Intro to DB

CHAPTER 19 INFORMATION RETRIEVAL

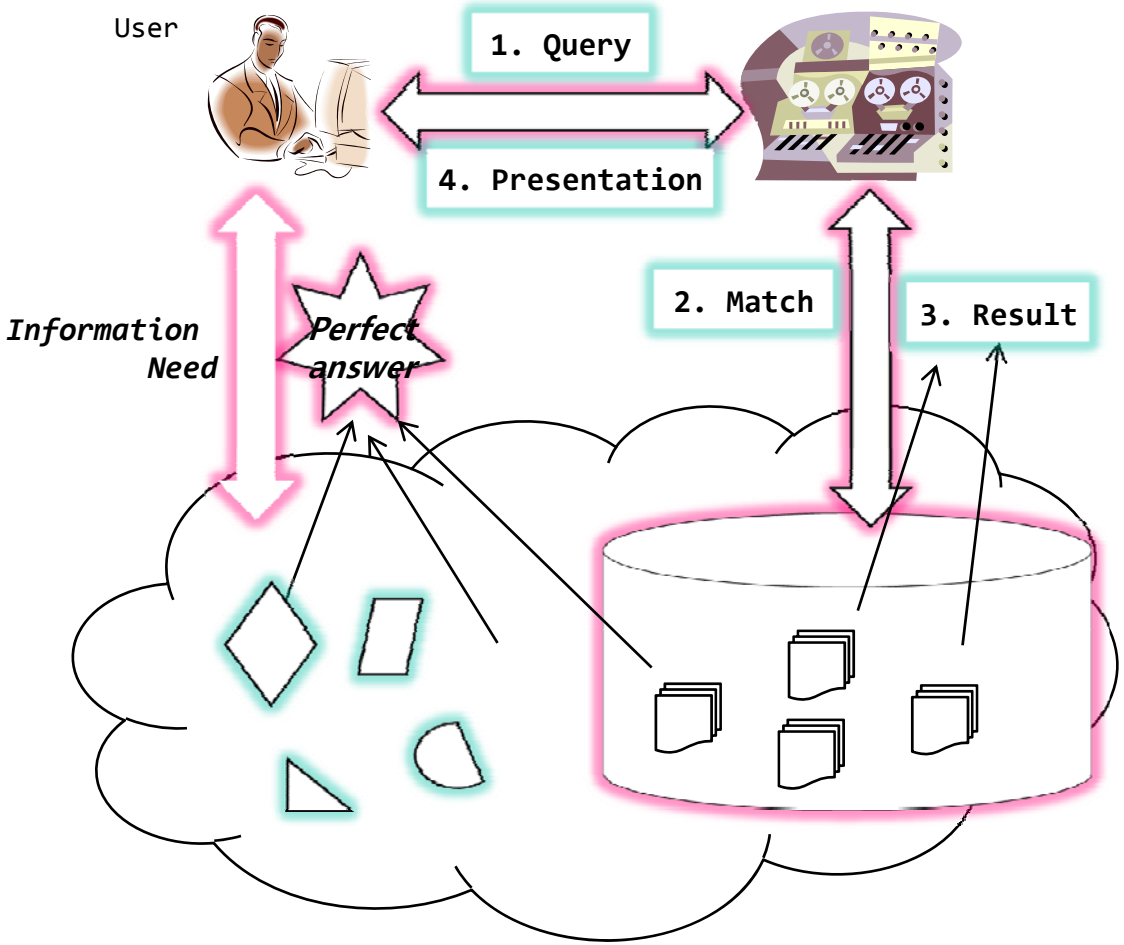
Chapter 19: Information Retrieval

- Relevance Ranking Using Terms
- Relevance Using Hyperlinks
- Synonyms., Homonyms, and Ontologies
- Indexing of Documents
- Measuring Retrieval Effectiveness
- Web Search Engines
- Information Retrieval and Structured Data
- Directories

Information Retrieval Systems

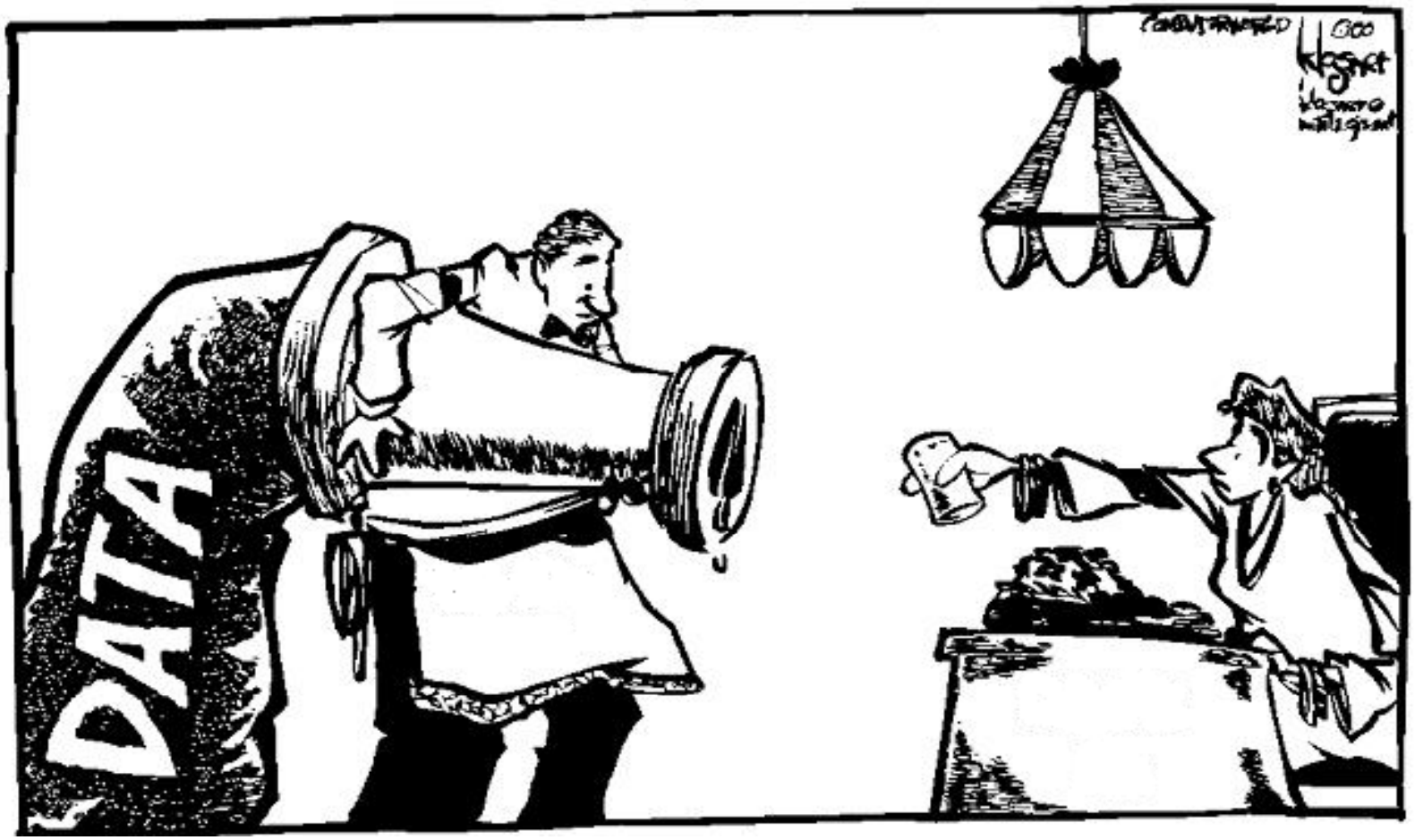
- **Information retrieval** (**IR**) systems
  - Use a simpler data model than database systems
  - Information organized as a collection of documents
  - Documents are unstructured, no schema
  - Can be used even on textual descriptions provided with non-textual data such as images
  - Web search engines are the most familiar example of IR systems
- Differences from database systems
  - IR systems don't deal with transactional updates (including concurrency control and recovery)
  - Database systems deal with structured data, with schemas that define the data organization
  - IR systems deal with some querying issues not generally addressed by DBMSs
    - Approximate searching by keywords
    - Ranking of retrieved answers by estimated degree of relevance

The Search Process



1. *Query*:
   - Can I represent my info. need accurately?
2. *Match*:
   - Is the matching algorithm adequate?
3. *Result*:
   - Do I get the type of information I was looking for?
4. *Presentation*:
   - Is the information presented in a comprehensible form?

The Search Problem



Matching Criteria

- Exact match
  - Relational DB (Business data)
  - numeric or alphabet (simple, atomic, well defined)
- Approximate match
  - IRS
  - docs are not well organized and queries are not precise
  - matching semantics is ambiguous
- Queries
  - natural language
  - list of terms

Information Retrieval (IR) Model

- How do we decide which documents are relevant to a given query?
  - What's our view (and representation) of a document?
  - What's our view (and representation) of a query?
  - What's our view of "relevance"? How do we compute it?

- An IR model is a conceptual structure in which these issues are defined.

Boolean Model

- Query
  - list of terms that need to be present in a doc
  - AND, OR, NOT
- e.g.
  - database AND medical
  - protocol AND NOT computer
  - (A AND B) OR (B AND C) OR (A AND C)
- Matching
  - A query is a characteristic function: value 1 for relevant documents and 0 for others
- Most popular in Web
  - Over 95% of queries are simple term queries
  - Simple (too simple!!!)
    - Unable to represent significance (weights)
    - Document ranking is difficult

Vector Model

- A document is represented as a vector (ordered list) of terms
- A query is also a vector of terms

(comp, database, OS, HW, ...) ← predetermined keyword list
d_1 = (0, 0, 1, 1, 0, 0, 1, 0, 1)
q = (0, 1, 0, 1, 1, 0, 0, 0, 1)
or
d_1 = (0, 0, 0.5, 0.8, 0, 0, 0.3, 0, 0.9) ← weights
q = (0, 0.6, 0, 0.7, 0.8, 0, 0, 0, 0.9)

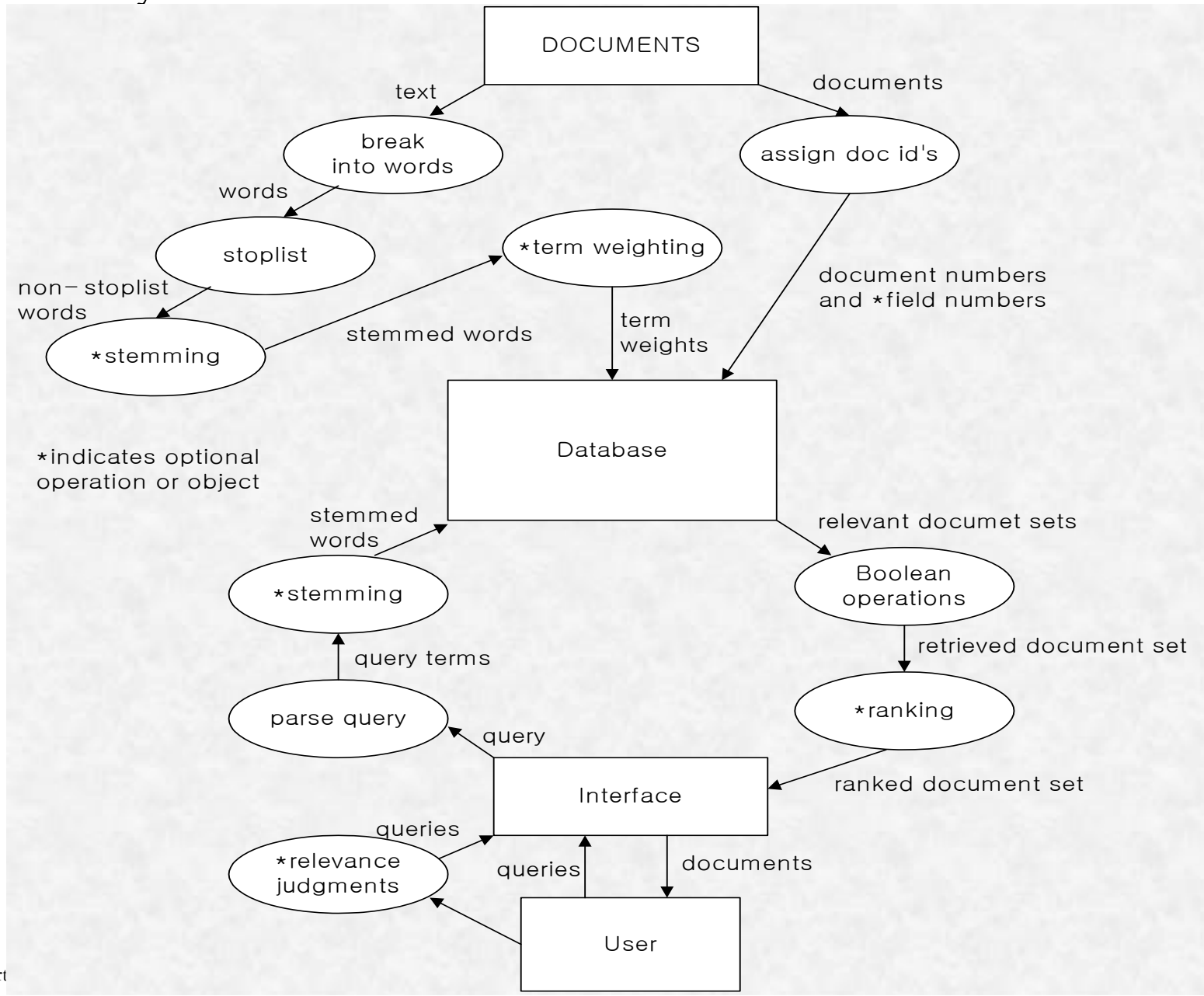
- Documents and query are points in a vector space.
- Similarity measures : more flexible than Boolean
  - Distance measure
  - Angular measure: cosine measure
- Not able to represent logical connectives

$$s(d,q) = \frac{d \bullet q}{|d| \times |q|} = \frac{\sum_k (t_k \times q_k)}{\sqrt{\sum_k (t_k)^2} \times \sqrt{\sum_k (q_k)^2}}$$

Indexing

- Indexing: assigning index terms to a document
  - to permit easy location of documents by topic
  - to define topic areas (relate one document to another)
  - to predict relevance of documents to a specified information need
- Indexing Language
  - set of index terms; also called the *vocabulary*
  - single words VS phrases
  - controlled VS uncontrolled
- Important words occur frequently but not too frequently
  - most frequently occurring words are *is*, *and*, *the*, … (stop words)
  - terms that occur only once or twice do not make good search terms

Text Analysis Process



Term Frequency

- **Term matching**
  - fundamental underlying process in IR
  - existence of a (query) term in a document doesn't always mean relevance
    - "This document is not about …", "The effect of … will be explained in subsequent documents", …
    - but almost always the best heuristic to relevance
- Absolute term frequency can be very misleading
  - term *t1* occurred 100 times in each of documents *d1* and *d2*
  - *d1* is 5000 words while *d2* is 50000 words
  - average doc in *d1*'s database has >100 occurrence of *t1* while average doc in *d2*'s database has < 10 occurrence
- Normalize term frequency counts to take into account document size
  - *atf* / *doc_size*
  - *atf* / *max*(*atf* in *doc*)

Inverse Document Frequency Weight

- Relative term frequency: take into account
  - document and collection size
  - document and collection characteristics
- Parameters

  N: # of docs in collection

  d_k: # of docs that contain term k (assume > 0)

  tf_{ik}: term frequency of term k in doc i (normalized by doc size)
- Candidates for normalized weight (TF.IDF)
  - tf_{ik} / d_k
  - $tf_{ik} / [(d_k/N)+1]$
  - $tf_{ik} / [\log_2(d_k/N) + 1]$
  - $tf_{ik} [\log_2 N - \log_2 d_k + 1]$
    - increases with tf_{ik}
    - decreases with d_k

IDF Example

- d_k
  - "oil" is found in 128 items, "Mexico" is found in 16 items, "refinery" is found in 1024 items
- tf_{ik} for the given document i
  - TF of "oil" is 4, TF of "Mexico" is 8, TF of "refineries" is 10;
- N: there are 2048 items in the database

$$Weight_{i,\text{ oil}} = 4^* (\text{Log}_2(2048) - \text{Log}_2(128) + 1) = 20$$

$$Weight_{i,\text{ Mexico}} = 8^* (\text{Log}_2(2048) - \text{Log}_2(16) + 1) = 64$$

$$Weight_{i,\text{ refinery}} = 10^* (\text{Log}_2(2048) - \text{Log}_2(1024) + 1) = 20$$

TF-IDF (as defined in textbook)

- **TF-IDF** (Term frequency/Inverse Document frequency) ranking:
  - Let $n(d)$ = number of terms in the document d
  - $n(d, t)$ = number of occurrences of term t in the document d.
  - Relevance of a document d to a *term* t

$$TF(d, t) = log\left(1 + \frac{n(d, t)}{n(d)}\right)$$

    - The log factor is to avoid excessive weight to frequent terms
  - Relevance of document to *query* Q

$$r(d, Q) = \sum_{t \in Q} \frac{TF(d, t)}{n(t)}$$

Other Issues in Indexing

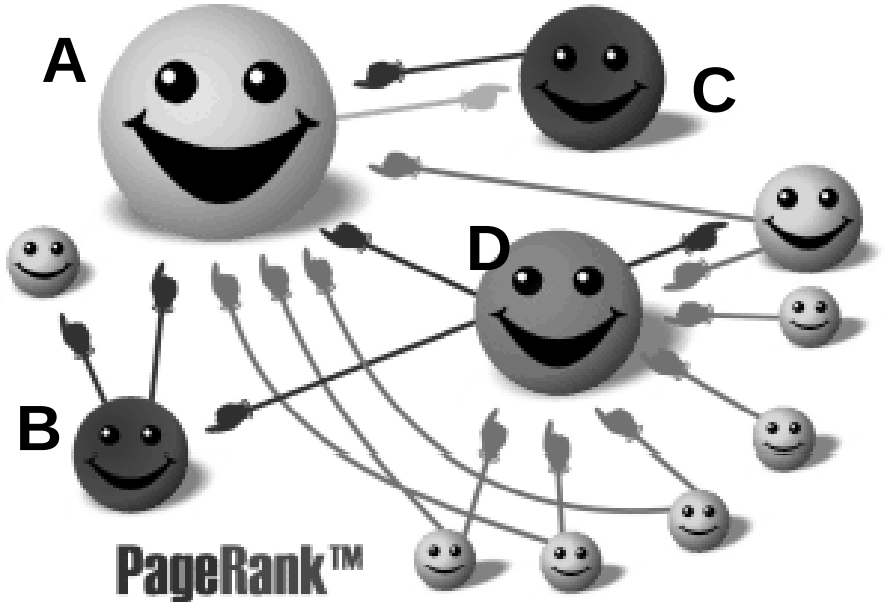
- *co-occurrence*: occur together within a document
- *phrases*: occur consecutively in certain order
- *proximity*: occur together within certain distance
  - if a relationship is observed sufficiently often then it should be included in the vocabulary
- *synonyms* and *homonyms*
  - "jaguar and lion" vs "jaguar and BMW"
  - "motorcycle repair" vs "motorcycle maintenance"
- *stemming*
  - education, educational, educating, …

Similarity Based Retrieval

- Similarity based retrieval - retrieve documents similar to a given document
  - Similarity may be defined on the basis of common words
    - E.g. find *k* terms in A with highest $TF(d, t) / n(t)$ and use these terms to find relevance of other documents.
- Relevance feedback
  - Similarity can be used to refine answer set to keyword query
  - User selects a few relevant documents from those retrieved by keyword query, and system finds other documents similar to these
- Vector space model is used widely for similarity based methods

Relevance Using Hyperlinks

- Most of the time people are looking for pages from popular sites
  - Idea: use popularity of Web site (e.g. how many people visit it) to rank site pages that match given keywords
- Use number of hyperlinks to a site as a measure of the popularity or prestige of the site
  - When computing prestige based on links to a site, give more weight to links from sites that themselves have higher prestige



$$PR(A) = \frac{PR(B)}{L(B)} + \frac{PR(C)}{L(C)} + \frac{PR(D)}{L(D)}.$$

Relevance Using Hyperlinks (Cont.)

- Hub and authority based ranking
  - A **hub** is a page that stores links to many pages (on a topic)
  - An **authority** is a page that contains actual information on a topic
  - Each page gets a **hub prestige** based on prestige of authorities that it points to
  - Each page gets an **authority prestige** based on prestige of hubs that point to it
- Refinements
  - Count only one hyperlink from each site (why?)
  - Popularity measure is for site, not for individual page
    - But, most hyperlinks are to root of site
    - Also, concept of "site" difficult to define since a URL prefix like cs.yale.edu contains many unrelated pages of varying popularity
- Prestige definitions are cyclic, and can be obtained by solving linear equations

Concept-Based Querying

- Approach
  - For each word, determine the concept it represents from context
  - Use one or more ontologies:
    - Hierarchical structure showing relationship between concepts
    - E.g.: the ISA relationship that we saw in the E-R model
- This approach can be used to standardize terminology in a specific field
- Ontologies can link multiple languages
- Foundation of the Semantic Web (not covered here)

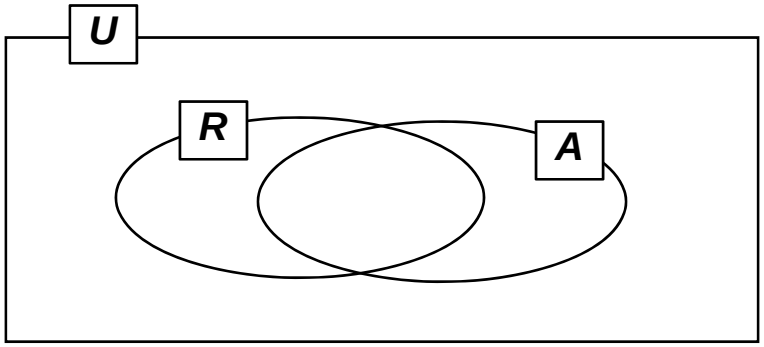
Inverted Index (File)

- Most commonly used index structure for IR
- Maps each keyword K_i to a set of documents S_i that contain the keyword
  - Documents identified by identifiers
- Inverted index may record
  - Keyword locations within document to allow proximity based ranking
  - Counts of number of occurrences of keyword to compute TF
- ***and*** operation: Finds documents that contain all of K_1, K_2, ..., K_n.
  - Intersection $S_1 \cap S_2 \cap ..... \cap S_n$
- ***or*** operation: documents that contain at least one of K_1, K_2, …, K_n
  - union, $S_1 \cup S_2 \cup ..... \cup S_n$,.
- Each S_i is kept sorted to allow efficient intersection/union by merging
  - "**not**" can also be efficiently implemented by merging of sorted lists

Measuring Retrieval Effectiveness

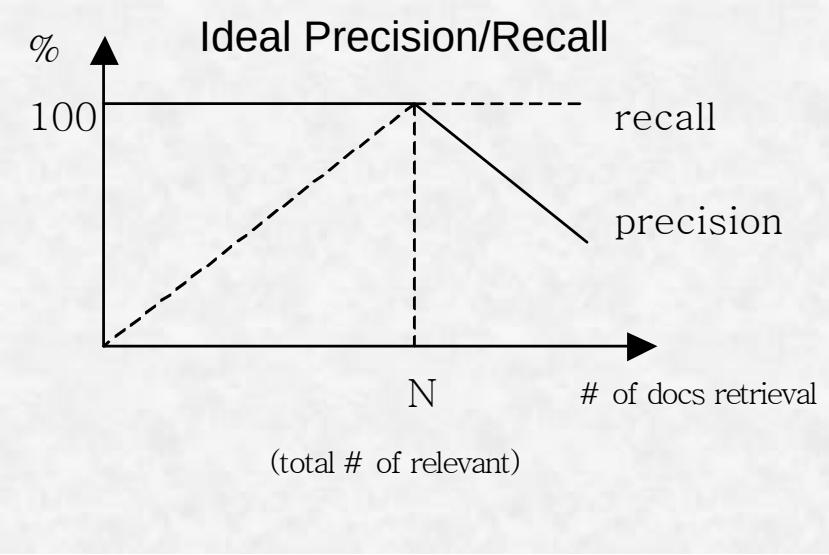
- Approximate retrieval may result in:
  - **false negative (false drop)** - some relevant documents may not be retrieved.
  - **false positive** - some irrelevant documents may be retrieved.
- Relevant performance metrics:
  - **precision** - what percentage of the retrieved documents are relevant to the query.
  - **recall** - what percentage of the documents relevant to the query were retrieved.

Precision & Recall



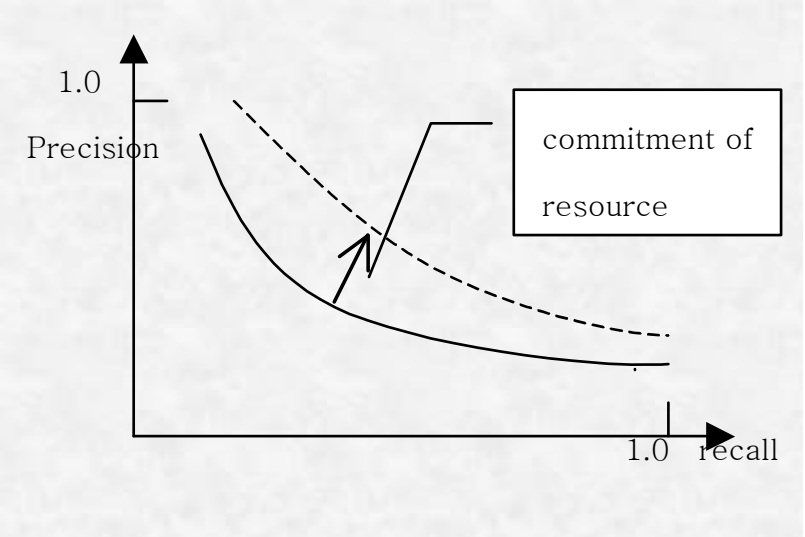
- U : a set of all documents
- R : Relevant document set
- A : Answer document set

- Measuring the effectiveness of an IRS
- Recall
  - card(R ∩ A) / card(R)
- Precision
  - card(R ∩ A) / card(A)



Measuring Retrieval Effectiveness (Cont.)

- Recall vs. precision tradeoff:



- Measures of retrieval effectiveness:
  - Recall as a function of number of documents fetched, or
  - Precision as a function of recall
    - Equivalently, as a function of number of documents fetched
  - E.g. "precision of 75% at recall of 50%, and 60% at a recall of 75%"
- Problem: which documents are actually relevant, and which are not

Web Search Engines

- **Web crawlers** are programs that locate and gather information on the Web
  - Recursively follow hyperlinks present in known documents, to find other documents
    - Starting from a *seed* set of documents
  - Fetched documents
    - Handed over to an indexing system
    - Can be discarded after indexing, or store as a *cached* copy
- Crawling the entire Web would take a very large amount of time
  - Search engines typically cover only a part of the Web, not all of it
  - Take months to perform a single crawl

Web Crawling

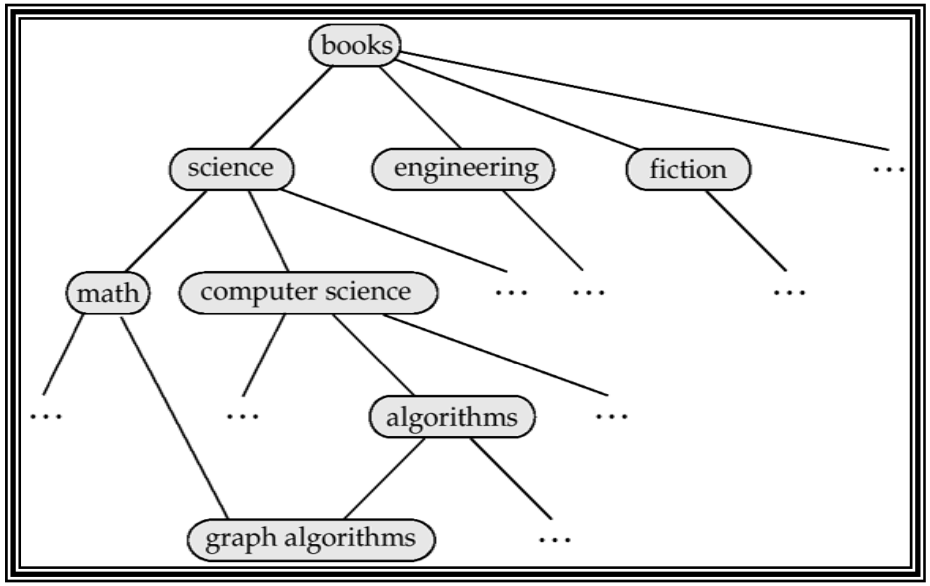
- Crawling is done by multiple processes on multiple machines, running in parallel
  - Set of links to be crawled stored in a database
  - New links found in crawled pages added to this set, to be crawled later
- Indexing process also runs on multiple machines
  - Creates a new copy of index instead of modifying old index
  - Old index is used to answer queries
  - After a crawl is "completed" new index becomes "old" index
- Multiple machines used to answer queries
  - Indices may be kept in memory
  - Queries may be routed to different machines for load balancing

Information Retrieval and Structured Data

- Information retrieval systems originally treated documents as a collection of words
- Information extraction systems infer structure from documents, e.g.:
  - Extraction of house attributes (size, address, number of bedrooms, etc.) from a text advertisement
  - Extraction of topic and people named from a new article
- Relations or XML structures used to store extracted data
  - System seeks connections among data to answer queries
  - Question answering systems

Web Directories

- A Web directory is a classification directory on Web pages
  - Yahoo! Directory, Open Directory project
  - Issues:
    - What should the directory hierarchy be?
    - Given a document, which nodes of the directory are categories relevant to the document
  - Often done manually
    - Classification of documents into a hierarchy may be done based on term similarity



END OF CHAPTER 19