

Outline

- Finish Indexing Methods
- Discuss Project #1
- Vector Space Models

Inverted Files

- Popular, fast, probably best choice for medium to large datasets.
- Assumes we want to treat text as set of terms.
- Recall the relations:

Index : $doc_i \xrightarrow{\text{about}} \{kw_j\}$

Index⁻¹ : $\{kw_j\} \xrightarrow{\text{describes}} \{doc_i\}$

Inverted File Structures

- Document Files
 - Each doc given identifier
 - All terms identified
- Dictionary
 - Sorted list of terms in collection
 - or hash table or prefix trie
 - might track total count in collection
- Inversion list
 - Pointers from term to document containing term
 - Might also specify where in doc

Basic Algorithm

- For every document in corpus
 - while there are more terms
 - get next term
 - stem or normalize if needed
 - record posting(term, doc)
- For every term in posting
 - record the number of documents
 - record the total frequency in all docs
 - sort the set of postings for a term in descending order by frequency in document
 - write term, numdocs, total freq., and postings to file(s)

Example: Bread Search

There once was a searcher named Hanna,
Who needed some info on manna.
She put 'rye' and 'wheat' in her query
Along with 'potato' or 'cranbeery,'
But no mention of 'sourdough' or 'banana.'

Instead of rye, cranberry, or wheat,
The results had more spiritual meat.
So Hanna was not pleased,
Nor was her hunger eased,
'Cause she was looking for something to eat.

We're going to index each line.

Example: Bread Search docs

- 1) there once was a searcher named hanna
- 2) who needed some info on manna
- 3) she put rye and wheat in her query
- 4) along with potato or cranbeery
- 5) but no mention of sourdough or banana
- 6) instead of rye cranberry or wheat
- 7) the results had more spiritual meat
- 8) so hanna was not pleased
- 9) nor was her hunger eased
- 10) cause she was looking for something to eat

Which terms are useful to index?

Example: Bread Search Terms

- 1) searcher hanna
- 2) manna
- 3) rye wheat query
- 4) potato cranbeery
- 5) sourdough banana
- 6) rye cranberry wheat
- 7) spiritual meat
- 8) hanna
- 9) hunger
- 10) eat

Now work out the inversion file structures for each line.

Bread Search Dictionary List

<u>Term</u>	<u>Global frequency</u>
banana	1
cranbeery	1
cranberry	1
eat	1
hanna	2
hunger	2
manna	1
meat	1
potato	1
query	1
rye	2
sourdough	1
spiritual	1
wheat	2

Bread Search Inversion List

<u>Term</u>	<u>(Document Num, Pos)</u>
banana	(5,7)
cranbeery	(4,5)
cranberry	(6,4)
eat	(10,8)
hanna	(1,7); (8,2)
hunger	(9,4)
manna	(2,6)
meat	(7,6)
potato	(4,3)
query	(3,8)
rye	(3,3); (6,3)
sourdough	(5,5)
spiritual	(7,5)
wheat	(3,5); (6,6)

Index Size Exercise

- According to MIR, in well-written code:
 - Inverted files can use as little as 5-40% of original collection space.
 - Suffix trees use 120-240% of original text.
 - Suffix arrays use approx. 40% of original text.
 - Signature files use 10-20% of original text.
- Given a WWW with 4B docs, each with at least 10KB, what is expected minimum index size?