Data Mining:
Concepts and Techniques

— Chapter 10. Part 2 —
— Mining Text and Web Data —

Jiawei Han and Micheline Kamber
Department of Computer Science
University of Illinois at Urbana-Champaign
www.cs.uiuc.edu/~hanj

Mining Text and Web Data

- Text mining, natural language processing and information extraction: An Introduction
- Text categorization methods
- Mining Web linkage structures
- Summary

Mining Text Data: An Introduction

Data Mining / Knowledge Discovery

Structured Data

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

Multimedia

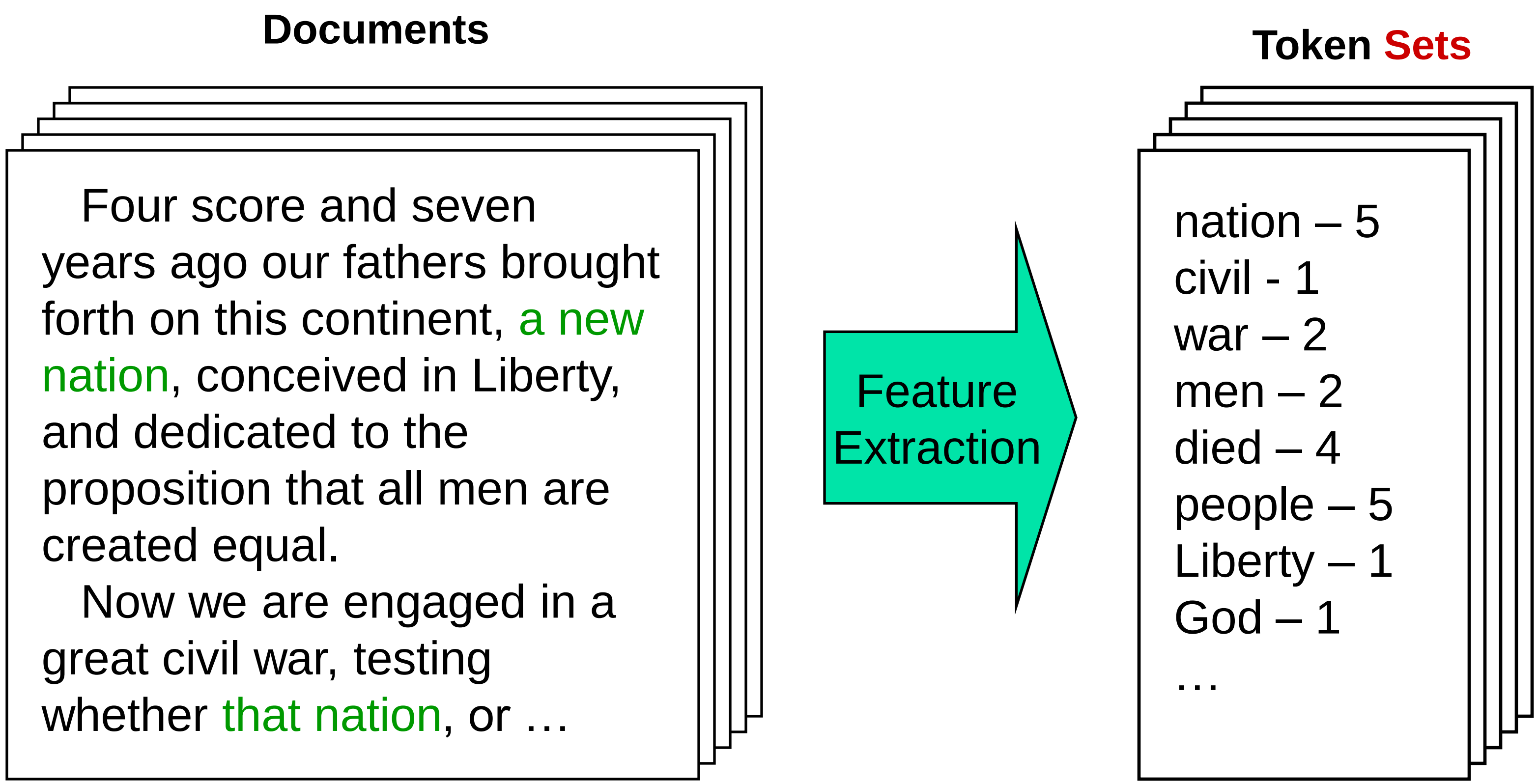
Free Text

Frank Rizzo bought his home from Lake View Real Estate in 1992.
He paid $200,000 under a15-year loan from MW Financial.

Hypertext

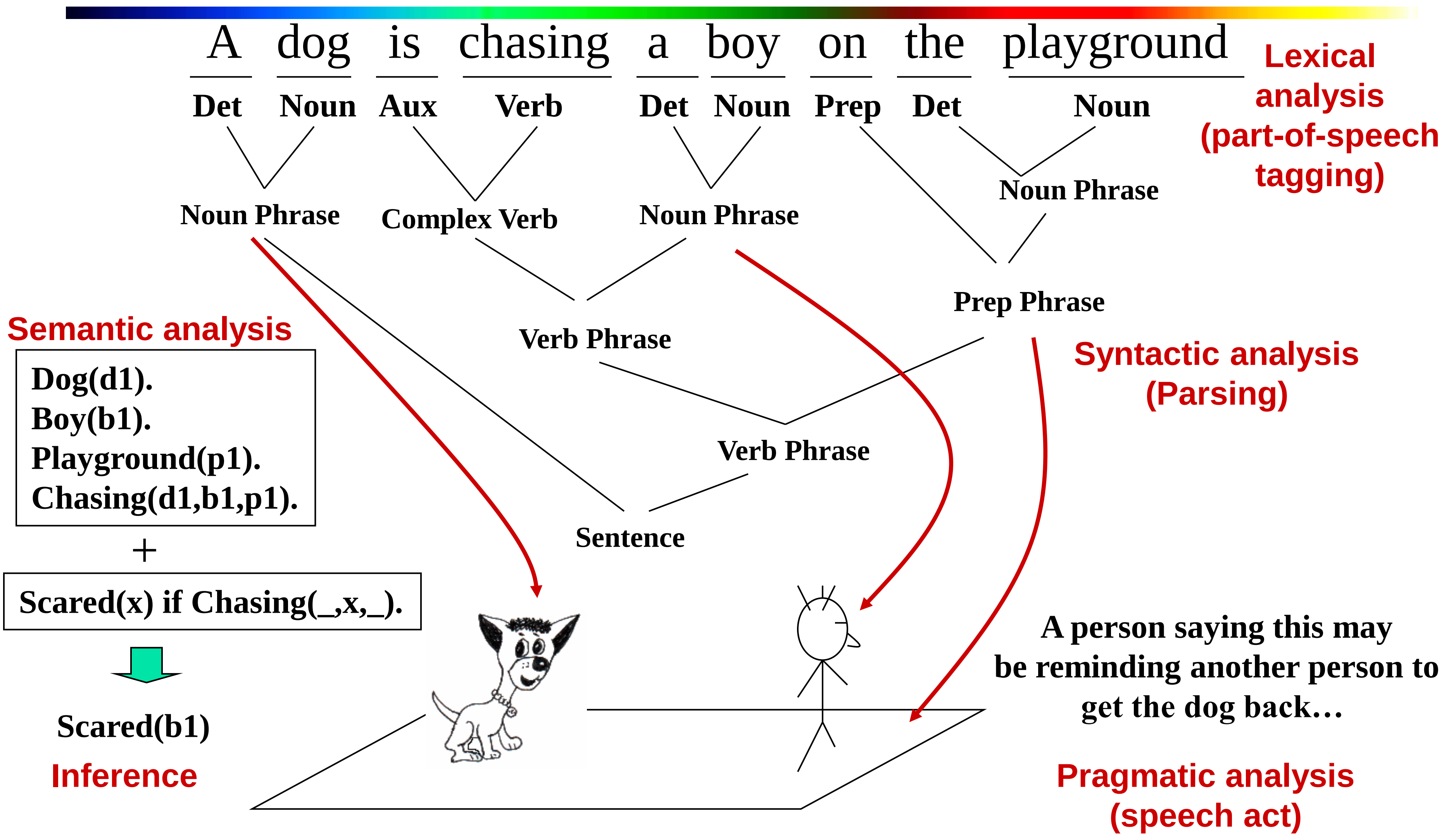
<a href>*Frank Rizzo* *Bought*
<a hef>*this home*
from <a href>*Lake View Real Estate*
In *1992*.
<p>...

Bag-of-Tokens Approaches



Loses all order-specific information!
Severely limits <u>context</u>!

Natural Language Processing



(Taken from ChengXiang Zhai, CS 397cxz – Fall 2003)

General NLP—Too Difficult!

- Word-level ambiguity
  - **"design" can be a noun or a verb** (Ambiguous POS)
  - **"root" has multiple meanings** (Ambiguous sense)
- Syntactic ambiguity
  - **"natural language processing"** (Modification)
  - **"A man saw a boy *with a telescope*."** (PP Attachment)
- Anaphora resolution
  - **"John persuaded Bill to buy a TV for *himself*."**
    (*himself* = John or Bill?)
- Presupposition
  - **"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!

(Taken from ChengXiang Zhai, CS 397cxz – Fall 2003)

Shallow Linguistics

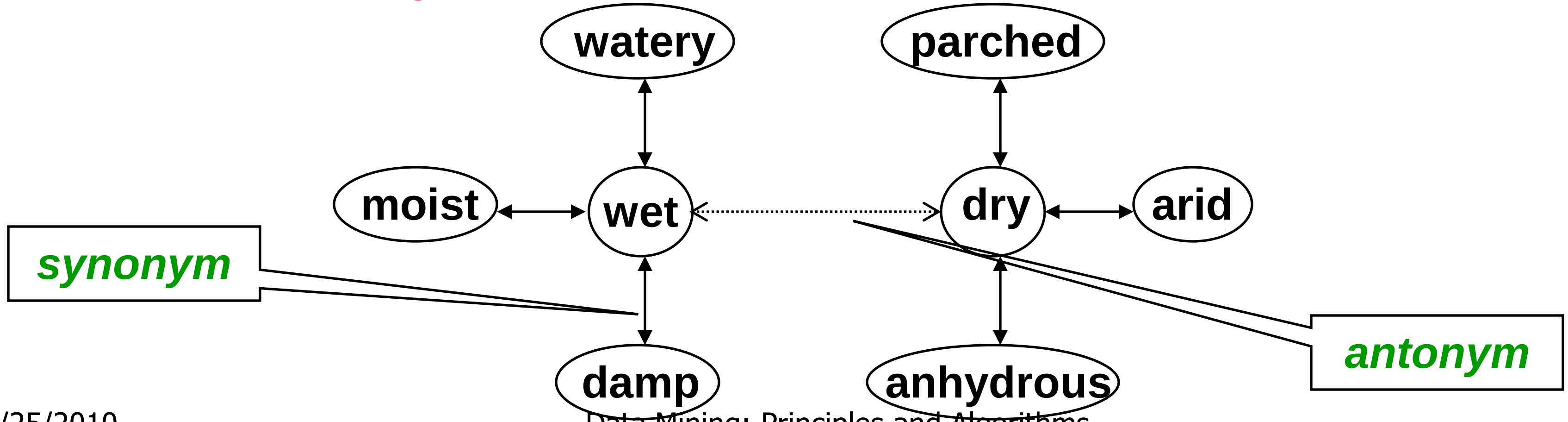
Progress on Useful Sub-Goals:

- English Lexicon
- Part-of-Speech Tagging
- Word Sense Disambiguation
- Phrase Detection / Parsing

WordNet

An extensive lexical network for the English language

- Contains over 138,838 words.
- Several graphs, one for each part-of-speech.
- *Synsets* (synonym sets), each defining a semantic sense.
- Relationship information (antonym, hyponym, meronym …)
- Downloadable for free (UNIX, Windows)
- Expanding to other languages (Global WordNet Association)
- Funded >$3 million, mainly government (translation interest)
- Founder George Miller, National Medal of Science, 1991.



Part-of-Speech Tagging

Training data (Annotated text)

This	*sentence*	*serves*	*as*	*an*	*example*	*of*	*annotated*	*text…*
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.

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

Independent assignment
Most common tag

Partial dependency
(HMM)

(Adapted from ChengXiang Zhai, CS 397cxz – Fall 2003)

Word Sense Disambiguation

?

*"The difficulties of computational linguistics are **rooted** in ambiguity."*

N Aux V P N

Supervised Learning

Features:

- Neighboring POS tags (N Aux V P N)
- Neighboring words (linguistics are rooted in ambiguity)
- Stemmed form (root)
- Dictionary/Thesaurus entries of neighboring words
- High co-occurrence words (plant, tree, origin,…)
- Other senses of word within discourse

Algorithms:

- Rule-based Learning (*e.g.* IG guided)
- Statistical Learning (*i.e.* Naïve Bayes)
- Unsupervised Learning (*i.e.* Nearest Neighbor)

Parsing

Choose most likely parse tree…

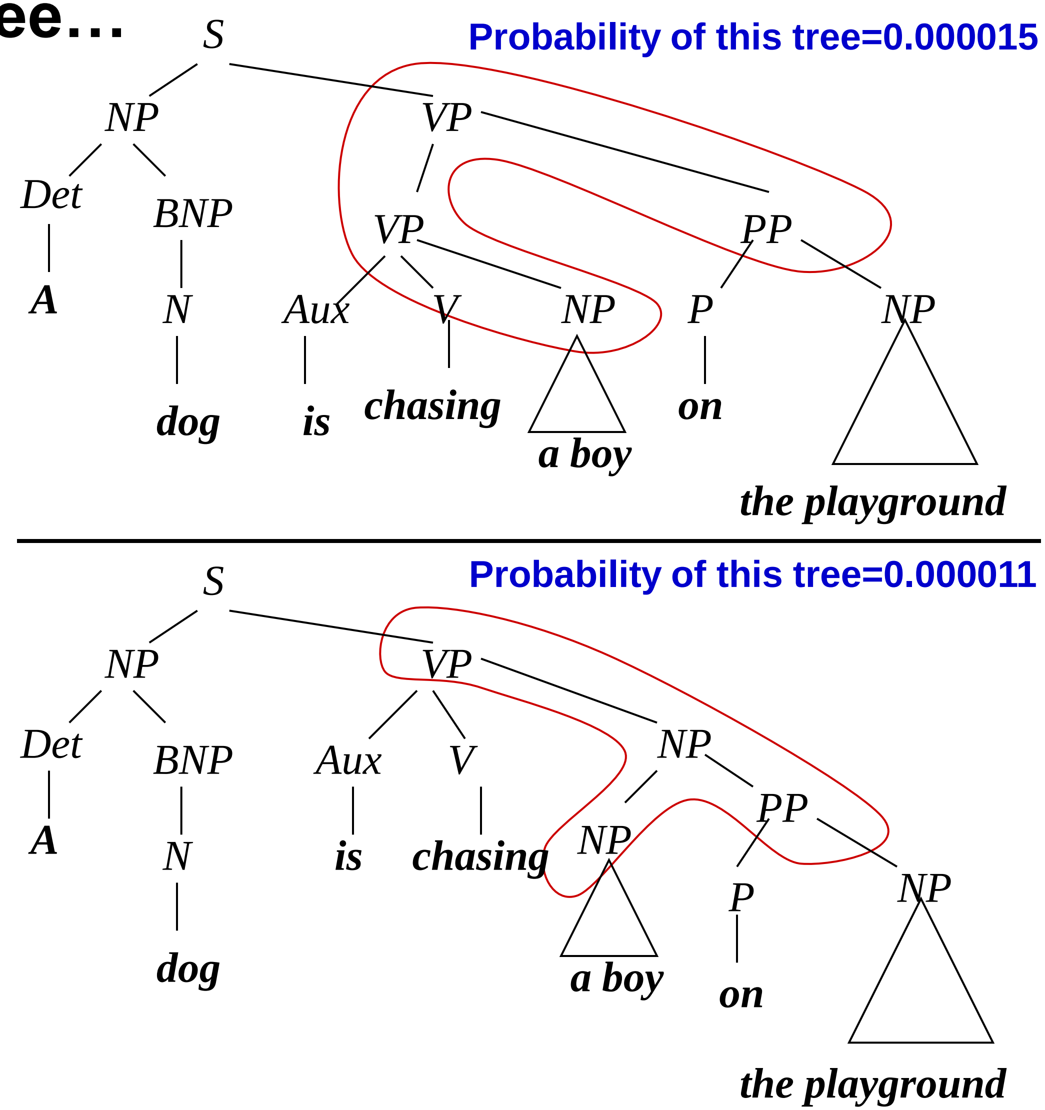
Probabilistic CFG

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
$N \rightarrow boy$	
$N \rightarrow playground$	…
$Det \rightarrow the$	
$Det \rightarrow a$	…
$P \rightarrow on$	



Obstacles

- **Ambiguity**

  "A man saw a boy with a telescope."

- **Computational Intensity**

  Imposes a context horizon.

Text Mining NLP Approach:

1. **Locate promising fragments using fast IR methods (bag-of-tokens).**
2. **Only apply slow NLP techniques to promising fragments.**

Summary: Shallow NLP

However, shallow NLP techniques are feasible and useful:

- **Lexicon – machine understandable linguistic knowledge**
  - possible senses, definitions, synonyms, antonyms, typeof, etc.
- **POS Tagging – limit ambiguity (word/POS), entity extraction**
  - **"***...**research interests** include **text mining** as well as **bioinformatics**."*

    NP **N**
- **WSD – stem/synonym/hyponym matches (doc and query)**
  - Query: *"Foreign cars"* Document: *"I'm selling a 1976 Jaguar…"*
- **Parsing – logical view of information (inference?, translation?)**
  - *"A man saw a boy with a telescope."*

Even without complete NLP, any additional knowledge extracted from text data can only be beneficial.

Ingenuity will determine the applications.

References for Introduction

1. C. D. Manning and H. Schutze, "Foundations of Natural Language Processing", MIT Press, 1999.
2. S. Russell and P. Norvig, *"Artificial Intelligence: A Modern Approach"*, Prentice Hall, 1995.
3. S. Chakrabarti, *"Mining the Web: Statistical Analysis of Hypertext and Semi-Structured Data"*, Morgan Kaufmann, 2002.
4. G. Miller, R. Beckwith, C. FellBaum, D. Gross, K. Miller, and R. Tengi. *Five papers on WordNet.* Princeton University, August 1993.
5. C. Zhai, *Introduction to NLP*, Lecture Notes for CS 397cxz, UIUC, Fall 2003.
6. M. Hearst, *Untangling Text Data Mining*, ACL'99, invited paper. *http://www.sims.berkeley.edu/~hearst/papers/acl99/acl99-tdm.html*
7. R. Sproat, *Introduction to Computational Linguistics*, LING 306, UIUC, Fall 2003.
8. A Road Map to Text Mining and Web Mining, University of Texas resource page. *http://www.cs.utexas.edu/users/pebronia/text-mining/*
9. Computational Linguistics and Text Mining Group, IBM Research, *http://www.research.ibm.com/dssgrp/*

Mining Text and Web Data

- Text mining, natural language processing and information extraction: An Introduction
- Text information system and information retrieval
- Text categorization methods
- Mining Web linkage structures
- Summary

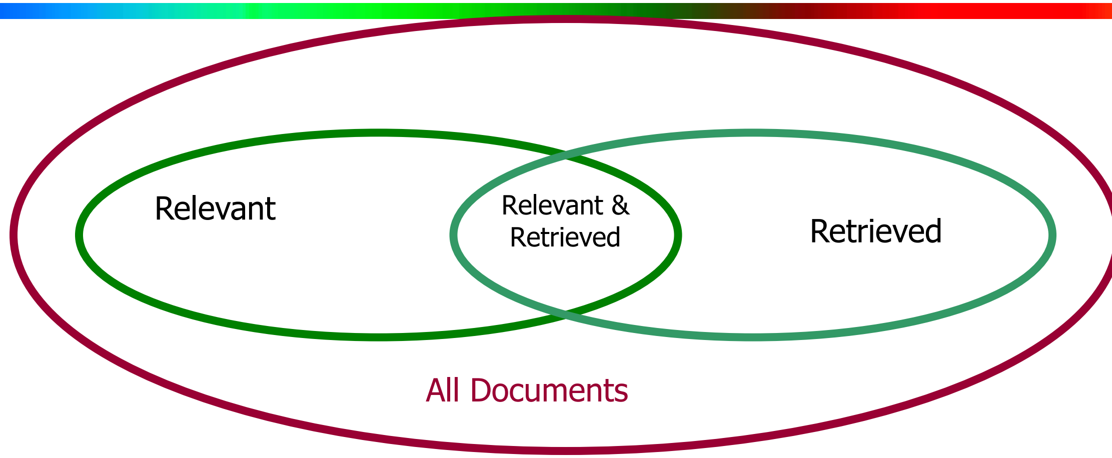
Text Databases and IR

- Text databases (document databases)
  - Large collections of documents from various sources: news articles, research papers, books, digital libraries, e-mail messages, and Web pages, library database, etc.
  - Data stored is usually *semi-structured*
  - Traditional information retrieval techniques become inadequate for the increasingly vast amounts of text data
- Information retrieval
  - A field developed in parallel with database systems
  - Information is organized into (a large number of) documents
  - Information retrieval problem: locating relevant documents based on user input, such as keywords or example documents

Information Retrieval

- Typical 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

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

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

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)
- DBMS Analogy
  - Index Terms → Attributes
  - Weights → Attribute Values

Information Retrieval Techniques

- Index Terms (Attribute) Selection:
  - Stop list
  - Word stem
  - Index terms weighting methods
- Terms $\times$ 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

Keyword-Based Retrieval

- A document is represented by a string, which can be identified by a set of keywords
- Queries may use expressions of keywords
  - E.g., car *and* repair shop, tea *or* coffee, DBMS *but not* Oracle
  - Queries and retrieval should consider synonyms, e.g., repair and maintenance
- Major difficulties of the model
  - Synonymy: A keyword *T* does not appear anywhere in the document, even though the document is closely related to *T*, e.g., data mining
  - Polysemy: The same keyword may mean different things in different contexts, e.g., mining

Similarity-Based Retrieval in Text Data

- 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.
- Basic techniques
- 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

Similarity-Based Retrieval in Text Data

- 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_i 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 distance:

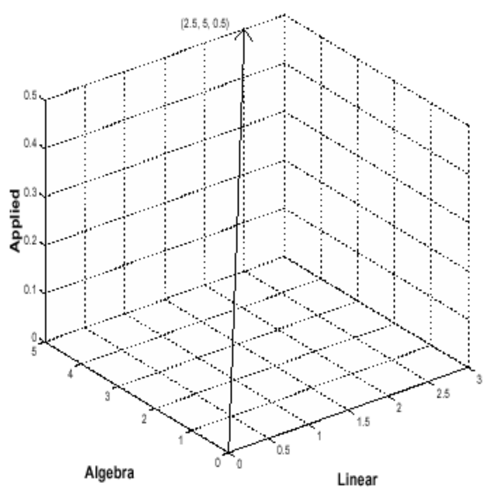
$$sim(v_1, v_2) = \frac{v_1 \cdot v_2}{|v_1||v_2|}$$

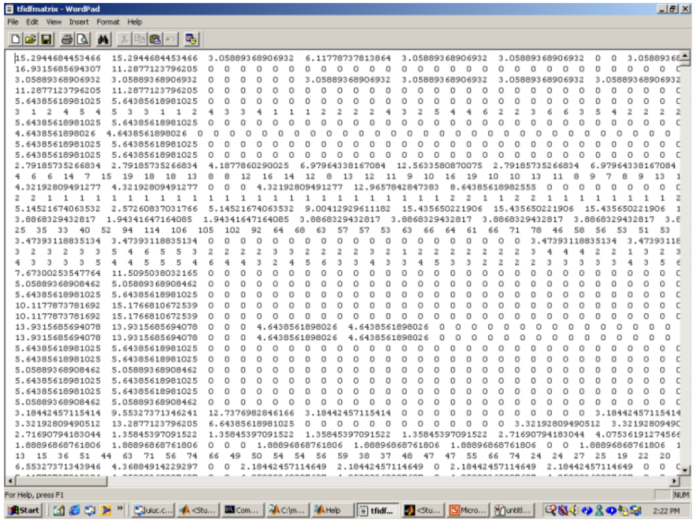
Indexing Techniques

- Inverted index
  - Maintains two hash- or B+-tree indexed tables:
    - document_table: a set of document records <doc_id, postings_list>
    - term_table: a set of term records, <term, postings_list>
  - Answer query: Find all docs associated with one or a set of terms
  - + easy to implement
  - – do not handle well synonymy and polysemy, and posting lists could be too long (storage could be very large)
- Signature file
  - Associate a signature with each document
  - A signature is a representation of an ordered list of terms that describe the document
  - Order is obtained by frequency analysis, stemming and stop lists

Vector Space Model

- Documents and user queries are represented as m-dimensional vectors, where m is the total number of index terms in the document collection.
- The degree of similarity of the document d with regard to the query q is 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.



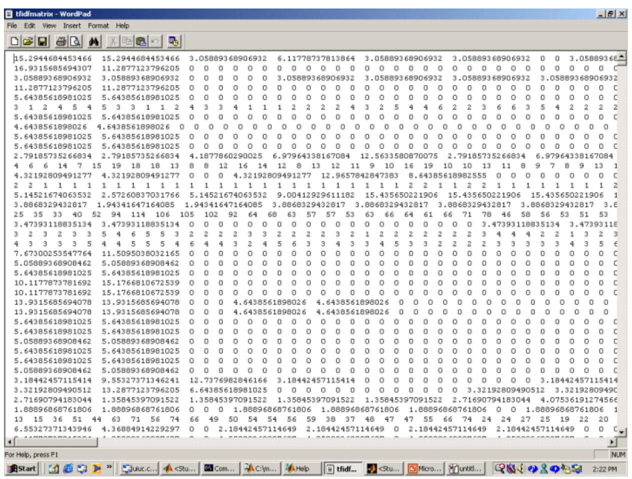


Latent Semantic Indexing

- Basic idea
  - Similar documents have similar word frequencies
  - Difficulty: the size of the term frequency matrix is very large
  - Use a singular value decomposition (SVD) techniques to reduce the size of frequency table
  - Retain the *K* most significant rows of the frequency table
- Method
  - Create a term x document weighted frequency matrix A
  - SVD construction: A = U * S * V'
  - Define K and obtain U_k ,, S_k , and V_k.
  - Create query vector q' .
  - Project q' into the term-document space: Dq = q' * U_k * S_k^{-1}
  - Calculate similarities: cos α = Dq . D / ||Dq|| * ||D||

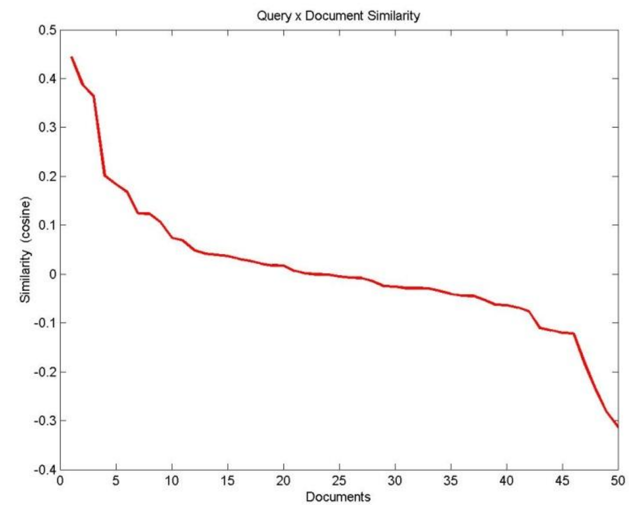
Latent Semantic Indexing (2)

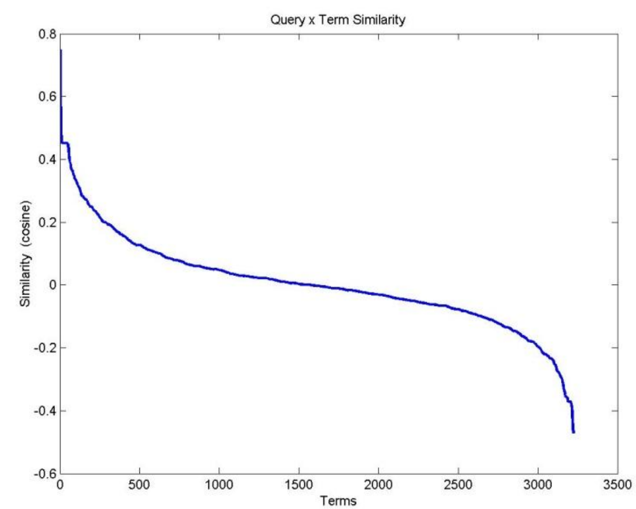
Weighted Frequency Matrix



Query Terms:
- Insulation
- Joint

```
DOCUMENTS:
   'CM031.txt'
   'CM046.txt'
   'CM001.txt'
   'CM029.txt'
   'CM040.txt'
K>> return
TERMS:
   'joint'
   'insulation'
   'roofing'
   'expansion'
   'saw'
```





Probabilistic Model

- Basic assumption: Given a user query, there is a set of documents which contains exactly the relevant documents and no other (ideal answer set)
- Querying process as a process of specifying the properties of an ideal answer set. Since these properties are not known at query time, an initial guess is made
- This initial guess allows the generation of a preliminary probabilistic description of the ideal answer set which is used to retrieve the first set of documents
- An interaction with the user is then initiated with the purpose of improving the probabilistic description of the answer set

Types of Text Data Mining

- Keyword-based association analysis
- Automatic document classification
- Similarity detection
  - Cluster documents by a common author
  - Cluster documents containing information from a common source
- Link analysis: unusual correlation between entities
- Sequence analysis: predicting a recurring event
- Anomaly detection: find information that violates usual patterns
- Hypertext analysis
  - Patterns in anchors/links
    - Anchor text correlations with linked objects

Keyword-Based Association Analysis

- Motivation
  - Collect sets of keywords or terms that occur frequently together and then find the association or correlation relationships among them
- Association Analysis Process
  - Preprocess the text data by parsing, stemming, removing stop words, etc.
  - Evoke association mining algorithms
    - Consider each document as a transaction
    - View a set of keywords in the document as a set of items in the transaction
  - Term level association mining
    - No need for human effort in tagging documents
    - The number of meaningless results and the execution time is greatly reduced

Text Classification

- 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 sets
  - Creation of the classification model using the selected classification algorithm
  - Classification model validation
  - Classification of new/unknown text documents
- Text document classification differs from the classification of relational data
  - Document databases are not structured according to attribute-value pairs

Text Classification(2)

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

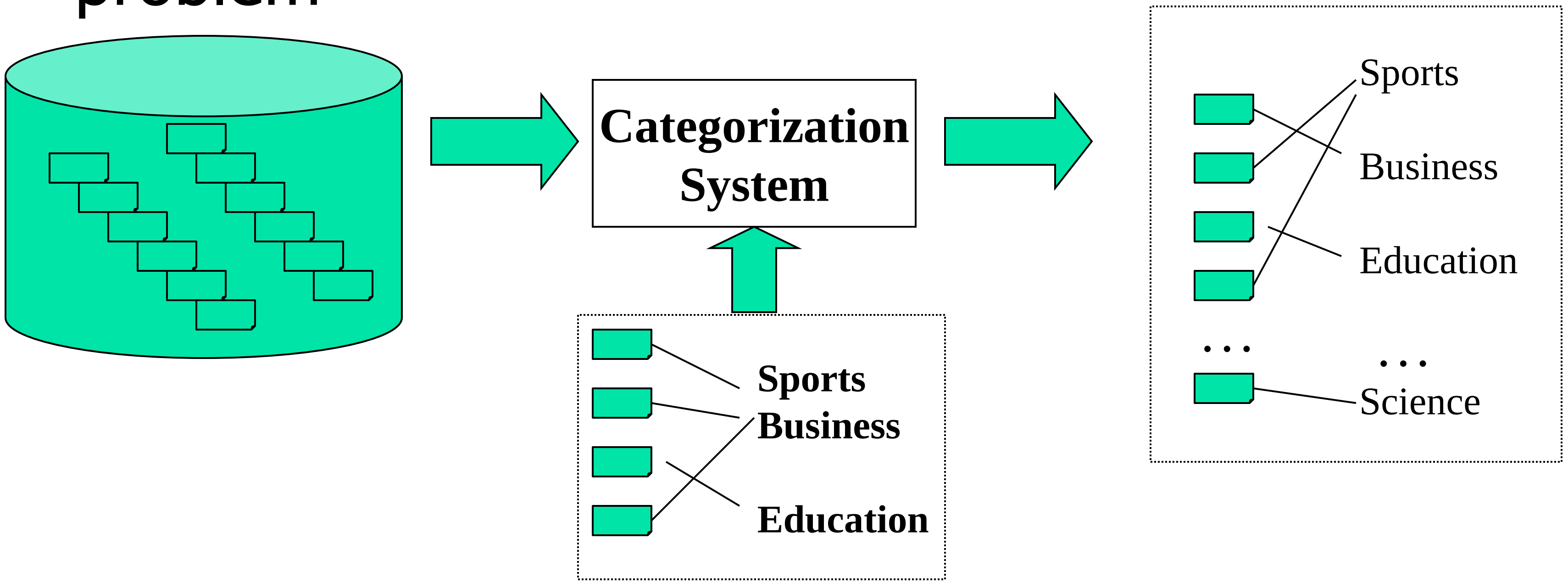
			#1	#2	#3	#4	#5
		# of documents	21,450	14,347	13,272	12,902	12,902
		# of training documents	14,704	10,667	9,610	9,603	9,603
		# of test documents	6,746	3,680	3,662	3,299	3,299
		# of categories	135	93	92	90	10
System	**Type**	**Results reported by**					
WORD	(non-learning)	[Yang 1999]	.150	.310	.290		
	probabilistic	[Dumais et al. 1998]				.752	.815
	probabilistic	[Joachims 1998]					.720
	probabilistic	[Lam et al. 1997]	.443 (MF_1)				
PROPBAYES	probabilistic	[Lewis 1992a]	.650				
BIM	probabilistic	[Li and Yamanishi 1999]				.747	
	probabilistic	[Li and Yamanishi 1999]				.773	
NB	probabilistic	[Yang and Liu 1999]				.795	
	decision trees	[Dumais et al. 1998]					.884
C4.5	decision trees	[Joachims 1998]					.794
IND	decision trees	[Lewis and Ringuette 1994]	.670				
SWAP-1	decision rules	[Apté et al. 1994]		.805			
RIPPER	decision rules	[Cohen and Singer 1999]	.683	.811		.820	
SLEEPINGEXPERTS	decision rules	[Cohen and Singer 1999]	**.753**	.759		.827	
DL-ESC	decision rules	[Li and Yamanishi 1999]				.820	
CHARADE	decision rules	[Moulinier and Ganascia 1996]		.738			
CHARADE	decision rules	[Moulinier et al. 1996]		.783 (F_1)			
LLSF	regression	[Yang 1999]		.855	.810		
LLSF	regression	[Yang and Liu 1999]				.849	
BALANCEDWINNOW	on-line linear	[Dagan et al. 1997]	.747 (M)	.833 (M)			
WIDROW-HOFF	on-line linear	[Lam and Ho 1998]				.822	
ROCCHIO	batch linear	[Cohen and Singer 1999]	.660	.748		.776	
FINDSIM	batch linear	[Dumais et al. 1998]				.617	.646
ROCCHIO	batch linear	[Joachims 1998]					.799
ROCCHIO	batch linear	[Lam and Ho 1998]				.781	
ROCCHIO	batch linear	[Li and Yamanishi 1999]				.625	
CLASSI	neural network	[Ng et al. 1997]		.802			
NNET	neural network	[Yang and Liu 1999]				.838	
	neural network	[Wiener et al. 1995]			**.820**		
GIS-W	example-based	[Lam and Ho 1998]				.860	
k-NN	example-based	[Joachims 1998]					.823
k-NN	example-based	[Lam and Ho 1998]				.820	
k-NN	example-based	[Yang 1999]	.690	.852	**.820**		
k-NN	example-based	[Yang and Liu 1999]				.856	
	SVM	[Dumais et al. 1998]				.870	**.920**
SVMLIGHT	SVM	[Joachims 1998]					.864
SVMLIGHT	SVM	[Li and Yamanishi 1999]				.841	
SVMLIGHT	SVM	[Yang and Liu 1999]				.859	
ADABOOST.MH	committee	[Schapire and Singer 2000]		**.860**			
	committee	[Weiss et al. 1999]				**.878**	
	Bayesian net	[Dumais et al. 1998]				.800	.850
	Bayesian net	[Lam et al. 1997]	.542 (MF_1)				

Document Clustering

- Motivation
  - Automatically group related documents based on their contents
  - No predetermined training sets or taxonomies
  - Generate a taxonomy at runtime
- Clustering Process
  - Data preprocessing: remove stop words, stem, feature extraction, lexical analysis, etc.
  - Hierarchical clustering: compute similarities applying clustering algorithms.
  - Model-Based clustering (Neural Network Approach): clusters are represented by “exemplars”. (e.g.: SOM)

Text Categorization

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



Applications

- News article classification
- Automatic email filtering
- Webpage classification
- Word sense disambiguation
- … …

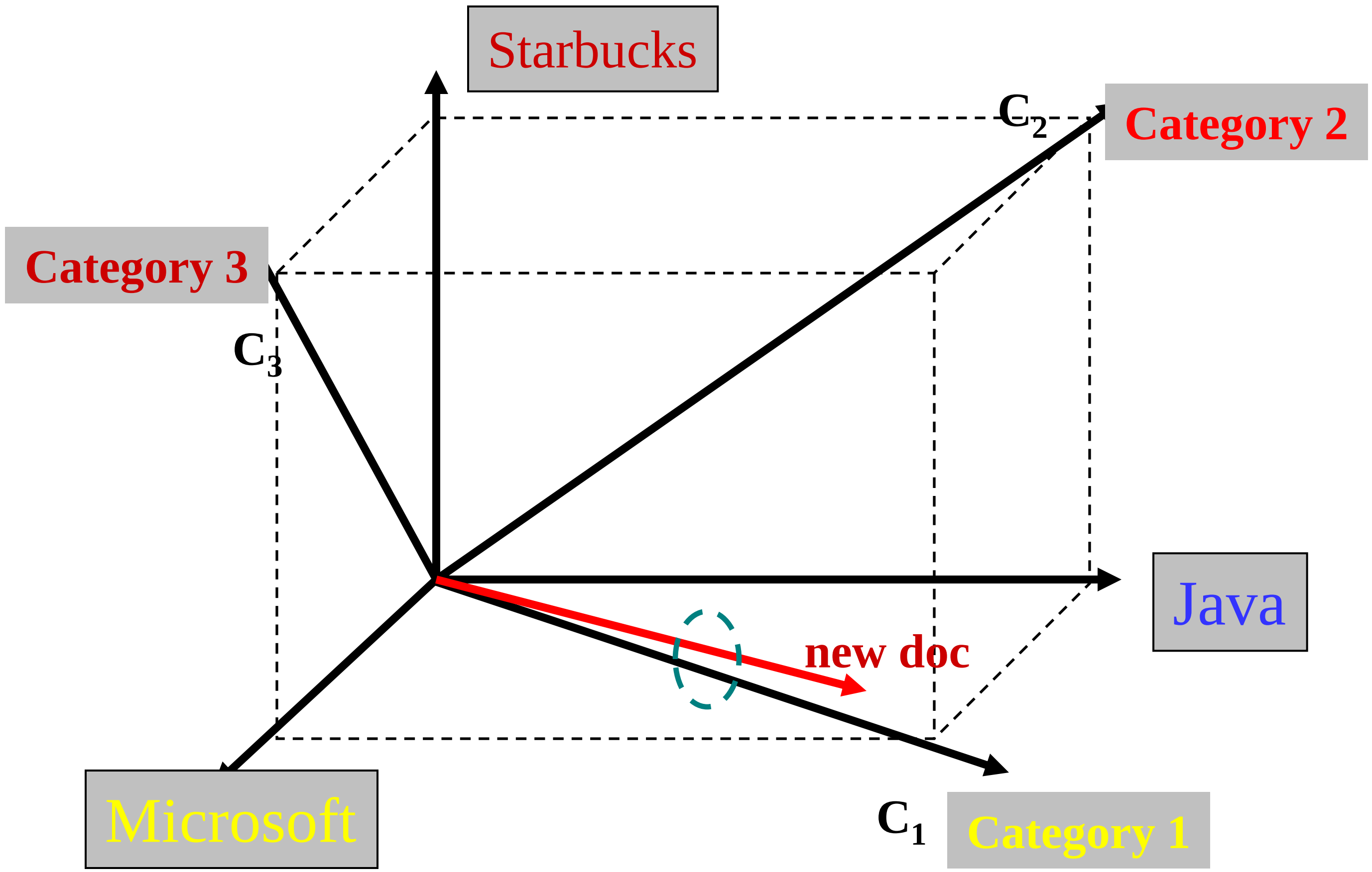
Categorization Methods

- Manual: Typically rule-based
  - Does not scale up (labor-intensive, rule inconsistency)
  - May be appropriate for special data on a particular domain
- Automatic: Typically exploiting machine learning techniques
  - Vector space model based
    - Prototype-based (Rocchio)
    - K-nearest neighbor (KNN)
    - Decision-tree (learn rules)
    - Neural Networks (learn non-linear classifier)
    - Support Vector Machines (SVM)
  - Probabilistic or generative model based
    - Naïve Bayes classifier

Vector Space Model

- Represent a doc 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., $d = (x_1,\ldots,x_N)$, x_i is "importance" of term i
- New document is assigned to the most likely category based on vector similarity.

VS Model: Illustration



What VS Model Does Not Specify

- How to select terms to capture “basic concepts”
  - Word stopping
    - e.g. “a”, “the”, “always”, “along”
  - Word stemming
    - e.g. “computer”, “computing”, “computerize” => “compute”
  - Latent semantic indexing
- 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"
  - IDF (Inverse document frequency)
    - Less frequent ***among*** documents → more discriminative
    - e.g. "algebra" vs. "science"

TF Weighting

- Weighting:
  - More frequent => more relevant to topic
    - e.g. "query" vs. "commercial"
    - Raw TF= f(*t,d*): how many times term *t* appears in doc *d*
- Normalization:
  - Document length varies => relative frequency preferred
    - e.g., Maximum frequency normalization

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

IDF Weighting

- Ideas:
  - Less frequent ***among*** documents → more discriminative
- Formula:

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

n — total number of docs

k — # docs with term t appearing

(the DF document frequency)

TF-IDF Weighting

- TF-IDF weighting : **weight(t, d) = TF(t, d) * IDF(t)**
  - Freqent within doc → high tf → high weight
  - Selective among docs → high idf → high weight
- Recall VS model
  - Each selected term represents one dimension
  - Each doc is represented by a feature vector
  - Its *t*-term coordinate of document *d* is the TF-IDF weight
  - This is more reasonable
- Just for illustration …
  - Many complex and more effective weighting variants exist in practice

How to Measure Similarity?

- Given two document

$$D_i = (w_{i1}, w_{i2}, \cdots, w_{iN}) \qquad D_j = (w_{j1}, w_{j2}, \cdots, w_{jN})$$

- Similarity definition
  - dot product

$$Sim(D_i, D_j) = \sum_{t=i}^{N} w_{it} * w_{jt}$$

  - normalized dot product (or cosine)

$$Sim(D_i, D_j) = \frac{\sum_{t=i}^{N} w_{it} * w_{jt}}{\sqrt{\sum_{t=1}^{N} (w_{it})^2 * \sum_{t=1}^{N} (w_{jt})^2}}$$

Illustrative Example

doc1

text
mining
search
engine
text

doc2

travel
text

map
travel

doc3

government
president
congress

……

Sim(newdoc,doc1)=4.8*2.4+4.5*4.5

Sim(newdoc,doc2)=2.4*2.4

Sim(newdoc,doc3)=0

To whom is newdoc more similar?

	text	mining	travel	map	search	engine	govern	president	congress
IDF(faked)	2.4	4.5	2.8	3.3	2.1	5.4	2.2	3.2	4.3
doc1	2(4.8)	1(4.5)			1(2.1)	1(5.4)			
doc2	1(2.4)		2 (5.6)	1(3.3)					
doc3							1 (2.2)	1(3.2)	1(4.3)
newdoc	1(2.4)	1(4.5)							

VS Model-Based Classifiers

- What do we have so far?
  - A feature space with similarity measure
  - This is a classic supervised learning problem
    - Search for an approximation to classification hyper plane
- VS model based classifiers
  - K-NN
  - Decision tree based
  - Neural networks
  - Support vector machine

Probabilistic Model

- Main ideas
  - Category C is modeled as a probability distribution of pre-defined random events
  - Random events model the process of generating documents
  - Therefore, how likely a document d belongs to category C is measured through the probability for category C to generate d.

Quick Revisit of Bayes' Rule

Category Hypothesis space: $H = \{C_1, \ldots, C_n\}$
One document: D

$$P(C_i \mid D) = \frac{P(D \mid C_i) P(C_i)}{P(D)}$$

As we want to pick the most likely category C*, we can drop p(D)

Posterior probability of C_i

$$C^* = \arg\max_C P(C \mid D) = \arg\max_C P(D \mid C) P(C)$$

Document model for category C

Probabilistic Model

- Multi-Bernoulli
  - Event: word presence or absence
  - $D = (x_1, \ldots, x_{|V|})$, $x_i = 1$ for presence of word w_i; $x_i = 0$ for absence

$$p(D=(x_1,\ldots,x_{|V|})\,|\,C) = \prod_{i=1}^{|V|} p(w_i = x_i\,|\,C) = \prod_{i=1, x_i=1}^{|V|} p(w_i = 1\,|\,C) \prod_{i=1, x_i=0}^{|V|} p(w_i = 0\,|\,C)$$

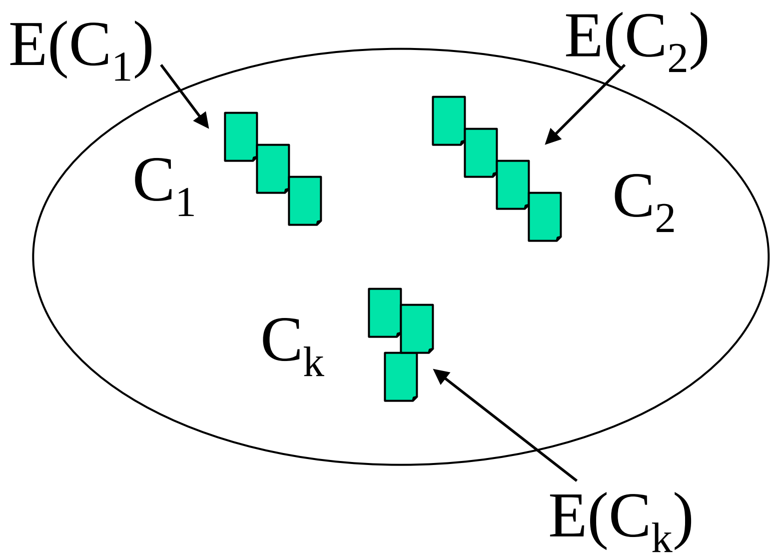
  - Parameters: $\{p(w_i=1|C), p(w_i=0|C)\}$, $p(w_i=1|C) + p(w_i=0|C) = 1$
- Multinomial (Language Model)
  - Event: word selection/sampling
  - $D = (n_1, \ldots, n_{|V|})$, n_i: frequency of word w_i $n = n_1, + \ldots + n_{|V|}$

$$p(D=(n_1,\ldots,n_{|v|})\,|\,C) = p(n\,|\,C)\binom{n}{n_1 \ \ldots \ n_{|V|}}\prod_{i=1}^{|V|} p(w_i\,|\,C)^{n_i}$$

  - Parameters: $\{p(w_i|C)\}$ $p(w_1|C) + \ldots \ p(w_{|v|}|C) = 1$

Parameter Estimation

Training examples:



Vocabulary: $V = \{w_1, \ldots, w_{|V|}\}$

- Category prior

$$p(C_i) = \frac{|E(C_i)|}{\sum_{j=1}^{k} |E(C_j)|}$$

- Multi-Bernoulli Doc model

$$p(w_j = 1 | C_i) = \frac{\sum_{d \in E(C_i)} \delta(w_j, d) + 0.5}{|E(C_i)| + 1} \qquad \delta(w_j, d) = \begin{cases} 1 & \text{if } w_j \text{ occurs in } d \\ 0 & \text{otherwise} \end{cases}$$

- Multinomial doc model

$$p(w_j | C_i) = \frac{\sum_{d \in E(C_i)} c(w_j, d) + 1}{\sum_{m=1}^{|V|} \sum_{d \in E(C_i)} c(w_m, d) + |V|} \qquad c(w_j, d) = \text{counts of } w_j \text{ in } d$$

Classification of New Document

Multi-Bernoulli

$$d = (x_1, \dots, x_{|V|}) \quad x \in \{0,1\}$$

$$C^* = \arg\max_C P(D \mid C) P(C)$$

$$= \arg\max_C \prod_{i=1}^{|V|} p(w_i = x_i \mid C) P(C)$$

$$= \arg\max_C \log p(C) + \sum_{i=1}^{|V|} \log p(w_i = x_i \mid C)$$

Multinomial

$$d = (n_1, \dots, n_{|V|}) \quad |d| = n = n_1 + \dots + n_{|V|}$$

$$C^* = \arg\max_C P(D \mid C) P(C)$$

$$= \arg\max_C p(n \mid C) \prod_{i=1}^{|V|} p(w_i \mid C)^{n_i} P(C)$$

$$= \arg\max_C \log p(n \mid C) + \log p(C) + \sum_{i=1}^{|V|} n_i \log p(w_i \mid C)$$

$$\approx \arg\max_C \log p(C) + \sum_{i=1}^{|V|} n_i \log p(w_i \mid C)$$

Categorization Methods

- Vector space model
  - K-NN
  - Decision tree
  - Neural network
  - Support vector machine
- Probabilistic model
  - Naïve Bayes classifier
- Many, many others and variants exist [F.S. 02]
  - e.g. Bim, Nb, Ind, Swap-1, LLSF, Widrow-Hoff, Rocchio, Gis-W, … …

Evaluations

- Effectiveness measure
  - Classic: Precision & Recall

Table II. The Contingency Table for Category c_i

Category c_i		Expert judgments YES	Expert judgments NO
Classifier Judgments	**YES**	TP_i	FP_i
	NO	FN_i	TN_i

- Precision $$\hat{\pi}_i = \frac{TP_i}{TP_i + FP_i}$$
- Recall $$\hat{\rho}_i = \frac{TP_i}{TP_i + FN_i}.$$

Evaluation (con't)

- Benchmarks
  - Classic: Reuters collection
    - A set of newswire stories classified under categories related to economics.
- Effectiveness
  - Difficulties of strict comparison
    - different parameter setting
    - different "split" (or selection) between training and testing
    - various optimizations … …
  - However widely recognizable
    - Best: Boosting-based committee classifier & SVM
    - Worst: Naïve Bayes classifier
  - Need to consider other factors, especially efficiency

Summary: Text Categorization

- Wide application domain
- Comparable effectiveness to professionals
  - Manual TC is not 100% and unlikely to improve substantially.
  - A.T.C. is growing at a steady pace
- Prospects and extensions
  - Very noisy text, such as text from O.C.R.
  - Speech transcripts

Research Problems in Text Mining

- Google: what is the next step?
- How to find the pages that match approximately the sohpisticated documents, with incorporation of user-profiles or preferences?
- Look back of Google: inverted indicies
- Construction of indicies for the sohpisticated documents, with incorporation of user-profiles or preferences
- Similarity search of such pages using such indicies

References

- Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization", ACM Computing Surveys, Vol. 34, No.1, March 2002
- Soumen Chakrabarti, "Data mining for hypertext: A tutorial survey", ACM SIGKDD Explorations, 2000.
- Cleverdon, "Optimizing convenient online accesss to bibliographic databases", Information Survey, Use4, 1, 37-47, 1984
- Yiming Yang, "An evaluation of statistical approaches to text categorization", Journal of Information Retrieval, 1:67-88, 1999.
- Yiming Yang and Xin Liu "A re-examination of text categorization methods". Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99, pp 42--49), 1999.

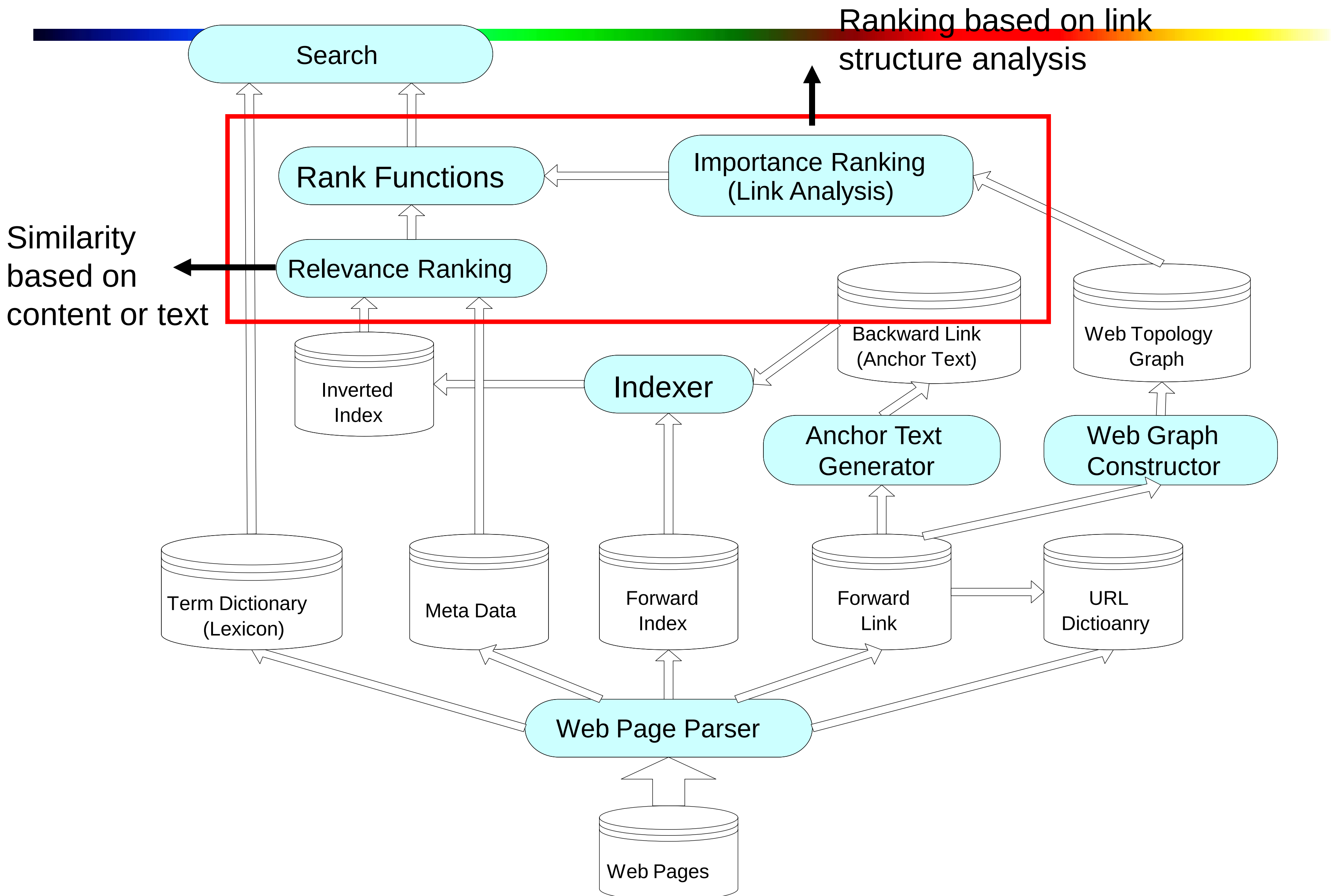
Mining Text and Web Data

- Text mining, natural language processing and information extraction: An Introduction
- Text categorization methods
- Mining Web linkage structures
  - Based on the slides by Deng Cai
- Summary

Outline

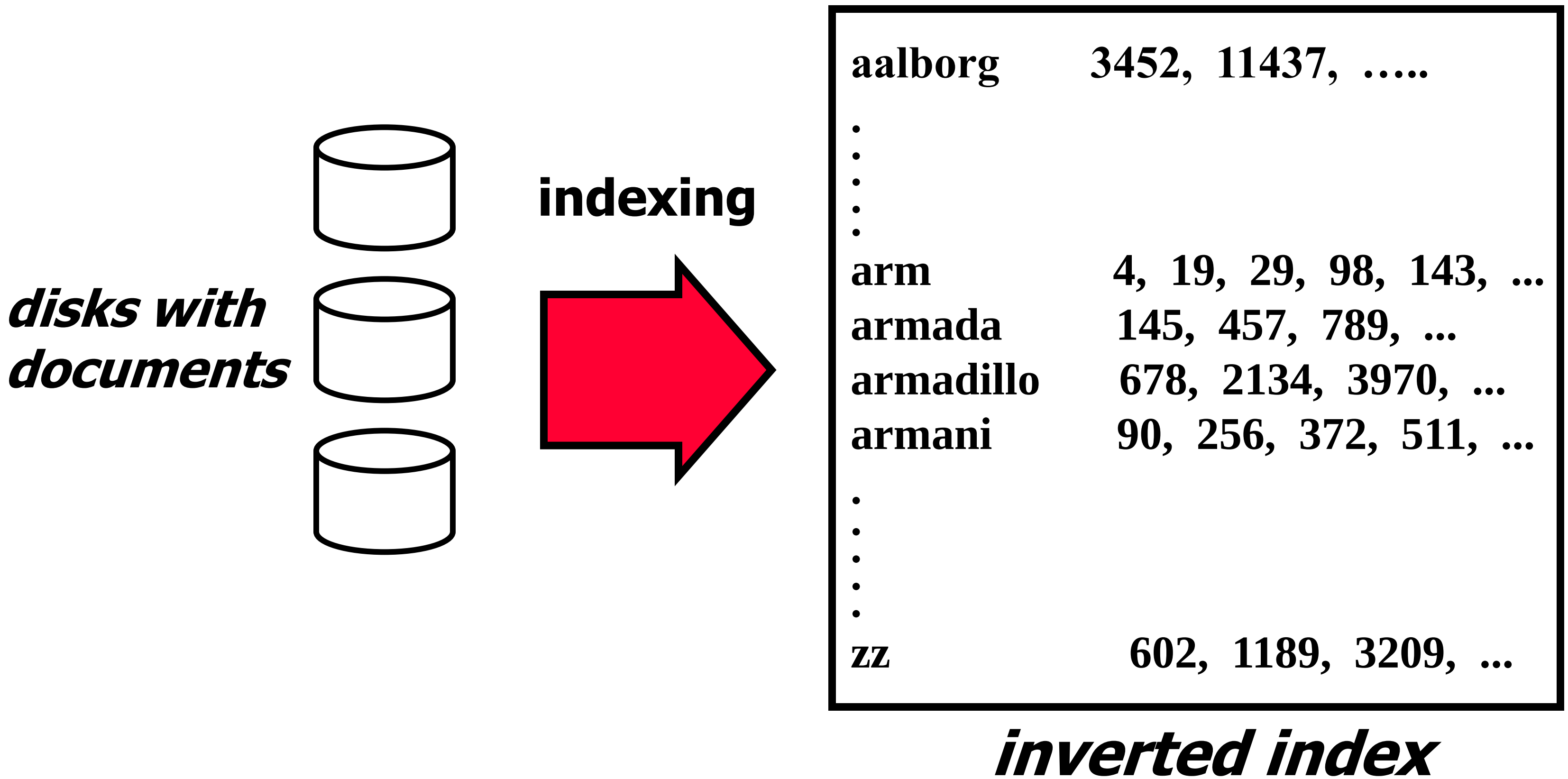
- Background on Web Search
- VIPS (VIsion-based Page Segmentation)
- Block-based Web Search
- Block-based Link Analysis
- Web Image Search & Clustering

Search Engine – Two Rank Functions

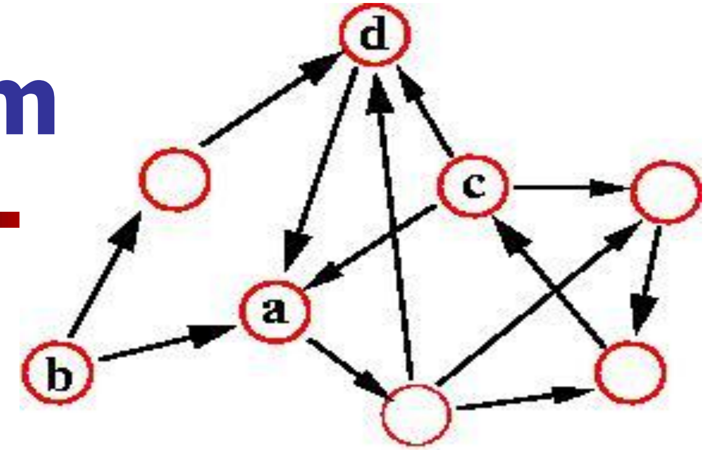


Relevance Ranking

- **Inverted index**
  - A data structure for supporting text queries
  - like index in a book



The PageRank Algorithm



- Basic idea
  - ***significance of a page is determined by the significance of the pages linking to it***
- More precisely:

$$A_{ij} = \begin{cases} 1 & if \;\; page\; i \; links\; to\; page\;\; j \\ 0 & otherwise \end{cases}$$

  - Link graph: adjacency matrix *A*,
  - Constructs a probability transition matrix *M* by renormalizing each row of *A* to sum to 1 $\varepsilon U + (1-\varepsilon)M \qquad U_{ij} = 1/n \;\; for\; all\; i, j$
  - Treat the web graph as a markov chain (random surfer)
  - The vector of PageRank scores *p* is then defined to be the stationary distribution of this Markov chain. Equivalently, p is the principal right eigenvector of the transition matrix $(\varepsilon U + (1-\varepsilon)M)^T$

$$(\varepsilon U + (1-\varepsilon)M)^T p = p$$

Layout Structure

- Compared to plain text, a web page is a 2D presentation
  - Rich visual effects created by different term types, formats, separators, blank areas, colors, pictures, etc
  - Different parts of a page are not equally important

Title: CNN.com International

H1: IAEA: Iran had secret nuke agenda

H3: EXPLOSIONS ROCK BAGHDAD

…

TEXT BODY (with position and font type): The International Atomic Energy Agency has concluded that Iran has secretly produced small amounts of nuclear materials including low enriched uranium and plutonium that could be used to develop nuclear weapons according to a confidential report obtained by CNN…

Hyperlink:

- URL: http://www.cnn.com/…
- Anchor Text: Al oaeda…

Image:

- URL: http://www.cnn.com/image/…
- Alt & Caption: Iran nuclear …

Anchor Text: CNN Homepage News …

Web Page Block—Better Information Unit

Web Page Blocks

Importance = Low

Importance = Med

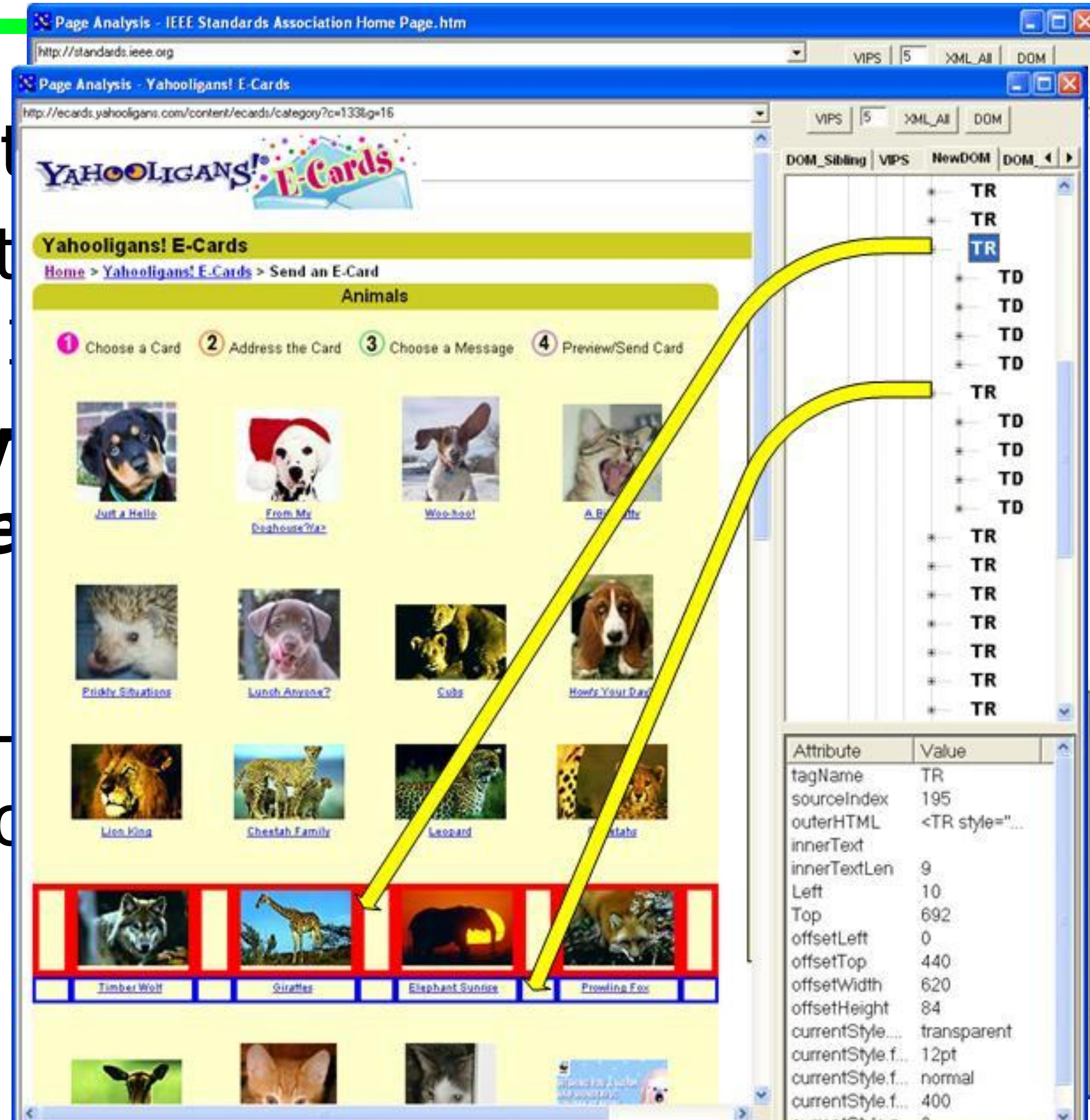
Importance = High

Motivation for VIPS (VIsion-based Page Segmentation)

- Problems of treating a web page as an atomic unit
  - Web page usually contains not only pure content
    - Noise: navigation, decoration, interaction, …
  - Multiple topics
  - Different parts of a page are not equally important
- Web page has internal structure
  - Two-dimension logical structure & Visual layout presentation
  - > Free text document
  - < Structured document
- Layout – the 3rd dimension of Web page
  - 1st dimension: content
  - 2nd dimension: hyperlink

Is DOM a Good Representation of Page Structure?

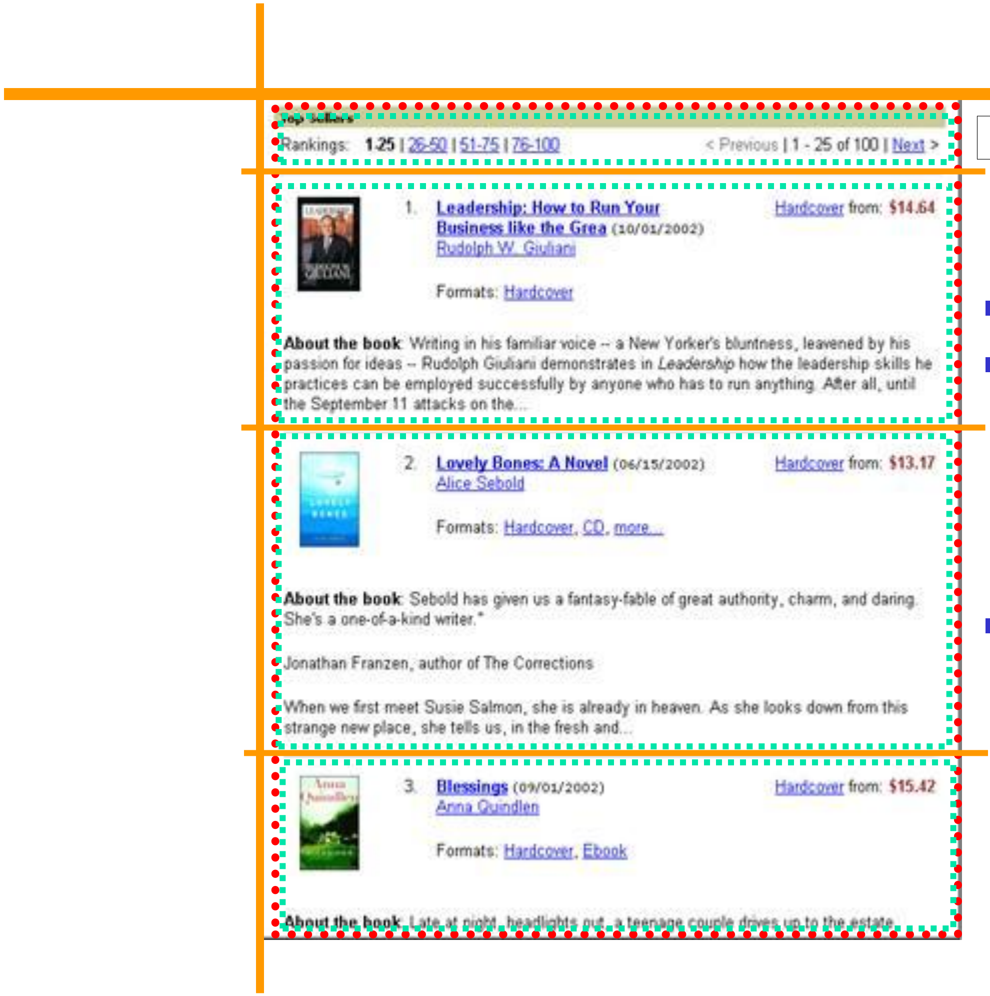
- Page segmentat
  - Extract struct UL, TITLE, H
  - ***DOM is mor does not ne structure***
- How about XML
  - A long way to

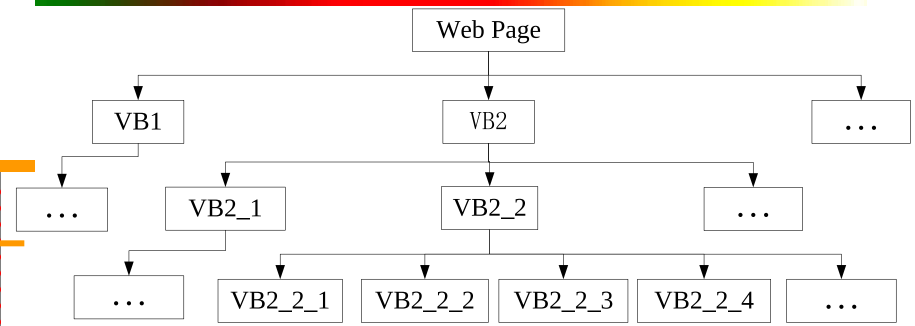


VIPS Algorithm

- Motivation:
  - In many cases, topics can be distinguished with visual clues. Such as position, distance, font, color, etc.
- Goal:
  - Extract the semantic structure of a web page based on its visual presentation.
- Procedure:
  - Top-down partition the web page based on the separators
- Result
  - A tree structure, each node in the tree corresponds to a block in the page.
  - Each node will be assigned a value (Degree of Coherence) to indicate how coherent of the content in the block based on visual perception.
  - Each block will be assigned an importance value
  - Hierarchy or flat

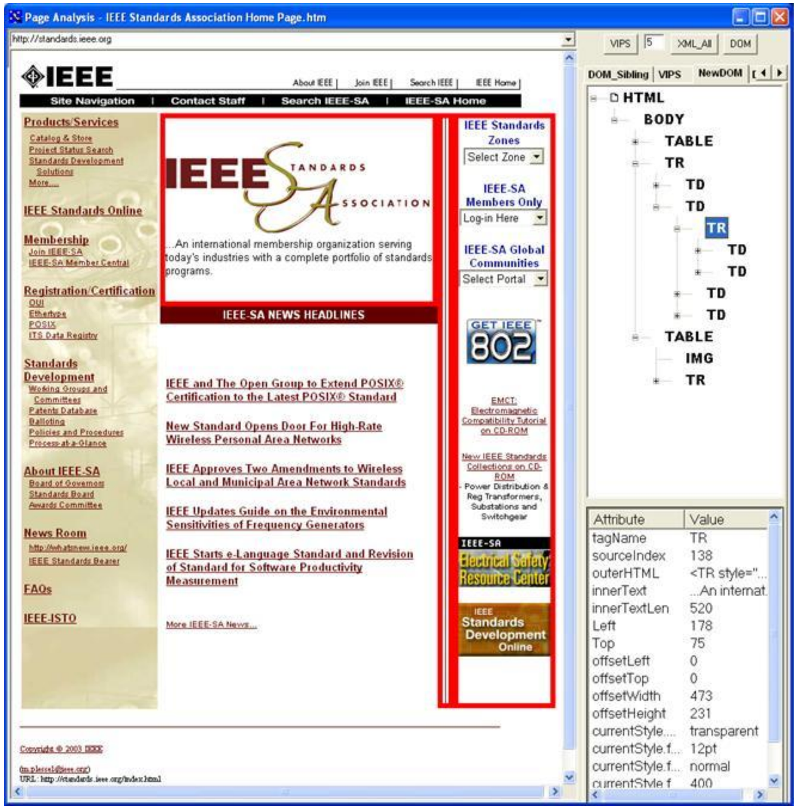
VIPS: An Example



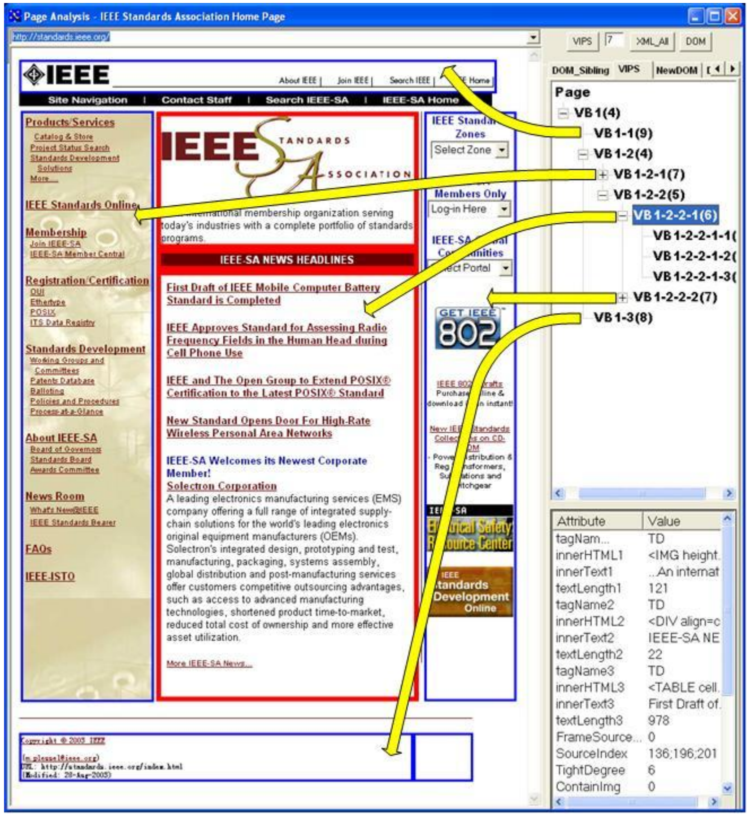


- A hierarchical structure of layout block
- A *Degree of Coherence (DOC)* is defined for each block
  - Show the intra coherence of the block
  - *DoC* of child block must be no less than its parent's
- The *Permitted Degree of Coherence (PDOC)* can be pre-defined to achieve different granularities for the content structure
  - The segmentation will stop only when all the blocks' *DoC* is no less than *PDoC*
  - The smaller the *PDoC*, the coarser the content structure would be

Example of Web Page Segmentation (1)

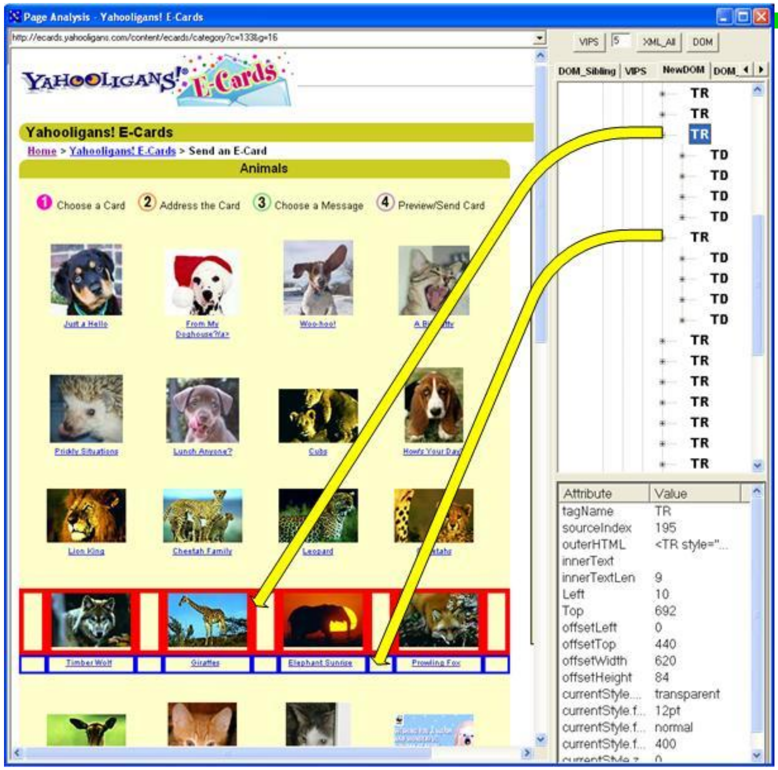


(DOM Structure)

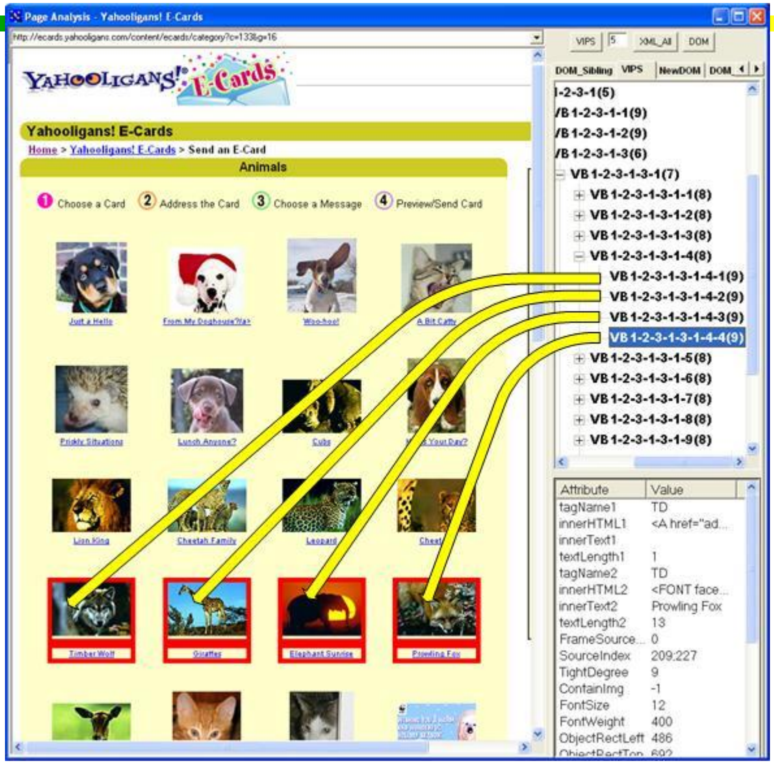


(VIPS Structure)

Example of Web Page Segmentation (2)



(DOM Structure)



(VIPS Structure)

- Can be applied on web image retrieval
  - Surrounding text extraction

Web Page Block—Better Information Unit

Page Segmentation
- **Vision based approach**

→

Block Importance Modeling
- **Statistical learning**

Web Page Blocks

Importance = Low

Importance = Med

Importance = High

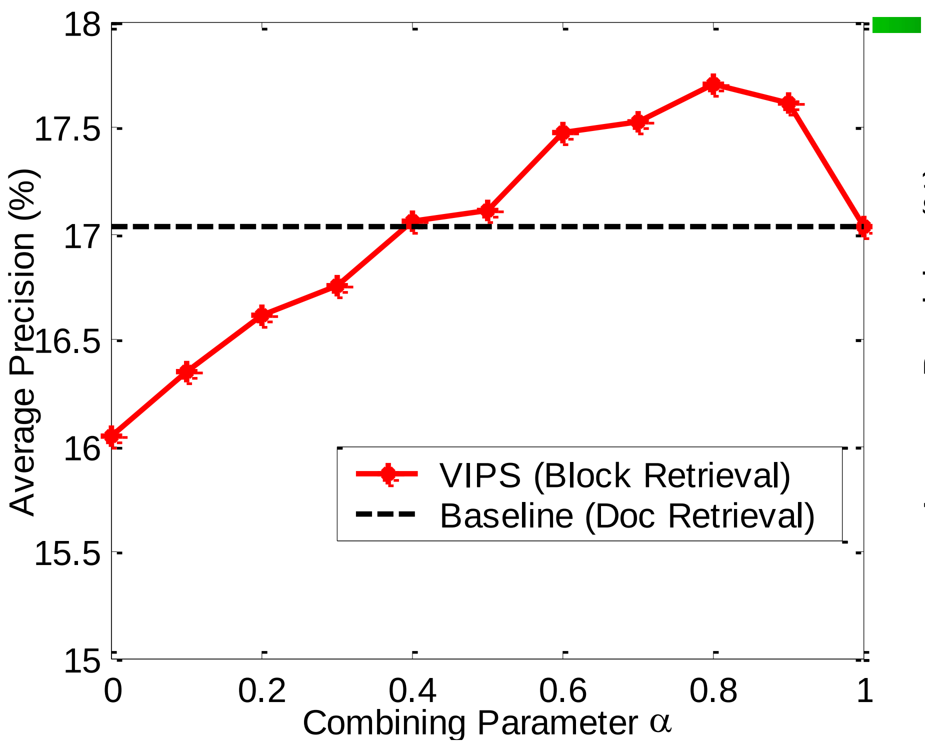
Block-based Web Search

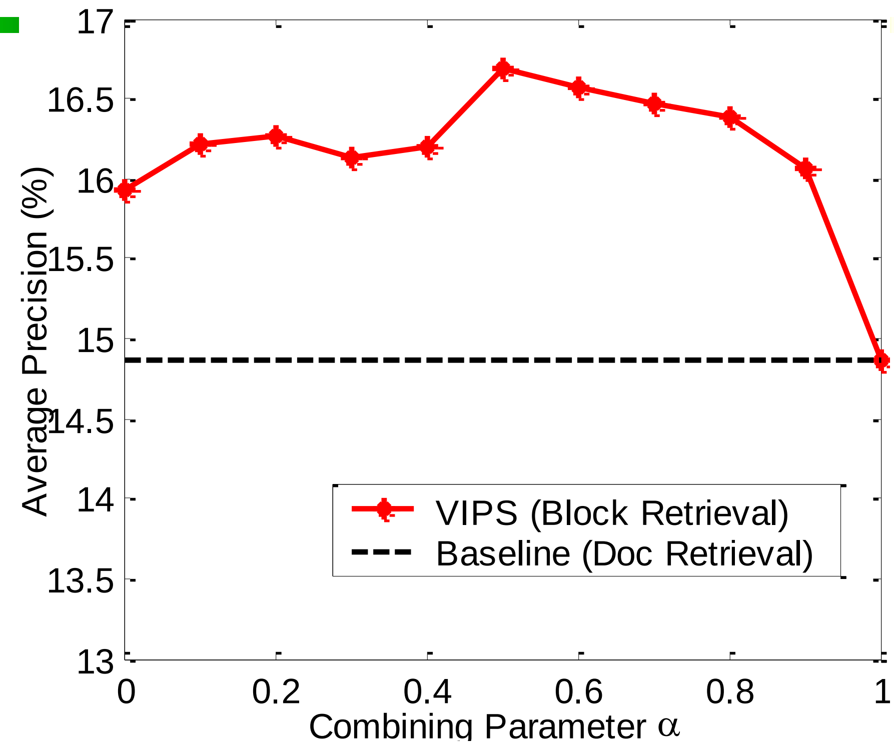
- Index block instead of whole page
- Block retrieval
  - Combing DocRank and BlockRank
- Block query expansion
  - Select expansion term from relevant blocks

Experiments

- Dataset
  - TREC 2001 Web Track
    - WT10g corpus (1.69 million pages), crawled at 1997.
    - 50 queries (topics 501-550)
  - TREC 2002 Web Track
    - .GOV corpus (1.25 million pages), crawled at 2002.
    - 49 queries (topics 551-560)
- Retrieval System
  - Okapi, with weighting function *BM2500*
- Preprocessing
  - Stop-word list (about 220)
  - Do not use stemming
  - Do not consider phrase information
- Tune the b, k_1 and k_3 to achieve the best baseline

Block Retrieval on TREC 2001 and TREC 2002

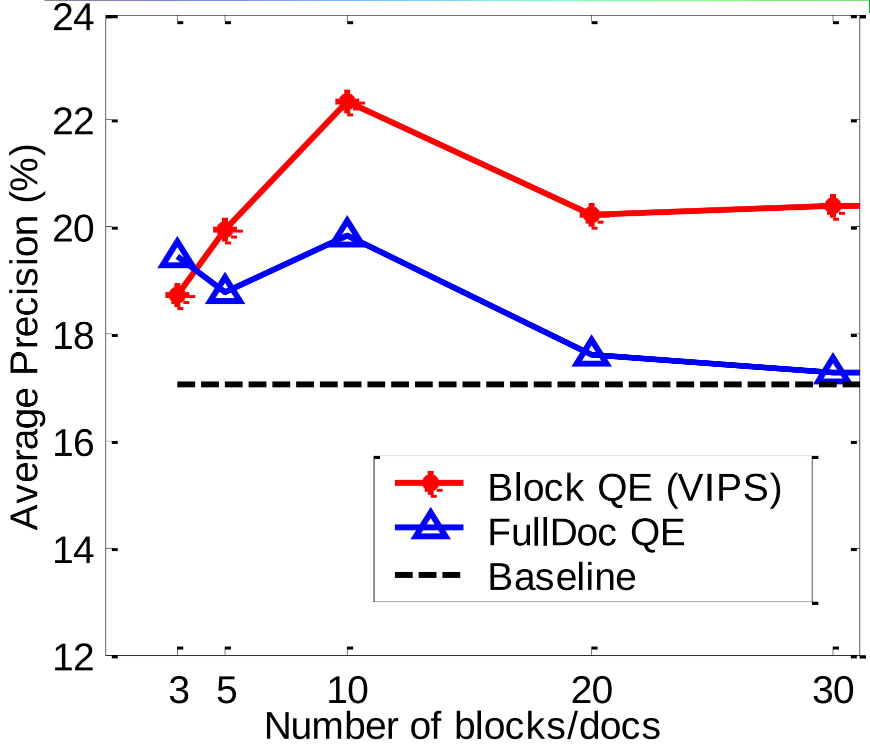




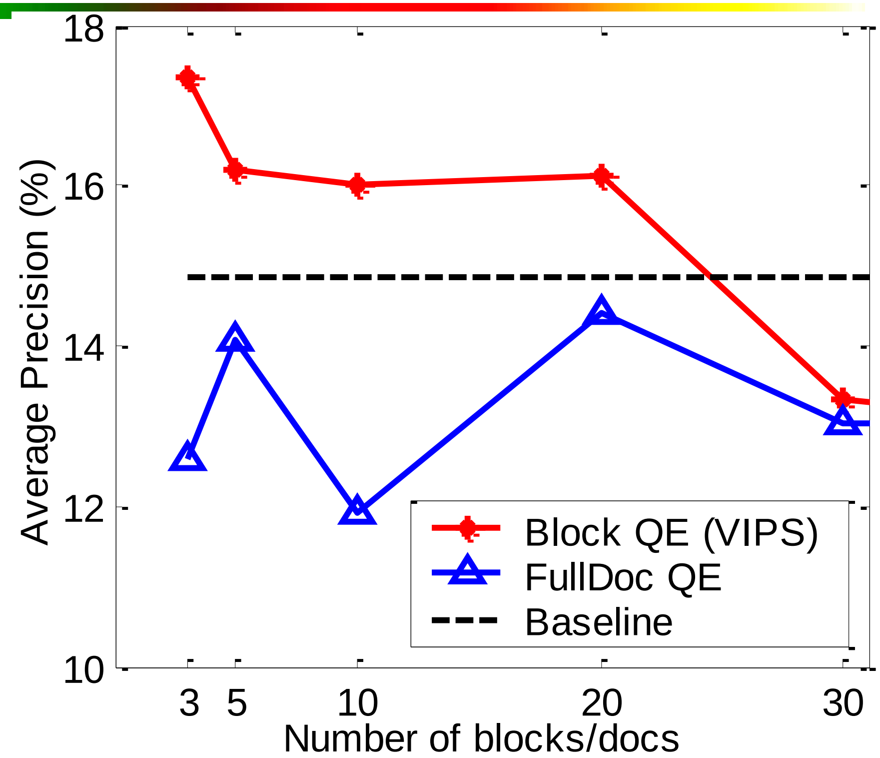
TREC 2001 Result

TREC 2002 Result

Query Expansion on TREC 2001 and TREC 2002

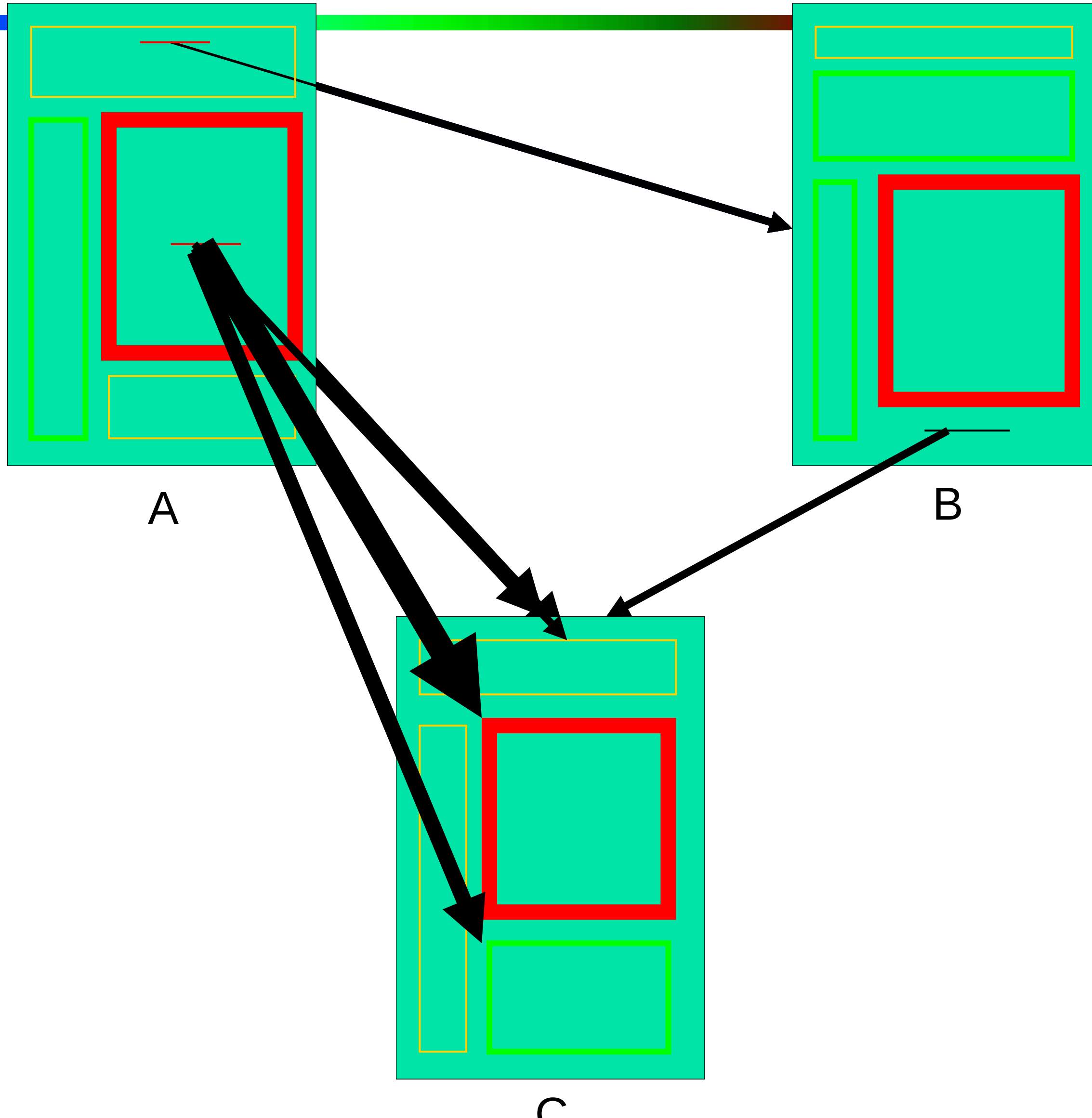


TREC 2001 Result

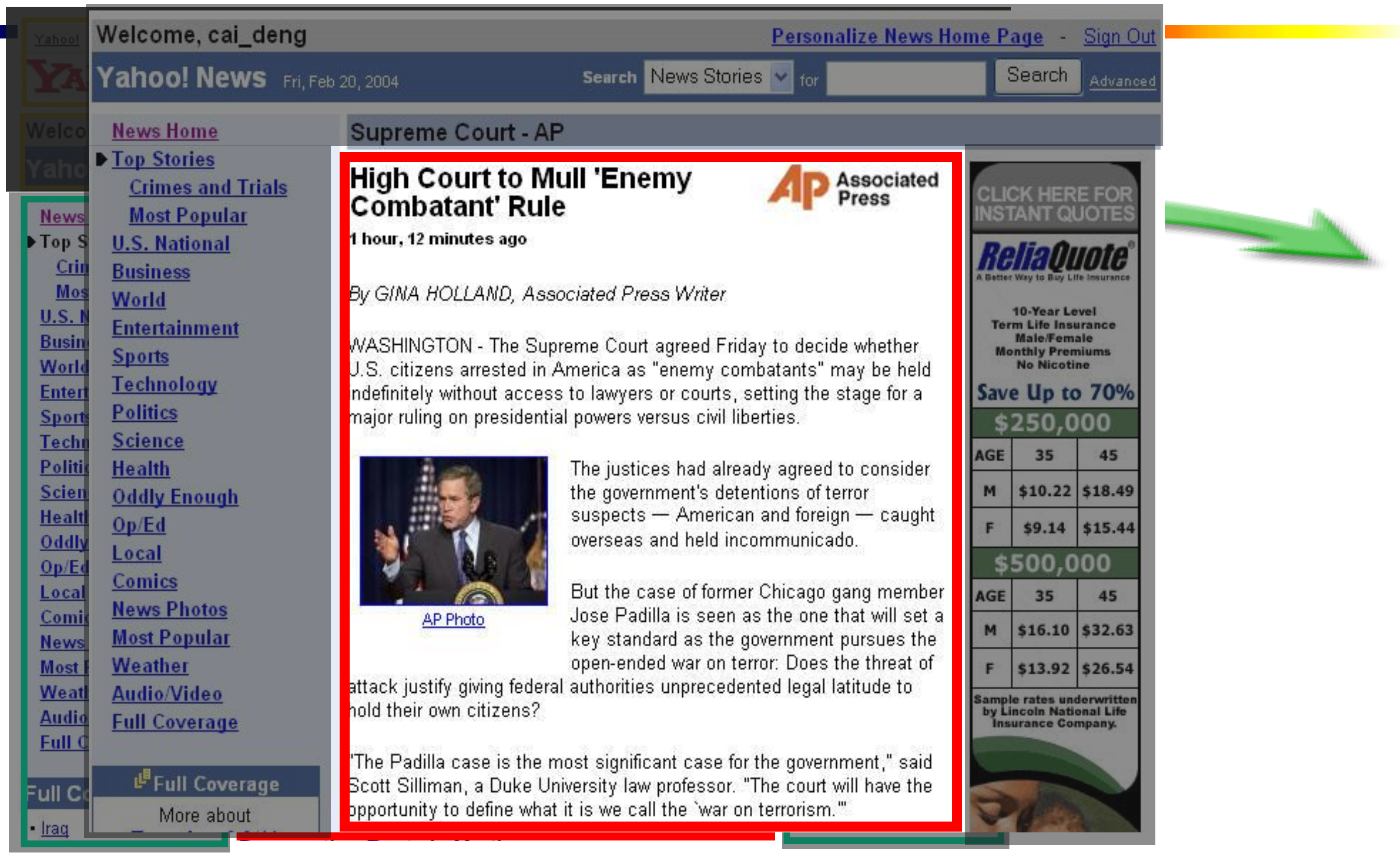


TREC 2002 Result

Block-level Link Analysis



A Sample of User Browsing Behavior



Improving PageRank using Layout Structure

- ***Z***: **block-to-page matrix (link structure)**

$$Z_{bp} = \begin{cases} 1/s_b & \text{if there is a link from the } b^{th} \text{ block to the } p^{th} \text{ page} \\ 0 & \text{otherwise} \end{cases}$$

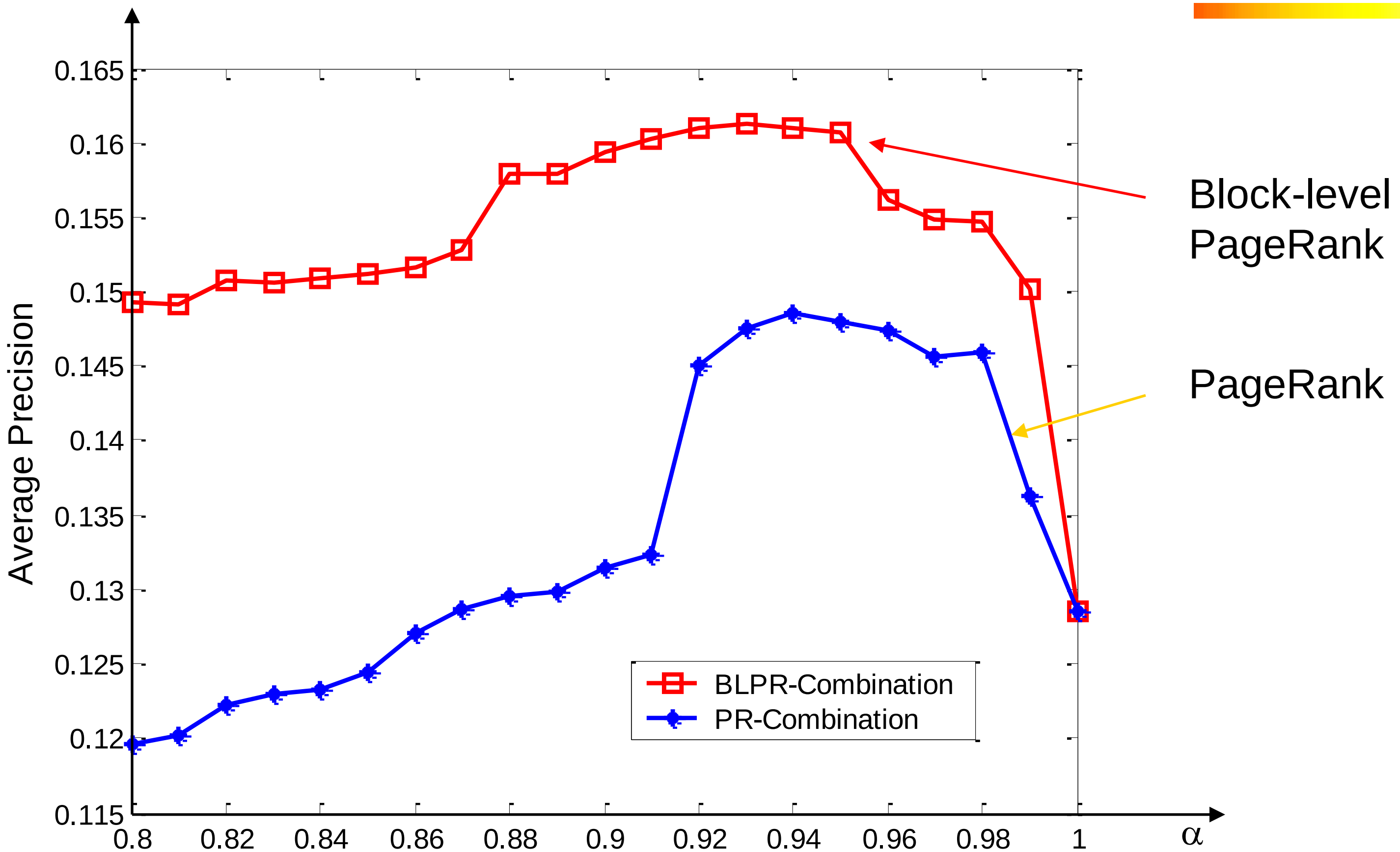
- ***X***: **page-to-block matrix (layout structure)**

$$X_{pb} = \begin{cases} f_p(b) & \text{if the } b^{th} \text{ block is in the } p^{th} \text{ page} \\ 0 & \text{otherwise} \end{cases}$$

f is the block importance function

- **Block-level PageRank**: $W_P = XZ$
  - Compute PageRank on the page-to-page graph
- **BlockRank**: $W_B = ZX$
  - Compute PageRank on the block-to-block graph

Using Block-level PageRank to Improve Search



Search = α ∗ IR_Score + (1- α) ∗ PageRank

Block-level PageRank achieves 15-25% improvement over PageRank (SIGIR'04)

Mining Web Images Using Layout & Link Structure (ACMMM'04)

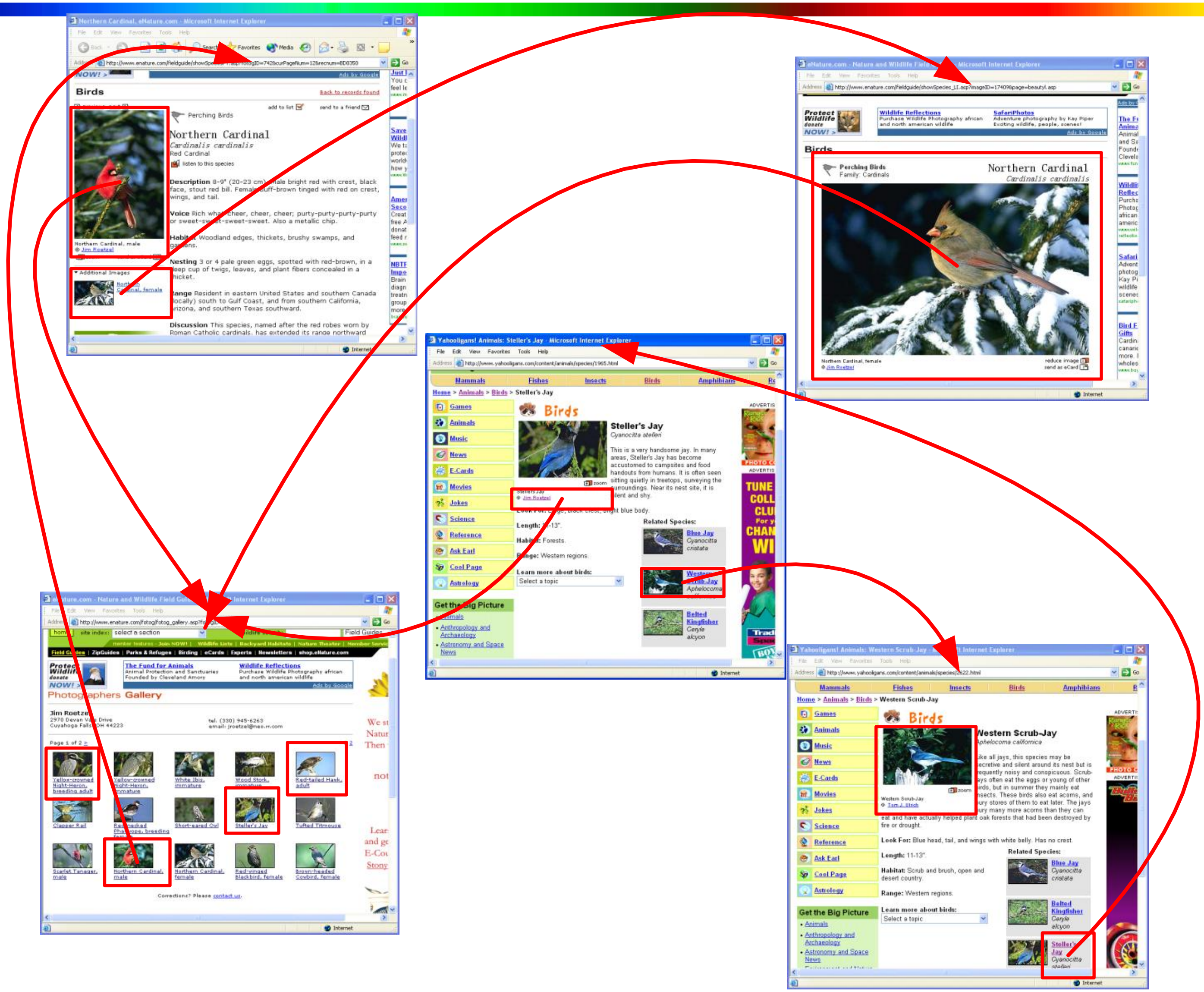


Image Graph Model & Spectral Analysis

- **Block-to-block graph**: $W_B = ZX$
- **Block-to-image matrix (container relation):** Y

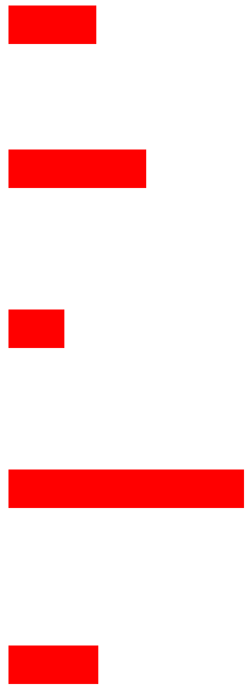
$$Y_{ij} = \begin{cases} 1/s_i & if\ I_j \in b_i \\ 0 & otherwise \end{cases}$$

- **Image-to-image graph:** $W_I = Y^T W_B Y$
- **ImageRank**
  - Compute PageRank on the image graph
- **Image clustering**
  - Graphical partitioning on the image graph

ImageRank

- Relevance Ranking

- Importance Ranking



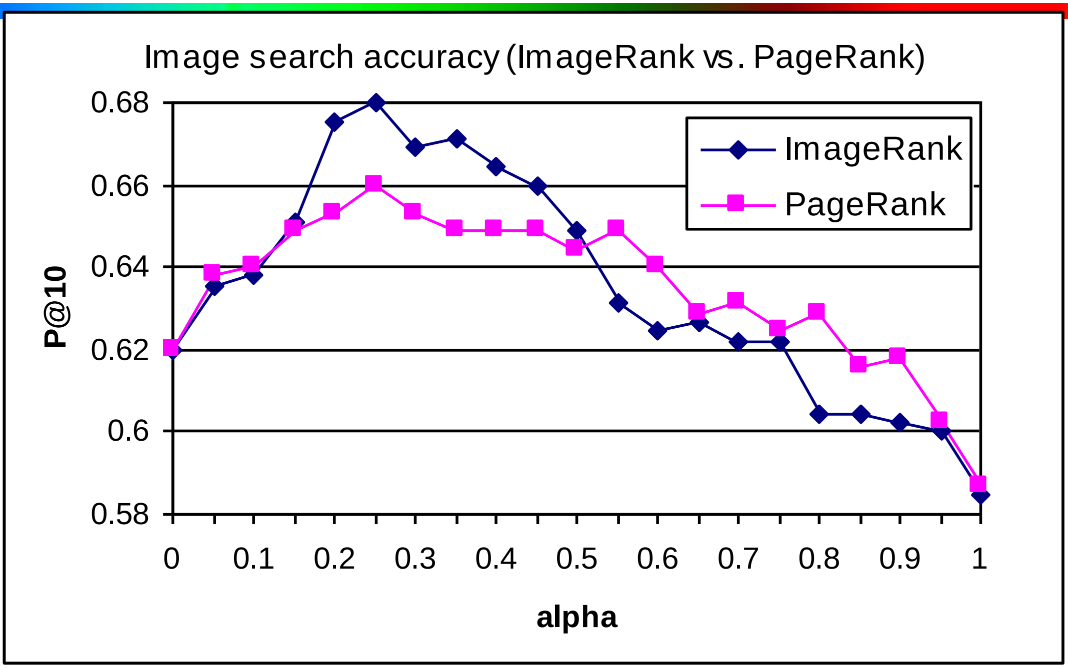
- Combined Ranking

ImageRank vs. PageRank

- Dataset
  - 26.5 millions web pages
  - 11.6 millions images
- Query set
  - 45 hot queries in Google image search statistics
- Ground truth
  - Five volunteers were chosen to evaluate the top 100 results re-turned by the system (iFind)
- Ranking method

$$s(\mathbf{x}) = \alpha \cdot rank_{importance}(\mathbf{x}) + (1-\alpha) \cdot rank_{relevance}(\mathbf{x})$$

ImageRank vs PageRank



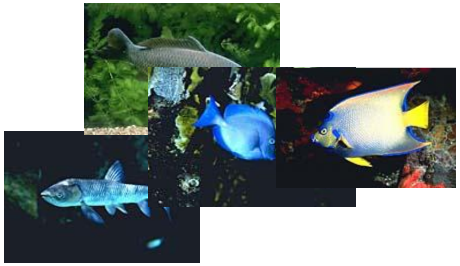
- **Image search accuracy using ImageRank and PageRank. Both of them achieved their best results at α=0.25.**

Example on Image Clustering & Embedding

1710 JPG images in 1287 pages are crawled within the website
http://www.yahooligans.com/content/animals/

Six Categories

Mammal



Fish

Reptile

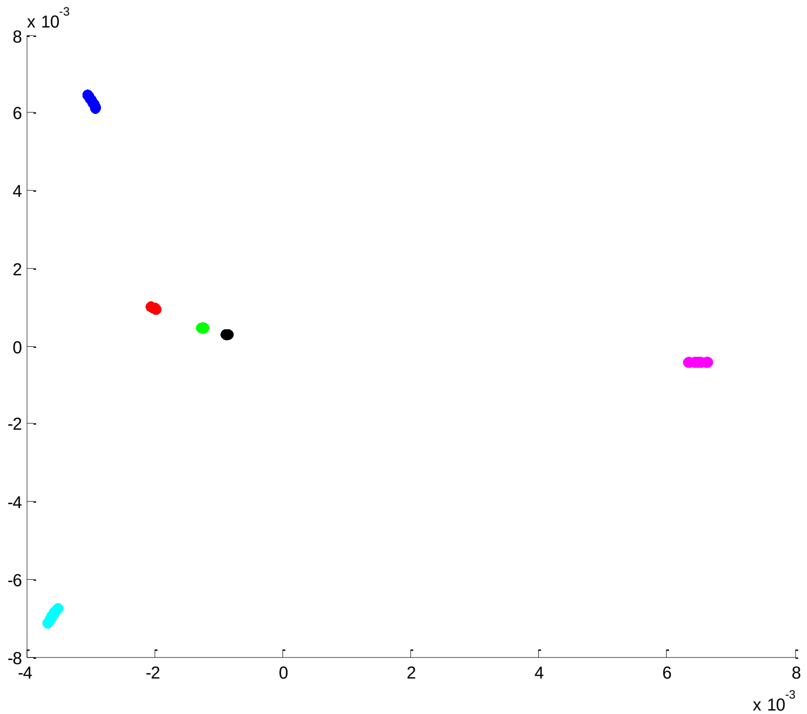
Bird

Amphibian

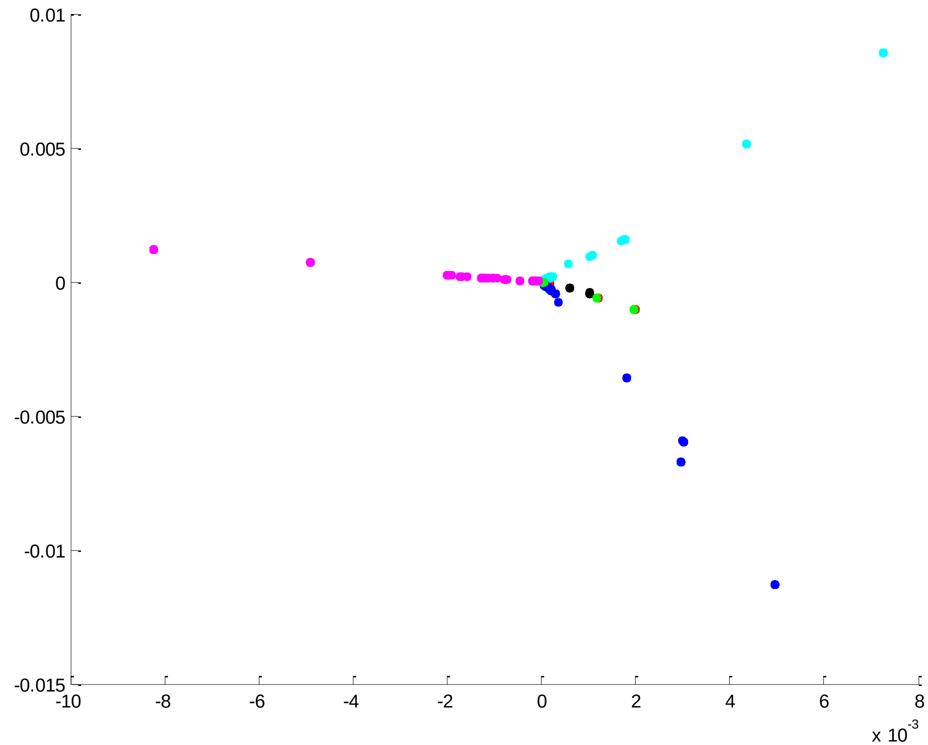
Insect


Fishes
Birds
Mammals
Yahooligans! - Help
Yahooligans! Animals
Search for Animals:
Fishes
Insects
Birds
Arctic Fox
Mammals
Animals
Music
News
E-Cards
Movies
Jokes
Science
Reference
Ask Earl
Cool Page
Astrology
Arctic Fox
Alopex lagopus
The Arctic Fox is well suited to its subzero habitat: it has a compact body with short legs and ears (body heat is lost through long ears and legs), dense fur, and thick hair on the footpads, which insulates against the cold and provides traction on ice. Winter fur develops in October: The coat thickens, and the new hairs are much lighter, providing camouflage against snow and ice. Sadly, this fox has been heavily hunted for its beautiful fur coat.
zoom
Arctic Fox
© Rita Summers
Look For: A fox of the extreme north, pure white in winter, brownish-gray in summer.
Length: Body 19-22" long.
Habitat: Tundra and sea ice.
Range: Alaska and northern Canada.
Learn more about mammals:
Select a topic
info
Related Speci
Red Fox
Vulpes vulpes
Kit Fox
Vulpes macrotis
Get the Big Picture
Animals
Environment and Nature News
All Stories
PHOTO CONTEST
ADVERTISEMENT
IMPRESS YOUR PARENTS

2-D embedding of WWW images

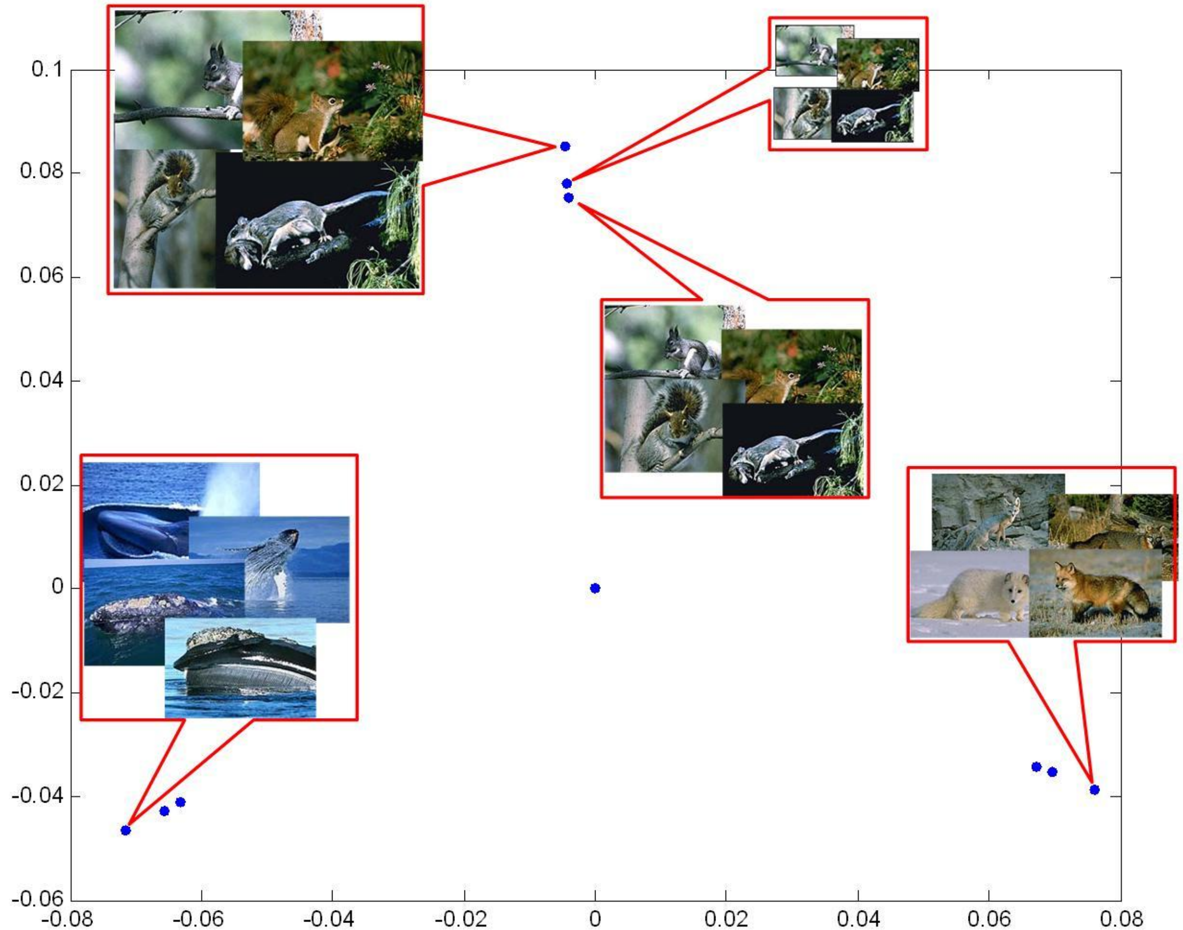


The image graph was constructed from block level link analysis



The image graph was constructed from traditional page level link analysis

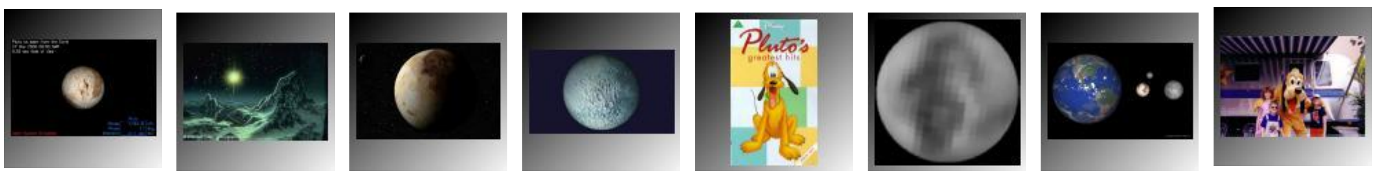
2-D Embedding of Web Images



- 2-D visualization of the mammal category using the second and third eigenvectors.

Web Image Search Result Presentation

(a)



(b)

Figure 1. Top 8 returns of query "pluto" in Google's image search engine (a) and AltaVista's image search engine (b)

- Two different topics in the search result
- A possible solution:
  - Cluster search results into different semantic groups

Three kinds of WWW image representation

- Visual Feature Based Representation
  - Traditional CBIR
- Textual Feature Based Representation
  - Surrounding text in image block
- Link Graph Based Representation
  - Image graph embedding

Hierarchical Clustering

- Clustering based on three representations
  - Visual feature
    - Hard to reflect the semantic meaning
  - Textual feature
    - Semantic
    - Sometimes the surrounding text is too little
  - Link graph:
    - Semantic
    - Many disconnected sub-graph (too many clusters)
- Two Steps:
  - Using texts and link information to get semantic clusters
  - For each cluster, using visual feature to re-organize the images to facilitate user's browsing

Our System

- Dataset
  - 26.5 millions web pages

    http://dir.yahoo.com/Arts/Visual_Arts/Photography/Museums_and_Galleries/
  - 11.6 millions images
    - Filter images whose ratio between width and height are greater than 5 or smaller than 1/5
    - Removed images whose width and height are both smaller than 60 pixels
- Analyze pages and index images
  - VIPS: Pages → Blocks
  - Surrounding texts used to index images
- An illustrative example
  - Query "Pluto"
  - Top 500 results

Clustering Using Visual Feature

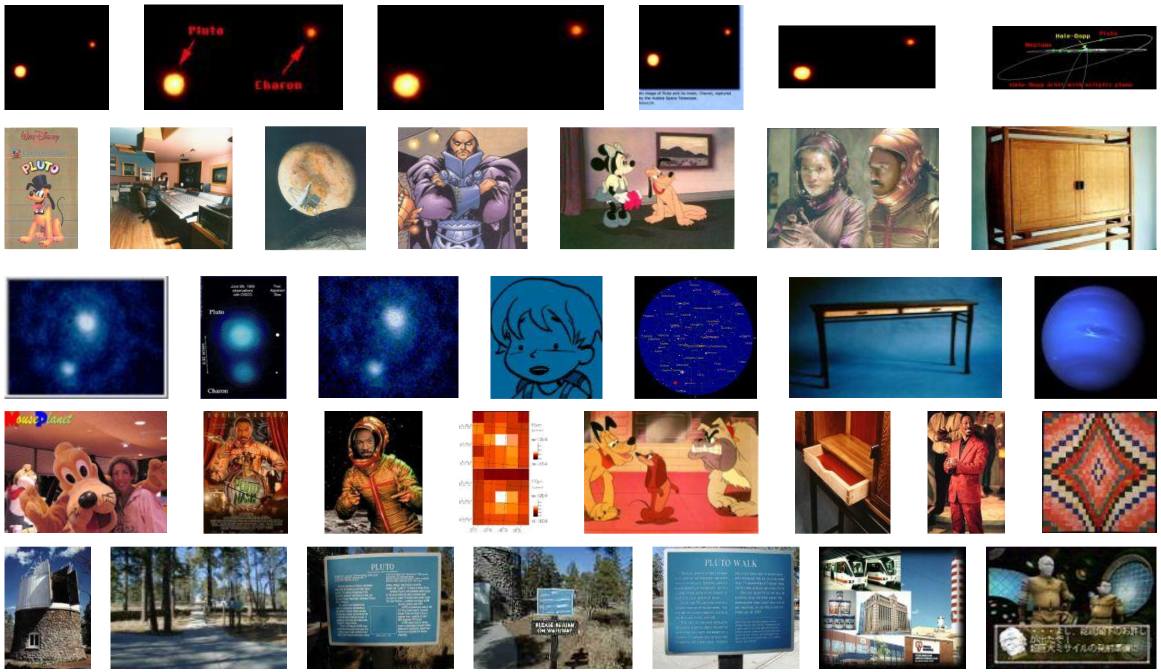


Figure 5. Five clusters of search results of query "pluto" using low level visual feature. Each row is a cluster.

- From the perspectives of color and texture, the clustering results are quite good. Different clusters have different colors and textures. However, from semantic perspective, these clusters make little sense.

Clustering Using Textual Feature

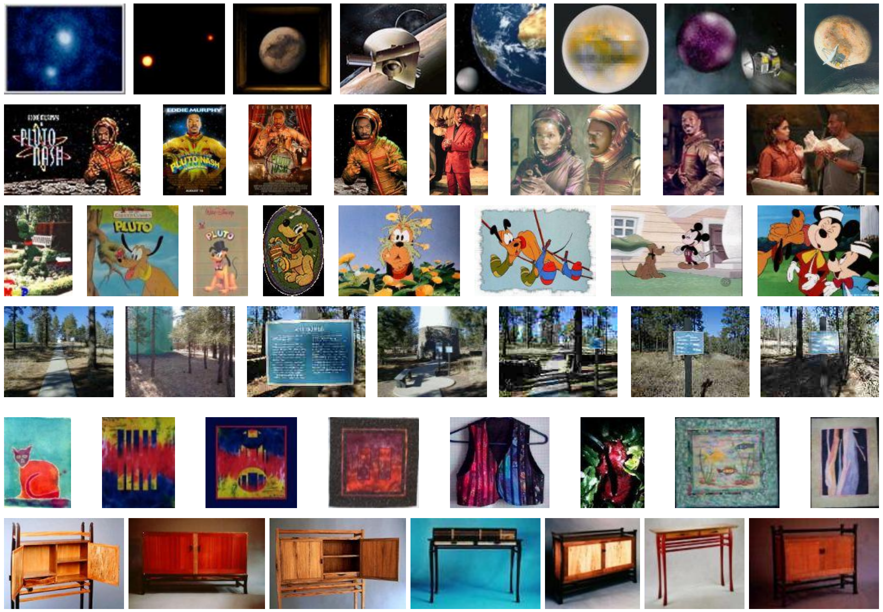


Figure 7. Six clusters of search results of query "pluto" using textual feature. Each row is a cluster

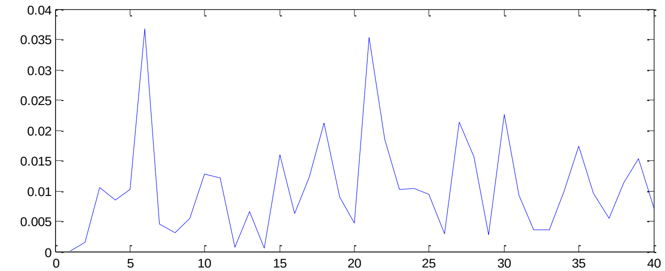


Figure 6. The Eigengap curve with *k* for the "pluto" case using textual representation

- Six semantic categories are correctly identified if we choose $k = 6$.

Clustering Using Graph Based Representation

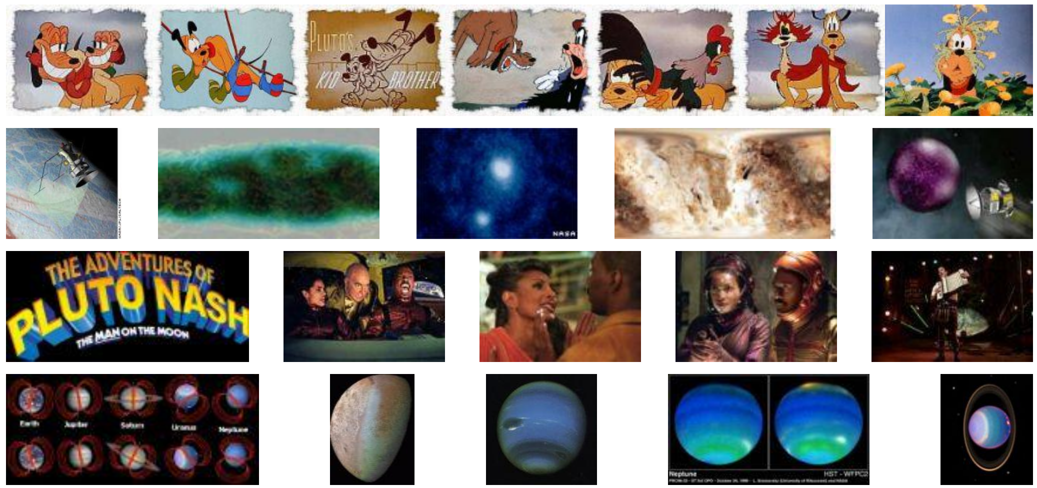


Figure 8. Five clusters of search results of query "pluto" using image link graph. Each row is a cluster

- Each cluster is semantically aggregated.
- Too many clusters.
- In "pluto" case, the top 500 results are clustered into 167 clusters. The max cluster number is 87, and there are 112 clusters with only one image.

Combining Textual Feature and Link Graph

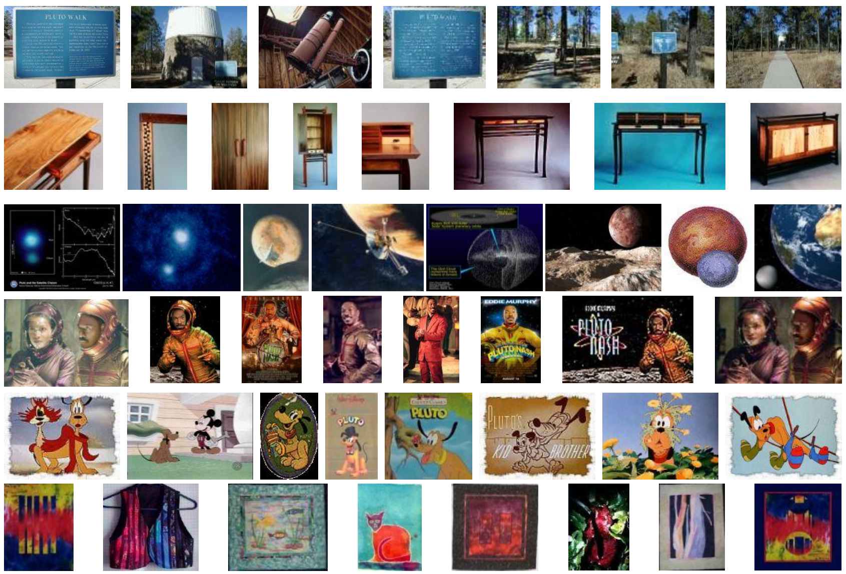


Figure 9. Six clusters of **search results of query "pluto"** using combination of textual feature and image link graph. Each row is a cluster

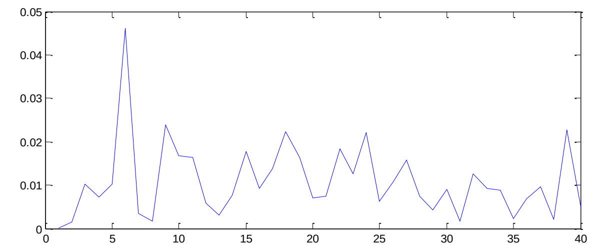
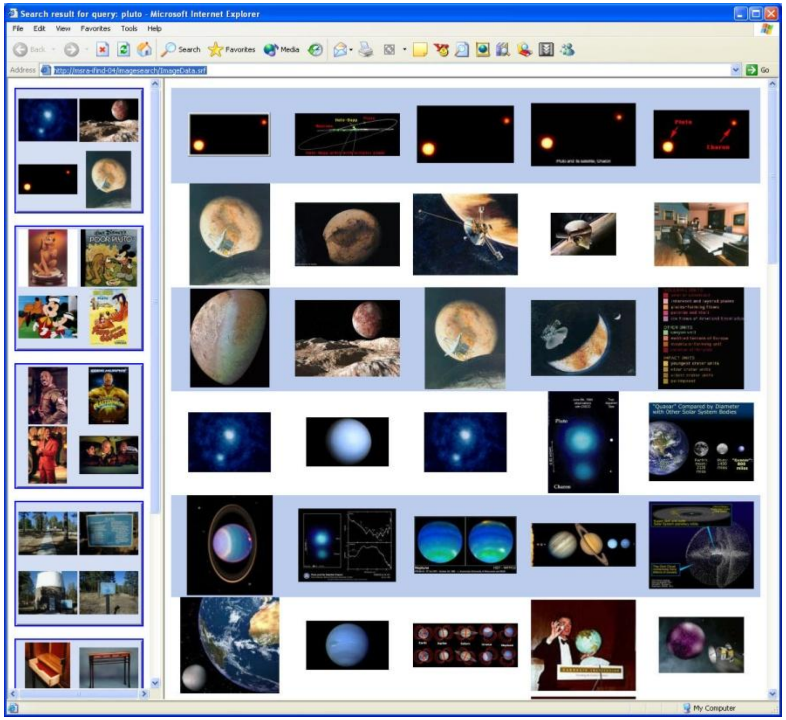


Figure 10. The Eigengap curve with *k* for the "pluto" case using textual and link combination

- Combine two affinity matrix

$$S_{combine}(i,j)=\begin{cases} S_{textual}(i,j) & if\ S_{link}(i,j)=0 \\ 1 & if\ S_{link}(i,j)>0 \end{cases}$$

Final Presentation of Our System



- Using textual and link information to get some semantic clusters
- Use low level visual feature to cluster (re-organize) each semantic cluster to facilitate user's browsing

Summary

- More improvement on web search can be made by mining webpage Layout structure
- Leverage visual cues for web information analysis & information extraction
- Demos:
  - http://www.ews.uiuc.edu/~dengcai2
    - Papers
    - VIPS demo & dll

References

- Deng Cai, Shipeng Yu, Ji-Rong Wen and Wei-Ying Ma, "Extracting Content Structure for Web Pages based on Visual Representation", The Fifth Asia Pacific Web Conference, 2003.
- Deng Cai, Shipeng Yu, Ji-Rong Wen and Wei-Ying Ma, "VIPS: a Vision-based Page Segmentation Algorithm", Microsoft Technical Report (MSR-TR-2003-79), 2003.
- Shipeng Yu, Deng Cai, Ji-Rong Wen and Wei-Ying Ma, "Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation", 12th International World Wide Web Conference (WWW2003), May 2003.
- Ruihua Song, Haifeng Liu, Ji-Rong Wen and Wei-Ying Ma, "Learning Block Importance Models for Web Pages", 13th International World Wide Web Conference (WWW2004), May 2004.
- Deng Cai, Shipeng Yu, Ji-Rong Wen and Wei-Ying Ma, "Block-based Web Search", SIGIR 2004, July 2004 .
- Deng Cai, Xiaofei He, Ji-Rong Wen and Wei-Ying Ma, "Block-Level Link Analysis", SIGIR 2004, July 2004 .
- Deng Cai, Xiaofei He, Wei-Ying Ma, Ji-Rong Wen and Hong-Jiang Zhang, "Organizing WWW Images Based on The Analysis of Page Layout and Web Link Structure", The IEEE International Conference on Multimedia and EXPO (ICME'2004) , June 2004
- Deng Cai, Xiaofei He, Zhiwei Li, Wei-Ying Ma and Ji-Rong Wen, "Hierarchical Clustering of WWW Image Search Results Using Visual, Textual and Link Analysis",12th ACM International Conference on Multimedia, Oct. 2004 .