I 202: Information Organization & Retrieval Fall 2025

Class 23: IR evaluation, Boolean Queries, Search Ranking, Zifp's Law

Today's Outline

Relevance in Search

Ranking Evaluation

Boolean Search

TF-IDF, BM25, Zipfian Distribution

What’s the right answer for this query?

How to Rank Search Results?

- Assume you have
  - 300M web pages or
  - 30M images or
  - 300,000 resumes
- All of which match a query
  - “best car”
  - “beautiful”
  - “software engineer”
- How do you decide which is most relevant? Second most?

What Is Relevance (for search ranking)?

And how can we measure it?

Relevance: A Key Topic in IR

In what ways can a document be relevant to a query?

- Answer precise question precisely.
  - Who is buried in grant's tomb? Grant.
- Partially answer question.
  - Where is Berkeley? Near Oakland.
- Suggest a source for more information.
  - What is lymphodema? Look in this Medical Dictionary.
- Give background information.
- Suggest an expert person
- Others ...

A Definition of Relevance

Relevance is the	measure degree dimension estimate appraisal relation	**of a**	correspondence utility connection satisfaction fit bearing matching
existing between a	document article textual form reference information provided fact	**and a**	query request information used point of view information need statement
as determined by	person judge user requester Information specialist		

Saracevic. (1975) Relevance: A Review of and a Framework for Thinking on the Notion JASIST, 26(6), 321-343;

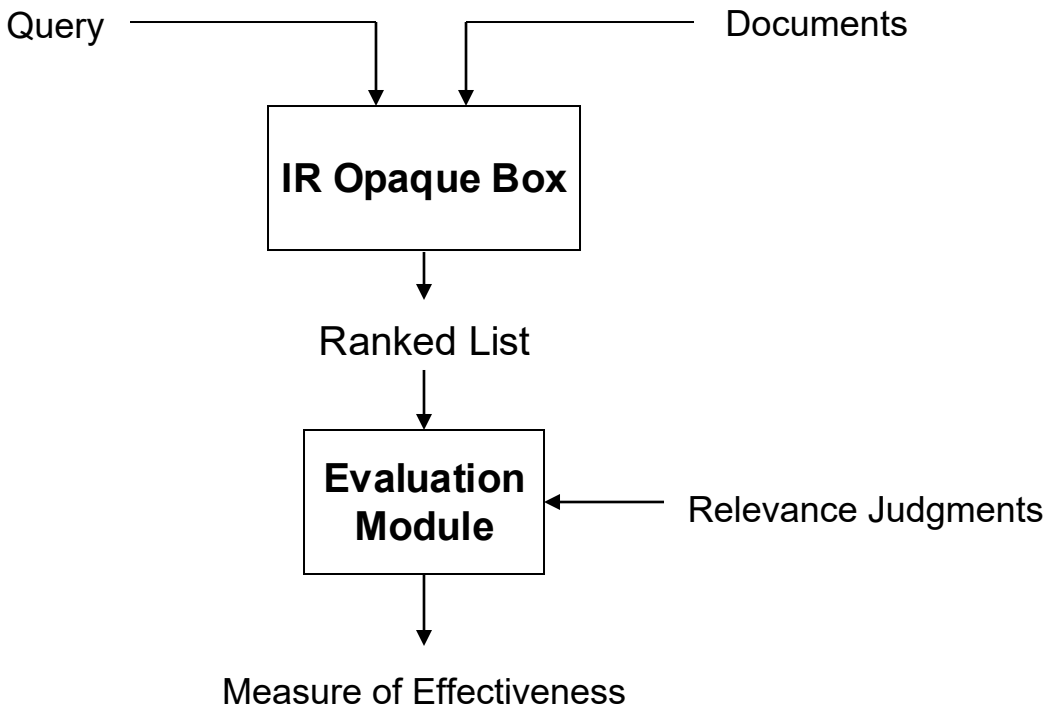
Types of IR Evaluation Strategies

- **System-centered studies**
  - Given documents, queries, and relevance judgments
  - Try several variations of the system
  - Measure which system returns the “best” hit list
- **User-centered studies**
  - Given several users, and at least two retrieval systems
  - Have each user try the same task on both systems
  - Measure which system works the “best”

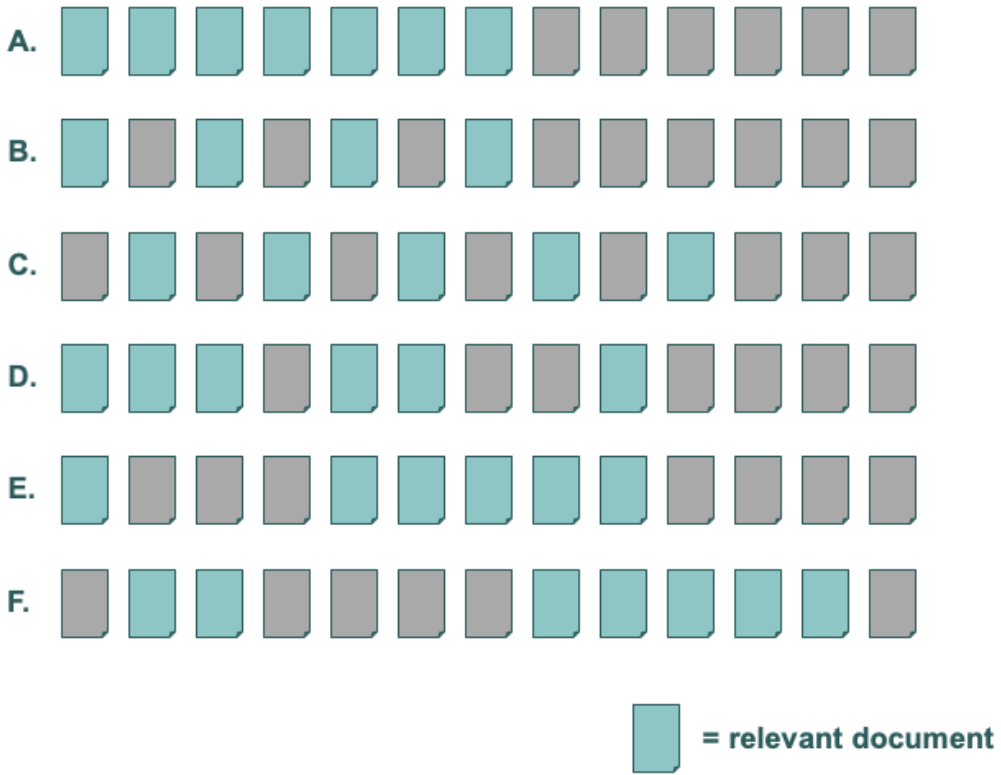
What to Evaluate?

- Effectiveness
  - How “good” are the items that are returned?
  - How “accurate” is the ranking? (precision & recall)
- Efficiency
  - Retrieval time, indexing time, index size
- Usability
  - Learnability, frustration
  - Novice vs. expert users

Automatic Evaluation Model



Assume you have a Query and 6 ways to Rank it. Which of These is Best?



Precision and Recall for Assessing Ranking

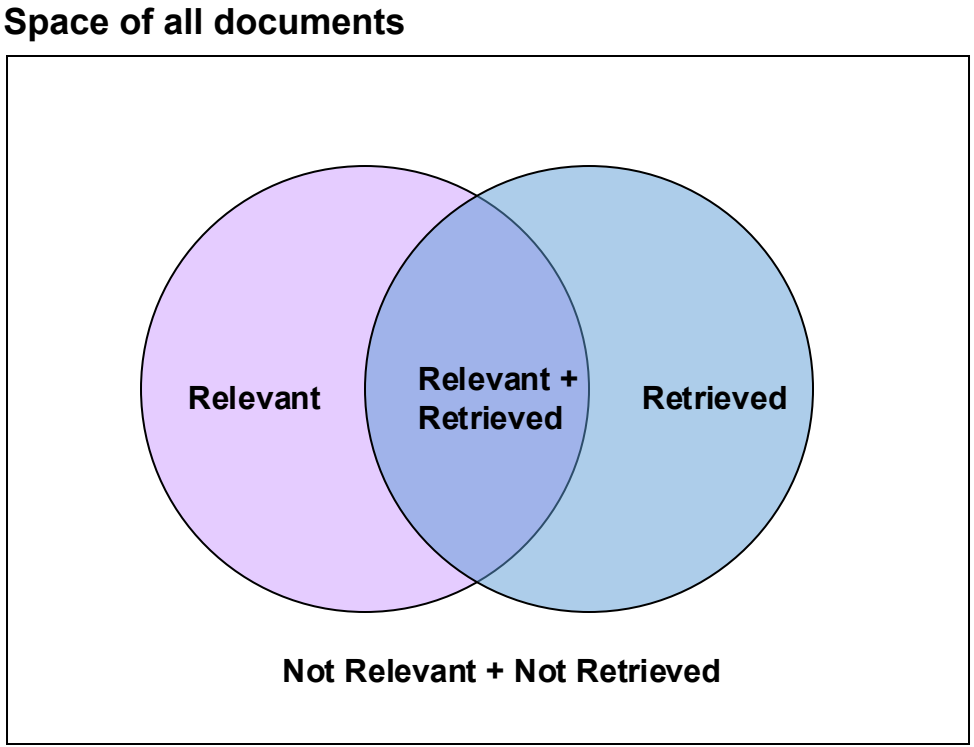
- Precision
  - proportion of retrieved material that is relevant
- Recall
  - proportion of relevant material that is retrieved
- F-Measure
  - Balances between the two

Viewing As a Set

Precision:
Relevant that are retrieved /
Retrieved

(purple /
blue + purple)

"Are all the retrieved items relevant?"



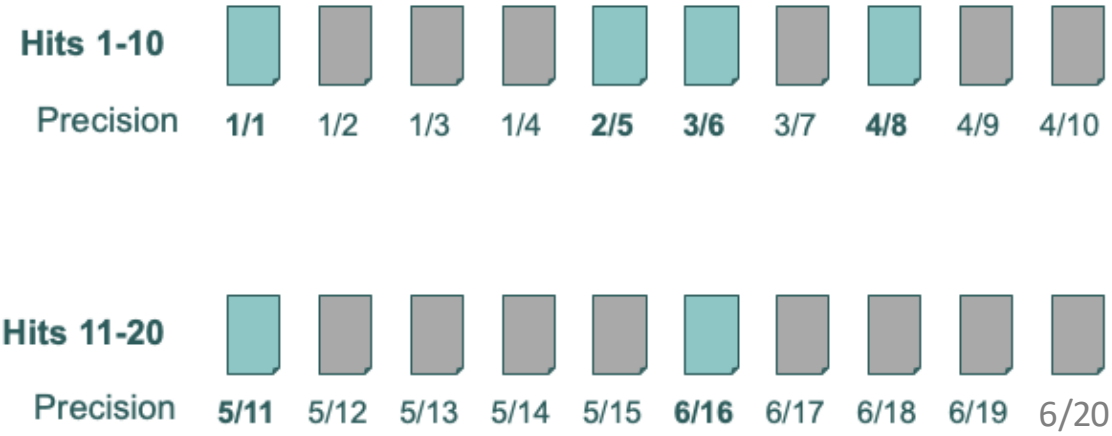
Recall:
Relevant that are retrieved /
Relevant

(purple /
lavender + purple)

"Are the all the relevant items retrieved?"

Measuring Precision

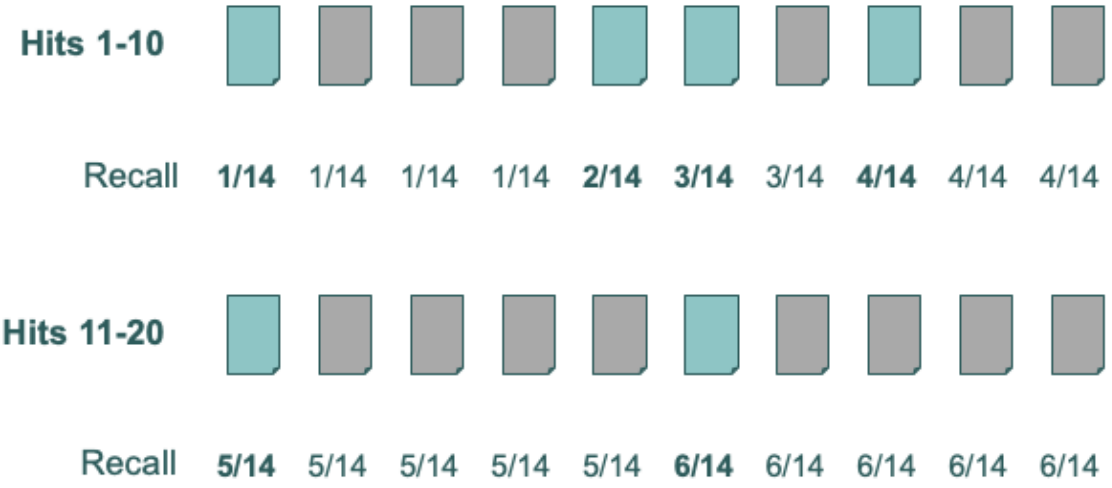
Assume there are a total of 14 relevant documents



Relevant that are retrieved / # Retrieved

Measuring Recall

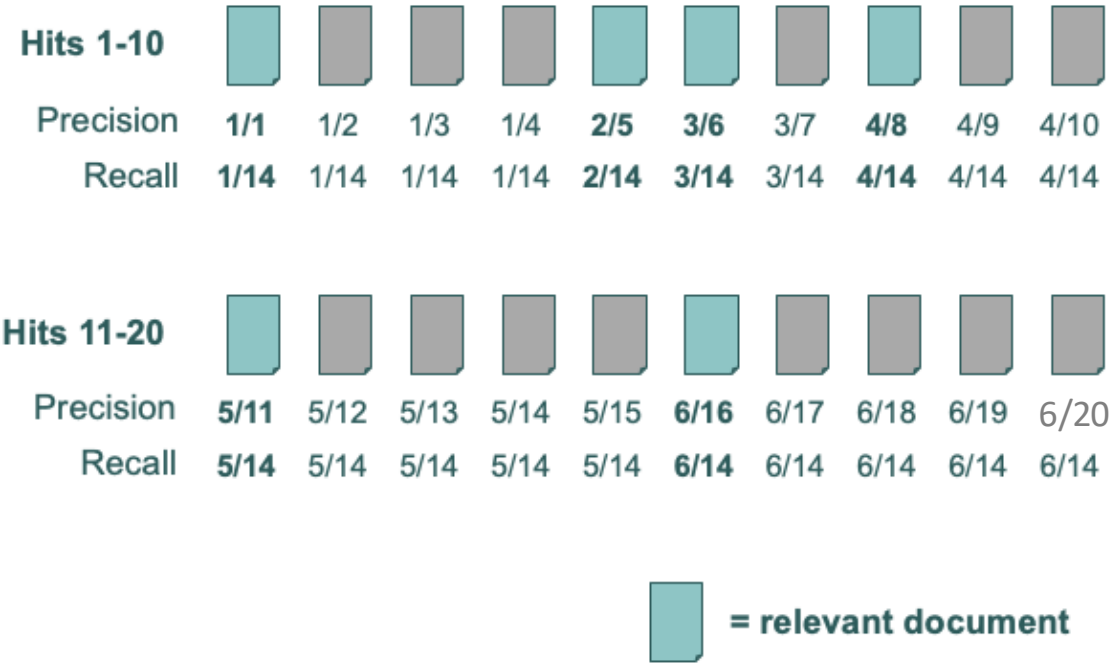
Assume there are a total of 14 relevant documents



Relevant that are retrieved / # Relevant

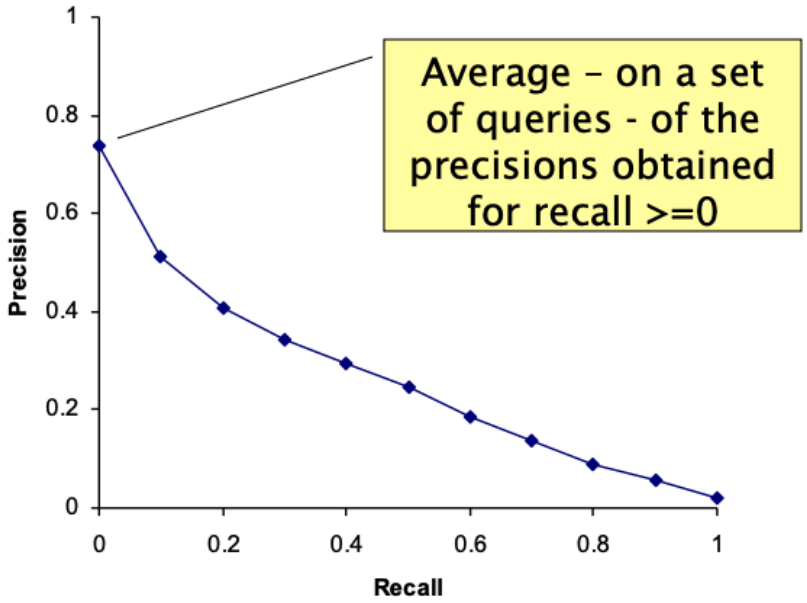
Measuring Precision and Recall

Assume there are a total of 14 relevant documents



Typical Precision-Recall Tradeoff

- SabIR/Cornell 8A1 11pt precision from TREC 8 (1999)



Slide from Manning & Raghavan via Ricci

F-Measure: Balancing Precision and Recall

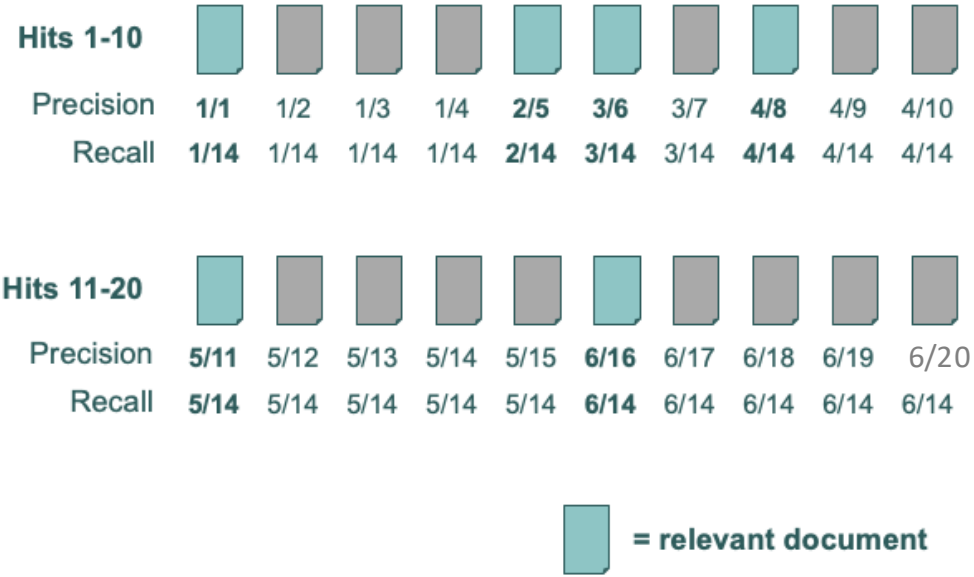
$$F = \frac{\left(\beta^2 + 1\right)PR}{\beta^2 P + R}$$

- Harmonic mean of recall and precision
- Beta controls relative importance of precision and recall
  - Beta = 1: precision and recall equally important
  - Beta = 5: recall five times more important than precision

FYI: Harmonic mean is the reciprocal of the arithmetic mean of the reciprocals of the given set of observations

Precision and Recall and F-measure

Assume there are a total of 14 relevant documents



Position	Precision	Recall	F (Beta=1)	F (Beta=5)
8	0.5	0.286	0.364	0.289
16	0.375	0.429	0.400	0.427
20	0.3	0.429	0.353	0.426

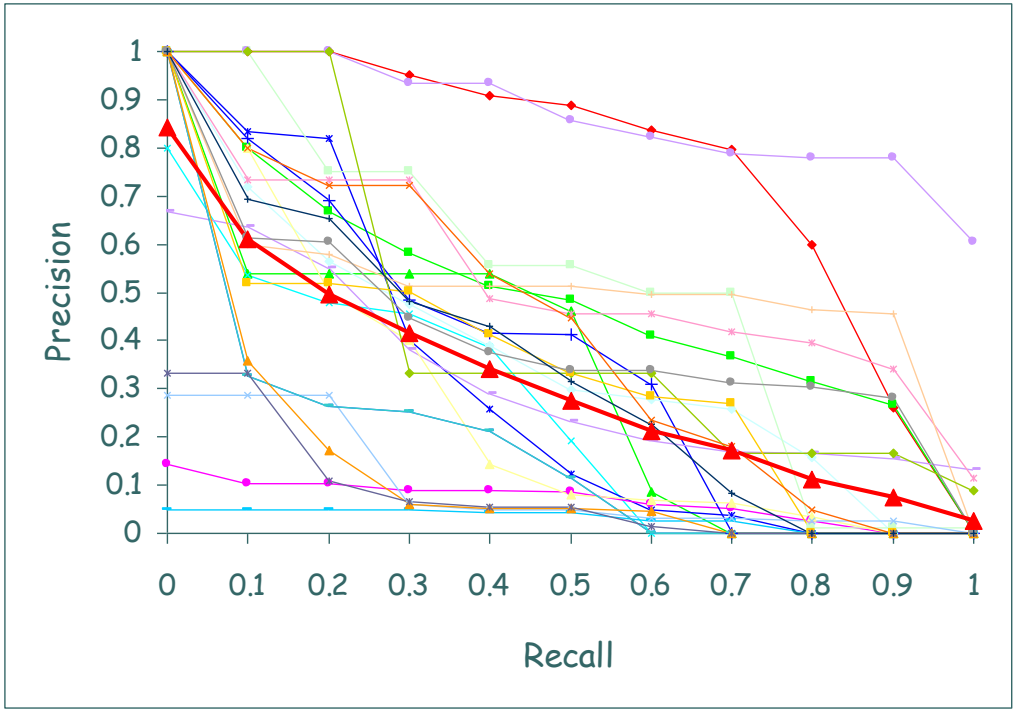
Beta controls relative importance:

Beta = 1, P and R equally important

Beta = 5, R five times more important than P

$$F = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

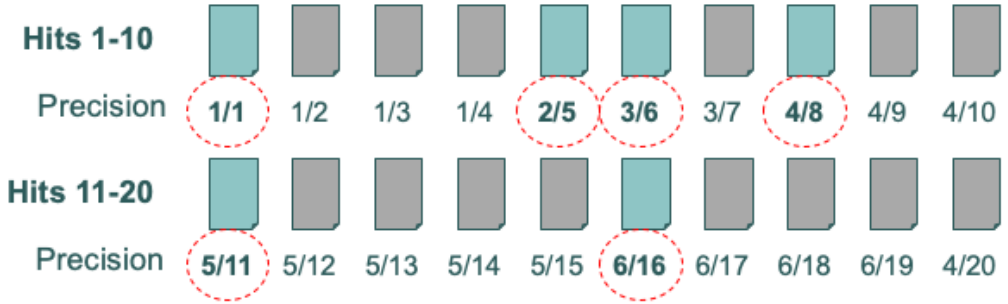
How do we compare Different Search Systems' Precision-Recall Curves?



Adapted from a presentation by Ellen Voorhees at the University of Maryland, March 29, 1999

Mean Average Precision (MAP)

- Average of precision at each retrieved relevant document
- Relevant documents not retrieved contribute zero to score



Assume total of 14 relevant documents: 8 relevant documents not retrieved contribute eight zeros

MAP = .2307

Compute precision at each position
Average the scores for only where there are relevant document (the circled values)

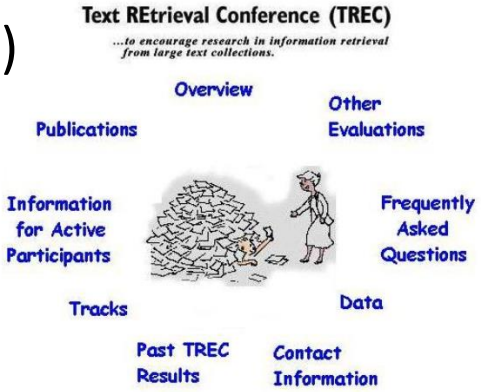
= relevant document

Single-Valued Versions of P and R

- Precision at a fixed number of documents
  - *Precision at 10 docs is often useful for Web search*
- R-precision
  - *Precision at r documents, where r is the total number of relevant documents*
- Expected search length
  - *Average rank of the first relevant document*

BUILDING TEST COLLECTIONS

- Where do test collections come from?
  - Someone goes out and builds them (expensive)
  - Or as the byproduct of some project
- TREC = Text REtrieval Conferences
  - Sponsored by NIST
  - Series of annual evaluations, started in 1992
  - Organized into “tracks”
  - Larger tracks may draw a few dozen participants



See proceedings online at http://trec.nist.gov/

Ad Hoc Topics

(as opposed to monitoring type information needs)

In TREC, a statement of information need is called a *topic*

Number: 312

Title: Hydroponics

Description: Document will discuss the science of growing plants in water or some substance other than soil.

Narrative: A relevant document will contain specific information on the necessary nutrients, experiments, types of substrates, and/or any other pertinent facts related to the science of hydroponics. Related information includes, but is not limited to, the history of hydroponics, advantages over standard soil agricultural practices, or the approach of suspending roots in a humid enclosure and spraying them periodically with a nutrient solution to promote plant growth.

Sample TREC topic (number 312) from the TREC-6 ad hoc track (Voorhees and Harman, 2000)

How to Obtain Enough Relevance Judgments?

- Exhaustive assessment is usually impractical
  - TREC has 50 queries
  - Collection has >1 million documents
- Random sampling won't work
  - If relevant docs are rare, none may be found!
- IR systems can help focus the sample (pooling method)
  - Each system finds some relevant documents
  - Different systems find different relevant documents
  - Together, enough systems will find most of them
  - Leverages cooperative evaluations

Pooling Methodology For Obtaining Relevance Judgements

- Systems submit top 1000 documents per topic
- Top 100 documents from each are judged
  - *Single pool, duplicates removed, arbitrary order*
  - *Judged by the person who developed the topic*
- Treat unevaluated documents as not relevant
- Compute MAP down to 1000 documents
- To make pooling work:
  - *Systems must do reasonably well*
  - *Systems must not all "do the same thing"*
- Gather topics and relevance judgments to create a reusable test collection

Lessons From TREC

- Absolute scores are not trustworthy
  - Who's doing the relevance judgment?
  - How complete are the judgments?
- Relative rankings are stable
  - Comparative conclusions are most valuable
- Cooperative evaluations produce reliable test collections
- Evaluation technology is predictive

Recap: Automatic Evaluation

- Test collections focus on IR ranking
- Automatic evaluation is one shot
  - *Ignores the richness of human interaction*
- Evaluation measures focus on *one* notion of performance
  - *But users care about other things*
- Goal is to compare systems
  - *Values may vary, but relative differences are stable*

How Search Engines Work

Three main parts:

i. Gather the contents of all web pages (using a program called a **crawler** or **spider**)

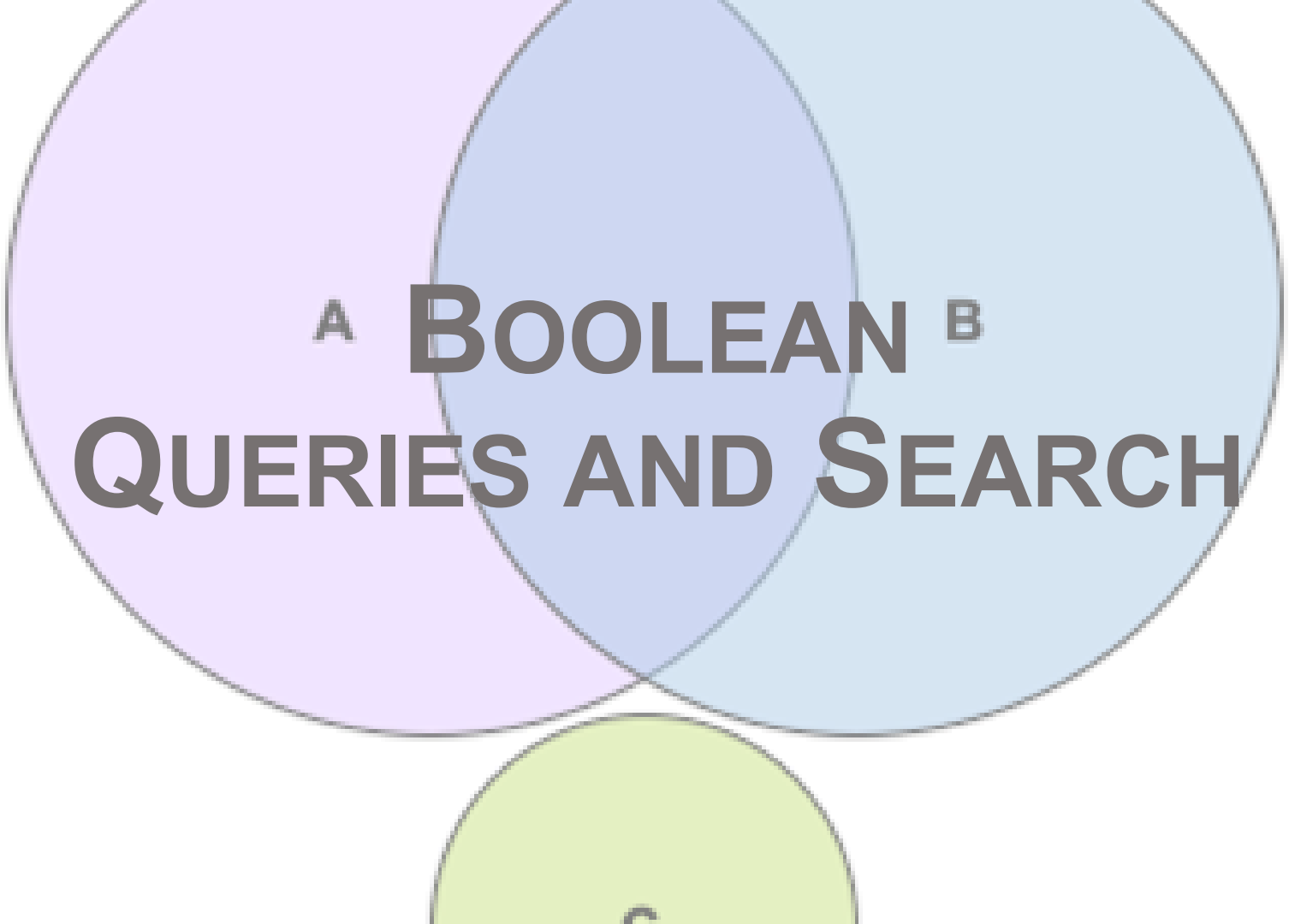
ii. Organize the contents of the pages in a way that allows efficient retrieval (**indexing**)

iii. Take in a query, determine which pages match, and show the results (**ranking** and **display** of results)

Query Languages and Search Types

- A way to express the information need
- Main Types
  - Boolean
  - Keyword / Natural Language
  - Graphical User Interface (GUI)

Boolean: specify a precise set of results, not ranked
Keyword/ NL: fuzzier matching, ranking is key


A
B
BOOLEAN
QUERIES AND SEARCH

Simple Query Language: Boolean

Consists of Terms + Connectors (operators)

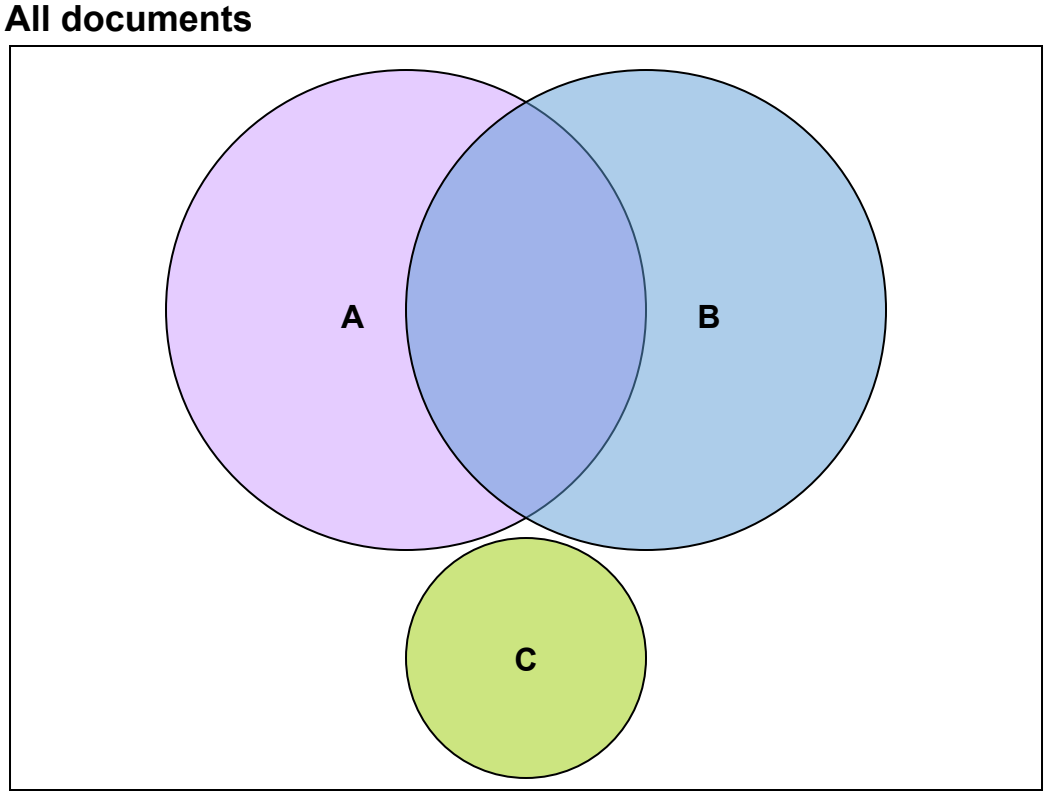
A statement defines which documents to retrieve

- *Terms: words, phrases, synonym expansions*
- *Connectors: AND, OR, NOT*
- *Also can have parentheses for grouping*

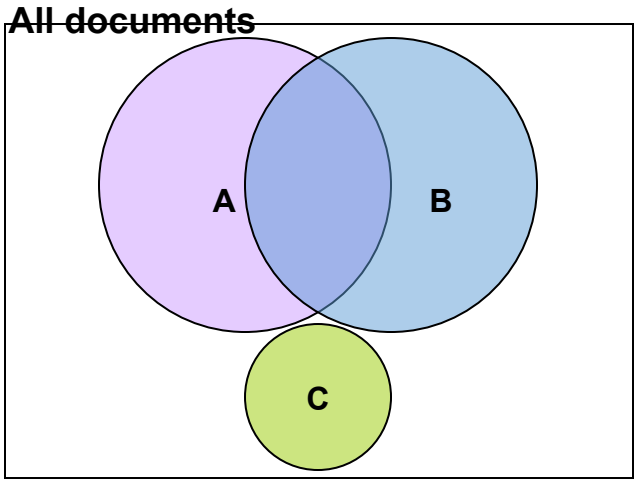
Meaning of Boolean Queries

Cat	All (and only) documents containing “cat” (at least once)
Cat OR Dog	All (and only) documents containing either “cat” or “dog” (or both)
Cat AND Dog	All (and only) documents containing both “cat” and “dog”
Cat AND NOT Dog	All (and only) documents containing “cat” BUT NOT “dog”

Set Representation for Boolean Queries



BOOLEAN OR



A \ B	0	1
0	0	1
1	1	1

A OR B

All of the Lavender Documents

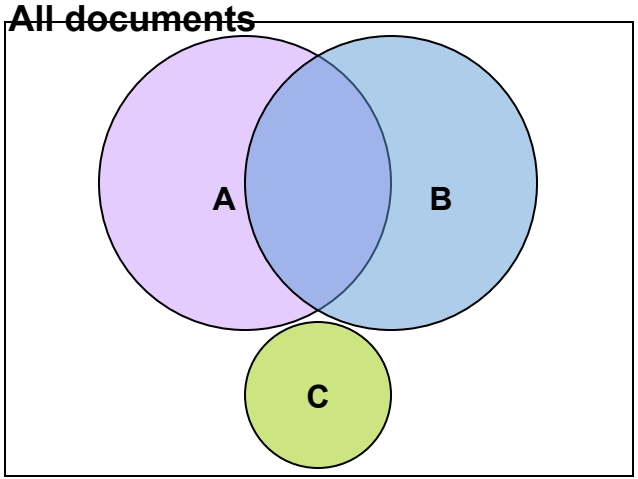
All of the Blue Documents

All of the Purple Documents

Cat OR Dog

All (and only) documents containing either “cat” or “dog” (or both)

BOOLEAN AND



A \ B	0	1
0	0	0
1	0	1

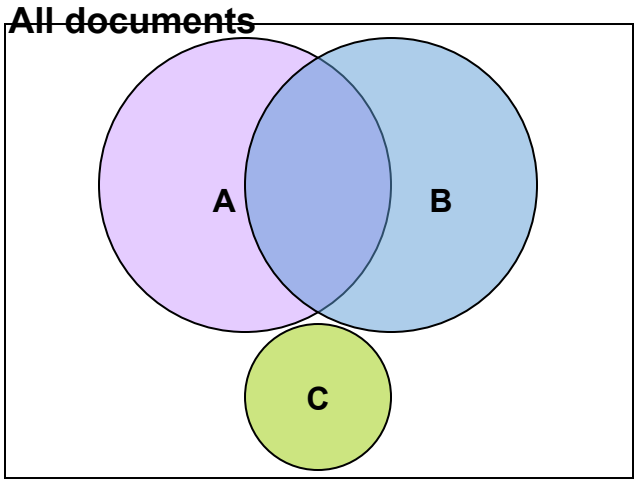
A AND B

All of the Documents at the Intersection of Blue AND Green

Cat AND Dog

All (and only) documents containing BOTH “cat” AND “dog”

BOOLEAN NOT



B	0	1
	1	0

NOT B

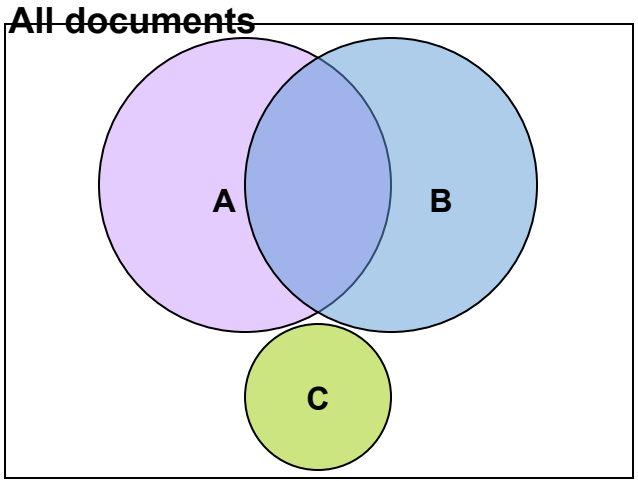
All of the Lavender Documents (NOT the purple part)

All of the Green Documents

Not Dog

All documents EXCEPT those that contain Dog

Combination: A and Not B



A \ B	0	1
0	0	0
1	1	0

A NOT B
(= A AND NOT B)

All of the Lavendar Documents (NOT the purple part)

Cat AND NOT Dog

All documents with Cat EXCEPT those that contain Dog

Boolean Queries and Retrieval

- Weights assigned to terms are either “0” or “1”
  - “0” represents “absence”: term **isn’t** in the document
  - “1” represents “presence”: term **is** in the document
- Build queries by combining terms with Boolean operators
  - AND, OR, NOT
- The system returns all documents that satisfy the query

Boolean View of a Collection

Term	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6	Doc 7	Doc 8
aid	0	0	0	1	0	0	0	1
all	0	1	0	1	0	1	0	0
back	1	0	1	0	0	0	1	0
brown	1	0	1	0	1	0	1	0
come	0	1	0	1	0	1	0	1
dog	0	0	1	0	1	0	0	0
fox	0	0	1	0	1	0	1	0
good	0	1	0	1	0	1	0	1
jump	0	0	1	0	0	0	0	0
lazy	1	0	1	0	1	0	1	0
men	0	1	0	1	0	0	0	1
now	0	1	0	0	0	1	0	1
over	1	0	1	0	1	0	1	1
party	0	0	0	0	0	1	0	1
quick	1	0	1	0	0	0	0	0
their	1	0	0	0	1	0	1	0
time	0	1	0	1	0	1	0	0

Each column represents the view of a particular document: What terms are contained in this document?

Each row represents the view of a particular term: What documents contain this term?

To execute a query, pick out rows corresponding to query terms and then apply the logic table of corresponding Boolean operator

Representing Documents for Boolean Queries

Document 1

The quick brown fox jumped over the lazy dog's back.

Document 2

Now is the time for all good men to come to the aid of their party.

Term	Document 1	Document 2
aid	0	1
all	0	1
back	1	0
brown	1	0
come	0	1
dog	1	0
fox	1	0
good	0	1
jump	1	0
lazy	1	0
men	0	1
now	0	1
over	1	0
party	0	1
quick	1	0
their	0	1
time	0	1

Stopword List

for
is
of
the
to

SAMPLE QUERIES

Term	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6	Doc 7	Doc 8
dog	0	0	1	0	1	0	0	0
fox	0	0	1	0	1	0	1	0
dog ∧ fox	0	0	1	0	1	0	0	0
dog ∨ fox	0	0	1	0	1	0	1	0
dog ¬ fox	0	0	0	0	0	0	0	0
fox ¬ dog	0	0	0	0	0	0	1	0

dog AND fox → Doc 3, Doc 5

dog OR fox → Doc 3, Doc 5, Doc 7

dog NOT fox → empty

fox NOT dog → Doc 7

Term	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5	Doc 6	Doc 7	Doc 8
good	0	1	0	1	0	1	0	1
party	0	0	0	0	0	1	0	1
g ∧ p	0	0	0	0	0	1	0	1
over	1	0	1	0	1	0	1	1
g ∧ p ¬ o	0	0	0	0	0	1	0	0

good AND party → Doc 6, Doc 8

good AND party NOT over → Doc 6

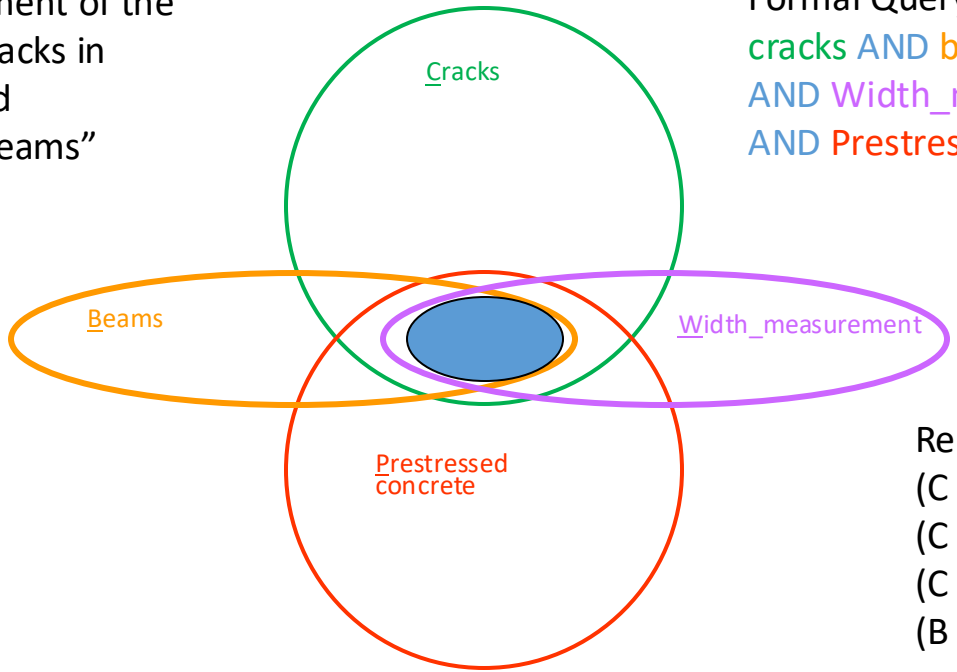
Slide adapted from Jimmy Lin

Combinations of Boolean Operations

- Cat
- Cat OR Dog — Disjunct
- Cat AND Dog — Conjunct
- (Cat AND Dog) — Conjunct
- (Cat AND Dog) OR (Collar AND Leash) — Disjunct of Conjuncts
- (Cat OR Dog) AND (Collar OR Leash) — Conjunct of Disjuncts

Converting Natural Language to Boolean

"Measurement of the width of cracks in prestressed concrete beams"



Formal Query:
cracks AND beams
AND Width_measurement
AND Prestressed_concrete

Relaxed Query:
(C AND B AND P) OR
(C AND B AND W) OR
(C AND W AND P) OR
(B AND W AND P)

Faceted Boolean Query

Strategy: break query into facets

- Conjunction of disjunctions

{ a1 OR a2 OR a3
  b1 OR b2
  c1 OR c2 OR c3 OR c4 } AND

Faceted Boolean Query

Strategy: break query into facets

- Conjunction of disjunctions

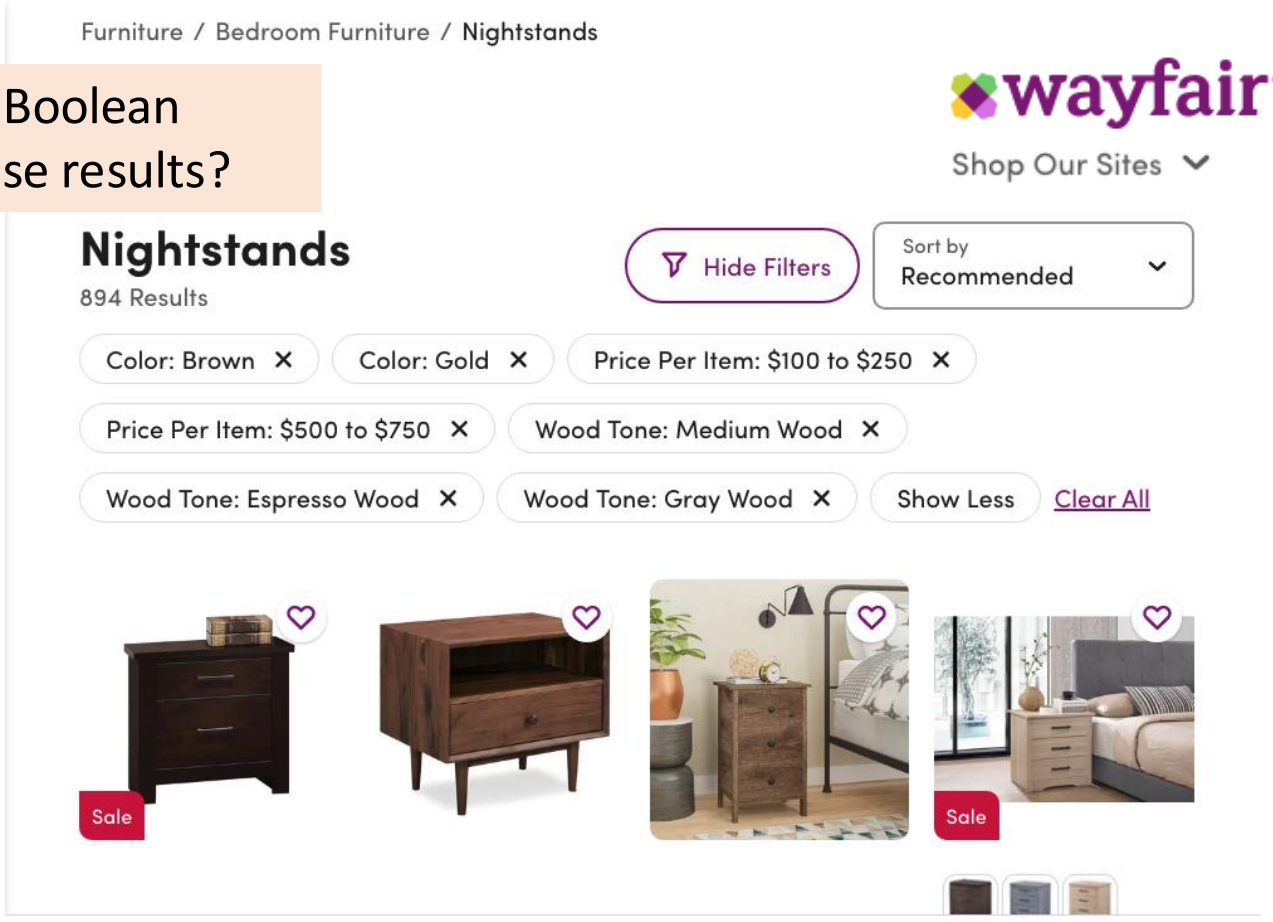
{ a1 OR a2 OR a3
b1 OR b2
c1 OR c2 OR c3 OR c4 } AND

- Each facet expresses a topic

{ “rain forest” OR jungle OR amazon
medicine OR remedy OR cure
Smith OR Zhou } AND across each line

Faceted Navigation Supports a Conjunct of Disjuncts

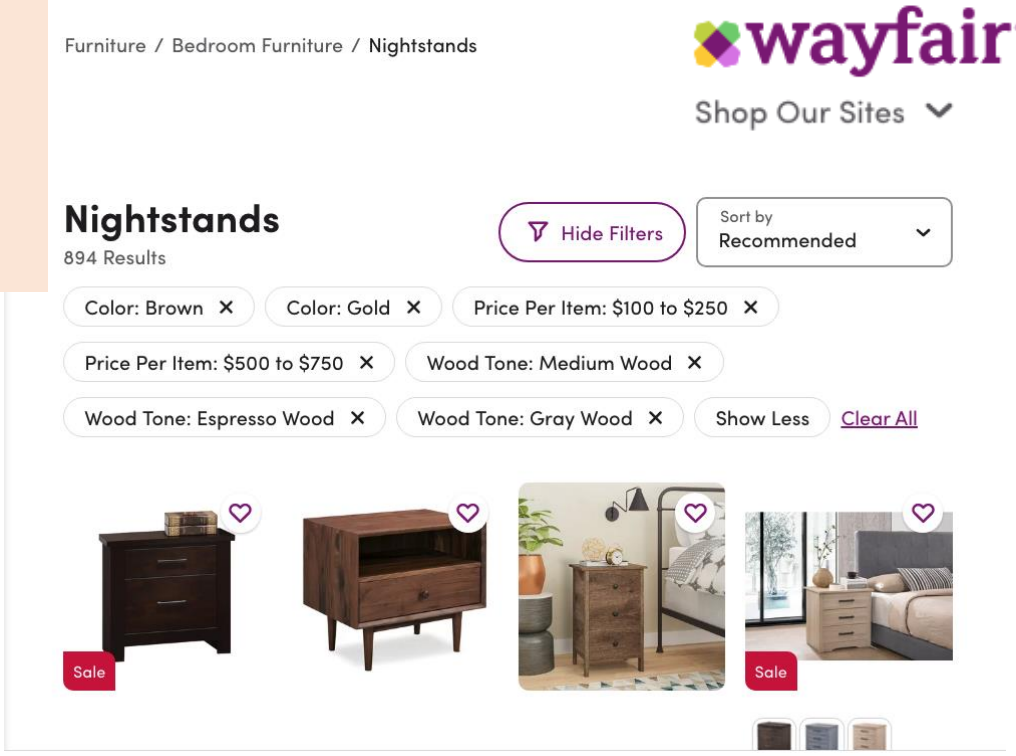
Exercise: what is the Boolean query expressing these results?



Faceted Navigation Supports a Conjunct of Disjuncts

Show all products that are:

Nightstands AND
(brown OR gold) AND
($100-250 OR $500-750) AND
(Medium OR Expresso OR Gray)



Ordering of Retrieved Documents

Pure Boolean Search has no ordering

In practice:

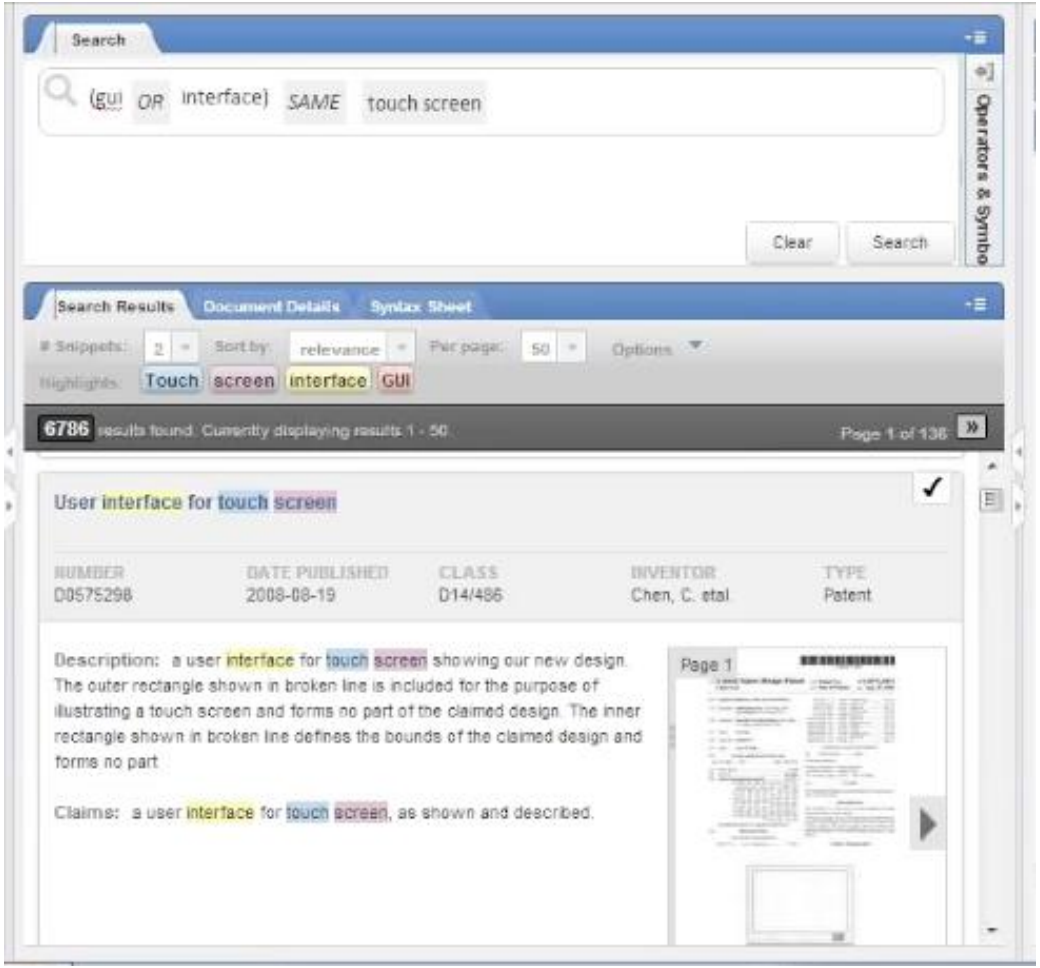
- *Order chronologically (by time)*
- *Order by total number of "hits" on query terms*
  - What if one term has more hits than others?
  - Is it better to one of each term or many of one term?
- *Use some other statistical properties*

Proximity Searches

- Proximity: terms occur within K positions of one another
  - *pen w/5 paper*
- A “Near” function can be more vague
  - *near(pen, paper)*
- Sometimes order can be specified
- Also, Phrases and Collocations
  - *“United Nations” “Barack Obama”*
- Phrase Variants
  - *“retrieval of information” “information retrieval”*

Boolean Proximity Queries Are Still Heavily Used in Legal Search

Prototype of search
For patent examiners:
Note use of “same” operator
Meaning same paragraph



https://wiki.piug.org/pages/viewpage.action?pageId=33850015

Google Patents

(touch screen) ((user interface) OR (gui)) (touch screen NEAR gui) status:

SEARCH TERMS

touch screen × or + Synonym

user interface × or gui × or + Synonym

touch screen NEAR gui × or + Synonym

+ Synonym G06F3/0362

SEARCH FIELDS

Date · Priority

YYYY-MM-DD – YYYY-MM-DD

+ Inventor

+ Assignee

Patent Office · Language

Status · Grant · Type

Litigation

About 135,295 results

Download with Cor

Sort by · Relevance · Group by · None · Deduplicate by · Family · Results / page · 10

Graphical user interface touch screen with auto zoom feature

WO EP US ~~CN~~ JP ~~KR~~ ~~DE~~ ~~TW~~ · JP5543426B2 · ソン エム チョイ · コーニンクレッカ フィリップス エヌ ヴェ

Priority 1998-04-17 · Filed 2011-12-26 · Granted 2014-07-09 · Published 2014-07-09

Graphical **user interfaces** are well known in the art. US Pat. No. 5,463,725 is an example of a **GUI** having **touch screen** functionality. This US patent is included for reference. 1A and 1B show a PDA 10 having a **touch screen GUI** 13 according to the present invention. The keyboard icon 12 is displayed …

Enhanced target selection for a touch-based input enabled user interface

WO EP US CN · US10684768B2 · Tovi Grossman · Autodesk, Inc.

Priority 2011-10-14 · Filed 2011-10-14 · Granted 2020-06-16 · Published 2020-06-16

Field of the Invention The invention relates generally to a **user interface** and, more specifically, to enhanced target selection for a touch-based input enabled **user interface**. Description of the Related Art **Touch screen** displays are becoming increasingly common in consumer devices. For example, …

For checking the application of image

WO EP US CN JP KR · CN103729115B · R·乌比洛斯 · 苹果公司

Priority 2012-03-06 · Filed 2013-01-03 · Granted 2017-03-01 · Published 2017-03-01

In certain embodiments, application provides the **user interface** mechanism for switching in both modes.For example, exist In some embodiments, thumbnail viewing area is moveable, and applies the position in the **gui** being moved to based on thumbnail And switch between left hand and right-handed mode. …

Wearable electronic device

WO EP US CN JP KR AU BR *IN* MX RU · JP6509486B2 · プラナヴ・ミストリー · 三星電子株式会社Ｓａｍｓｕｎｇ Ｅｌｅｃｔｒｏｎｉｃｓ Ｃｏ．，Ｌｔｄ

SUMMARY: BOOLEAN SEARCHING

- Advantages
  - *simple queries are easy to understand*
  - *relatively easy to implement*
- Disadvantages
  - *difficult to specify what is wanted (have to state synonyms)*
  - *too much returned, or too little*
  - *ordering not well determined*
- Dominant language in commercial systems until the WWW
  - *Still used quite a lot in legal searches*
- Still heavily used in UIs in website search (faceted navigation)

RANKING ALGORITHMS

Ranked Retrieval

Order documents by how **likely they are to be relevant** to the information need

RESULTS RANKING

- Search engine receives a query, then
- Looks up the words in the index, retrieves many documents, then
- Rank orders the pages and extracts “snippets” or summaries containing query words.
- These are complex algorithms

Slide adapted from Lew & Davis

Kerry Rodden

Published Oct 2, 2019 2 forks 108 Likes

Introduction to text analysis with TF-IDF

Earlier this year, the *New York Times* published an "Overlooked no more" belated obituary of Karen Spärck Jones, who invented the concept of Inverse Document Frequency in 1972. This is now best known as part of the text analysis technique TF-IDF (Term Frequency - Inverse Document Frequency).

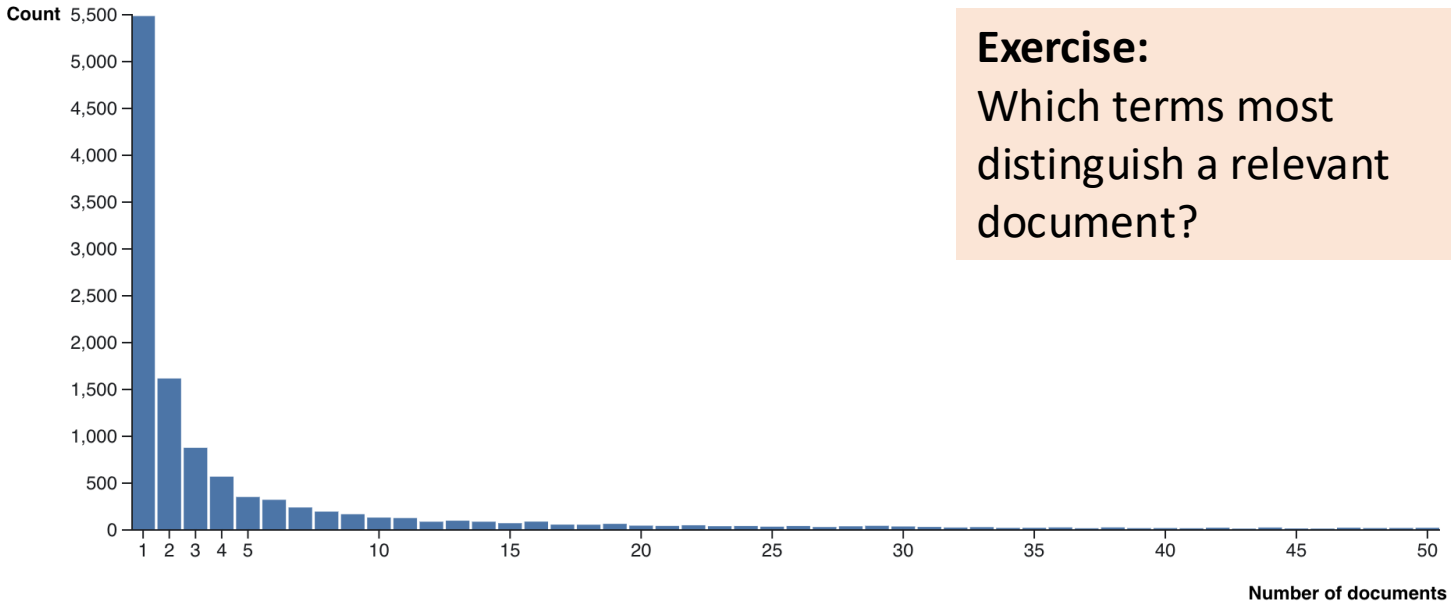
Dataset: State Governor Speeches

Mr. Speaker, thank you for being a champion for all Californians – and for welcoming Jen and me into your house today.
Madam Pro Tem – thank you for your commitment to collaboration, which has helped make our first month together so productive.
I also have the honor of saying for the first time ever in this chamber: thank you Madam Lieutenant Governor for that very kind and short introduction.
To all the constitutional officers and legislators assembled here today – thank you for your service to our state.
And let me reassure everyone: our son Dutch is not here. We learned our lesson at the inauguration.
It was just over four weeks ago that I stood in front of this Capitol and pledged to defend not just the California constitution but the California dream.
Today, I want to talk about how we can do that together.
By every traditional measure, the state of our state is strong.
We have a record-breaking surplus.
We've added 3 million jobs since the depths of the recession.
Wages are rising.
We have more scientists, researchers, and engineers, more Nobel laureates, and the finest system of higher education anywhere in the world.
But along with that prosperity and progress, there are problems that have been deferred for too long and that threaten to put the California dream out of reach for too many.
We face hard decisions that are coming due.
The choices we make will shape our future for decades.
This is what I want to talk about today, as frankly and directly as I can:
The tough calls we must make together on rail, water, and energy. How we protect migrants, care for seniors, and help the homeless, and how we will tackle the affordability crisis that is coming to define life in this state.
I won't pretend to have all the answers. But the only way to find them is to face these issues honestly.
Let's start with the fear mongering from the White House about the so-called "emergency" at our border.
For me, this is an echo from 15 years ago.
I was a new mayor sitting in the gallery at the State of the Union when President Bush said LGBT Americans should not be able to get married.

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Document frequencies follow a power law
(Zipf's curve)

Here's a histogram showing the number of terms that appear in each number of documents. You can see that a very large number of terms appear in only one document, and a very small number appear in every document.



Exercise:
Which terms most distinguish a relevant document?

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Word Frequencies have Long Tails

Zipf's Law

Zdeněk Macháček on Unsplash

Zipf's Law

(Empirical behavior, noted by linguist George K. Zipf)

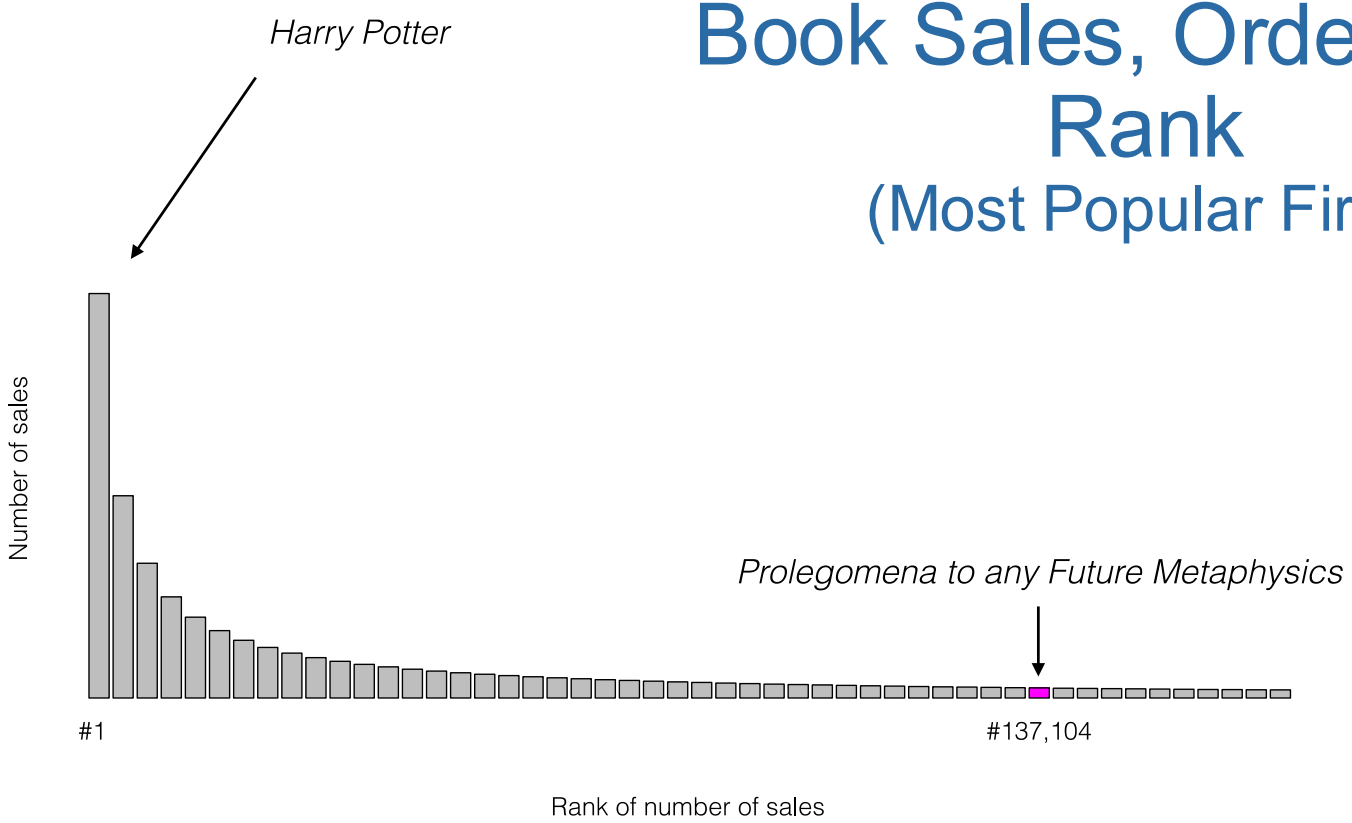
For many phenomena, there is a relationship (a power law) between the frequency of an event and the rank of that frequency among all events.

- Token occurrences in text are not uniformly distributed
- They are also not normally distributed
- They do exhibit a Zipf distribution

What Kinds of Data Exhibit a Zipf Distribution?

- Words in a text collection
- Social Network Popularity
- Website Popularity
- Document Size on Web
- And many many other phenomena like this

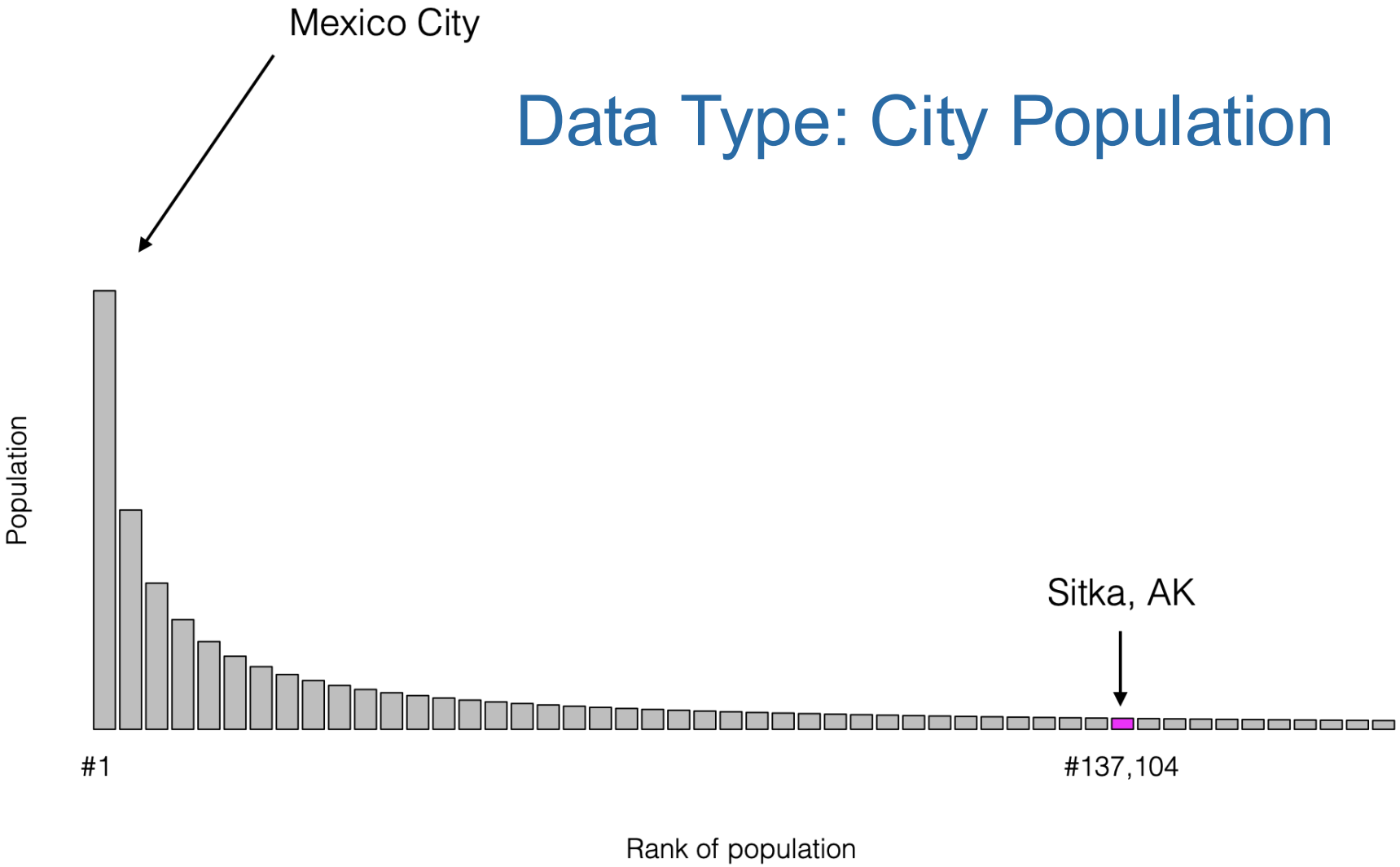
Book Sales, Ordered by Rank
(Most Popular First)



r = **rank** (x-axis). Most popular is ranked #1

f = **frequency** (y-axis) The # items in the rank

Data Type: City Population



Assigning Weights with TF-IDF

Intuition:

Terms that appear **often in a document** should get **higher** weight

BUT terms that appear in **many document**s should get **lower** weights

This is linked to the fact that terms frequencies follow a power law (Zipf's curves)

TF x IDF measure:

term frequency (tf)
inverse document frequency (idf)

Assign a TF x IDF weight to each term in each document

There are many ways to compute this measure

The reading shows BM-25 (also called Okapi); this is now frequently used

TF-IDF Term Weighting: Simple, Yet Effective!

$$w_{i,j} = \mathrm{tf}_{i,j} \cdot \log \frac{N}{n_i}$$

This is the idf part

$w_{i,j}$ **weight assigned to term *i* in document *j***

$\mathrm{tf}_{i,j}$ **number of occurrences of term *i* in document *j***

N **number of documents in entire collection**

n_i **number of documents with term *i***

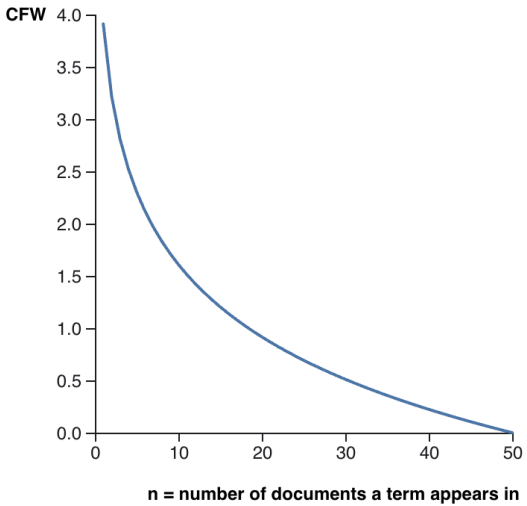
$$log(N/n) = log(N) - log\ (n)$$

Inverse Document Frequency (IDF)
(here called Collection Frequency Weight)

the Collection Frequency Weight for a term is then

$$CFW(i) = \log(N) - \log(n)$$

This is visualized in the graph below, for our collection of $N = 50$ documents. So, for example, a term that occurs in 30 documents would have a CFW of about 0.5.



Page 1

Term	CF	CFW
state	50	0.0198
also	50	0.0198
people	50	0.0198
together	50	0.0198
years	50	0.0198
one	50	0.0198
us	50	0.0198
new	50	0.0198
working	50	0.0198
make	50	0.0198
first	50	0.0198
education	50	0.0198
school	50	0.0198
work	50	0.0198
thank	50	0.0198

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Computing BM-25 (also called Okapi)
(This is still a very popular method)

The term frequency for term $t(i)$ in document $d(j)$ is:

$TF(i,j) =$ the number of occurrences of term $t(i)$in document $d(j)$

Here are the most frequent terms for the California and the Arizona speech, after stopword filtering. These are the top TF(i,j) where i is the term and j is the document

Term	Frequency
california	32
state	27
let	18
water	16
need	16
must	15
new	14
many	13
like	13
support	11

Term	Frequency
arizona	37
let	23
state	17
today	15
people	14
work	13
government	13
got	13
last	12
new	11

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Computing BM-25

n = the number of documents term t(i) occurs in
N = the number of documents in the collection

the Collection Frequency Weight for a term is then

$$CFW(i) = \log(N) - \log(n)$$

log(N/n) = log(N) – log (n)

$DL(j)$ = the total of term occurrences in document $d(j)$

Normalize weight of term i across documents

terms in d(j) (doc length)

$$NDL(j) = \frac{DL(j)}{(\text{Average DL for all documents})}$$

Normalizes across all document lengths

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

COMPUTING BM-25

For one term $t(i)$ and one document $d(j)$, the Combined Weight is

$$CW(i,j) = \frac{CFW(i) * TF(i,j) * (K1+1)}{K1 * ((1-b) + (b * NDL(j))) + TF(i,j)}$$

- K1 modifies the importance of term frequency
  - Higher values would increase the influence of TF; K1=0 eliminates the influence altogether.
- b modifies document length (between 0 and 1)
  - "Setting b towards 1, e.g. b=.75, will reduce the effect of term frequency on the ground that it is primarily attributable to verbosity. If b=0 there is no length adjustment effect, so greater length counts for more" (perhaps the doc has multiple topics)

Computing BM-25

Say we have document j = the speech by the governor of California, and the term we care about $t(i)$ = climate.

- the frequency of climate in this document is $TF(i,j) = 5$,
- climate appears in 21 documents, so $CFW(i) = log(50) - log(21) = .376$
- We know that the length of the California speech, $DL(j) = 4429$,
- and that the average speech length in the collection is 4272, so
- $NDL(j) = 4429 \div 4272 = 1.036$

Let's also set $K1 = 2$ and $b = .75$. Now we can compute the tf-idf weight of this (term, document) pair

$$CW(i,j) = \frac{.376 * 5 * (2+1)}{2 * ((.25) + (.75 * 1.036)) + 5}$$

$$: \frac{CFW(i) * TF(i,j) * (K1+1)}{K1 * ((1-b) + (b * NDL(j))) + TF(i,j)}$$

$$CW(i,j) = \frac{5.64}{7.054} = .80$$

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Computing BM-25

Now we are able to score the documents in the collection, for a given query. To do this, we take every document, identify which of the query terms appear in it, and add together their combined (TF-IDF) weights.

Query="hurricane"

Document	Score
North Carolina	4.7991
Georgia	4.2730
Texas	3.7918
South Carolina	3.5736
Florida	3.1130
Mississippi	1.9270
Virginia	1.6914

Query="hurricane Florence"

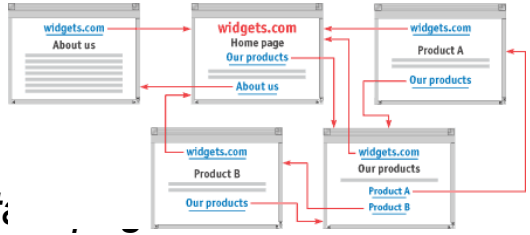
Document	Score
North Carolina	11.4307
South Carolina	8.6719
Georgia	4.2730
Virginia	4.1044
Texas	3.7918
Florida	3.1130
Mississippi	1.9270

https://observablehq.com/@kerryrodden/introduction-to-text-analysis-with-tf-idf

Some Additional Ranking criteria

- For a given candidate result page, use:
  - *Number of matching query words in the page*
  - *Proximity of matching words to one another*
  - *Location of terms within the page*
  - *Location of terms within tags e.g. <title>, <h1>, link text, body text*
  - *Anchor text on pages pointing to this one*
  - *Frequency of terms on the page and in general*
  - *Link analysis of which pages point to this one*
  - *(Sometimes) Click-through analysis: how often the page is clicked on*
  - *How "fresh" is the page*
- Complex formulae combine these together.
- Combining these via machine learning started in the 2000's

Measuring the Importance of Linking



- **PageRank Algorithm**
  - *Idea: important pages are pointed to by other importa*
  - *Method:*
    - Each link from one page to another is counted as a “vote” for the destination page
    - But the importance of the starting page also influences the importance of the destination page.
    - And those pages scores, in turn, depend on those linking to them.

Image and explanation from http://www.economist.com/science/tq/displayStory.cfm?story_id=3172188

Measuring the Importance of Linking

PageRank

- Example: each page starts with 100 points.
- Each page's score is recalculated by adding up the score from each incoming link.
  - This is the score of the linking page divided by the number of outgoing links it has.
  - Example: the page in green has 2 outgoing links and so its "points" are shared evenly by the 2 pages it links to.
- Keep repeating the updates until no more changes.

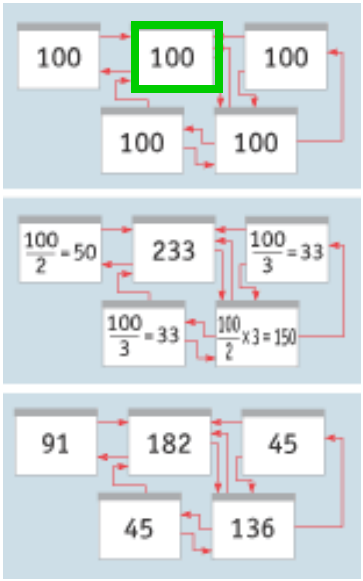


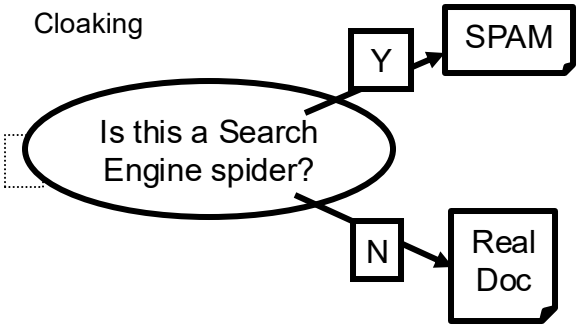
Image and explanation from http://www.economist.com/science/tq/displayStory.cfm?story_id=3172188

MANIPULATING RANKING

- Motives
  - *Commercial, political, religious*
  - *Promotion funded by advertising budget*
- Operators
  - *Search Engine Optimizers*
  - *Web masters*
  - *Hosting services*
- Forum and Websites
  - *Web master world (www.webmasterworld.com)*
  - *Searchengineland.com*

A FEW SPAM TECHNOLOGIES

- **Cloaking**
  - *Serve fake content to search engine robot*
  - *DNS cloaking: Switch IP address. Impersonate*
- **Doorway pages**
  - *Pages optimized for a single keyword that re-direct to the real target page*
- **Keyword Spam**
  - *Misleading meta-keywords, excessive repetition of a term, fake "anchor text"*
  - *Hidden text with colors, CSS tricks, etc.*
- **Link spamming**
  - *Mutual admiration societies, hidden links, awards*
  - *Domain flooding: numerous domains that point or re-direct to a target page*
- **Robots**
  - *Fake click stream*
  - *Fake query stream*
  - *Millions of submissions via Add-Url*



Meta-Keywords =
"… London hotels, hotel, holiday inn, hilton, discount, booking, reservation, sex, mp3, britney spears, viagra, …"

Slide adapted from Manning, Raghavan, & Schuetze

PAID RANKING

Pay-for-inclusion

- Deeper and more frequent indexing
- Sites are not distinguished in results display

Paid placement

- Keyword bidding for targeted ads

SUMMARY

- Document ranking is complex; we've only scratched the surface and talked about classic approaches
- Machine learning using click data has become dominant on the web (pre-LLM)
- That data isn't available however for other search (on your desktop, in your organization, etc)