CS54701

Basic Concepts of Information Retrieval

Luo Si
Department of Computer Science
Purdue University

Basic Concepts of IR: Outline

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

Ad-hoc IR: Terminologies

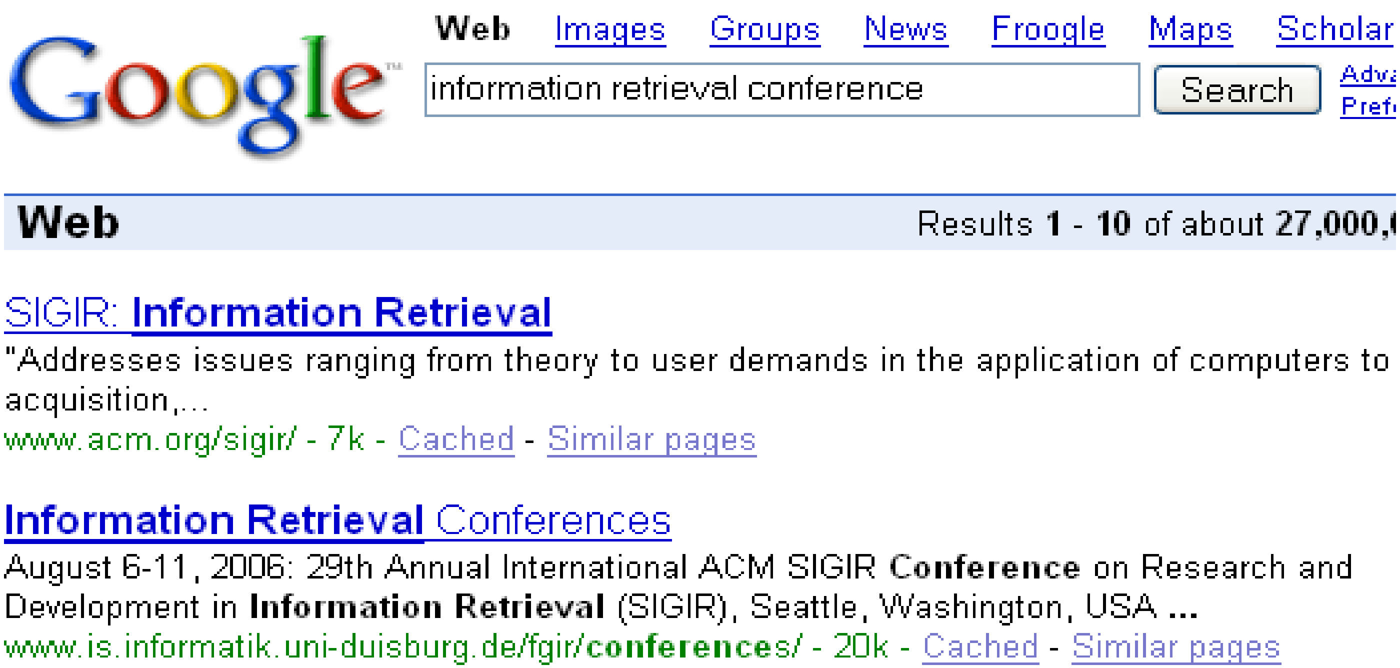
Terminologies:

- Query
  - Representative data of user's information need: text (default) and other media
- Document
  - Data candidate to satisfy user's information need: text (default) and other media
- Database|Collection|Corpus
  - A set of documents
- Corpora
  - A set of databases
  - Valuable corpora from TREC (Text Retrieval Evaluation Conference)

Ad-hoc IR: Introduction

Ad-hoc Information Retrieval:

- Search a collection of documents to find relevant documents that satisfy different information needs (i.e. queries)
- Example: Web search



Ad-hoc IR: Introduction

Ad-hoc Information Retrieval:

- Search a collection of documents to find relevant documents that satisfy different information needs (i.e. queries)

  Relatively Stable

  Changes

  - Queries are created and used dynamically; change fast
  - "Ad-hoc": formed or used for specific or immediate problems or needs" – Merriam-Webster's collegiate Dictionary

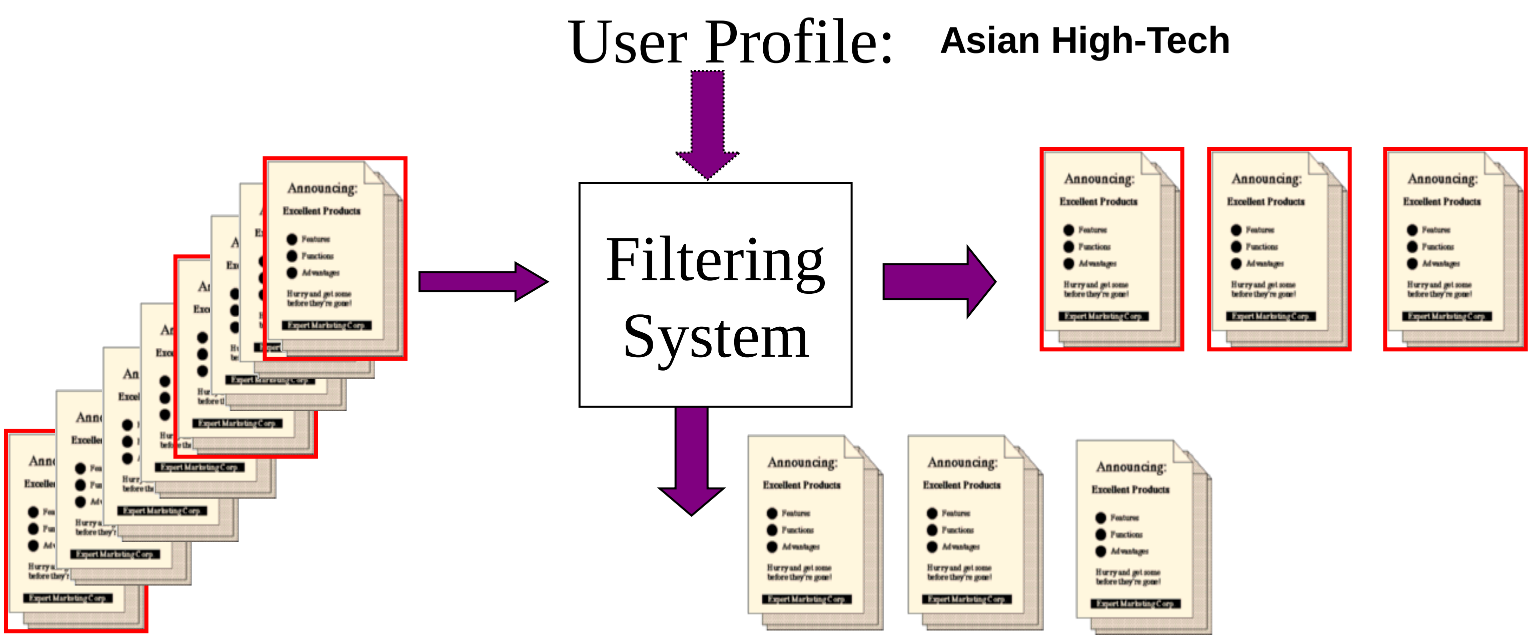
Ad-hoc IR vs. Filtering

- Filtering: Queries are stable (e.g., Asian High-Tech) while the collection changes (e.g., news)
- More for filtering in later lectures

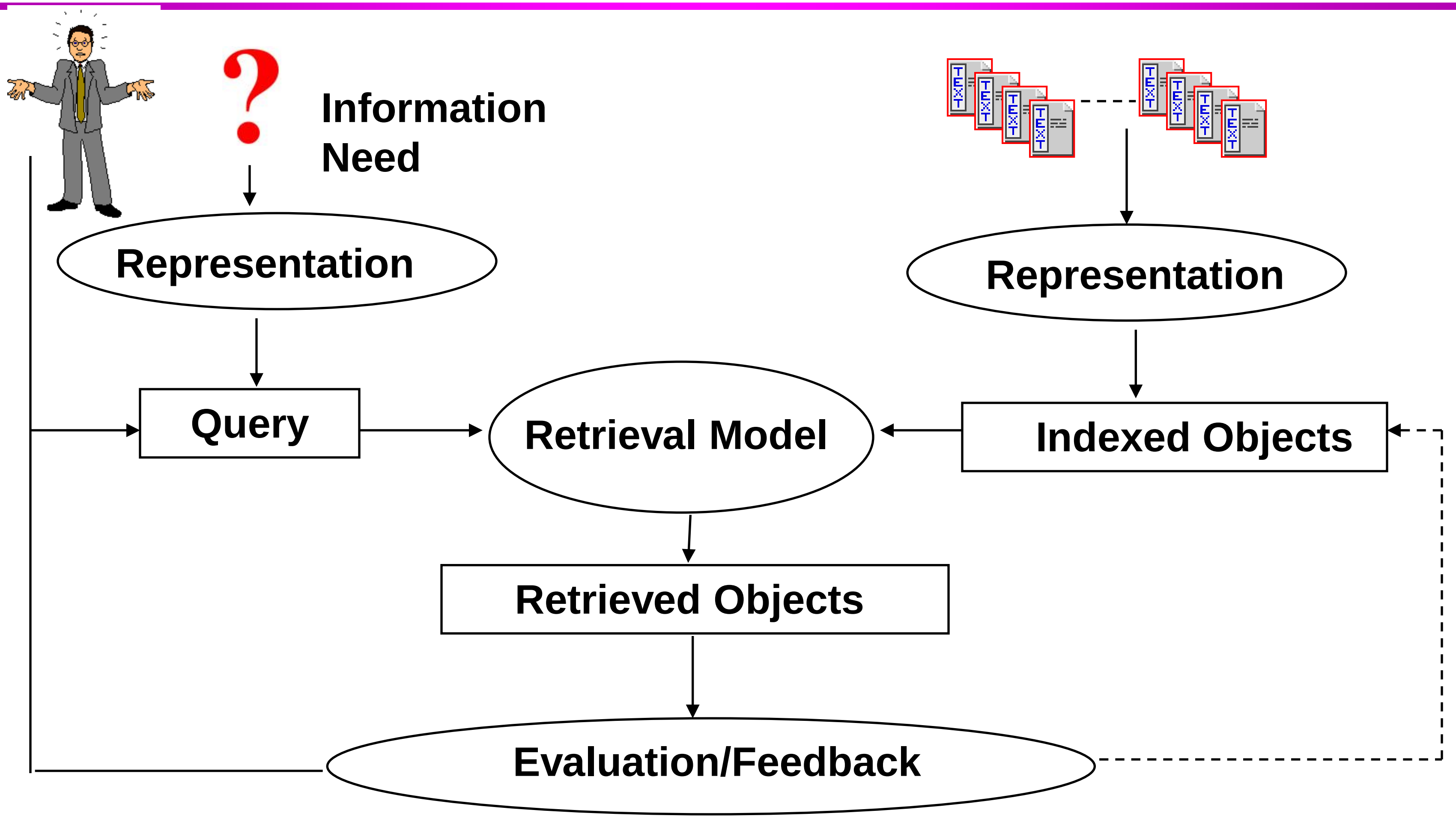
Content Based Filtering Filtering

Information Needs are Stable

System should make a delivery decision on the fly when a document “arrives”



AD-hoc IR: Basic Process



AD-hoc IR: Overview of Retrieval Model

Retrieval Models

- Boolean
- Vector space
  - Basic vector space SMART, LUCENE
  - Extended Boolean
- Probabilistic models
  - Statistical language models Lemur
  - Two Possion model Okapi
  - Bayesian inference networks Inquery
- Citation/Link analysis models
  - Page rank Google
  - Hub & authorities Clever

AD-hoc IR: Overview of Retrieval Model

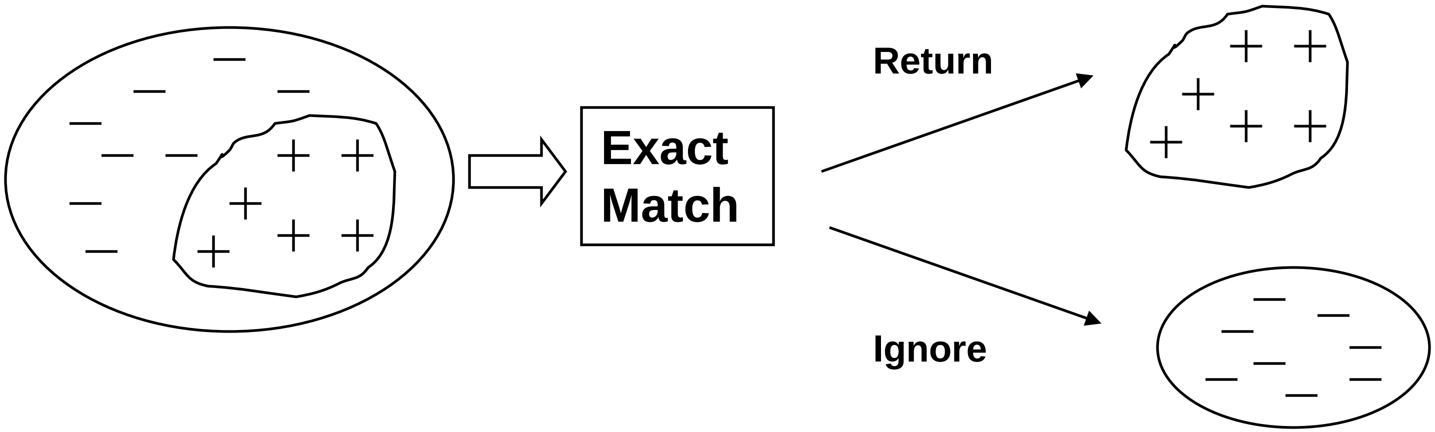
Retrieval Model

Determine whether a document is relevant to query

- Relevance is difficult to define
  - Varies by judgers
  - Varies by context (i.e., jointly by a set of documents and queries)
- Different retrieval methods estimate relevance differently
  - Word occurrence of document and query
  - In probabilistic framework, P(query|document) or P(Relevance|query,document)
  - Estimate semantic consistency between query and document

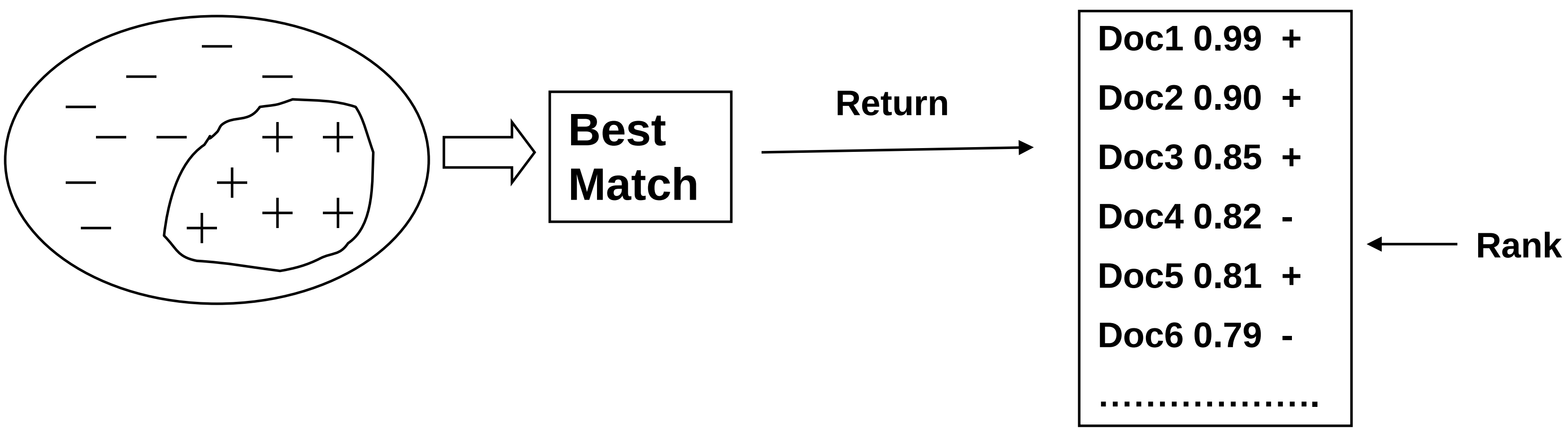
Types of Retrieval Models

- Exact Match (Document Selection)
  - Example: Boolean Retrieval Method
  - Query defines the exact retrieval criterion
  - Relevance is a binary variable; a document is either relevant (i.e., match query) or irrelevant (i.e., mismatch)
  - Result is a set of documents
    - Documents are unordered
    - Often in reverse-chronological order (e.g., Pubmed)



Types of Retrieval Models

- Best Match (Document Ranking)
  - Example: Most probabilistic models
  - Query describes the desired retrieval criterion
  - Degree of relevance is a continuous/integral variable; each document matches query to some degree
  - Result in a ranked list (top ones match better)
    - Often return a partial list (e.g., rank threshold)



Types of Retrieval Models

Exact Match (Selection) vs. Best Match (Ranking)

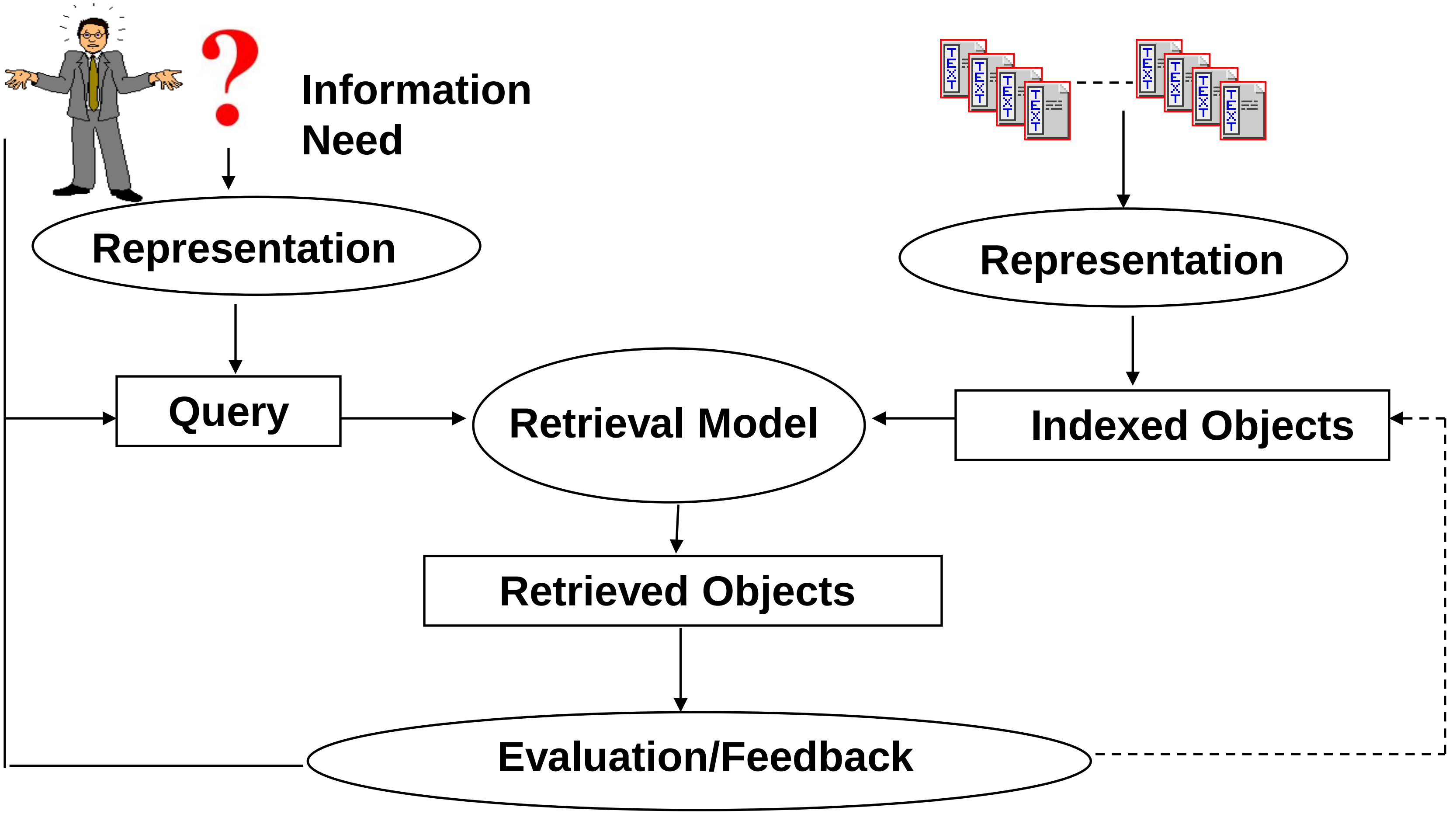
- Best Match is usually more accurate/effective
  - Do not need precise query; representative query generates good results
  - Users have control to explore the rank list: view more if need every piece; view less if need one or two most relevant
- Exact Match
  - Hard to define the precise query; too strict (terms are too specific) or too coarse (terms are too general)
  - Users have no control over the returned results
  - Still prevalent in some markets (e.g., legal retrieval)

AD-hoc IR: Overview of Retrieval Model

Retrieval Models

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

AD-hoc IR: Basic Process



Text Representation: What you see

It never leaves my side, April 6, 2002

Reviewer:"dage456" (Carmichael, CA USA) - See all my reviewsIt fits in the palm of your hand and is the size of a deflated wallet (wonder where the money went). I have had my ipod now for 4 months and cannot imagine how I used to get by with my old rio 600 with is 64 megs of ram and.. usb connection. Because of its size this little machine goes with my everywhere and its ten hour battery life means I can listen to stuff all day long.

Pros: size, both physical and capacity.
design: It looks beautiful
controls: simple and very easy to use
connection: FIREWIRE!!

Cons: needs the ability to bookmark. I use my ipod mostly for audiobooks. the ipod needs to include a bookmark feature for those like me.

From Amazon Customer Review of IPod

Text Representation: What computer see

<table><tr><td valign="top">
Reviewer:</td>

**<td>"dage456" (Carmichael, CA USA) - **

**See all my reviews</td></tr></table>It fits in the palm of your hand and is the size of a deflated wallet (wonder where the money went). <p>I have had my ipod now for 4 months and cannot imagine how I used to get by with my old rio 600 with is 64 megs of ram and.. usb connection. Because of its size this little machine goes with my everywhere and its ten hour battery life means I can listen to stuff all day long.<p>Pros: size, both physical and capacity.
design: It looks beautiful
controls: simple and very easy to use<p>connection: FIREWIRE!!<p>Cons: needs the ability to bookmark. I use my ipod mostly for audiobooks. the ipod needs to include a bookmark feature for those like me.

**

From Amazon Customer Review of IPod

Text Representation: TREC Format

<DOC>
<DOCNO> AP900101-0001 </DOCNO>
<FILEID>AP-NR-01-01-90 2345EDT</FILEID>
<FIRST>r i PM-Iran-Population Bjt 01-01 0777</FIRST>
<SECOND>PM-Iran-Population, Bjt,0800</SECOND>
<HEAD>Iran Moves To Curb A Baby Boom That Threatens Its Economic Future</HEAD>
<HEAD>An AP Extra</HEAD>
<BYLINE>By ED BLANCHE</BYLINE>
<BYLINE>Associated Press Writer</BYLINE>
<DATELINE>NICOSIA, Cyprus (AP) </DATELINE>
<TEXT>
Iran's government is intensifying a birth
control program _ despite opposition from radicals _ because the
country's fast-growing population is imposing strains on a
struggling economy.
…………
</TEXT>
</DOC>

Text Representation: Indexing

Indexing

Associate document/query with a set of keys

- Manual or human Indexing
  - Indexers assign keywords or key concepts (e.g., libraries, Medline, Yahoo!); often small vocabulary
  - Significant human efforts, may not be thorough
- Automatic Indexing
  - Index program assigns words, phrases or other features; often large vocabulary
  - No human efforts

Text Representation: Indexing

Controlled Vocabulary vs. Full Text

- Controlled Vocabulary Indexing
  - Assign words from a small vocabulary or a node from an ontology
  - Often manually but can be done by learning algorithms
- Full Indexing:
  - Often index with an uncontrolled vocabulary of full text
  - Automatically while good algorithm can generate more representative keywords/ key concepts

Text Representation: Indexing
Controlled Vocabulary

Mutation of a mutL homolog in hereditary colon cancer.

Papadopoulos N, **Nicolaides NC**, **Wei YF**, **Ruben SM**, **Carter KC**, **Rosen CA**, **Haseltine WA**, **Fleischmann RD**, **Fraser CM**, **Adams MD**, et al.

Johns Hopkins Oncology Center, Baltimore, MD 21231.

Some cases of hereditary nonpolyposis colorectal cancer (HNPCC) are due to alterations in a mutS-related mismatch repair gene. A search of a large database of expressed sequence tags derived from random complementary DNA clones revealed three additional human mismatch repair genes, all related to the bacterial mutL gene. One of these genes (hMLH1) resides on chromosome 3p21, within 1 centimorgan of markers previously linked to cancer susceptibility in HNPCC kindreds. Mutations of hMLH1 that would disrupt the gene product were identified in such kindreds, demonstrating that this gene is responsible for the disease. These results suggest that defects in any of several mismatch repair genes can cause HNPCC.

Text Representation: Indexing
Controlled Vocabulary

MeSH Tree Structures

1. Anatomy [A]
2. Organisms [B]
3. Diseases [C]
   - Bacterial Infections and Mycoses [C01] -
   - Virus Diseases [C02] +
   - Parasitic Diseases [C03] +
   - Neoplasms [C04] +
   - Musculoskeletal Diseases [C05] +
   - Digestive System Diseases [C06] +
4. Chemicals and Drugs [D]
5. Analytical, Diagnostic and Therapeutic T
6. Psychiatry and Psychology [F]
7. Biological Sciences [G]
8. Physical Sciences [H]

Neoplasms by Site
- Digestive System Neoplasms
  - Gastrointestinal Neoplasms
    - Intestinal Neoplasms
      - Colorectal Neoplasms
        - Colorectal Neoplasms, Hereditary Nonpolyposis

Text Representation: Indexing
Controlled Vocabulary

```
PMID- 8128251
TI  - Mutation of a mutL homolog in hereditary colon cancer.
MH  - *Adenosinetriphosphatase
MH  - Amino Acid Sequence
MH  - Bacterial Proteins/chemistry/*genetics
MH  - Base Sequence
MH  - Carrier Proteins
MH  - Chromosome Mapping
MH  - *Chromosomes, Human, Pair 3
MH  - Codon
MH  - Colorectal Neoplasms, Hereditary Nonpolyposis/*genetics
MH  - *DNA Repair
MH  - *DNA-Binding Proteins
```

Text Representation: Indexing
Controlled Vocabulary

Pros and cons of controlled vocabulary indexing

- Advantages
  - Many available vocabularies/ontologies (e.g., MeSH, Open Directory, UMLS)
  - Normalization of indexing terms: less vocabulary mismatch, more consistent semantics
  - Easy to use by RDBMS (e.g., semantic Web)
  - Support concept based retrieval and browsing
- Disadvantages
  - Substantial efforts to be assigned manually
  - Inconvenient for users not familiar with the controlled vocabulary
  - Coarse representation of semantic meaning

Text Representation: Indexing
Full Text Indexing

Full text Indexing: index all text with uncontrolled vocabulary

- Advantages
  - (Possibly) Keep all the information within the text
  - Often no human efforts; easy to build
- Disadvantages
  - Difficult to cross vocabulary gap (e.g., “cancer” in query, “neoplasm” in document)
  - Large storage space

How to build full text Indexing:

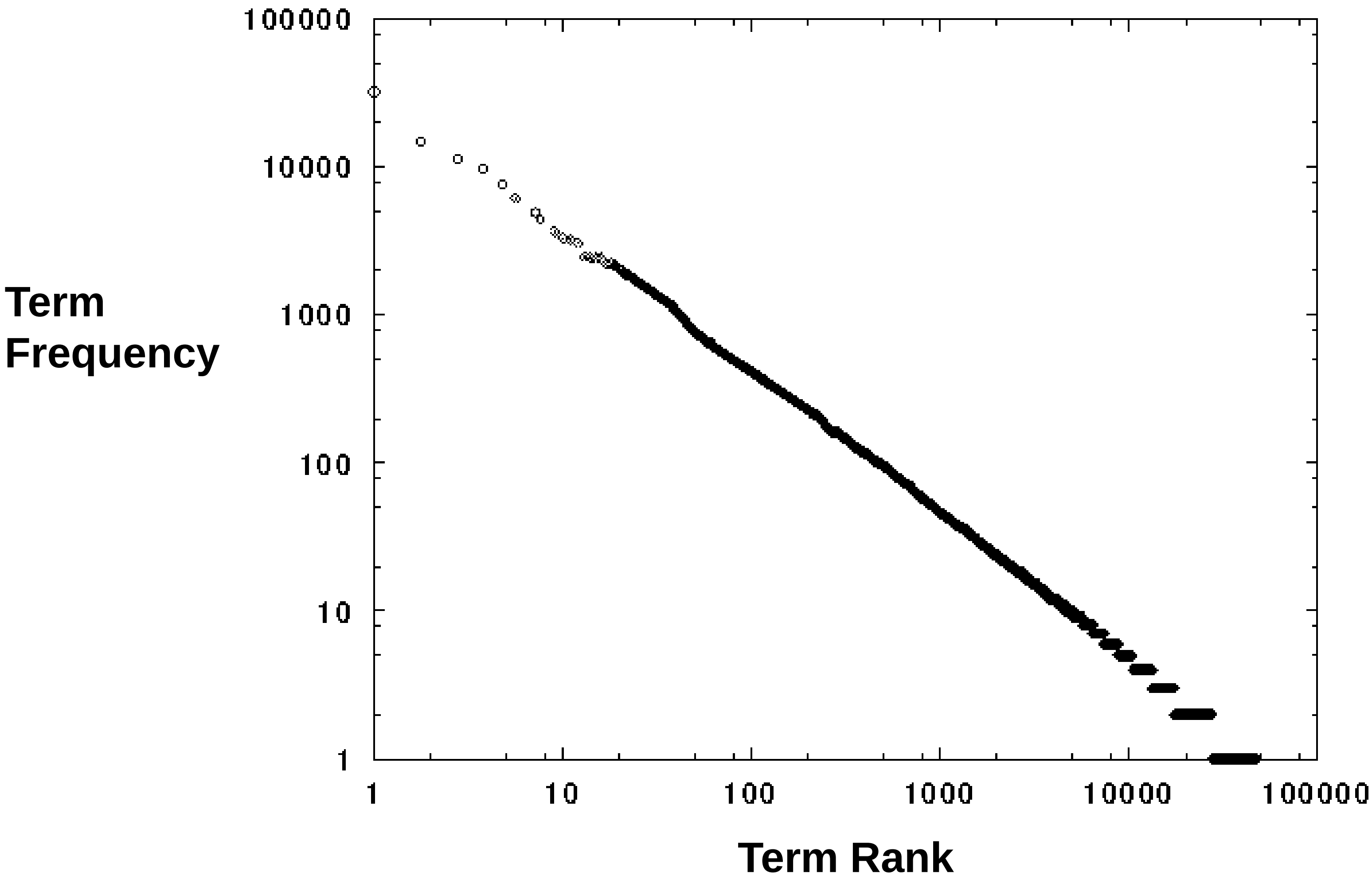
- What are the candidates in the word vocabulary? Are they effective to represent semantic meanings
- How to bridge small vocabulary gap (e.g., car and cars)

Text Representation: Indexing
Statistical Properties of Text

Word	Frequency	Word	Frequency
the	1130021	market	52110
of	547311	bank	47940
to	516636	stock	47401
a	464736	trade	47310
in	390819	…	…
and	387703	…	...
….	…	…	…

Statistics collected from Wall Street Journal (WSJ), 1987

Text Representation: Indexing
Statistical Properties of Text



Text Representation: Indexing
Statistical Properties of Text

Observations from language/corpus independent features

- A few words occur very frequently (High Peak)
  - Top 2 words: 8%-15% (e.g., words that carry no semantic meanings like “the”, “to”)
- Most words occur rarely (Heavy Tail)
- Representative words often in the middle
  - e.g., market and stock for WSJ
- Rules formally describe word occurrence patterns: Zipf’s law, Heaps’ Law

Text Representation: Indexing
Statistical Properties of Text

Zipf's law: relate a term's frequency to its rank

- Rank all terms with their frequencies in descending order, for a term at a specific rank (e.g., r) collects and calculates

f_r **: term frequency**

$$p_r = \frac{f_r}{N}$$ **: relative term frequency**

Total number of words

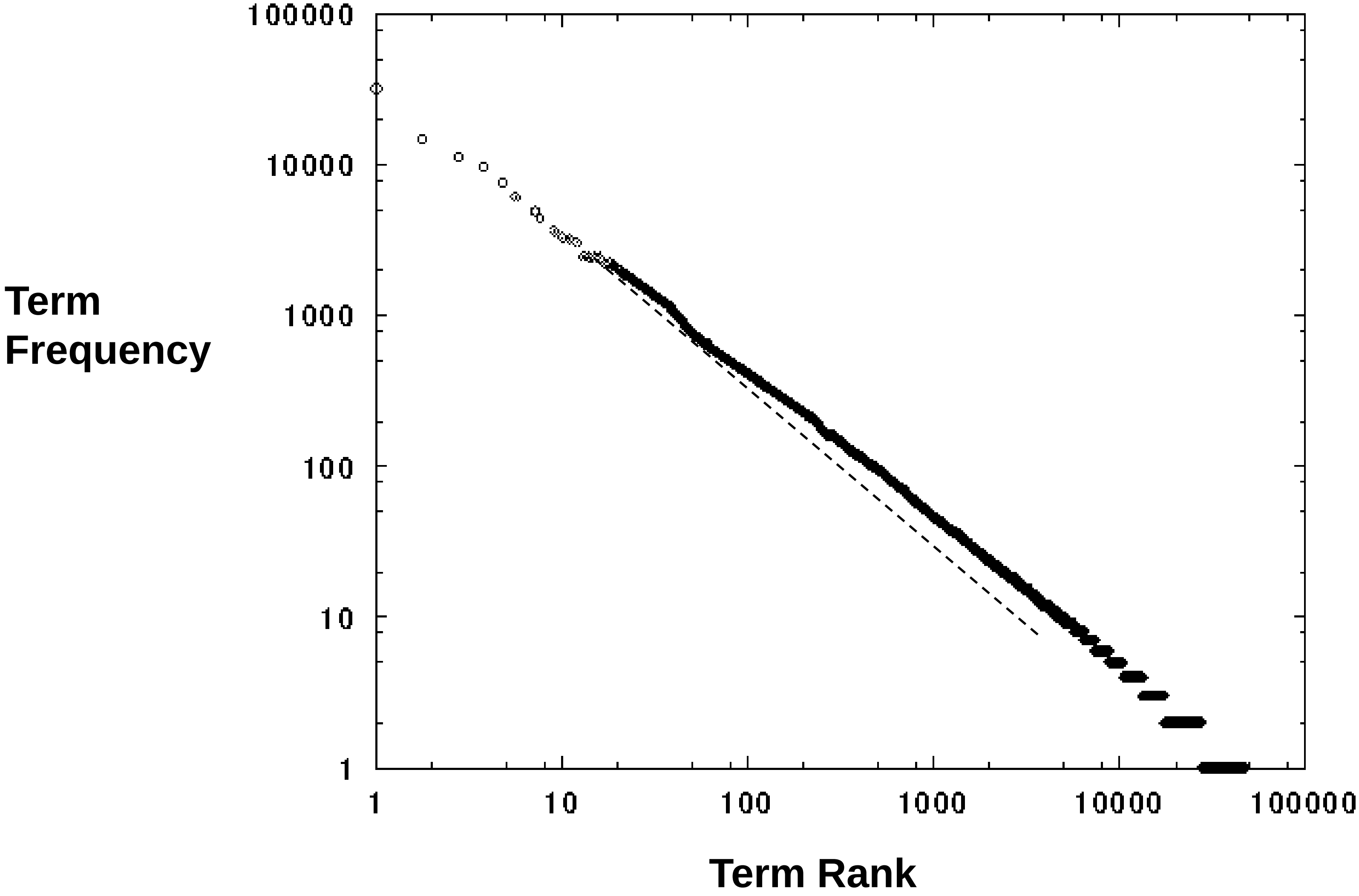
- Zipf's law (by observation):

$$p_r = A / r \qquad A \approx 0.1$$

So $$p_r = \frac{f_r}{N} = \frac{A}{r} \Rightarrow rf_r = AN \Rightarrow \log(r) = -\log(f_r) + \log(AN)$$

So Rank X Frequency = Constant

Text Representation: Indexing
Statistical Properties of Text



Text Representation: Indexing
Statistical Properties of Text

Word	Frequency	$r * p_r$	Word	Frequency	$r * p_r$
the	1130021	0.059	market	52110	0.101
of	547311	0.058	bank	47940	0.109
to	516636	0.082	stock	47401	0.110
a	464736	0.098	trade	47310	0.112
in	390819	0.103	…	…	…
and	387703	0.122	…	...	…
….	…	….	…	…	….

Statistics collected from Wall Street Journal (WSJ), 1987

Text Representation: Text Preprocessing

Text Preprocessing: extract representative index terms

- Parse query/document for useful structure
  - E.g., title, anchor text, link, tag in xml…..
- Tokenization
  - For most western languages, words separated by spaces; deal with punctuation, capitalization, hyphenation
  - For Chinese, Japanese: more complex word segmentation…
- Remove stopwords: (remove “the”, “is”,..., existing standard list)
- Morphological analysis (e.g., stemming):
  - Stemming: determine stem form of given inflected forms
- Other: extract phrases; decompounding for some European languages “*rörelseuppskattningssökningsintervallsinställningar*”

Text Representation: Text Preprocessing

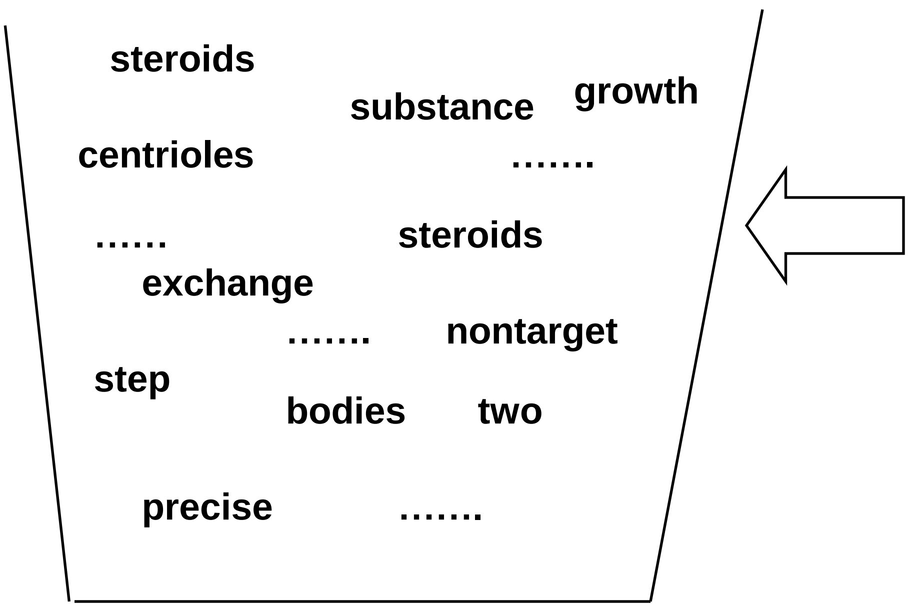
4 the	1 at	1 different	1 may	1 step
3 and	1 basal	1 exchange	1 nontarget	1 substance
3 by	1 be	1 exogenous	1 not	1 suggests
3 steroids	1 been	1 fluorescent	1 may	1 target
2 centrioles	1 bodies	1 from	1 of	1 technique
2 in	1 can	1 growth	1 precise	1 two
1 affect	1 at	1 has	1 receptor	1 unexpected
1 already	1 cell	1 identity	1 regularly	1 vitally
1 Although	1 cells	1 level	1 reveal	1 way
1 antibodies	1 cilia-bearing	1 localization	1 Specific	1 with

24 stopwords out of total 61 words

Text Representation: Bag of Words

The simplest text representation: “bag of words”

- Query/document: a bag that contains words in it
- Order among words is ignored



3 steroids	1 cilia-bearing	1 precise	1 two
2 centrioles	1 different	1 receptor	1 unexpected
1 affect	1 exchange	1 regularly	1 vitally
1 already	1 exogenous	1 reveal	1 way
1 Although	1 fluorescent	1 Specific	
1 antibodies	1 growth	1 step	
1 basal	1 identity	1 substance	
1 bodies	1 level	1 suggests	
1 cell	1 localization	1 target	
1 cells	1 nontarget	1 technique	

Text Representation: Phrases

- Single word/stem indexing may not be sufficient

  e.g., “hit a home run yesterday”

- More complicated indexing includes phrases (thesaurus classes)

- How to automatically identify phrases
  - Dictionary
  - Find the most common N word phrases by corpus statistics (be careful of stopwords)
  - Syntactic analysis, noun phrases
  - More sophisticated segmentation algorithm like “Hidden Markov Model”

Text Representation: Word Stemming

Word Stemming

- Associate morphological variants of words into a single form
  - E.g., plurals, adverbs, inflected word forms
  - May lose the precise meaning of a word

- Different types of stemming algorithms
  - Rule-based systems: Porter Stemmer, Krovetz Stemmer

    Porter Stemmer Example: describe/describes -> describ
  - Statistical method: Corpus-based stemming

Text Representation: Word Stemming

Porter Stemmer

- It is based on a pattern of vowel-consonant sequence
  - $[C](VC)^m[V]$, m is an integer
- Rules are divided into steps and examined in sequence
  - Step 1a: ies → i; s →; …..

    cares → care
  - Step 1b: if m>0 eed   ee

    agreed → agree
  - ….. Step 5a, Step 5b
- Pretty aggressively:
  - nativity → native

Text Representation: Word Stemming

K Stemmer: based on morphological rules

- If word occurs in a dictionary, do not stem it
- For all other words
  - Remove inflectional endings: plurals to singular; paste tense to present tense; remove “ing”
  - Remove derivational endings by a sequence of rules: may make mistake when suffixes indicate different meanings like “sign” to “signify”

Text Representation: Word Stemming

Examples of Stemming:

- Original Text:

  Information retrieval deals with the representation, storage, organization of, and access to information items

- Porter Stemmer (Stopwords removed):

  Online example: http://maya.cs.depaul.edu/~classes/ds575/porter.html

  Inform retrieve deal represent storag organ access inform item

Text Representation: Word Stemming

Problems with Rule-based Stemming

- Rule-based stemming may be too aggressive

  e.g., execute/executive, university/universe

- Rule-based stemming may be too conservative

  e.g., European/Europe, matrices/matrix

- It is difficult to understand the meaning the stems

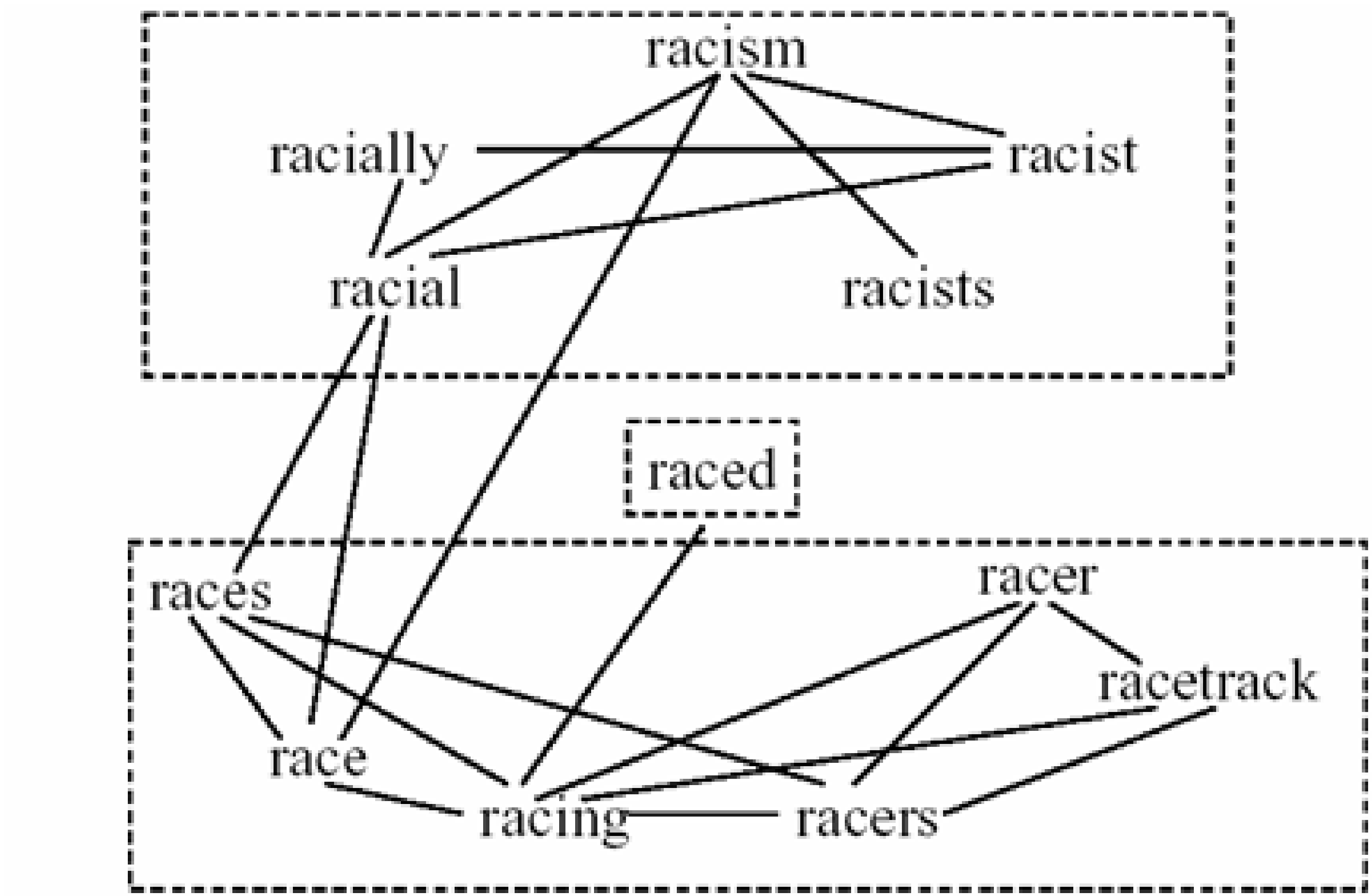
  e.g., Iteration/iter, general/gener

Text Representation: Word Stemming

Corpus-Based Stemming

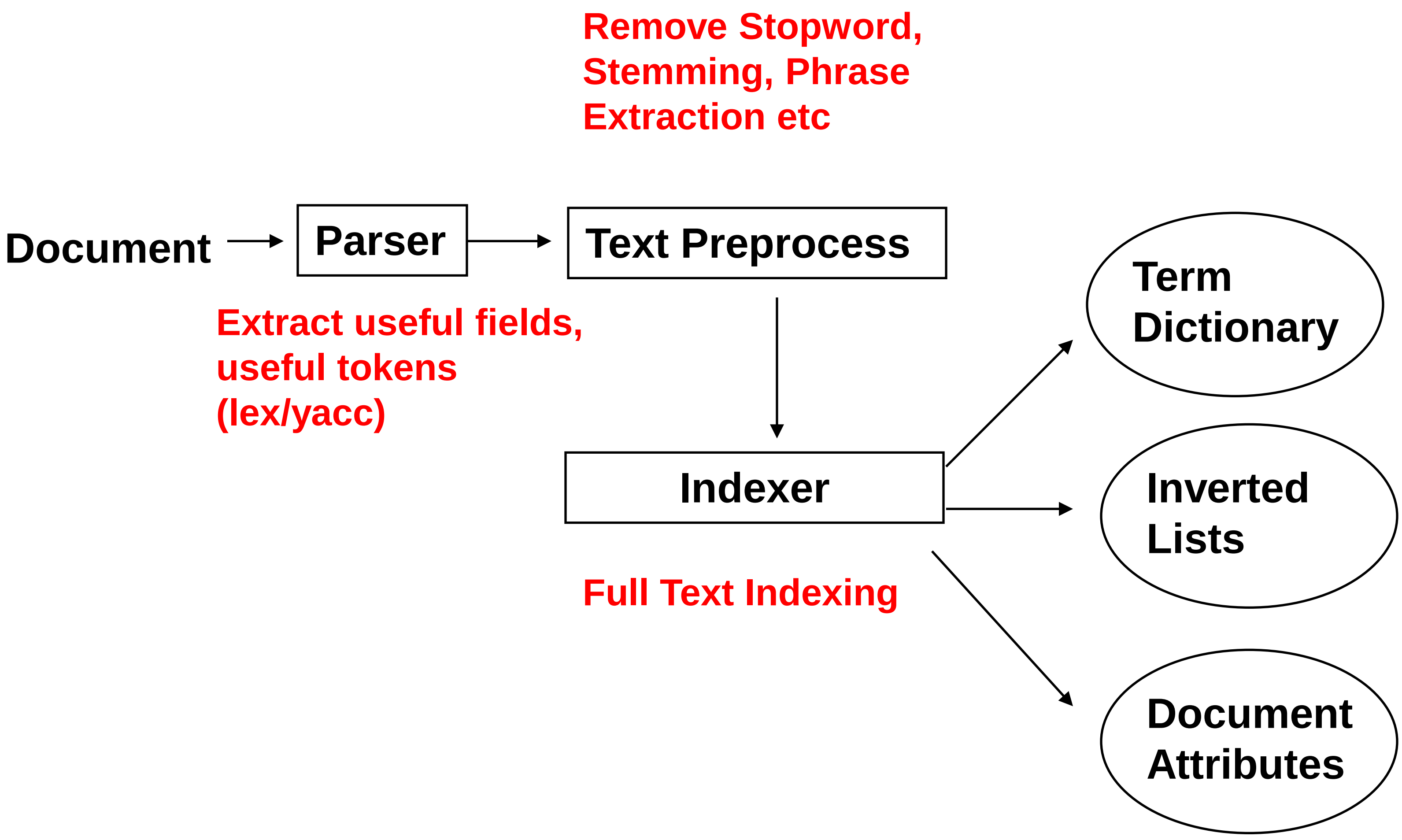
- Hypothesis: Word variants that should be considered equally often co-occur in documents (passages or text windows) in the corpus
- Collect the statistics of co-occurrence of words in the corpus and form the connected graph
- Cut the graph by different methods and find the connected subgraphs to form equivalence classes

Text Representation: Word Stemming



(Xu & Croft, 1998)

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

Documents

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

Inverted Lists

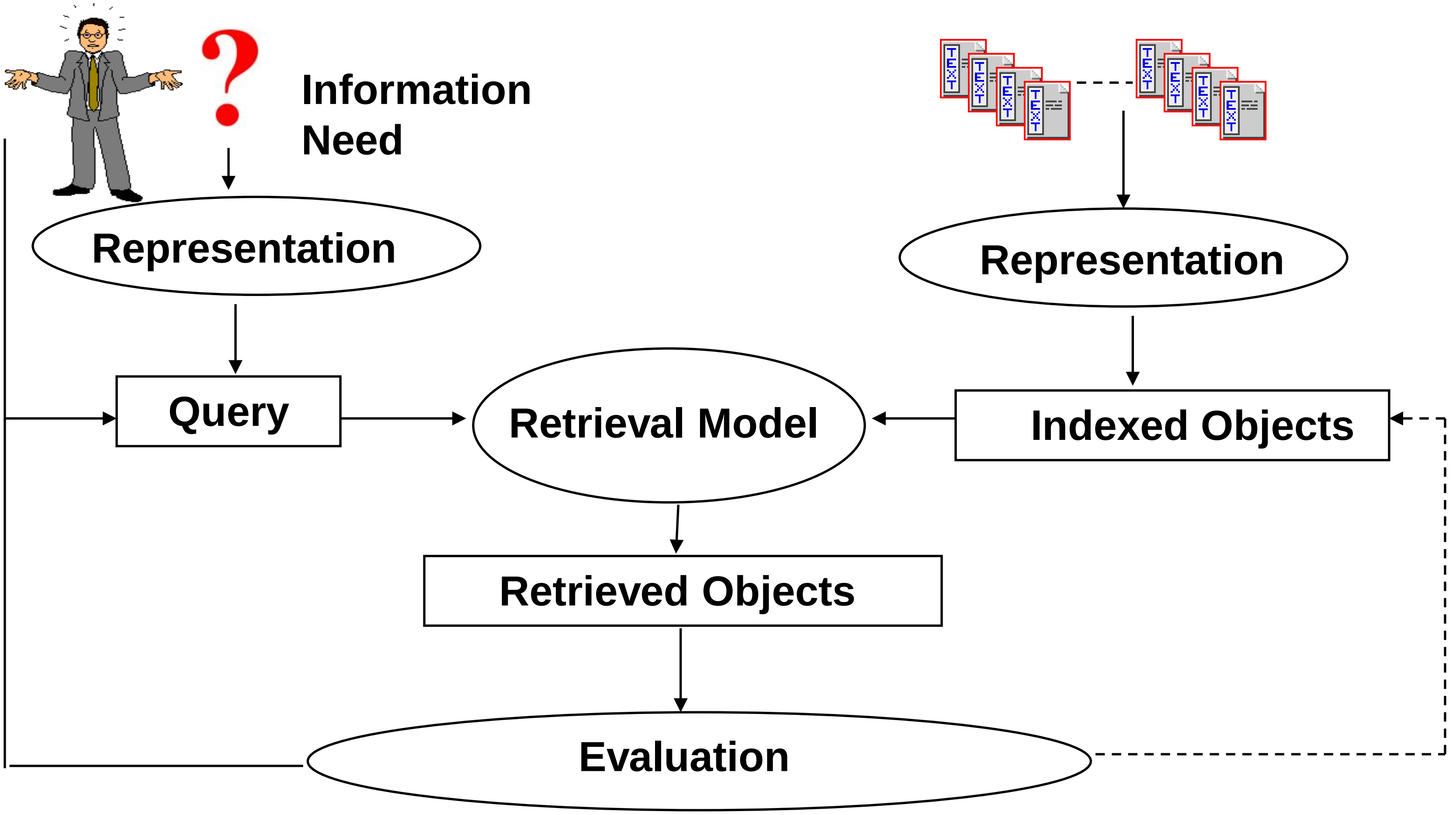
Term ID	Term	Documents
1	kids	1,3
2	question	1,2,3,4
3	nothing	1,4
4	in	1,2,3,4
5	19060s	1
6	young	2,4
7	man	2,4
8	everything	2
9	1970s	2
10	questions	3
11	1980s	3
11	2000s	4

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)

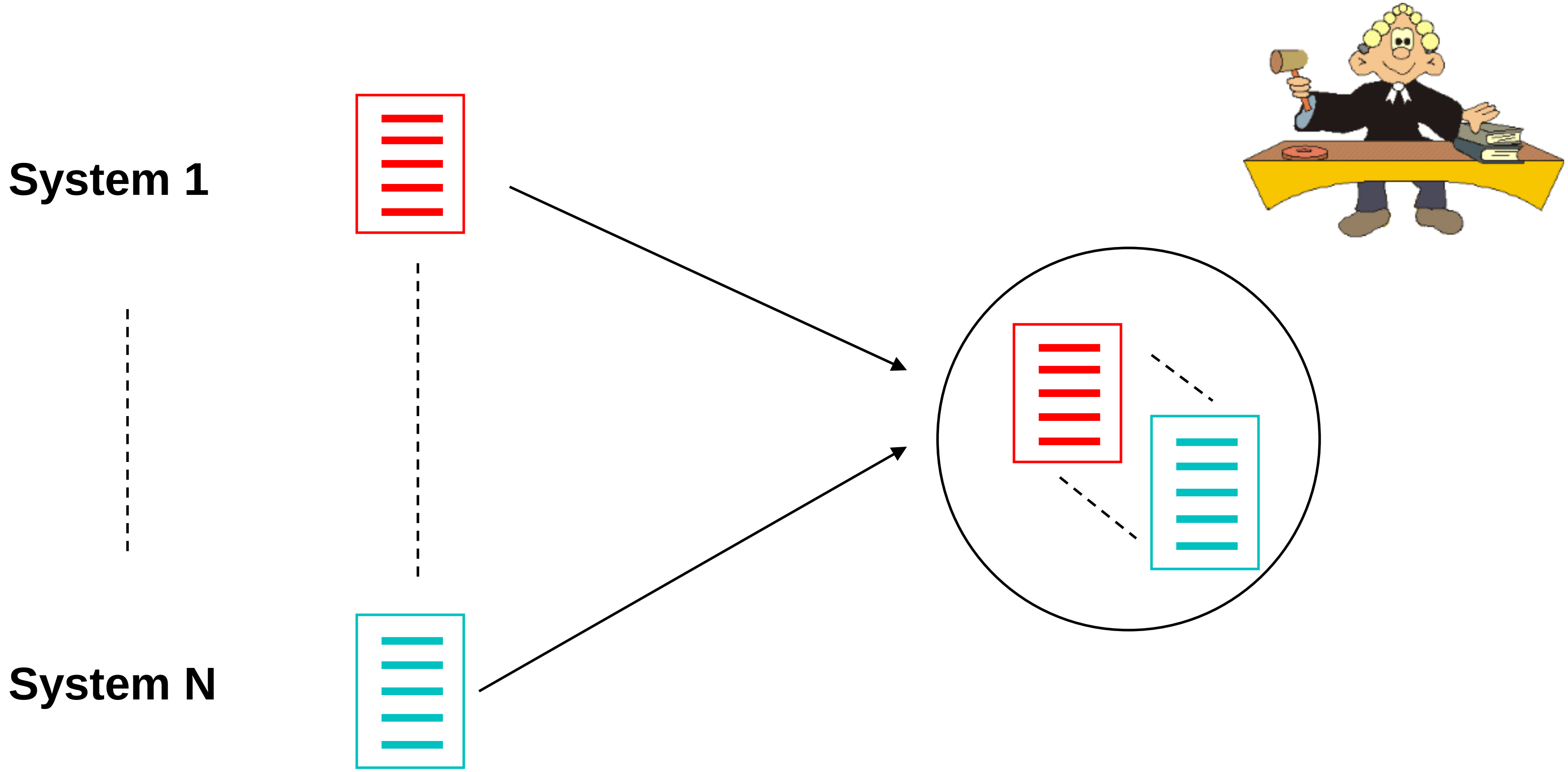
The Pooling process

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

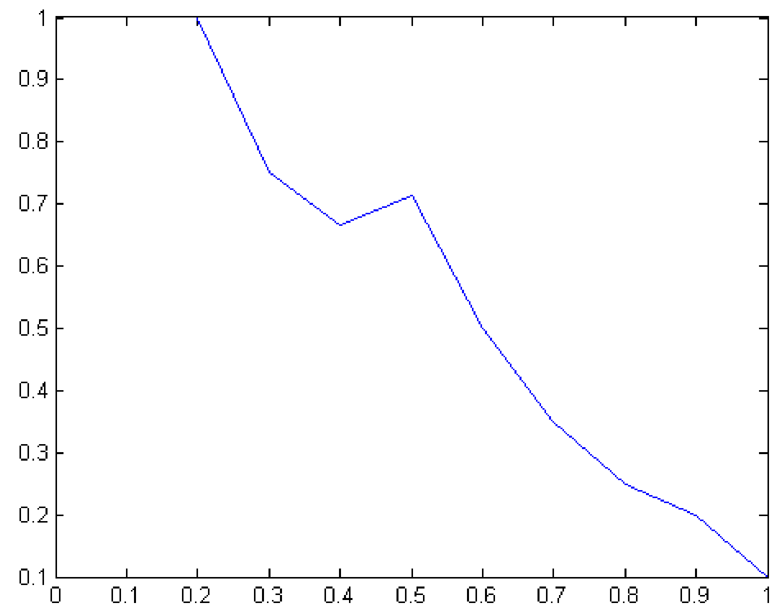
Precision at Recall

- 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

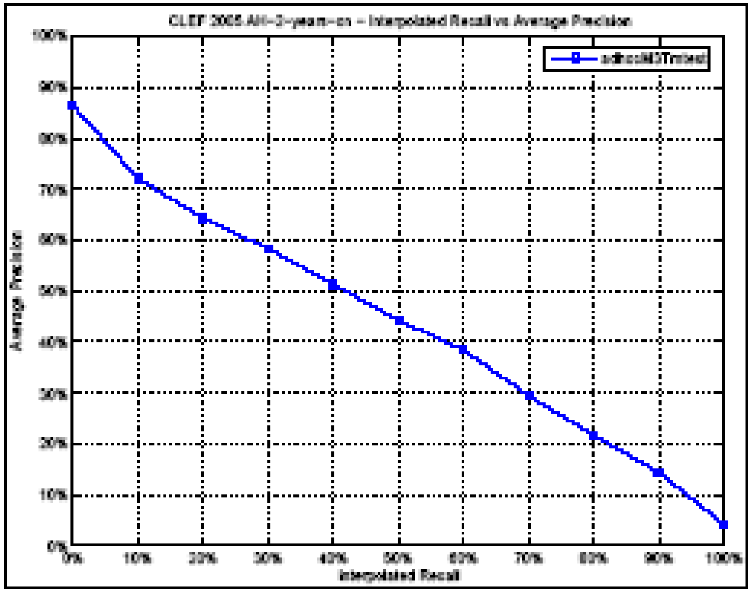
Single value metrics

- 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

Interploated 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