



Maschinelle Sprachverarbeitung

Introduction to Information Retrieval

Ulf Leser

Content of this Lecture

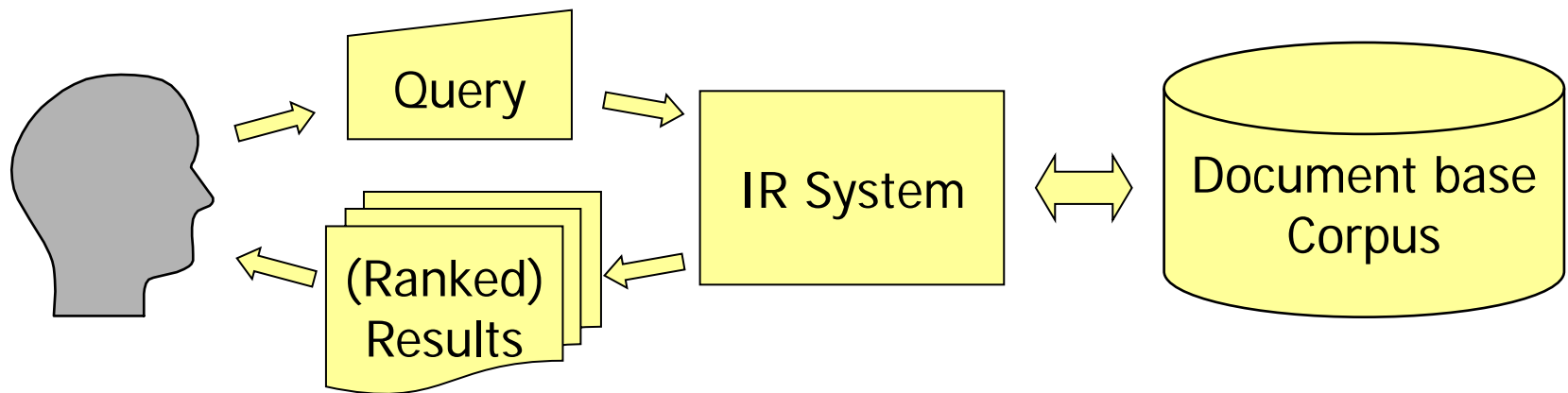
- **What is Information Retrieval**
- Documents & Queries
- Text Preprocessing
- Evaluating IR Systems

Information Retrieval (aka "Search")

- Naïve: Find all **documents** containing the following **words**
- Advanced: „Leading the user to those documents that will best enable him/her to satisfy his/her **need for information**“ [Robertson 1981]
 - A user wants to know something
 - The user needs to tell the machine what he wants to know: query
 - Posing exact queries is difficult: room for interpretation
 - **Machine interprets query** to compute the (hopefully) best answer
 - Goodness of answer (relevance) depends on original intention of user, not on the query
 - “Leading”: Sensible **ranking** of all potentially relevant docs

The Problem

- Help user in **quickly** finding the **requested information** from a **given set of documents**
 - Documents: Corpus, library, collection, ...
 - Quickly: **Few queries**, fast responses, **simple interfaces**, ...
 - Requested: The “best-fitting” documents; the “right” passages; the most “relevant” content



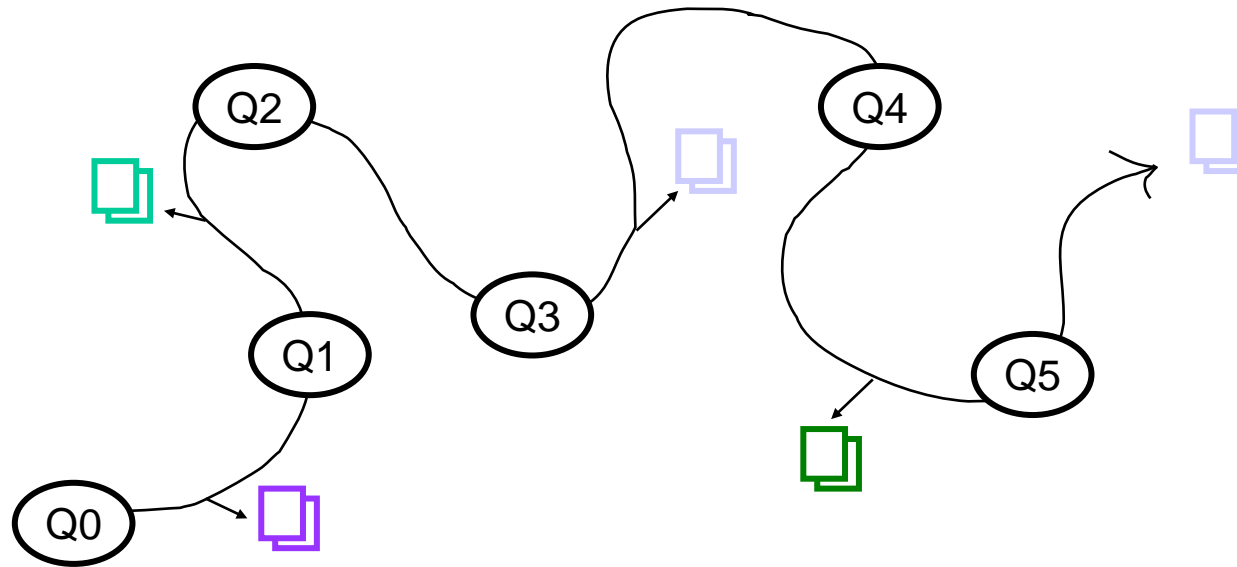
Why is it hard?

- Homonyms (context)
 - Fenster (Glas, Computer, Brief, ...), Berlin (BRD, USA, ...), Boden (Dach, Fussboden, Ende von etwas, ...), ...
- Synonyms
 - Computer, PC, Rechner, Server, Kiste, Bolide, Knoten, Desktop, ...
- **Specific queries** (subfield Question Answering)
 - What was the score of Bayern München versus Stuttgart in the DFB Pokal finals in 1998? Who scored the first goal for Stuttgart?
 - How many hours of sunshine on average has a day in Crete in May?
- Typical web queries have 1,6 terms
- “Information broker” was (is?) a profession

Quickly

- Time to **execute a query**
 - Indexing, parallelization, compression, ...
- Time to **answer the request** (may involve multiple queries)
 - Understand request, find best matches
 - Success of search engines: Better results (and fast!)
 - **Process-orientation**: Exploit user feedback, query history, ...
- Information overload
 - If the corpus is large, **ranking is a must**
 - Result summarization, result clustering
 - Different **search modes**: What's new? What's certain?

IR: An Iterative, Multi-Stage Process



- IR process: “Moving through many actions towards a general goal of satisfactory completion of research related to an information need.”
 - “Berry-picking” [Bates 89]

Prominent Systems I: Digital Libraries

- E.g. OPAC
 - Combination of structured attributes and IR-style queries

Universitätsbibliothek der Humboldt-Universität
Digitale Bibliothek

Anmelden | Hilfe | Schnellsuche | Ressource finden | Suche in Datenbanken | Suchen | Ergebnisse

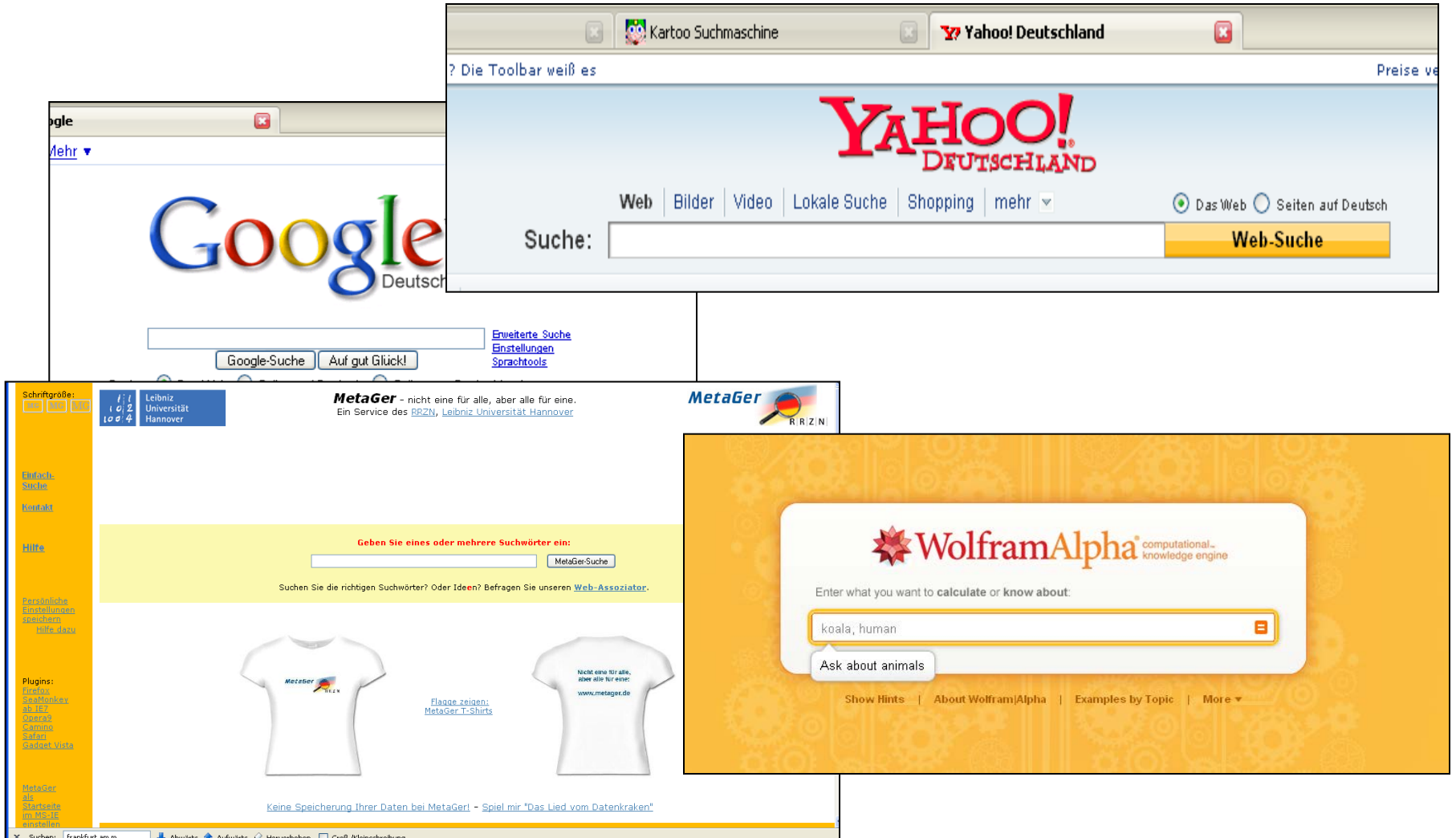
Schnellsuche
Einfach | Erweitert

Suche: Alle Felder | ulf leser | und | Alle Felder

Allg. Fachinform. | Geisteswissenschaft | Literat. Berlin/B. Katalog
Naturwissenschaft. | Sozialw. und Recht | Sprach-
eBooks

No.	Autor	Titel	Jahr	Quelle	Volltext?
1	Leser, Ulf	Informationsintegration Integration verteilter u			
2	Leser, Ulf	A query language for b			
3	Leser, Ulf	Informationsintegration Integration verteilter u			
4	Leser, Ulf [Hrsg.]	Data integration in the life sciences: third International Workshop, DILS 2006, Hinxton, UK, July 20 - 22	2006	KOBV Berlin-Brandenburg	
5	Leser, Ulf	Informationsintegration :Architekturen und Methoden zur Integration verteilter und heterogener Datenquellen	2007	KOBV Berlin-Brandenburg	
Eintrag doppelt - siehe # 2					
6	Leser, Ulf	A query language for biological networks	2005	KOBV Berlin-Brandenburg KOBV Berlin-Brandenburg	
7	Leser, Ulf	Query planning in mediator based information systems	2000	KOBV Berlin-Brandenburg KOBV Berlin-Brandenburg	
Eintrag doppelt - siehe # 7					
8	Leser, Ulf	Query planning in mediator based information systems	2000	KOBV Berlin-Brandenburg KOBV Berlin-Brandenburg	
9	Heyden, Ulf	Zielgruppen des Romans	1986	Staatsbibliothek Berlin	
10	Heyden, Ulf	Zielgruppen des Romans :Analysen, Franz. Romanvorworte d. 19. Jh.	1986	KOBV Berlin-Brandenburg	

Prominent Systems II: Web Search Engines



Content of this Lecture

- What is Information Retrieval
- Documents & Queries
- Text Preprocessing
- Evaluating IR Systems

Document or Passage

The image displays three search results for the query "shakespeare death":

- Universitätsbibliothek (Left):** Shows a search interface with a table of results. The table lists 10 entries, each with a number, author (Shakespeare, William), and title (e.g., "A Catalogue of the Shakespeare Exhibition...").
- Google (Middle):** Shows search results with snippets. Snippets include "THE DEATH OF SHAKESPEARE. Shakespeare died in 1616 on his birthday..." and "Shakespeare is one of the world's most respected dramatists, and, quite a bit to say about various aspects of death...".
- WolframAlpha (Right):** Shows a computational knowledge engine result. The input is "when did shakespeare die?". The result is "Saturday, April 23, 1616". It also provides date formats for Julian, Jewish, and Islamic calendars, and time differences from today (e.g., "394 years 5 months 28 days ago").

Searching only
metadata

Searching tokens
within documents

Interpreting
natural text

Documents

- This lecture: Natural language text
- Might be grammatically correct (books, newspapers) or not (Blogs, Twitter, spoken language)
- May have structure (title, abstract, chapters, ...) or not
- May have associated (explicit or in-text) metadata or not
- May be in many different languages or even mixed
 - Foreign characters
- May refer to other documents (hyperlinks)
- May have various formats (ASCII, PDF, DOC, XML, ...)

IR Queries

- Elements of IR-style queries
 - Keywords, phrases
 - Logical operations (AND, OR, NOT, ...)
 - Web search: “-ulf +leser”
 - Structured queries on metadata (author=... AND title~ ...)
- Documents as queries: Find **documents similar to** this one
- **Query refinement** based on previous results
 - Find documents matching the new query **within the result set** of the previous search
 - Use relevant answers from previous queries to create next query

Searching with Metadata (PubMed/Medline)

The screenshot shows the PubMed website interface. The search bar contains the query "Myers-g[au] sequence[ti]". The search results show 9 items. The first item is highlighted, showing the title "Genome sequence and identification of candidate vaccine antigens from the animal pathogen Dichelobacter nodosus." and the journal "Nat Biotechnol. 2007 May;25(5):569-75. Epub 2007 Apr 29.".

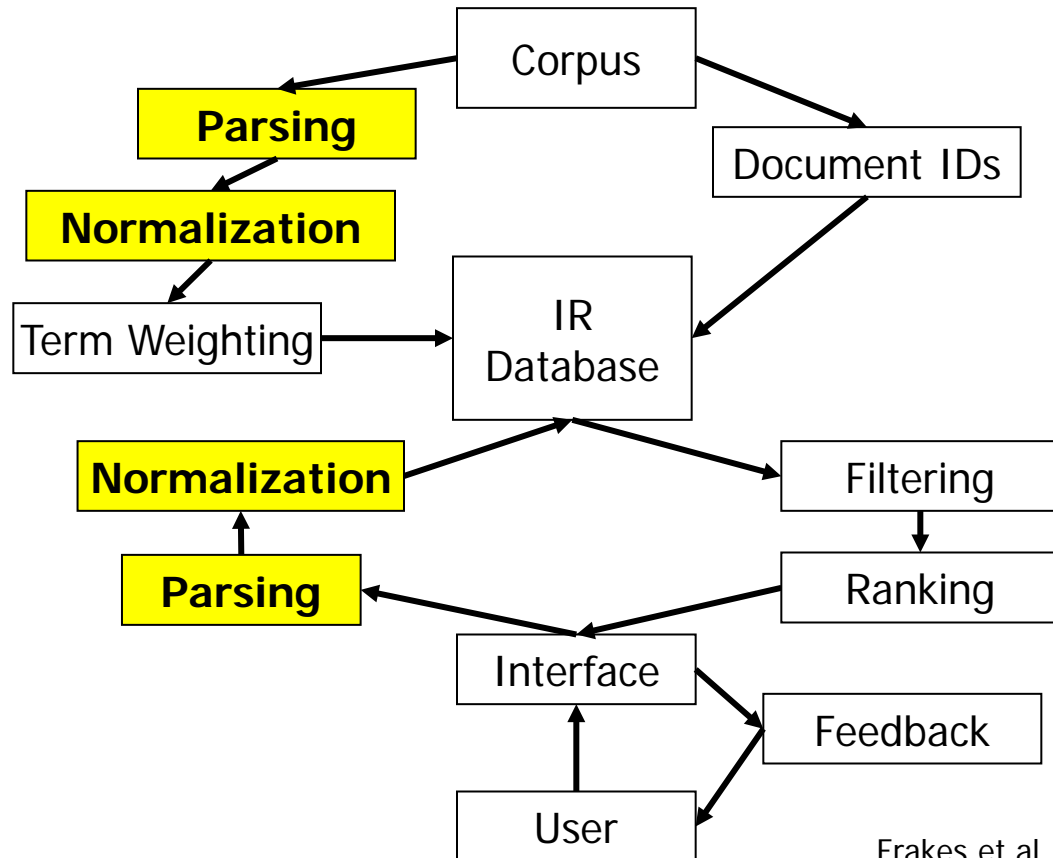
Search Field Descriptions and Tags

Affiliation [AD]	Issue [IP]	Place of Publication [PL]
Article Identifier [AID]	Journal Title [TA]	Publication Date [DP]
All Fields [ALL]	Language [LA]	Publication Type [PT]
Author [AU]	Last Author [LASTAU]	Secondary Source ID [SI]
Comment Corrections	Location ID [LID]	Subset [SB]
Corporate Author [CN]	MeSH Date [MHDA]	Substance Name [NM]
EC/RN Number [RN]	MeSH Major Topic [MAJR]	Text Words [TW]
Entrez Date [EDAT]	MeSH Subheadings [SH]	Title [TI]
Filter [FILTER]	MeSH Terms [MH]	Title/Abstract [TIAB]
First Author Name [1AU]	NLM Unique ID [JID]	Transliterated Title [TT]
Full Author Name [FAU]	Other Term [OT]	UID [PMID]
Full Investigator Name [FIR]	Owner	Volume [VI]
Grant Number [GR]	Pagination [PG]	
Investigator [IR]	Personal Name as Subject [PS]	
	Pharmacological Action MeSH Terms [PA]	

Content of this Lecture

- What is Information Retrieval
- Documents & Queries
- Text Preprocessing
 - Special characters and case
 - Sentence splitting
 - Tokenization
 - Stop words
 - Zipf's law
- Evaluating IR Systems

Processing Pipeline



Frakes et al. 1992

Logical View

- Definition
 - *The **logical view** of a document denotes its representation inside the IR system*
- Determines what users can address in a query
 - Only metadata, only title, only abstract, full text, phrases, stop words, special characters, ...
- Creating the logical view involves transformations
 - Stemming, stop word removal
 - Transformation of special characters (Umlaute, Greek letters, ...)
 - Removal of formatting information (HTML), tags (XML), ...
 - **Bag of words** (BoW)
 - Arbitrary yet fixed order (e.g. sorted alphabetically)
 - See next lecture

Definitions

- Definition
 - A *document* as a sequence of sentences
 - A *sentence* is a sequence of tokens
 - A *token* is the smallest unit of text (words, numbers, ...)
 - A *concept* is the mental representation of a “thing”
 - A *term* is a token or a set of tokens representing a *concept*
 - “San” is a token, but not a term
 - “San Francisco” has two tokens but is only one term
 - Dictionaries usually contain terms, not tokens
 - A *homonym* is a term representing multiple concepts
 - A *synonym* is a term representing a concept which may also be represented by other terms
 - A *syn-set* is a set of synonyms representing the same concept
- “Word” can denote either a token or a term
- We will mostly make no difference between token and terms (sadly ...)

Format Conversion

ABSTRACT

New generation of e-commerce applications require data schemas that are constantly evolving and sparsely populated. The conventional horizontal row representation fails to meet these requirements. We represent objects in a vertical format storing an object

1.1 Issues

In relational database systems, data objects are conventionally stored using a horizontal scheme. A data object is represented as a row of a table. There are as many columns in the table as the number of attributes the objects have. In trying to store all our

New generation of e-commerce applications require data **schemas** In relational database systems, data objects are **conventionally that** are constantly evolving and sparsely populated. The **conven-** stored using a horizontal scheme. A data object is **represented as tional horizontal** row representation fails to meet these require- ...

- Transform PDF, XML, DOC, ... into ASCII / UniCode
- Problems: Formatting instruction, **special characters**, formulas, **tables**, section headers, **footnotes**, ...
- Web: Find the **net content** (no ads, navigation bars, ...)
- Diplomacy: To what extend can one **reconstruct the original document** from its logical view?

Special Characters

- Umlaute, Greek letters, math symbols, ...
- Often part of ASCII/Unicode, but IR systems don't like them
 - **Small alphabets** make indexing, searching, GUIs etc. much easier
- Different way of representation
 - XML/HTML: ` `, `ä`, `<`
- Removing special chars makes querying them impossible
 - How to query for α , Σ , ϵ , ?
- Options
 - Remove special characters
 - **Transcribe**: `ü->ue`, `α -> alpha`, `\forall ->for all`, `Σ ->sum? sigma? ...`
 - Work with large alphabets (Unicode)

Case – A Difficult Case

- Should all text be converted to lower case letters?
- Advantages
 - Makes queries simpler
 - Decreases [index size](#)
 - Allows for some “fuzziness” in search
- Disadvantages
 - No abbreviations
 - Loss of important hints for sentence splitting
 - Loss of important hints for tokenization, NER, ...
 - Loss of semantic info (proper names, Essen versus essen,...)
- Different impact in [different languages](#) (German / English)
- Often: Convert only [after all other preprocessing steps](#)

Sentence Splitting

- Most linguistic analysis works on **sentence level**
- Sentences group together entities and statements
- Naive approach: Reg-Exp search for “[.?!;] ”
 - (note the blank!)
 - **Abbreviations**
 - “C. Elegans is a worm which ...”; “This does not hold for the U.S.”
 - Errors (due to previous normalization steps)
 - “is not clear.Conclusions.We reported on ...”
 - Proper names
 - “.NET is a technique for ...”
 - Direct speech
 - “By saying “It is the economy, stupid!”, B. Clinton meant that ...”
 - ...

Tokenization

- Fundamental elements of all IR query languages are tokens
- Simple approach: search for „ „ (blanks)
 - “A state-of-the-art Z-9 Firebird was purchased on 3/12/1995.”
 - „SQL commands comprise SELECT ... FROM ... WHERE clauses; the latter may contain functions such as leftstr(String, INT).“
 - “This LCD-TV-Screen cost 3,100.99 USD.”
 - “[Bis[1,2-cyclohexanedionedioximato(1-)-O]-[1,2-cyclohexanedione dioximato(2-)-O]methyl-borato(2-)-N,N0,N00,N000,N0000,N00000)-chlorotechnetium) belongs to a family of ...“
- Typical approach (but many (domain-specific) variations)
 - Treat hyphens / parentheses as blanks
 - Remove “.” (after sentence splitting)

Stop Words

- Words that are so frequent that their removal (hopefully) **does not change the meaning** of a doc
 - English: Top-2: 10% of all tokens; Top6: 20%; Top-50: 50%
 - English (top-10; LOB corpus): the, of, and, to, a, in, that, is, was, it
 - German(top-100): aber, als, am, an, auch, auf, aus, bei, bin, ...
- Consequences
 - Removing top-100 stop words **reduces a positional index by ~40%**
 - Hopefully increases precision due to less spurious hits
 - Makes many **phrase queries** impossible
- Variations
 - Remove top 10, 100, 1000, ... words
 - Language-specific, domain-specific, **corpus-specific**

Example

The children of obese and overweight parents have an increased risk of obesity. Subjects with two obese parents are fatter in childhood and also show a stronger pattern of tracking from childhood to adulthood. As the prevalence of parental obesity increases in the general population the extent of child to adult tracking of BMI is likely to strengthen.



100 stop words

children obese overweight parents increased risk obesity. Subjects obese parents fatter childhood show stronger pattern tracking childhood adulthood. prevalence parental obesity increases general population extent child adult tracking BMI likely strengthen.

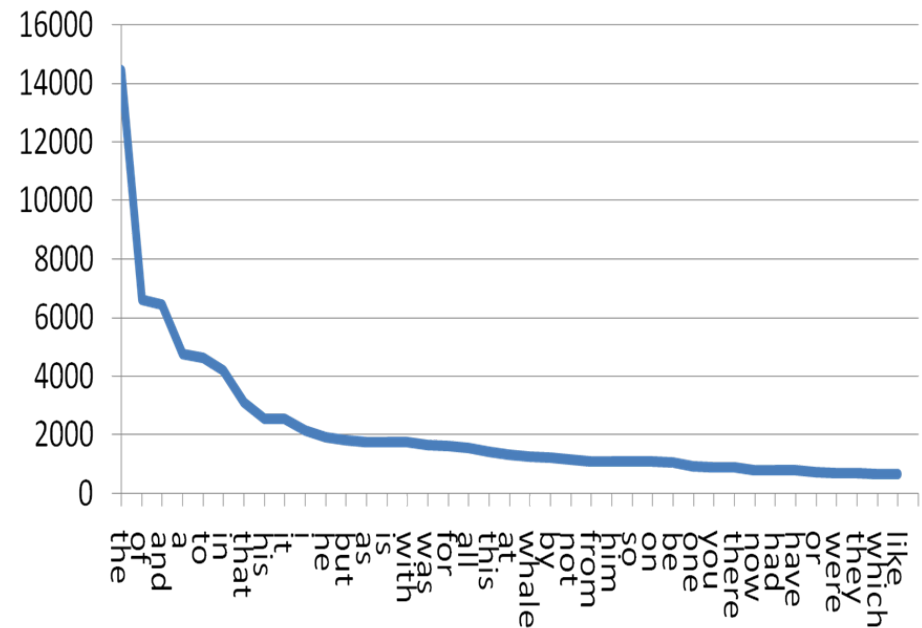


10 000 stop words

obese overweight obesity obese fatter adulthood prevalence parental obesity
BMI

Zipf's Law

- Let f be the **frequency of a word** and r its rank in the list of all words sorted by frequency
- Zipf's law: $f \sim k/r$ for some constant k
- Example
 - Word ranks in Moby Dick
 - Good fit to Zipf's law
 - Some domain-dependency (whale)
- **Fairly good approximation** for most corpora



Source: <http://searchengineland.com/the-long-tail-of-search-12198>

Content of this Lecture

- What is Information Retrieval
- Documents & Queries
- Text Preprocessing
- Evaluating IR Systems

Evaluation: Binary Model

- Assume that for a given query q and many docs $d \in D$, **somebody determines** whether d is relevant for q or not
 - An expert? An average user?
- The IR systems returns all docs it thinks that are relevant
 - **No ranking** for now
- Let T be the set of all truly relevant docs, X the set of all returned docs: $|T| = TP + FN$, $|X| = TP + FP$

	Truth: relevant	Truth: not relevant
IR: relevant	True positives	False positives
IR: not relevant	False negatives	True negatives

Precision and Recall

- **Precision** = $TP / (TP + FP)$
 - What is the fraction of relevant answers in X?
- **Recall** = $TP / (TP + FN)$
 - What is the fraction of found answers in T?
- The perfect world

	Real: Positive	Real: Negative
IR: Positive	A	0
IR: Negative	0	B

Example

- Let $|D| = 10.000$, $|X|=15$, $|T|=20$, $|X \cap T|=9$

	Real: Positive	Real: Negative
IR: Positive	TP = 9	FP = 6
IR: Negative	FN = 11	TN = 9.974

- Precision = $TP/(TP+FP) = 9/15 = 60\%$
- Recall = $TP/(TP+FN) = 9/20 = 45\%$

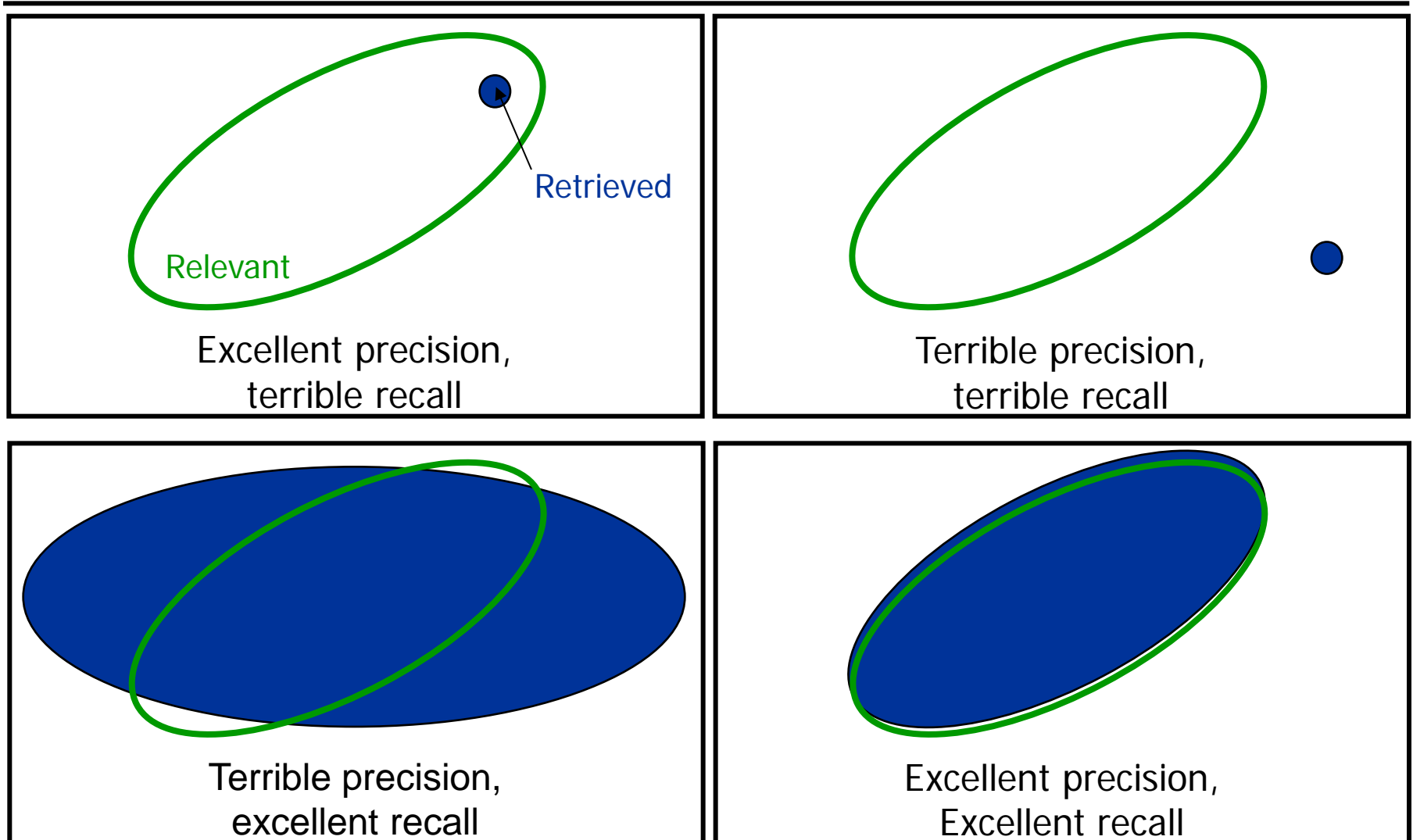
- Assume another result: $|X|=10$, $|X \cap T|=7$

	Real: Positive	Real: Negative
IR: Positive	TP = 7	FP = 3
IR: Negative	FN = 13	

- Precision: 70%, recall = 35%

A Different View

Quelle: A. Nürnberger, VL IR

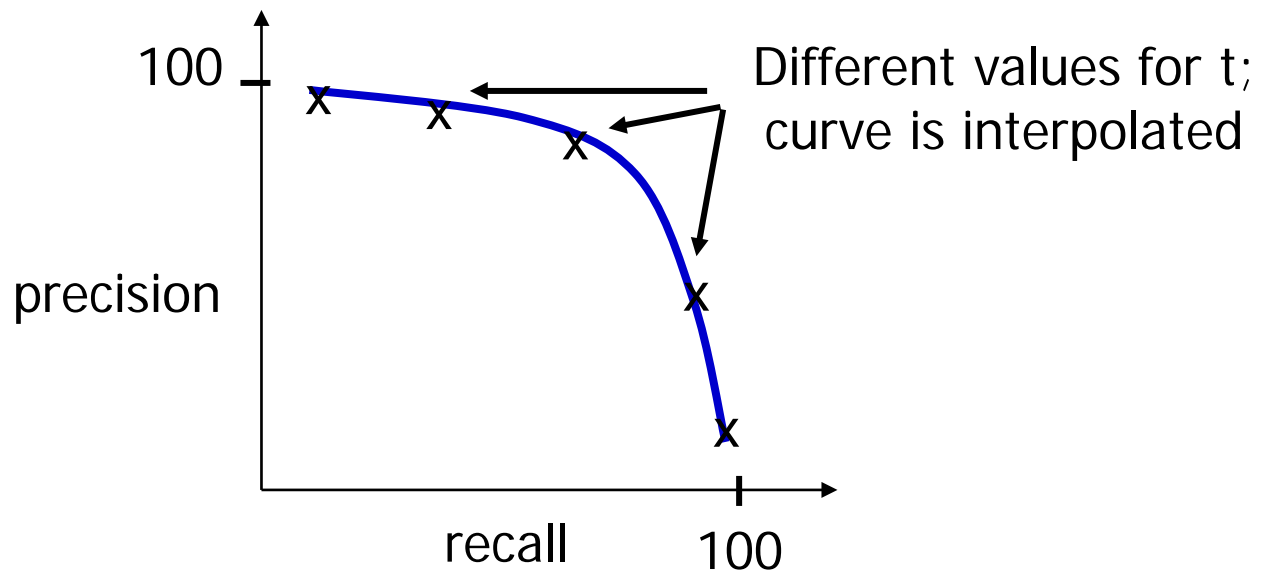


Trade-off

- Inherent **Trade-off** between precision and recall
- Example
 - Think VSM with a **threshold t** to enforce a binary decision
 - Assume that docs with high relevance score are most likely also relevant for the user
 - Increase t : **Less results**, most of them very likely relevant
Precision increases, recall drops
Set $t=1$: $P \sim 100\%$, $R=?$
 - Decrease t : **More results**, some might be wrong
Precision drops, recall increases
Set $t=0$: $P=?$, $R=100\%$

Precision / Recall Curve

- Sliding the threshold t gives a **curve**
 - Similar to Receiver-Operating-Characteristic (ROC)



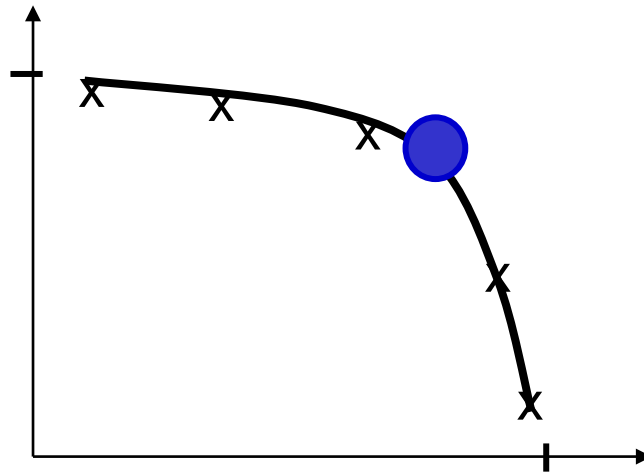
Accuracy

	Truth: relevant	Truth: not relevant
IR: relevant	TP	FP
IR: not relevant	FN	TN

- Sometimes, one wants one measure instead of two
 - E.g. to rank different IR-systems
- Accuracy = $(TP + TN) / (TP + FP + FN + TN)$
 - Which percentage of the **systems decision were correct?**
 - Makes only sense with small corpora and large result set
 - Typically in IR, $TN \gg TP + FP + FN$
 - Thus, accuracy is always good ($\sim 0,9999995$)
- Used in problems with balanced sets of TN / TP

F-Measure

- F-Measure = $2 * P * R / (P + R)$
 - F-Measure is **harmonic mean** between precision and recall
 - Favors balanced P/R values



- Alternative: **Area-under-the-curve** (AUC)
 - AUC doesn't help in finding the best t

From user/query to users/queries

- What if we have **many queries**?
 - Evaluation should always use a range of different queries
 - Compute average P/R values over all queries
 - Of course, mean and stddev are also important
- What if we have **different users**?
 - Different users may have different thoughts about what is relevant
 - Leads to different **gold standards**
 - Compute inter-annotator agreement as upper bound
- Who can judge millions of docs?
 - Evaluate on **small gold standard corpus**
 - But: Extrapolation is difficult: Are the properties of application/corpus really equal to properties of GS?
 - Use implicit feedback, e.g. **click-through rates** in top-K results

Micro- versus Macro Averages

- Two ways of computing an average over many queries
 - **Macro-Average**: Average P and R over P_1, R_1, \dots values of queries
 - **Micro-Average**: Compute P and R over all TP_1, FP_1, \dots values

$$\frac{\sum_{i=1..m} P_i}{m} \neq \frac{\sum_{i=1..m} TP_i}{\sum_{i=1..m} TP_i + \sum_{i=1..m} FP_i}$$

- Comparison
 - Micro-Average can cope with queries without results
 - Micro-Average implicitly **weights queries** with result size
 - Macro-Average is less affected by outliers (with large result sizes)

Self Assessment

- Give a definition of „Information retrieval“
- How is information retrieval different from database query evaluation?
- What are possible types of answers to a IR query?
- List 5 important steps in document preprocessing and their expected impact on precision and recall
- Give a definition of recall, precision, and accuracy
- What is the difference between micro and macro average
- Which preprocessing steps would be affected if you work with a multi language corpus?