X-Informatics
Web Search and Text Mining Part I

July 5 2013

Geoffrey Fox
gcf@indiana.edu
http://www.infomall.org/X-InformaticsSpring2013/index.html

Associate Dean for Research,
School of Informatics and Computing
Indiana University Bloomington
2013

Big Data Ecosystem in One Sentence

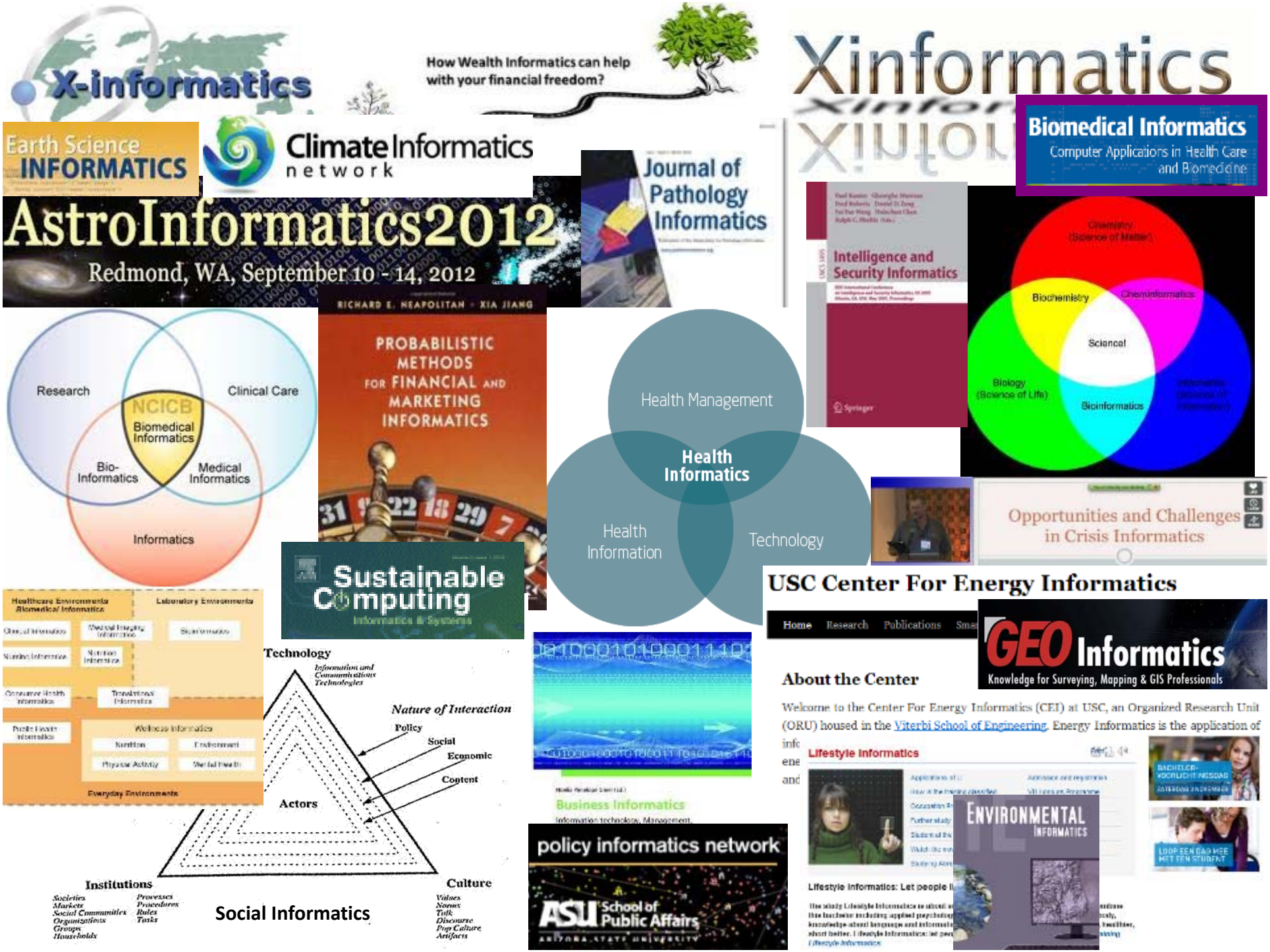
Use Clouds running Data Analytics Collaboratively processing Big Data to solve problems in X-Informatics (or e-X)

X = Astronomy, Biology, Biomedicine, Business, Chemistry, Climate, Crisis, Earth Science, Energy, Environment, Finance, Health, Intelligence, Lifestyle, Marketing, Medicine, Pathology, Policy, Radar, Security, Sensor, Social, Sustainability, Wealth and Wellness with more fields (physics) defined implicitly

Spans Industry and Science (research)

Education: Data Science see recent New York Times articles

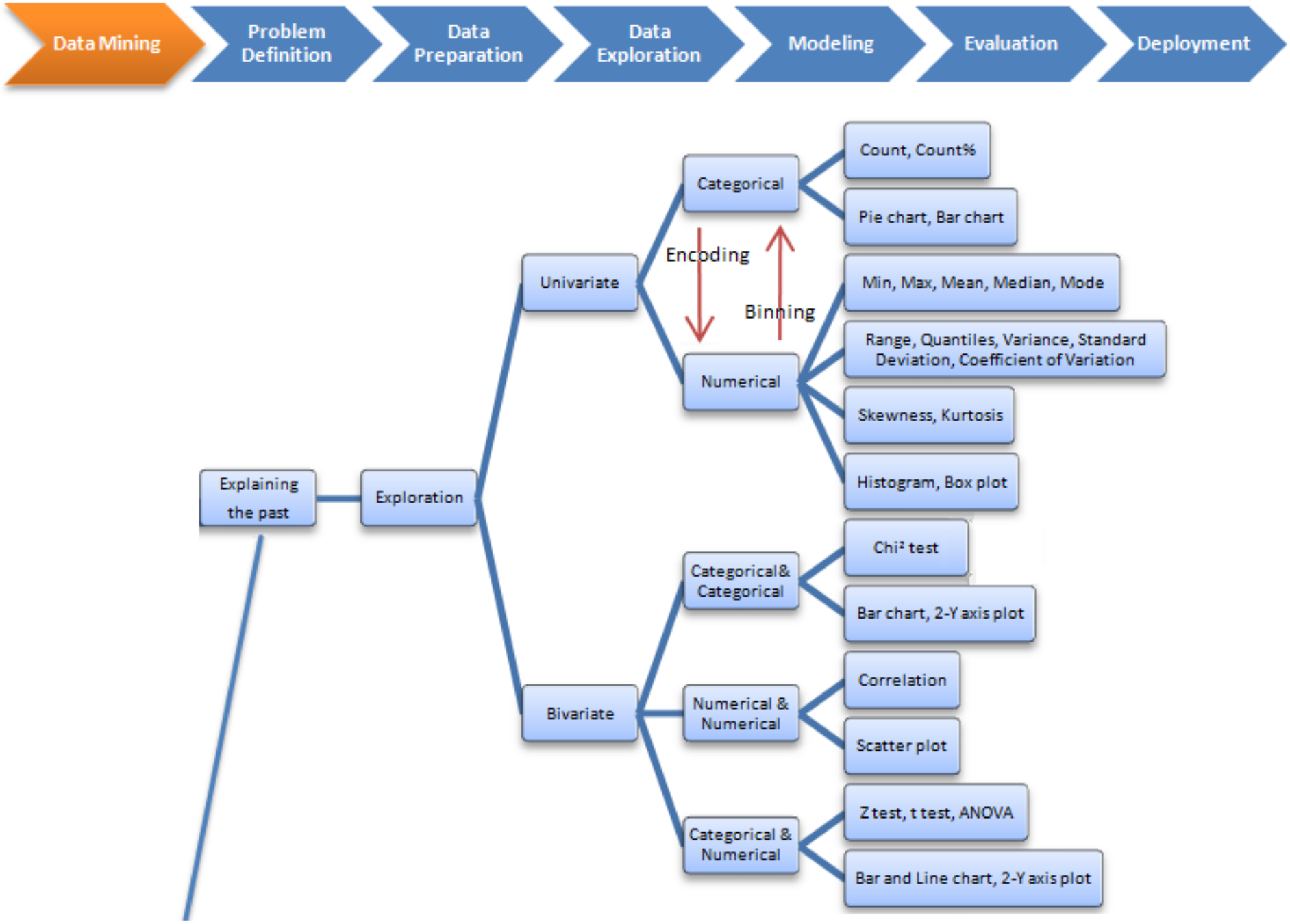
http://datascience101.wordpress.com/2013/04/13/new-york-times-data-science-articles/


X-informatics
How Wealth Informatics can help with your financial freedom?
Xinformatics
Earth Science INFORMATICS
Climate Informatics network
Biomedical Informatics
Computer Applications in Health Care and Biomedicine
Journal of Pathology Informatics
AstroInformatics2012
Redmond, WA, September 10 - 14, 2012
Intelligence and Security Informatics
Chemistry (Science of Matter)
Biochemistry
Chemiinformatics
Sciencal
Biology (Science of Life)
Bioinformatics
RICHARD E. NEAPOLITAN · XIA JIANG
PROBABILISTIC METHODS FOR FINANCIAL AND MARKETING INFORMATICS
Research
Clinical Care
NCICB
Biomedical Informatics
Bio-Informatics
Medical Informatics
Informatics
Health Management
Health Informatics
Health Information
Technology
Opportunities and Challenges in Crisis Informatics
Sustainable Computing
Informatics & Systems
USC Center For Energy Informatics
Home
Research
Publications
GEO Informatics
Knowledge for Surveying, Mapping & GIS Professionals
About the Center
Welcome to the Center For Energy Informatics (CEI) at USC, an Organized Research Unit (ORU) housed in the Viterbi School of Engineering. Energy Informatics is the application of
Healthcare Environments Biomedical Informatics
Laboratory Environments
Everyday Environments
Technology
Information and Communications Technologies
Nature of Interaction
Policy
Social
Economic
Content
Actors
Institutions
Societies
Markets
Social Communities
Organizations
Groups
Households
Processes
Procedures
Rules
Tasks
Culture
Values
Norms
Talk
Discourse
Pop Culture
Artifacts
Social Informatics
Business Informatics
policy informatics network
ASU School of Public Affairs
Lifestyle Informatics
Lifestyle Informatics: Let people
ENVIRONMENTAL INFORMATICS

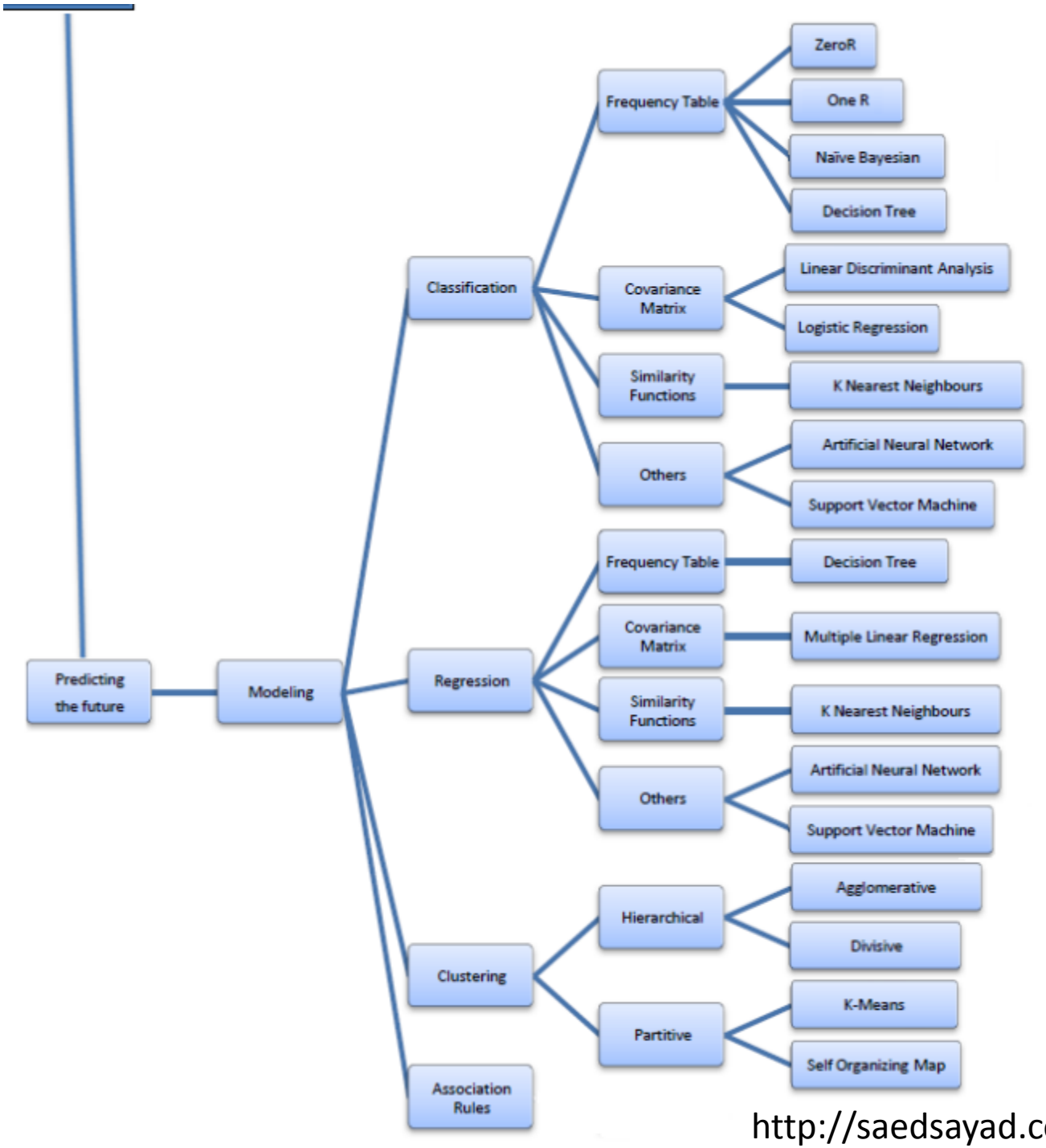
Data Mining Survey

Survey of Data mining

- http://saedsayad.com/data_mining_map.htm
- This site has a survey of data mining areas
- In next units we will discuss application of “classification” and “clustering” to text and web data (see second side of datamining survey)
- This is similar to recommender systems in broad area
- The Physics Higgs study was “exploration” on first slide



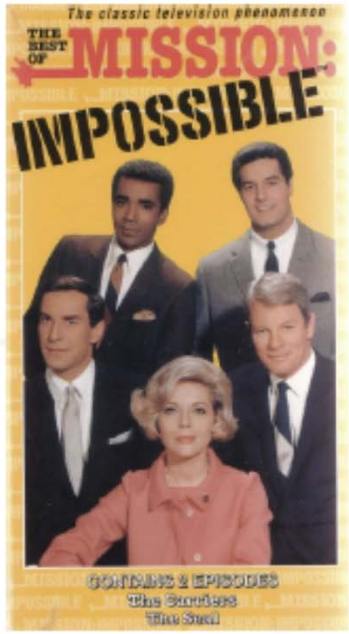
http://saedsayad.com/data_mining_map.htm



http://saedsayad.com/data_mining_map.htm

Web and Document/Text Search
The Problem

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html



Mission Impossible?

Search engines:

- Crawl and index tens of billions of documents
- Answer hundreds of millions of queries per day
- Devote less than 1 second of processing time per query execution

Users:

- Submit very short queries (averaging about 2.6 terms)
- Expect to receive the most relevant results on the Web
- In a blink of an eye

In terms of 1990 IR research - almost unimaginable!

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

The Static Web "Corpus"

- **Bulk** ..... 10s of billions of "static" pages; doubling every 8-24 months
- **Lack of stability**..... Estimates vary widely (half life of page is several months)
- **Lexicon Size** ..... 10s-100s of millions of words
- **Heterogeneity**
  - Type of documents ..... Text, pictures, audio, video, scripts,...
  - Quality ..... From junk to the best content on earth
  - Languages ..... 100+
- **Duplication**
  - Syntactic..... 30%-40% (near) duplicates
  - Semantic..... ??
- **Complex graph topology**..... ≥ 8 links/page in the average; bow-tie structure
- **Spam**..... 100s of millions of pages

Plus the deep Web

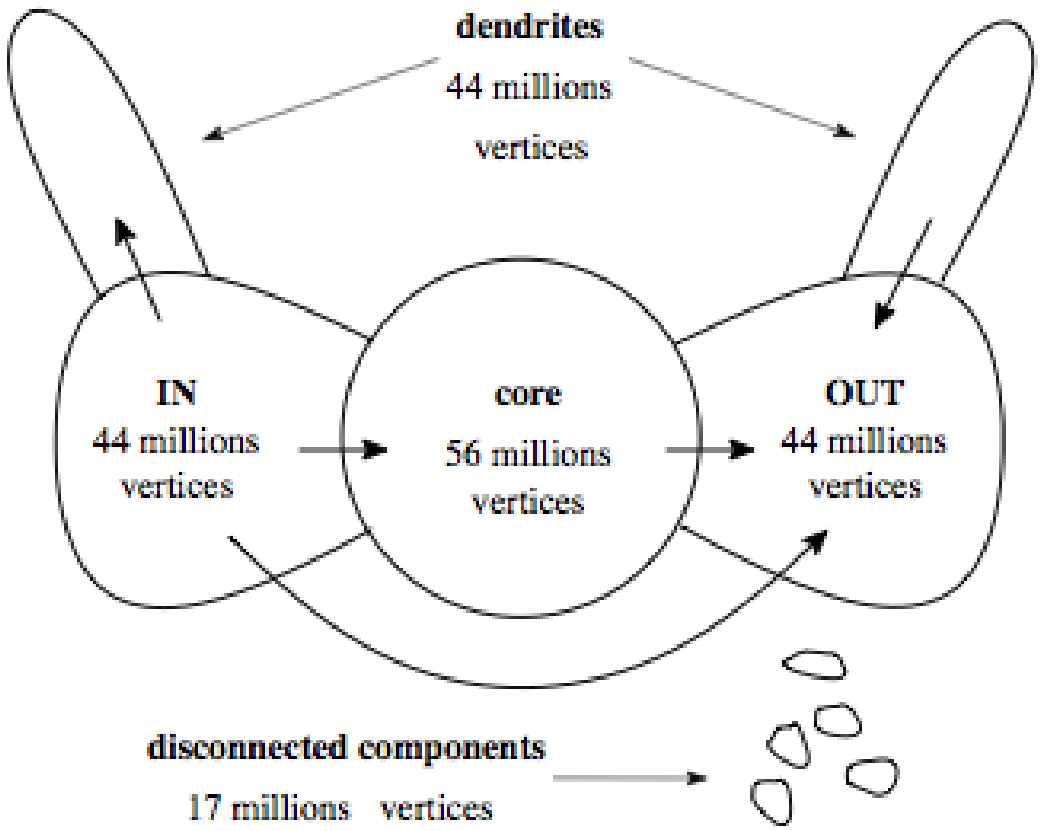
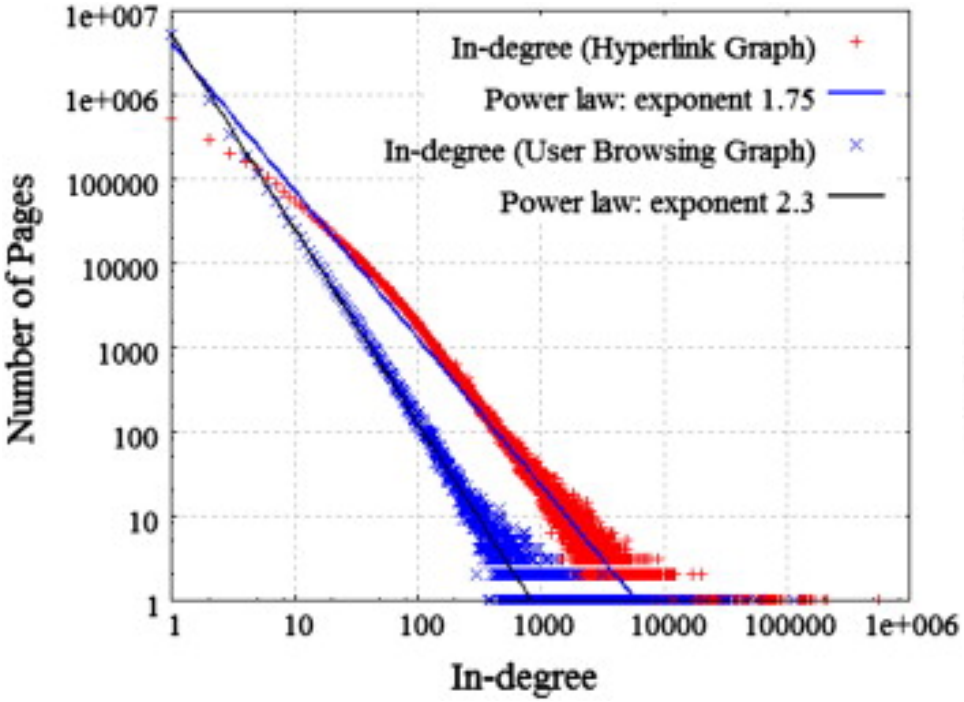


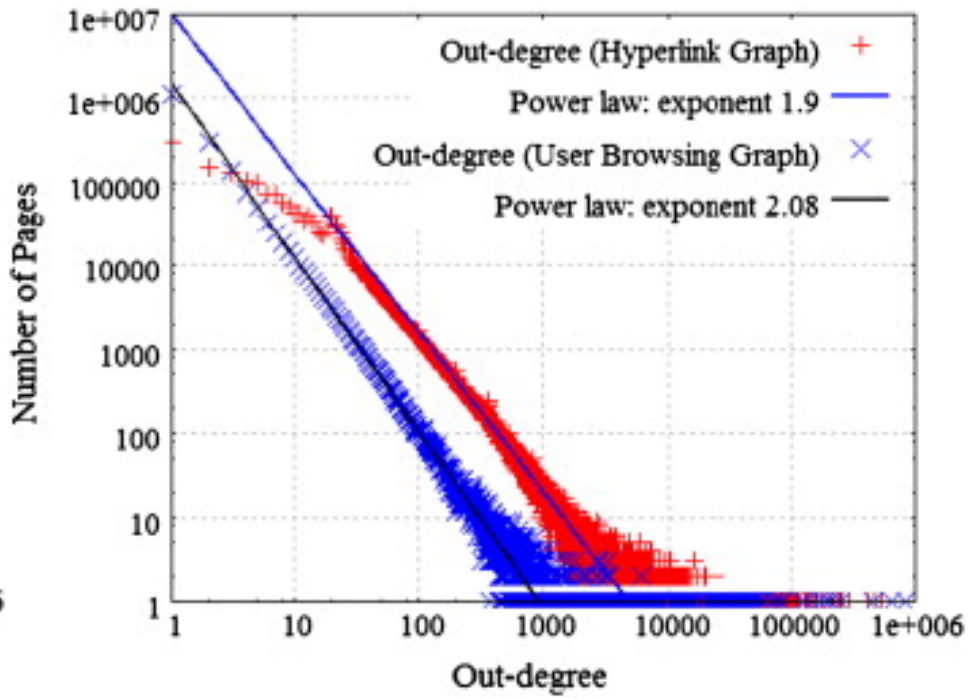
Fig. 3: *The bow-tie macroscopic structure of the Web graph [BKM+00]: the core, the IN component, the OUT component and the dendrites. Each of these parts contains around one quarter of the pages, the disconnected part being reduced to less than 10% of the whole.*

http://hal.archives-ouvertes.fr/docs/00/05/44/58/PDF/webgraph.pdf

Universal Power Laws for Links

http://www.sciencedirect.com/science/article/pii/S0167923612001844



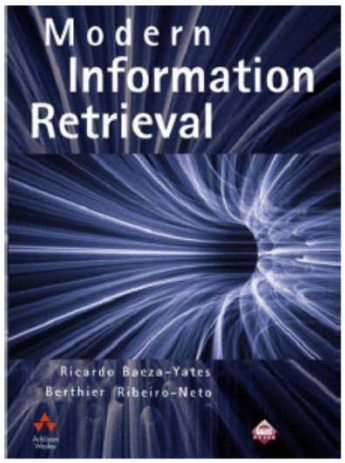


http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Web Searchers - Observations

- Make ill defined queries
  - Short (2.54 terms average, 80% contain less than 3 words)
  - Use imprecise (and often misspelled) terms
  - Unfamiliar with query syntax (80% queries without operator)
- Wide variance in information needs, expectations, education/knowledge, IP bandwidth, human patience
  - Different modalities (mobile, desktop) = different needs, even with same person
- Specific behavior
  - 85% look over one result screen only (mostly "above the fold")
  - 78% of queries are not modified
- Overall, we as users are investing low cognitive effort per query (formulating and looking at results)

What is Information Retrieval (IR)?



IR: Part of computer science which studies the **retrieval of information (not data)** from a collection of **written documents.** The retrieved documents aim at satisfying a **user information need** usually expressed in **natural language.**

- **Documents, unstructured, text, large**
- **Information need**
- **Store, search, find**
- **The World Wide Web?**
- **Relational databases?**

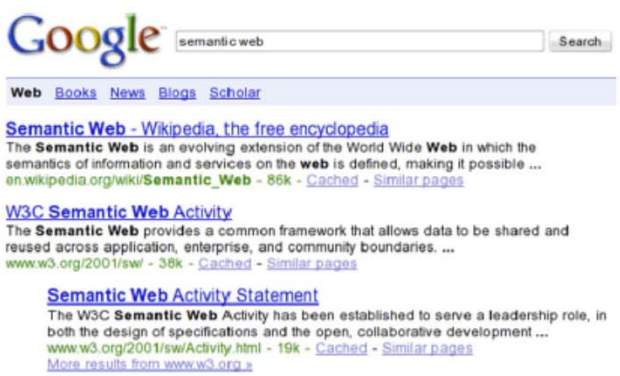
http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

DIKW

- **Data** = Raw web pages
- **Information** = Result of Query
- **Knowledge** = Result of processing query results by user
- **Wisdom** = Synthesis of many such actions by a set of users
- One possible classification of steps in process

Information Retrieval vs. Databases

Information retrieval	Data retrieval
Retrieve all objects **relevant** to some **information need**	Retrieve all objects satisfying some **clearly defined conditions**
Find all documents about the **topic** "semantic web"!	**SELECT** id **FROM** document **WHERE** title **LIKE** '%semantic web%'
Result **list**	**Well-defined** result **set**




```
[selke@tbdb ~]$ db2 "SELECT id FROM document WHERE title L
IKE '%semantic web%' FETCH FIRST 3 ROWS ONLY"

ID
--------------------
               45489
             9635899
               98556

  3 record(s) selected.
```


http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Keywords v Full Text

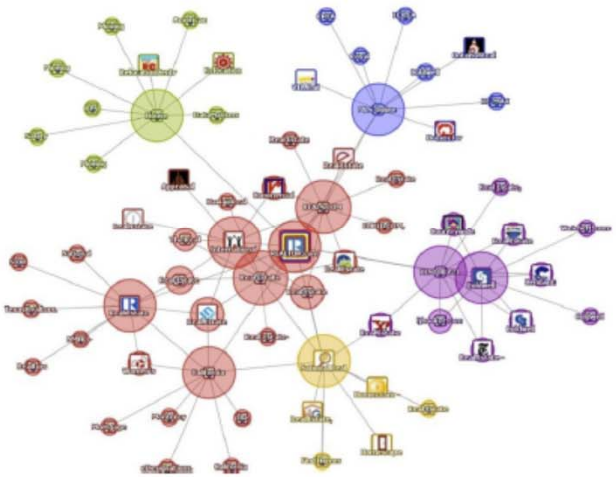
- Keywords are typically more precise
  - Need an ontology (list of possible keywords) but hard to get agreement except on basic things
  - See http://www.acm.org/about/class/ccs98-html
- Dublin core specifies core document information like Author and Title
- Tags tend not to enforce ontology
- Full text can be misleading but
- Full Text can give information where keywords missing
- User can provide additional insight but with keywords you don't have opportunity!

The ACM Computing Classification System (1998)

- A. General Literature
  - A.0 GENERAL
    - *Biographies/autobiographies*
    - *Conference proceedings*
    - *General literary works (e.g., fiction, plays)*
  - A.1 INTRODUCTORY AND SURVEY
  - A.2 REFERENCE (e.g., dictionaries, encyclopedias, glossaries)
  - A.m MISCELLANEOUS
- B. Hardware
  - B.0 GENERAL
  - B.1 CONTROL STRUCTURES AND MICROPROGRAMMING (D.3.2)
    - B.1.0 General
    - B.1.1 Control Design Styles
      - *Hardwired control* [**]
      - *Microprogrammed logic arrays* [**]
      - *Writable control store* [**]
    - B.1.2 Control Structure Performance Analysis and Design Aids
      - *Automatic synthesis* [**]
      - *Formal models* [**]
      - *Simulation* [**]
    - B.1.3 Control Structure Reliability, Testing, and Fault-Tolerance [**] (B.8)
      - *Diagnostics* [**]
      - *Error-checking* [**]
      - *Redundant design* [**]
      - *Test generation* [**]
    - B.1.4 Microprogram Design Aids (D.2.2, D.2.4, D.3.2, D.3.4)
      - *Firmware engineering* [**]
      - *Languages and compilers*
      - *Machine-independent microcode generation* [**]
      - *Optimization*
      - *Verification* [**]
    - B.1.5 Microcode Applications
      - *Direct data manipulation* [**]
      - *Firmware support of operating systems/instruction sets* [**]
      - *Instruction set interpretation*
      - *Peripheral control* [**]
      - *Special-purpose* [**]
    - B.1.m Miscellaneous
- D. Software
  - D.0 GENERAL
  - D.1 PROGRAMMING TECHNIQUES (E)
    - D.1.0 General
    - D.1.1 Applicative (Functional) Programming
    - D.1.2 Automatic Programming (I.2.2)
    - D.1.3 Concurrent Programming
      - *Distributed programming*
      - *Parallel programming*
    - D.1.4 Sequential Programming
    - D.1.5 Object-oriented Programming
    - D.1.6 Logic Programming
    - D.1.7 Visual Programming
    - D.1.m Miscellaneous
  - D.2 SOFTWARE ENGINEERING (K.6.3)
    - D.2.0 General (K.5.1)
      - *Protection mechanisms*
      - *Standards*
    - D.2.1 Requirements/Specifications (D.3.1)
      - *Elicitation methods (e.g., rapid prototyping, interviews, JAD)* NEW!
      - *Languages*
      - *Methodologies (e.g., object-oriented, structured)* Revised
      - *Tools*
    - D.2.2 Design Tools and Techniques Revised
      - *Computer-aided software engineering (CASE)*
      - *Decision tables*
      - *Evolutionary prototyping* NEW!
      - *Flow charts*
      - *Modules and interfaces*
      - *Object-oriented design methods* NEW!
      - *Petri nets*
      - *Programmer workbench* [**]
      - *Software libraries*
      - *State diagrams* NEW!
      - *Structured programming* [**]
      - *Top-down programming* [**]
      - *User interfaces*

Web Search

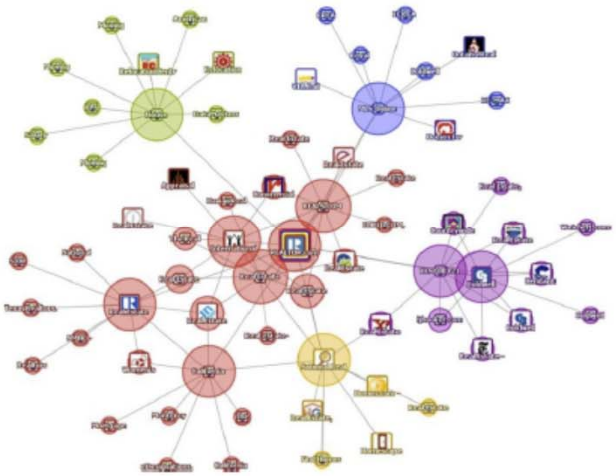
- Very similar to information retrieval
- Main differences:
  - **Links** between Web pages can be exploited
  - **Collecting,** storing, and **updating** documents is more difficult
  - Usually, the **number of users** is very large
  - **Spam** is a problem



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Web Search

- Very similar to information retrieval
- Main differences:
  - **Links** between Web pages can be exploited
  - **Collecting,** storing, and **updating** documents is more difficult
  - Usually, the **number of users** is very large
  - **Spam** is a problem

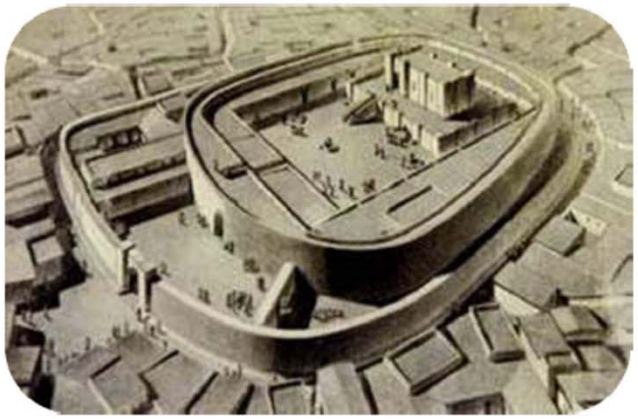


http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Web Search Solution in General starting with history

Ancient Libraries

- Sumerian archives
  - Around 3000–2000 BC
  - About **25,000 clay tablets** stored in temple rooms
  - Mostly inventories and records of commercial transactions



- The Great Library of Alexandria
  - Founded about 300 BC
  - Idea: A **universal library** holding copies of all the world's books
  - At its height, the library held nearly **750,000 scrolls**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Medieval Libraries

- Monastic libraries
  - Educated monks saved many ancient texts from getting lost by **hand-copying**
  - The Vatican Library was formally founded in 1475 but is in fact much older

- Gutenberg's printing press
  - Around 1450, Johannes Gutenberg introduced **movable type** to Europe
  - The technique spread rapidly, copying books became much easier and less expensive

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Modern Libraries

- German National Library
  - **25 million** items
  - Located in Leipzig, Frankfurt (Main), and Berlin
- Library of Congress
  - **150 million** items
  - 20 million new items since 2009
  - The world's **largest library** (according to the Guinness Book)
  - **Classification** system: Library of Congress Classification

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Origin of Semantic Web

Library Catalogs

Items are cataloged by **metadata:**

- **Author/Editor, ISBN,** …
- **Keyword,** e.g. "information retrieval"
- **Subject area,** e.g. "information systems"
- **Specialized classification systems,** e.g. Library of Congress

Titel:	Introduction to information retrieval / Christopher D. Manning; Prabhakar Raghavan; Hinrich Schütze
Verfasser:	Manning, Christopher D. ; Raghavan, Prabhakar ; Schütze, Hinrich
Erschienen:	Cambridge [u.a.] : Cambridge Univ. Press, 2008
Umfang:	XXI, 482 S. : Ill., graph. Darst.
ISBN:	0-521-86571-9, 978-0-521-86571-5*hbk : £ 32.99
Schlagwörter:	*Dokumentverarbeitung / Information Retrieval / Abfrageverarbeitung Text processing (Computer science) Information retrieval Document clustering Semantic Web
Sachgebiete:	54.64 ; Datenbanken 06.74 ; Informationssysteme
Mehr zum Thema:	Klassifikation der Library of Congress: QA76.9.T48 Dewey Dezimal-Klassifikation: 025.04
Link:	http://www.loc.gov/catdir/enhancements/fy0810/2008001257-b.html [Contributor biographical information] http://www.loc.gov/catdir/enhancements/fy0810/2008001257-d.html [Publisher description] http://www.loc.gov/catdir/enhancements/fy0810/2008001257-t.html [Table of contents only] http://www-csli.stanford.edu/~hinrich/information-retrieval-book.html [Companion web site]
Standort:	L all 702
Signatur:	**U 08 B 1668**
Ausleihstatus:	verleihbar Noch nicht verfuegbar

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Full Text Search?

- Catalogue cards are **document proxies**
- Often, they suffice to judge the relevance of a particular item for your information need
- But:
  - A **clever classification scheme** is required:
    - Extensive enough to allow detailed classification
    - Simple enough to be easily understandable
  - **Experts** must catalogue each item individually
- **Problem:** A lot of manual work!
- **Full text search: Every word is a keyword!**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Full Text Search? (2)

- Pre-computer area: **Concordances**
  - Alphabetical list of the **principal words** used in a book
  - Only for works of **special importance,** such as the Bible
  - First Bible concordance by Hugo de Saint Charo, with the help of 500 monks, around 1250

Greek-English Keyword Concordance
© CONCORDANT PUBLISHING CONCERN 1983
All Rights Reserved
Printed in U.S.A.

a tha n a si'a UN-DEATH
immortality. this mortal must put on 1C15[53 54] Christ alone has 1Ti6[16].
immortality, **incorruption**[2], **immutability.** See **immutable.**

a ph'thar t on UN-CORRUPTible
incorruptible. God (men change the glory of) Ro1[23] (King of the eons) 1Ti1[17] wreath F1C 9[25] the dead roused 1C15[52] allotment 1Pt1[4] seed 1Pt1[23] incorruptibility of a meek spirit 1Pt3[4], immortal[1], **incorruptible**[6].

a phthar si'a UN-CORRUPTion
incorruption. to those seeking Ro2[7] saints roused in 1C15[42] allotment of 1C15[50] this corruptible must put on 1C15[53 54] loving Christ in FEp6[24] Christ illuminates 2Ti1[10] (As[1*]Ti2[7] bTi2[7]). immortality[2], incorruption[4], **sincerity**[1].

a kata'lu t on UN-DOWN-LOOSED
indissoluble, the negative of demolish, dissolve. life (Christ) FHb7[16]. endless[1].

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

A Brief History of Web Search

- 1991: **Tim Berners-Lee** "invents" the World Wide Web
- First Web search engines:
  - **Archie:** Query **file names** by regular expressions
  - **Architext/Excite:** Full text search, simple ranking (1993)
- Until 1998, web search meant information retrieval
- 1998: **Google** was founded
  - Exploits **link structure** using the **PageRank** algorithm

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Document

- A **document** is a **coherent passage of free text**
- "Coherent" means: **is about related topics**
- "Free" means: **natural, written language**
- Examples:
  - Newspaper article
  - Scientific article
  - Dictionary entry
  - Web page
  - Email message

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Document Collection

- A **document collection** is a **set of documents**
- Also known as **corpus**
- Usually, all documents within a collection are **similar** with respect to some criterion
- Examples:
  - MEDLINE
  - The articles covered by Google News
  - The Web

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Information Need

- An **information need** is the **topic** about which the user desires to know more
- Refers to an individual, hidden **cognitive state**
- Depends on what the users knows **and** doesn't know
- **Ill-defined**
- Examples:
  - What is the capital of Uganda?
  - Is it really true that McDonald's hamburgers contain worm meat?
  - What is "cloud computing"?

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Query

- A **query** is what the user conveys to the computer in an attempt to **communicate the information need**
- Stated using a **formal query language**
  - Usually a list of search terms
  - But also: "Panda NEAR Jaguar BUT NOT animal"

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Relevance

- A document is **relevant**
  with respect to some user's **information need**

  if

  the user **perceives** it as containing **information of value** with respect to this **information need**
- Usually assumed to be a **binary concept,**
  but could also be graded
- Example:
  - Information need:
    "What is relevance in IR?"
    Relevant document:
    Wikipedia's entry "Relevance (information retrieval)"

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Evolution of Search Engines

- Pre-search: human edited directories
  - High quality, authoritative, reliable
  - But inherently non-scalable and doesn't cater to niches

1994, the *Yahoo!* directory

- First generation – use only "on page", text data
  - Information Retrieval techniques such as TF/IDF, Vector Space model adapted to HTML
  - Scalable, automatic
  - But ranking is poor and easy to spam

1995-1997 AV, Excite, Lycos, etc

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Evolution of Search Engines (2)

Second generation - use off-page, web-specific data

- Link analysis
- Anchor-text (how authors refer to a page when linking to it)
- Click-through data (what results people click on)

From 1998.
Made popular by Google and used by everyone now

- Ranking is much improved, especially for "navigational" queries
- But popularity rules and results are very homogenous
- Spam evolving into a cat-and-mouse game

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Evolution of Search Engines (3)

Third generation – answer the "need behind the query", focusing on the need rather than on the keywords

- Simple context determination
  - Spatial (user location, target location)
  - Query stream (previous queries)
  - Shortcuts via regular expressions
- Integrate multiple sources of data
  - Maps, videos, images, news
- Search assistance tools that help users create good queries and streamline search efforts
  - Spell correction, expand/narrow suggestions, facets
  - From 2006 and on: some of this happens as you type, before the query is actually evaluated (AJAX)

First half of decade: Google, Yahoo!, Ask

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Evolution of Search Engines (3.5)

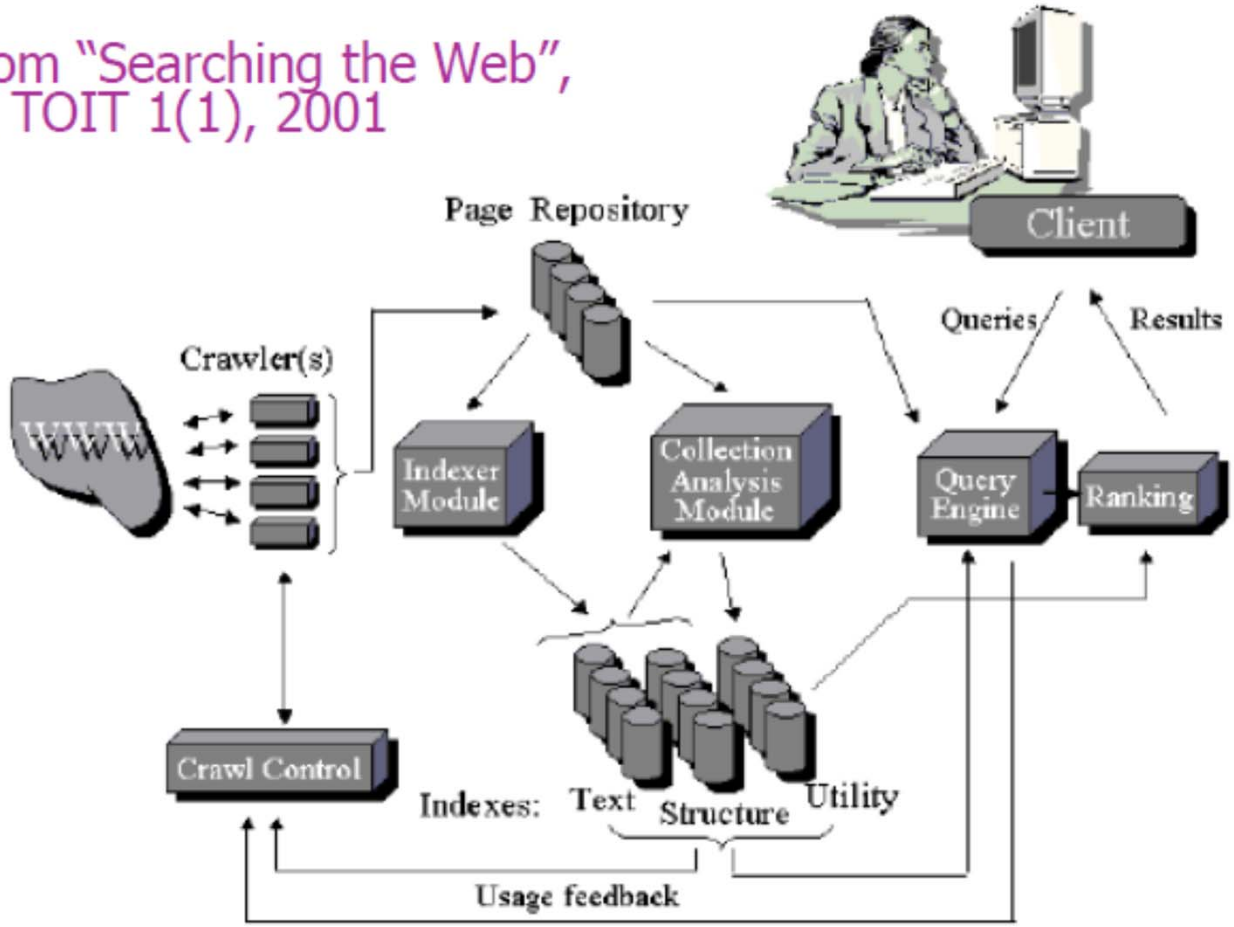
More recent efforts (2005 to date):

- Integration of search and text analysis, e.g. understanding concepts in documents and matching those to queries
  - Named entities, directives/modifiers ("download", "reviews" etc.)
- "Social search" – making use of UGC (user-generated content), social networks, etc.
- Optimizing the experience delivered by the search result page as a whole rather than as a collection of 10 independent links
  - Page real-estate is scarce, need to make every pixel count
  - Emphasis on diversity of results
- More elaborate personalization
- Apps

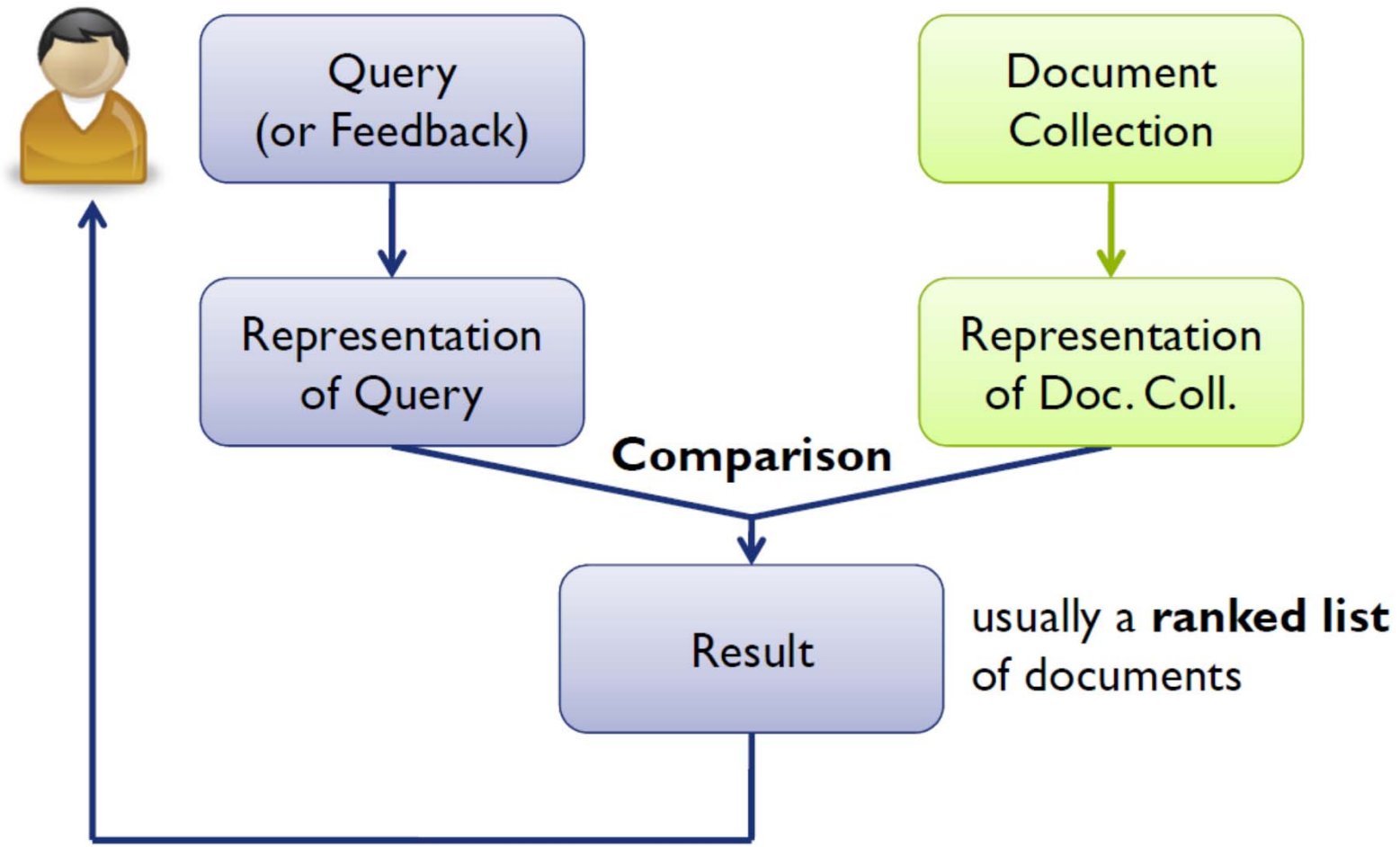
http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Anatomy of a Search Engine

Image taken from "Searching the Web", Arasu et al., TOIT 1(1), 2001



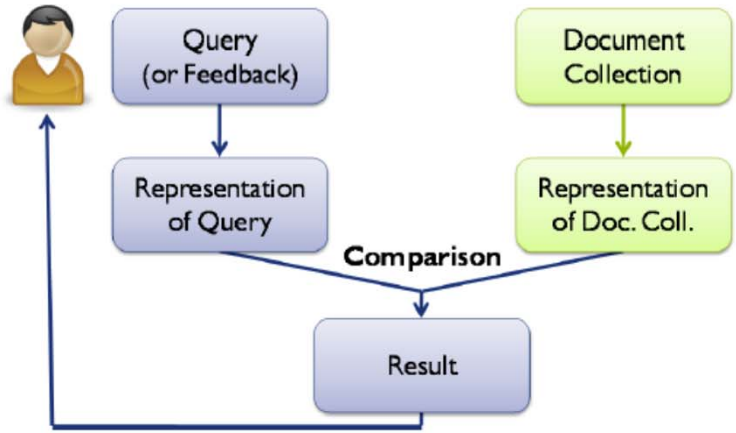
Schematic Diagram of an IR System



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

IR Models

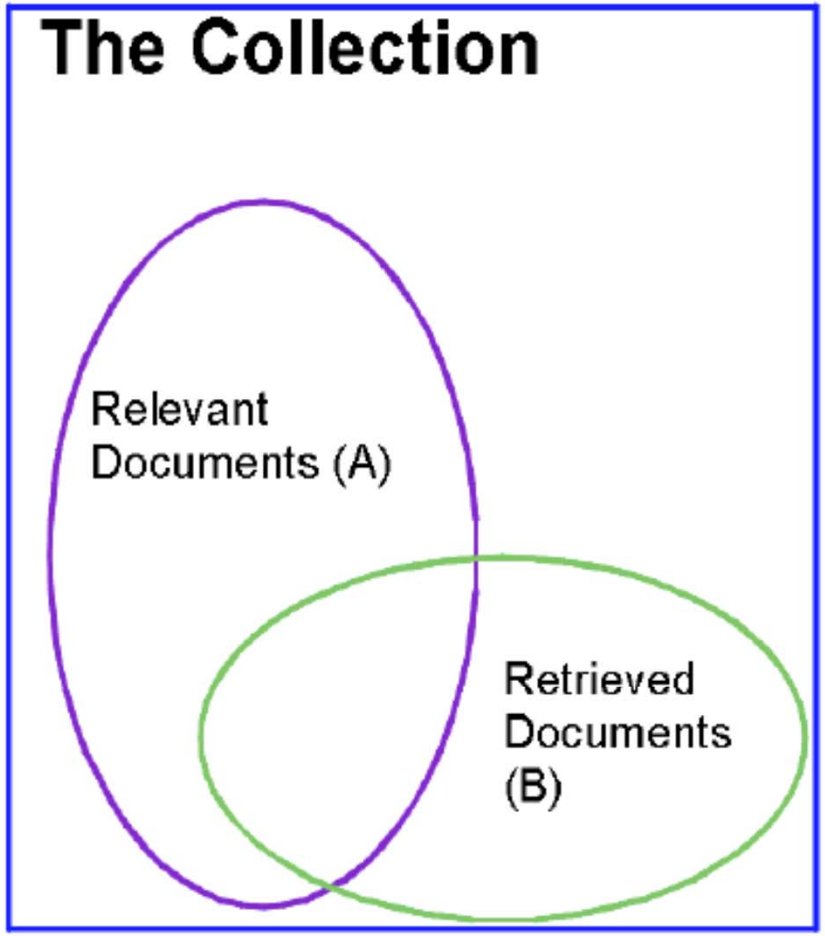
- Any IR system is based on an **IR model**
- The model defines …
  - … a **query language,**
  - … an internal **representation of queries,**
  - … an internal **representation of documents,**
  - … a **ranking function** which associates a real number with each query–document pair.
- Optional: A mechanism for **relevance feedback**



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

"Classic" Information Retrieval



- Given a query, the system retrieves a set B of documents
- Every retrieved document is either relevant or irrelevant to the query

Quality metrics:

- *Recall* : $(A \cap B) / A$
- *Precision* : $(A \cap B) / B$

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Recall and Precision on the Web

- Relevance of document to queries is not binary – there are many shades of gray
- Broad-topic queries:
  - abundance problem
  - Precision is the dominating factor: users mostly satisfied with a few good results (a few *authoritative* pages)
- Narrow-topic queries:
  - Find a needle in an enormous haystack
  - Recall demands engines cover significant portions of the Web
- Common measure: precision@10
- Nowadays larger emphasis on *diversity*

precision@10 is precision over top 10 ranked documents

Diversity of Search Results

New Google Update "Improves Diversity" Of Domains In Results

Sep 14, 2012 at 5:03pm ET by Danny Sullivan

Like 117 Tweet 993 +1 92 Share 110 6

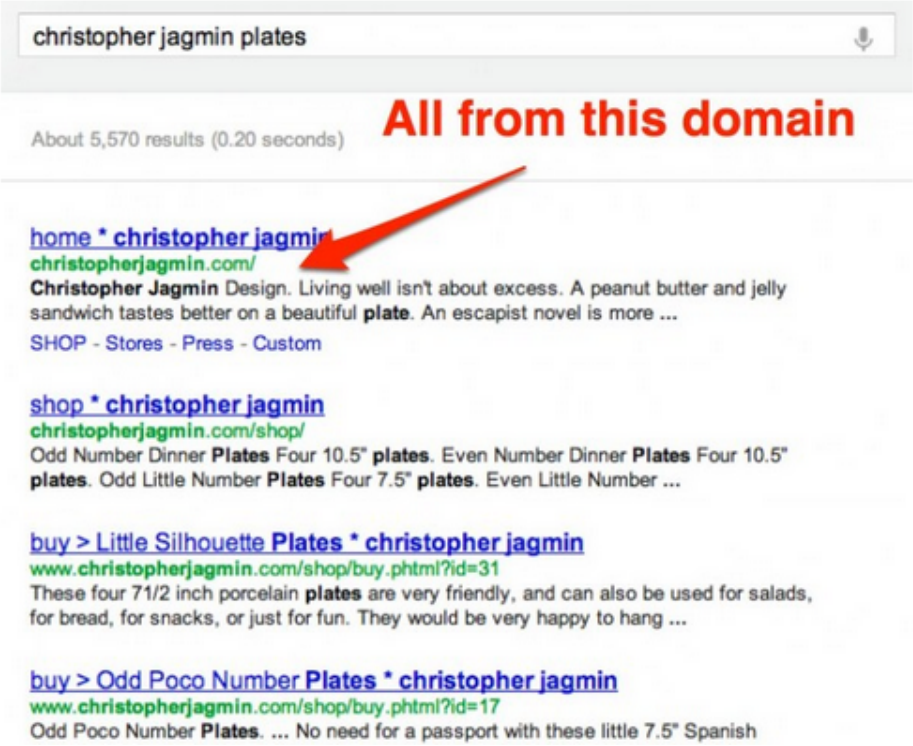
There's been chatter about Google having done some type of algorithm update this week that's having an impact on rankings. Now it's finally confirmed, a new change meant to allow more domains to appear in search results.

The head of Google's spam fighting team Matt Cutts tweeted this about it:

> Just fyi, we rolled out a small algo change this week that improves the diversity of search results in terms of different domains returned.

Lately, more and more people have been noting that Google's search results can sometimes be dominated by pages that all come from the same domain. In other words, do a search, and all the listings seem to come from the same web site.

Here's an example of this from last month, when we wrote about the problem:



Information Retrieval (Web Search) Technology

The Bag of Words Representation

- A very popular **representation of documents** is the **bag of words model**
- Each document is represented by a bag (= multiset) of **terms** from a predefined **vocabulary**
- Standard case:
  - Vocabulary
    = set of all the words occurring in the collection's documents
  - Each document is represented by the words it contains

That's one small step for a man, a giant leap for mankind → { that's, one, small, step, for (2), a (2), man, giant, leap, mankind }

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

The Bag of Words Model (2)

- Cons:
  - Word order gets lost
  - Very different documents could have similar representations
  - Document structure (e.g. headings) and metadata is ignored

- Pros:
  - Simple set-theoretic representation of documents
  - Efficient storage and retrieval of individual terms
  - IR models using the bag of words representation work well!

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

The Bag of Words Model (3)

- Any document can be represented by an **incidence vector:**

vocabulary (aka index terms) →

that's	one	small	step	for	a	man	giant	leap	mankind	taikonaut	Zhai's	is	China
1	1	1	1	2	2	1	1	1	1	0	0	0	0
0	0	1	1	0	1	0	1	1	0	1	1	1	1

That's one small step for a man, a giant leap for mankind

Taikonaut Zhai's small step is a giant leap for China

incidence matrix (aka term-document matrix)

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Boolean Query

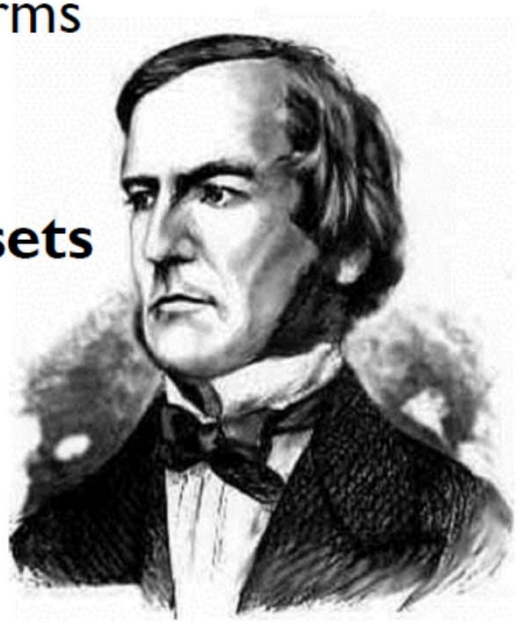
http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

The Boolean Model

- Simple model based on set theory
- Queries are specified as Boolean expressions.
  - indexing units are words
  - Boolean Operator: OR, AND, NOT
  - Example: q ="java" AND "compilers" AND ("unix" OR "linux")
- Relevance: A document is considered relevant to the query if it satisfies the query Boolean expression.
- Pros:
  - Fast (bitmap vector operations)
  - Binary decision (doc is "relevant" or not)
- Cons:
  - Binary decision - What about ranking?
  - Who speaks Boolean?

Boolean Retrieval

- The simplest (and arguably oldest) IR model
- Documents = **sets** of words (index terms)
- Query language
  = **Boolean expressions** over index terms
- Binary ranking function, i.e. 0/1-valued
- Retrieval is based on **membership in sets**
  - "Find all documents indexed by the word 'mankind'!"
  - "Find all documents indexed by the word 'man' or 'mankind'!"



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Example

- $Document_1$ = {step, mankind}
- $Document_2$ = {step, China}

- $Query_1$ = "step AND mankind"
  - Result set: {$Document_1$}

- $Query_2$ = "step OR mankind"
  - Result set: {$Document_1$, $Document_2$}

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

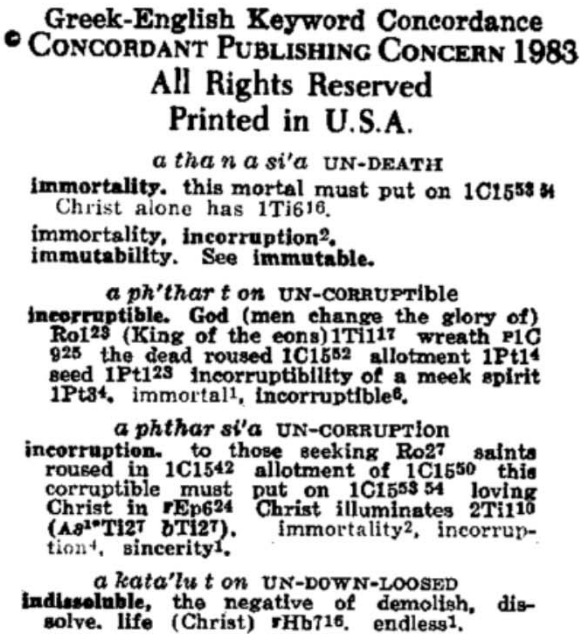
Boolean Queries in Practice

- **Warning:**
  Exclusive use of negation will result in large result sets!
  - $Query_3$ = "NOT mankind"
- To match natural language better,
  "BUT NOT" can be used instead of "AND NOT"
  - $Query_4$ = "step BUT NOT China"
- Use **"OF"** to search for subsets of a given size:
  - $Query_5$ = "2 of {step, mankind, China}"
  - $Query_5 \equiv$ "(step AND mankind)
    OR (step AND China)
    OR (mankind AND China)"

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Query Processing

- Usually, documents are indexed by an **inverted index**
  - For each index term, the set of documents containing this term is **pre-computed** and stored on disk
  - This enables **fast query processing**
- Document collection:
  - Document_1 = {step, mankind}
  - Document_2 = {step, China}
- Inverted index:
  - step: {Document_1, Document_2}
  - mankind: {Document_1}
  - China: {Document_2}


Greek-English Keyword Concordance
© CONCORDANT PUBLISHING CONCERN 1983
All Rights Reserved
Printed in U.S.A.

a tha n a si'a UN-DEATH
immortality. this mortal must put on 1C15[53 54] Christ alone has 1Ti6[16].
immortality, **incorruption**[2], **immutability.** See **immutable.**

a ph'thar t on UN-CORRUPTible
incorruptible. God (men change the glory of) Ro1[23] (King of the eons) 1Ti1[17] wreath PIC 9[25] the dead roused 1C15[52] allotment 1Pt1[4] seed 1Pt1[23] incorruptibility of a meek spirit 1Pt3[4], immortal[1], **incorruptible**[6].

a phthar si'a UN-CORRUPTion
incorruption. to those seeking Ro2[7] saints roused in 1C15[42] allotment of 1C15[50] this corruptible must put on 1C15[53 54] loving Christ in PEp6[24] Christ illuminates 2Ti1[10] (As[1a]T12[7] bTi2[7]). immortality[2], incorruption[4], sincerity[1].

a kata'lu t on UN-DOWN-LOOSED
indissoluble, the negative of demolish, dissolve. life (Christ) PHb7[16]. endless[1].

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Query Processing (2)

- Thanks to the inverted index, queries of the type
  "Show me all documents containing term *X*"
  can be answered quickly
- Also quick to compute: unions and intersections of sets
- Example:
  - result of "mankind AND step"
    = (result of "mankind") ∩ (result of "step")
  - result of "mankind OR step"
    = (result of "mankind") ∪ (result of "step")
- **Idea:** Convert all queries to **conjunctive normal form** or **disjunctive normal form**

∩
∪

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Pros

- Simple query paradigm, **easy to understand**
- If all document representations are mutually distinct, **any possible subset of documents can be retrieved** by a suitable query

  ⇒ cut out the set of relevant documents
- But: This advantage is rather **theoretical,** since the "right" query usually is unknown

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Cons

- A binary ranking function returns a **set of results,** i.e. it is **unordered**
- Controlling the **result size** is difficult
- **Similarity queries** are not supported
- Usually, most of the documents found are relevant; but **many relevant documents are not found**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Fuzzy Index

Previous Lessons

- Boolean retrieval:
  - **Documents:** Sets of words (index terms)
  - **Queries:** Propositional formulas
  - **Result:** The set of documents satisfying the query formula
  - **Example:**

    $Document_1$ = {step, mankind, man}
    $Document_2$ = {step, China, taikonaut}
    $Document_3$ = {step, China, mountaineer}

    Query = "step AND ((China AND taikonaut) OR man)"

    Result = {$Document_1$, $Document_2$}

Fuzzy Index Terms

- **Observation:**
  Not all index terms representing a document are equally important, or equally characteristic
  - Are there any synonyms to the document's terms?
  - Does a term occur more than once in the document?
- Can we assign **weights** to terms in documents?
- **Idea:**
  Improve Boolean retrieval!
  Describe documents by **fuzzy sets** of terms!
  - No binary set membership, but **graded membership!**
  - **Advantage:** Fuzzy (i.e. ordered!) results sets

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Fuzzy Index Terms (3)

- **Jaccard index:**
  - Measures which terms **co-occur** in the document collection
  - The Jaccard index $c(t, u)$ of the term pair (t, u) is

$$\frac{\text{\#documents containing } \textbf{both} \text{ term } t \text{ and term } u}{\text{\#documents containing } \textbf{at least one} \text{ of term } t \text{ and term } u}$$

  - Also known as **term-term correlation coefficient,** although it is not a correlation in the usual sense
    - A usual correlation coefficient would be high, if most documents do not contain any of the two terms

But here we can get low correlation coefficient even if 99% of documents contain neither t nor u

Fuzzy Index Terms (4)

- **Jaccard index:**
  - Document_1 = {step, man, mankind}
  - Document_2 = {step, man, China}
  - Document_3 = {step, mankind}

c(*t*, *u*)	step	man	mankind	China
step	1	0.67	0.67	0.33
man		1	0.33	0.5
mankind			1	0
China				1

$$\frac{\text{\#documents containing } \textbf{both} \text{ term } t \text{ and term } u}{\text{\#documents containing } \textbf{at least one} \text{ of term } t \text{ and term } u}$$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Fuzzy Index Terms (5)

- Ogawa *et al.* (1991) compute the fuzzy index terms as follows:
  - The **fuzzy membership degree** of term t with respect to document D (represented as crisp set of terms) is

$$W(D, t) = 1 - \prod_{u \in D} (1 - c(t, u))$$

  - $1 - c(t, u)$ is the fraction of documents containing one of term t and term u but not both
  - $\boldsymbol{t \in D}$ **implies** $\boldsymbol{W(D, t) = 1}$
  - **Idea:** Give terms a high fuzzy membership degree that usually occur together with the other document terms; those terms will capture the document's topic best

Example

- Document_1 = {step, man, mankind}
- Document_2 = {step, man, China}
- Document_3 = {step, mankind}

$W(D, t)$	step	man	mankind	China
Document_1	1	1	1	0.67
Document_2	1	1	0.78	1
Document_3	1	0.78	1	0.33

$c(t, u)$	step	man	mankind	China
step	1	0.67	0.67	0.33
man		1	0.33	0.5
mankind			1	0
China				1

$$W(D, t) = 1 - \prod_{u \in D} (1 - c(t, u))$$

Fuzzy Retrieval Model

- Cons:
  - Computation of fuzzy membership weights usually is difficult
    - Main problem: All weights must be within [0, 1]
  - Lack of intuitive query processing
    - But: There are many other ways to define fuzzy conjunction and disjunction (using **t-norms** and **t-conorms**)

- Pros:
  - Supports non-binary assignment of index terms to documents
    - It is possible to find relevant documents that do not satisfy the query in a strict Boolean sense
  - Ranked result sets

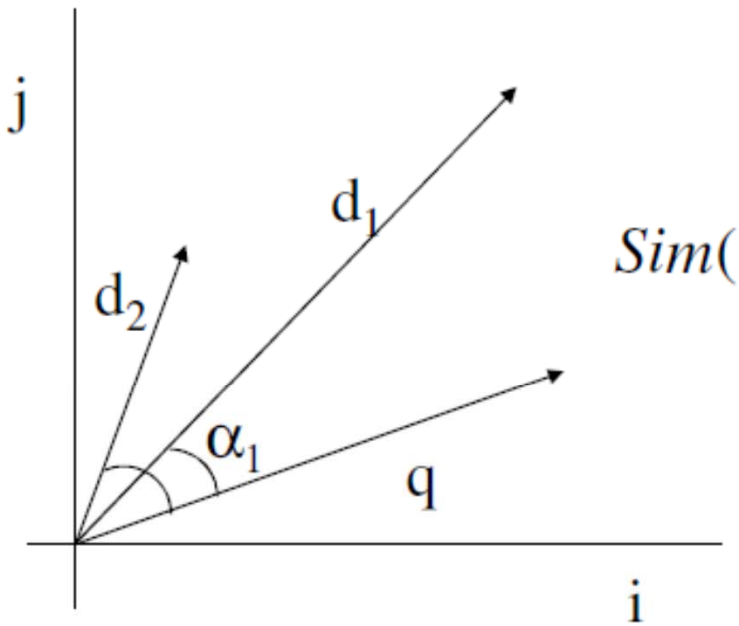
http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Vector Space Model

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Vector Space Model

- Documents and queries are both represented as vectors in a high-dimensional vector space in which each word is a dimension
- "Relevance" is measured by similarity, i.e. by the cosine of the angle between doc-vectors and the query-vector



$$Sim(q,d) = \frac{q \bullet d}{|q| \bullet |d|} = \frac{\sum_i w_{iq} w_{id}}{\sqrt{\sum_i w_{iq}^2 \sum_i w_{id}^2}}$$

Information Spaces

- **Spatial structure of libraries:**
  Topically related books
  are standing side by side

- Can we transfer this principle
  to information retrieval?
- **Idea:**
  Represent documents and queries
  as points in an **abstract semantic space**
  - Measure **similarity** by **proximity**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

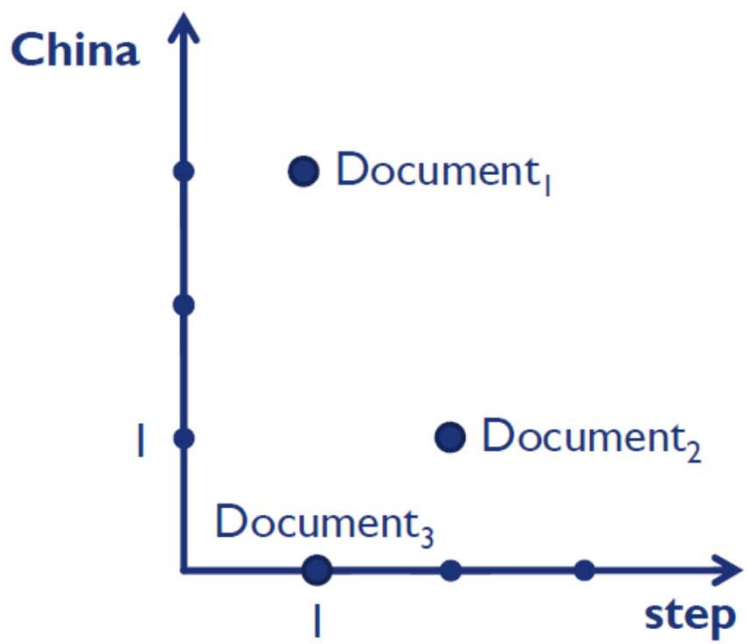
Vector Space Model

- The **vector space model** was proposed by Gerard Salton (Salton, 1975)
- Documents and queries are represented as point in ***n*-dimensional real vector space $\mathbb{R}^n$,** where *n* is the size of the index vocabulary
  - Usually, ***n* is very large:** 500,000 terms (at least)
- Each index term spans its own dimension
- Obvious first choice:
  Represent documents by its **incidence vectors**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Example

- $Document_1$ = {step, China/3}
  $Document_2$ = {step/2, China}
  $Document_3$ = {step}



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Distance and Similarity

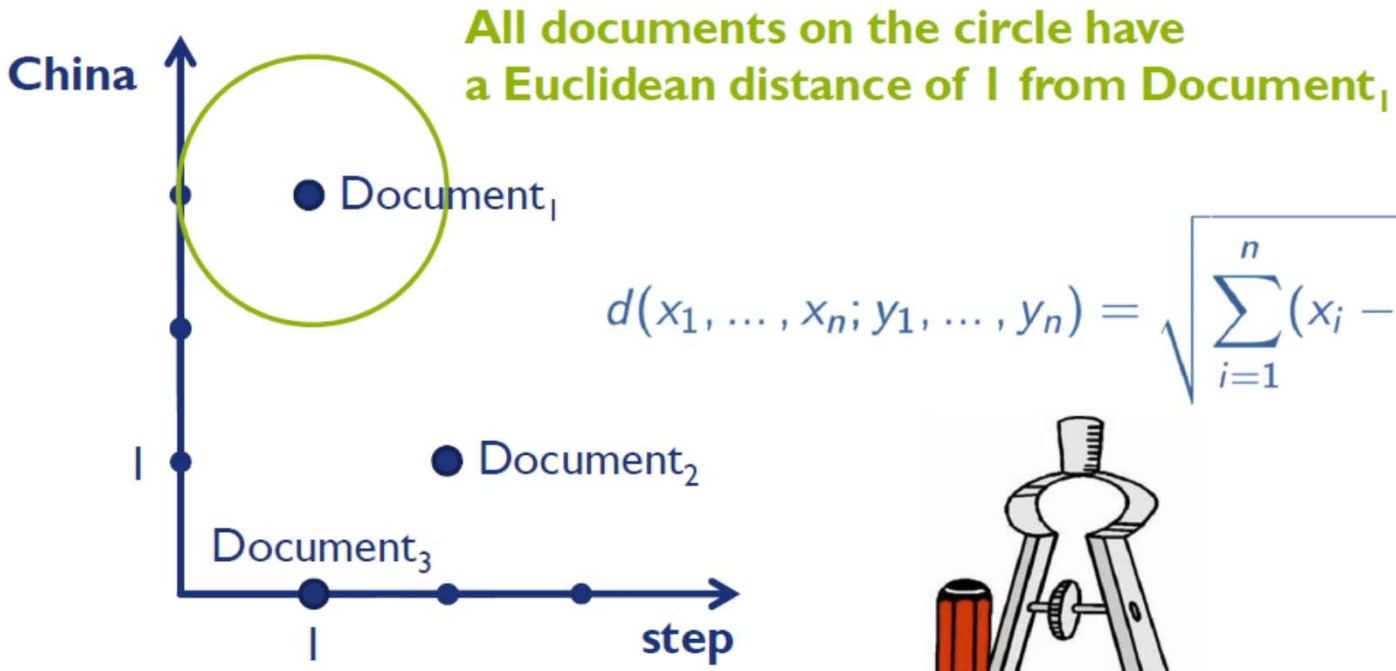
- **How to define similarity/proximity?**
- A **metric** on a set X is a function $d : X \times X \rightarrow \mathbb{R}$ having the following properties:
  - $d(x, y) \geq 0$, for any $x, y \in X$ (non-negativity)
  - $d(x, y) = 0$ iff $x = y$, for any $x, y \in X$ (identity)
  - $d(x, y) = d(y, x)$, for any $x, y \in X$ (symmetry)
  - $d(x, z) \leq d(x, y) + d(y, z)$, for any $x, y, z \in X$ (triangle inequality)
- Example: **Euclidean distance**

$$d(x_1, \ldots, x_n; y_1, \ldots, y_n) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Euclidean Distance

- **Geometric meaning** of Euclidean distance:



$$d(x_1, \ldots, x_n; y_1, \ldots, y_n) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Similarity 1-distance

- A **similarity measure** on a set X is a function $s : X \times X \rightarrow [0, 1]$ where
  - $s(x, y) = 1$ means that x and y are **maximally similar**
  - $s(x, y) = 0$ means that x and y are **maximally dissimilar**
- There is no general agreement on what additional properties a similarity measure should possess
- Example: **Cosine similarity** in vector spaces

$$s(x, y) = \cos(\alpha)$$

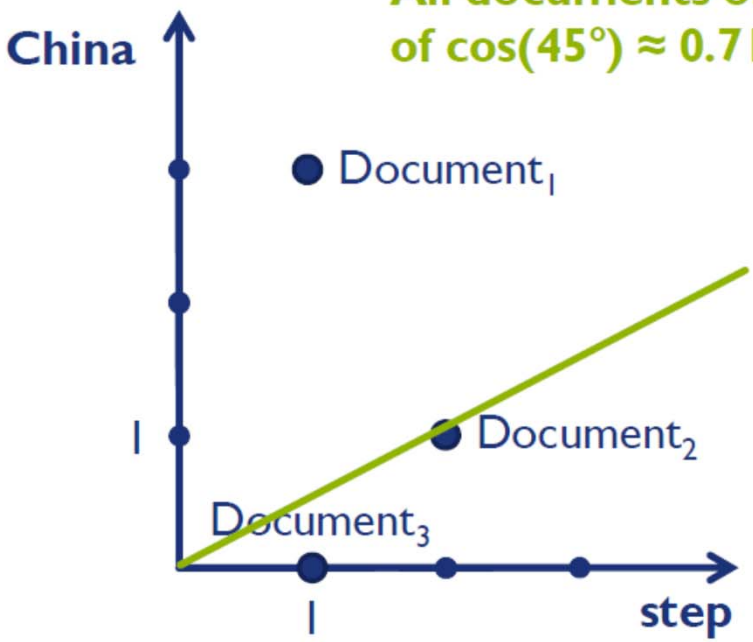
  - α is the angle between these two vectors:
    - The vector pointing from the origin to x
    - The vector pointing from the origin to y

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Cosine Similarity

- **Geometric meaning** of cosine similarity:

All documents on the line have a cosine similarity of cos(45°) ≈ 0.71 to Document$_1$



$s(x, y) = \cos(\alpha)$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Cosine Similarity (2)

- How to **compute the angle α** between two vectors?

$$\cos(\alpha) = \frac{x \cdot y}{\|x\| \cdot \|y\|}$$

- "·" denotes the **dot product** (aka scalar product), i.e.

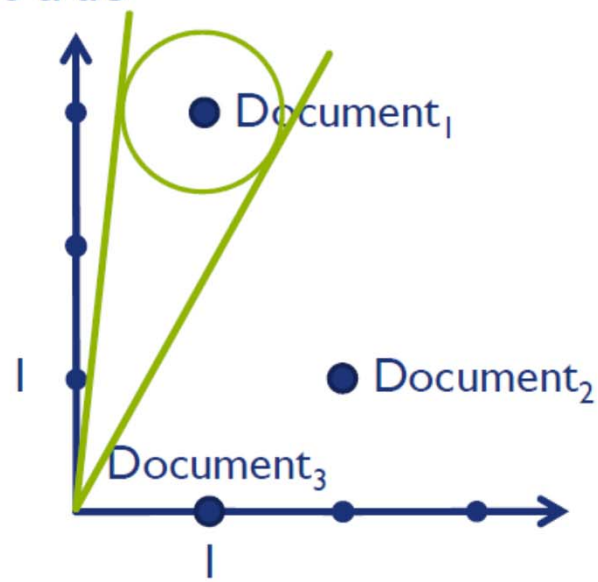
$$x \cdot y = \sum_{i=1}^{n} x_i \cdot y_i$$

- "‖·‖" denotes the **Euclidean norm** (aka ℓ^2-norm), i.e.

$$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

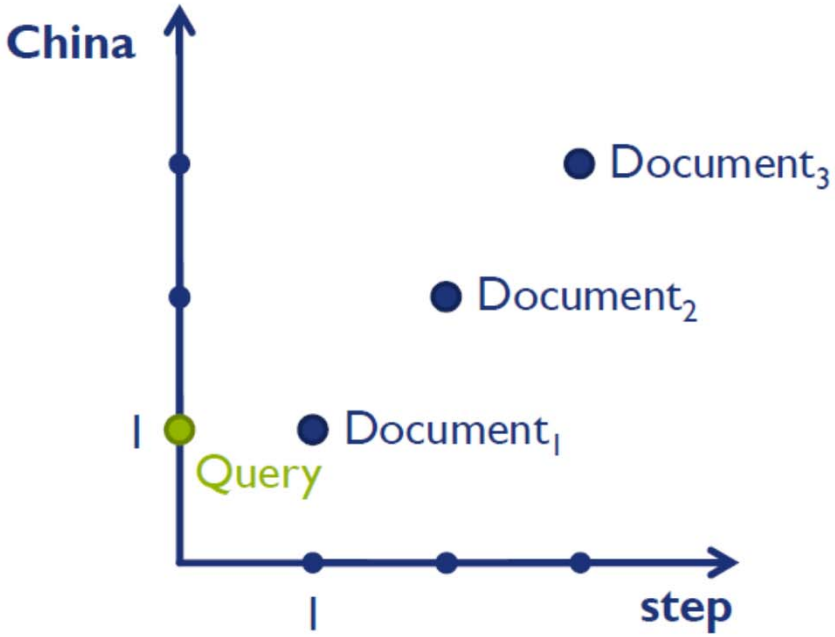
The "Right" Measure

- **Be careful!**
  - The choice of distance or similarity measure always depends on the current application!
- Different measures often behave similar, but not always …
  - **Low Euclidean distance implies high cosine similarity,** the converse is not true



Normalization

- Cosine similarity does not depend on the **length of document and query vectors**
- But using other measures, this might make a difference …



Using e.g. Euclidean distance, are shorter documents more similar to the query than longer ones?

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Normalization (2)

- There are many ways to **normalize** the vector representation of documents and queries
- Most popular:
  - Divide each coordinate by the vector's length, i.e. **normalize to length 1:**

$$\frac{x}{\|x\|}$$

  - Divide each coordinate by the vector's largest coordinate:

$$\frac{x}{\max_{i=1}^{n} x_i}$$

  - Divide each coordinate by the sum the vector's coordinates:

$$\frac{x}{\sum_{i=1}^{n} x_i}$$

Normalization (4)

- Often, longer documents cover a topic more in-depth
- Therefore, **accounting for document length might be reasonable**
  - There are several strategies how this can be done
  - Straightforward:
    1. Compute query result on normalized documents and query
    2. **Give long documents a small boost** proportional to their length (maybe you should apply a dampening factor to account for extremely large documents)
  - More advanced:
    - **Measure the effect of document length on relevance** within your current document collection
    - Adjust the ranking according to these insights

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Vector Representation

- Are there any **more advanced ways of representing documents in vector space** than just copying their bag of words representation?
- **Of course!**
- Luhn's observation (1961):
  Repetition of words is an indication of emphasis
  - We are already exploiting this by using the bag of words model!
  - The number of occurrences of a term in a document or query is called its **"term frequency"**
  - **Notation:**
    tf(*d*, *t*) is the term frequency of term *t* in document *d*

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Vector Representation (2)

- **Discrimination:**
  - Not every term in a collection is equally important
  - For example, the term **"psychology"** might be highly **discriminating** in a **computer science corpus;** in a **psychology corpus,** it doesn't carry much information
  - Denote the **discriminative power** of a term *t* by **disc(*t*)**
  - There are many ways to formalize discriminative power …
- General term weighting framework:
  - Higher term frequency ⇒ Higher term weight
  - Higher discriminative power ⇒ Higher term weight
- **Term weight should be proportional to**

$$\mathbf{tf}(d, t) \cdot \mathbf{disc}(t)$$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

TF-IDF

- **Karen Spärck Jones** observed that, from a discrimination point of view, what we'd really like to know is a term's **specificity** (Spärck Jones, 1972):
  - **In how many documents a given term is contained?**
  - The term specificity is negatively correlated with this number!
  - **The more specific a term is, the larger its discriminative power is**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

TF-IDF (2)

- The number of documents containing a given term *t* is called *t*'s **document frequency,** denoted by **df(*t*)**
- Karen Spärck Jones proposed the **TF-IDF term weighting scheme:**
  - Define the weight of term *t* in document *d* as:

$$\mathrm{tf}(d, t) \cdot \frac{1}{\mathrm{df}(t)}$$

  - "IDF" = **"inverse document frequency"**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

TF-IDF (3)

- Spärck Jones: The relationship between specificity and inverse document frequency is **logarithmic!**
- This leads to today's most common form of TF-IDF, as proposed by Robertson and Spärck Jones (1976):

$$\mathrm{tf}(d, t) \cdot \log\left(\frac{N + 0.5}{\mathrm{df}(t) + 0.5}\right)$$

  - N is the number documents in the collection
  - "+ 0.5" accounts for very frequent and very rare terms
  - "N / df(t)" normalizes with respect to the collection size

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Vector Space Model: Pros

- **Pros:**
  - Simple and clear
  - **Intuitive querying** yields high usability
  - Founded on "real" document rankings, not based on result sets
  - **Highly customizable** and adaptable to specific collections:
    - Distance/similarity functions
    - Normalization schemes
    - Methods for term weighting
  - **High retrieval quality**
  - Relevance feedback possible (will be covered soon…)

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Vector Space Model: Cons

- **Cons:**
  - High-dimensional vector spaces, specialized algorithms are required (next lectures…)
  - Relies on **implicit assumptions,** which do not hold in general:
    - **Cluster hypothesis:**
      "Closely associated documents tend to be relevant with respect to the same queries"
    - **Independence/orthogonality assumption:**
      "Whether a term occurs in a document, is independent of other terms occurring in the same document"

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Probabilistic Models

http://webcourse.cs.technion.ac.il/236621/Winter2011-2012/en/ho_Lectures.html

Probabilistic Retrieval Models

- Attempt to predict the probability that a given document is "relevant" to a given query: Pr(R|d,q)
- Rank retrieved documents according to this probability of relevance
  - Probability Ranking Principle: "if a retrieval system's response to each request is a ranking of the documents in decreasing probability of usefulness to the user … then the overall effectiveness of the system to its users will be the best" - **Stephen E. Robertson, *J. Documentation* 1977**
- Rely on accurate estimates of probabilities

Probabilistic Retrieval Models

- Probabilistic IR models use

Pr(document *d* is useful for the user asking query *q*)

as underlying **measure of similarity** between queries and documents

- Advantages:
  - Probability theory is the right tool to reason under **uncertainty** in a formal way
  - **Methods** from probability theory can be **re-used**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Usefulness and Relevance

- Characterizing **usefulness** is really tricky, we will discuss this later…
- Instead of usefulness we will consider **relevance**
- Given a **document representation** *d* and a **query** *q*, one can **objectively** determine whether *d* is relevant with respect to *q* or not
- This means in particular:
  - Relevance is a **binary** concept
  - Two documents having the same representation are either **both relevant** or **both irrelevant**

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Relevance

- Denote the set of **all document representations** contained in our current collection by $\boldsymbol{C}$
- For any query q, denote the set of **relevant documents** contained in our collection by $\boldsymbol{R_q}$, i.e.

$$R_q = \{d \in C \mid d \text{ is relevant with respect to } q\}$$

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Probability of Relevance

- **Our task then becomes:**
  - **Input:** The user's query q and a document d
  - **Output:** $\Pr(d \in R_q)$
- Precisely what does **this probability** mean?
  - As we have defined it,
    it is either $d \in R_q$ or $d \notin R_q$
  - Is $\Pr(d \in R_q)$ a sensible concept?

- What does **probability in general** mean?
  - Maybe we should deal with that first…

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequency v Bayes

Interpretations of Probability Detour

- There are different interpretations of "probability," we will look at the two most common ones
- **Frequentists vs. Bayesians**
- Frequentists
  - Probability = expected **frequency** on the long run
  - Neyman, Pearson, Wald, …
- Bayesians:
  - Probability = **degree of belief**
  - Bayes, Laplace, de Finetti, …

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

The Frequency Interpretation Detour

- An event can be assigned a probability only if
  1. it is based on a **repeatable(!) random experiment,** and
  2. within this experiment, the event occurs at a persistent rate on the long run, its **relative frequency**
- An event's probability is the **limit of its relative frequency** in a large number of trials
- Examples:
  - Events in dice rolling
  - The probability that it will rain tomorrow in the whether forecast (if based on historically collected data)

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

The Bayesian Interpretation

Detour

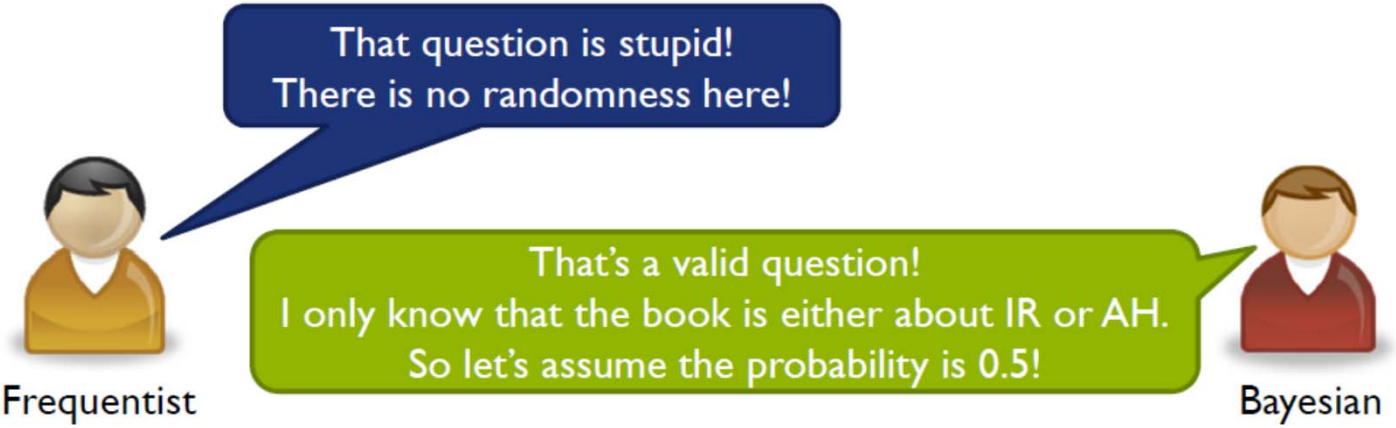
- Probability is the **degree of belief** in a proposition
- The belief can be:
  - **subjective,** i.e. personal, or
  - **objective,** i.e. justified by rational thought
- Unknown quantities are treated probabilistically
- Knowledge can always be updated
- Named after Thomas Bayes
- Examples:
  - The probability that there is life on other planets
  - The probability that **you** pass this course's exam

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequentist vs. Bayesian

Detour

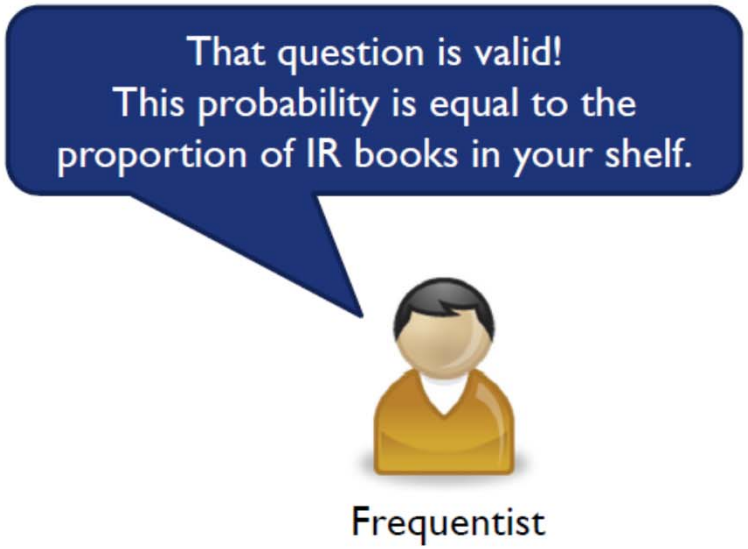
- There is a book lying on my desk
- I know it is about one of the following two topics:
  - Information retrieval
  - Animal health
- What's Pr("the book is about IR")?



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequentist vs. Bayesian (2) Detour

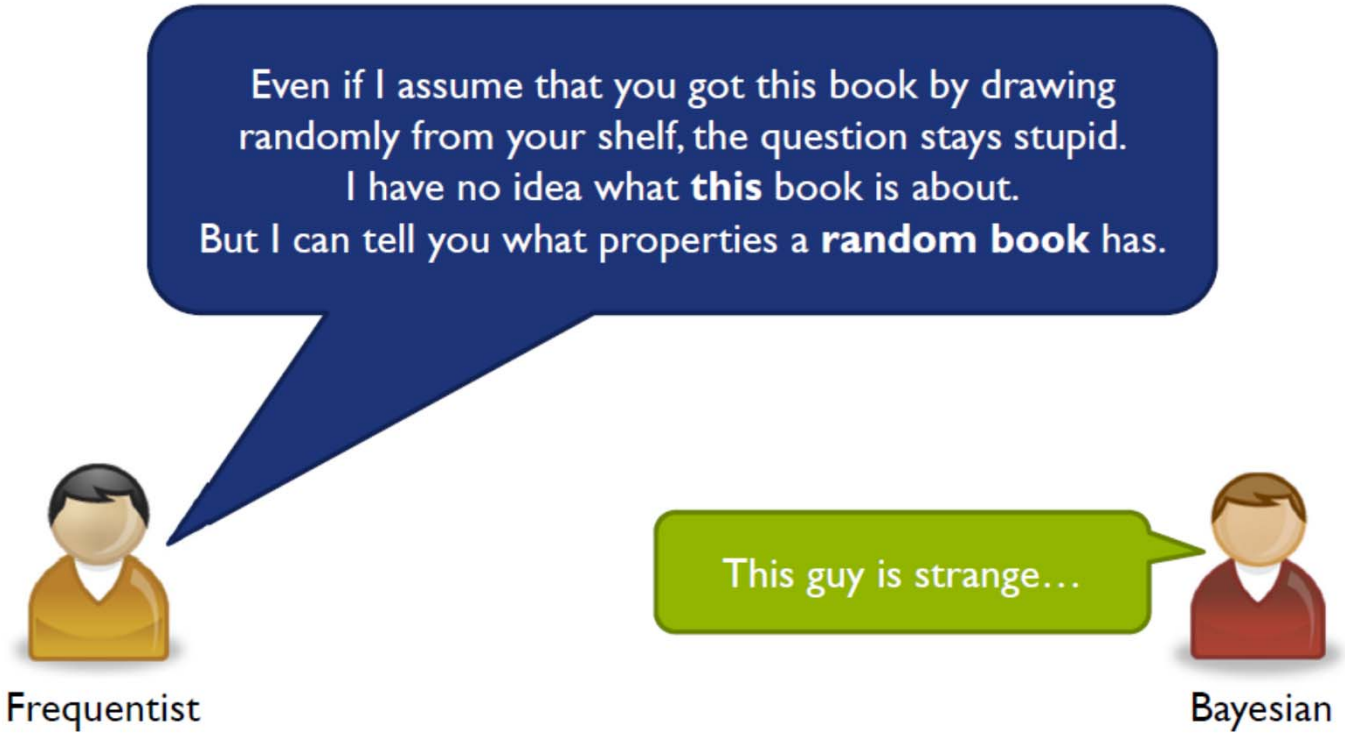
- But: Let's assume that the book is lying on my desk due to a **random draw** from my bookshelf…
- Let *X* be the "topic result" of a random draw
- What's Pr("*X* is about IR")



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequentist vs. Bayesian (3)

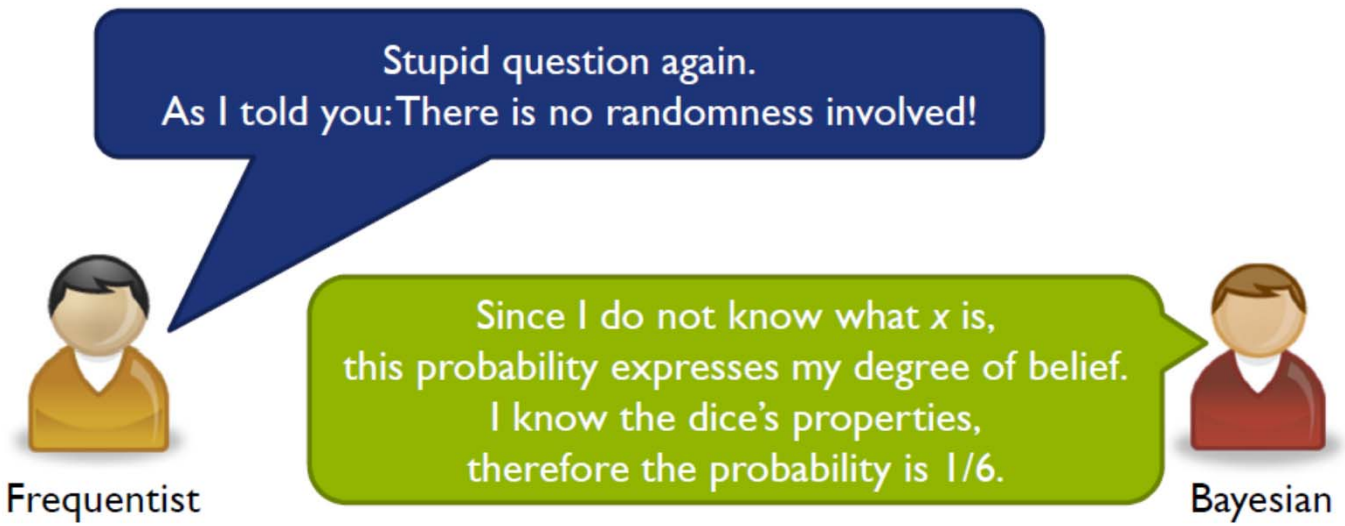
- Back to the book lying on my desk
- What's Pr("the book is about IR")?



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequentist vs. Bayesian (4) Detour

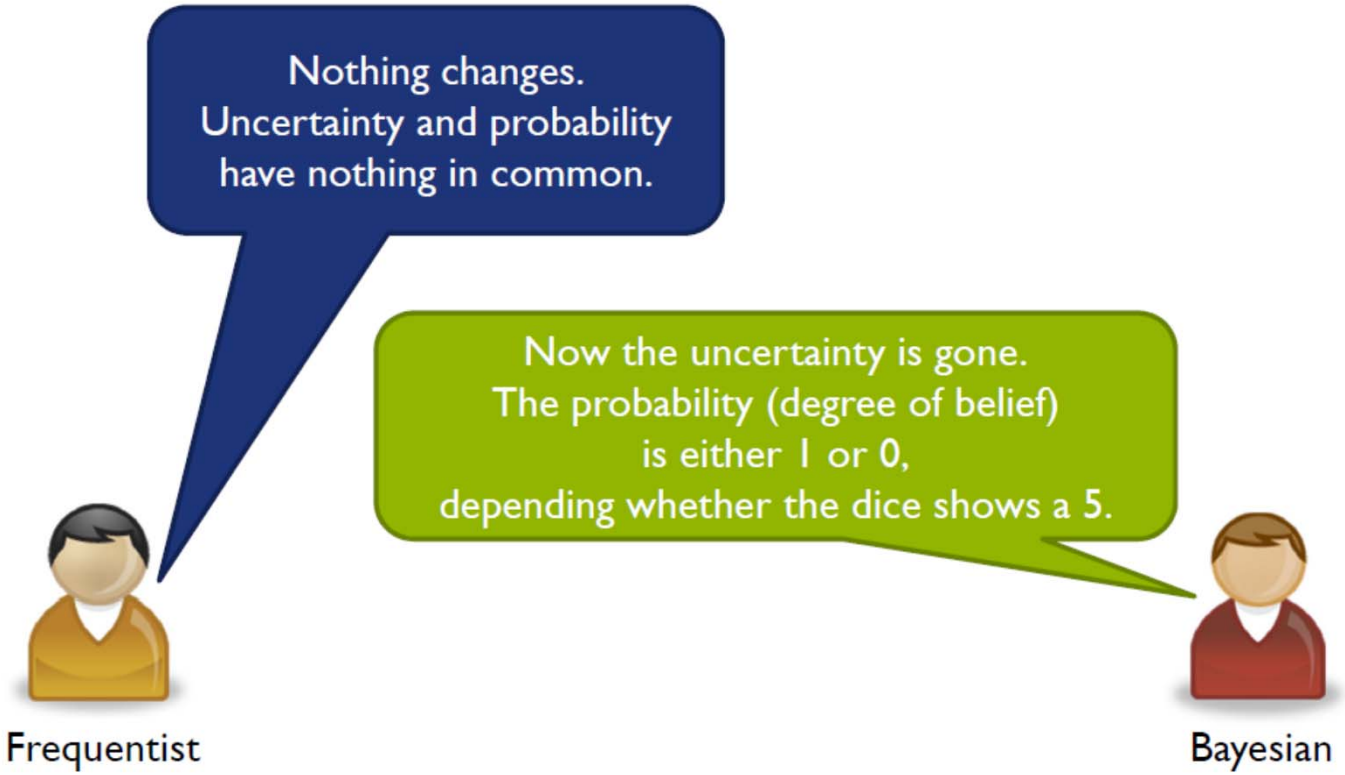
- A more practical example: Rolling a dice
- Let x be the (yet hidden) number on the dice that lies on the table
  Note: ***x* is a number,** not a random variable!
- What's $\Pr(x = 5)$?



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Frequentist vs. Bayesian (5) Detour

- What changes if I show you the value of *x*?



http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Bayes is most useful

- Only do a Physics experiment once
- Read recent news about “Belief in Higgs increasing”
- Not about frequency of experiments seeing Higgs

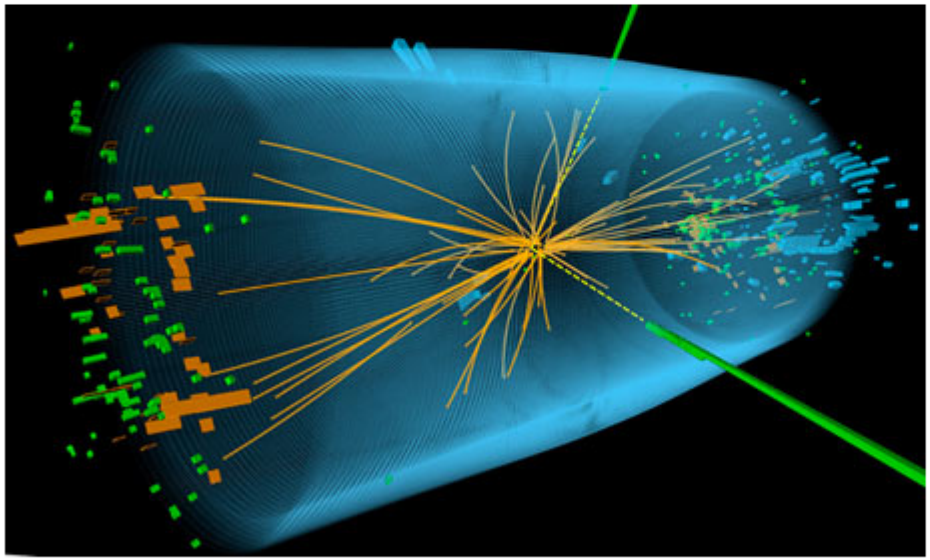
Cern scientists believe newly discovered particle is the real Higgs boson

Results of analysis at Cern in Switzerland show particle behaves precisely as expected

Share 1097
Tweet 190
+1 105
Share 26
Email

Ian Sample, science correspondent
The Guardian, Thursday 14 March 2013

Article history



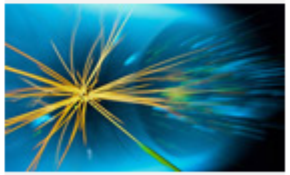
A representation of traces of a proton-proton collision issued by Cern as part of its efforts to find the elusive Higgs boson. New data strengthens the belief that a subatomic particle discovered last year is the Higgs boson, European scientists said. Photograph: Cern/AFP/Getty Images

Science
Higgs boson · Cern · Particle physics · Physics

World news
Switzerland · Europe

More news

More on this story



Editorial: In praise of Higgs boson
Editorial: The vanishingly small speck whose spin has been confirmed as being Higgs-like ought

Probability of Relevance, Again

- How to interpret $\Pr(d \in R_q)$?
  - Clearly: **Bayesian** (expressing uncertainty regarding R_q)
- Although there is a crisp set R_q (by assumption), **we do not know what R_q looks like**
- Bayesian approach: **Express uncertainty in terms of probability**
- Probabilistic models of information retrieval:
  - Start with $\Pr(d \in R_q)$ and **relate it to other probabilities,** which might be more easily accessible
  - On this way, make some **reasonable assumptions**
  - Finally, **estimate $\Pr(d \in R_q)$** using other probabilities' estimates

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws

Pros and Cons for Probability Methods

- **Pros**
  - Very successful in experiments
  - Probability of relevance as intuitive measure
  - Well-developed mathematical foundations
  - All assumptions can be made explicit

- **Cons**
  - Estimation of parameters usually is difficult
  - Doubtful assumptions
  - Much less flexible than the vector space model
  - Quite complicated

http://www.ifis.cs.tu-bs.de/teaching/ss-11/irws