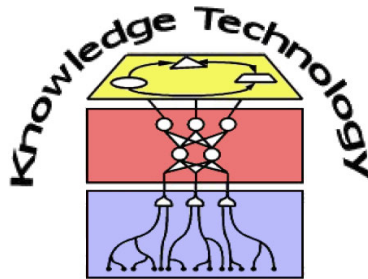


Data Mining

Lecture 12 Text Mining



<http://www.informatik.uni-hamburg.de/WTM/>

Word Mining for Language Acquisition



Video from BBC documentary with Prof. Deb Roy
<http://www.media.mit.edu/people/dkroy>

Goal and Definition of Text Mining

- **Goal**: appreciate relevance and issues in Text Mining
- Text mining is the process of compiling, organizing, and **analyzing large document** collections
- Goal is to support the delivery of **targeted types of information** to analysts and decision makers
- **Discovery of relationships** between related facts that span wide domains of inquiry

Mining Text Data comes with different Names

- Data mining from text, text mining
- Natural language processing
- Information extraction
- Information retrieval from text
- Text categorization methods
- Material based on book by Han and Kamber, 2006 and additional slides from Cheng Xiang Zhai, Mooney, Volinsky

Free Text versus Structured Data

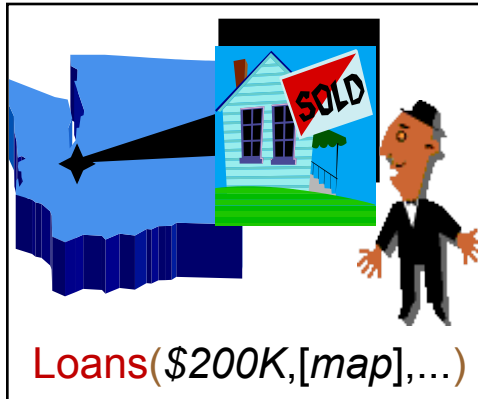
Data Mining / Knowledge Discovery



Structured Data

HomeLoan (
 Loatee: Frank Rizzo
 Lender: MWF
 Agency: Lake View
 Amount: \$200,000
 Term: 15 years
)

Multimedia



Free Text

Frank Rizzo bought his home from Lake View Real Estate in 1992.
He paid \$200,000 under a 15-year loan from MW Financial.

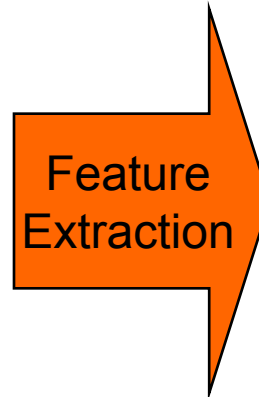
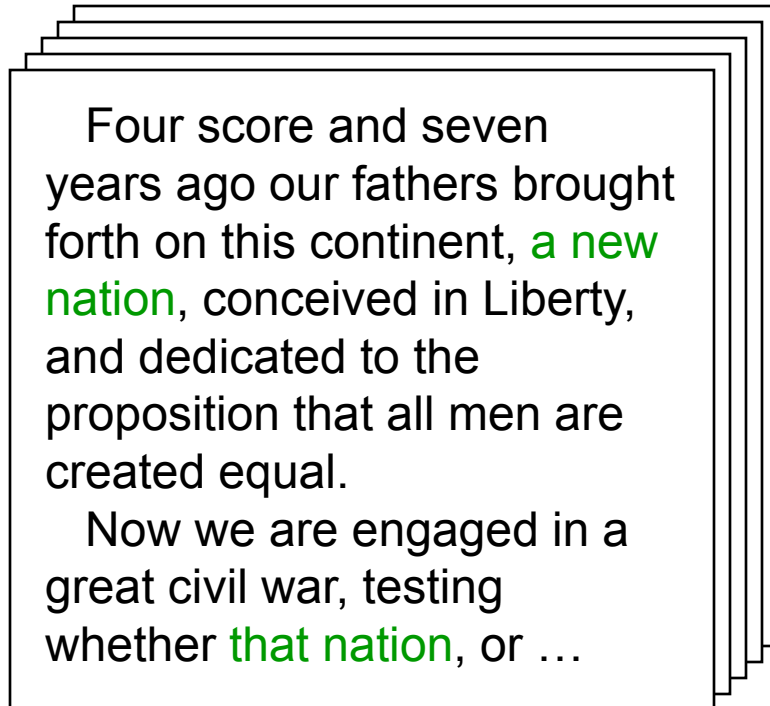
Hypertext

[Frank Rizzo](#) bought
[this home](#) from [Lake View Real Estate](#)
In **1992**.

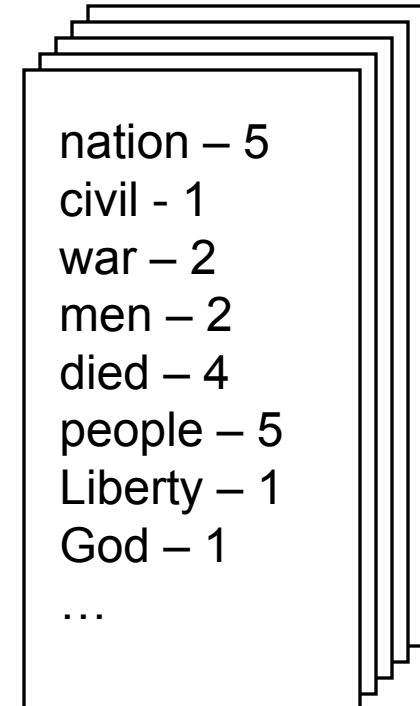
...

Bag-of-Tokens Approaches

Documents



Token **Sets**

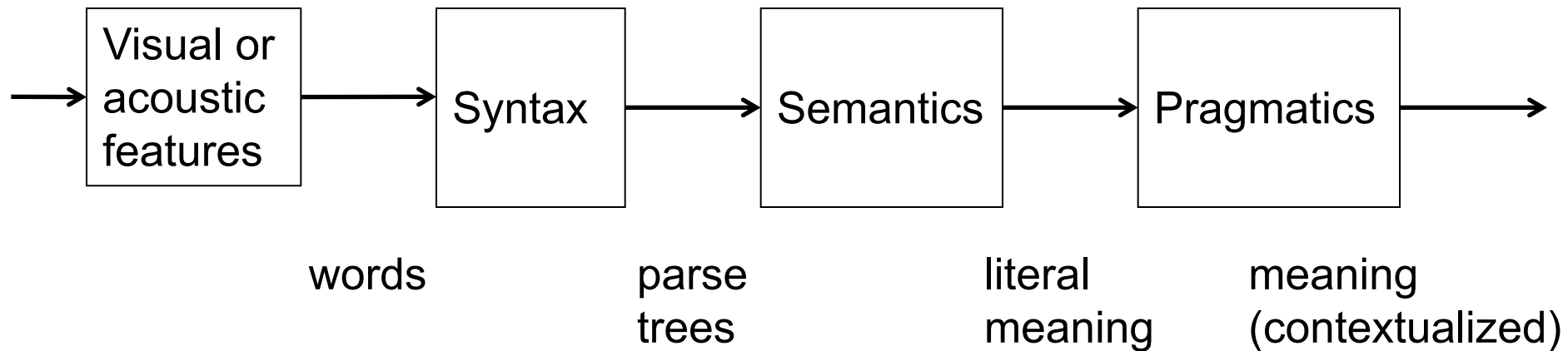


Looses all order-specific information!
Severely limits *context*!

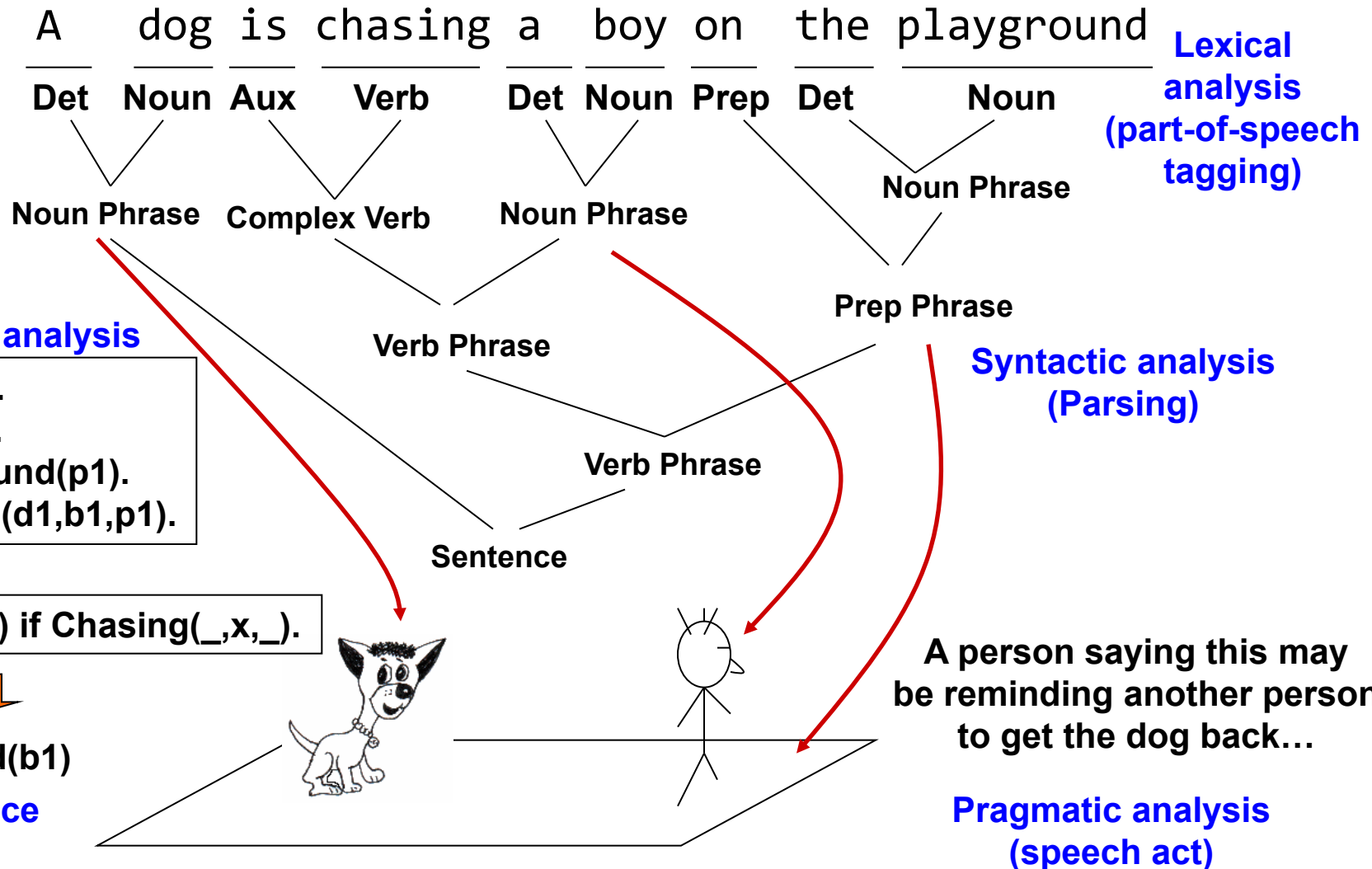
Syntax, Semantic, Pragmatics

- **Syntax:** proper ordering of words and its possible effect on meaning.
 - The dog bit the boy.
 - The boy bit the dog.
 - * Bit boy dog the the.
 - Colorless green ideas sleep furiously.
- **Semantics:** concerns the (literal) meaning of words, phrases, and sentences.
 - “plant” as a photosynthetic organism
 - “plant” as a manufacturing facility
- **Pragmatics:** concerns the overall communicative and social context and its effect on interpretation.
 - The ham sandwich wants another beer. (co-reference, anaphora)
 - John thinks vanilla. (ellipsis)

Comprehension as a Simplified Sequential Model



From flat Text to Meaning and Structure



Language is challenging but so effective for Communication!

- Word-level ambiguity
 - “design” can be a noun or a verb (Ambiguous Part of Speech)
 - “root” has multiple meanings (Ambiguous semantic sense)
- Syntactic ambiguity
 - “natural language processing” (Modification/Bracketing)
 - “A man saw a boy **with a telescope**.” (Prepositional Phrase Attachment)
- Semantics and Anaphora resolution
 - “John persuaded Bill to buy a TV for **himself**.”
(**himself** = John or Bill?)
- Presupposition and pragmatic inferences
 - “He has quit smoking.”
 - implies that he smoked before.

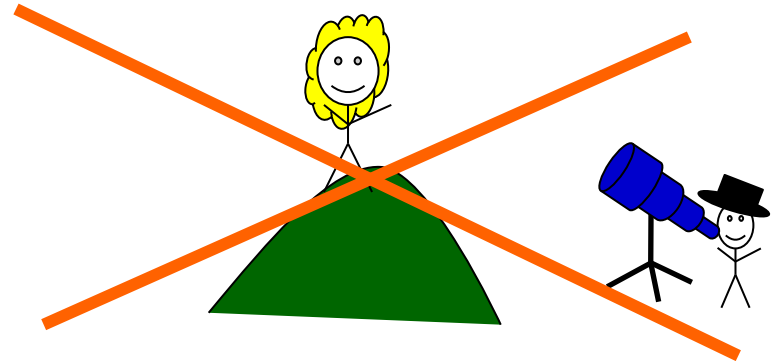
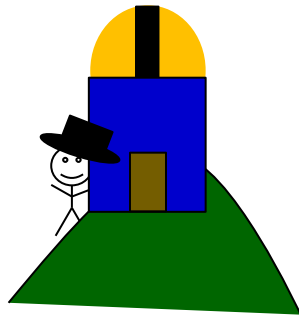
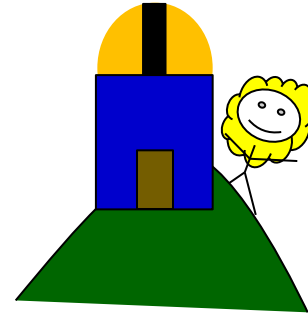
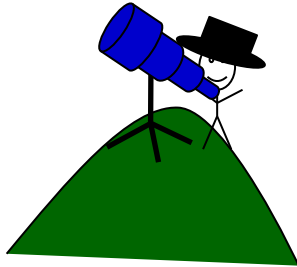
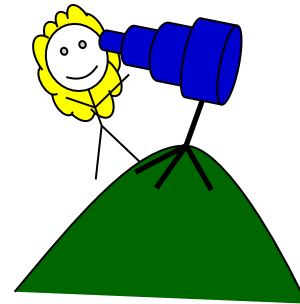
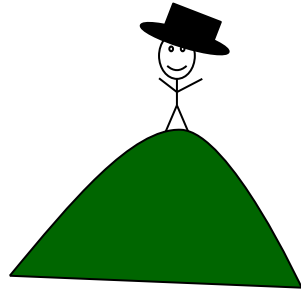
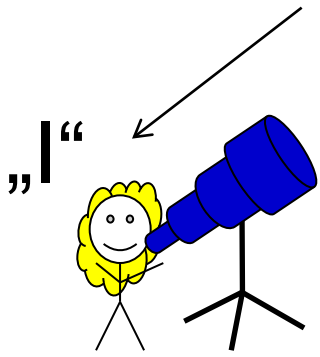
**Humans rely on *context* to interpret (when possible).
This context may extend beyond a given document!**

Ambiguity: Different Interpretations?

- Natural language can be highly ambiguous
- Can you find ambiguities?
 - I saw the Grand Canyon flying to LA.
 - Time flies like an arrow.
 - I saw the man on the hill with a telescope.

Ambiguity

I saw the man on the hill with a telescope.



Ambiguity is Ubiquitous but we may not notice

■ Speech Recognition

- “recognize speech” vs. “wreck a nice beach”
- “youth in Asia” vs. “euthanasia”

■ Syntactic Analysis

- “I ate spaghetti **with** chopsticks” vs. “I ate spaghetti **with** meatballs.”

■ Semantic Analysis

- “I put the **plant** in the window” vs. “Ford put the **plant** in Mexico”

■ Pragmatic Analysis

- **Example** from “The Pink Panther Strikes Again”:

Clouseau: Does your dog bite?

Hotel Clerk: No.

Clouseau: [*bowing down to pet the dog*] Nice doggie.

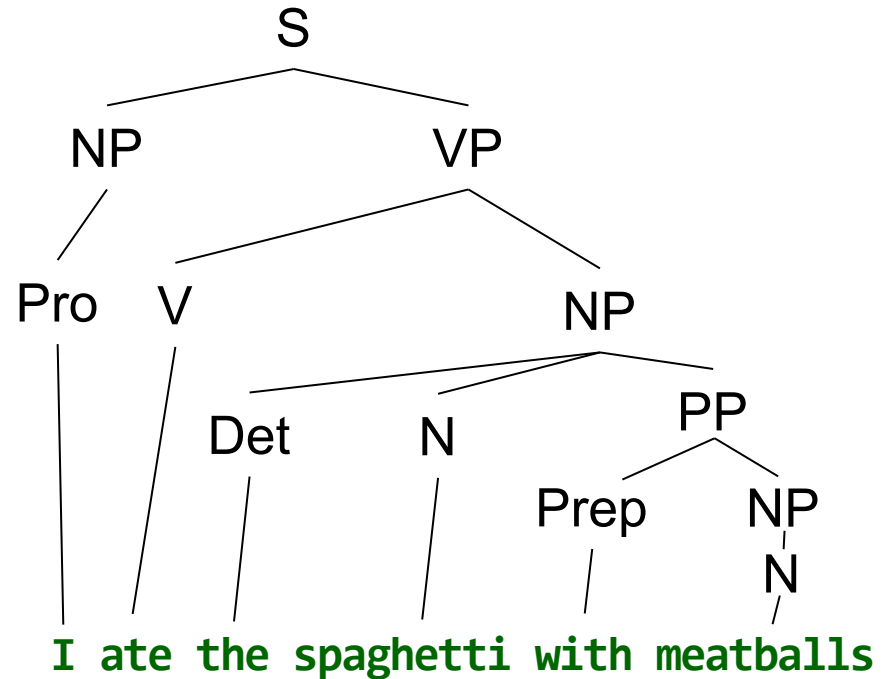
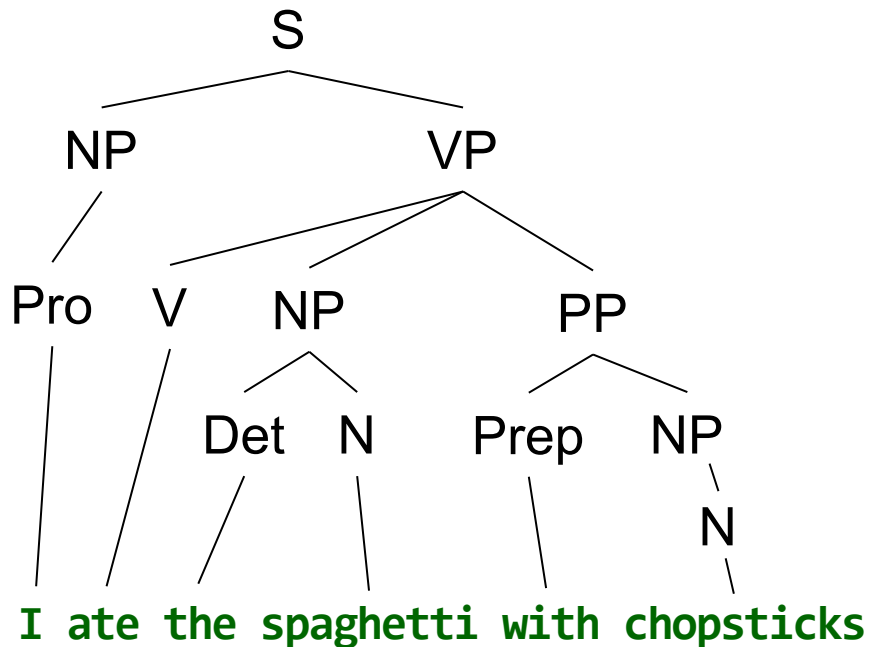
[*Dog barks and bites Clouseau in the hand*]

Clouseau: I thought you said your dog did not bite!

Hotel Clerk: That is not my dog.

Syntactic Parsing

- Produce the correct syntactic parse tree for a sentence



Ambiguity is Explosive

- Ambiguities compound to generate enormous numbers of possible interpretations.
- In English, a sentence ending in n prepositional phrases has *over* 2^n syntactic interpretations.
 - “I saw the man with the telescope.”: 2 parses
 - “I saw the man on the hill with the telescope.”: 5 parses
 - “I saw the man on the hill in Texas with the telescope.”: 14 parses
 - “I saw the man on the hill in Texas with the telescope at noon.”: 42 parses
 - “I saw the man on the hill in Texas with the telescope at noon on Monday.” 132 parses

Mining Language in complex Environments (Knowledge Technology Lab, WTM)



How can we deal with Mining from text at all?

Shallow natural language processing

- Progress on *useful Sub*-Goals:
 - English Lexicon
 - Part-of-Speech Tagging
 - Word Sense Disambiguation
 - Phrase Detection / Parsing

Morphological Analysis

- **Morphology** is the field of linguistics that studies the internal structure of words.
- A **morpheme** is the smallest linguistic unit that has semantic meaning
 - **E.g.** “carry”, “pre”, “ed”, “ly”, “s”
- Morphological analysis is the task of segmenting a word into its morphemes:
 - carried \Rightarrow carry + ed (past tense)
 - independently \Rightarrow in + (depend + ent) + ly
 - Googlers \Rightarrow (Google + er) + s (plural)
 - unlockable \Rightarrow un + (lock + able) ?
 \Rightarrow (un + lock) + able ?

Part-of-Speech (POS) Tagging

Training data (Annotated text)

<i>This</i>	<i>sentence</i>	<i>serves</i>	<i>as</i>	<i>an</i>	<i>example</i>	<i>of</i>	<i>annotated</i>	<i>text...</i>
Det	N	V1	P	Det	N	P	V2	N

"This is a new sentence." → **POS Tagger** → *This is a new sentence.*
Det Aux Det Adj N

Pick the **most likely** tag sequence.

$$p(w_1, \dots, w_k, t_1, \dots, t_k) = \begin{cases} p(t_1 | w_1) \dots \underline{p(t_k | w_k)} p(w_1) \dots p(w_k) \\ \prod_{i=1}^k \underline{p(w_i | t_i)} \underline{p(t_i | t_{i-1})} \end{cases}$$

Independent assignment
Most common tag

Partial dependency
(HMM)

Phrase Chunking rather than Full Parsing

- Find all non-recursive noun phrases (**NPs**) and verb phrases (**VPs**) in a sentence.
 - [NP I] [VP ate] [NP the spaghetti] [PP with]
[NP meatballs].
 - [NP He] [VP reckons] [NP the current account deficit]
[VP will narrow] [PP to] [NP only # 1.8 billion]
[PP in] [NP September]

Probabilistic Structure Parsing to reduce Ambiguity (only if necessary)

Choose *most likely* parse tree...

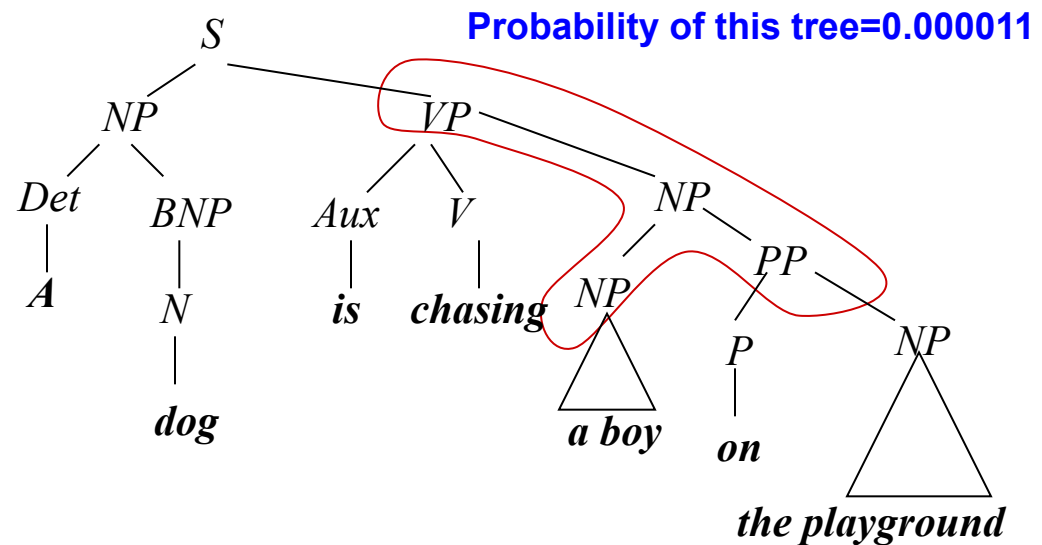
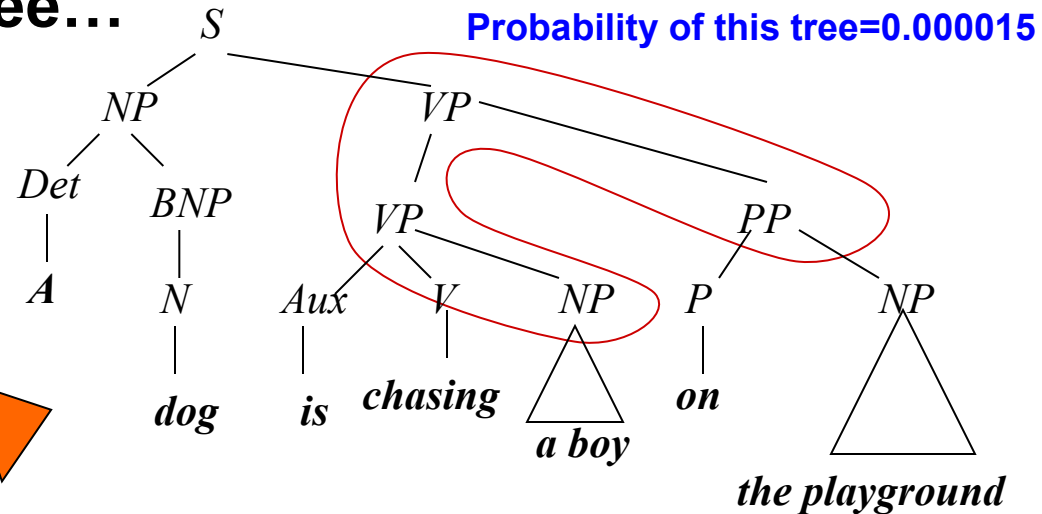
Probabilistic CFG

Grammar

$S \rightarrow NP VP$ 1.0
 $NP \rightarrow Det BNP$ 0.3
 $NP \rightarrow BNP$ 0.4
 $NP \rightarrow NP PP$ 0.3
 $BNP \rightarrow N$...
 $VP \rightarrow V$...
 $VP \rightarrow Aux V NP$...
 $VP \rightarrow VP PP$...
 $PP \rightarrow P NP$ 1.0

Lexicon

$V \rightarrow chasing$ 0.01
 $Aux \rightarrow is$...
 $N \rightarrow dog$ 0.003
 $N \rightarrow boy$...
 $N \rightarrow playground$...
 $Det \rightarrow the$...
 $Det \rightarrow a$...
 $P \rightarrow on$...



From Structure to Semantics:

Word Sense Disambiguation (WSD)

- Words in natural language usually have a fair number of different possible meanings.
 - Ellen has a strong **interest** in computational linguistics.
 - Ellen pays a large amount of **interest** on her credit card.
- For many tasks (question answering, translation), the proper sense of each ambiguous word in a sentence must be determined.

Semantic Role Labeling (SRL)

- For each clause, determine the semantic role played by each noun phrase that is an argument to the verb.

agent patient source destination instrument

- John drove Mary from Austin to Dallas in his Toyota Prius.
 - The hammer broke the window.
- Also referred to a “**case role analysis**”, “**thematic analysis**”, and “**shallow semantic parsing**”

Semantic Information Extraction (IE)

- Identify phrases in language that refer to specific types of entities and relations in text.
- **Named entity recognition** for identifying names of people, places, organizations, etc. in text.

people organizations places

- Michael Dell is the CEO of Dell Computer Corporation and lives in Austin Texas.

- **Relation extraction** identifies specific relations between entities.

- Michael Dell is the CEO of Dell Computer Corporation and lives in Austin Texas.
-

Question Answering

- Directly answer natural language questions based on information presented in a corpora of textual documents (e.g. the web).
 - When was Barack Obama born? (*factoid*)
 - ⇒ August 4, 1961
 - Who was president when Barack Obama was born?
 - ⇒ John F. Kennedy
 - How many presidents have there been since Barack Obama was born? (*towards more inferences*)
 - ⇒ 9
- ⇒ Much but not all information may be directly available

Text Summarization

- Produce a short summary of a longer document or article.
 - **Article**: With a split decision in the final two primaries and a flurry of super-delegate endorsements, Sen. Barack Obama sealed the Democratic presidential nomination last night after a grueling and history-making campaign against Sen. Hillary Rodham Clinton that will make him the first African American to head a major-party ticket. Before a chanting and cheering audience in St. Paul, Minn., the first-term senator from Illinois savored what once seemed an unlikely outcome to the Democratic race with a nod to the marathon that was ending and to what will be another hard-fought battle, against Sen. John McCain, the presumptive Republican nominee....
 - **Summary**: Senator Barack Obama was declared the presumptive Democratic presidential nominee.

Mining Text Data in Internet (Video)



Information Retrieval as Start for Text Mining

- Typical traditional IR systems
 - Online library catalogs
 - Online document management systems
- Information retrieval vs. database systems
 - Some DB problems are not present in IR,
 - **E.g.**, update, transaction management, complex objects
 - Some IR problems are not addressed well in DBMS
 - **E.g.**, unstructured documents, approximate search using keywords and relevance

Information Retrieval vs Information Extraction

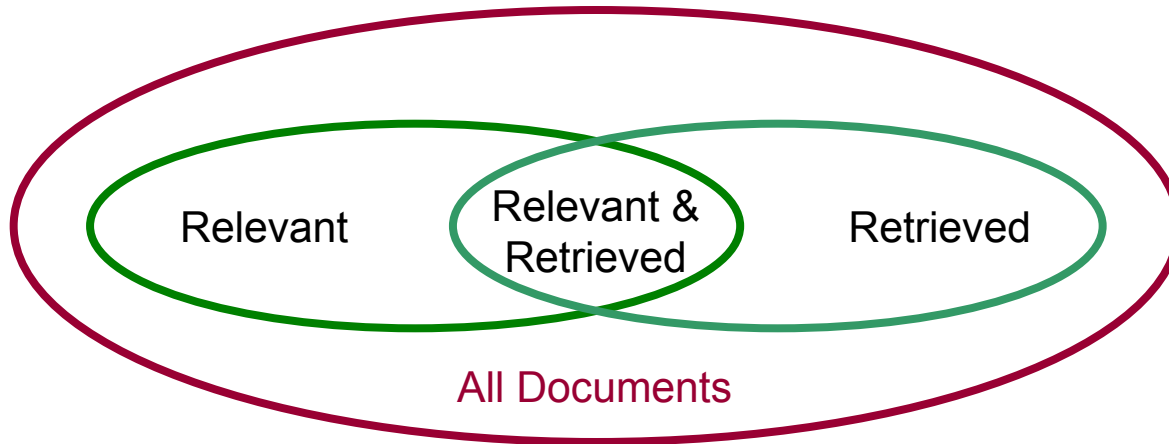
■ *Information Retrieval*

- Given a set of query terms and a set of document terms select only
 - the most relevant documents [*precision*], and
 - preferably all the relevant [*recall*].

■ *Information Extraction*

- Extract what the document contains from the text
-
- IR systems can FIND documents but does not need to “understand” them

Basic Measures for Text Retrieval



- **Precision**: the percentage of retrieved documents that are in fact relevant to the query (i.e., “correct” responses)

$$precision = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Retrieved\}|}$$

- **Recall**: the percentage of documents that are relevant to the query and were, in fact, retrieved

$$recall = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

Precision vs. Recall

- In other words (we have been here before!)
 - Precision = $TP/(TP+FP)$
 - Recall = $TP/(TP+FN)$

	Truth:Relvant	Truth:Not Relevant
Algorithm:Relevant	TP	FP
Algorithm: Not Relevant	FN	TN

- Trade off:
 - If algorithm is 'picky': precision high, recall low
 - If algorithm is 'relaxed': precision low, recall high
- BUT: recall often hard if not impossible to calculate

Information Retrieval Techniques

■ Basic Concepts

- A document can be described by a set of representative keywords called *index terms*.
- Different index terms have varying relevance when used to describe document contents.
- This effect is captured through the *assignment of numerical weights to each index term* of a document. (e.g.: frequency, tf-idf: term frequency-inverse document frequency)

■ DBMS Analogy

- Index Terms → *Attributes*
- Weights → *Attribute Values*

Information Retrieval Techniques

- ***Effective Index Terms (Attribute) Selection:***
 - Stop list
 - Word stem
 - Index terms weighting methods
- Terms **×** Documents Frequency Matrices
- ***Information Retrieval Models:***
 - Boolean Model
 - Vector Model
 - Probabilistic Model

Boolean Model

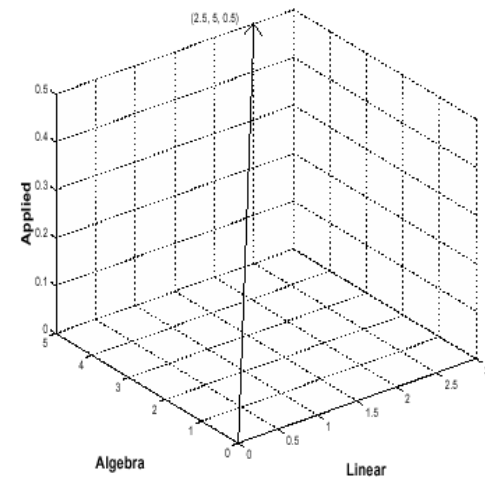
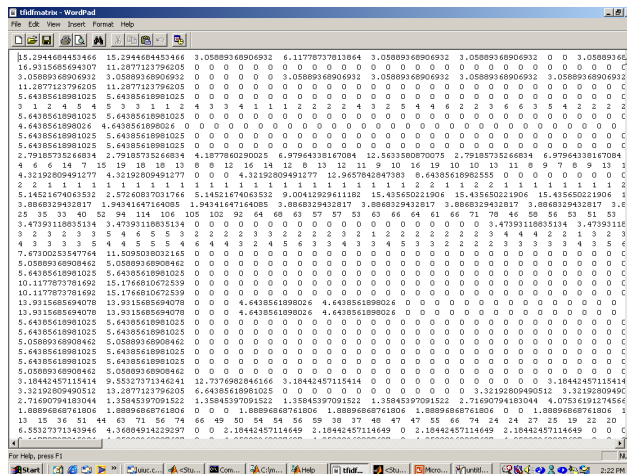
- Consider that *index terms are either present or absent* in a document
- As a result, the index term weights are assumed to be all *binaries*
- A query is composed of index terms linked by three connectives: *not*, *and*, and *or*
 - **E.g.:** car *and* repair, plane *or* airplane
- The Boolean model predicts that *each document is either relevant or non-relevant* based on the match of a document to the query
- Think about the advantages / disadvantages !

Vector Space Model

- ***Represent a document by a term vector***
 - Term: basic concept, e.g., word or phrase
 - Each term defines one dimension
 - N terms define a N-dimensional space
 - Element of vector corresponds to term weight
 - E.g., $d = (x_1, \dots, x_N)$, x_i is ***importance*** of term i
- New document is assigned to the most likely category based on ***vector similarity***.

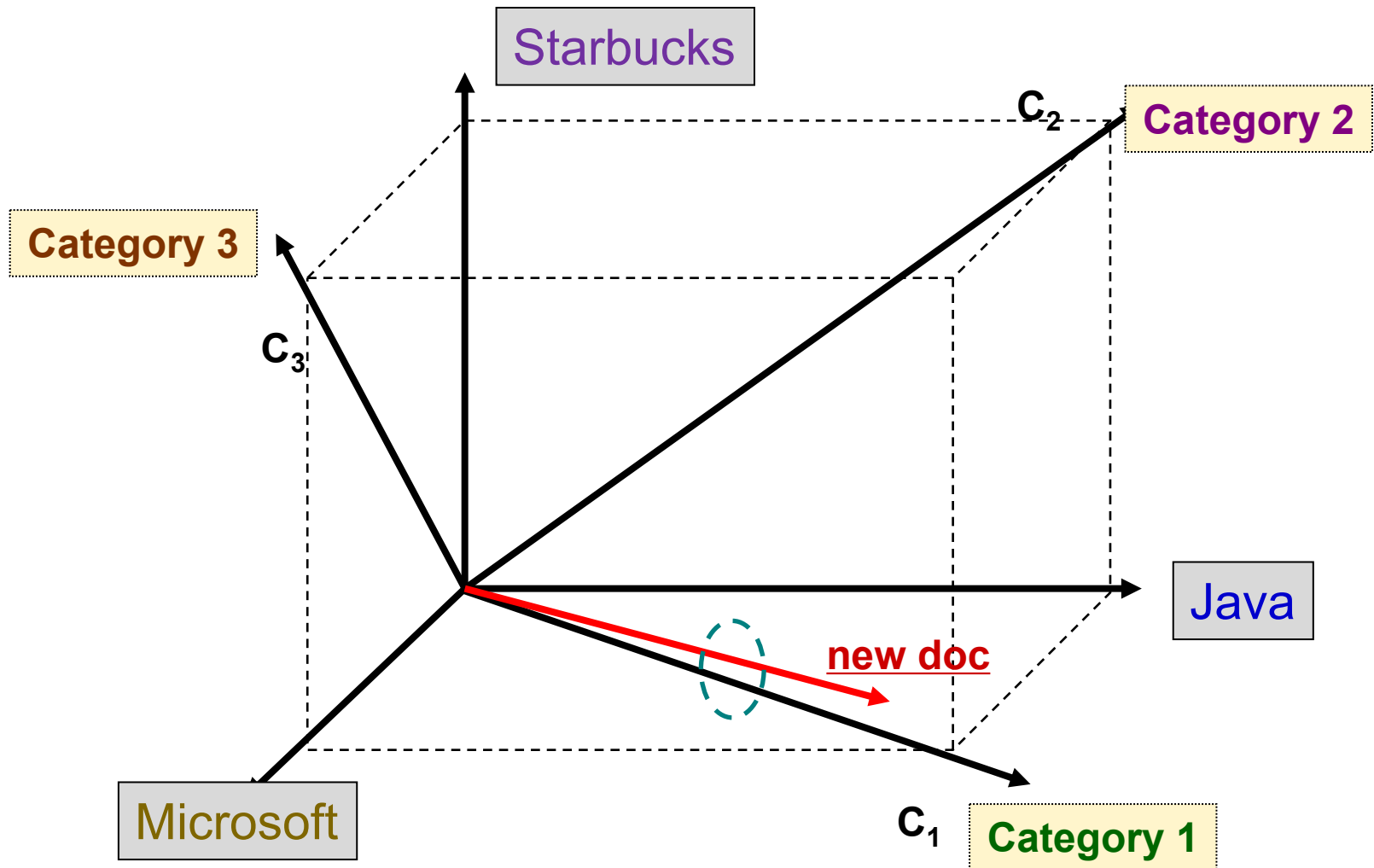
Vector Space Model

- Documents & user queries represented as m -dimensional vectors
- m is total number of index terms in document collection



- Degree of *similarity* of the document d with regard to the query q:
 - Calculated as the *correlation* between the vectors that represent them
 - Using measures such as the *Euclidian distance or the cosine* of the angle between these two vectors

Vector Space Model: Illustration



What VS Model does not specify

- How to select terms to capture “*basic concepts*”
 - Stop words
 - E.g. “a”, “the”, “always”, “along”
 - Word stemming
 - E.g. “computer”, “computing”, “computerize” => “compute”
- How to *assign weights*
 - Not all words are equally important: Some are more indicative than others
 - E.g. “algebra” vs. “science”
- How to measure the similarity?

How to assign Weights

- Two-fold heuristics based on frequency
 - TF (Term frequency)
 - More frequent *within* a document → more relevant to semantics
 - e.g., “query” vs. “commercial” in a commercial document
 - IDF (Inverse document frequency)
 - Less frequent *among* documents → more discriminative
 - e.g. “algebra” vs. “science”

TF Weighting

- **Weighting:**

- More frequent \Rightarrow more relevant to topic
 - Raw TF= $f(t,d)$: how many times term t appears in doc d

- **Normalization:**

- Document length varies \Rightarrow relative frequency preferred
 - **E.g.**, Maximum frequency normalization

$$\text{TF}(t, d) = 0.5 + \frac{0.5 \cdot f(t, d)}{\text{MaxFreq}(d)}$$

- After normalization: values between 0.5 and 1
- Augmented frequency to prevent bias for longer documents

IDF Weighting

- Ideas:

- Measure of how much information the word provides
- Less frequent *among* documents → more discriminative

- Formula:

$$\text{IDF}(t) = \log\left(\frac{n}{1+k}\right)$$

n — total number of docs

k — number of docs with term t appearing
(the DF document frequency)

1: avoid division by 0

TF-IDF Weighting

- Ideas:
 - Combine term frequency and inverse document frequency

- Formula:

$$\text{TFIDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D)$$

high weighting values for 1) high term frequency in a document and 2) for a low document frequency of term t in all documents D

Similarity-based Retrieval in Text Data (1)

- Finds similar documents based on a set of common keywords
- Answer should be based on the *degree of relevance* based on the nearness of the keywords, relative frequency of the keywords, etc.
- *Stop list*
 - Set of words that are deemed *irrelevant*, even though they may appear frequently
 - **E.g.**, a, the, of, for, to, with, etc.
 - Stop lists may vary when document set varies

Stop Words

- Many of the most frequently used words in English are almost worthless in retrieval and text mining – these words are called **stop words**
 - the, of, and, to,
 - Typically up to about 400 to 500 such words
 - For an application or domain specific stop words list may be constructed
- **Why do we need to remove stop words?**
 - Reduce indexing (or data) file size
 - stopwords accounts 20-30% of total word counts.
 - Improve efficiency
 - stop words are not useful for searching or text mining
 - stop words always have a large number of hits

Similarity-based Retrieval in Text Data (2)

■ **Word stem**

- Several words are small syntactic variants of each other since they share a common word stem
- **E.g.**, drug, drugs, drugged

■ A **term frequency table**

- Each entry $frequent_table(i, j) = \#$ of occurrences of the word t_j in document d_i
- Usually, the **ratio** instead of the absolute number of occurrences is used

■ **Similarity metrics**: measure the closeness of a document to a query (a set of keywords)

- Relative term occurrences
- Cosine similarity:

$$sim(v_1, v_2) = \frac{v_1 \cdot v_2}{|v_1| |v_2|}$$

Stemming: additional examples

- Techniques used to find the root/stem of a word:
 - E.g.,
 - user engineering
 - users engineered
 - used engineer
 - using
 - stem: use engineer
- **Usefulness:**
 - improving effectiveness of retrieval and text mining
 - matching similar words
 - reducing indexing size
 - combining words with same roots may reduce indexing size as much as 40-50%.

Basic stemming algorithms (e.g. Porter Algorithm)

- remove ending
 - if a word ends with a **consonant** other than **s**, followed by an **s**, then delete **s**.
 - if a word ends in **es**, drop the **s**.
 - if a word ends in **ing**, delete the **ing** unless the remaining word consists only of one letter or of **th**.
 - If a word ends with **ed**, preceded by a consonant, delete the **ed** unless this leaves only a single letter.
 -
- transform words
 - if a word ends with “ies” but not “eies” or “aies” then “ies --> y.”

Term / Document Matrix

- Most common form of representation in text mining is the ***term - document*** matrix
 - Term: typically a single word, but could be a word phrase like “data mining”
 - Document: a generic term meaning a collection of text to be retrieved
 - Can be large - terms are often 50k or larger, documents can be in the billions (www).
 - Can be binary or use counts

Term / Document Matrix Example (1)

Example: 10 documents: 6 terms

	Database	SQL	Index	Regression	Likelihood	linear
D1	24	21	9	0	0	3
D2	32	10	5	0	3	0
D3	12	16	5	0	0	0
D4	6	7	2	0	0	0
D5	43	31	20	0	3	0
D6	2	0	0	18	7	6
D7	0	0	1	32	12	0
D8	3	0	0	22	4	4
D9	1	0	0	34	27	25
D10	6	0	0	17	4	23

$$D_1 = (d_{i1}, d_{i2}, \dots, d_{it})$$

- Each document now is just a vector of terms, sometimes boolean

Term / Document Matrix Example (2)

Example: 10 documents: 6 terms

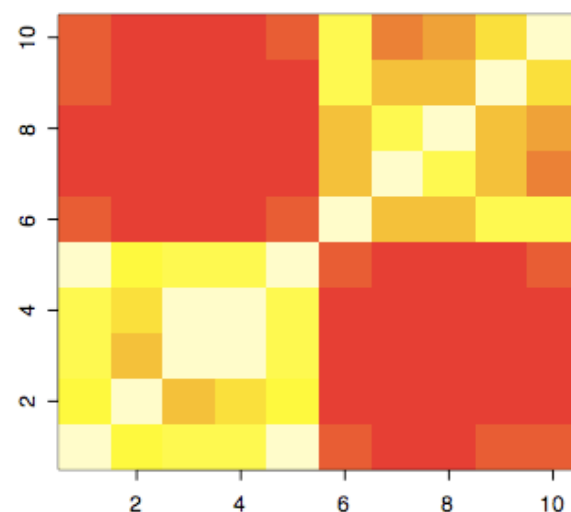
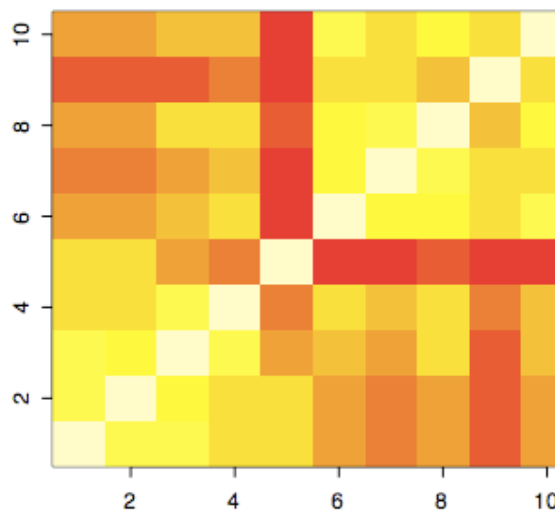
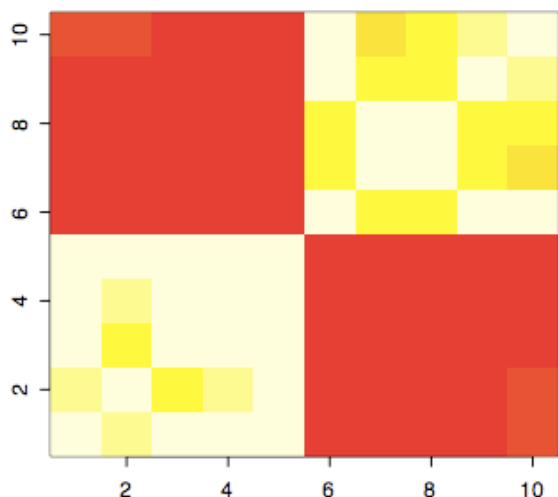
$$D_1 = (d_{i1}, d_{i2}, \dots, d_{it})$$

	Database	SQL	Index	Regression	Likelihood	linear
D1	24	21	9	0	0	3
D2	32	10	5	0	3	0
D3	12	16	5	0	0	0
D4	6	7	2	0	0	0
D5	43	31	20	0	3	0
D6	2	0	0	18	7	6
D7	0	0	1	32	12	0
D8	3	0	0	22	4	4
D9	1	0	0	34	27	25
D10	6	0	0	17	4	23

- We can calculate cosine and Euclidean distance for this matrix
- What would you want the distances to look like?

Visualisation of Document distance

- Pairwise distances between documents
- Image plots of cosine distance, Euclidean, and scaled Euclidean



R function: 'image'

Queries and towards Latent Semantic Indexing

- A query is a representation of the user's information needs
 - Normally a list of words.
- Once we have a TD matrix, queries can be represented as a vector in the same space
 - “Database Index” = $(1,0,1,0,0,0)$
- Query can be a simple question in natural language



- Calculate cosine distance between query and documents
 - Returns a ranked vector of documents

Latent Semantic Indexing (1)

- Criticism: queries can be posed in many ways, but still mean the same
 - Data mining and knowledge discovery
 - Car and automobile
 - Beet and beetroot
- **Semantically**, these are (almost) the same, and documents with either term are relevant.
- Using **synonym lists or thesauri** address the problem, but are messy and difficult.
- Latent Semantic Indexing (LSI): tries to **extract hidden semantic structure** in the documents
- Search what I meant, not what I said!

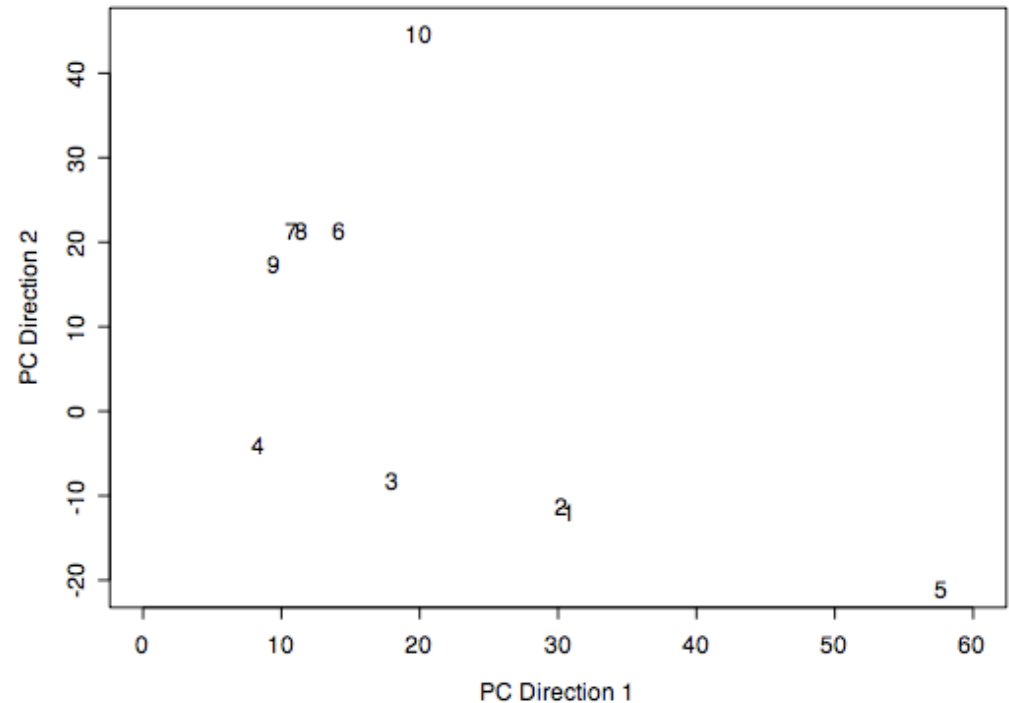
Latent Semantic Indexing (2)

- Approximate the T-dimensional term space using principal components calculated from the TD matrix
- The first k **Principal Components** (PCA) directions provide the best set of k orthogonal basis vectors - these explain the most variance in the data.
 - Data is reduced to an $N \times k$ matrix, without much loss of information (N number of documents)
- Each **direction** is a linear combination of the input terms, and define a clustering of “topics” in the data.

LSI Example (1)

```
> pc2
```

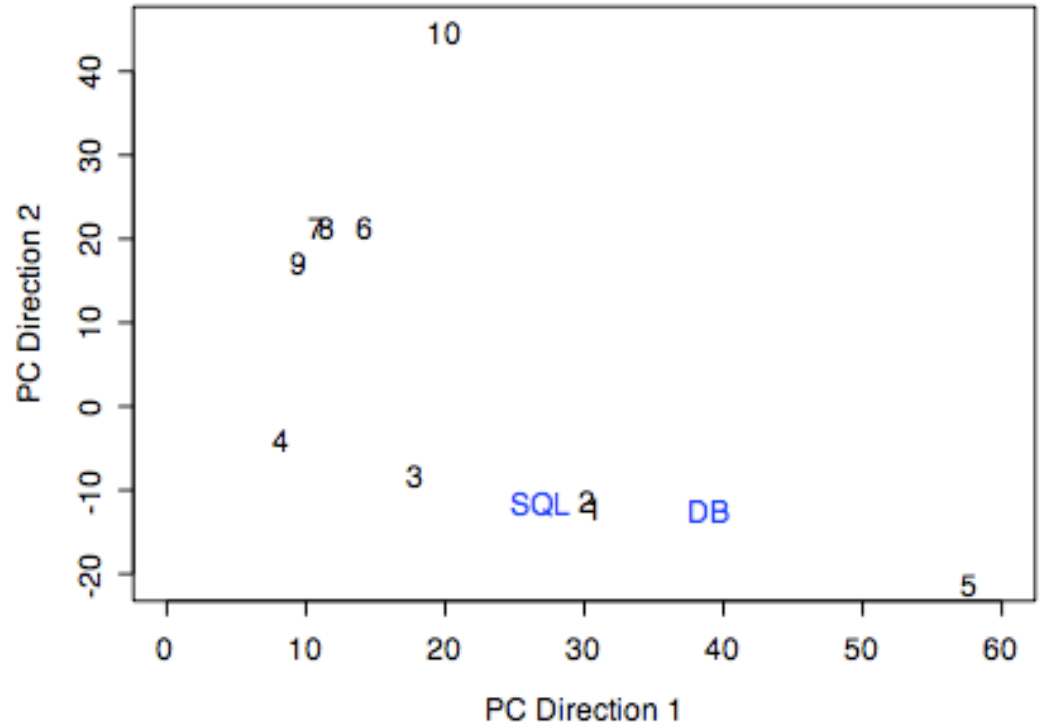
	[,1]	[,2]
[1,]	30.90	-11.5
[2,]	30.30	-10.8
[3,]	18.00	-7.7
[4,]	8.37	-3.5
[5,]	57.70	-20.6
[6,]	14.20	21.8
[7,]	10.80	21.9
[8,]	11.50	21.8
[9,]	9.50	17.8
[10,]	20.00	45.1



- Top 2 PC make new pseudo-terms to define documents...
- distance and angle (from the origin) shows similarity

LSI Example (2)

- Here we show the same plot, but with two new documents, one with the term “SQL” 50 times, another with the term “Databases” 50 times.
- Even though they have no phrases in common, they are close in LSI space



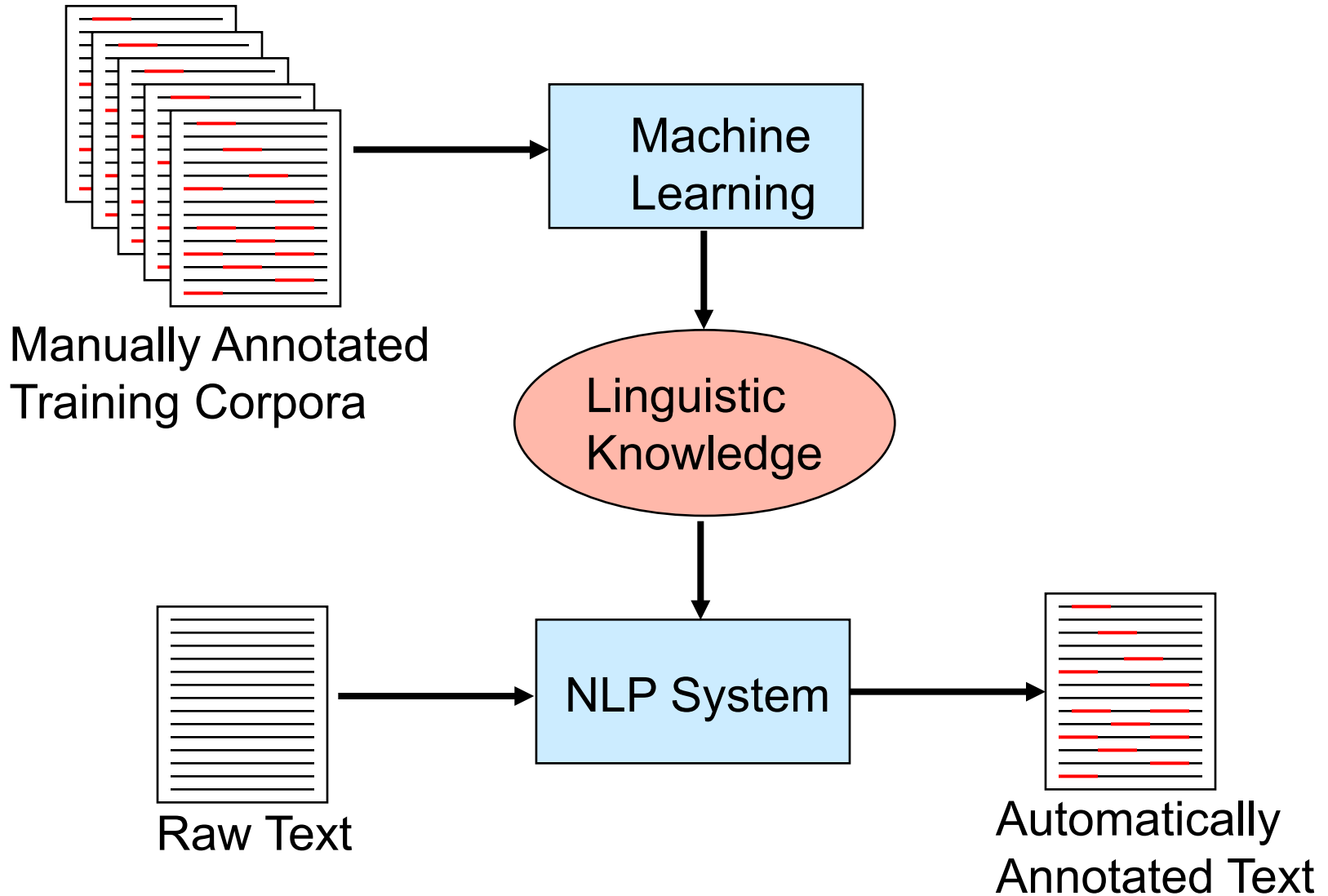
Manual Knowledge Acquisition

- Traditional, *rationalist*, approaches to language processing require human specialists to specify and formalize the required knowledge.
- Manual knowledge engineering, is difficult, time-consuming, and error prone.
- *Rules* in language have numerous exceptions and irregularities.
 - “All grammars leak.”: Edward Sapir (1921)
- Manually developed systems were expensive to develop and their abilities were limited and “brittle” (*not robust*).

Automatic Learning Approach

- Use machine learning methods to automatically acquire the required knowledge from appropriately annotated text corpora.
- Various referred to as the *corpus based*, *statistical*, or *empirical* approach.
- Statistical learning methods widely used in NLP and Speech processing

Learning Approach

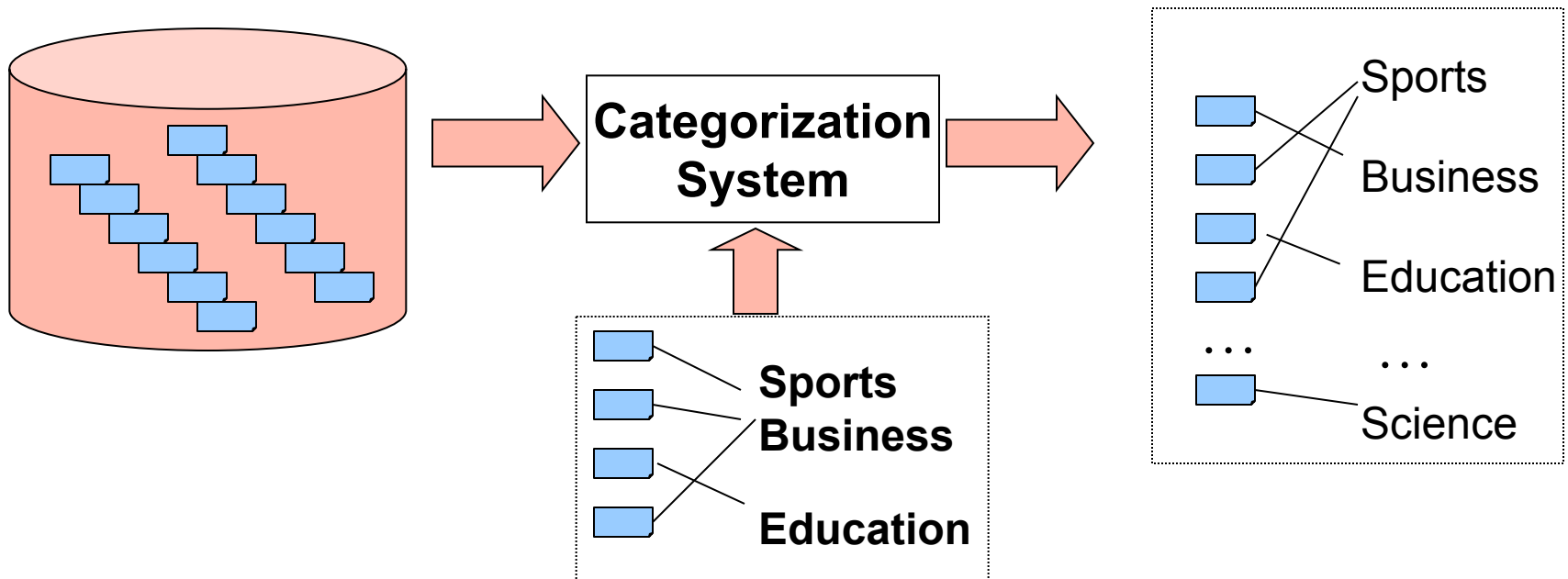


Advantages of the Learning Approach

- *Large amounts* of electronic text are now available.
- Annotating corpora is easier and requires less expertise than manual knowledge engineering.
- Learning algorithms have progressed to be able to handle large amounts of data and produce accurate *probabilistic knowledge*.
- The probabilistic knowledge acquired allows *robust* processing that handles linguistic regularities as well as exceptions.

Real World Case Study: Text Classification

- Pre-given categories and labeled document examples
(Categories may form hierarchy)
- Classify new documents
- A standard classification (supervised) learning problem



Text Classification (1)

■ Motivation

- Automatic classification for the large number of on-line text documents (Web pages, e-mails, corporate intranets, etc.)

■ Classification Process

- Data preprocessing
- Definition of training set and test set
- Creation of the classification model using the selected classification algorithm
- Classification model validation
- Classification of new/unknown text documents

Text Classification (2)

- Classification Algorithms: classes usually known
 - K-Nearest Neighbors
 - Naïve Bayes
 - Neural Networks
 - Decision Trees
 - Association rule-based
 - Boosting
 - Support Vector Machines

Text Classification with Tools

- Classification Algorithms:
<http://www.cs.waikato.ac.nz/ml/weka/>



Machine Learning Group at University of Waikato.

Project

Software

Book

Publications

People

Related

Home

Getting started

Requirements

Download

Documentation

FAQ

Citing Weka

Weka 3: Data Mining Software in Java

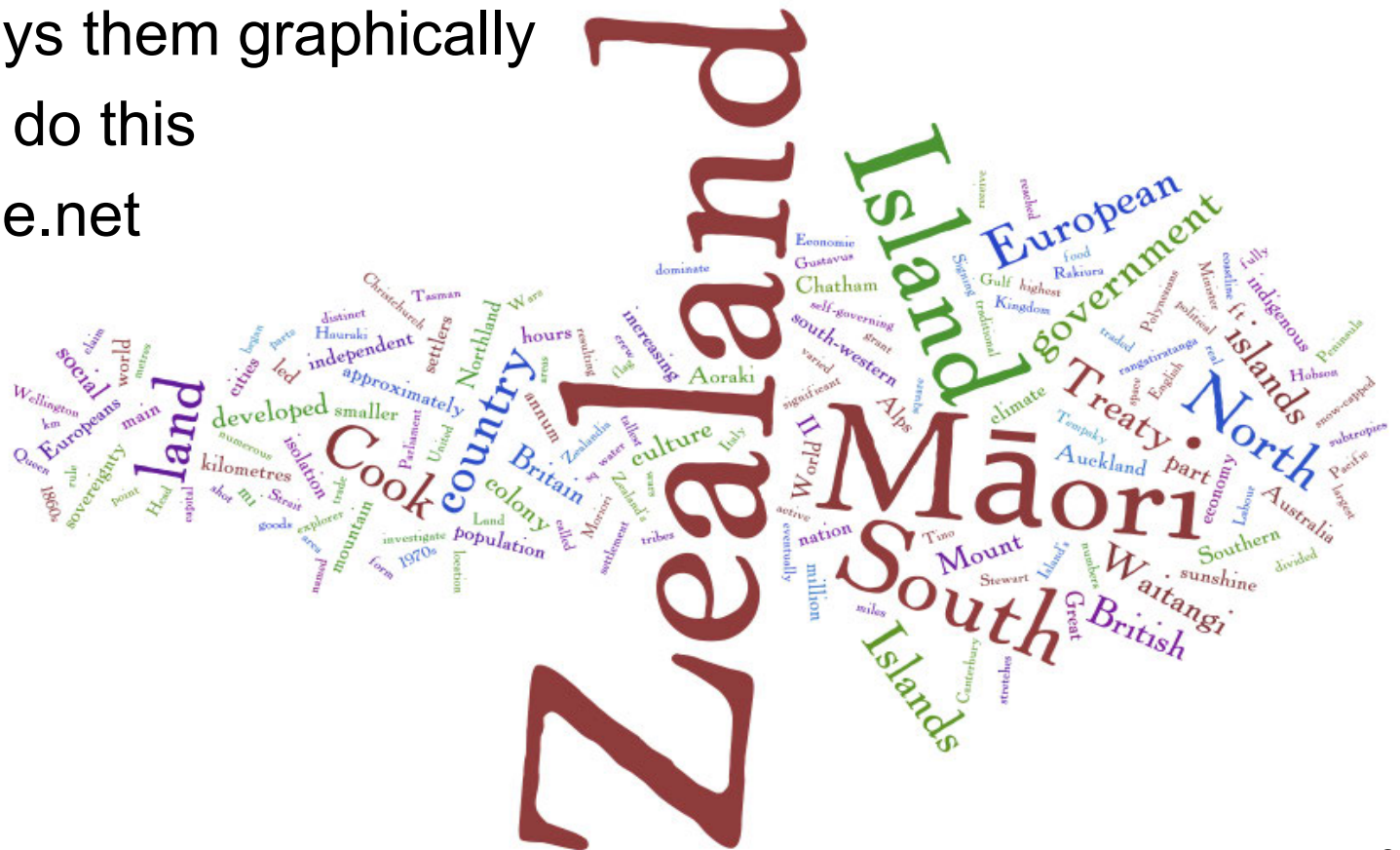
Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Weka is open source software issued under the **GNU General Public License**.

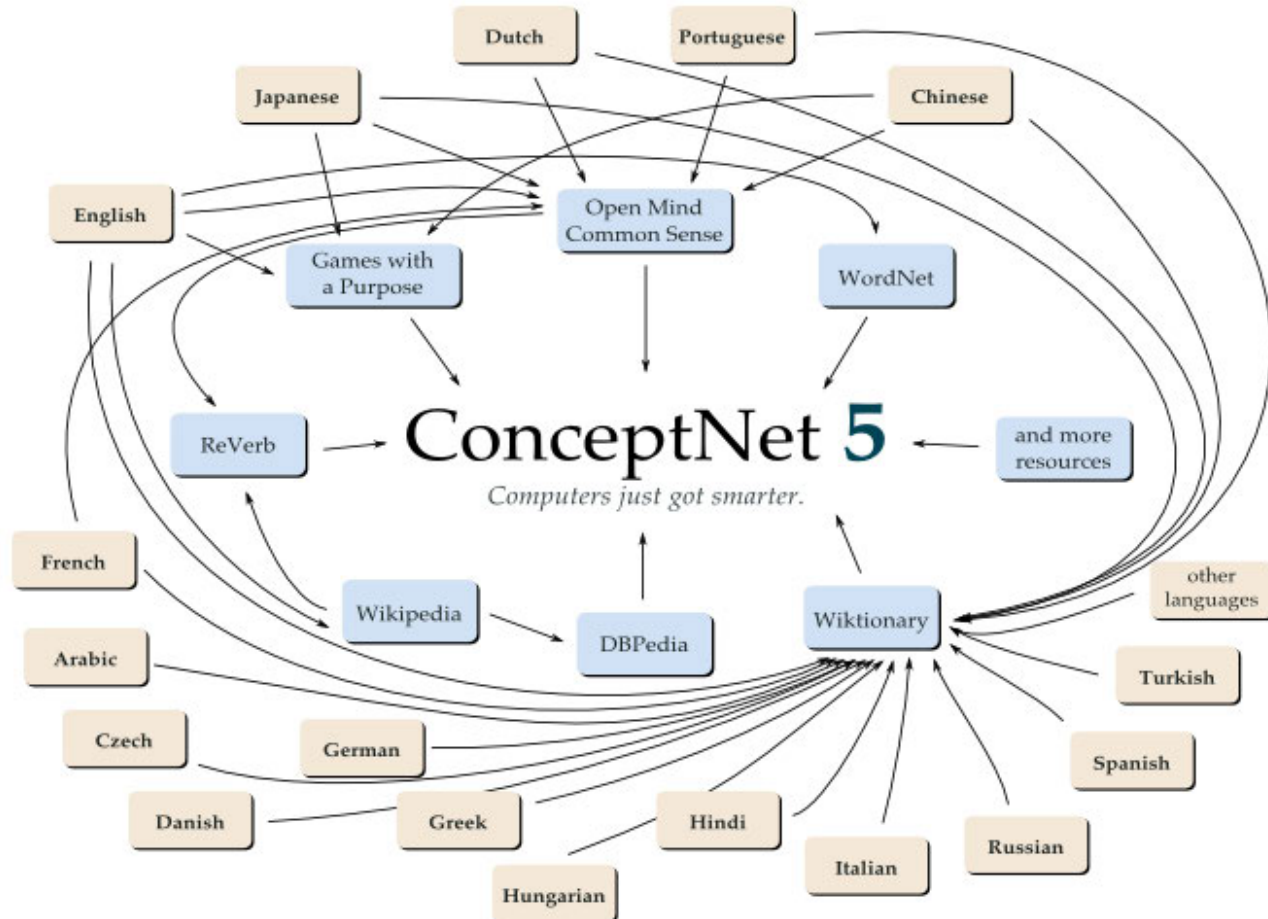
Current Developments and Pointers beyond Data and Text Mining...

- Summarizing text: Word Clouds

- Takes text as input, finds the most interesting ones, and displays them graphically
- Blogs do this
- Wordle.net



ConceptNet – an ontology with rich semantic Relationships

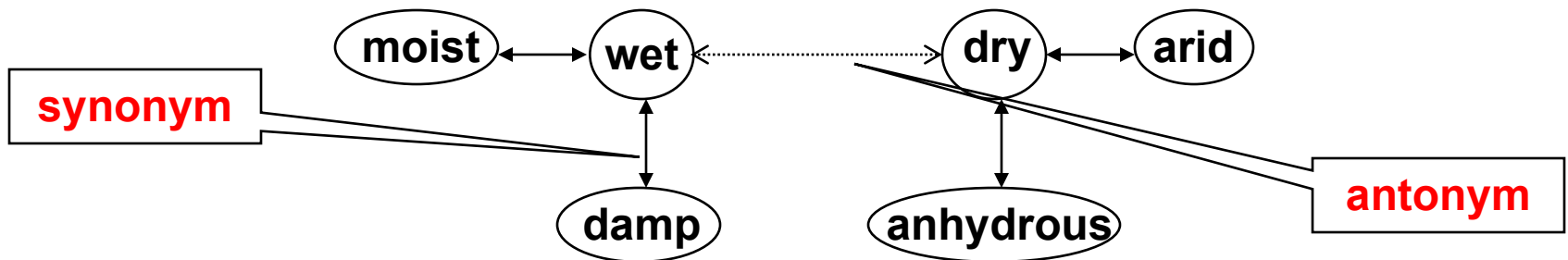


<http://conceptnet5.media.mit.edu/>

WordNet Lexicon

An extensive *lexical network* for the English language

- Contains over **138,838 words**.
- Several graphs, one for each *part-of-speech*.
- **Synsets** (sets of cognitive synonyms), each defining a semantic sense.
- **Relationship** information (antonym, hyponym, ...)
- Downloadable for **free** (UNIX, Windows)
- Founder **George Miller, National Medal of Science**.



<http://wordnet.princeton.edu/>

Meaning Bank



[GMB online Explorer](#)

[Downloads](#)

[Documentation](#)

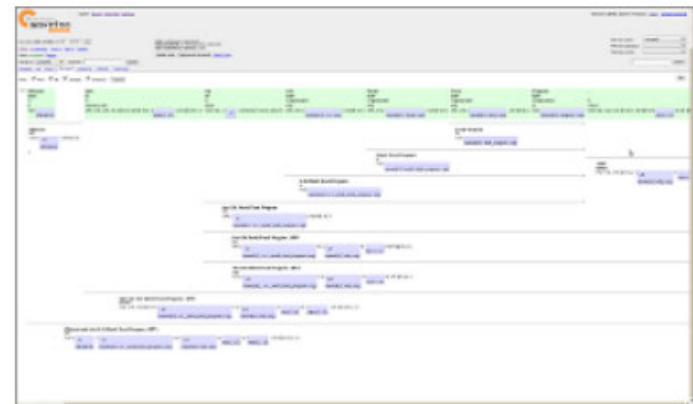
[Publications](#)

[People](#)

Groningen Meaning Bank

A free semantically annotated corpus that anyone can edit!

The current (development) version of the GMB is accessible via the [GMB Explorer](#), and comprises thousands of texts in raw and tokenised format, tags for part of speech, named entities and lexical categories, and discourse representation structures compatible with first-order logic.



BabelNet – a new multilingual ontology



A **very large multilingual ontology** with **5.5 millions** of concepts • A wide-coverage "**encyclopedic dictionary**" • Obtained from the automatic integration of **WordNet** and **Wikipedia** • Enriched with **automatic translations** of its concepts • Connected to the **Linguistic Linked Open Data** cloud!



<http://lcl.uniroma1.it/babelnet/index.jsp>

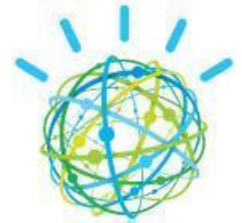
Further related projects

- CyC: Knowledge base and ontology framework to understand large collections of real-world concepts
 - Very important **question and answer system** in the 80s
 - Today an intensive library to build up knowledge bases that can **reason** things which were **never told**
 - Based on a **first-order predicate logic** extended by **modal operators** and higher order quantification
- RoboEarth: a world wide web and database repository for robots [www.roboearth.org]
 - **Cloud** robotics **infrastructure**: Data encoding, action & object recognition & labelling, learning
 - Research community aims to extend **data mining** and **learning** for autonomous robots to perform **complex tasks**



Watson and the DeepQA Text Mining project

Watson: computer system to compete in real time with expert humans in the Jeopardy Quiz



- Content acquisition: **Domain analysis**, automatic corpus expansion, leveraging of the content
- Question analysis: Parsing, lexical answer type detection, **semantic role labelling**, co-referencing, syntactic and semantic reasoning, decomposition
- Hypothesis generation: Get best candidates based on **search** and **constraint satisfaction**,
- Filtering, scoring and ranking: Machine learning and much more to estimate confidence.



Watson on Jeopardy!



Further reading:

- IBM's Watson/DeepQA: <http://dl.acm.org/citation.cfm?id=2019525>
- In the news: <http://www.bbc.co.uk/news/technology-20159531>

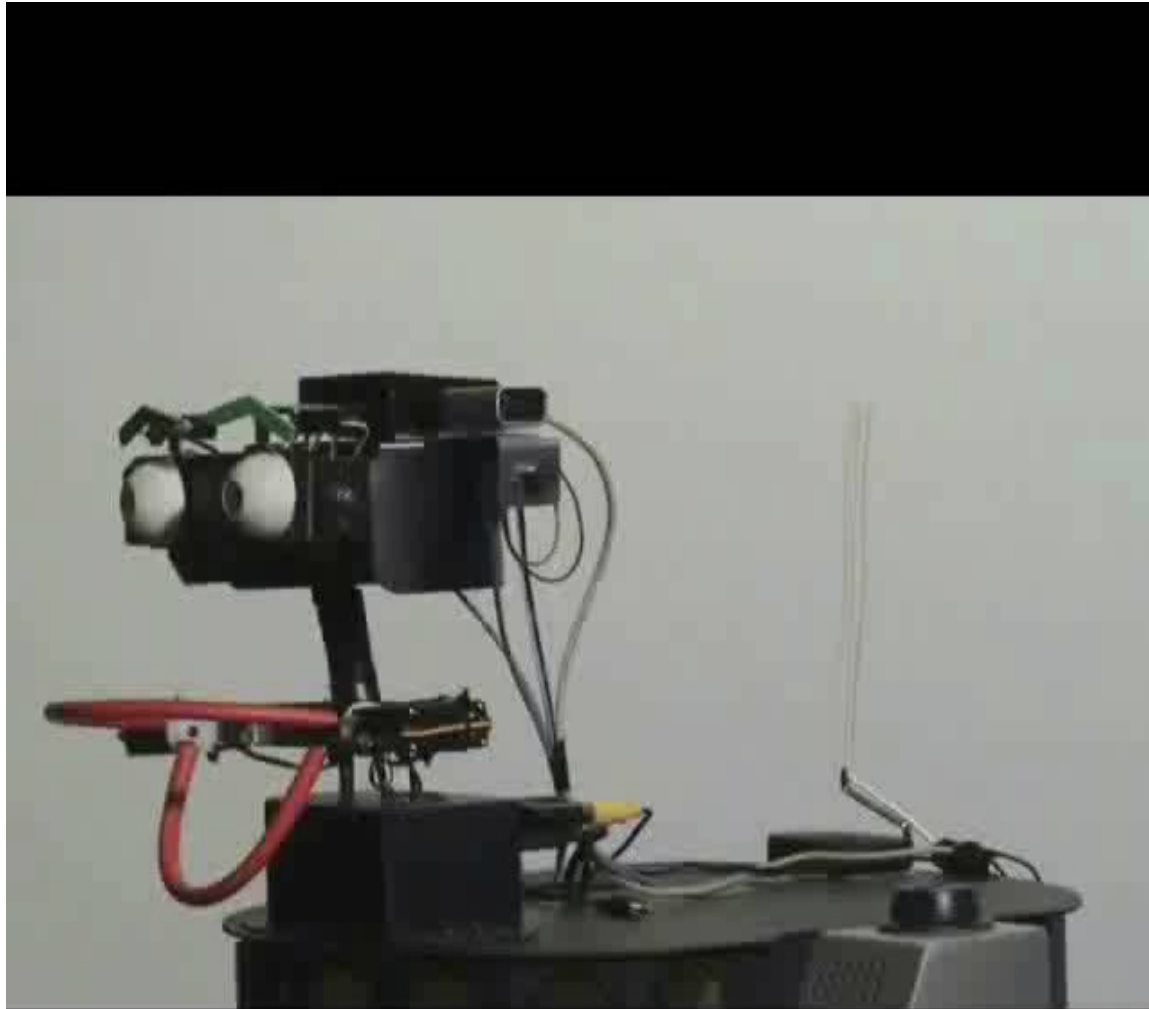
Summary

- Substantial portion of the available information gets more and more stored in text databases
 - Large collections of documents from various sources
 - Ext databases are rapidly growing
- Data in most text databases are **semistructured** data:
 - ... neither completely unstructured
 - ... nor completely structured
- Today's tools become increasingly essential
 - **Compare** different documents
 - **Rank** importance
 - **Find patterns** and trends


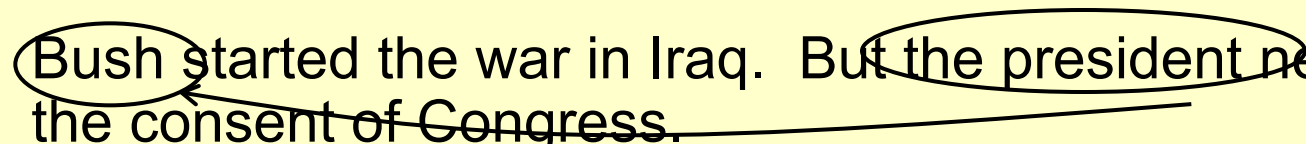
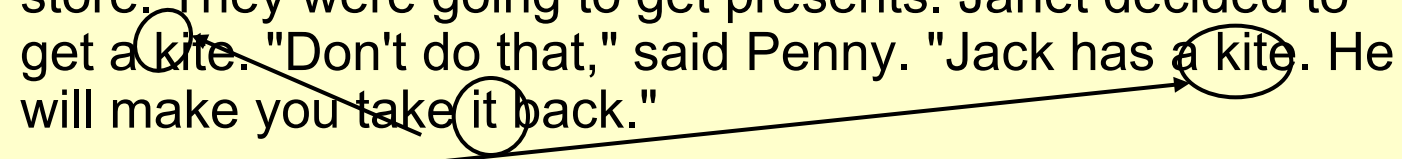
Examinations!

- Written Exams:
15. July and 29. September
- Registration:
Mon 23. Jun – Thu 3. Jul 2014; 9-15h; Studienbüro;
(exception: Studienbüro closed on Friday 27. Jun)

Finish with some fun?
(built by our team some time ago)



Anaphora Resolution/ Co-Reference

- Determine which phrases in a document refer to the same underlying entity.
 - John put the carrot on the plate and ate it.
 - Bush started the war in Iraq. But the president needed the consent of Congress.
- Some cases require difficult reasoning.
 - Today was Jack's birthday. Penny and Janet went to the store. They were going to get presents. Janet decided to get a kite. "Don't do that," said Penny. "Jack has a kite. He will make you take it back."

Summary: Hybrid Text Categorization

- Wide application domain
- Comparable effectiveness to professionals
 - Manual TC is not 100% and unlikely to improve substantially.
 - Automated TC is growing
- Prospects and extensions
 - Very noisy text, such as text from optical character recognition
 - Speech transcripts