Graph and Web Mining - Motivation, Applications and Algorithms

PROF. EHUD GUDES
DEPARTMENT OF COMPUTER SCIENCE
BEN-GURION UNIVERSITY, ISRAEL

Course Outline

- Basic concepts of Data Mining and Association rules
  - Apriori algorithm
  - Sequence mining
- Motivation for Graph Mining
- Applications of Graph Mining
- Mining Frequent Subgraphs - Transactions
  - BFS/Apriori Approach (FSG and others)
  - DFS Approach (gSpan and others)
  - Diagonal and Greedy Approaches
  - Constraint-based mining and new algorithms
- Mining Frequent Subgraphs – Single graph
  - The support issue
  - The Path-based algorithm

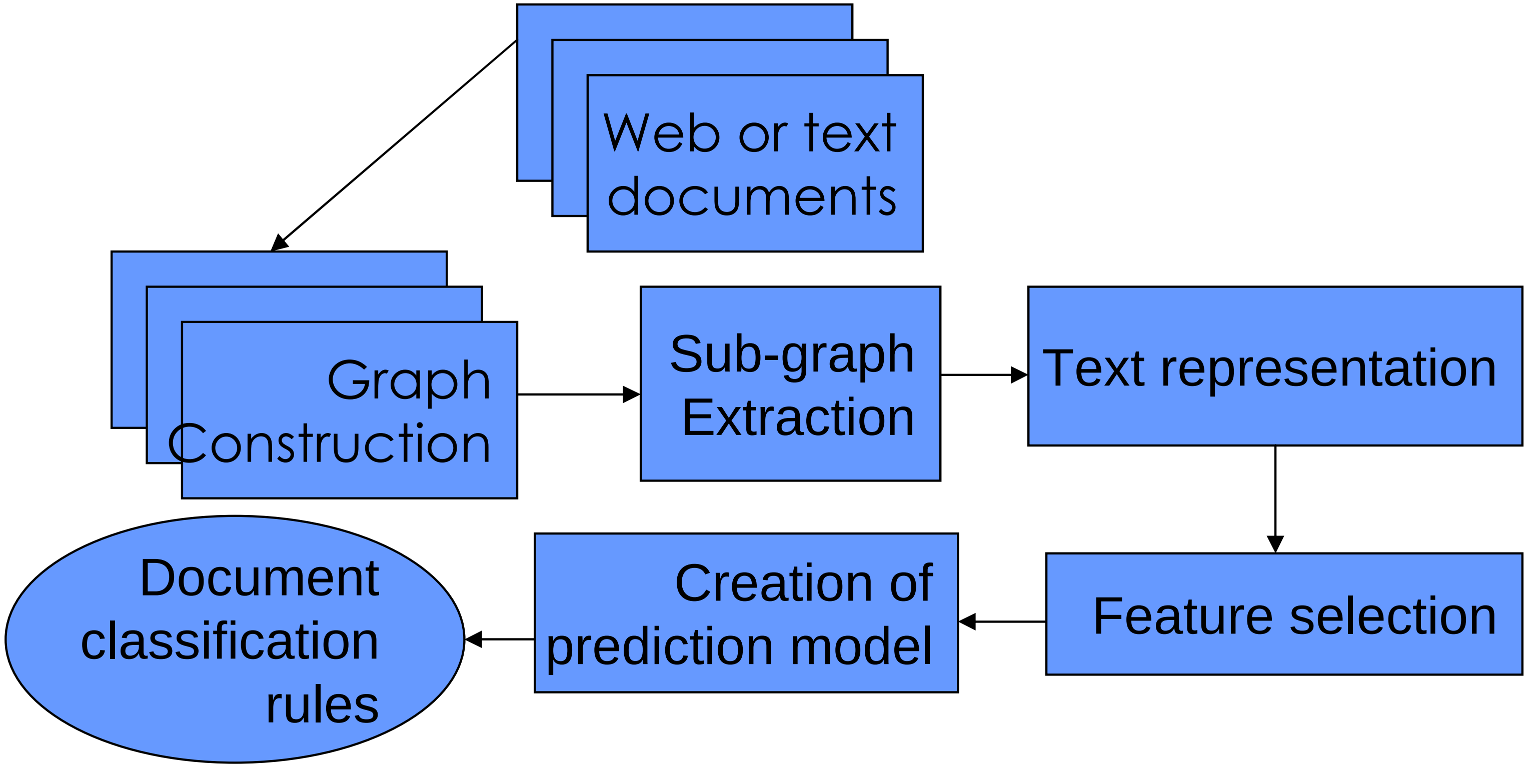
Course Outline (Cont.)

- Searching Graphs and Related algorithms
  - Sub-graph isomorphism (Sub-sea)
  - Indexing and Searching – graph indexing
  - A new sequence mining algorithm
- Web mining and other applications
  - Document classification
  - Web mining
  - Short student presentation on their projects/papers
- Conclusions

Course Outline (Cont.)

- Searching Graphs and Related algorithms
  - Sub-graph isomorphism (Sub-sea)
  - Indexing and Searching – graph indexing
  - A new sequence mining algorithm
- Web mining and other applications
  - Document classification
  - Web mining
  - Short student presentation on their projects/papers
- Conclusions

Documents classification – Last et. al. Predictive Model Induction with Hybrid Representation



Summary

- Different document representations were empirically compared in terms of classification accuracy and execution time
- The proposed hybrid methods were found to be more accurate in most cases and generally much faster than their vector-space and graph-based counterparts

Course Outline (Cont.)

- Searching Graphs and Related algorithms
  - Sub-graph isomorphism (Sub-sea)
  - Indexing and Searching – graph indexing
  - A new sequence mining algorithm
- Web mining and other applications
  - Document classification
  - Web mining
  - Short student presentation on their projects/papers
- Conclusions

Web Data Mining

EXPLORING HYPERLINKS CONTENTS,AND USAGE DATA.

Outline

- **Introduction**
- **Web Content Mining**
- **Web usage mining**
- **Web Structure Mining -** Link Analysis Algorithms
- **Web Crawlers**

Introduction

- The World-Wide Web provides every internet citizen with access to an abundance of information, but it becomes increasingly difficult to identify the relevant pieces of information.
- Web mining is a new research area that tries to address this problem by applying techniques from data mining and machine learning to Web data and documents.
- Web mining aims to discover useful information or knowledge from the Web hyperlink structure, page content and usage data.

What is Web Mining?

- **Web Content Mining:** application of data mining techniques to unstructured or semi-structured data, usually HTML-documents
- **Web Structure Mining:** use of the hyperlink structure of the Web as an (additional) information source
- **Web Usage Mining:** analysis of user interactions with a Web server (e.g., click-stream analysis)

Web Content Mining

Web Content Data Structure

- Unstructured – free text
- Semi-structured – HTML, XML and RDF data
- More structured – Table or Dynamic generated HTML pages, Images, Multi-media data
- Multi-media data mining is a "hot" area (but out of scope here…)

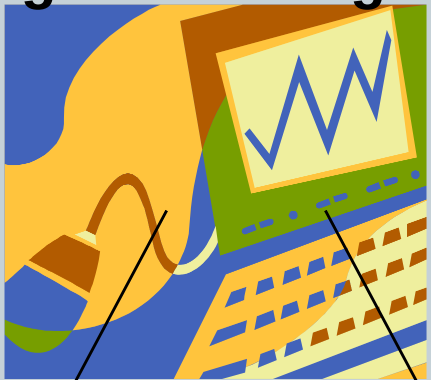
Web Content Mining - Methods

- Text Mining
  - Natural Language Processing (NLP)
  - Information Retrieval (IR)
  - Text categorization

- Structured Web page/record mining

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

Documents

Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.

Now we are engaged in a great civil war, testing whether that nation, or …

Feature Extraction

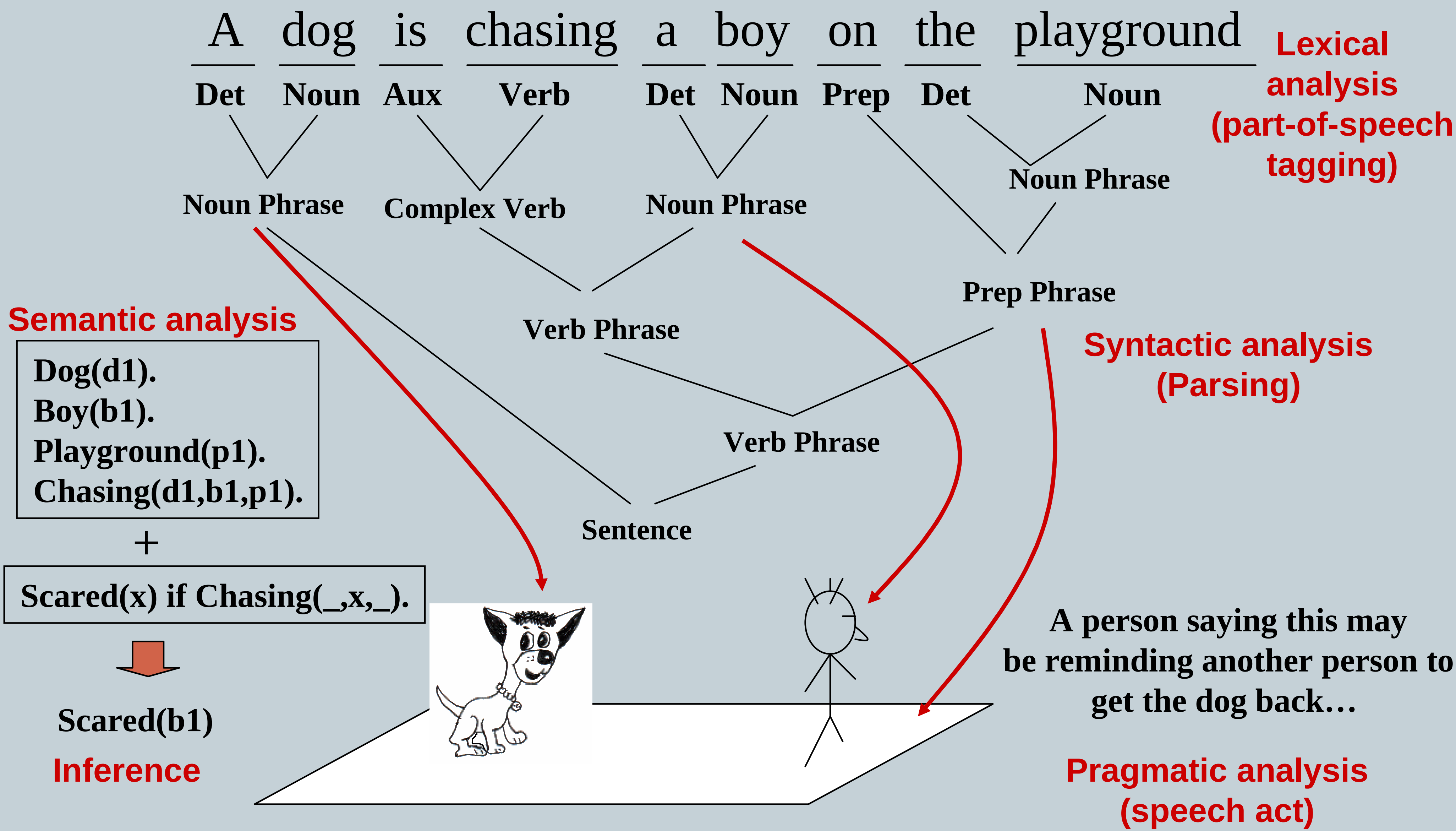
Token Sets

nation – 5
civil - 1
war – 2
men – 2
died – 4
people – 5
Liberty – 1
God – 1
…

Loses all order-specific information!

Severely limits context!

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

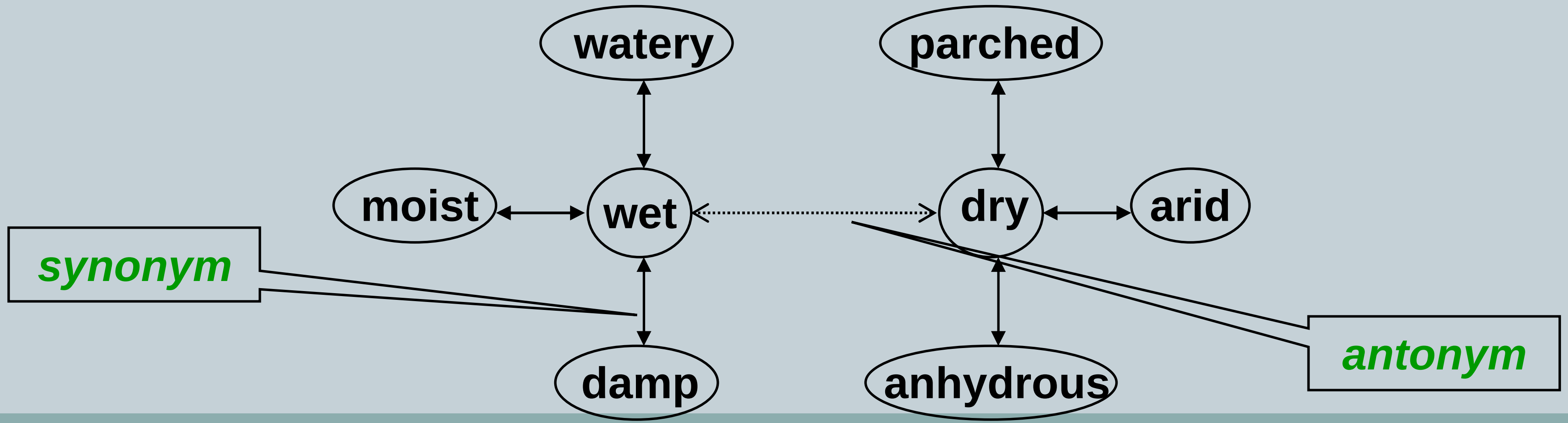
Progress on Useful Sub-Goals:

- English Lexicon e.g. Wordnet
- 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.

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

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…

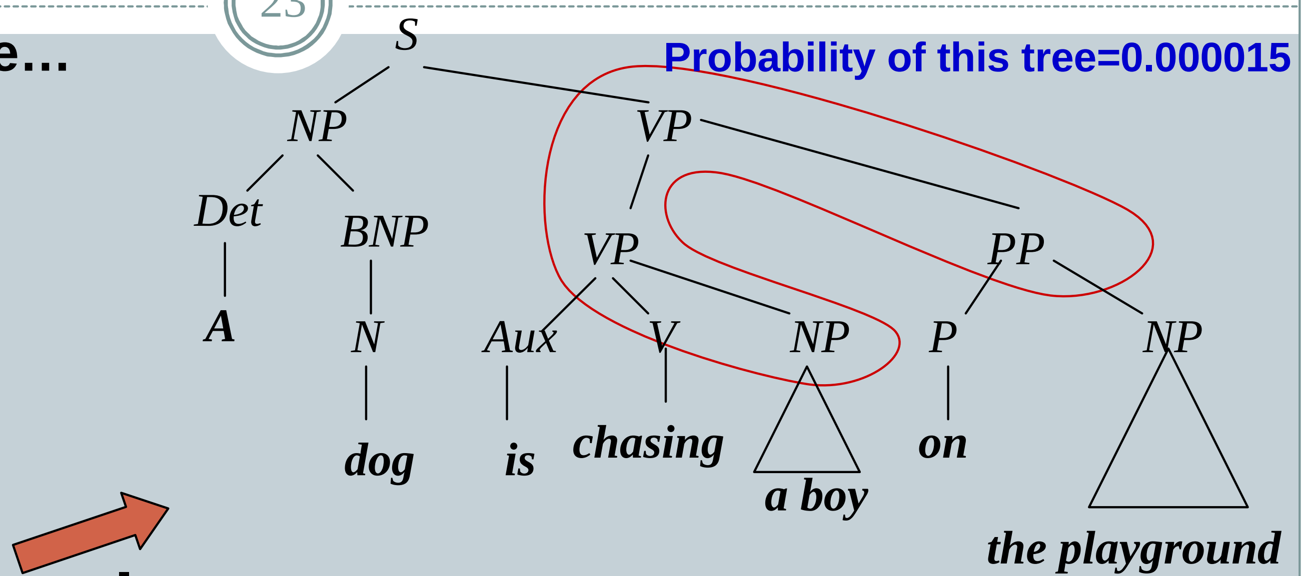
Probabilistic CFG

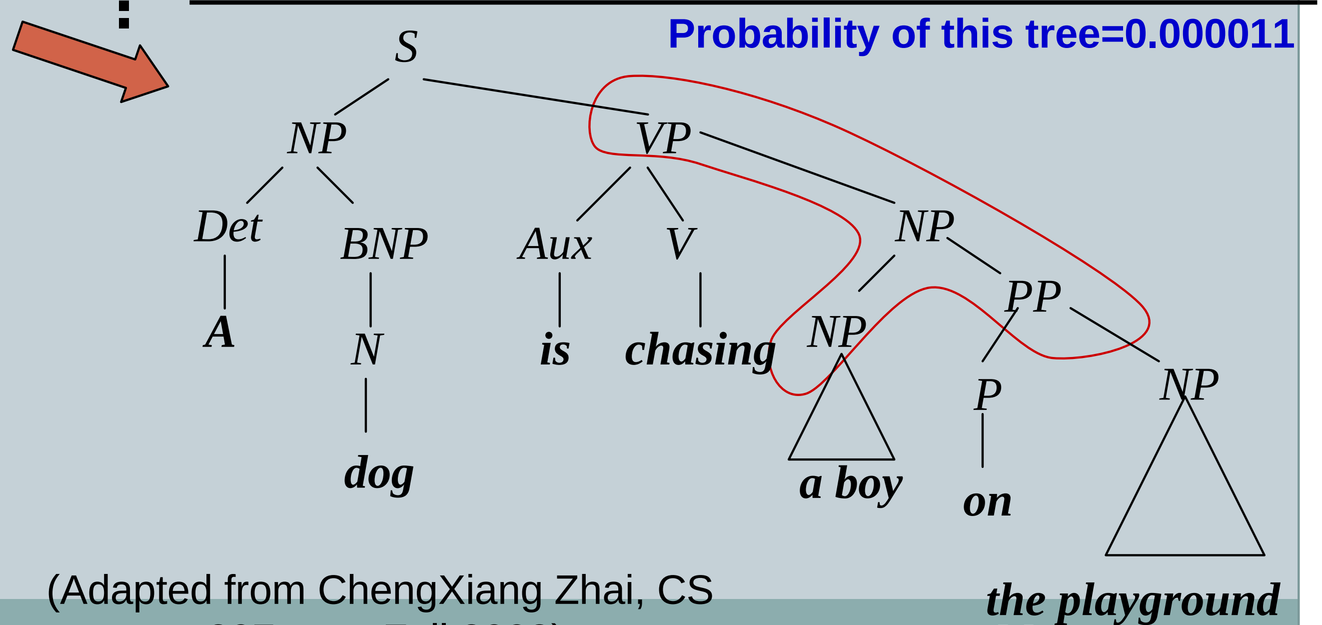
Grammar

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





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

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**.*"
- **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.

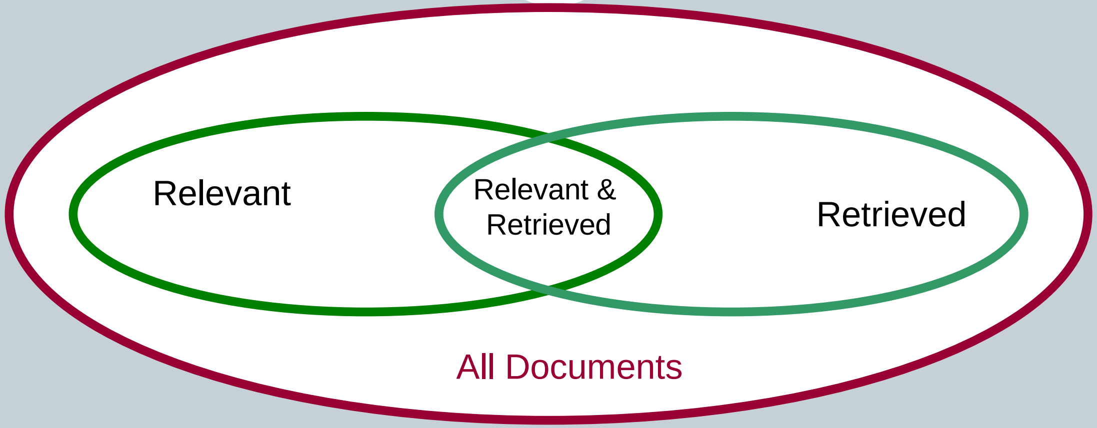
Text Databases and IR

- 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*
  - SQL or other DB query languages
- 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

$$\text{Re}\ call = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

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 × Documents Frequency Matrices
- Information Retrieval Models:
  - Boolean Model
  - Vector Model
  - Probabilistic Model

  and
  - Graph 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:

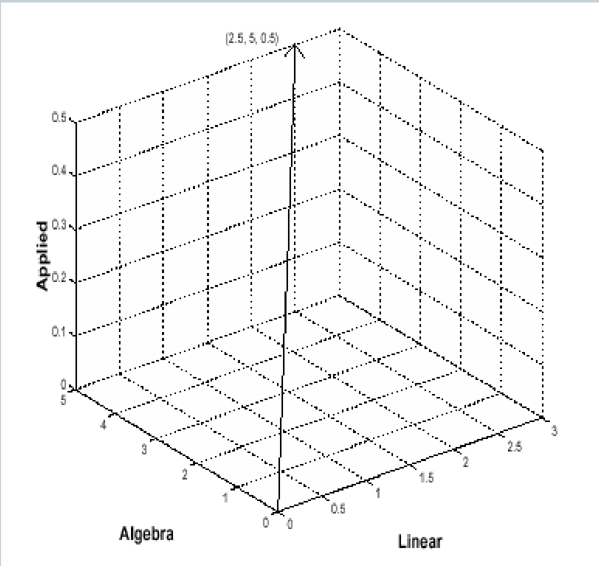
$$sim(v_1, v_2) = \frac{v_1 \cdot v_2}{|v_1||v_2|}$$

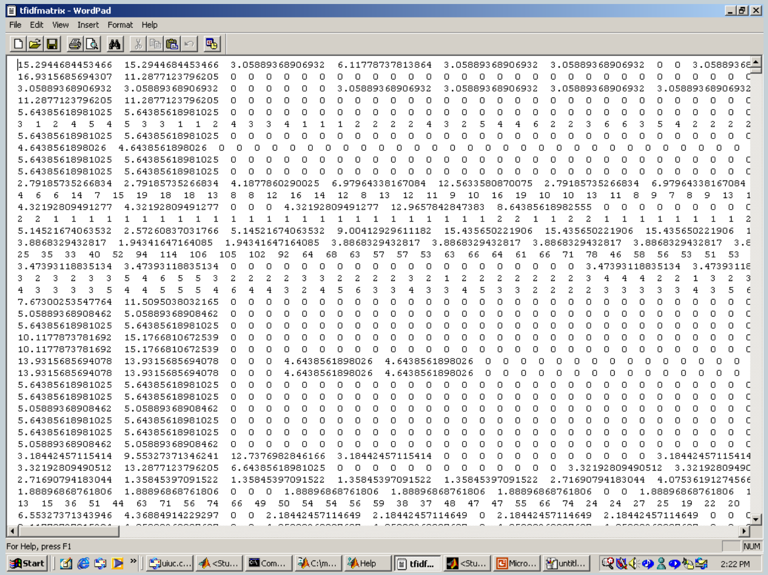
Indexing Techniques

- 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.





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

$$D_i = (w_{i1}, w_{i2}, \cdots, w_{iN})$$

$$D_j = (w_{j1}, w_{j2}, \cdots, w_{jN})$$

- 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

To whom is newdoc more similar?

Sim(newdoc,doc3)=0

map search engine govern president congress text mining travel
2.8 3.3 2.1 5.4 2.2 3.2 4.3IDF(faked) 2.4 4.5

1(2.1) 1(5.4) 2(4.8) 1(4.5) doc1
1(2.4) 2 (5.6) 1(3.3) doc2
1 (2.2) 1(3.2) 1(4.3) doc3

1(2.4) 1(4.5) newdoc

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

- 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

Text CategorizationTechniques

- Keyword-based association analysis
- Automatic document classification
- Similarity detection
  - Cluster documents by a common author
  - Cluster documents containing information from a common source
- 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

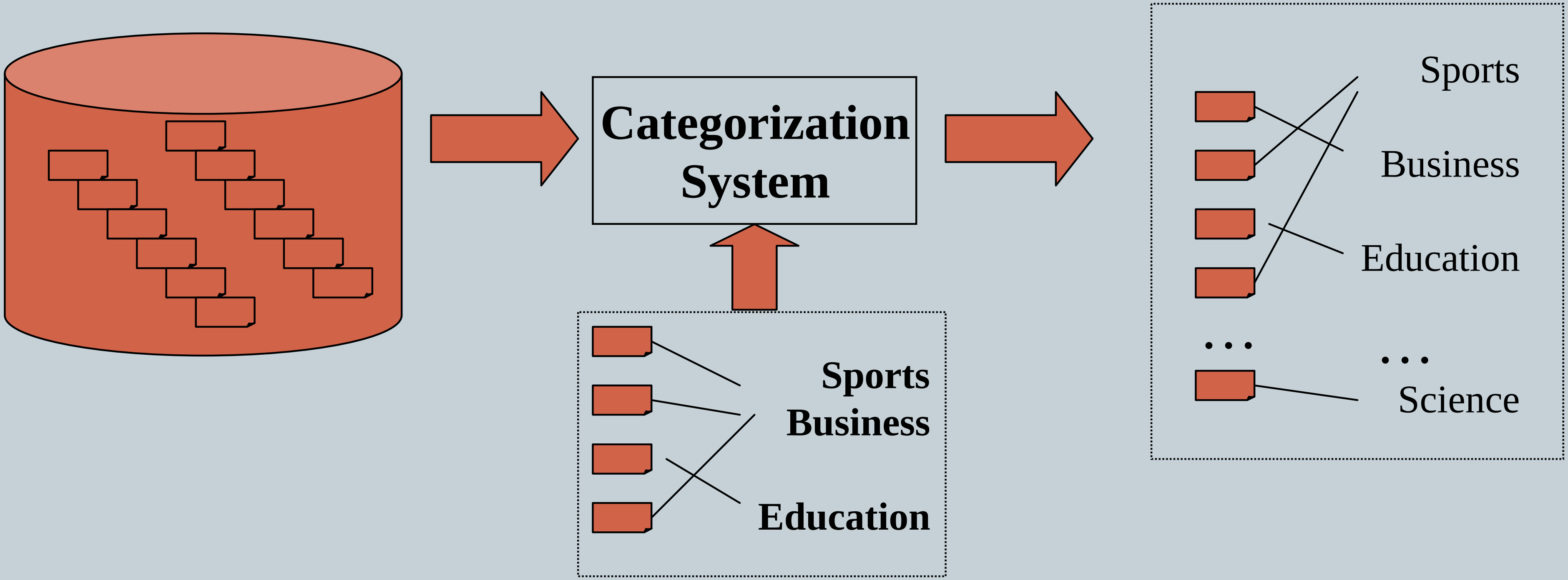
			#1	#2	#3	#4	#5
		# of documents	21,450	14,347	13,272	12,902	12,902
		# of training documents	14,704	10,667	9,610	9,603	9,603
		# of test documents	6,746	3,680	3,662	3,299	3,299
		# of categories	135	93	92	90	10
System	**Type**	**Results reported by**					
Word	(non-learning)	[Yang 1999]	.150	.310	.290		
	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



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

References

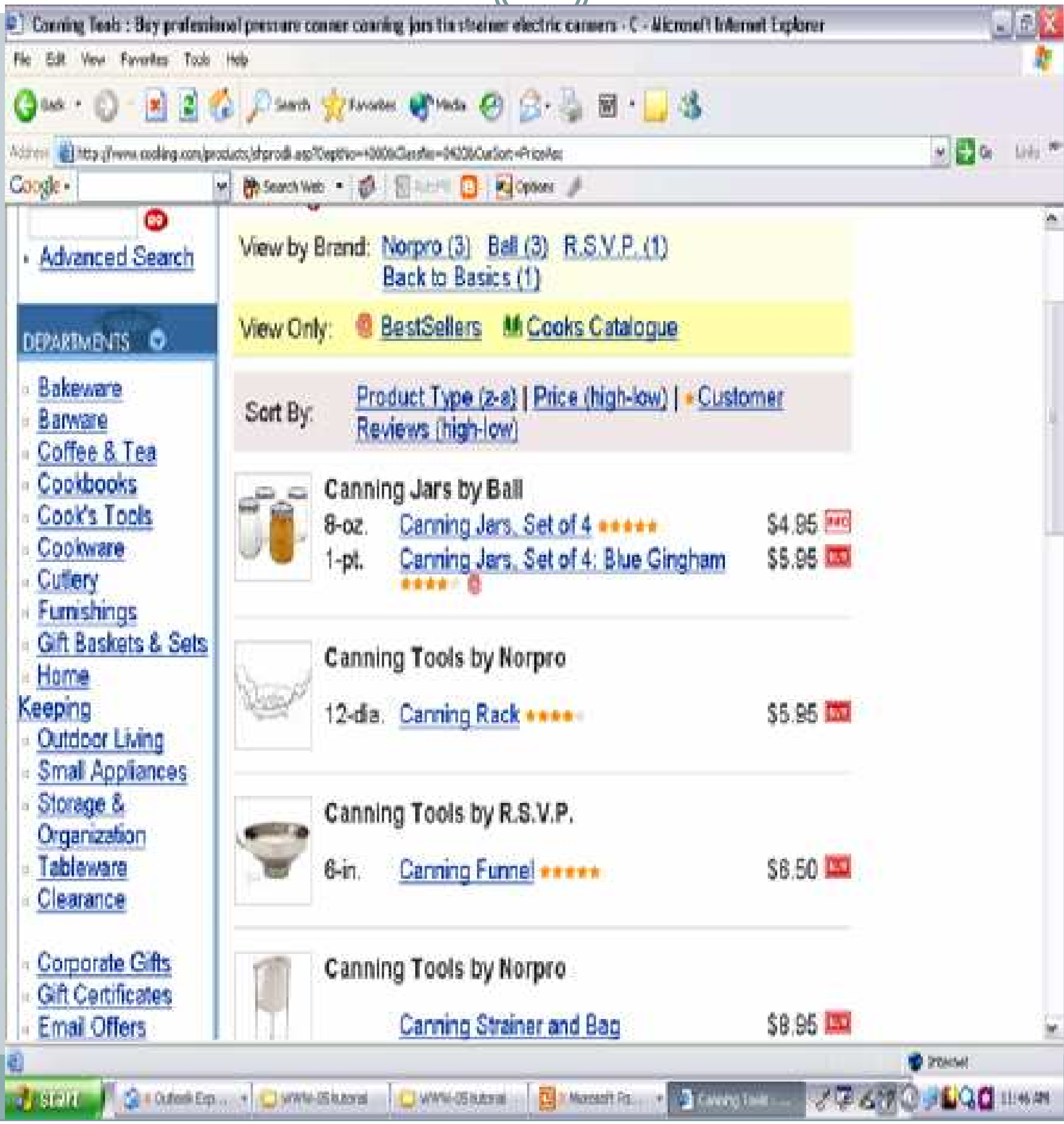
- Fabrizio Sebastiani, “Machine Learning in Automated Text Categorization”, ACM Computing Surveys, Vol. 34, No.1, March 2002
- Soumen Chakrabarti, “Data mining for hypertext: A tutorial survey”, ACM SIGKDD Explorations, 2000.
- Cleverdon, “Optimizing convenient online accesss to bibliographic databases”, 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.

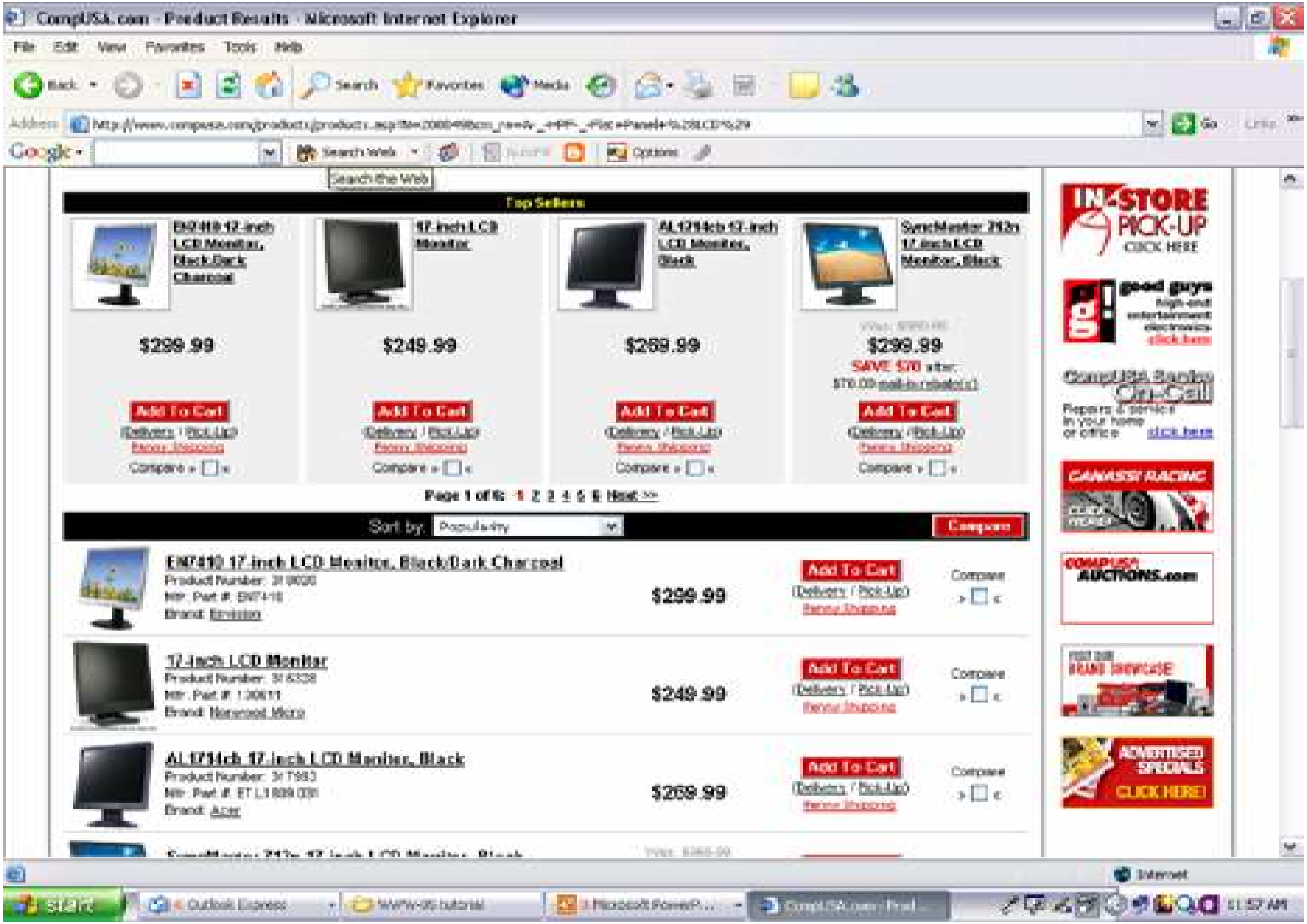
Web Content Mining - Methods

- Text Mining
  - Natural Language Processing (NLP)
  - Information Retrieval (IR)
  - Text Categorization

- **Structured Web page/record mining**
  - Deriving wrapper rules
  - Identifying data regions
  - Using vision based methods

Some Example Pages




CompUSA.com - Product Results - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Top Sellers
17-inch LCD Monitor
$299.99
$249.99
$269.99
$239.99
Add To Cart
Compare
Sort by: Popularity
Compare
17-inch LCD Monitor
$299.99
$249.99
$269.99
IN-STORE PICK-UP
good guys
ADVERTISED SPECIALS
CLICK HERE!
start

Wrapper Induction - a useful content mining method

- Given a set of manually labeled pages, a machine learning method is applied to learn extraction rules or patterns.
- The user marks the target items in a few training pages.
- The system learns extraction rules from these pages.
- The rules are applied to extract target items from other pages.

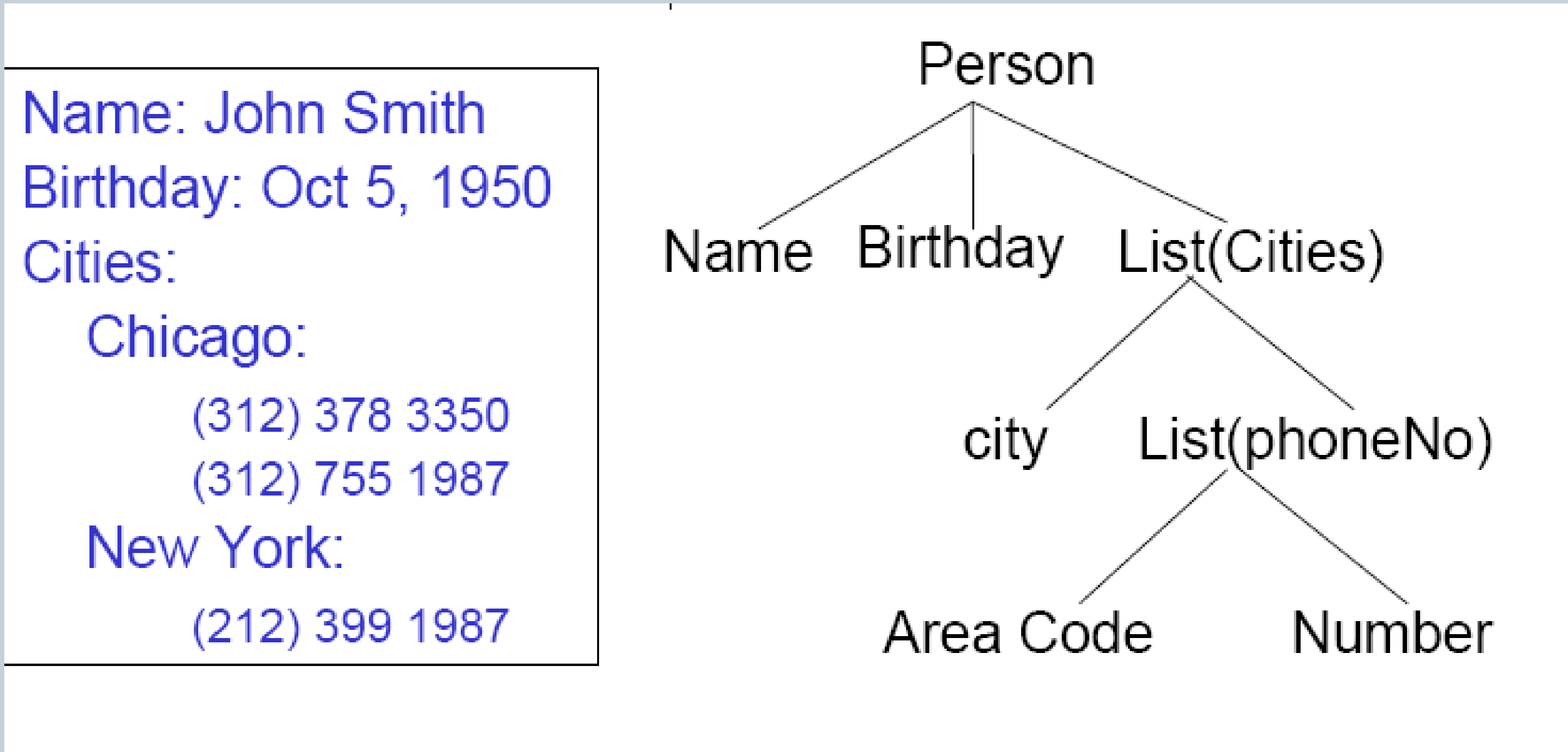
Stalker: A hierarchical wrapper induction system

- Hierarchical wrapper learning:
  - Extraction is isolated at different levels of hierarchy
  - This is suitable for nested data records (embedded list)

- Each target item is extracted using two rules
  - A start rule for detecting the beginning of the target item.
  - A end rule for detecting the ending of the target item.

Hierarchical extraction based on tree

- To extract each target item (a node), the wrapper needs a rule that extracts the item from its parent.



An example

- E1: 513 Pico, Venice, Phone 1-**800**-5551515
- E2: 90 Colfax, Palms, Phone **(800)** 508-1570
- E3: 523 1st St., LA, Phone 1-**800**-578-2293
- E4: 403 La Tijera, Watts, Phone: **(310)** 798-0008

- We want to extract area code.
  Start rules:
  R1: SkipTo(**(**)
  R2: SkipTo(-)
  End rules:
  R3: SkipTo(**)**)
  R4: SkipTo()

Learning extraction rules

- Stalker uses sequential covering to learn extraction rules for each target item.
  - In each iteration, it learns a perfect rule that covers as many positive examples as possible
- without covering any negative example.
  - Once a positive example is covered by a rule, it is removed.
  - The algorithm ends when all the positive examples are covered. The result is an ordered list of all learned rules.

Rule induction through an example

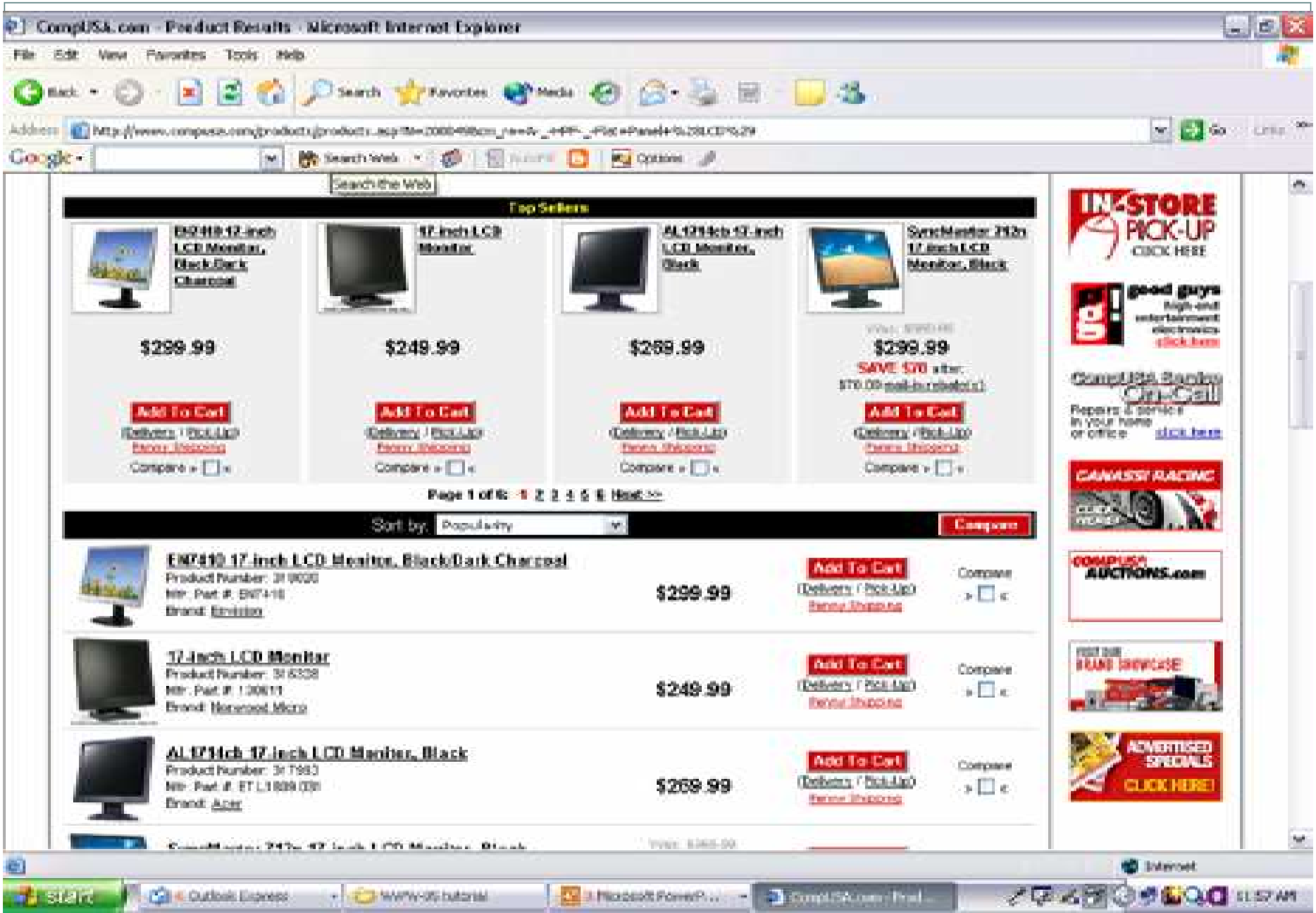
- E1: 513 Pico, Venice, Phone 1-800-555-1515
- E2: 90 Colfax, Palms, Phone (800) 508-1570
- E3: 523 1st St., LA, Phone 1-800-578-2293
- E4: 403 La Tijera, Watts, Phone: (310) 798-0008
- We learn start rule for area code.
  - Assume the algorithm starts with E2. It creates three initial candidate rules with first prefix symbol and two wildcards:
  - R1: SkipTo(()
  - R2: SkipTo(Punctuation)
  - R3: SkipTo(Anything)
  - R1 is perfect. It covers two positive examples but no negative example.

Limitations of Supervised Learning

- Manual Labeling is labor intensive and time consuming, especially if one wants to extract data from a huge number of sites.
- Wrapper maintenance is very costly:
  - If Web sites change frequently
  - It is necessary to detect when a wrapper stops to work properly.
  - Any change may make existing extraction rules invalid.
  - Re-learning is needed, and most likely manual relabeling as well.

Automatic data extraction

- Input: A single Web page with multiple data records (at least 2).
- Objective: Automatically (no human involvement)
  - Step1: Identify data records in a page, and
  - Step 2: align and extract data items from them
- Method: Identify data regions


CompUSA.com - Product Results - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Top Sellers
17-inch LCD Monitor
$299.99
$249.99
$269.99
$299.99
Add To Cart
Add To Cart
Add To Cart
Add To Cart
Sort by: Popularity
Compare
$299.99
17-inch LCD Monitor
$249.99
$269.99
IN-STORE PICK-UP
CLICK HERE
ADVERTISED SPECIALS
CLICK HERE!

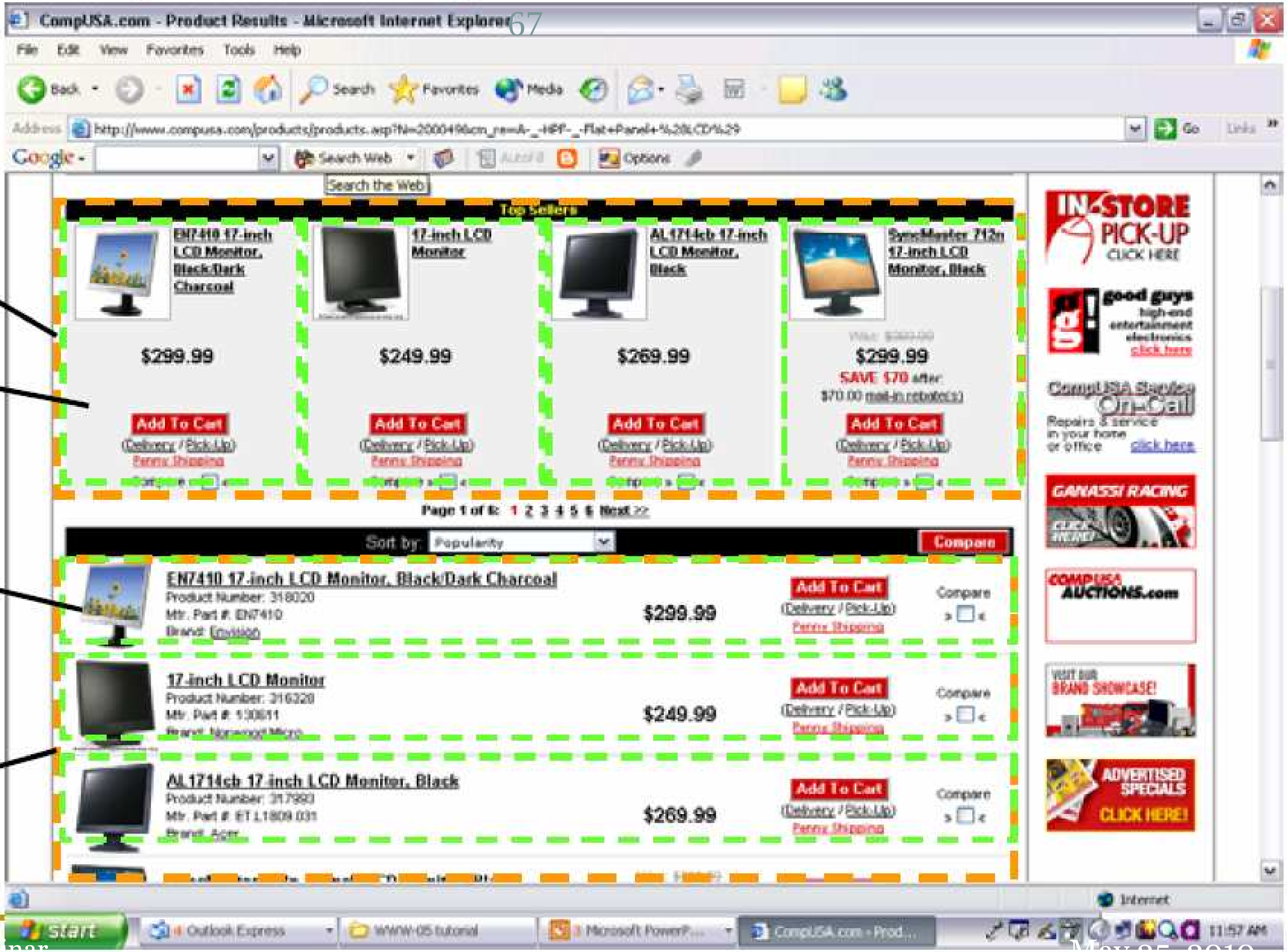
1. Identify data regions and data records

Data region1

A data record

A data record

Data region2



2. Align and extract data items (e.g., region1)

image 1	EN7410 17-inch LCD Monitor Black/Dark charcoal		$299.99		Add to Cart	(Delivery / Pick-Up)	Penny Shopping	Compare
image 2	17-inch LCD Monitor		$249.99		Add to Cart	(Delivery / Pick-Up)	Penny Shopping	Compare
image 3	AL1714 17-inch LCD Monitor, Black		$269.99		Add to Cart	(Delivery / Pick-Up)	Penny Shopping	Compare
image 4	SyncMaster 712n 17-inch LCD Monitor, Black	Was: $369.99	$299.99	Save $70 After: $70 mail-in-rebate(s)	Add to Cart	(Delivery / Pick-Up)	Penny Shopping	Compare

Mining Data Records

- Given a single page with multiple data records, MDR extracts data records, but not data items (step 1)

Mining Data Records is based on

- two observations about data records in a Web page
- a string matching algorithm

Two observations

- A group of data records are presented in a **contiguous region** (a data region) of a page and are formatted using similar tags.

- A group of data records being placed in a data region are **under one parent** node and consists of children nodes.

Example

1.

Customer Rating:

Apple iBook Notebook M8600LL/A (600-MHz PowerPC G3, 128 MB RAM, 20 GB hard drive)

Buy new: $1,194.00

Usually ships in 1 to 2 days

Best use: (what's this?)	Business:	Portability:	Desktop Replacement:	Entertainment:
	●●●○○	●●●●●	●●●○○	●●●○○

600 MHz PowerPC G3, 128 MB SRAM, 20 GB Hard Disk, 24x CD-ROM, AirPort ready, and Mac OS X, Mac OS X,Mac OS 9.2,Quick Time,iPhoto,iTunes 2,iMovie 2,AppleWorks,Microsoft IE

2.

Customer Rating:

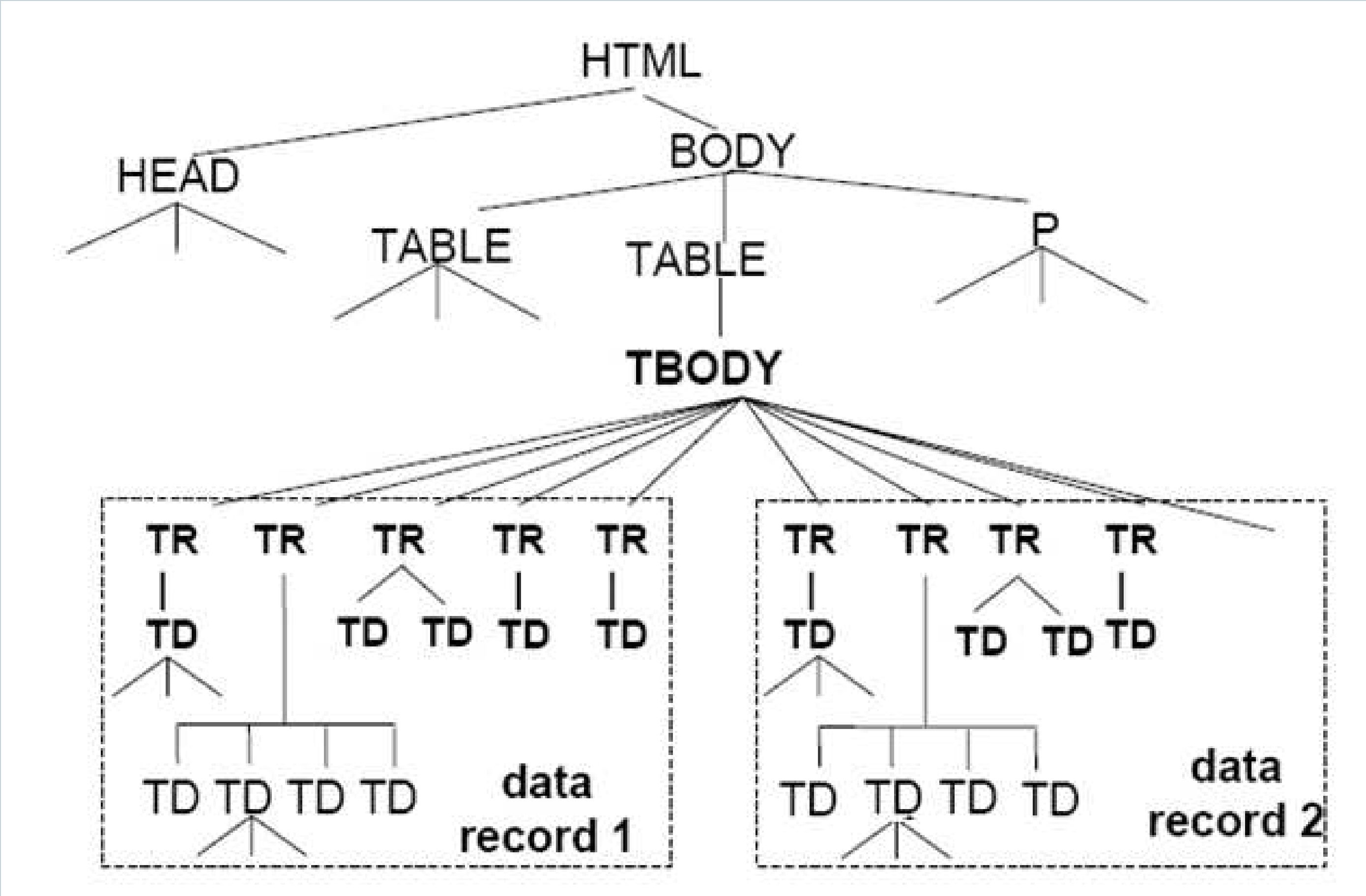
Apple Powerbook Notebook M8591LL/A (667-MHz PowerPC G4, 256 MB RAM, 30 GB hard drive)

Buy new: $2,399.99

Best use: (what's this?)	Portability:	Desktop Replacement:	Entertainment:
	●●●●○	●●●●○	●●●●○

667 MHz PowerPC G4, 256 MB SDRAM, 30 GB Ultra ATA Hard Disk, 24x (read), 8x (write) CD-RW, 8x; included via combo drive DVD-ROM, and Mac OS X, QuickTime, iMovie 2, iTunes(6), Microsoft Internet Explorer, Microsoft Outlook Express, ...

Tag tree of the previous page



The approach

Given a page, three steps:

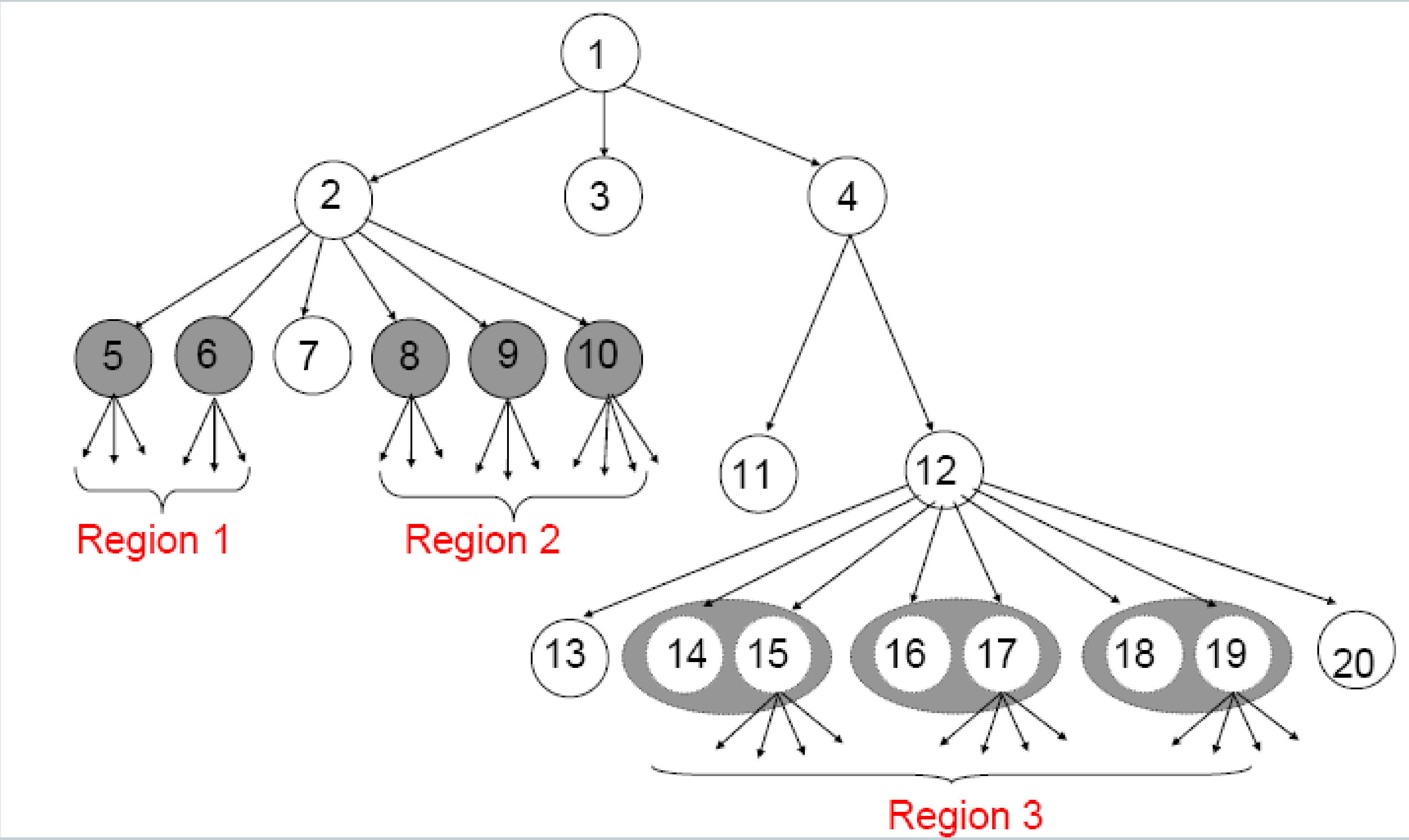
- Building the HTML Tag Tree
- Mining Data Regions
- Identifying Data Records

Mining data regions

- Find every data region with similar data records.
- **Definition: A *generalized node*** of length *r consists of r (r ≥ 1) nodes in the HTML tag* tree with the following two properties:
  1. the nodes all have the same parent.
  2. the nodes are adjacent.
- **Definition: A *data region*** *is a collection of two or more* generalized nodes with the following properties:
  1. the generalized nodes all have the same parent.
  2. the generalized nodes are all adjacent.
  3. adjacent generalized nodes are similar.

An illustration of generalized nodes and data regions

- Shaded nodes are generalized nodes



Identify Data Records

- A generalized node may not be a data record.
- Extra mechanisms are needed to identify true atomic objects.

Name 1 Description of object 1	Name 2 Description of object 2
Name 3 Description of object 3	Name 4 Description of object 4

Name 1	Name 2
Description of object 1	Description of object 2
Name 3	Name 4
Description of object 3	Description of object 4

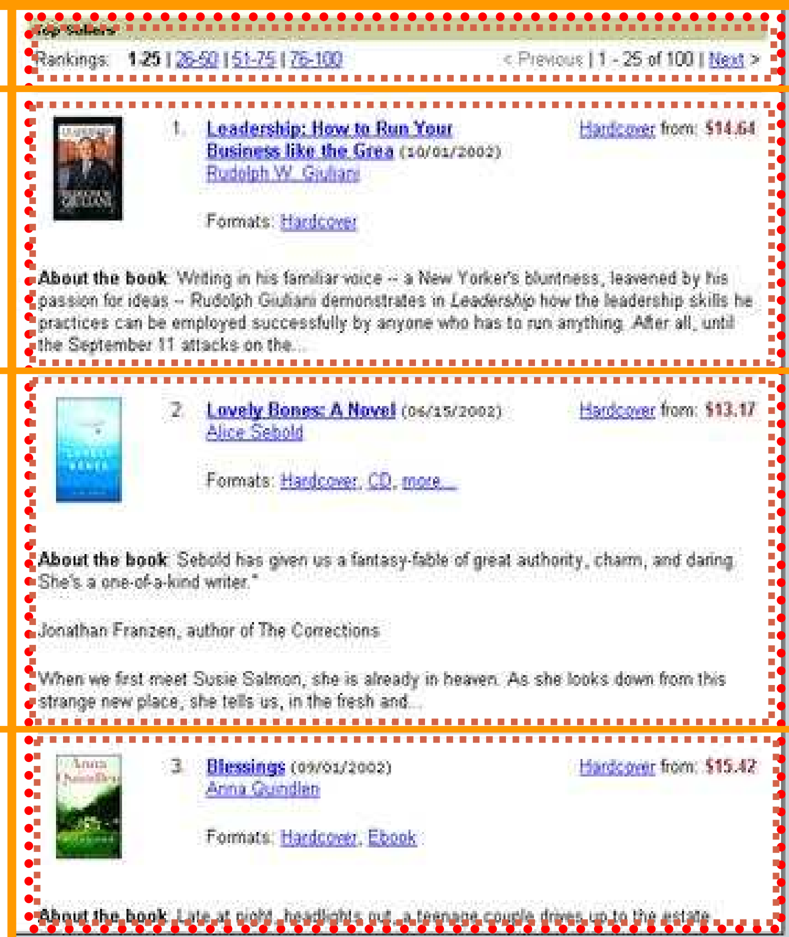
Once I got the data record…

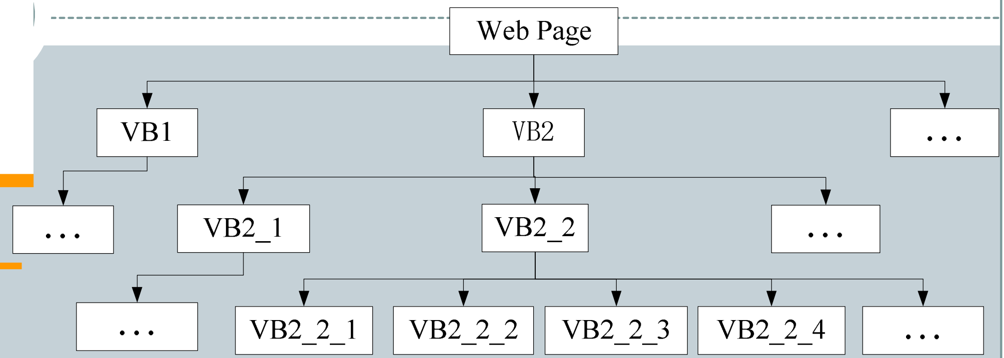
- Data records enable object level search (rather than current page level search): E.g.,
- if one can extract all the product data records on the Web, one can built a product search engine, by treating each data record/product as a Web page.
- Meta-search: re-ranking of search results from multiple search engines.
- Extract data items from data records and put them in tables for querying.

VIPS Algorithm – Vision based

- 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

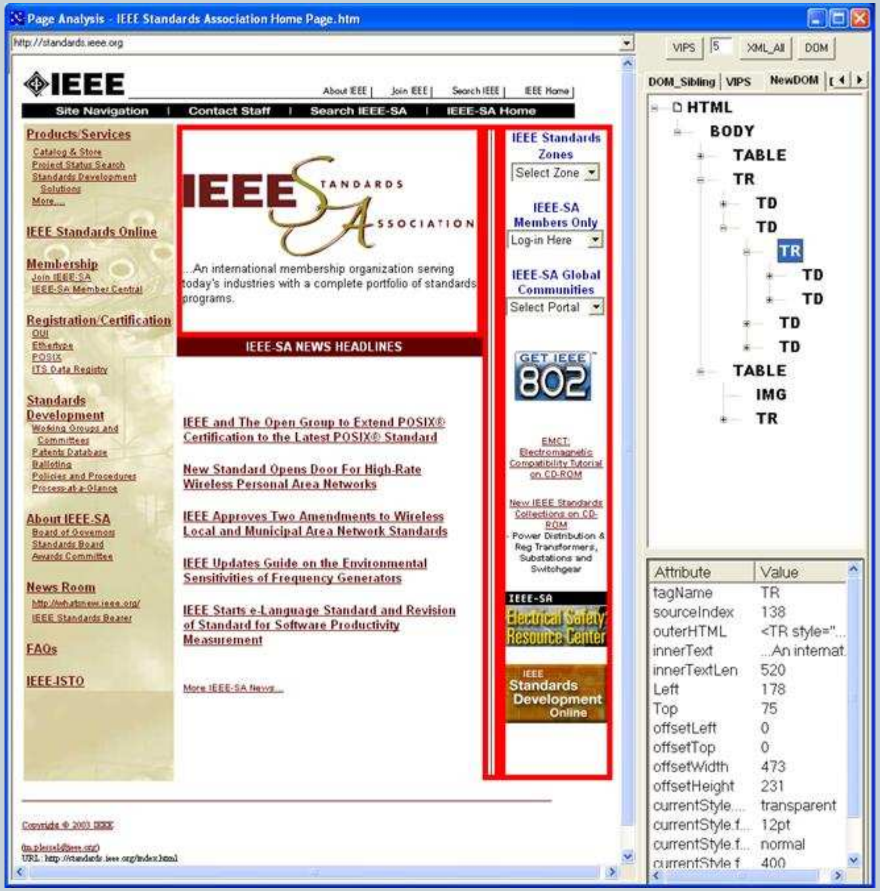
VIPS: An Example



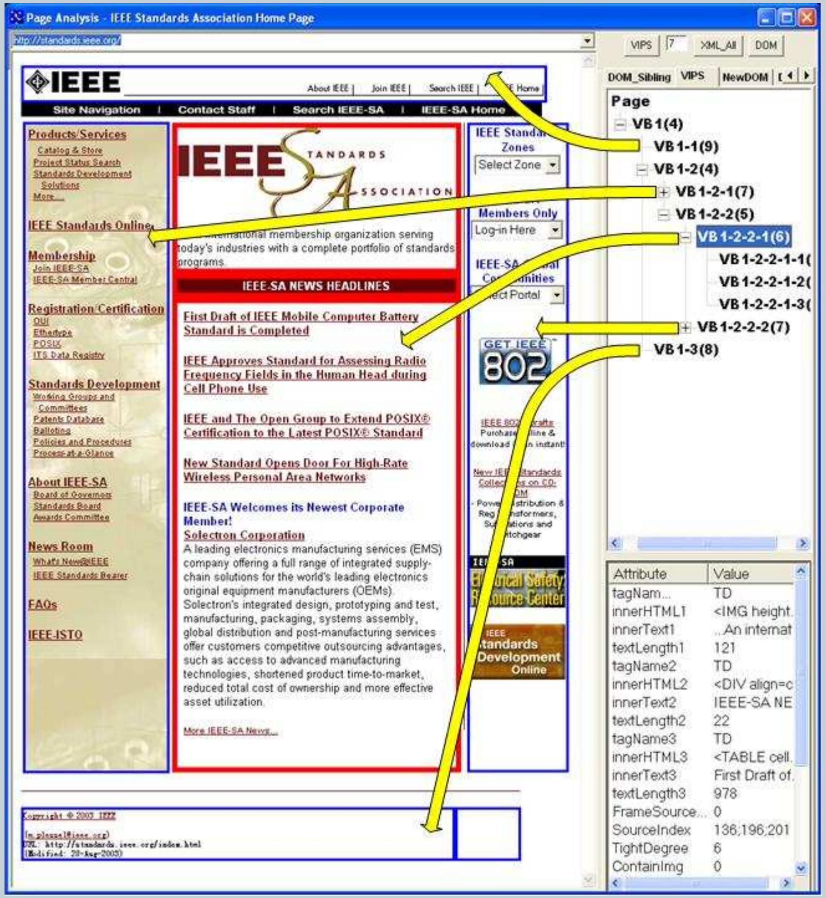


- A hierarchical structure of layout block
- A *Degree of Coherence (DOC)* is defined for each block
  - Show the intra coherence of the block
  - *DoC* of child block must be no less than its parent's
- The *Permitted Degree of Coherence (PDOC)* can be pre-defined to achieve different granularities 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)

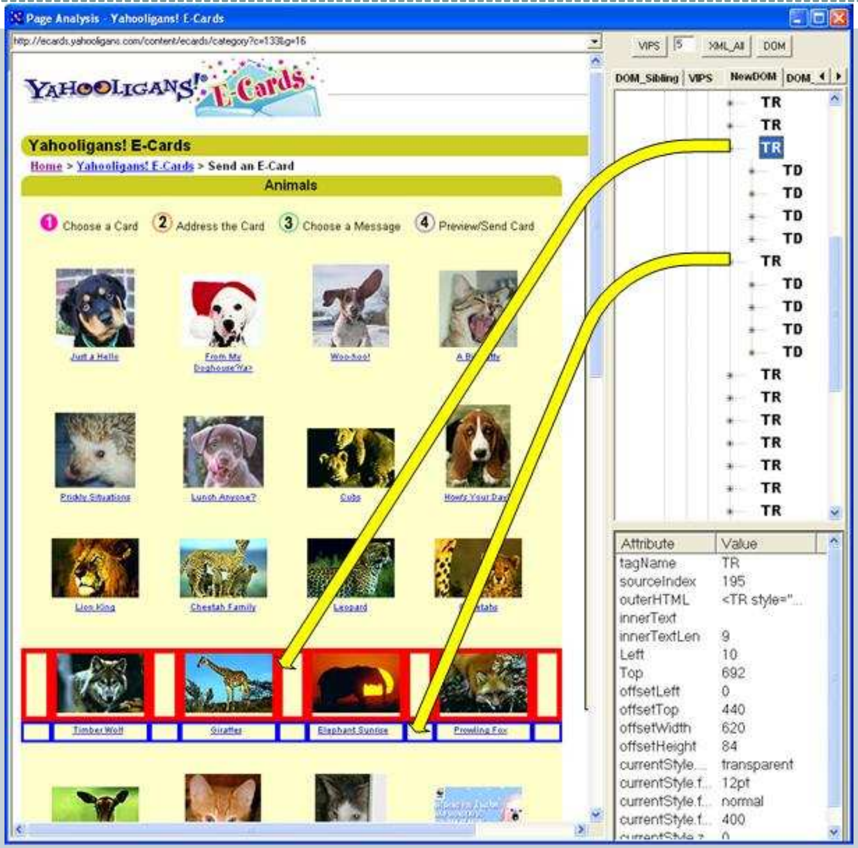


(DOM Structure)



(VIPS Structure)

Example of Web Page Segmentation (2)



(DOM Structure) (VIPS Structure)

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

References

- **Ion Muslea, Steven Minton, Craig A. Knoblock: Hierarchical Wrapper Induction for Semistructured Information Sources. Autonomous Agents and Multi-Agent Systems 4(1/2): 93-114 (2001)**

- **Bing Liu, Yanhong Zhai: NET - A System for Extracting Web Data from Flat and Nested Data Records. WISE 2005: 487-495**

- **Deng Cai, Shipeng Yu, Ji-Rong Wen, Wei-Ying Ma: Extracting Content Structure for Web Pages Based on Visual Representation. APWeb 2003: 406-417**

Web usage mining

Introduction

- **Web usage mining**: automatic discovery of patterns in clickstreams and associated data collected or generated as a result of user interactions with one or more Web sites.
- **Goal**: analyze the behavioral patterns and profiles of users interacting with a Web site.
- The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed by groups of users with common interests.

Web Usage Mining

- Typical problems: Distinguishing among unique users, server sessions, episodes, etc in the presence of caching and proxy servers
- Often Usage Mining uses some background or domain knowledge

E.g. site topology, Web content, etc

Web Usage Mining

- Two main categories:
  - ✓ Learning a user profile (personalized)

    Web users would be interested in techniques that learn their needs and preferences automatically
  - ✓ Learning user navigation patterns (impersonalized)

    Information providers would be interested in techniques that improve the effectiveness of their Web site or biasing the users towards the goals of the site

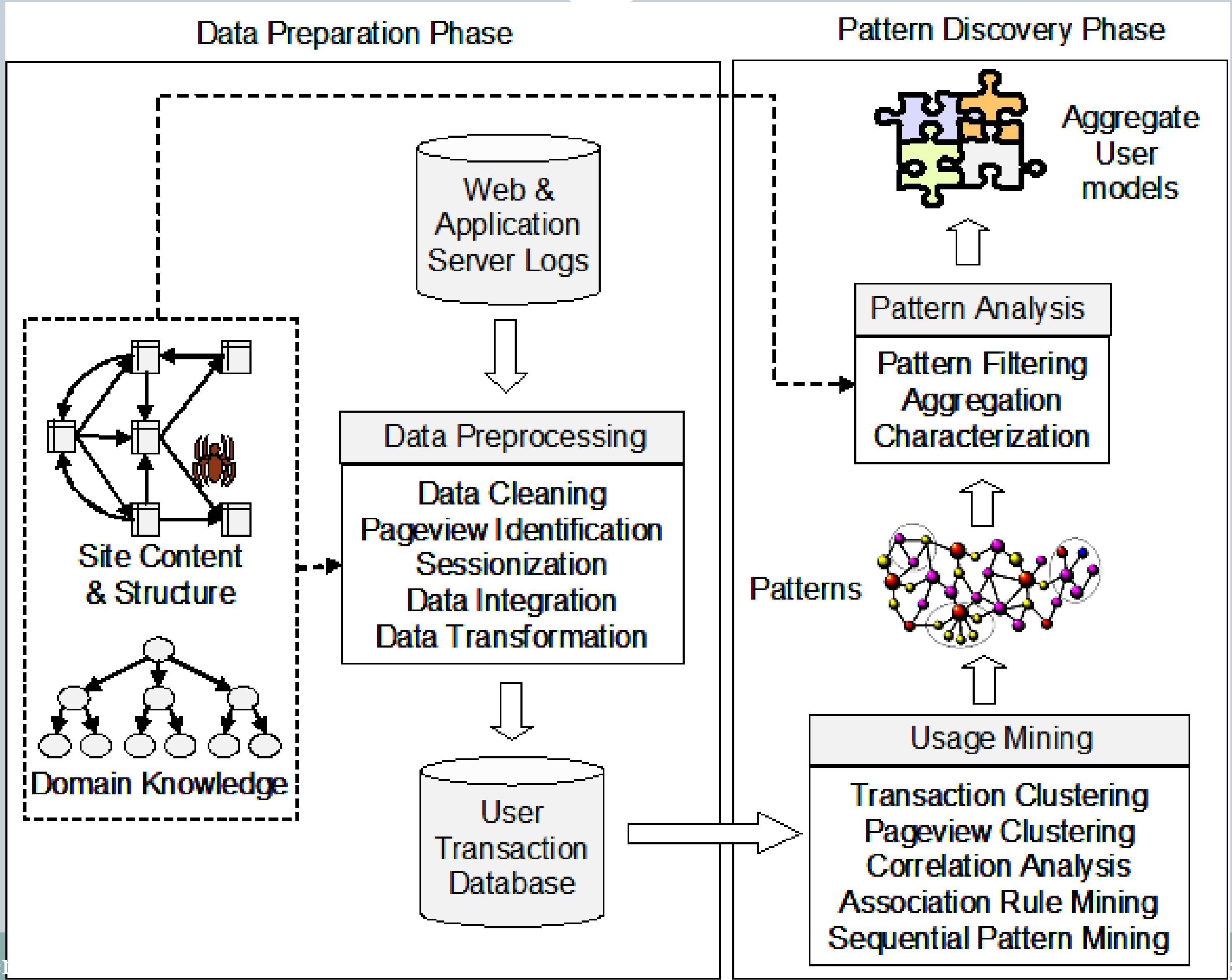
Introduction

- Data in Web Usage Mining:
  - **Web server logs**
  - Site contents
  - Data about the visitors, gathered from external channels
- Not all these data are always available.
- When they are, they must be integrated.
- A large part of Web usage mining is about processing usage/ clickstream data.
  - After that various data mining algorithm can be applied.

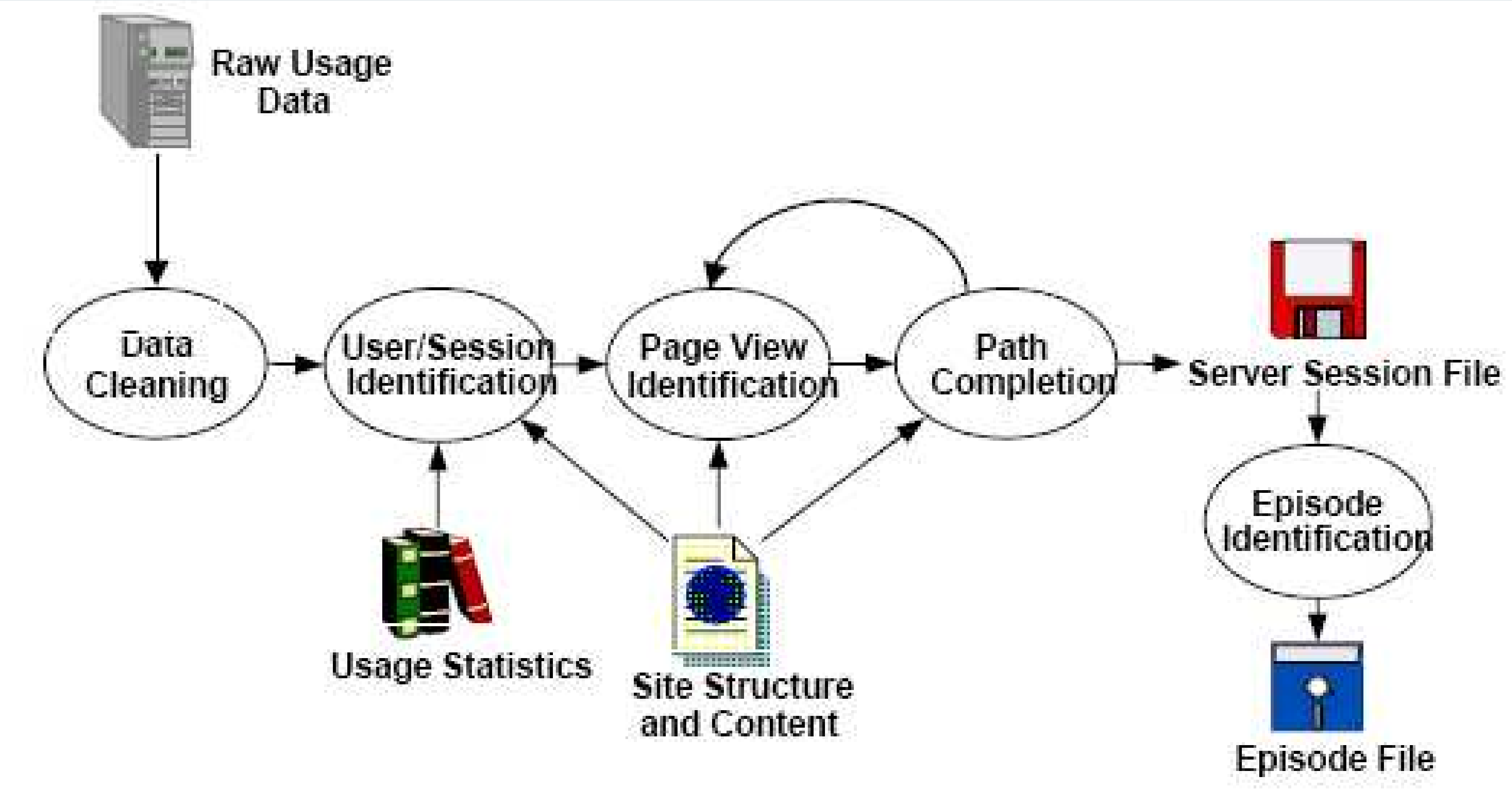
Web server logs

1	2006-02-01 00:08:43 1.2.3.4 - GET /classes/cs589/papers.html - 200 9221 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+.NET+CLR+2.0.50727) http://dataminingresources.blogspot.com/
2	2006-02-01 00:08:46 1.2.3.4 - GET /classes/cs589/papers/cms-tai.pdf - 200 4096 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+.NET+CLR+2.0.50727) http://maya.cs.depaul.edu/~classes/cs589/papers.html
3	2006-02-01 08:01:28 2.3.4.5 - GET /classes/ds575/papers/hyperlink.pdf - 200 318814 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1) http://www.google.com/search?hl=en&lr=&q=hyperlink+analysis+for+the+web+survey
4	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/announce.html - 200 3794 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/
5	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/styles2.css - 200 1636 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/announce.html
6	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/header.gif - 200 6027 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/announce.html

Web usage mining process



Pre-processing of web usage data



Data cleaning

- Data cleaning
  - remove irrelevant references and fields in server logs
  - remove references due to spider navigation
  - remove erroneous references
  - add missing references due to caching (done after sessionization)

Identify sessions (sessionization)

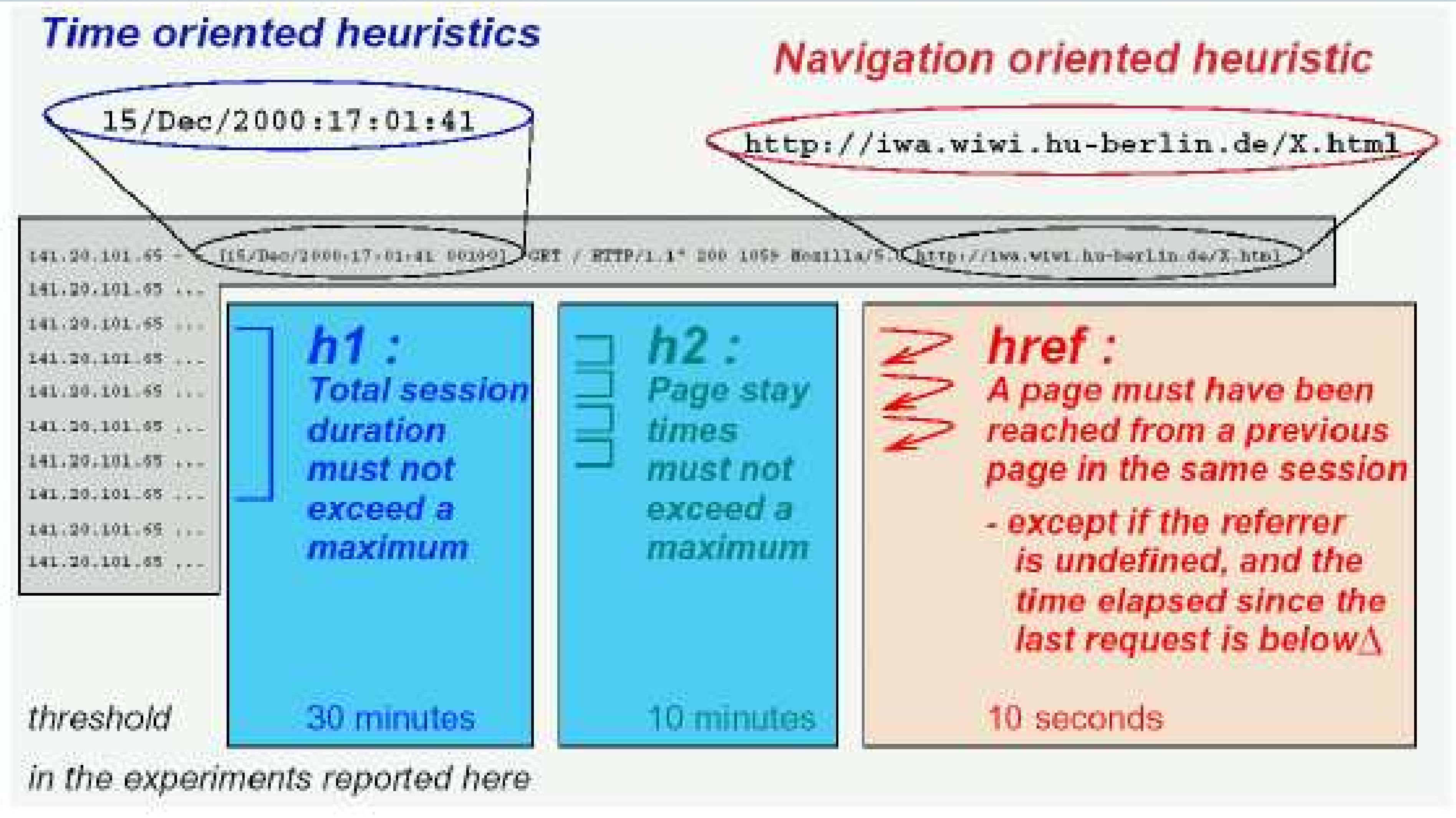
- In Web usage analysis, these data are the sessions of the site visitors: the activities performed by a user from the moment she enters the site until the moment she leaves it.
- Difficult to obtain reliable usage data due to proxy servers and anonymizers, dynamic IP addresses, missing references due to caching, and the inability of servers to distinguish among different visits.

Sessionization strategies

Session reconstruction =
correct mapping of activities to different individuals +
correct separation of activities belonging to different visits of the same individual

While users navigate the site: identify …		In the analysis of log files: identify …		Resulting partitioning of the log file
users by	sessions by	users by	sessions by	
—	—	IP & Agent	sessionization heuristics	constructed sessions (“**u-ipa**”)
cookies	—	—	sessionization heuristics	constructed sessions (“**cookies**”)
cookies	embedded session IDs	—	—	real sessions

Sessionization heuristics



Sessionization example

User 1

Time	IP	URL	Ref
0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Session 1

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C

Session 2

1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Fig. 12.5. Example of sessionization with a time-oriented heuristic

User identification

Method	Description	Privacy Concerns	Advantages	Disadvantages
IP Address + Agent	Assume each unique IP address/Agent pair is a unique user	Low	Always available. No additional technology required.	Not guaranteed to be unique. Defeated by rotating IPs.
Embedded Session Ids	Use dynamically generated pages to associate ID with every hyperlink	Low to medium	Always available. Independent of IP addresses.	Cannot capture repeat visitors. Additional overhead for dynamic pages.
Registration	User explicitly logs in to the site.	Medium	Can track individuals not just browsers	Many users won't register. Not available before registration.
Cookie	Save ID on the client machine.	Medium to high	Can track repeat visits from same browser.	Can be turned off by users.
Software Agents	Program loaded into browser and sends back usage data.	High	Accurate usage data for a single site.	Likely to be rejected by users.

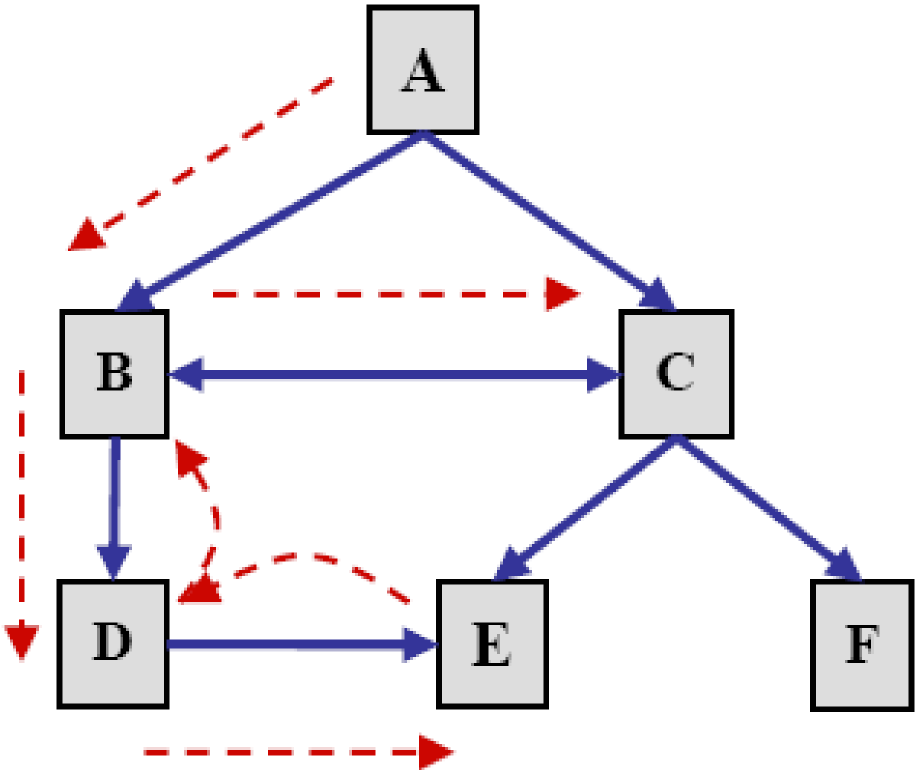
Pageview

- A pageview is an aggregate representation of a collection of Web objects contributing to the display on a user's browser resulting from a single user action (such as a click-through).
- Conceptually, each pageview can be viewed as a collection of Web objects or resources representing a specific "user event," e.g., reading an article, viewing a product page, or adding a product to the shopping cart.

Path completion

- Client- or proxy-side caching can often result in missing access references to those pages or objects that have been cached.
- For instance,
  - if a user returns to a page A during the same session, the second access to A will likely result in viewing the previously downloaded version of A that was cached on the client-side, and therefore, no request is made to the server.
  - This results in the second reference to A not being recorded on the server logs.

Missing references due to caching



User's actual navigation path:

A →B → D → E → D → B → C

What the server log shows:

URL	Referrer
A	--
B	A
D	B
E	D
C	B

Fig. 12.7. Missing references due to caching.

Path completion

- The problem of inferring missing user references due to caching.
- Effective path completion requires extensive knowledge of the link structure within the site
- Referrer information in server logs can also be used in disambiguating the inferred paths.

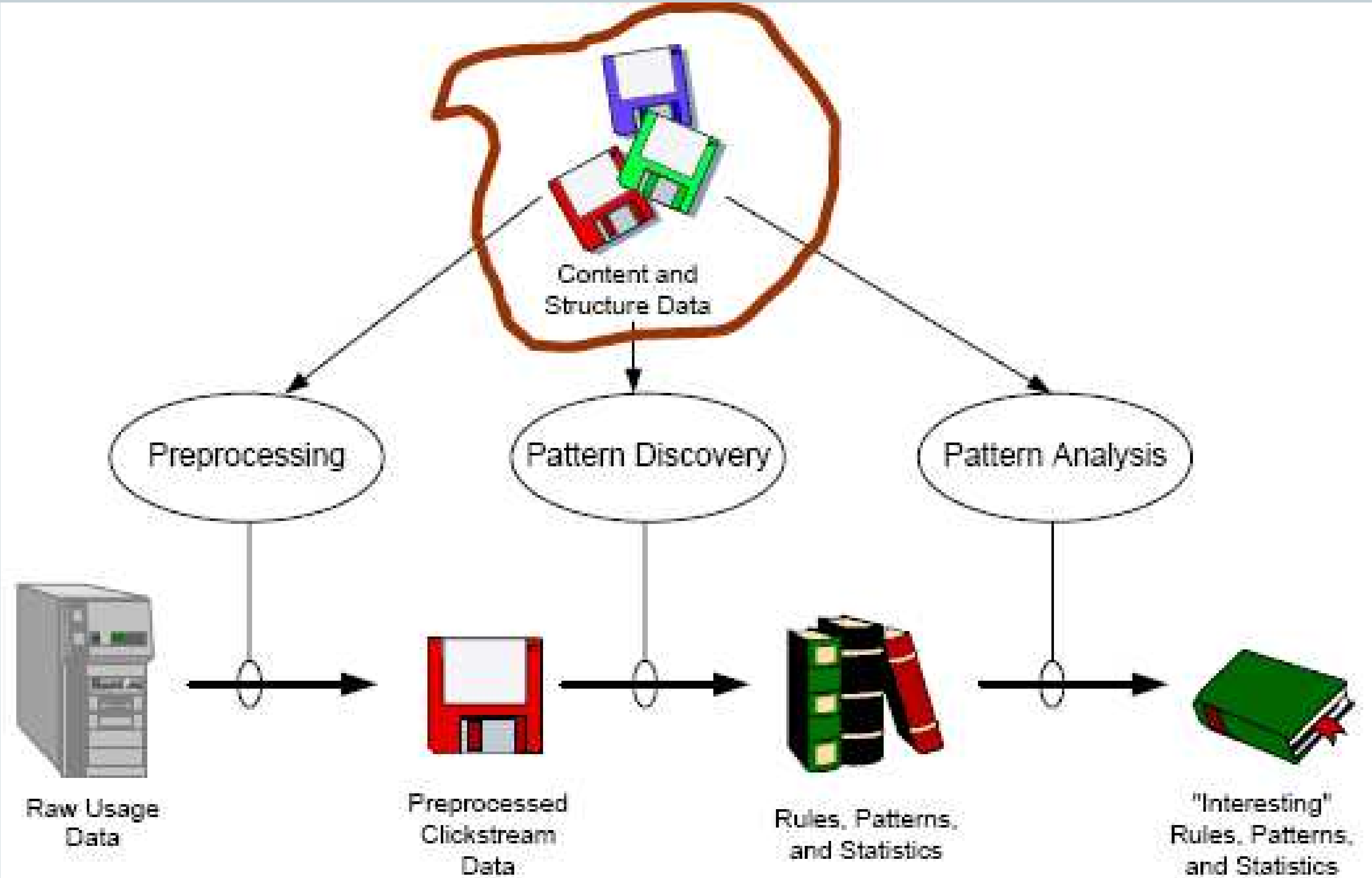
Product-Oriented Events

- Product View
  - Occurs every time a product is displayed on a page view
  - Typical Types: Image, Link, Text
- Product Click-through
  - Occurs every time a user "clicks" on a product to get more information

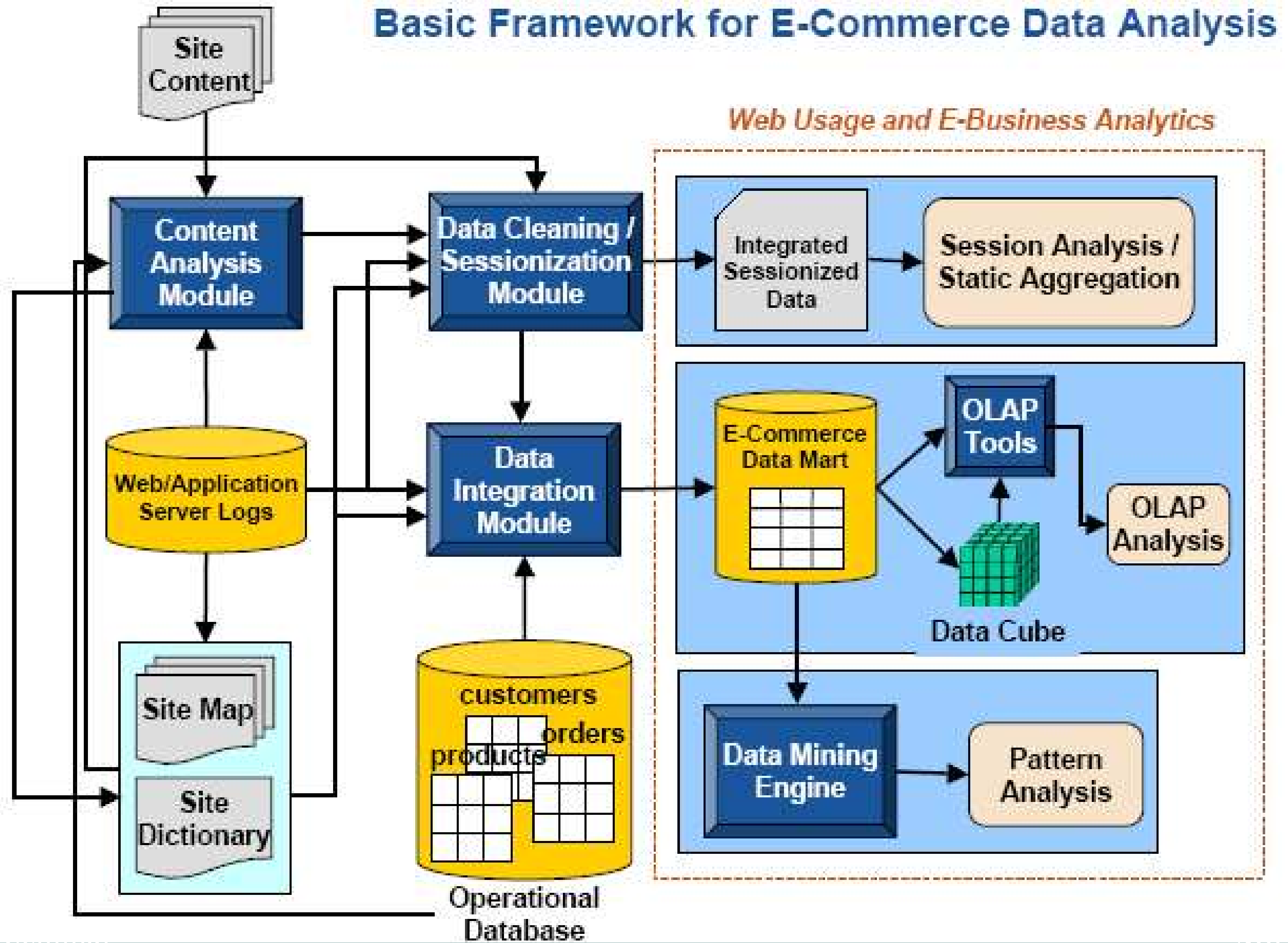
Product-Oriented Events

- Shopping Cart Changes
  - Shopping Cart Add or Remove
  - Shopping Cart Change - quantity or other feature (e.g. size) is changed
- Product Buy or Bid
  - Separate buy event occurs for each product in the shopping cart
  - Auction sites can track bid events in addition to the product purchases

Web usage mining process



E-commerce data analysis



Data mining

Frequent Itemsets

- The “Home Page” and “Shopping Cart Page” are accessed together in 20% of the sessions.
- The “Donkey Kong Video Game” and “Stainless Steel Flatware Set” product pages are accessed together in 1.2% of the sessions.

Association Rules

- When the “Shopping Cart Page” is accessed in a session, “Home Page” is also accessed 90% of the time.
- When the “Stainless Steel Flatware Set” product page is accessed in a session, the “Donkey Kong Video” page is also accessed 5% of the time.

Sequential Patterns

- add an extra dimension to frequent itemsets and association rules - time
- “x% of the time, when A appears in a transaction, B appears within z transactions.”
- Example:The “Video Game Caddy” page view is accessed after the “Donkey Kong Video Game” page view 50% of the time. This occurs in 1% of the sessions.

Data mining (cont.)

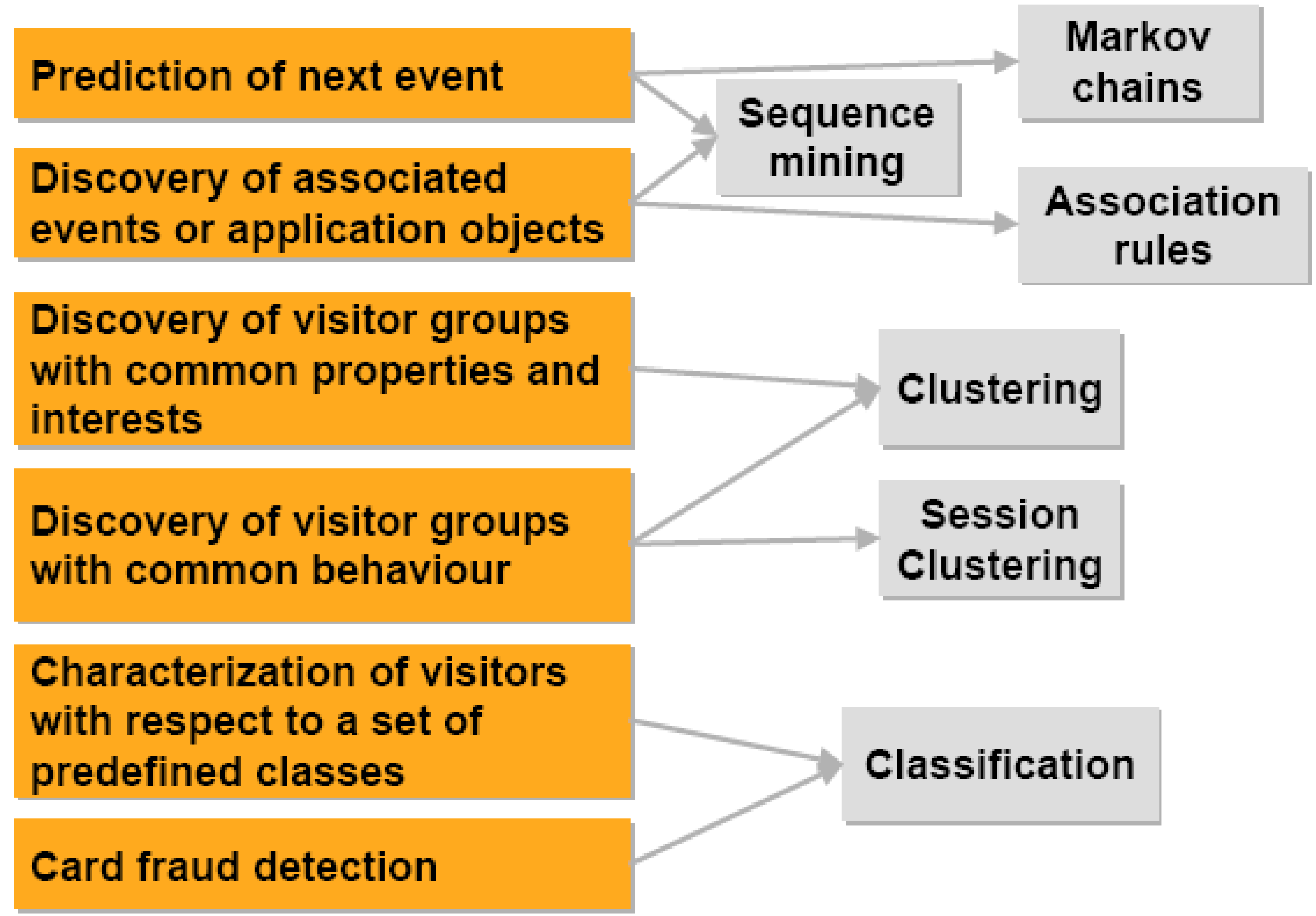
Clustering: Content-Based or Usage-Based

- Customer/visitor segmentation
- Categorization of pages and products

Classification

- "Donkey Kong Video Game", "Pokemon Video Game", and "Video Game Caddy" product pages are all part of the Video Games product group.
- customers who access Video Game Product pages, have income of 50K+, and have 1 or more children, should be get a banner ad for Xbox in their next visit.

Some usage mining applications



Important application - Personalization

Web Personalization: "personalizing the browsing experience of a user by dynamically tailoring the look, feel, and content of a Web site to the user's needs and interests."

Why Personalize?

- broaden and deepen customer relationships
- provide continuous relationship marketing to build customer loyalty
- help automate the process of proactively market products to customers
  - lights-out marketing
  - cross-sell/up-sell products
- provide the ability to measure customer behavior and track how well customers are responding to marketing efforts

Standard approaches

Rule-based filtering

- provide content to users based on predefined rules (e.g., "if user has clicked on A and the user's zip code is 90210, then add a link to C")

Collaborative filtering

- give recommendations to a user based on responses/ratings of other "similar" users

Content-based filtering

- track which pages the user visits and recommend other pages with similar content

Hybrid Methods

- usually a combination of content-based and collaborative

Summary

- Web usage mining has emerged as the essential tool for realizing more personalized, user-friendly and business-optimal Web services.
- The key is to use the user-clickstream data for many mining purposes.
- Traditionally, Web usage mining is used by e-commerce sites to organize their sites and to increase profits.
- It is now also used by search engines to improve search quality and to evaluate search results, etc, and by many other applications.

Data Mining of User Navigation Patterns

- Given set of pages user visited so far, what page he will visit next?
  - Customizing and adapting site's interface for individual user
  - Improving site's static structure
  - Building better navigation system (related links etc)

- José Borges, Mark Levene: Data Mining of User Navigation Patterns. WEBKDD 1999: 92-111

How to analyze

- User navigation data is stored in web server logs
  - Automatically generated, thus very good target for automatic analyze
- Two main approaches
  - Log data is mapped into relation tables,
    - Standard data mining techniques are used(etc association rules)
  - Direct mining of web logs

HPG approach

- User navigation session – sequence of page requests that no two consequent requests separated by more then X minutes
- Model user navigation records as hypertext probabilistic grammar (HPG)
  - String that correspond to user's proffered trails generated with higher probability
- HPG is probabilistic regular grammar which have one-to-one mapping between set of non-terminal symbols and the set of terminal symbols.
  - Non-terminal symbol – web page
  - Production rule – link between pages
  - Two more states: S,F – start and end states

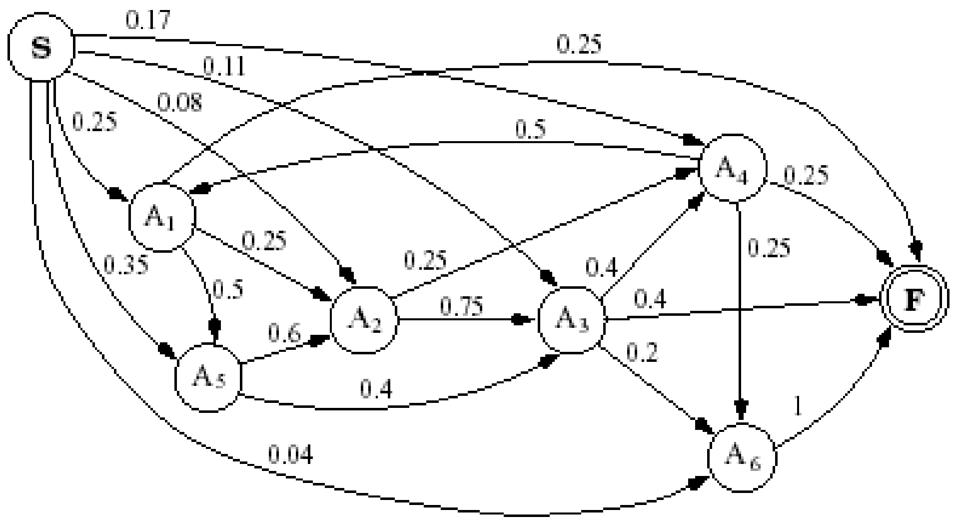
HPG

- ά – parameter that attaches desired weight to state being first in user navigation sequence
  - ά = 0, only states that where first in the session can appear in production from start state
- Probability of production from start state
  - Π(n)= ά * (prob. that n was visited + prob that n was visited first)
- Probability of production from start state is proportional to the number of times corresponding state was visited

Example

- $\Pi(A1) = 0.5(4/24 + 2/6) = 0.25$

ID	Trail
1	$A_1 \to A_2 \to A_3 \to A_4$
2	$A_1 \to A_5 \to A_3 \to A_4 \to A_1$
3	$A_5 \to A_2 \to A_4 \to A_6$
4	$A_5 \to A_2 \to A_3$
5	$A_5 \to A_2 \to A_3 \to A_6$
6	$A_4 \to A_1 \to A_5 \to A_3$



HPG

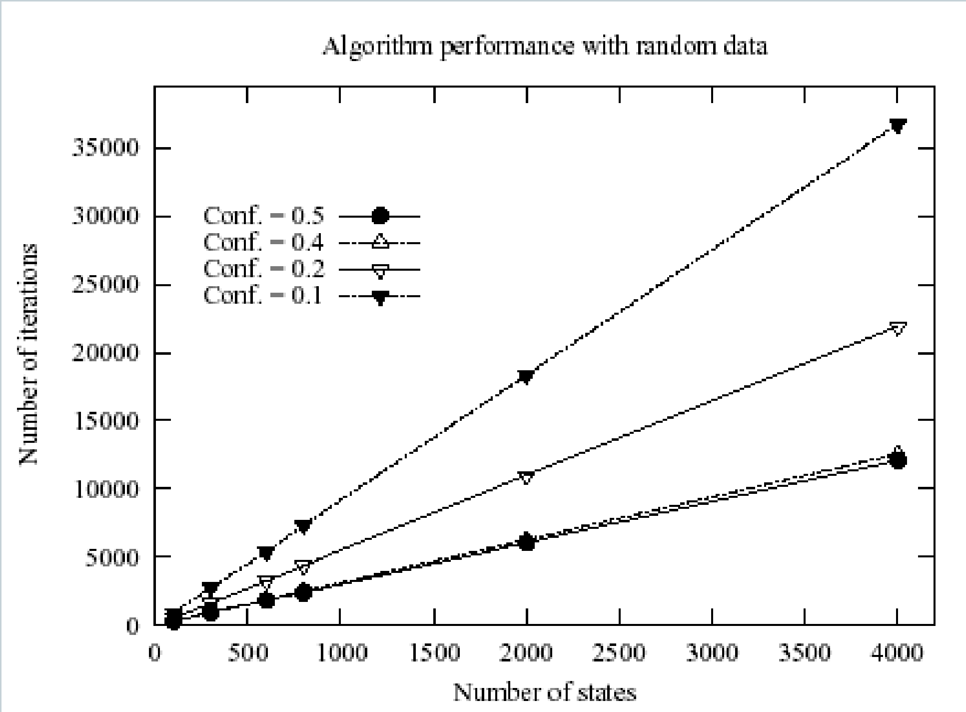
- Probability of first derivation step is defined as *support* (θ)
- Probability of production from start state used to prune strings that might have high probability but belong to rarely visited part of web site
- String is included in grammar language if it's derivation probability is above confidence threshold - λ
- N-grammar – N previously visited pages influence the next choice
  - User have limited memory and remember only N prev pages.

Experiments – Random data

- To evaluate algorithm performance and scalability
- Configurations
  - 100 < N (number of pages) <4000
  - 0.1 < confidence <0.5
  - Support = 1/n
  - For each configuration 150 runs were performed

Experiments – Random data

- For a given confidence number of iteration is linear with grammar size
- CPU follows similar trend

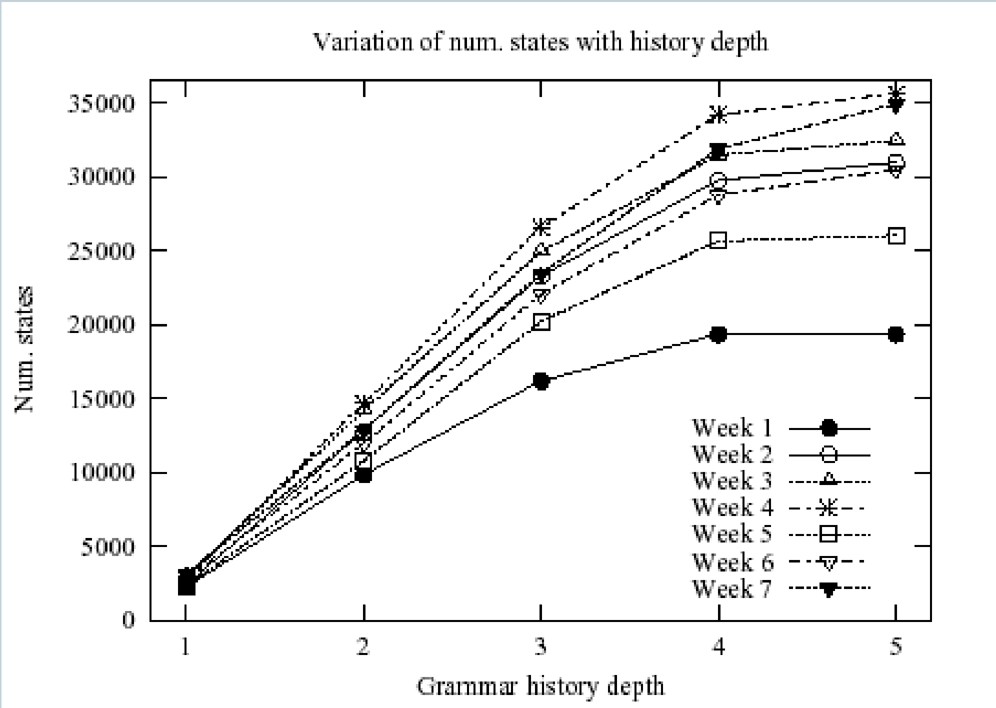


Experiments – real data

- Real data contained two month of usage from site www.hyperreal.org/music/machines
- Each month was divided into 4 subsets, each corresponding for a week
- For each subset corresponding HPG for several values of history depth was build

Experiments – real data

- Size of N-grammar model increases slower then worst case, stabilizing for history values of order 5, probably due to sparseness of data
- Performance showed results similar for those of random data



Web Structure Mining
Link Analysis Algorithms

PAGE RANK