~~Frontiers~~ Fundamentals of Text Identification and Retrieval

Jamie Callan
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
callan@cs.cmu.edu

What is Information Retrieval

- **The amount of information available is growing quickly, but human capacity remains constant**
- **IR creates tools for finding, organizing, summarizing, and analyzing and unstructured information**
  - E.g., monolingual and cross-lingual search engines
  - E.g., clustering, categorization
  - E.g., text summarization, question answering
  - E.g., topic detection, text mining, information distillation
- **IR does not require deep "understanding" of information**
  - **Key idea:** Similarity of a document to some other text
    - » Could be a query, a sentence, another document …

Areas of IR Related to Automatic Coding/Labeling of Text

- **Text similarity measures**
  - How similar are two chunks of text?
- **Automatic categorization**
  - How similar is a chunk of text to a model of a topic
- **Text extraction**
  - Identify the (small) segment of text that has certain properties

All of these can be done without deep understanding of meaning

Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Text Representation

Full Review

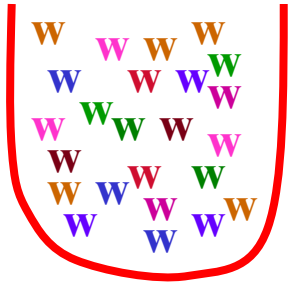
I have been looking and looking for a new camera to replace our bulky, but simple and reliable (but only fair picture taker) Sony Mavica FD73. My other choice (Besides the more expensive Nikon Coolpix 3100) was the (also more expensive) Sony Cybershot P72. I recommend any of these cameras, and I was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

– topjimmy5150, Apr 21, 2003, Epinions.com

This format isn't useful for many software applications

- So the next step is to transform the text into the target representation

Text Representation: The "Bag of Words" Representation



- **The simplest text representation is a "bag of words"**
  - The document is the "bag"
  - The "bag" contains word tokens (or other features)
    - » A particular word may occur more than once in the bag
  - Word order is ignored
  - Also called the "word histogram" approach
- **The bag of words can represent text of any size**
  - Document, paragraph, sentence, set of documents, …
- **This is a very simplistic approach to text representation**
  - But…surprisingly effective

Text Representation

Term	Tf	Term	Tf	Term	tf
the	78	up	8	pictures	6
to	35	for	7	red	6
i	31	have	7	digital	5
and	29	image	7	eye	5
a	19	like	7	not	5
camera	17	mode	7	on	5
is	17	much	7	or	5
in	12	software	7	shutter	5
with	11	very	7	sony	5
be	9	can	6	than	5
but	9	images	6	that	5
it	9	movies	6	after	4
of	9	my	6	also	4
this	9	no	6	: :	:

Are these terms useful?

What Goes into the Bag of Words?

Main Idea: Use features ("indexing terms") derived from the document

- **There are many heuristics for determining which features to use**
  - Stopword removal
  - Stemming
  - Phrase recognition
  - Named entity recognition
  - …

This is ad-hoc, but … text representation is usually the single greatest determiner of overall system effectiveness

Text Representation: Stopwords

- **Stopwords:** Words that are discarded from a document representation
  - Function words: a, an, and, as, for, in, of, the, to, …
  - Other frequent words: "IBM" in an IBM Customer Support db
  - There is no "master list" … typically adjusted for each task
- **Why remove stopwords? Isn't that throwing away information?**
  - Some argue against stopword removal for this reason

    … but it is common because it usually improves effectiveness
- **Removing stopwords makes some concepts impossible to recognize**
  - "Sit in", "Take over"

  …so the list must be developed carefully

Text Representation: Stopword Removal

Term	Tf	Term	Tf	Term	tf
camera	17	after	4	lcd	3
up	8	any	4	looking	3
image	7	auto	4	mavica	3
like	7	buy	4	problem	3
mode	7	flash	4	recorded	3
software	7	2100	3	reduction	3
images	6	bought	3	size	3
movies	6	button	3	zoom	3
pictures	6	down	3	15	2
red	6	feature	3	2mp	2
digital	5	focus	3	8x10	2
eye	5	included	3	98	2
shutter	5	lag	3	automatically	2
sony	5	last	3	batteries	2

Text Representation: Stopword Removal

morphological variants

Term	Tf	Term	Tf	Term	tf
camera	17	after	4	lcd	3
up	8	any	4	looking	3
image	7	auto	4	mavica	3
like	7	buy	4	problem	3
mode	7	flash	4	recorded	3
software	7	2100	3	reduction	3
images	6	bought	3	size	3
movies	6	button	3	zoom	3
pictures	6	down	3	15	2
red	6	feature	3	2mp	2
digital	5	focus	3	8x10	2
eye	5	included	3	98	2
shutter	5	lag	3	automatically	2
sony	5	last	3	batteries	2

Morphological Analysis: Word Stemming

- **Group morphological variants**
  - Examples: plurals, adverbs, inflected word forms
  - Grouping process is called “conflation”
- **Better than string matching**
  - Example: “river*” matches “river”, “rivers”, “riverdale”

Stemming Examples

- **Original Text**

  Document will describe marketing strategies carried out by U.S. companies for their agricultural chemicals, report predictions for market share of such chemicals, or report market statistics for agrochemicals.

- **Porter Stemmer (stopwords removed)**

  market strateg carr compan agricultur chemic report predict market share chemic report market statist agrochem

- **KSTEM (stopwords removed)**

  marketing strategy carry company agriculture chemical report prediction market share chemical report market statistic

Full-Text Indexing

Can you tell what this document is about?

Term	Tf	Term	Tf	Term	tf
camera	18	sony	5	lag	3
image	13	after	4	last	3
like	8	any	4	lcd	3
mode	8	auto	4	mavica	3
up	8	battery	4	record	3
buy	7	flash	4	reduce	3
movie	7	problem	4	size	3
picture	7	zoom	4	15	2
software	6	include	3	2mp	2
red	6	2100	3	8x10	2
digital	5	button	3	98	2
eye	5	down	3	automatic	2
look	5	feature	3	bag	2
shutter	5	focus	3	best	2

Phrases

- **Why use phrases?**
  - "interest rate" more precise than "interest, rate" or "interest AND rate"
- **There are three main methods for recognizing phrases**
  - A phrase dictionary (not covered)
  - Statistical recognition
  - Part-of-speech patterns

Phrase Recognition Methods: Statistical Recognition

- **Consider all word n-grams (sequences of n words)**
  - **Sentence:** "… recent interest rate hikes have …"
  - **Bigrams:** "recent interest", "interest rate", "rate hikes", "hikes have"
- **Evaluate by corpus term frequency (*ctf*) or document frequency (*df*)**
  - **E.g.,** "interest rate" (50), "rate hikes" (5), "hikes have" (7)
- **Discard less frequent bigrams**
- **Reasonably accurate, but makes mistakes**
  - If a pattern occurs often, it is probably a phrase
  - Counter-example: "recent interest" (43)
- **Very fast, used often**

Phrase Recognition Methods: Part of Speech Tagging

- **Assign part of speech tags**
  - Usually with a probabilistic or rule-based part of speech tagger
  - **Example:** "… recent/JJ interest/NN rate/NN hikes/NNS have/VBP …."
- **Match phrases by POS patterns**
  - Example: NN+ or JJ* NN+
- **More accurate (maybe)**
  - NN+: "interest rate"
  - JJ* NN+: "recent interest rate"
- **Reasonably fast, but slower than statistical recognition**
- **Used often**

JJ: Adjective
NN: Noun
NNS: Plural noun

Part-of-Speech Phrase Recognition: Top Phrases From a TREC Corpus

65,824 United States
61,327 Article Type
33,864 Los Angeles
18,062 Hong Kong
17,788 North Korea
17,308 New York
15,513 San Diego
15,009 Orange County
12,869 prime minister
12,067 Soviet Union
10,811 Russian Federation
9,912 United Nations
8,127 Southern California
7,640 South Korea
7,620 end recording
7,524 European Union
7,436 South Africa
7,362 San Francisco
7,086 news conference
6,792 City Council
6,348 Middle East
6,157 peace process
5,955 human rights
5,837 White House
5,778 long time
5,776 Armed Forces
5,636 Santa Ana
5,619 Foreign Ministry
5,527 Bosnia-Herzegovina
5,458 words indistinct
5,452 international community
5,443 vice president
5,247 Security Council
5,098 North Korean

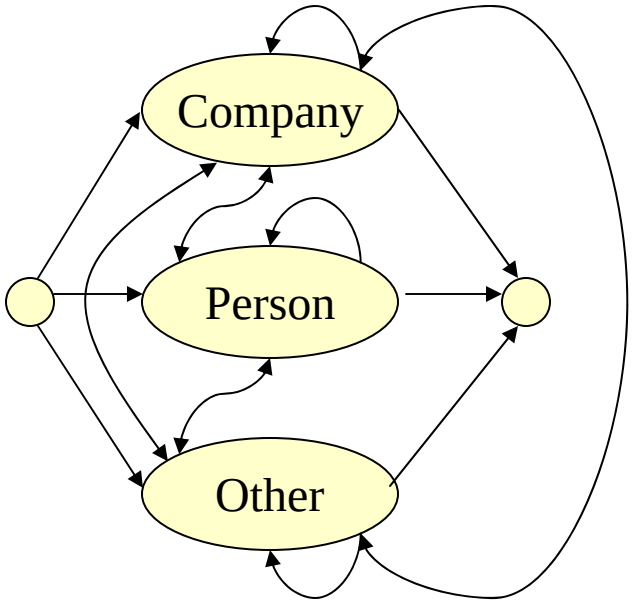
Named-Entities

- **Named entities are features that are names…**
  - Company names, people names, organization names, location names

  …or that behave like names
  - Monetary amounts, dates, telephone numbers, …

- **Named entity recognition is well-developed**
  - E.g., Hidden Markov Models (HMMs)
    - » Can be trained from manually-labeled data
    - » Fast, robust

Named Entity Recognition: Hidden Markov Models



P (w | Company)	
Apple	0.0100
apple	0.0001
Clinton	0.0001
:	:

P (w | Person)	
Apple	0.00010
apple	0.00001
Clinton	0.01000
:	:

President Clinton visited Apple Computer yesterday to announce ….
person person other company company other other other ….

(Bikel, et al, 1999)

Part of Speech and Named-Entity Text Annotations

In/IN the/DT early/JJ part/NN of/IN this/DT century/NN ,/, the/DT only/RB means/NNS of/IN transportation/NN for/IN travelers/NNS and/CC mail/NN between/IN Europe/NNP and/CC North/NNP America/NNP was/VBD by/IN passenger/NN steamship/NN ./. By/IN 1907/CD ,/, the/DT Cunard/NN Steamship/NNP Company/NNP introduced/VBD the/DT largest/JJS and/CC fastest/JJS steamers/NNS in/IN the/DT North/NNP Atlantic/NNP service/NN :/: the/DT Lusitania/NNP and/CC the/DT Mauritania/NNP ./. Each/DT had/VBD a/DT gross/JJ tonnage/NN of/IN 31,000/CD tons/NNS and/CC a/DT maximum/NN speed/NN of/IN 26/CD knots/NNS ./.

In the early part of this century, the only means of transportation for travelers and mail between **<LOCATION> Europe </LOCATION>** and **<LOCATION> North America </LOCATION>** was by passenger steamship. By <DATE> 1907 </DATE>, the **<COMPANY> Cunard Steamship Company </COMPANY>** introduced the largest and fastest steamers in the **<LOCATION> North Atlantic </LOCATION>** service: the **<NAME> Lusitania </NAME>** and the **<NAME> Mauritania </NAME>**. Each had a gross tonnage of **<WEIGHT> 31,000 tons </WEIGHT>** and a maximum speed of **<SPEED> 26 knots </SPEED>**.

– From K. Felkins, H.P. Leighly, Jr., and A. Jankovic. "The Royal Mail Ship Titanic: Did a Metallurgical Failure Cause a Night to Remember?" *JOM*, 50 (1), 1998, pp. 12-18.

What Goes into the Bag of Words?

Main Idea: Use features ("indexing terms") derived from the document

- **There are many heuristics for determining which features to use**
  - Stopword removal
  - Stemming
  - Phrase recognition
  - Named entity recognition
  - …

This is ad-hoc, but … text representation is usually the single greatest determiner of overall system effectiveness

Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Feature Weights

The "importance" of text features is indicated by weights

- **Tf.idf weights are a standard method of weighting features**
- **tf stands for term frequency**
  - Words that occur a lot in a document represent its meaning well
  - There are many heuristic methods of using tf…
  - E.g., Log (tf) + 1
  - E.g., tf / (tf + 0.5 + 1.5 * doclen / avg_doclen)
- **idf stands for inverse document frequency**
  - Words that occur in many documents aren't good at discriminating among documents
  - Log (N / df) + 1

Document Retrieval

- **Okay, we've got a bag of words derived from the document**
  - Now what?
- **Next: A quick overview of two popular retrieval models**
  - The vector space model
  - Statistical language models
- **These models are motivated by very different views of retrieval**
  - But they work very similarly
  - Historically, the vector space model has been dominant
  - Recently, statistical language models are becoming dominant

Retrieval Models

Vector Space Model

- **Any text object can be represented by a term vector**
  - Examples: Documents, queries, sentences, ….
- **Similarity is determined by distance in a vector space**
  - Example: The cosine of the angle between the vectors
- **The SMART system:**
  - Developed at Cornell University, 1960-1999
  - Used widely

Statistical Language Models

- **Any text object can be represented by a language model**
  - Examples: Documents, queries, sentences, ….
- **Similarity is determined in various ways**
  - $P(d_i \mid q)$ **(today)**
  - $P(q \mid d_i)$
  - Similarity of the probabilistic distributions q and d_i
- **The Lemur system:**
  - Developed at CMU, 2000-2007
  - Used widely

How Different Retrieval Models Use The Bag-of-Words Representation

Vector Space Model

$\vec{D}$

"a vector"

Word	tf(t,D)
camera	17
up	8
image	7
like	7
mode	7
software	7
images	6
movies	6
pictures	6
red	6
digital	5
eye	5
shutter	5
sony	5

Unigram Language Model

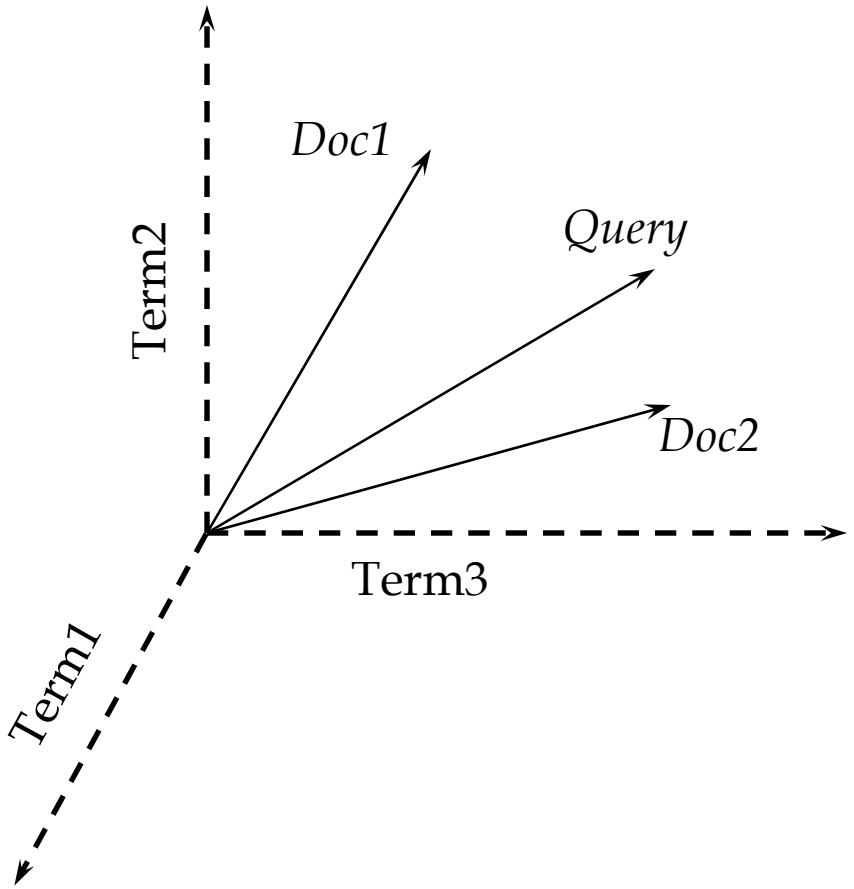
Word	$P(t\|\theta_D)$
camera	0.09551
up	0.04494
image	0.03933
like	0.03933
mode	0.03933
software	0.03933
images	0.03371
movies	0.03371
pictures	0.03371
red	0.03371
digital	0.02809
eye	0.02809
shutter	0.02809
sony	0.02809

$P(t|\theta_D) = tf(t,D)/length(D)$

θ_D

"a statistical language model"

Vector Space Similarity



Similarity is inversely related to the angle between the vectors.

***Doc2* is the most similar to the *Query*.**

Rank the documents by their similarity to the *Query*.

Vector Space Similarity

- **There have been many vector space similarity metrics…**
  …but the most common is the normalized cosine coefficient

$$\frac{\sum q_i \cdot d_i}{\sqrt{\sum q_i^{\,2}} \cdot \sqrt{\sum d_i^{\,2}}}$$

- **There have been many vector space weighting functions…**
  …a common choice is lnc.ltc

$$\frac{\sum d_i \cdot q_i}{\sqrt{\sum d_i^{\,2}} \cdot \sqrt{\sum q_i^{\,2}}} = \frac{\sum (\log(tf)+1) \cdot \left((\log(qtf)+1) \log \frac{N}{df} \right)}{\sqrt{\sum (\log(tf)+1)^2} \cdot \sqrt{\sum \left((\log(qtf)+1) \log \frac{N}{df} \right)^2}}$$

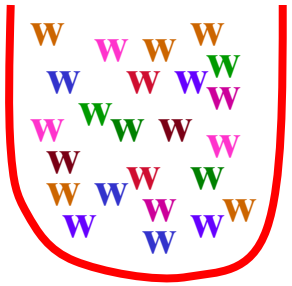
"tf" **user query weight** **"idf"**

doc length normalization **query length normalization**

Retrieval Models Based on Statistical Language Models

Model a document as an urn ("bag of words")

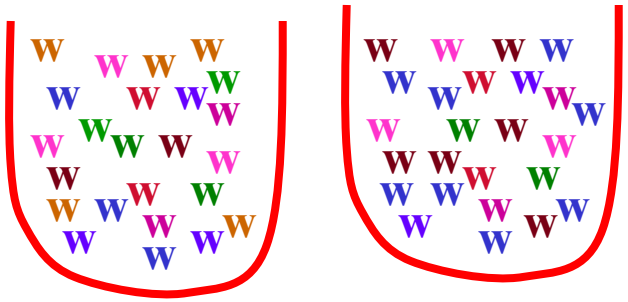
- **Each document defines a different urn**
  - A language model
- **Given an urn, what is P(w|D) for some word w?**
  - Maximum likelihood estimate: tf / doclen
  - tf: frequency of term w in document D
  - doclen: length of document D



Similarity of Statistical Language Models

How do we use this model to think about text similarity?

- **Let document 1 define one urn**
- **Let document 2 be another urn**
- **$P(D_2|D_1) = \Pi\ P(w_i|D_1)$**
  - Zero probabilities are a problem
- **$P(D_2|D_1) \approx \log \Sigma\ P(w_i|D_1)$**
  - Inaccurate, because words like "the" occur a lot
- **$P(D_2|D_1) \approx \log \Sigma\ [P(w_i|D_1) - E(P(w_i|D_1))]$**
  - What is the expected value of $P(w_i|D_1)$?
  - How about its probability in a large collection of documents?
- **$P(D_2|D_1) \approx \log \Sigma\ [P(w_i|D_1) - P(w_i|C)]$**
  - There are many more details, but this is the basic idea



Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

IR Applications

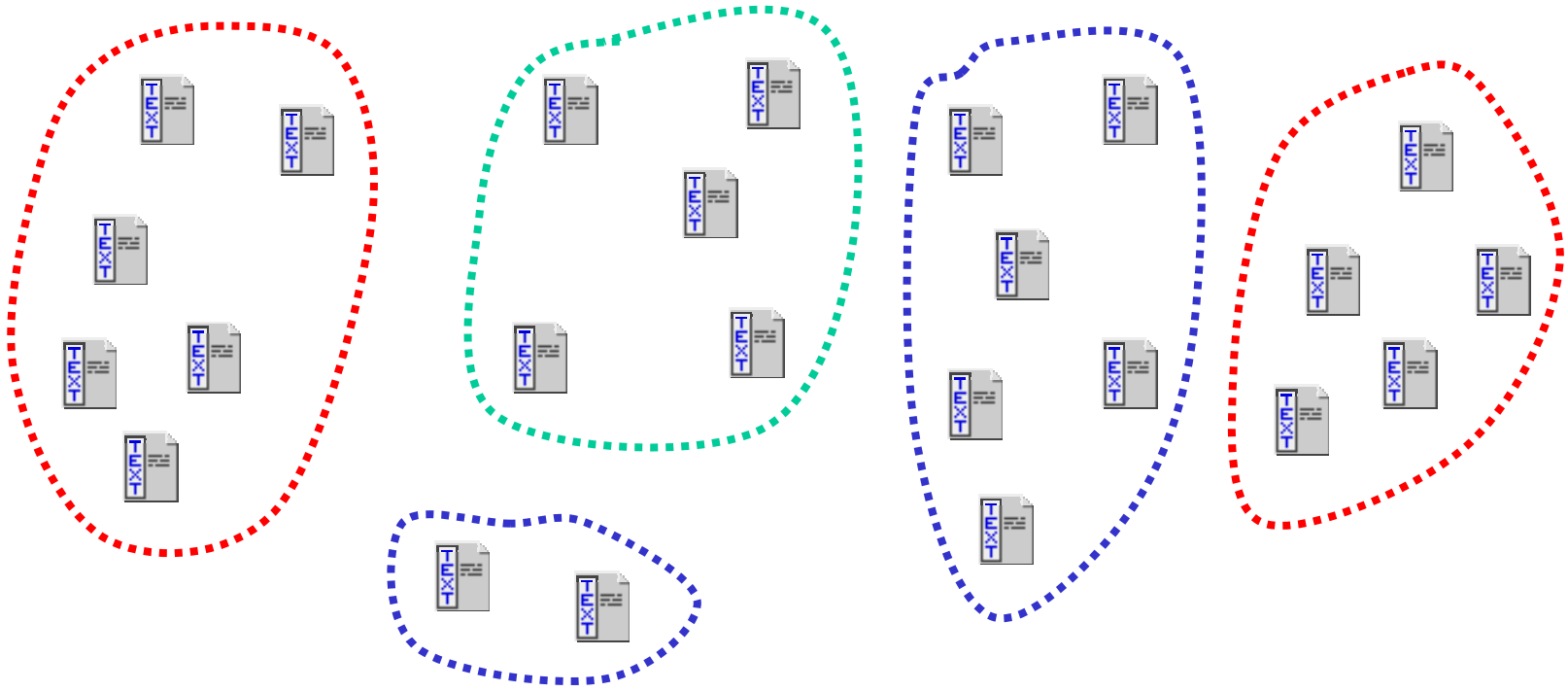
The ability to compare texts enables many applications…

- **Retrieval:** A query to a document
- **Summarization:** A sentence to a document
- **Clustering:** A document to other documents
- **Categorization:** A document to the language models of topics
- …

Introduction to Clustering

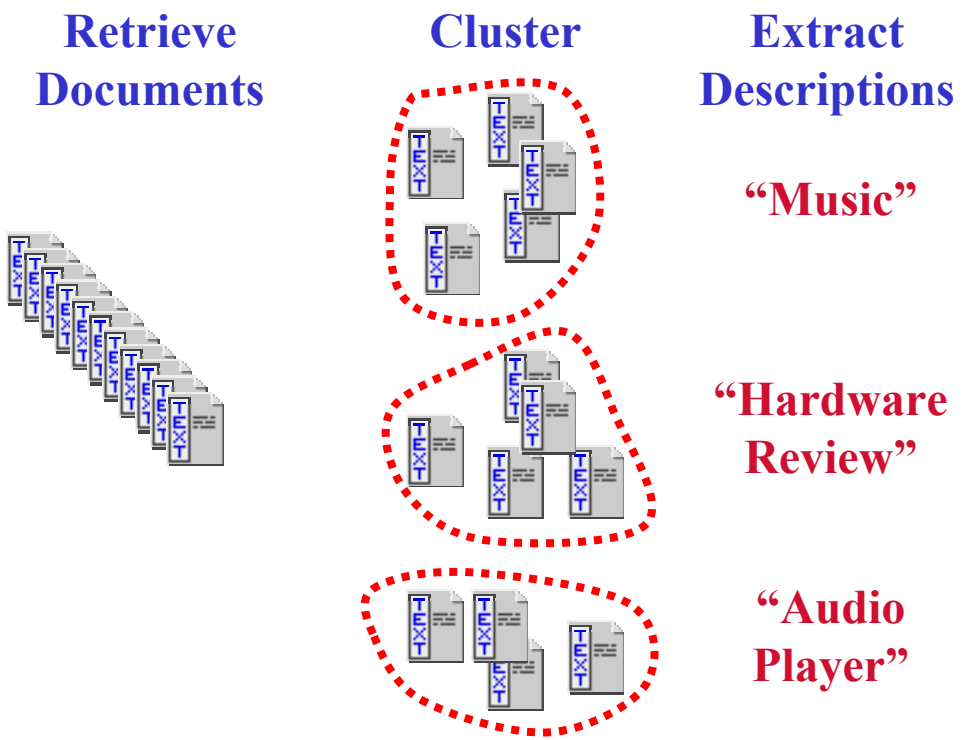
Clustering is often considered the simplest form of text mining

- **A clustering algorithm partitions a set of objects into subsets**
  - The objects in a subset are considered similar according to some metric



Clustering Example

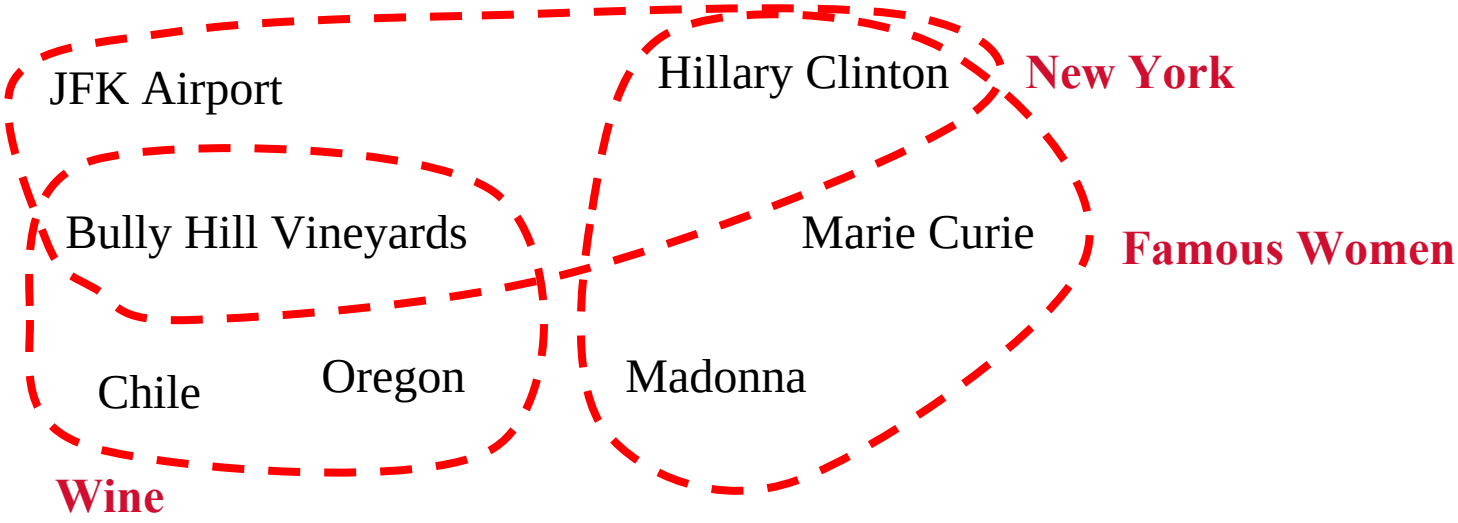
Clustering can organize retrieved documents



Clustering Example

Why is Clustering Hard?

- **Clustering algorithms reveal structure…but, what structure?**



Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Evaluation

Suppose you have two search engines

- Or, two methods of representing text

Which one works best?

Evaluation: The Cranfield Methodology

- **The Cranfield experimental methodology is the most common IR experimental methodology today**
  - Obtain a corpus of documents
  - Obtain a set of information needs
    - » Sometimes expressed as queries, sometimes not
  - Obtain relevance judgements indicating which documents satisfy (the information need expressed by) each query
    - » Requires a person to assess relevance of document to query
  - Measure how well each method does at finding relevant documents

Sample Research Test Collections

Characteristics of current test collections

- **Types of documents**
  - Excellent coverage of U.S. news
  - Some coverage of U.S. government data
  - Weak coverage of U.S. Web data
- **Only a few hundred queries per corpus with relevance judgements**

Characteristic	RCV1	WT10g	GOV2
Size (docs)	807 K	1.7 M	25 M
Size	2.5 GB	11 GB	427 GB
Year Created	2000	2000	2004
Stems	557 K	4.7 M	51.2 M
Stem Occurrences	203 M	1 B	22.8 B
Avg Doc Length	252	606	905
Queries	50	100	100

These collections are very useful…

…but they are a small, and somewhat biased sample of the world

- **"Your mileage may vary"**

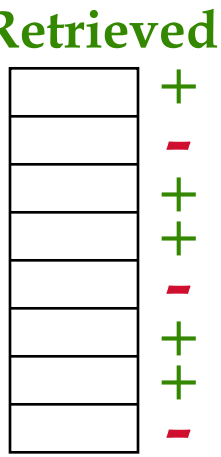
Basic IR Evaluation

- **Recall:**
  - The percentage of all relevant documents that are found by a search

$$R = \frac{\text{Number of relevant items retrieved}}{\text{Number of relevant items in collection}}$$

- **Precision**
  - The percentage of returned documents that are relevant

$$P = \frac{\text{Number of relevant items retrieved}}{\text{Number of items retrieved}}$$



R = 5/10 = 50%

P = 5/8 = 62.5%

+ Relevant

- Not relevant

Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Text Categorization

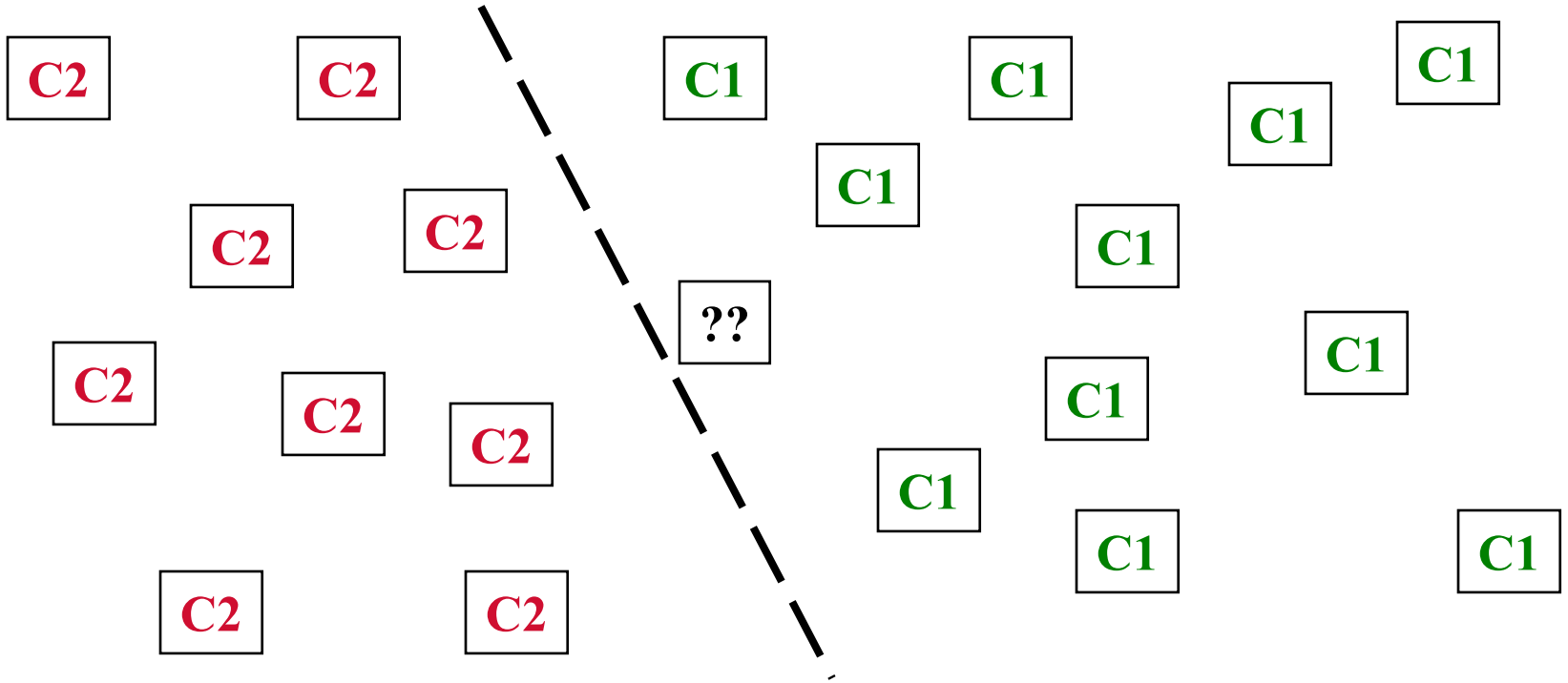
- **A set of training data is provided to a machine learning algorithm**
  - A set of representative objects, with labels
  - The larger the set, the better (usually)
  - The algorithm searches for patterns correlated with each label
  - Patterns are used to create a classifier
- **Good training data is crucial**
  - The labels must be assigned accurately and consistently
  - The objects must be described accurately and consistently
- **How should a text document be described?**
  - A bag of words representation is common

Machine Learning Algorithms

- **Many machine learning algorithms are used for text categorization**
- **Some (currently) popular choices**
  - k-Nearest Neighbor
  - Naïve Bayes
  - Logistic Regression
  - Support Vector Machines (SVMs)
  - …
- **Different methods often perform approximately equally**
  - Because the text representation limits what can be learned

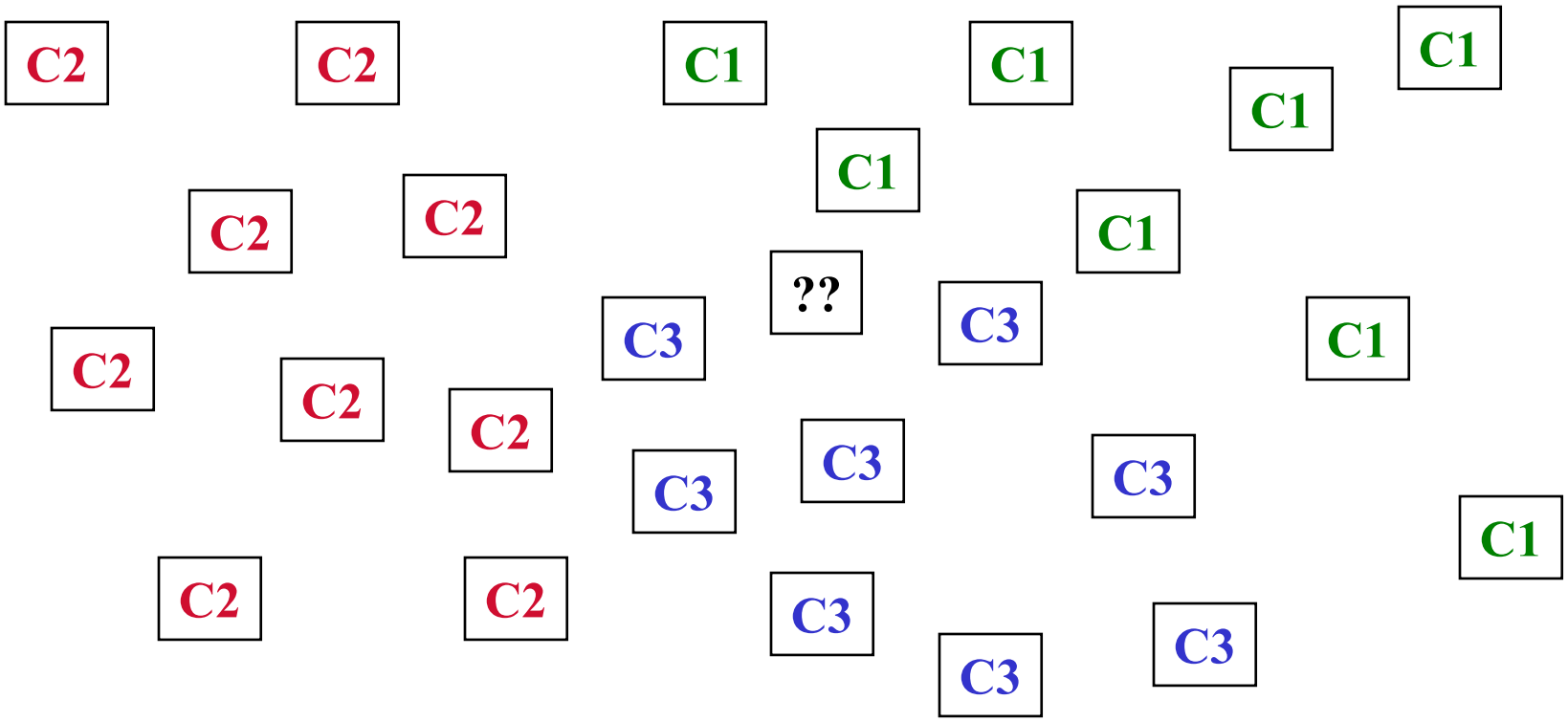
How Do Machine Learning Algorithms Work?

One approach is to learn a linear classifier that divides objects into two groups



How Do Machine Learning Algorithms Work?

Another approach is to find the *k* most similar objects in a training set. Assign the new object the same label(s).



Opinion-Recognition as Text Classification

A bad recommendation for Bank of America

Extracted Phrase	Part-of-Speech Tags	Semantic Orientation
little difference	JJ NN	-1.615
clever tricks	JJ NNS	-0.040
programs such	NNS JJ	0.117
possible moment	JJ NN	-0.668
unethical practices	JJ NNS	-8.484
low funds	JJ NNS	-6.843
old man	JJ NN	-2.566
other problems	JJ NNS	-2.748
probably wondering	RB VBG	-1.830
virtual monopoly	JJ NN	-2.050
other bank	JJ NN	-0.850
extra day	JJ NN	-0.286
direct deposits	JJ NNS	5.771
online web	JJ NN	1.936
cool thing	JJ NN	0.395
very handy	RB JJ	1.349
lesser evil	RBR JJ	-2.288
Average Semantic Orientation		-1.218

(Turney, 2002)

Automatic Categorization: What Makes it Hard?

- **Many similar categories**
- **Categories with small classes**
- **Hierarchical categorization**
- **Monothetic vs Polythetic categories:**
  - Human categories tend to be monothetic
    - » Every object shares one or more traits
    - » Monotheism is often conceptual, not vocabulary-based
    - » Example: Every document is about cancer
  - Machine learning categories are often polythetic
    - » Objects share a set of traits, but no trait is common to all
    - » Example: Documents contain words correlated with cancer

Automatic Categorization: State of the Art

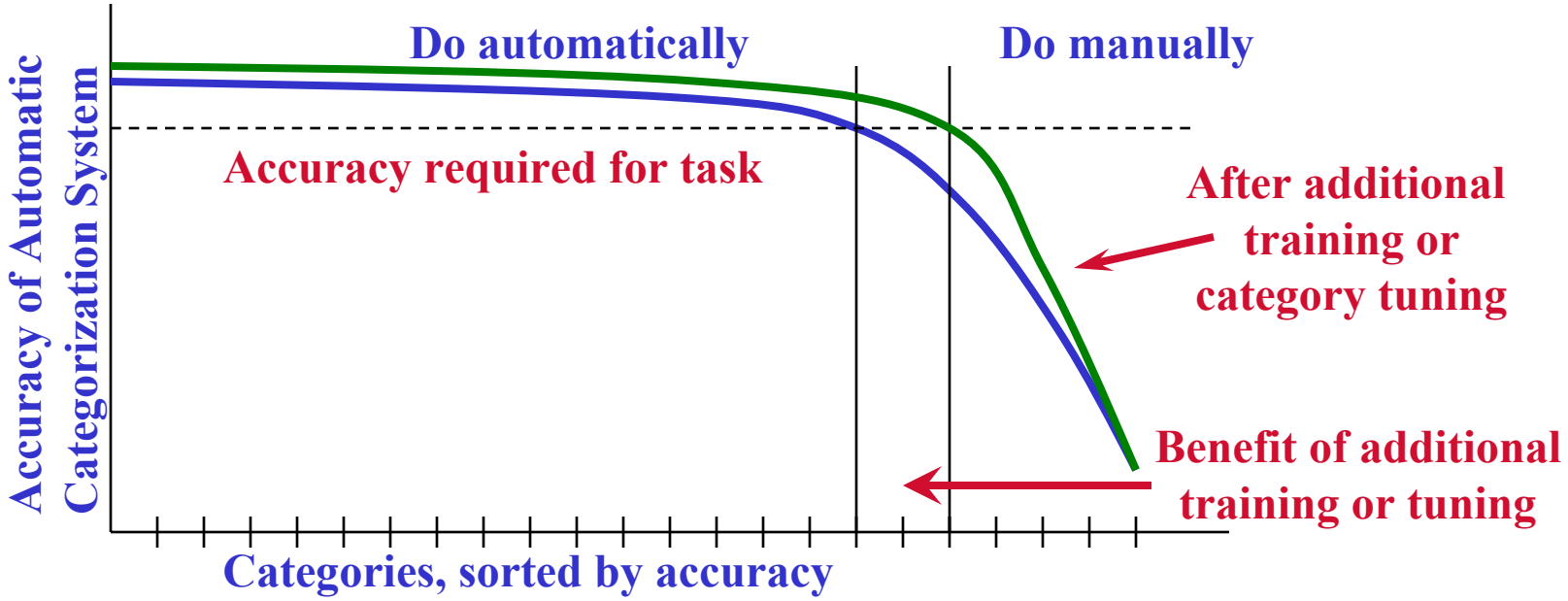
Task	Computers	Humans
Essay grading (e.g., GMAT)	96-97%	95%
Medical (OHSUMED, MESH)	50-60%	?
Medical (ICD9)	45-60%	?
Newswire (Reuters)	80-90%	?
Yahoo! Science categories	60-70%	?
Web pages	80-90%	?
Internet newsgroups	80-90%	?
TREC relevance assessments	?	70%

Automatic Categorization: Assessment

- **Humans are not perfect**
  - but human error-rate is often ignored
- **Computers are not perfect**
  - but computer error-rate is often discussed
- **Cost factors encourage greater use of automatic categorization**
  - automatic categorization in relatively easy domains
  - the 80/20 rule applies in some domains (80% automatic, …)
  - human-assisted categorization
- **Current algorithms appear reasonably accurate**
  - significant research activity, considerable progress

Automatic categorization is practical

Managing Text Categorization in a Business Setting



(Tom Mitchell)

Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Introduction to Information Extraction

Information extraction is the mapping of unstructured data (usually text) into a structured form

- **Some fields may be full-text (strings)**
  - Unrestricted
  - Typed (e.g., person name, noun)
- **Some fields may be derived data types**
  - Text → Controlled vocabulary (enumerated)
  - Text → Date
  - Text → Numeric range
  - …

Field Name	Data Type
Employer	String
City	String or Enumerated
State	Enumerated
Title	String or Enumerated
Salary	Numeric range
Education	String or Enumerated
: :	: :
: :	: :

A Sample Job Posting

https://acis.as.cmu.edu/gale2/open/servlet/TMSv2SEO?Form_Name=WebPosting&....

Job Posting Number: 1173

Title: SENIOR RESEARCH PROGRAMMER/ANALYST

Dept: LTI RESEARCH

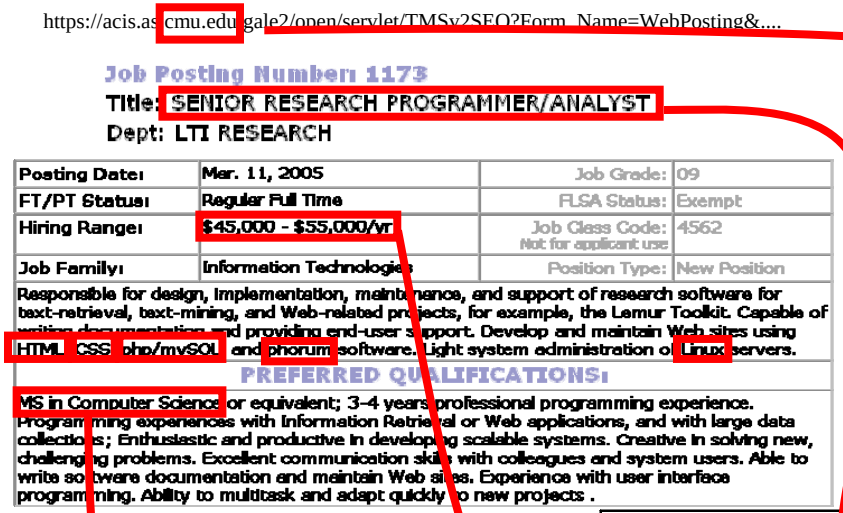
Posting Date:	**Mar. 11, 2005**	Job Grade:	09
FT/PT Status:	**Regular Full Time**	FLSA Status:	Exempt
Hiring Range:	**$45,000 - $55,000/yr**	Job Class Code: Not for applicant use	4562
Job Family:	**Information Technologies**	Position Type:	New Position

Responsible for design, implementation, maintenance, and support of research software for text-retrieval, text-mining, and Web-related projects, for example, the Lemur Toolkit. Capable of writing documentation and providing end-user support. Develop and maintain Web sites using HTML, CSS, php/mySQL and phorum software. Light system administration of Linux servers.

PREFERRED QUALIFICATIONS:

MS in Computer Science or equivalent; 3-4 years professional programming experience. Programming experiences with Information Retrieval or Web applications, and with large data collections; Enthusiastic and productive in developing scalable systems. Creative in solving new, challenging problems. Excellent communication skills with colleagues and system users. Able to write software documentation and maintain Web sites. Experience with user interface programming. Ability to multitask and adapt quickly to new projects .

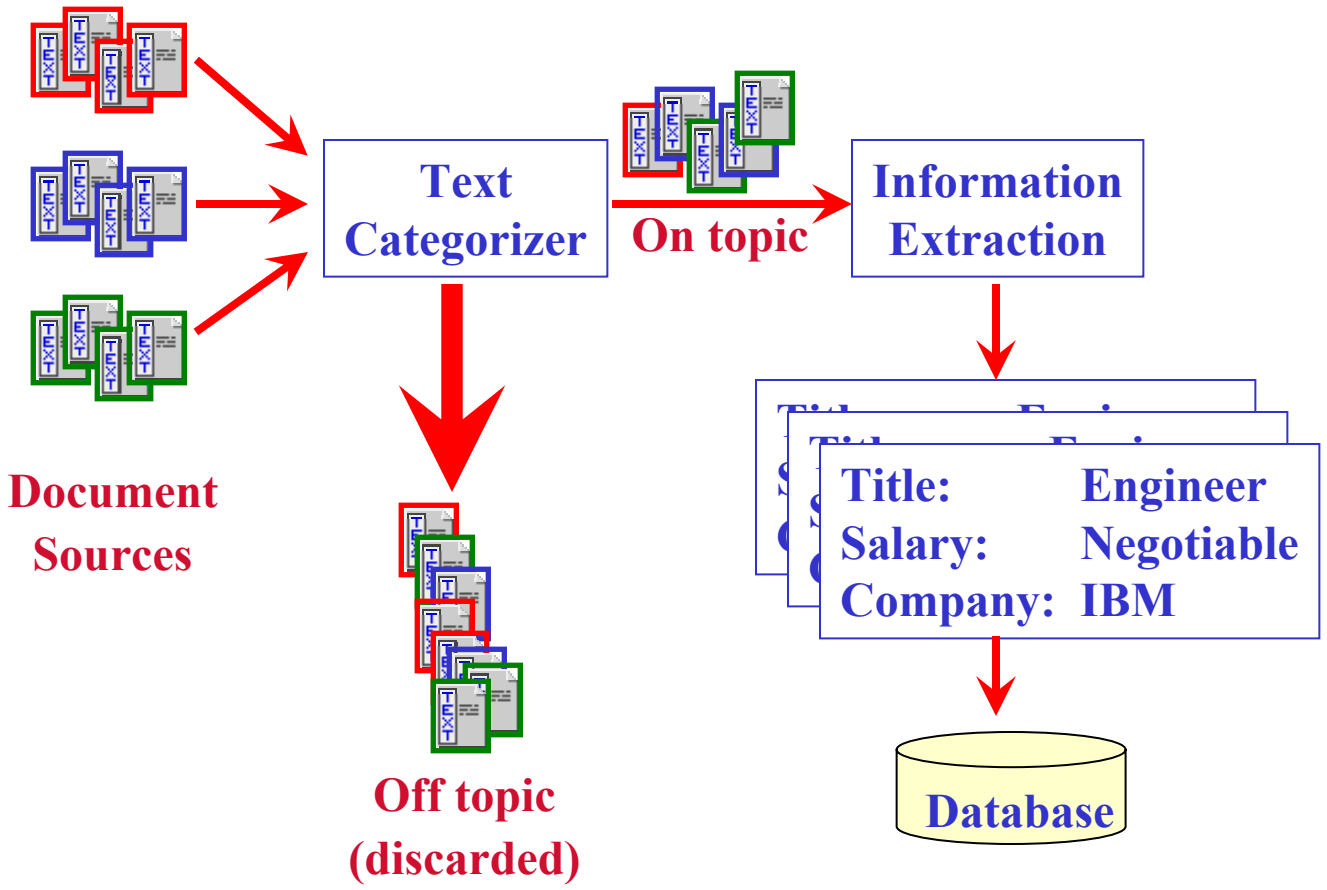
Transforming Unstructured Data Into Structured Data



Inference / lookup

Employer	Carnegie Mellon University
City	Pittsburgh
State	PA
Title	Senior Research Programmer/Analyst
Salary	45,000-55,000
Education	MS in Computer Science
OS	Linux
Web languages	HTML, CSS, php/mySQL, phorum

Simple Information Extraction Architecture



Automatic Creation of Extraction Patterns: AutoSlog

Linguistic Pattern (created manually)	Extraction Pattern (automatic specialization)	Slot Name (manual assignment)
<subject> passive-verb	<subject> was murdered	<victim>
<subject> active-verb	<subject> bombed	<perpetrator>
<subject> verb infin	<subject> attempted to kill	< perpetrator >
<subject> aux noun	<subject> was victim	<victim>
active-verb <direct object>	bombed <direct object>	<target>
infin <direct object>	to kill <direct object>	<victim>
verb infin <direct object>	tried to attack <direct object>	<target>
gerund <direct object>	killing <direct object>	<victim>
noun aux <direct object>	fatality was <direct object>	<victim>
noun prep <noun phrase>	bomb against <noun phrase>	<target>
active-verb prep <noun phrase>	killed with <noun phrase>	<instrument>

(Riloff, 1996)

Automatic Creation of Extraction Patterns: AutoSlog

How extraction patterns are created

- **Example sentence in text:**

  "Ricardo Castellar, the mayor, was kidnapped yesterday by the FMLN.

- **Ricardo Castellar is manually identified as a victim**
- **A syntactic parser labels words in the sentence**
  - "Ricardo Castellar" is the subject of the sentence
  - "was kidnapped" is a passive-verb
- **The sentence is compared to the stored patterns**
  - The "<subj> passive-verb" rule matches
  - The domain-specific pattern "<victim> was kidnapped" is created
  - Score patterns based on $si = (f_i / n_i) \log_2 f_i$
    - » F_i: number of times pattern matches relevant documents
    - » N_i: number of times pattern matches in all documents

AutoSlog:
Training Data

BOGOTA, **3 APR 90** (INRAVISION TELEVISION CADENA 1) – [REPORT] [JORGE ALONSO SIERRA VALENCIA][TEXT] LIBERAL SENATOR **FEDERICO ESTRADA VELEZ** WAS **KIDNAPPED** ON 3 APRIL AT THE CORNER OF 60TH AND 48TH STREETS IN WESTERN **MEDELLIN**, ONLY 100 METERS FROM A METROPOLITAN POLICE CAI [IMMEDIATE ATTENTION CENTER].

Incident Date	Human Target	Location
3 APR 90	Federico Estrada Velez	Medellin

Incident Type

Kidnapping

Text Representation

Part of Speech (POS) tagging:

Liberal/**NNP** senator/**NN** Federico/**NNP** Estrada/**NNP** Velez/**NNP** was/**VBD** kidnapped/**VBN** on/**IN** 3/**CD** April/**NNP** at/**IN** the/**DT** corner/**NN** of/**IN** 60/CD th/**DT** and/**CC** 48/**CD** th/**DT** streets/**NNS** in/**IN** western/**JJ** Medellin/**NNP** ,/, only/**RB** 100/**CD** meters/**NNS** from/**IN** a/**DT** metropolitan/**JJ** police/**NN** CAI/**NNP** (/(Immediate/**NNP** Attention/**NN** Center/**NN**)/) ./.

Named entity detection:

<ENAMEX type="**PERSON**">Federico Estrada Velez</ENAMEX>

<ENAMEX type="**LOCATION**">Medellin</ENAMEX>

<ENAMEX type="**ORGANIZATION**">CAI</ENAMEX>

Automatic Creation of Extraction Patterns: AutoSlog

- **A small amount of training data yields many domain-specific patterns**
  - Some are foolish, or too specific
- **Human review is necessary to reduce the set to just the patterns that are likely to be general (in the domain)**
  - Easy to for a highly trained person to do in a few hours
- **The resulting patterns are equivalent to high-quality patterns generated by humans**

Outline

- **Text similarity measures**
  - Text representation
  - Text similarity
  - Tasks that can be accomplished with this simple paradigm
  - Evaluation
- **Automatic categorization**
- **Text extraction**
  - Pattern-based methods

Whew! I hope that all made sense.

References: Text Books

- **Basic introduction to information retrieval**
  - R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*, Addison-Wesley, 2000.
  - D. A. Grossman, *Information Retrieval: Algorithms and Heuristics*, Springer, 2004.
- **More advanced introduction to information retrieval**
  - C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*, Cambridge University Press. Forthcoming. Draft available at http://www-csli.stanford.edu/~schuetze/information-retrieval-book.html.
- **Basic introduction to text mining**
  - S. Chakrabarti, *Mining the Web: Discovering Knowledge from Hypertext Data*, Morgan Kaufmann, 2002.
- **Advanced introduction to machine learning**
  - T.M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

References:
Research Papers

- J.G. Conrad and M.H. Utt. "A system for discovering relationships by feature extraction from text databases." In SIGIR-94 conference proceedings. Available as publication IR-45 at http://ciir.cs.umass.edu/.
- M. A. Hearst. "Untangling text data mining." *Proceedings of ACL'99: the 37th Annual Meeting of the Association for Computational Linguistics*. 1999.
- E. Riloff and J. Wiebe. "Learning Extraction Patterns for Subjective Expressions." *Conference on Empirical Methods in Natural Language Processing (EMNLP-03)*. ACL SIGDAT. Pages 105-112. 2003.

References:
Useful Software

- **Text analysis toolkit (many tools)**
  - LingPipe (http://www.alias-i.com/lingpipe/)
- **Machine learning toolkit (many learning algorithms)**
  - WEKA (http://www.cs.waikato.ac.nz/ml/weka/)
- **Word stemming libraries (for English)**
  - Porter stemmer (http://www.tartarus.org/~martin/PorterStemmer/)
  - KStem stemmer (http://www.lemurproject.org/)
- **Open source earch engines**
  - Lemur Toolkit (http://www.lemurproject.org/)
  - Lucene (http://lucene.apache.org/)

Appendix:
WEKA

What is WEKA?

- **WEKA is a machine learning workbench**
  - Tools for pre-processing data
  - Many popular machine learning algorithms
  - Data mining algorithms
  - Visualization tools
  - Experiment management tools
- **An open source package**
  - Available under the GNU Public License
- **Written in Java**
- **Often used with *Data Mining* by Witten and Frank**

WEKA Input:
Attribute Relation File Format (ARFF)

WEKA input must be in ARFF format:

- **A relation name**
  - E.g., @relation EducationLevel
- **A list of attribute definitions**
  - E.g., @attribute age real
  - E.g., @attribute sex {female, male}
  - E.g., @attribute degree {none, BS, MS, PhD}
  - The last attribute is the class to be predicted (the label/code)
- **A list of data elements**

  @data

  24, female, MS

  22, male, BS

WEKA Characteristics

- **WEKA is designed for traditional machine learning problems**
  - A moderate number of attributes (features) per element
    - » E.g., fewer than 100
  - A moderate number of values per attribute
    - » E.g., fewer than 10
    - » But real-valued attributes are okay
- **A typical text classification task has tens of thousands of features**
  - E.g., the size of the vocabulary
- **WEKA can probably run on problems with large feature sets**
  - E.g., using the sparse attribute representation scheme

  …but it would be very slow

Using WEKA on Text Problems

- **Transform text into ARFF representation (outside of WEKA)**
  - This is the bag of words representation
  - Discard stopwords, do stemming, etc ….
  - Do feature selection to reduce the size of the problem
    - » From thousands of text features to 100-200

  Text → feature selection → WEKA ARFF data
- **Use WEKA**
  - Read in data
  - Train the machine learning algorithm, using training data
  - Use the trained classifier to classify new, uncategorized data

  Very easy, very convenient