PURDUE UNIVERSITY

CS54701: Information Retrieval

Basic Concepts

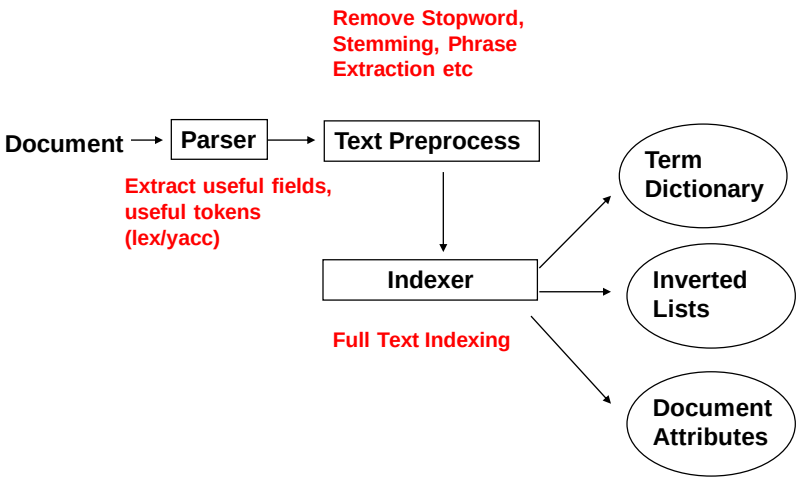
19 January 2016

Prof. Chris Clifton

Indiana Center for Database Systems

1

Text Representation: Process of Indexing



Text Representation: Inverted Lists

Inverted lists are one of the most common indexing techniques

- Source file: collection organized by documents
- Inverted list file: collection organized by term

  one record per term, the lists of documents that contain the specific term
- Possible actions with inverted lists
  - OR: the union of lists
  - And: the intersection of lists

Text Representation: Inverted Lists

Doc ID	Text
1	kids question noting in 1960s
2	young man question everything in 1970s
3	kids question questions in 1980s
4	young man question nothing in 2000s

Documents

Term ID	Term	Documents
1	kids	1,3
2	question	1,2,3,4
3	nothing	1,4
4	in	1,2,3,4
5	19060s	1
6	young	2,4
7	man	2,4
8	everything	2
9	1970s	2
10	questions	3
11	1980s	3
11	2000s	4

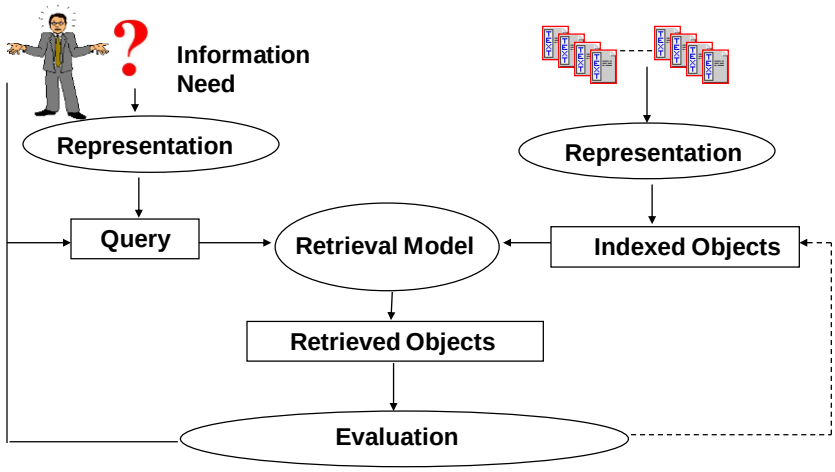
Inverted Lists

Text Representation: Inverted Lists

Many engineering details

- Update inverted lists: delete/insert a term or document
- Compression: trade off between I/O time and CPU time
- Add more information such as position information
- ……..

AD-hoc IR: Basic Process



Evaluation

Evaluation criteria

- Effectiveness
  - How to define effectiveness? Where can we find the correct answers?
- Efficiency
  - What about retrieval speed? What about the storage space? Particularly important for large-scale real-world system
- Usability
  - What is the most important factor for real user? Is user interface important?

Evaluation

Evaluation criteria

- Effectiveness
  - Favor returned document ranked lists with more relevant documents at the top
  - Objective measures

Recall and Precision

Mean-average precision

Rank based precision

For documents in a subset of a ranked lists, if we know the truth

	Retrieved	Not retrieved
Relevant	Relevant docs retrieved	Relevant docs not retrieved
Irrelevant	Irrelevant docs retrieved	Irrelevant docs not retrieved

$$\text{Precision} = \frac{\text{Relevant docs retrieved}}{\text{Retrieved docs}}$$

$$\text{Recall} = \frac{\text{Relevant docs retrieved}}{\text{Relevant docs}}$$

Evaluation

	Retrieved	Not retrieved
Relevant	Relevant docs retrieved	Relevant docs not retrieved
Irrelevant	Irrelevant docs retrieved	Irrelevant docs not retrieved

Question: How to find all relevant documents?

Difficult for Web, but possible on controllable corpus

- How to find all relevant documents? (difficult to check one by one)
- Judgers may have inconsistent decisions (subjective judgment)

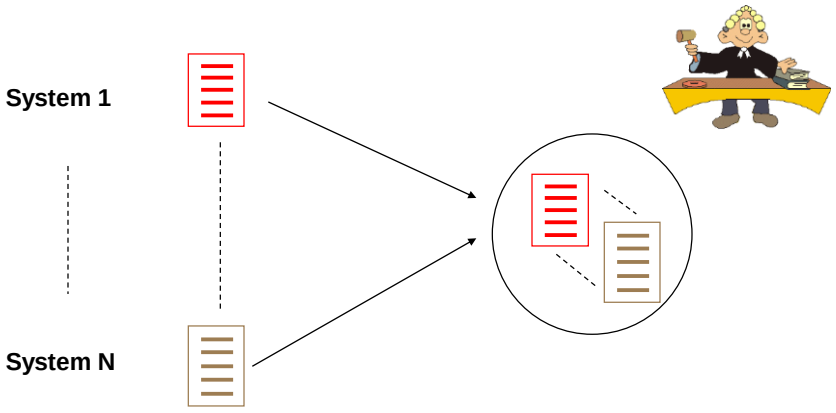
The Pooling process

Evaluation

Pooling Strategy

- Retrieve documents using multiple methods
- Judge top n documents from each method
- Whole retrieved set is the union of top retrieved documents from all methods
- Problems: the judged relevant documents may not be complete
- It is possible to estimate size of true relevant documents by randomly sampling

Evaluation



Evaluation

Inconsistent Judgment

- Discussion among multiple judgers to reduce bias
- Combine judgments from multiple judgers
  - Majority vote
- If it is hard to decide for human judgers, it is also hard for automatic system

Evaluation

Evaluate a ranked list

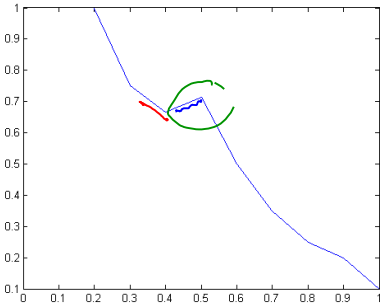
Precision at Recall

- Evaluate at every relevant document

+
+
-
+
+
-
+

Not Retrieved: +++++

Precision	Recall
1	0.1
1	0.2
0.667	0.2
0.75	0.3
0.8	0.4
0.667	0.4
0.714	0.5



Evaluation

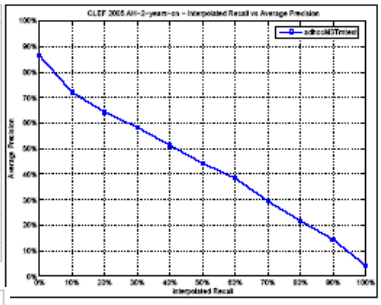
Single value metrics

- Mean average precision
  - Calculate precision at each relevant document; average over all precision values
- 11-point interpolated average precision
  - Calculate precision at standard recall points (e.g., 10%, 20%...); smooth the values; estimate 0 % by interpolation
  - Average the results
- Rank based precision
  - Calculate precision at top ranked documents (e.g., 5, 10, 15…)
  - Desirable when users care more for top ranked documents

Evaluation

Sample Results

Interpolated Recall (%)	Precision Averages (%)
0	86.49
10	72.16
20	64.25
30	58.40
40	51.33
50	44.30
60	38.43
70	29.43
80	21.68
90	14.40
100	4.15
Average precision (non-interpolated) for all relevant documents (averaged over queries)	
	43.06



Docs Cutoff Levels	Precision at DCL (%)
5 docs	72.50
10 docs	67.00
15 docs	61.83
20 docs	59.25
30 docs	55.42
100 docs	39.75
200 docs	30.92
500 docs	19.54
1000 docs	12.02
R-Precision (precision after R document retrieved, where R = Relevant retrieved)	
	44.99

Evaluation

TREC collections with queries and relevance judgment

- **TREC CDs 1-5**: 1.5 millions docs, 5GB, news and government reports (e.g., AP, WSJ, Dept of Energy abstracts)
- **TREC WT10g**: crawled from Web (open domain), 1.7 million docs, 10GB
- **TREC Terabyte**: crawled from U.S. government Web pages, 25 million docs, 426 GB
- All have more than 100 queries with relevance judgment

Evaluation

TREC query example

<title> airport security

<desc> Description:
What security measures are in effect or are proposed to go into effect in airports?

<narr> Narrative:
A relevant document could identify a specific airport and describe the security measures already in effect or proposed for use at that airport. Relevant items could also describe a failure of security that was cited as a contributing cause of a tragedy which came to pass or which was later averted. Comparisons between and among airports based on the effectiveness of the security of each are also relevant.

Evaluation

TREC relevance judgment example

451 WTX058-B50-85 0
451 WTX059-B06-411 0
451 WTX059-B07-154 0
451 WTX059-B09-203 0
451 WTX059-B11-245 0
451 WTX059-B30-262 1
451 WTX059-B37-11 0
451 WTX059-B37-149 1
451 WTX059-B37-217 0
451 WTX059-B37-268 0
451 WTX059-B37-27 0

Lecture(s) review:

- Basic Concepts of Information Retrieval:
- Task Definition of Ad-hoc IR
  - Terminologies and Concepts
  - Overview of Retrieval Models

- Text representation
  - Indexing
  - Text preprocessing
- Evaluation
  - Evaluation methodology
  - Evaluation metrics

PURDUE UNIVERSITY

CS54701: Information Retrieval

Retrieval Models

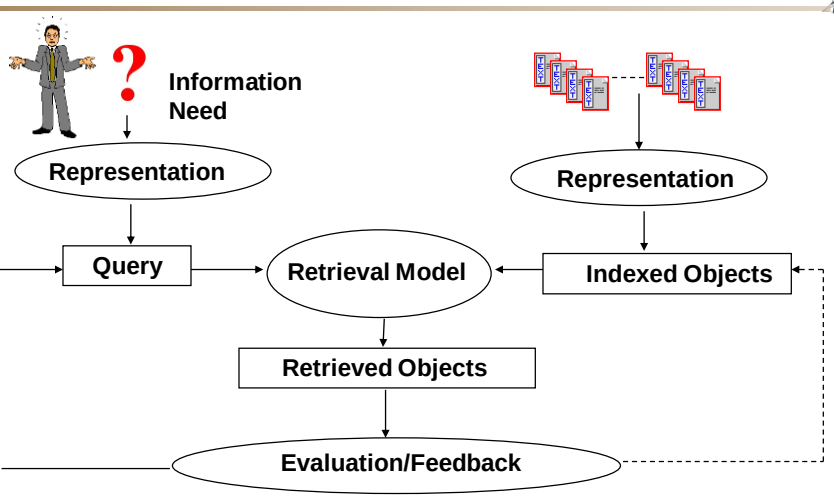
19 January 2016

Prof. Chris Clifton

Indiana Center for Database Systems

20

Retrieval Models



Overview of Retrieval Models

Retrieval Models

- Boolean
- Vector space
  - Basic vector space — SMART, Lucene
  - Extended Boolean
- Probabilistic models
  - Statistical language models — Lemur
  - Information Theoretic — Okapi
  - Bayesian inference networks — Inquery
- Citation/Link analysis models
  - Page rank — Google
  - Hub & authorities — Clever

Retrieval Models: Outline

Retrieval Models

- Exact-match retrieval method
  - Unranked Boolean retrieval method
  - Ranked Boolean retrieval method
- Best-match retrieval method
  - Vector space retrieval method
  - Latent semantic indexing

Retrieval Models: Unranked Boolean

- Unranked Boolean: Exact match method
- Selection Model
  - Retrieve a document iff it matches the precise query
  - Often return unranked documents (or with chronological order)
- Operators
  - Logical Operators: AND OR, NOT
  - Proximity operators: #1(white house) (i.e., within one word distance, phrase) #sen(Iraq weapon) (i.e., within a sentence)
  - String matching operators: Wildcard (e.g., ind* for india and indonesia)
  - Field operators: title(information and retrieval)…

Retrieval Models: Unranked Boolean

Unranked Boolean: Exact match method

- A query example
  (#2(distributed information retrieval) OR
  (#1 (federated search)) AND
  author(#1(Jamie Callan) AND NOT (Steve))

Retrieval Models: Unranked Boolean

WestLaw system: Commercial Legal/Health/Finance Information Retrieval System

- Logical operators
- Proximity operators: Phrase, word proximity, same sentence/paragraph
- String matching operator: wildcard (e.g., ind*)
- Field operator: title(#1(“legal retrieval”)) date(2000)
- Citations: Cite (Salton)

Retrieval Models: Unranked Boolean

Advantages:

- Work well if user knows exactly what to retrieve
- Predicable; easy to explain
- Very efficient

Disadvantages:

- It is difficult to design the query; high recall and low precision for loose query; low recall and high precision for strict query
- Results are unordered; hard to find useful ones
- Users may be too optimistic for strict queries. A few very relevant but a lot more are missing

Retrieval Models: Ranked Boolean

Ranked Boolean: Exact match

- Similar as unranked Boolean but documents are ordered by some criterion

Retrieve docs from Wall Street Journal Collection

Query: (Thailand AND stock AND market)

Which word is more important?

Many "stock" and "market", but fewer "Thailand". Fewer may be more indicative

Reflect importance of document by its words

Term Frequency (TF): Number of occurrence in query/doc; larger number means more important

Inversed Document Frequency (IDF): Larger means more important

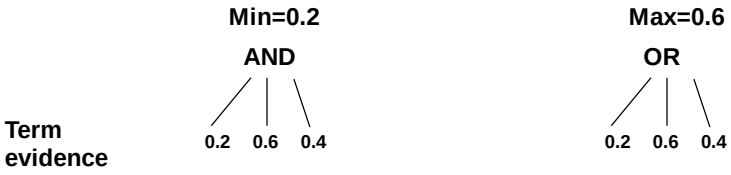
$$\frac{\text{Total number of docs}}{\text{Number of docs contain a term}}$$

There are many variants of TF, IDF: e.g., consider document length

Retrieval Models: Ranked Boolean

Ranked Boolean: Calculate doc score

- Term evidence: Evidence from term i occurred in doc j: (tfij) and (tfij*idfi)
- AND weight: minimum of argument weights
- OR weight: maximum of argument weights



Query: (Thailand AND stock AND market)

Retrieval Models: Ranked Boolean

Advantages:

- All advantages from unranked Boolean algorithm
  - Works well when query is precise; predictive; efficient
- Results in a ranked list (not a full list); easier to browse and find the most relevant ones than Boolean
- Rank criterion is flexible: e.g., different variants of term evidence

Disadvantages:

- Still an exact match (document selection) model: inverse correlation for recall and precision of strict and loose queries
- Predictability makes user overestimate retrieval quality

Retrieval Models: Vector Space Model

- Vector space model
- Any text object can be represented by a term vector
  - Documents, queries, passages, sentences
  - A query can be seen as a short document
- Similarity is determined by distance in the vector space
  - Example: cosine of the angle between two vectors
- The SMART system
  - Developed at Cornell University: 1960-1999
  - Still quite popular
- The Lucene system
  - Open source information retrieval library; (Based on Java)
  - Work with Hadoop (Map/Reduce) in large scale app (e.g., Amazon Book)

Retrieval Models: Vector Space Model

Vector space model vs. Boolean model

- Boolean models
  - Query: a Boolean expression that a document must satisfy
  - Retrieval: Deductive inference
- Vector space model
  - Query: viewed as a short document in a vector space
  - Retrieval: Find similar vectors/objects

Retrieval Models: Vector Space Model

- Vector representation

	Java	Sun	Starbucks
D1	1	1	0
D2	1	0	1
D3	1	0	0
Query	1	0.2	1

Retrieval Models: Vector Space Model

Vector representation