Information Retrieval

**Search**
- Using a computer to find relevant pieces of information

**Text search**
- Idea popularized in the article *As We May Think* by Vannevar Bush in 1945

**Where (or for what) do you do text search?**
- World Wide Web
  - Using, e.g., Google, Yahoo
- Library catalog
- Personal (desktop) search
  - Email, files
- Within a document
  - Search-n-replace a word
- Specific domain/database
  - Medline (free)
  - Westlaw (for a fee)

**Terminology**
- **Query**
  - What you tell the computer to look for
- **Document**
  - What you are hoping to find
  - A webpage that contains the info you’re after
  - A specific file on your computer
  - A specific email in your mail box

**Type of search**
- **Flat text**
  - Query: robot vision
- **Quoted phrases**
  - Query: “robot vision”
- **Fielded search**
Type of search
- Flat text
  - Query: robot vision
- Quoted phrases
  - Query: “robot vision”
- Fielded search
- Boolean operators
  - Query: flu and swine not human

The process
User issues a query
Query is matched to docs in database
"Relevant" docs are returned
Examples:
- Book titles in library catalog
- Webpages on the WWW

"Relevant" docs are ranked
"Relevant" docs are returned with ranking

Finding and comparing documents
The vector space model is one method that performs a ranked search
- Represent a document as a vector, i.e., a list of individual words
- Represent the query as a vector
- Compare the two vectors mathematically

Document → Vector (simple version)
I saw a sloth play soccer with a tortoise and a snail.

Compare document with query
Document: a and I play saw sloth small soccer tortoise with
Query: shell tortoise
1 match
Compare document with query

Document 1:

a  and  I  play  saw  sloth  snail  soccer  tortoise

Document 2:

birds  blue  fly  in  sky  the

Document 3:

blue  found  jewelry  on  shell  soccer  tortoise

Query: Shell  tortoise

1 match

0 match

2 matches

IR

Vector space model

- Vectors are very, very long
  - We say it is a “high-dimensional” problem
  - # dimensions = size of vocabulary
- Very computationally intensive
- Any other problems?

IR

Variation: term weighting

Some words are more discriminating than others. E.g., “the” appears in just about every document

- Term frequency (TF)
  - E.g., The more times “Potter” is in the doc, the more likely the doc is about him
- Inverse document frequency (IDF)
  - The more documents there are containing a certain word, the less likely that word is important

IR

Use term frequency to improve search

Document 1:

a  and  I  play  saw  sloth  snail  soccer  tortoise

Document 2:

birds  blue  fly  in  sky  the

Document 3:

blue  found  jewelry  on  shell  soccer  tortoise

Query: shell  tortoise

Score: 1

Score: 0

Score: 3

IR

Preparing documents for vector space model

- Stemming
  - Potter’s = Potters = Potter
- Stop-words
  - Ignore words like “the”, “of”, …
- Use statistical properties of text
  - E.g., Data from Jamie Callan’s Characteristics of Text, 1997 (Sample of 19 million words)
Commonest fifty words

<table>
<thead>
<tr>
<th></th>
<th>f</th>
<th></th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>the</td>
<td>1,130,021</td>
<td>from</td>
<td>96,900</td>
</tr>
<tr>
<td>of</td>
<td>547,311</td>
<td>be</td>
<td>94,585</td>
</tr>
<tr>
<td>to</td>
<td>516,635</td>
<td>million</td>
<td>93,515</td>
</tr>
<tr>
<td>a</td>
<td>464,736</td>
<td>year</td>
<td>90,164</td>
</tr>
<tr>
<td>in</td>
<td>390,819</td>
<td>is</td>
<td>86,774</td>
</tr>
<tr>
<td>and</td>
<td>387,703</td>
<td>be</td>
<td>85,588</td>
</tr>
<tr>
<td>that</td>
<td>204,351</td>
<td>was</td>
<td>83,398</td>
</tr>
<tr>
<td>for</td>
<td>199,340</td>
<td>company</td>
<td>83,070</td>
</tr>
<tr>
<td>is</td>
<td>152,483</td>
<td>an</td>
<td>76,974</td>
</tr>
<tr>
<td>said</td>
<td>148,302</td>
<td>has</td>
<td>74,405</td>
</tr>
<tr>
<td>it</td>
<td>134,323</td>
<td>are</td>
<td>74,097</td>
</tr>
<tr>
<td>on</td>
<td>121,173</td>
<td>have</td>
<td>73,132</td>
</tr>
<tr>
<td>by</td>
<td>118,863</td>
<td>but</td>
<td>71,887</td>
</tr>
<tr>
<td>as</td>
<td>109,135</td>
<td>will</td>
<td>71,494</td>
</tr>
<tr>
<td>at</td>
<td>101,779</td>
<td>say</td>
<td>66,807</td>
</tr>
<tr>
<td>mr</td>
<td>101,679</td>
<td>new</td>
<td>64,456</td>
</tr>
<tr>
<td>with</td>
<td>101,210</td>
<td>share</td>
<td>63,925</td>
</tr>
</tbody>
</table>

Finding documents

- **Brute-force approach?**
  - Look through every single document every time you have a query
- **Efficient way?**
  - Make an index

Criteria for evaluating IR methods

- **Precision**
  - How many of the returned documents are relevant?
- **Recall**
  - How many of the relevant documents are returned?
  - Cannot be the sole criterion in evaluation
- **Fall-out**
  - How many of the non-relevant documents are returned?

  Can combine these criteria

Web Search

- Following the links to determine the link structure
- What are some issue and considerations?

What’s special about web search?

- Hyperlinks
- Size—scalability issues
- Dynamic content
- Untrained users
- Economic model (advertising)
“Crawling” the web

- Following the links to determine the link structure
- What are some issues and considerations?
  - Broken links, timeouts, … cause failures
  - Update frequency
  - Coverage, duplicate detection
  - Legal issues (owners don’t want their pages indexed)
  - Advertising links
  - Types of content
  - …

Web search through link analysis

- Find relevant webpages by analyzing the link structure, not by the content
- Most famous algorithm is PageRank

- There are other kinds of link analysis
  - E.g., citation analysis—count the number of references to individual research papers (CiteSeer)