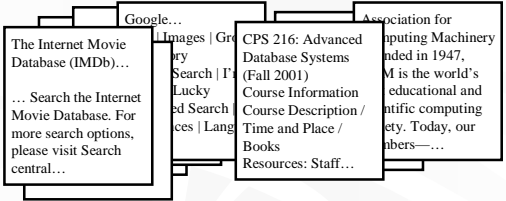


Even More Indexing!

CPS 216
Advanced Database Systems

Keyword search



The screenshot shows a search engine interface with a search bar containing the text "database AND search" and a "Search" button. Below the search bar, there are several search results displayed as overlapping boxes. The results include:

- Google...
- Association for...
- The Internet Movie Database (IMDb)...
- Images | Gr... | Lucky
- CPS 216: Advanced Database Systems (Fall 2001) Course Information / Time and Place / Books Resources: Staff...
- Computing Machinery... ended in 1947, M is the world's educational and scientific computing society. Today, our members—...

database AND search Search

What are the documents containing both "database" and "search"?

2

Search features

- Boolean searches
 - (database OR Web) AND search
- Phrase searches
 - "database search"
- Result ranking

Keywords × documents

All documents

Document 1 Document 2 Document 3 Document n

All keywords	1	1	1	...	1
"a"	1	1	0	...	0
"cat"	0	0	1	...	0
"database"	0	1	0	...	1
"dog"	0	0	1	...	0
"search"

1 means keyword appears in the document
0 means otherwise

4

Inverted lists

- Store the matrix by rows
- For each keyword, store an inverted list
 - $\langle \text{keyword}, \text{document-id-list} \rangle$
 - $\langle \text{"database"}, \{3, 7, 142, 857, \dots\} \rangle$
 - $\langle \text{"search"}, \{3, 9, 192, 512, \dots\} \rangle$
 - It helps to sort *document-id-list* (why?)
- Vocabulary index on keywords
 - B+-tree or hash-based

5

Using inverted lists

- Documents containing "database"
 - Use the vocabulary index to find the inverted list for "database"
 - Return documents in the inverted list
- Documents containing "database" AND "search"
 - Return documents in the intersection of the two inverted lists
- OR? NOT?

6

What are “all” the keywords?

- All sequences of letters?
 - ... that actually appear in documents!
- All words in English?
- Plus all phrases?
 - Alternative: approximate phrase search by proximity
- Minus all stop words
 - They appear in nearly every document; not useful in search
 - Example: a, of, the, it
- Combine words with common stems
 - They can be treated as the same for the purpose of search
 - Example: database, databases

7

Frequency and proximity

- Frequency
 - $\langle \text{keyword}, \{ \langle \text{doc-id}, \text{number-of-occurrences} \rangle, \langle \text{doc-id}, \text{number-of-occurrences} \rangle, \dots \} \rangle$
- Proximity (and frequency)
 - $\langle \text{keyword}, \{ \langle \text{doc-id}, \langle \text{position-of-occurrence}_1, \text{position-of-occurrence}_2, \dots \rangle, \langle \text{doc-id}, \langle \text{position-of-occurrence}_1, \dots \rangle, \dots \} \rangle$

8

Ranking Web pages using links

- Basic idea: A page is relevant if a lot of relevant pages have links pointing to it
 - Recursive definition?
 - No problem—fixed-point iteration!
- Google
 - Pre-compute the “general” ranking of all pages
 - This ranking can be used in the inverted lists
- HITS, Teoma
 - Compute the “topic-specific” ranking dynamically for pages that satisfy the search criteria

9

Keywords × documents

All documents

All keywords	Document 1	Document 2	Document 3	Document n	
"a"	1	1	1	...	1
"cat"	1	1	0	...	0
"database"	0	0	1	...	0
"dog"	0	1	0	...	1
"search"	0	0	1	...	0
...

10

Signatures

- Store the matrix by columns
- For each document, store a signature
 - If the document satisfies a search condition (e.g., contains "database"), set the corresponding bit in the signature
 - Signature too big? Compress!
 - Example: hash keywords and then set corresponding bits
 - Lossy compression can generate false positives

hash("database") = 0110 *doc*₁ contains "database": 0110
hash("dog") = 1100 *doc*₂ contains "dog": 1100
hash("cat") = 0010 *doc*₃ contains "cat" and "dog": 1110

11

Inverted lists versus signatures

- Inverted lists
 - High space overhead: could be bigger than the original documents!
- Signatures
 - Sequential scan through the signatures required

12

What's next

GiST

Presenters:
Andrew Danner & Sara Sprenkle

13
