filter
- then, do seq. scan on sign. file and discard ‘false alarms’
- Adv.: easy insertions; faster than seq. scan
- Disadv.: O(N) search (with small constant)
- Q: how to extract signatures?

15-826

Copyright: C. Faloutsos (2006)

16



## Signature files

- A: superimposed coding!! [Mooers49], ...

Word	Signature
data	001 000 110 010
base	000 010 101 001
doc. signature	001 010 111 011

m (=4 bits/word)  
F (=12 bits sign. size)

15-826

Copyright: C. Faloutsos (2006)

17



## Signature files

- A: superimposed coding!! [Mooers49], ...

Word	Signature
data	001 000 110 010
base	000 010 101 001
doc. signature	001 010 111 011

↑      ↑↑      ↑

actual match

15-826

Copyright: C. Faloutsos (2006)

18





## Signature files

- Q1: How to choose  $F$  and  $m$  ?
- • Q2: Why is it called ‘false drop’?
- Q3: other apps of signature files?

15-826

Copyright: C. Faloutsos (2006)

25



## Signature files

- Q2: Why is it called ‘false drop’?
- Old, but fascinating story [1949]
  - how to find qualifying books (by title word, and/or author, and/or keyword)
  - in  $O(1)$  time?
  - *without computers*

15-826

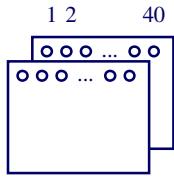
Copyright: C. Faloutsos (2006)

26



## Signature files

- Solution: edge-notched cards



- each title word is mapped to  $m$  numbers (how?)
- and the corresponding holes are cut out:

15-826

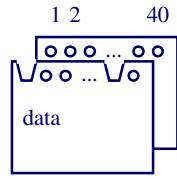
Copyright: C. Faloutsos (2006)

27



## Signature files

- Solution: edge-notched cards



'data' -&gt; #1, #39

15-826

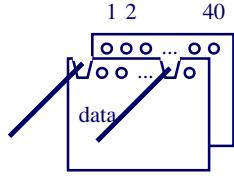
Copyright: C. Faloutsos (2006)

28



## Signature files

- Search, e.g., for ‘data’: activate needle #1, #39, and shake the stack of cards!



'data' -&gt; #1, #39

15-826

Copyright: C. Faloutsos (2006)

29



## Signature files

- Also known as ‘zatocoding’, from ‘Zator’ company.

15-826

Copyright: C. Faloutsos (2006)

30



## Signature files

- Q1: How to choose  $F$  and  $m$  ?
- Q2: Why is it called ‘false drop’?
- • Q3: other apps of signature files?

15-826

Copyright: C. Faloutsos (2006)

31



## Signature files

- Q3: other apps of signature files?
- A: anything that has to do with ‘membership testing’: does ‘data’ belong to the set of words of the document?

Word	Signature
data	001 000 110 010
base	000 010 101 001
doc. signature	001 010 111 011

15-826

Copyright: C. Faloutsos (2006)

32



## Signature files

- UNIX’s early ‘spell’ system [McIlroy]
- Bloom-joins in System R\* [Mackert+] and ‘active disks’ [Riedel99]
- differential files [Severance+Lohman]

15-826

Copyright: C. Faloutsos (2006)

33



## Signature files - conclusions

- easy insertions; slower than inversion
- brilliant idea of ‘quick and dirty’ filter: quickly discard the vast majority of non-qualifying elements, and focus on the rest.

15-826

Copyright: C. Faloutsos (2006)

34



## References

- Aho, A. V. and M. J. Corasick (June 1975). "Fast Pattern Matching: An Aid to Bibliographic Search." CACM 18(6): 333-340.
- Boyer, R. S. and J. S. Moore (Oct. 1977). "A Fast String Searching Algorithm." CACM 20(10): 762-772.
- Brown, E. W., J. P. Callan, et al. (March 1994). Supporting Full-Text Information Retrieval with a Persistent Object Store. Proc. of EDBT conference, Cambridge, U.K., Springer Verlag.

15-826

Copyright: C. Faloutsos (2006)

35



## References - cont’d

- Faloutsos, C. and H. V. Jagadish (Aug. 23-27, 1992). On B-tree Indices for Skewed Distributions. 18th VLDB Conference, Vancouver, British Columbia.
- Karp, R. M. and M. O. Rabin (March 1987). "Efficient Randomized Pattern-Matching Algorithms." IBM Journal of Research and Development 31(2): 249-260.
- Knuth, D. E., J. H. Morris, et al. (June 1977). "Fast Pattern Matching in Strings." SIAM J. Comput 6(2): 323-350.

15-826

Copyright: C. Faloutsos (2006)

36



## References - cont'd

- Mackert, L. M. and G. M. Lohman (August 1986). R\* Optimizer Validation and Performance Evaluation for Distributed Queries. Proc. of 12th Int. Conf. on Very Large Data Bases (VLDB), Kyoto, Japan.
- Manber, U. and S. Wu (1994). GLIMPSE: A Tool to Search Through Entire File Systems. Proc. of USENIX Techn. Conf.
- McIlroy, M. D. (Jan. 1982). "Development of a Spelling List." IEEE Trans. on Communications COM-30(1): 91-99.

15-826

Copyright: C. Faloutsos (2006)

37



## References - cont'd

- Mooers, C. (1949). Application of Random Codes to the Gathering of Statistical Information
- Bulletin 31. Cambridge, Mass, Zator Co.
- Pedersen, D. C. a. J. (1990). Optimizations for dynamic inverted index maintenance. ACM SIGIR.
- Riedel, E. (1999). Active Disks: Remote Execution for Network Attached Storage. ECE, CMU. Pittsburgh, PA.

15-826

Copyright: C. Faloutsos (2006)

38



## References - cont'd

- Severance, D. G. and G. M. Lohman (Sept. 1976). "Differential Files: Their Application to the Maintenance of Large Databases." ACM TODS 1(3): 256-267.
- Tomasic, A. and H. Garcia-Molina (1993). Performance of Inverted Indices in Distributed Text Document Retrieval Systems. PDIS.
- Tomasic, A., H. Garcia-Molina, et al. (May 24-27, 1994). Incremental Updates of Inverted Lists for Text Document Retrieval. ACM SIGMOD, Minneapolis, MN.

15-826

Copyright: C. Faloutsos (2006)

39



## References - cont'd

- Wu, S. and U. Manber (1992). "AGREP- A Fast Approximate Pattern-Matching Tool." .
- Zobel, J., A. Moffat, et al. (Aug. 23-27, 1992). An Efficient Indexing Technique for Full-Text Database Systems. VLDB, Vancouver, B.C., Canada.

15-826

Copyright: C. Faloutsos (2006)

40