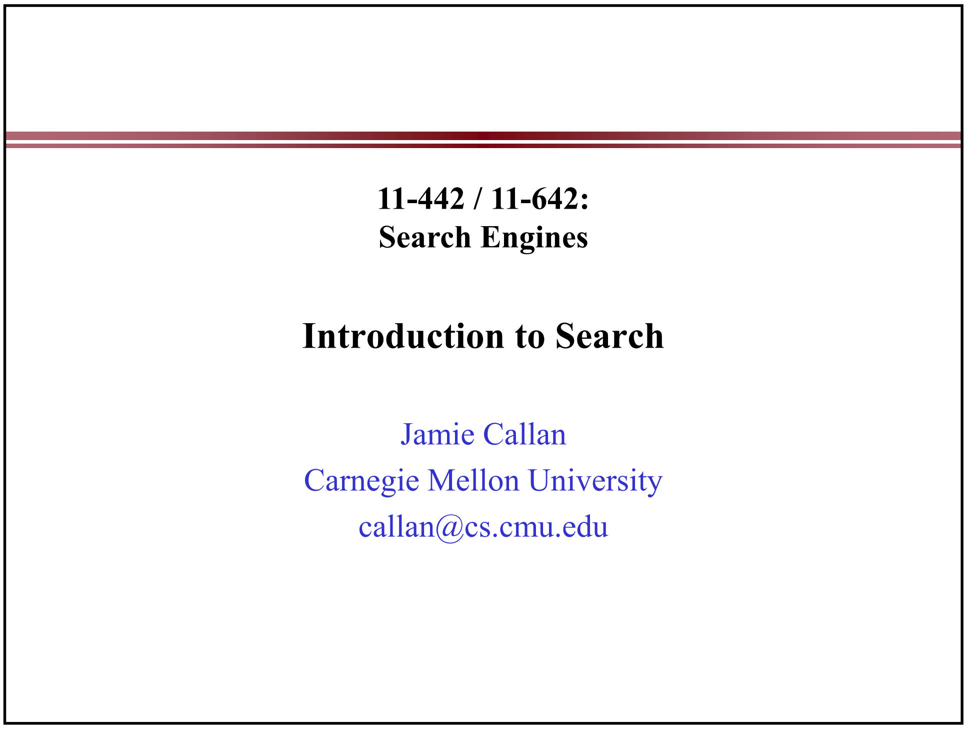
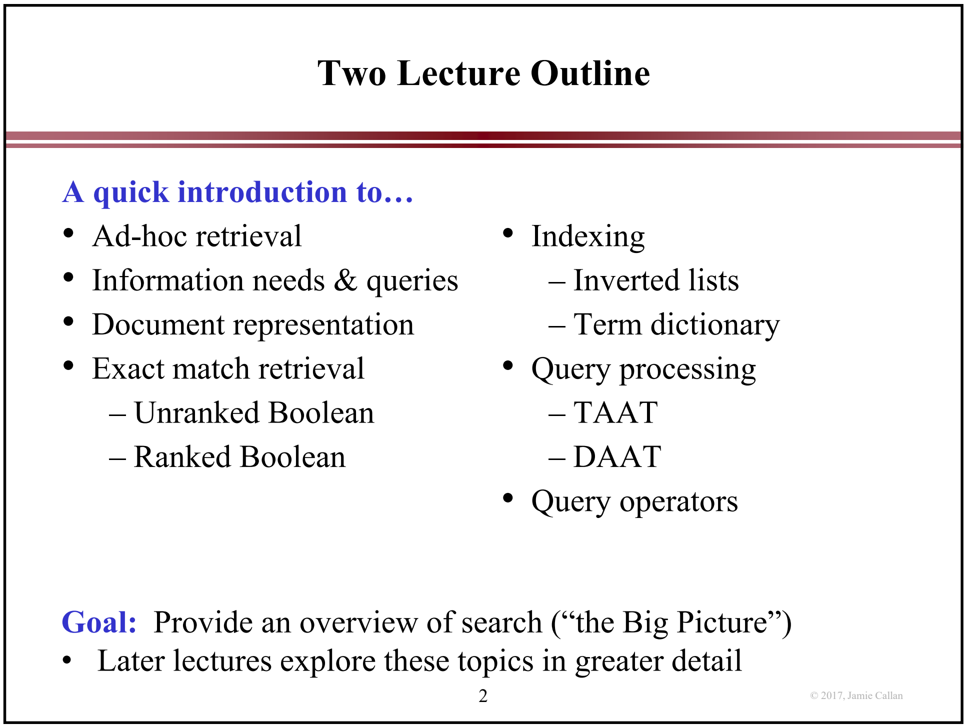


11-442 / 11-642:
Search Engines

Introduction to Search

Jamie Callan
Carnegie Mellon University
callan@cs.cmu.edu



Two Lecture Outline

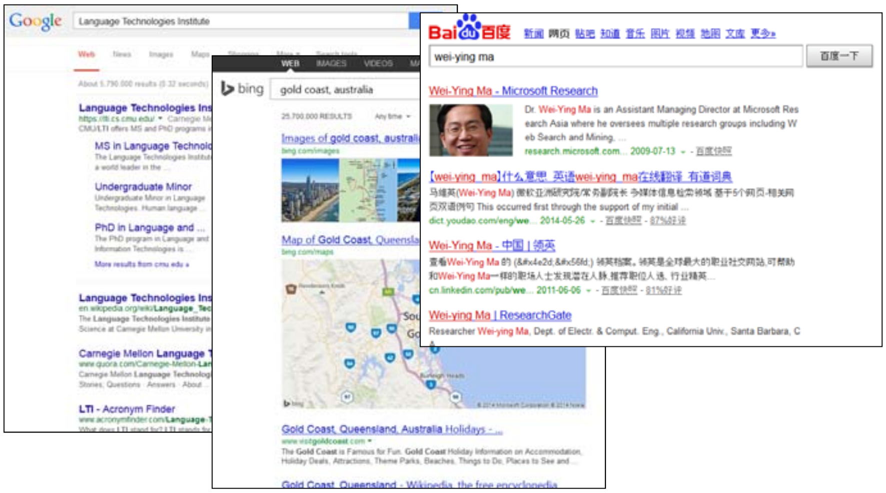
A quick introduction to…

- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexing
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search ("the Big Picture")

- Later lectures explore these topics in greater detail

Probably You are an Experienced Search Engine User



Google | Language Technologies Institute

Web News Images Maps Shopping More ▾ Search tools

About 5,790,000 results (0.32 seconds)

Language Technologies Ins
https://lti.cs.cmu.edu/ ▾ Carnegie M
CMU/LTI offers MS and PhD programs i

MS in Language Technolo
The Language Technologies Institute
a world leader in the ...

Undergraduate Minor
Undergraduate Minor in Language
Technologies. Human language ...

PhD in Language and ...
The PhD program in Language and
Information Technologies is ...

More results from cmu.edu »

Language Technologies Ins
en.wikipedia.org/wiki/Language_Tec
The Language Technologies Institute
Science at Carnegie Mellon University in

Carnegie Mellon Language T
www.quora.com/Carnegie-Mellon-Lan
Carnegie Mellon Language Technolog
Stories, Questions · Answers · About ...

LTI - Acronym Finder
www.acronymfinder.com/Language-T
What does LTI stand for? LTI stands for

WEB IMAGES VIDEOS MA

bing | gold coast, australia

25,700,000 RESULTS Any time ▾

Images of gold coast, australi
bing.com/images

Map of Gold Coast, Queensla
bing.com/maps

Gold Coast, Queensland, Australia Holidays - ...
www.visitgoldcoast.com ▾
The Gold Coast is Famous for Fun. Gold Coast Holiday Information on Accommodation, Holiday Deals, Attractions, Theme Parks, Beaches, Things to Do, Places to See and ...

Gold Coast, Queensland - Wikipedia, the free encyclopedia

Baidu百度 新闻 网页 贴吧 知道 音乐 图片 视频 地图 文库 更多»

wei-ying ma | 百度一下

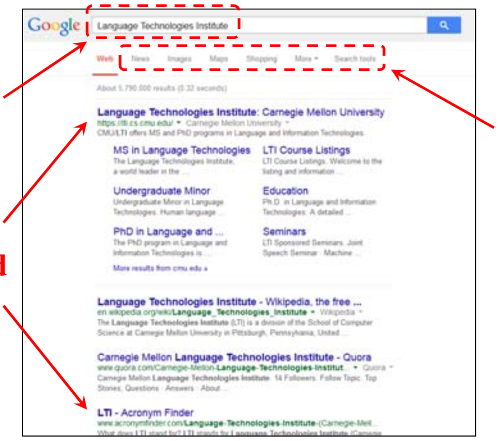
Wei-Ying Ma - Microsoft Research
Dr. Wei-Ying Ma is an Assistant Managing Director at Microsoft Research Asia where he oversees multiple research groups including Web Search and Mining. ...
research.microsoft.com... 2009-07-13 ▾ - 百度快照

【wei-ying_ma】什么意思_英语wei-ying_ma在线翻译_有道词典
马维英(Wei-Ying Ma) 微软亚洲研究院常务副院长 多媒体信息检索领域 基于5个网页-相关网页双语例句 This occurred first through the support of my initial ...
dict.youdao.com/eng/we... 2014-05-26 ▾ - 百度快照 - 87%好评

Wei-Ying Ma - 中国 | 领英
查看Wei-Ying Ma 的 (&#x4e2d;&#x56fd;) 领英档案。领英是全球最大的职业社交网站,可帮助和Wei-Ying Ma一样的职场人士发现潜在人脉,推荐职位人选、行业精英...
cn.linkedin.com/pub/we... 2011-06-06 ▾ - 百度快照 - 81%好评

Wei-ying Ma | ResearchGate
Researcher Wei-ying Ma, Dept. of Electr. & Comput. Eng., California Univ., Santa Barbara, C

How Does a Search Engine Developer or Researcher View the Search Process?



Google | Language Technologies Institute

Web News Images Maps Shopping More ▾ Search tools

About 5,790,000 results (0.32 seconds)

Language Technologies Institute: Carnegie Mellon University
https://lti.cs.cmu.edu/ ▾ Carnegie Mellon University ▾
CMU/LTI offers MS and PhD programs in Language and Information Technologies.

MS in Language Technologies
The Language Technologies Institute, a world leader in the ...

LTI Course Listings
LTI Course Listings. Welcome to the listing and information ...

Undergraduate Minor
Undergraduate Minor in Language Technologies. Human language ...

Education
Ph.D. in Language and Information Technologies. A detailed ...

PhD in Language and ...
The PhD program in Language and Information Technologies is ...

Seminars
LTI Sponsored Seminars. Joint Speech Seminar · Machine ...

More results from cmu.edu »

Language Technologies Institute - Wikipedia, the free ...
en.wikipedia.org/wiki/Language_Technologies_Institute ▾ Wikipedia ▾
The Language Technologies Institute (LTI) is a division of the School of Computer Science at Carnegie Mellon University in Pittsburgh, Pennsylvania, United ...

Carnegie Mellon Language Technologies Institute - Quora
www.quora.com/Carnegie-Mellon-Language-Technologies-Institut... ▾ Quora ▾
Carnegie Mellon Language Technologies Institute. 14 Followers. Follow Topic. Top Stories; Questions · Answers · About ...

LTI - Acronym Finder
www.acronymfinder.com/Language-Technologies-Institute-(Carnegie-Mell...
What does LTI stand for? LTI stands for Language Technologies Institute (Carnegie

A **query** that expresses an **information need**

Documents (unstructured information)

Vertical search engines

How Does a Search Engine Developer or Researcher View the Search Process?

A person starts with an information need

- The query is an approximate description of the information need

The person searches a corpus of unstructured information

- Documents

Goal: Find documents that satisfy the information need

- Search, retrieval

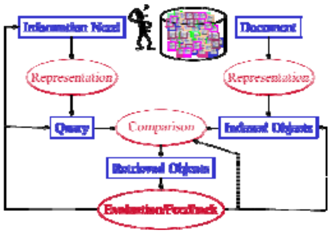
This lecture and the next present a simple end-to-end solution

- The "big picture"
- Later lectures go into more detail & more advanced material

Simple End-to-End Solutions

Requirements

- A way of representing information needs
- A way of representing document content
- A comparison or matching process

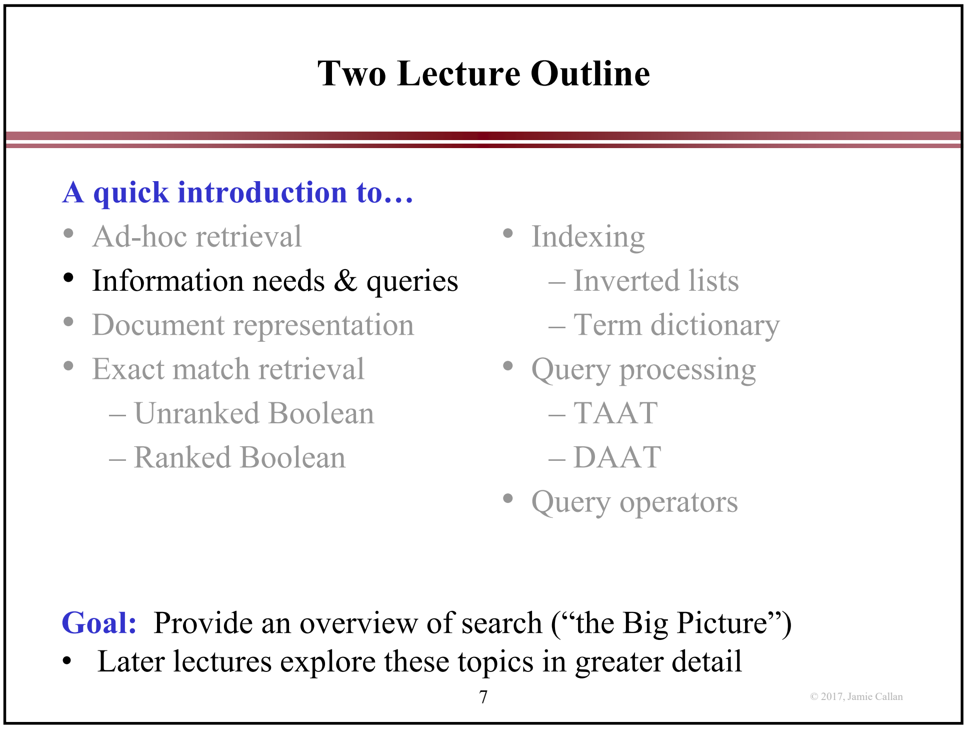


Initial solutions

- Boolean queries
- Exact-match retrieval models (unranked and ranked)

These solutions are primitive, but they are still used today

- Fast, easy to build, easy to understand



Two Lecture Outline

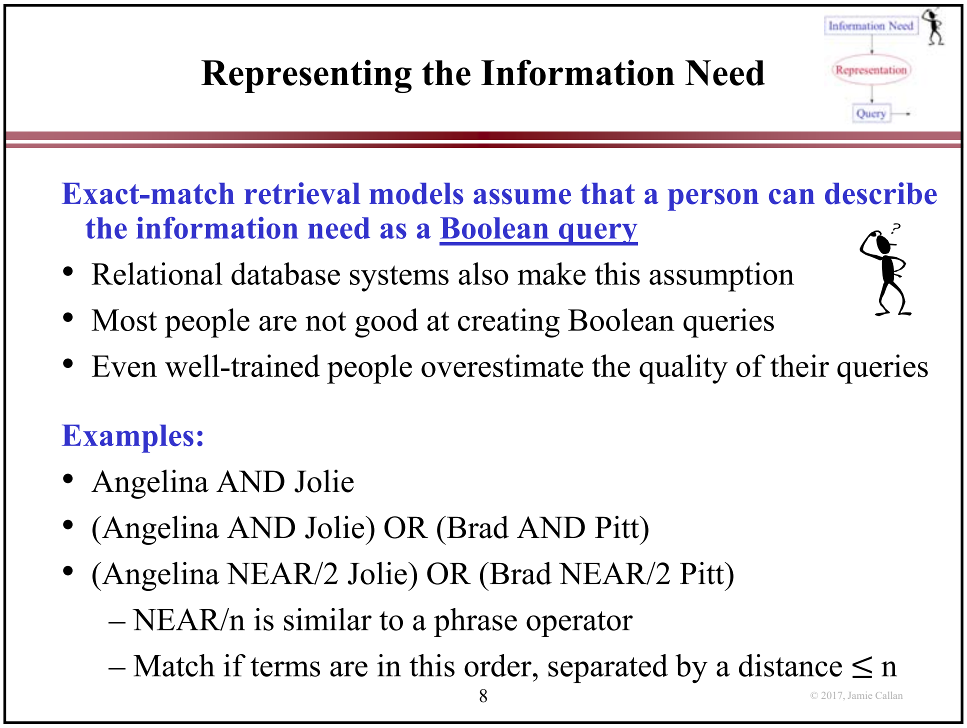
A quick introduction to…

- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexing
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search ("the Big Picture")

- Later lectures explore these topics in greater detail

7



Representing the Information Need

Information Need

Representation

Query

Exact-match retrieval models assume that a person can describe the information need as a Boolean query

- Relational database systems also make this assumption
- Most people are not good at creating Boolean queries
- Even well-trained people overestimate the quality of their queries

Examples:

- Angelina AND Jolie
- (Angelina AND Jolie) OR (Brad AND Pitt)
- (Angelina NEAR/2 Jolie) OR (Brad NEAR/2 Pitt)
  - NEAR/n is similar to a phrase operator
  - Match if terms are in this order, separated by a distance ≤ n

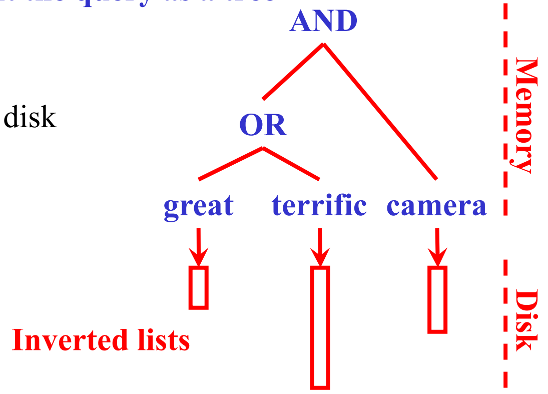
8

Query Trees

Query: (great OR terrific) AND camera

Search engines represent the query as a tree

- Leaves are index terms
- Leaves are pointers to inverted lists stored on disk



Two Lecture Outline

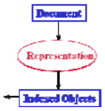
A quick introduction to…

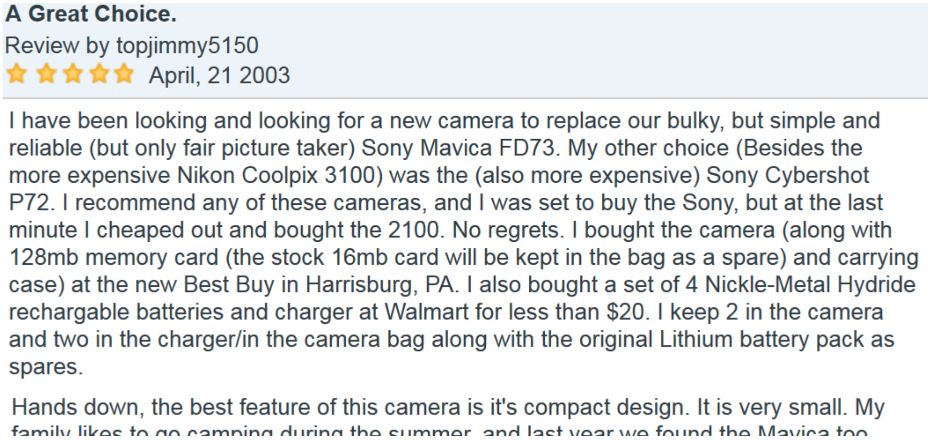
- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexing
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search (“the Big Picture”)

- Later lectures explore these topics in greater detail

Representing the Document: An Example Document





A Great Choice.

Review by topjimmy5150

★★★★★ April, 21 2003

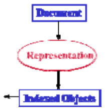
I have been looking and looking for a new camera to replace our bulky, but simple and reliable (but only fair picture taker) Sony Mavica FD73. My other choice (Besides the more expensive Nikon Coolpix 3100) was the (also more expensive) Sony Cybershot P72. I recommend any of these cameras, and I was set to buy the Sony, but at the last minute I cheaped out and bought the 2100. No regrets. I bought the camera (along with 128mb memory card (the stock 16mb card will be kept in the bag as a spare) and carrying case) at the new Best Buy in Harrisburg, PA. I also bought a set of 4 Nickle-Metal Hydride rechargable batteries and charger at Walmart for less than $20. I keep 2 in the camera and two in the charger/in the camera bag along with the original Lithium battery pack as spares.

Hands down, the best feature of this camera is it's compact design. It is very small. My family likes to go camping during the summer, and last year we found the Mavica too

. . . .

(topjimmy5150, Epinions.com)

Representing the Document



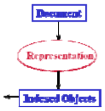
How should the contents of a document be represented?

Today, assume that we will use words from the document

- Free-text indexing: Use just some of the words
  - Developed first, but … which words?
- Full-text indexing: Use most or all of the words
  - Most search engines do this

Later lectures consider other possibilities

The Binary Full-Text Representation



Record which words occur in which documents

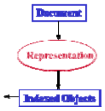
- Invented first
- Almost never used anymore, but a useful simplification

Corpus Vocabulary |V|

Corpus |C|

	a	abba	abend	ability	able	about	…	zooms
Doc_1	0	0	0	1	1	1	…	1
Doc_2	1	1	0	0	1	1	…	0
: : : :	:	: :	: :	: :	: :	: : :		: : :
Doc_n	0	0	1	1	0	1	…	0

The Binary Full-Text Representation (The Bag of Words)

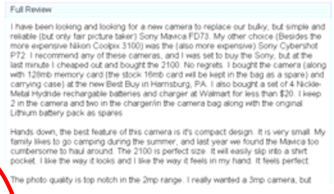


In the binary full-text representation, position and frequency are ignored

- The document is a "bag of words"
  - Or other features (covered later)

This is a simple representation of meaning

- But…surprisingly effective for search and other tasks (e.g., classification)

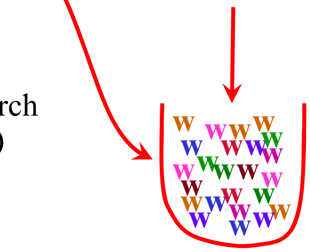


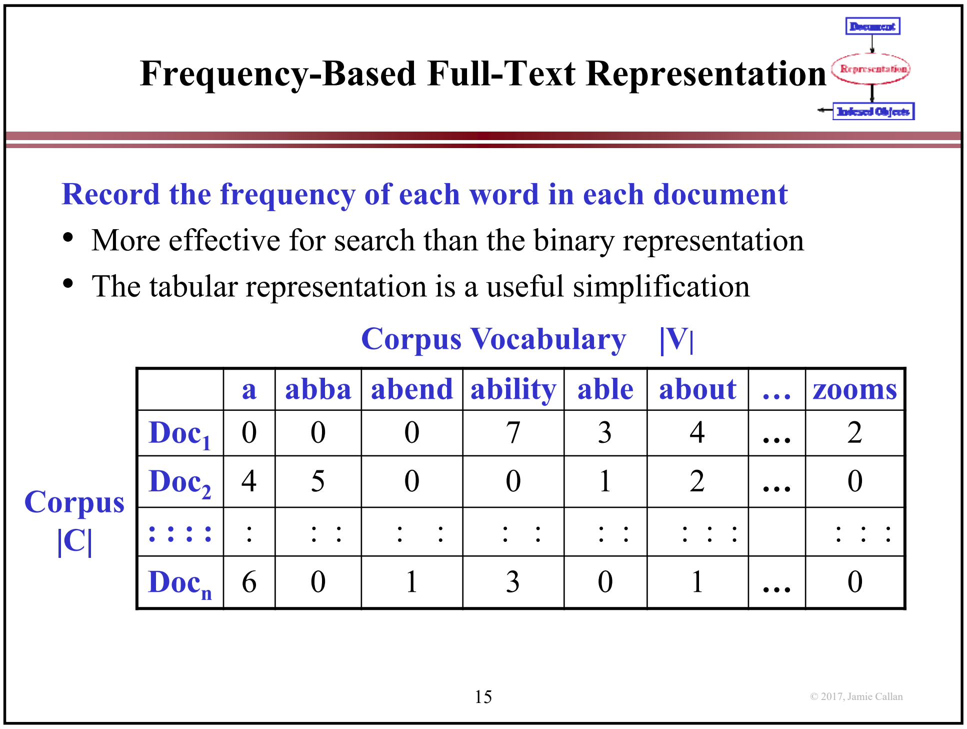
Full Review

I have been looking and looking for a new camera to replace our bulky, but simple and reliable (but only fair picture taker) Sony Mavica FD73. My other choice (Besides the more expensive Nikon Coolpix 3100) was the (also more expensive) Sony Cybershot P72. I recommend any of these cameras, and I was set to buy the Sony, but at the last minute I cheaped out and bought the 2100. No regrets. I bought the camera (along with 128mb memory card (the stock 16mb card will be kept in the bag as a spare) and carrying case) at the new Best Buy in Harrisburg, PA. I also bought a set of 4 Nickle-Metal Hydride rechargable batteries and charger at Walmart for less than $20. I keep 2 in the camera and two in the charger/in the camera bag along with the original Lithium battery pack as spares

Hands down, the best feature of this camera is it's compact design. It is very small. My family likes to go camping during the summer, and last year we found the Mavica too cumbersome to haul around. The 2100 is perfect size. It will easily slip into a shirt pocket. I like the way it looks and I like the way it feels in my hand. It feels perfect.

The photo quality is top notch in the 2mp range. I really wanted a 3mp camera, but





Frequency-Based Full-Text Representation

Document
Representation
Indexed Objects

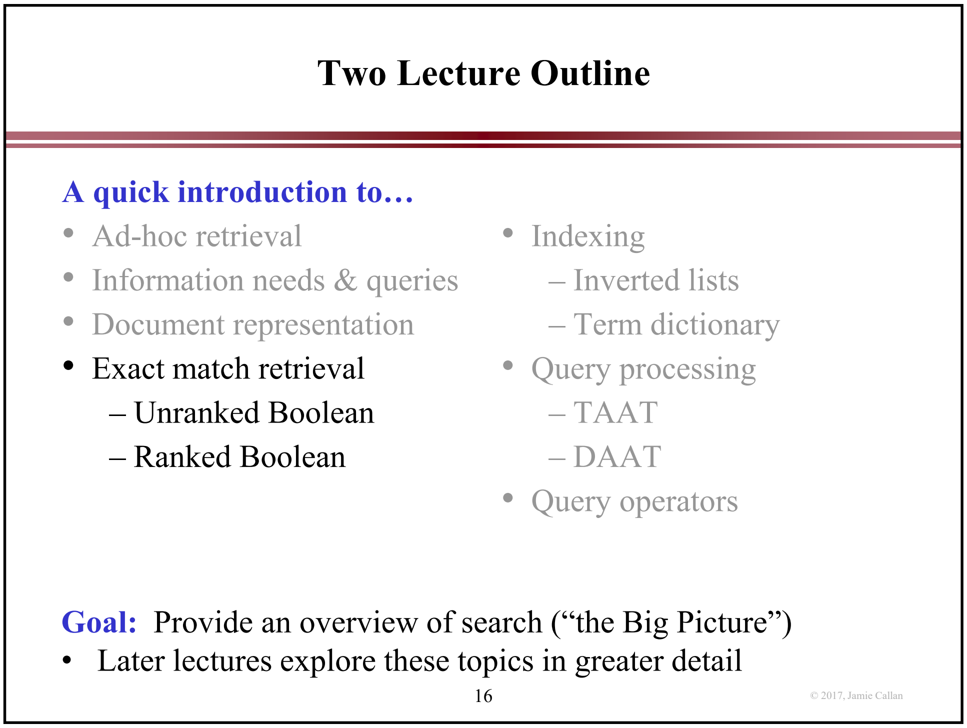
Record the frequency of each word in each document

- More effective for search than the binary representation
- The tabular representation is a useful simplification

Corpus Vocabulary |V|

Corpus |C|

	a	**abba**	**abend**	**ability**	**able**	**about**	**…**	**zooms**
Doc_1	0	0	0	7	3	4	**…**	2
Doc_2	4	5	0	0	1	2	**…**	0
: : : :	:	: :	: :	: :	: :	: : :		: : :
Doc_n	6	0	1	3	0	1	**…**	0



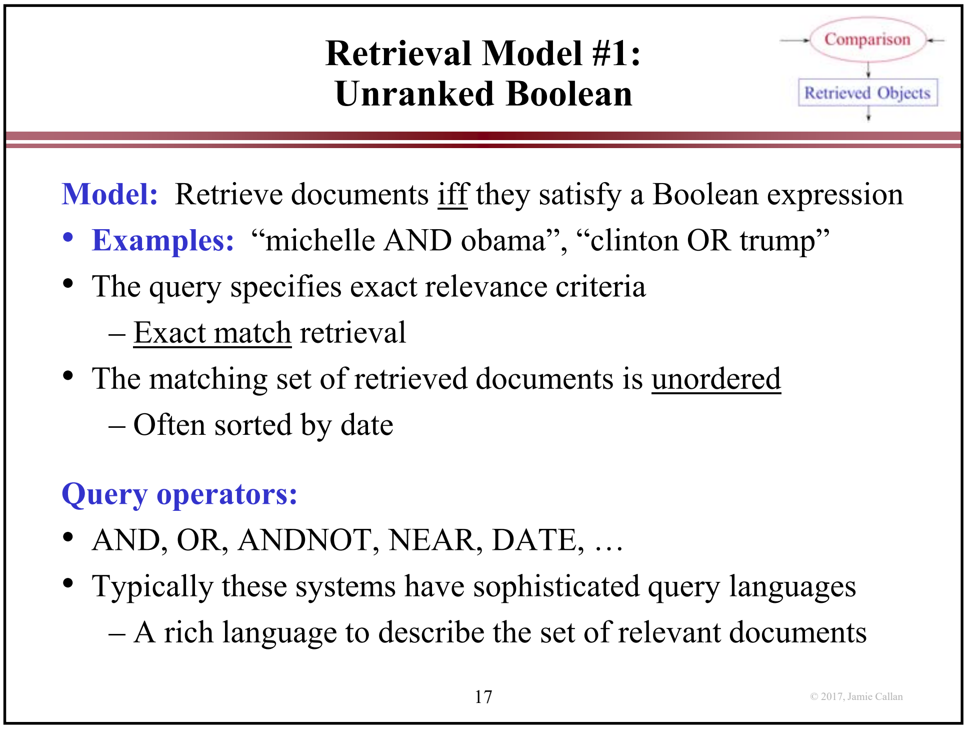
Two Lecture Outline

A quick introduction to…

- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexing
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search (“the Big Picture”)

- Later lectures explore these topics in greater detail



Comparison
Retrieved Objects

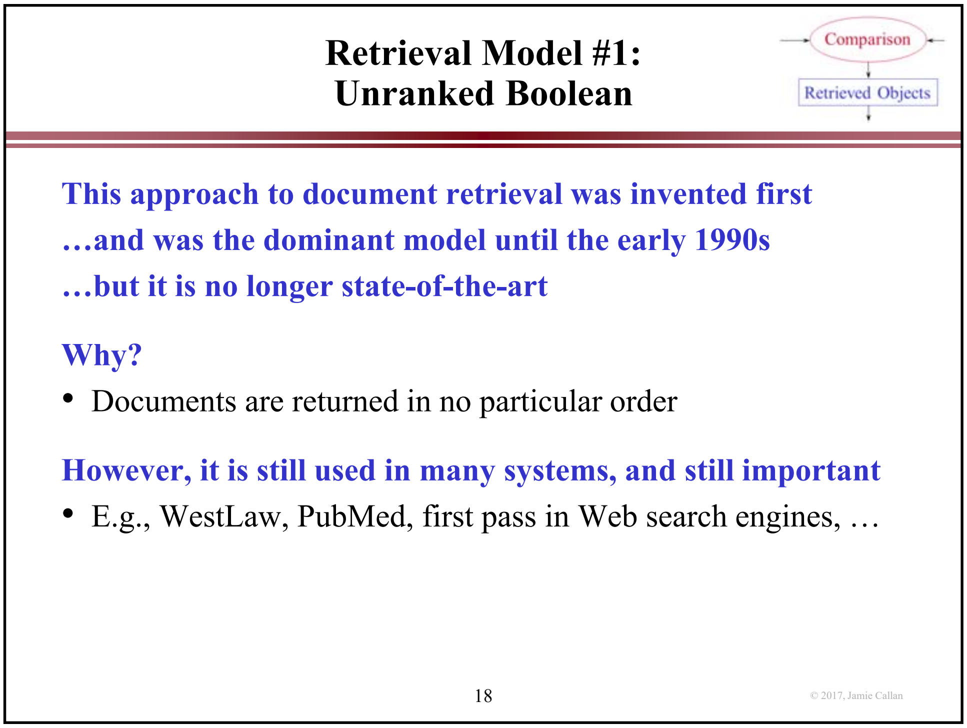
Retrieval Model #1: Unranked Boolean

Model: Retrieve documents iff they satisfy a Boolean expression

- **Examples:** “michelle AND obama”, “clinton OR trump”
- The query specifies exact relevance criteria
  – Exact match retrieval
- The matching set of retrieved documents is unordered
  – Often sorted by date

Query operators:

- AND, OR, ANDNOT, NEAR, DATE, …
- Typically these systems have sophisticated query languages
  – A rich language to describe the set of relevant documents



Comparison
Retrieved Objects

Retrieval Model #1: Unranked Boolean

This approach to document retrieval was invented first
…and was the dominant model until the early 1990s
…but it is no longer state-of-the-art

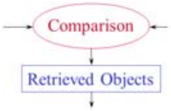
Why?

- Documents are returned in no particular order

However, it is still used in many systems, and still important

- E.g., WestLaw, PubMed, first pass in Web search engines, …

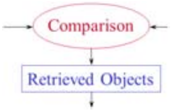
Retrieval Model #2: Ranked Boolean



Ranked Boolean addresses a flaw of Unranked Boolean

- Model & operators are the same as for Unranked Boolean
- But … matching documents are ranked by a document score
  - Unranked Boolean document scores are 0 and 1
  - Ranked Boolean document scores can be anything
    - » Typical Ranked Boolean systems have simple scores

Ranked Boolean: Calculating Scores

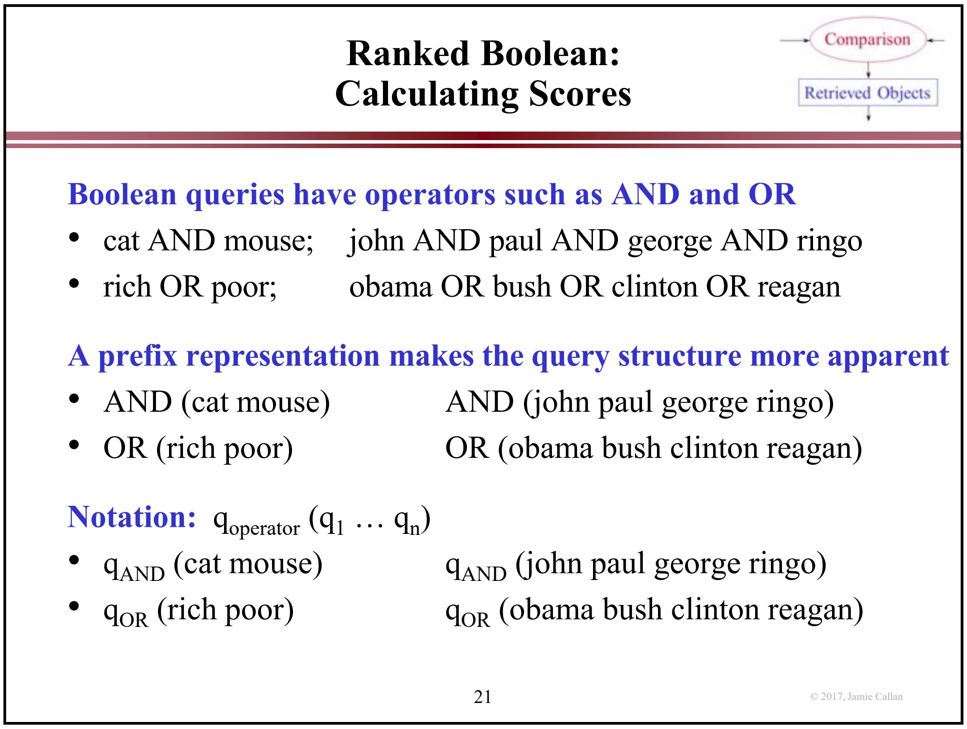


What is the score when a query term *t* occurs in document *d*?

- **$tf_{t,d}$:** The frequency of term *t* in document *d*
- **$tf_{t,d} \times idf_t = tf_{t,d} \times \log (N / df_t)$:** Penalize common terms
  - N: Number of documents in the corpus
  - df_t: The number of documents that contain term *t*

Both types of weights are used

- **$tf_{t,d}$:** Invented first, easy to implement, only considers the document
- **$tf_{t,d} \times idf_t$:** More effective
  - Rewards terms that are frequent in this document, but not frequent in the corpus



Ranked Boolean: Calculating Scores

Comparison
Retrieved Objects

Boolean queries have operators such as AND and OR

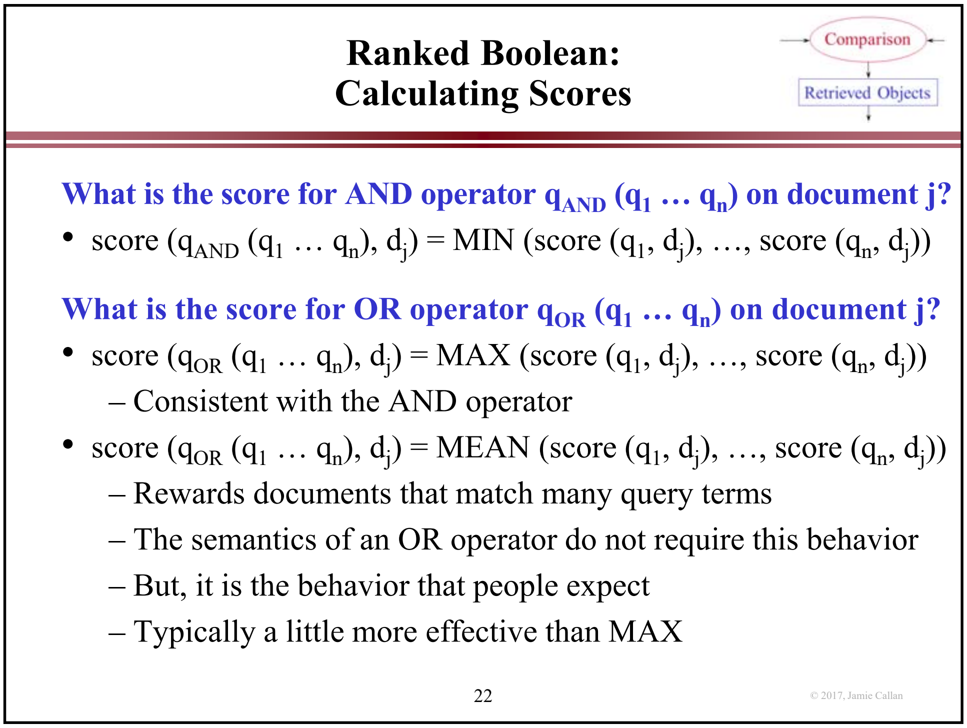
- cat AND mouse; john AND paul AND george AND ringo
- rich OR poor; obama OR bush OR clinton OR reagan

A prefix representation makes the query structure more apparent

- AND (cat mouse) AND (john paul george ringo)
- OR (rich poor) OR (obama bush clinton reagan)

Notation: $q_{operator}$ $(q_1 \ldots q_n)$

- q_{AND} (cat mouse) q_{AND} (john paul george ringo)
- q_{OR} (rich poor) q_{OR} (obama bush clinton reagan)



Ranked Boolean: Calculating Scores

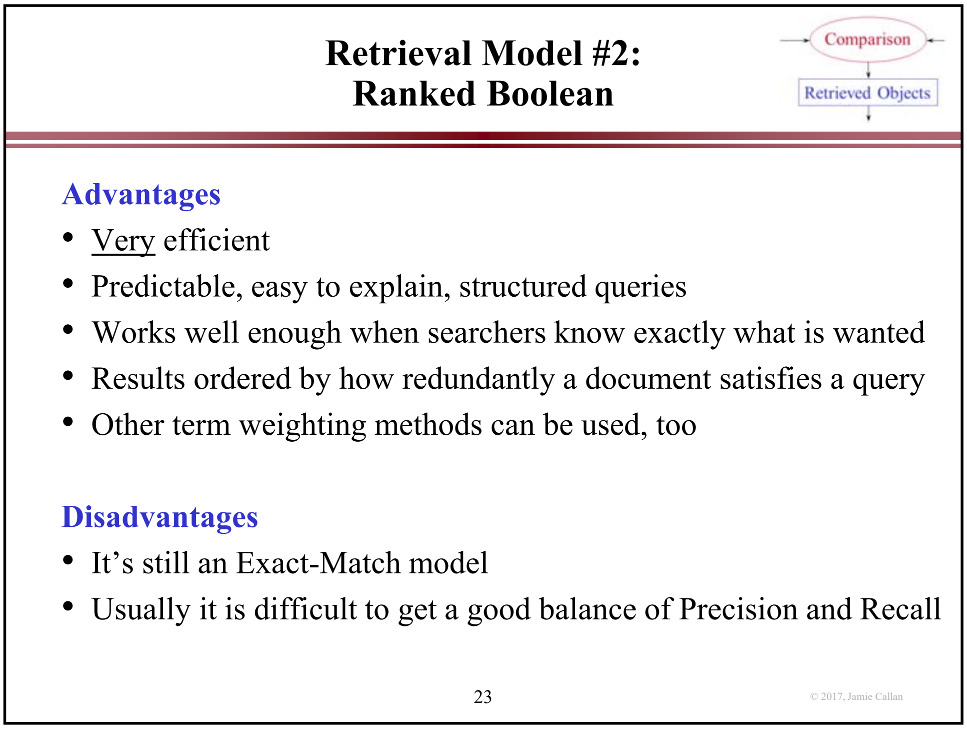
Comparison
Retrieved Objects

What is the score for AND operator q_{AND} $(q_1 \ldots q_n)$ on document j?

- score $(q_{AND}\ (q_1 \ldots q_n), d_j)$ = MIN (score (q_1, d_j), …, score (q_n, d_j))

What is the score for OR operator q_{OR} $(q_1 \ldots q_n)$ on document j?

- score $(q_{OR}\ (q_1 \ldots q_n), d_j)$ = MAX (score (q_1, d_j), …, score (q_n, d_j))
  - Consistent with the AND operator
- score $(q_{OR}\ (q_1 \ldots q_n), d_j)$ = MEAN (score (q_1, d_j), …, score (q_n, d_j))
  - Rewards documents that match many query terms
  - The semantics of an OR operator do not require this behavior
  - But, it is the behavior that people expect
  - Typically a little more effective than MAX


Comparison
Retrieved Objects

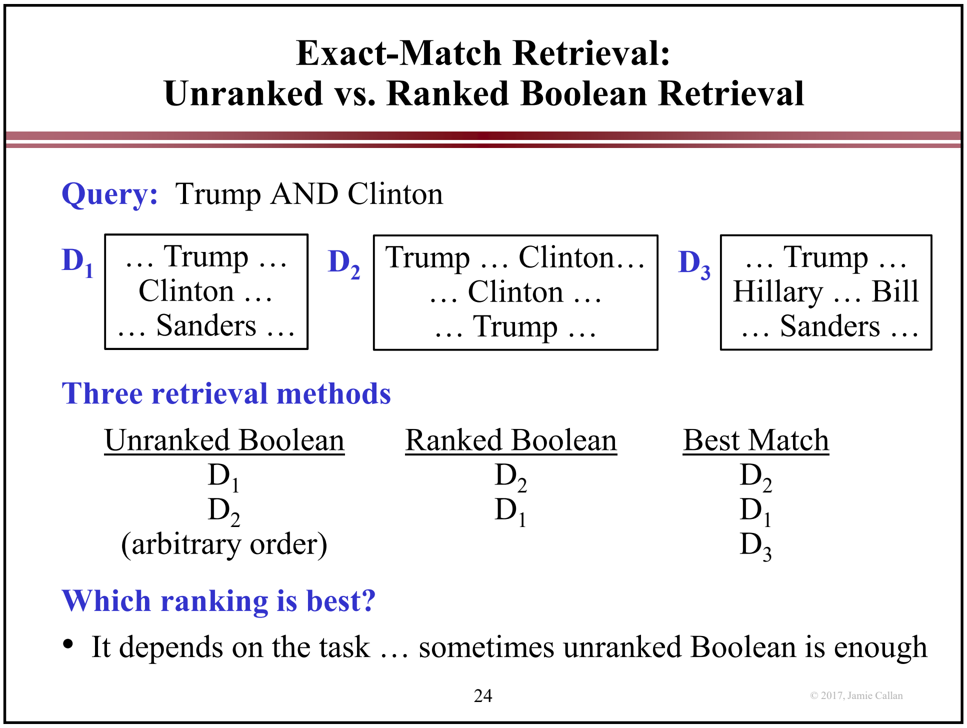
Retrieval Model #2:
Ranked Boolean

Advantages

- Very efficient
- Predictable, easy to explain, structured queries
- Works well enough when searchers know exactly what is wanted
- Results ordered by how redundantly a document satisfies a query
- Other term weighting methods can be used, too

Disadvantages

- It's still an Exact-Match model
- Usually it is difficult to get a good balance of Precision and Recall



Exact-Match Retrieval:
Unranked vs. Ranked Boolean Retrieval

Query: Trump AND Clinton

D_1 … Trump … Clinton … … Sanders …

D_2 Trump … Clinton… … Clinton … … Trump …

D_3 … Trump … Hillary … Bill … Sanders …

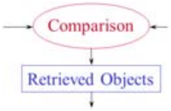
Three retrieval methods

Unranked Boolean	Ranked Boolean	Best Match
D_1	D_2	D_2
D_2	D_1	D_1
(arbitrary order)		D_3

Which ranking is best?

- It depends on the task … sometimes unranked Boolean is enough

Are Exact-Match Models Still Relevant?



Many people prefer exact-match Boolean models

- Professional searchers (e.g., librarians, paralegals)
- Some Web surfers (e.g., “Advanced Search” feature)
- What do they like? Control, predictability, understandability
- Preferred by 70% of WESTLAW searchers in a 2007 survey

  -- *James Allan, 2007*

Exact-match Boolean is a low-level part of Web search engines

- Massive corpus makes efficiency important
- Massive corpus makes partial matching less important

Two Lecture Outline

A quick introduction to…

- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexing
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search (“the Big Picture”)

- Later lectures explore these topics in greater detail

Data Structures for Index Terms

Search engines use <u>indexes</u> to process queries quickly

- Indexes are data structures that allow things to be found quickly
- Present a key, get back an object

What kinds of lookup does a search engine need?

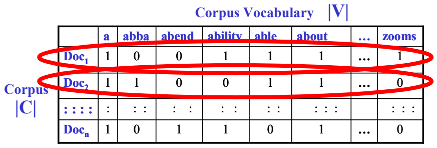
Term → DocId +	“apple” → doc_1 doc_{14} doc_{32} …
Attribute → Docid +	2014-08-06 → doc_8 doc_{19} doc_{63} …
Term → Corpus statistics	“apple” → df=3,731; ctf=8,839; …
Doc id → Attribute	doc_{19} → 2014-01-04
Doc id → Term +	doc_{19} → “listen” “live” “WYEP” …
Doc id → Document	doc_{19} →
…	

Data Structures for Index Terms

Task: Evaluate the query “ability AND about”

One could compare to each document (row)

- Invented first
- Rows are bit vectors
- Complexity is O(|C|×|V|)
- Is this good?



Corpus Vocabulary |V|

Corpus |C|

	a	abba	abend	ability	able	about	…	zooms
Doc_1	1	0	0	1	1	1	…	1
Doc_2	1	1	0	0	1	1	…	0
: : : :	:	: :	: :	: :	: :	: : :		: : :
Doc_n	1	0	1	1	0	1	…	0

Most terms are rare (occur in few documents)

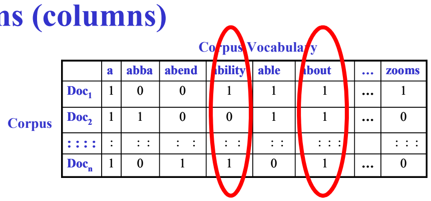
- The vocabulary V is huge
- Nearly all documents fail to match the query
- Most of the O(|C|×|V|) effort is wasted effort

Data Structures for Index Terms: Inverted Lists

Task: Evaluate the query “ability AND about”

One could compare to query terms (columns)

- Columns are bit vectors
- Columns are called inverted lists
- Complexity is O(|C|×|Q|)



Corpus Vocabulary

Corpus

	a	abba	abend	ability	able	about	…	zooms
Doc_1	1	0	0	1	1	1	…	1
Doc_2	1	1	0	0	1	1	…	0
: : : :	:	: :	: :	: :	: :	: : :		: : :
Doc_n	1	0	1	1	0	1	…	0

Inverted Lists

Most terms are rare (occur in few documents)

- Nearly all documents fail to match the query
- More efficient

… but still, most of the O(|C|×|Q|) effort is wasted effort

Really important data structure!

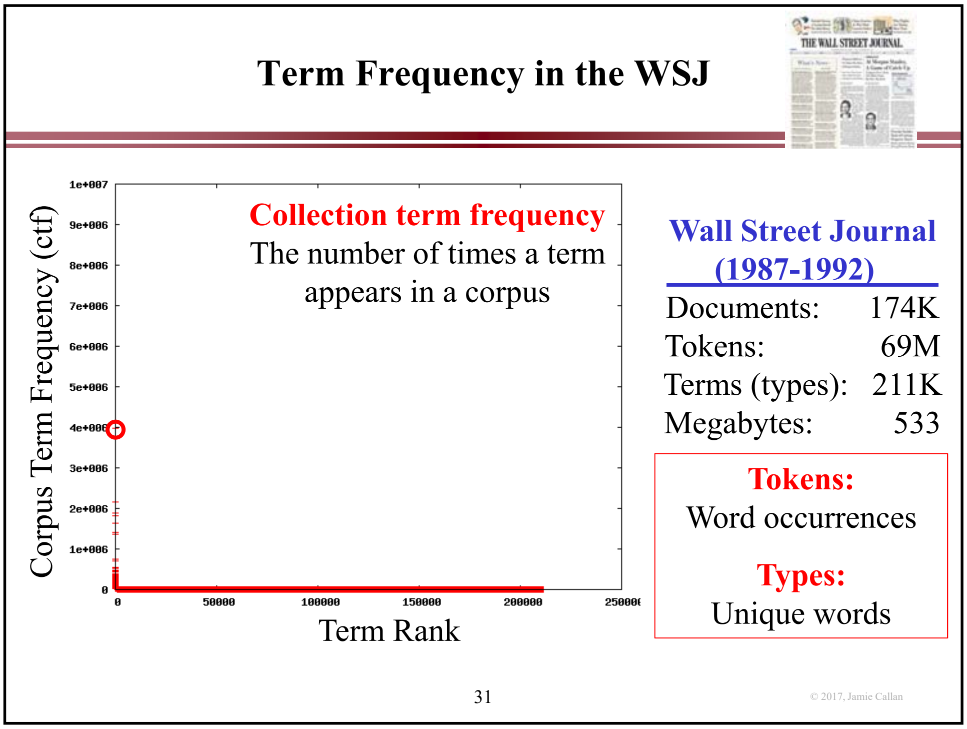
Are Fixed-Length Inverted Lists A Good Idea?

Fixed-length inverted lists were used in early search engines

- Simple to manage
- Bit-vector operations are fast and easy to parallelize

Fixed-length inverted lists are inefficient but … how inefficient?

- Length: |C| bits
  - – The number of documents in the corpus
- Number of bits set to 1 in a typical inverted list: $df_{typical}$
  - – The document frequency of the term with the median term frequency
- How does |C| compare to $df_{typical}$?



Term Frequency in the WSJ

Collection term frequency
The number of times a term appears in a corpus

Wall Street Journal (1987-1992)

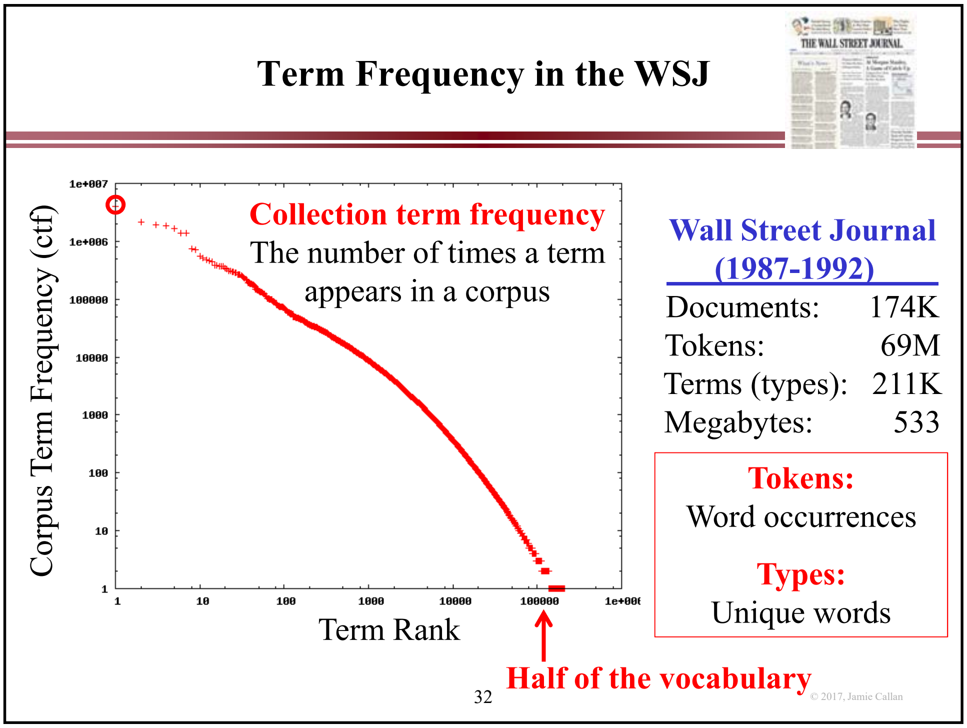
Documents:	174K
Tokens:	69M
Terms (types):	211K
Megabytes:	533

Tokens:
Word occurrences

Types:
Unique words

31

© 2017, Jamie Callan



Term Frequency in the WSJ

Collection term frequency
The number of times a term appears in a corpus

Wall Street Journal (1987-1992)

Documents:	174K
Tokens:	69M
Terms (types):	211K
Megabytes:	533

Tokens:
Word occurrences

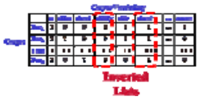
Types:
Unique words

Half of the vocabulary

32

© 2017, Jamie Callan

Are Fixed-Length Inverted Lists A Good Idea?



In ordinary text, |C| >> |$\text{df}_{\text{median}}$|

- In the Wall Street Journal, |C| = 174K and $\text{df}_{\text{median}}$ = 2
  – i.e., the median term appears in just <u>two</u> documents

So usually fixed-length inverted lists are <u>very</u> inefficient

- Often used for illustration
- Rarely used in real systems

Sparse Representation of Inverted Lists

Simple approach: Store ids of documents that contain the word

- **Full inverted list:** 10001100100…
  – The i$^{\text{th}}$ bit indicates whether the term occurs in the i$^{\text{th}}$ document
- **Sparse inverted list:** length=18, docids: 1, 5, 6, 9, …
  – The term occurs in 18 documents

Usually in this course inverted lists are represented as

- **df_t=18, docids 1, 5, 6, 9, …**
  df_t: document frequency (number of documents containing t)
  docids: document identifiers

You <u>must</u> know this data structure & more advanced variants

Inverted Lists

Different types of inverted lists support different capabilities

- **Binary inverted lists**
  - – Unranked retrieval
  - – AND, OR, AND-NOT, FIELD Boolean operators
- **Frequency inverted lists**
  - – Ranked retrieval
  - – The above operators plus SUM, SYNONYM
- **Positional inverted lists**
  - – Ranked retrieval
  - – The above operators plus positional operators
    - » NEAR/n, SENTENCE/n, PASSAGE/n, WINDOW/n

35

Different Types of Inverted Lists for the Term "Apple"

Binary

df:	4356
docid:	42
docid:	94
:	

Frequency

df:	4356
docid:	42
tf:	3
docid:	94
tf:	1
:	

Positional

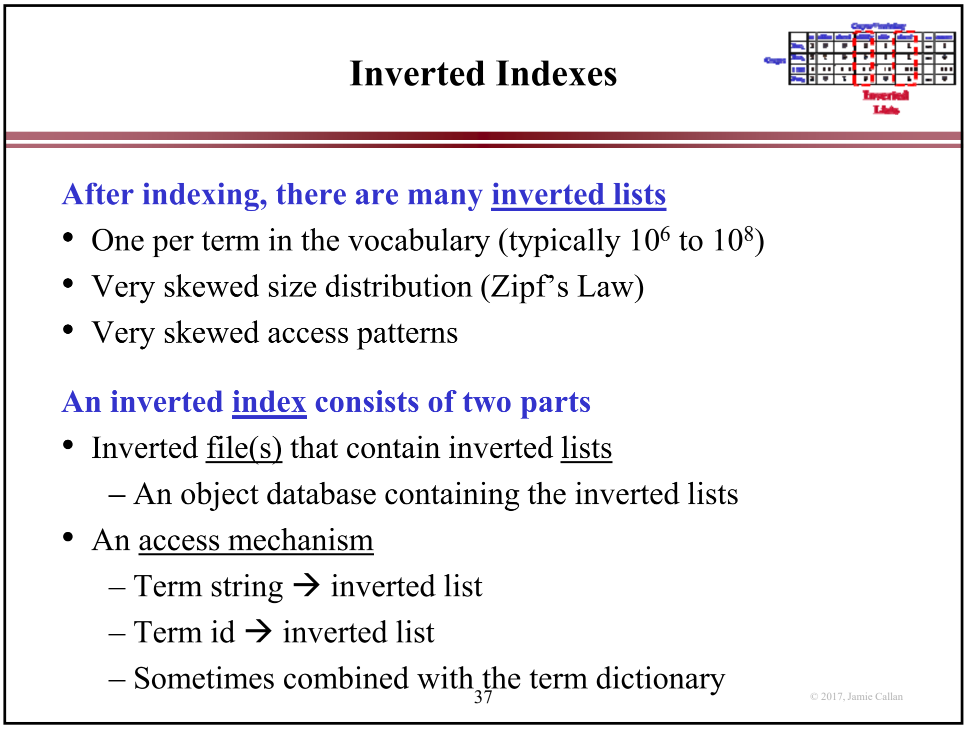
df:	4356
docid:	42
tf:	3
locs:	14
	83
	157
docid:	94
tf:	1
locs:	65
:	

Positional, With Embedded Document Info

df:	4356
docid:	42
doclen:	357
tf:	3
locs:	14
	83
	157
docid:	94
doclen:	172
tf:	1
locs:	65
:	

df: document frequency
tf: term frequency ($tf_{t,d}$)
doclen: document length (in terms)

36



Inverted Indexes

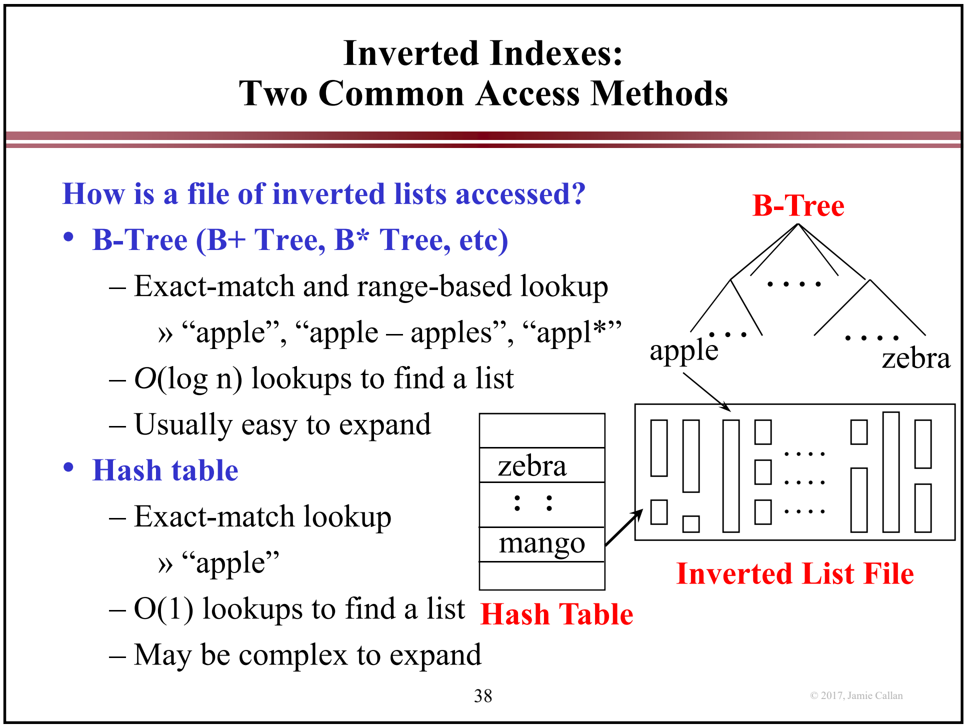
After indexing, there are many inverted lists

- One per term in the vocabulary (typically 10^6 to 10^8)
- Very skewed size distribution (Zipf's Law)
- Very skewed access patterns

An inverted index consists of two parts

- Inverted file(s) that contain inverted lists
  - – An object database containing the inverted lists
- An access mechanism
  - – Term string → inverted list
  - – Term id → inverted list
  - – Sometimes combined with the term dictionary

37

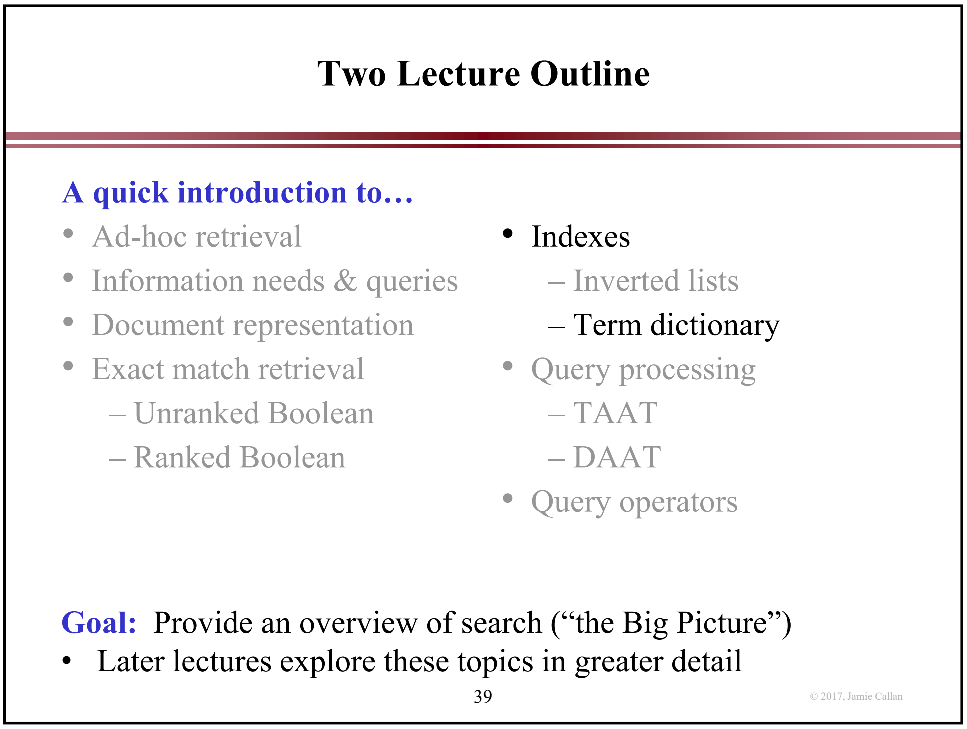


Inverted Indexes: Two Common Access Methods

How is a file of inverted lists accessed?

- **B-Tree (B+ Tree, B* Tree, etc)**
  - – Exact-match and range-based lookup
    - » "apple", "apple – apples", "appl*"
  - – *O*(log n) lookups to find a list
  - – Usually easy to expand
- **Hash table**
  - – Exact-match lookup
    - » "apple"
  - – O(1) lookups to find a list
  - – May be complex to expand

38



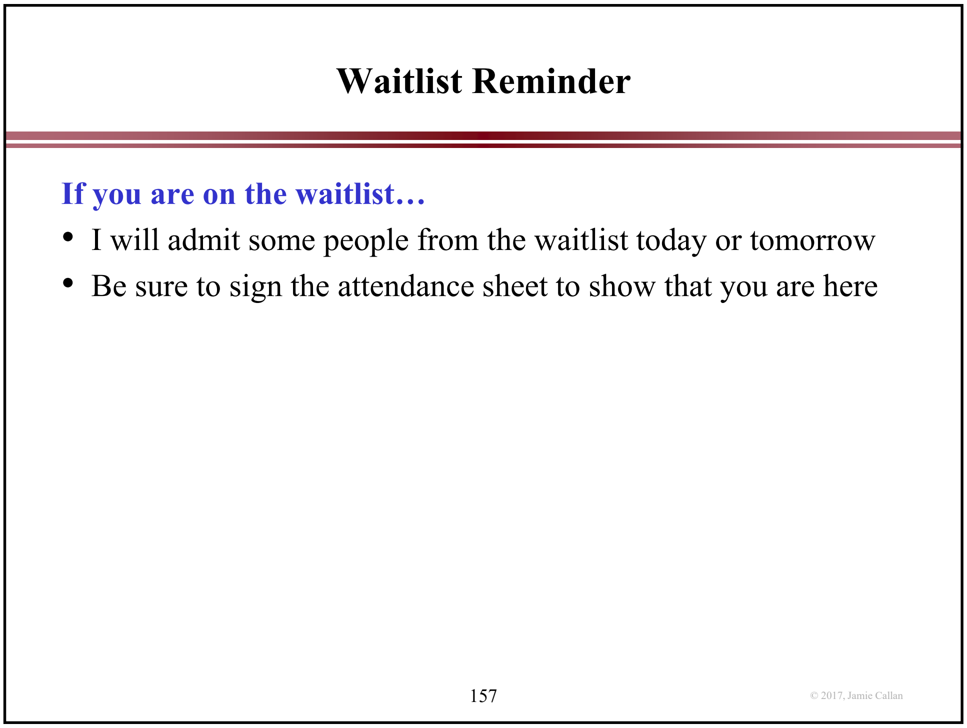
Two Lecture Outline

A quick introduction to…

- Ad-hoc retrieval
- Information needs & queries
- Document representation
- Exact match retrieval
  - Unranked Boolean
  - Ranked Boolean
- Indexes
  - Inverted lists
  - Term dictionary
- Query processing
  - TAAT
  - DAAT
- Query operators

Goal: Provide an overview of search ("the Big Picture")

- Later lectures explore these topics in greater detail



Waitlist Reminder

If you are on the waitlist…

- I will admit some people from the waitlist today or tomorrow
- Be sure to sign the attendance sheet to show that you are here