



IIT-H

Web Mining

Lecture 17: Mining Structured Information from the Web (Part 1)

Manish Gupta

30th Sep 2013

Slides borrowed (and modified) from

<http://www.mathcs.emory.edu/~eugene/kdd-webinar/kdd-webinar-ie-March-22-2007.ppt>

<http://www.cs.uic.edu/~liub/teach/cs583-fall-11/CS583-structured-data-extraction.ppt>

<http://akbcwekex2012.files.wordpress.com/2012/06/slides-nilesh.pptx>

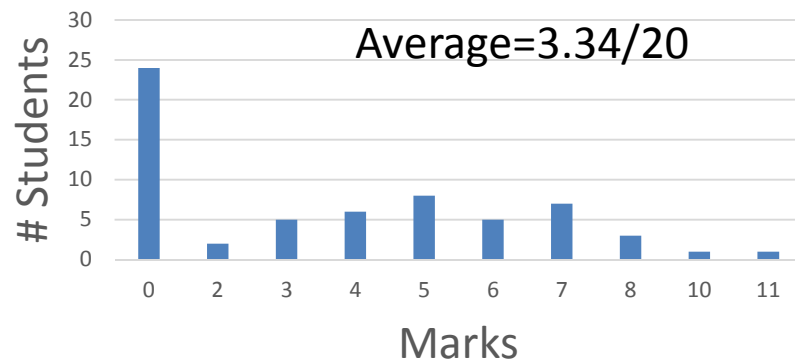
Recap of Lecture 16: Introduction to Computational Advertising (Part 2): Contextual Advertising

- Contextual Advertising Basics
- Ad Selection in Contextual Advertising
- Phrase Extraction for Contextual Advertising
- IR Methods for Content Match Ad Retrieval
- ~~Holistic View at the Page in Contextual Advertising~~
- ~~When (Not) to Advertise~~
- ~~Search-based Ad Selection for Sponsored Search~~
- ~~Predicting Clicks~~

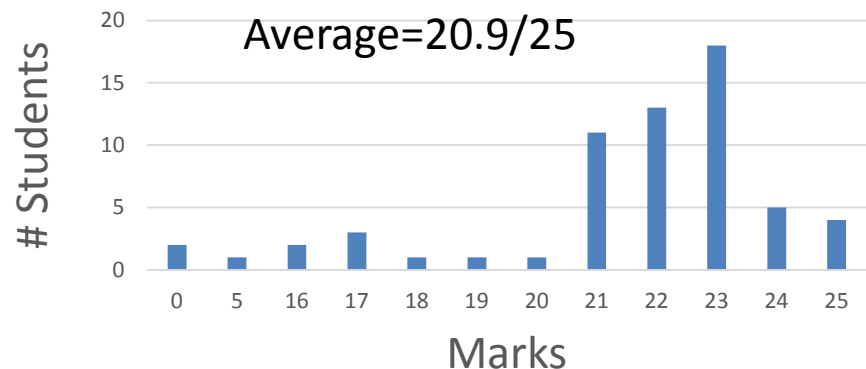
Announcements

- We will have a lecture tomorrow from 8:30am to 10am because Oct 2 is a public holiday

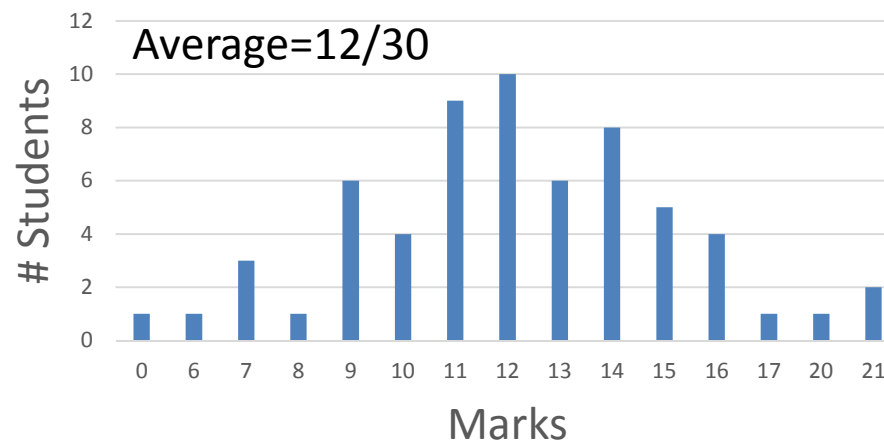
Surprise Quiz3 Marks Distribution



Assignment 1 Marks Distribution



Midsem Marks Distribution



Today's Agenda

- Introduction to Information Extraction
- Wrapper Induction
- List Extraction using Automatic Wrapper Generation
- Information Extraction for Unstructured Data

Today's Agenda

- **Introduction to Information Extraction**
- Wrapper Induction
- List Extraction using Automatic Wrapper Generation
- Information Extraction for Unstructured Data

Databases vs The Web

dbo.Sentinel
Columns
RecordID (float, null)
UserReferenceID (float, null)
TransactionInDate (nvarchar(255), null)
TransactionOutDate (nvarchar(255), null)
MinutesIn (float, null)
RWPNumber (float, null)
RWPRevision (float, null)
RWPTitle (nvarchar(255), null)
Task (float, null)
TaskDescription (nvarchar(255), null)
EDDose (float, null)
DepartmentCode (nvarchar(255), null)
DepartmentName (nvarchar(255), null)
CraftCode (nvarchar(255), null)
CraftName (nvarchar(255), null)
DoseAlarm (float, null)
RateAlarm (float, null)
WorkOrderTask (nvarchar(255), null)
WorkOrderTaskDescription (nvarchar(255), null)

DB Schema



The screenshot shows the homepage of the Sixth ACM International Conference on Web Search and Data Mining (WSDM) in Rome 2013. The header features the WSDM logo and the conference title. A navigation bar includes links for Home, About, Authors, Attendees, Program, and Sponsors. The main content area is divided into several sections: a search bar, quick links (Registration, Conference program, Co-located events Feb 4-5, Hotel information, Travel information, Become a sponsor), key dates (Conference, Workshops, Tutorials), and a list of news items. On the right, there is a section for keynote speakers, featuring a portrait of Qiang Yang. At the bottom, there are logos for Platinum sponsors (Microsoft Research, Google, Yahoo! Research).

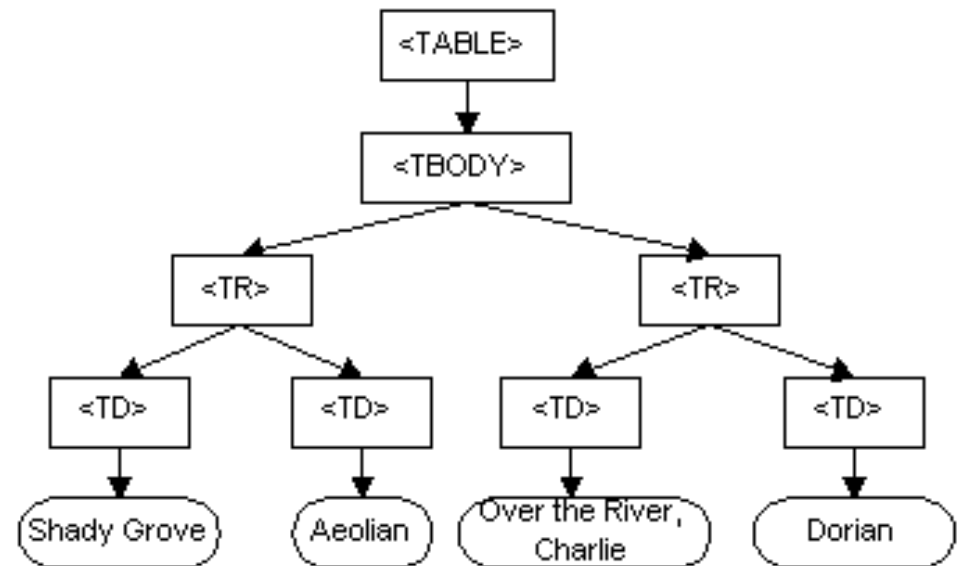
HTML is Schema-less

Document Object Model (DOM)

HTML -> DOM

- › DOM is a tree model of the HT markup language

```
<TABLE>  
<TBODY>  
<TR>  
<TD>Shady Grove</TD>  
<TD>Aeolian</TD>  
</TR>  
<TR>  
<TD>Over the River, Charlie</TD>  
<TD>Dorian</TD>  
</TR>  
</TBODY>  
</TABLE>
```

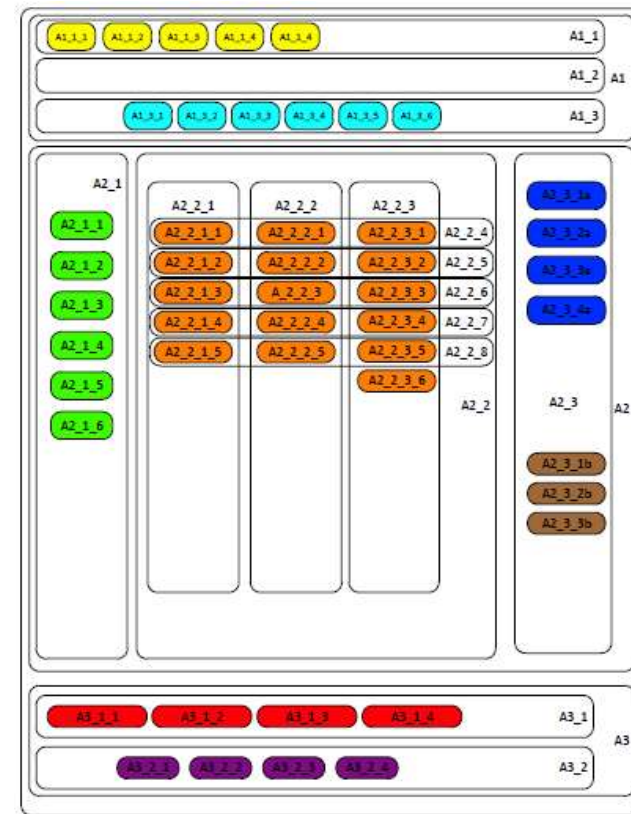


Web Page Rendering

HTML -> DOM -> WebPage

› Web page rendering according to Web standards

Uses the Boxes Model



Goal: Extract Information from the Web

Text paragraphs without
formatting (unstructured)

Astro Teller is the CEO and co-founder of BodyMedia. Astro holds a Ph.D. in Artificial Intelligence from Carnegie Mellon University, where he was inducted as a national Hertz fellow. His M.S. in symbolic and heuristic computation and B.S. in computer science are from Stanford University. His work in science, literature and business has appeared in international media from the New York Times to CNN to NPR.











Text paragraphs with some
formatting & links (unstructured)

Dr. Steven Minton - Founder/CTO
Dr. Minton is a fellow of the American Association of Artificial Intelligence and was the founder of the Journal of Artificial Intelligence Research. Prior to founding Fetch, Minton was a faculty member at USC and a project leader at USC's Information Sciences Institute. A graduate of Yale University and Carnegie Mellon University, Minton has been a Principal Investigator at NASA Ames and taught at Stanford, UC Berkeley and USC.

Frank Huybrechts - COO
Mr. Huybrechts has over 20 years of

- Press
- **Contact**
- General information
- Directions maps

Missing grammar, but rich formatting
& links (semi-structured)

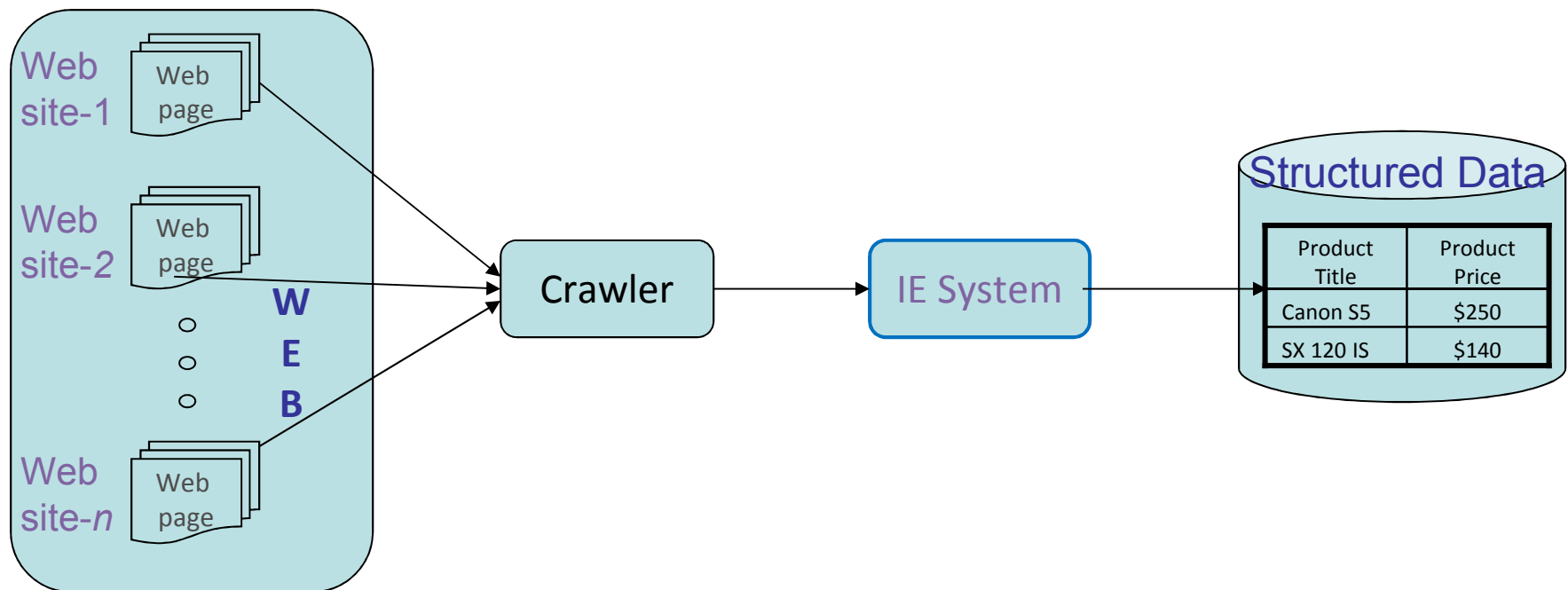
Barto, Andrew G.	(413) 545-2109	barto@cs.umass.edu	CS276
Professor. Computational neuroscience, reinforcement learning, adaptive motor control, artificial neural networks, adaptive and learning control, motor development.			 
Berger, Emery D.	(413) 577-4211	emery@cs.umass.edu	CS344
Assistant Professor.			 
Brock, Oliver	(413) 577-0334	oli@cs.umass.edu	CS246
Assistant Professor.			 
Clarke, Lori A.	(413) 545-1328	clarke@cs.umass.edu	CS304
Professor. Software verification, testing, and analysis; software architecture and design.			 
Cohen, Paul R.	(413) 545-3638	cohen@cs.umass.edu	CS278
Professor. Planning, simulation, natural language, agent-based systems, intelligent data analysis, intelligent user interfaces.			 

Tables (almost structured)

8:30 - 9:30 AM	Invited Talk: Plausibility Measures: A General Approach for Representing Uncertainty <i>Joseph Y. Halpern, Cornell University</i>				
9:30 - 10:00 AM	Coffee Break				
10:00 - 11:30 AM	Technical Paper Sessions:				
Cognitive Robotics	Logic Programming	Natural Language Generation	Complexity Analysis	Neural Networks	Games
739: A Logical Account of Causal and Topological Maps <i>Emilio Remolina and Benjamin Kuipers</i>	116: A-System: Problem Solving through Abduction <i>Marc Denecker, Antonis Kakas, and Bert Van Nuffelen</i>	758: Title Generation for Machine-Translated Documents <i>Rong Jin and Alexander G. Hauptmann</i>	417: Let's go Nats: Complexity of Nested Circumscription and Abnormality Theories <i>Marco Cadoli, Thomas Eiter, and Georg Gottlob</i>	179: Knowledge Extraction and Comparison from Local Function Networks <i>Kenneth McGarry, Stefan Wermter, and John MacIntyre</i>	71: Iterative Widening <i>Tristan Cazenave</i>
549: Online-Execution of ccGolog Plans <i>Henrik Grosskreutz and Gerhard Lakemeyer</i>	131: A Comparative Study of Logic Programs with Preference <i>Torsten Schaub and Kewen</i>	246: Dealing with Dependencies between Content Planning and Surface Realisation in a Pipeline Generation	470: A Perspective on Knowledge Compilation <i>Adnan Darwiche and Pierre Marquis</i>	258: Violation-Guided Learning for Constrained Formulations in Neural-Network Time-Series	353: Temporal Difference Learning Applied to a High Performance Game-Playing

Information Extraction

- Information Extraction is extracting **useful, structured information (data, knowledge)** from **semi-structured or unstructured documents**



Complexity of Mined Data

Closed set

Geographical location, gene names, etc.

He was born in Alabama...

The big Wyoming sky...

Regular set (Syntactic)

Phone numbers, Zip codes, etc.

Phone: (413) 545-1323

The CALD main office can be reached at 412-268-1299

Complex pattern

Postal Address, etc.

University of Arkansas
P.O. Box 140
Hope, AR 71802

Headquarters:
1128 Main Street, 4th Floor
Cincinnati, Ohio 45210

Ambiguous patterns, needing context & sources of evidence

Person names, company names, etc.

...was among the six houses sold by Hope Feldman that year.

1. The Big Web Show
URL: [http://www.bignetwork.com](#)
Hosted by: Jeffrey Zeldman and Dan Benjamin
Recorded in: New York City and Austin, Texas
Running since: April 29, 2010; 58 Episodes
Format: Weekly, live, audio, sometimes video, about an hour
Subjects covered: The Big Web Show features special guests and topics like web publishing, art direction, content strategy, typography, web technology, and more. It's everything web that matters

2. Boagworld
URL: [http://boagworld.com/seasons/](#)
Hosted by: Paul Boag and Marcus Lillington
Recorded in: Our beautiful barn in the heart of rural Hampshire
Running since: August 2005; 230 episodes
Format: Weekly, audio with occasional live video shows, Approximately 1hr
Subjects covered: Anything of interest to web designers, developers or website owners

3. Creative Coding
URL: [http://creativecodingpodcast.com](#)
Hosted by: Seb Lee-Delisle and Iain Lobb
Recorded in: Brighton, Truro and wherever international location Seb happens to be in that week
Running since: January 2011; 13 episodes
Format: Every two or three weeks, audio only, between 30 and 60 minutes

Enchanted Learning.com

The Presidents of the United States of America

[In the order in which they served](#) [Alphabetical order](#) [Short table of Data](#)

The President and Vice-President are elected every four years. They must be at least 35 years of age, they must be native-born citizens of the United States, and they must have been residents of the U.S. for at least 14 years. (Also, a person cannot be elected to a third term as President.)

President	Party	Term as President	Vice-President
1. George Washington (1732-1799)	None, Federalist	1789-1797	John Adams
2. John Adams (1735-1826)	Federalist	1797-1801	Thomas Jefferson
3. Thomas Jefferson (1743-1826)	Democratic-Republican	1801-1809	Aaron Burr, George Clinton
4. James Madison (1751-1836)	Democratic-Republican	1809-1817	George Clinton, Elbridge Gerry
5. James Monroe (1758-1831)	Democratic-Republican	1817-1825	Daniel Tompkins
6. John Quincy Adams (1767-1848)	Democratic-Republican	1825-1829	John Calhoun
7. Andrew Jackson (1767-1845)	Democrat	1829-1837	John Calhoun, Martin van Buren
8. Martin van Buren (1782-1862)	Democrat	1837-1841	Richard Johnson
9. William H. Harrison (1773-1841)	Whig	1841	John Tyler
10. John Tyler (1790-1862)	Whig	1841-1845	
11. James K. Polk (1795-1849)	Democrat	1845-1849	George Dallas
12. Zachary Taylor (1784-1850)	Whig	1849-1850	Millard Fillmore
13. Millard Fillmore (1800-1874)	Whig	1850-1853	
14. Franklin Pierce (1804-1869)	Democrat	1853-1857	William King
15. James Buchanan (1791-1868)	Democrat	1857-1861	John Breckinridge

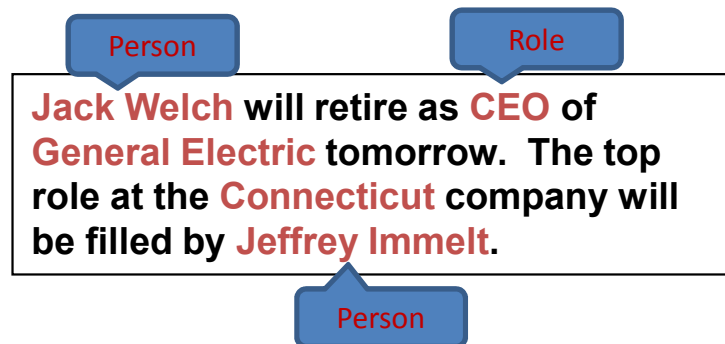
India, China, Canada
China, Canada, France
Delhi, Beijing, Ottawa
Beijing, Ottawa, Paris
Canada, England, France
London, Ottawa, Paris

Lists

Tables

Sets

What Structure to Extract?



Binary relationship

Relation: Person-Title
Person: Jack Welch
Title: CEO

N-ary record / Entity

Relation: Succession
Company: General Electric
Title: CEO
Out: Jack Welch
In: Jeffrey Immelt

Single Entity

★★★★★ (30 ratings)
Dance FX Studios
 (480) 968-6177
 1859 W Guadalupe Rd, #105, Mesa, AZ 85202 [Get directions](#)
 Cross Streets: Near the intersection of W Guadalupe Rd and S Pennington
 Neighborhood: Dobson Ranch Mesa, AZ
www.dancefxstudios.com



Business Name: Dance FX Studios
NumOfRatings: 30
Phone #: (480) 968-6177
Address: 1859 W... AZ 85202
Website: www.dancefxstudios.com
Photo: <http://a323.yahoofs.com/localcontent... D6GmvPrGP>

Multi-Entity

Microsoft Windows 7 Anytime Upgrade [Home Premium to Professional] by Microsoft Software (Oct. 22, 2009) - No Operating System

Buy new: ~~\$89.99~~ **\$79.99**
13 new from \$79.99 *1 used from \$72.95*
 Get it by **Tuesday, Aug. 31** if you order in the next **6 hours** and choose one-day shipping.
 Eligible for **FREE** Super Saver Shipping.
 ★★★★★ (129)
Software: See all 8,712 items

Windows 7 Home Premium 64 Bit System Builder 1pk by Microsoft Software (DVD-ROM - Oct. 22, 2009) - Windows 7

Buy new: ~~\$499.00~~ **\$99.99**
11 new from \$99.99
 Get it by **Tuesday, Aug. 31** if you order in the next **6 hours** and choose one-day shipping.
 Eligible for **FREE** Super Saver Shipping.
 ★★★★★ (146)
Software: See all 8,712 items

Microsoft Windows 7 Professional Upgrade by Microsoft Software (DVD-ROM - Oct. 22, 2009) - Windows 7

Buy new: ~~\$499.99~~ **\$169.39**
22 new from \$162.71 *2 used from \$119.88*
 In stock on September 2, 2010
 Eligible for **FREE** Super Saver Shipping.
 ★★★★★ (144)
Software: See all 8,712 items

Product Name: Microsoft Windows 7 Anytime Upgrade
Price: \$79.99
NumOfRatings: 55

Product Name: Windows 7 Home Premium 64 ... 1pk
Price: \$99.99
NumOfRatings: 16

Product Name: Microsoft Windows 7 ... Upgrade
Price: \$162.70
NumOfRatings: 144

Information Extraction Key Goals

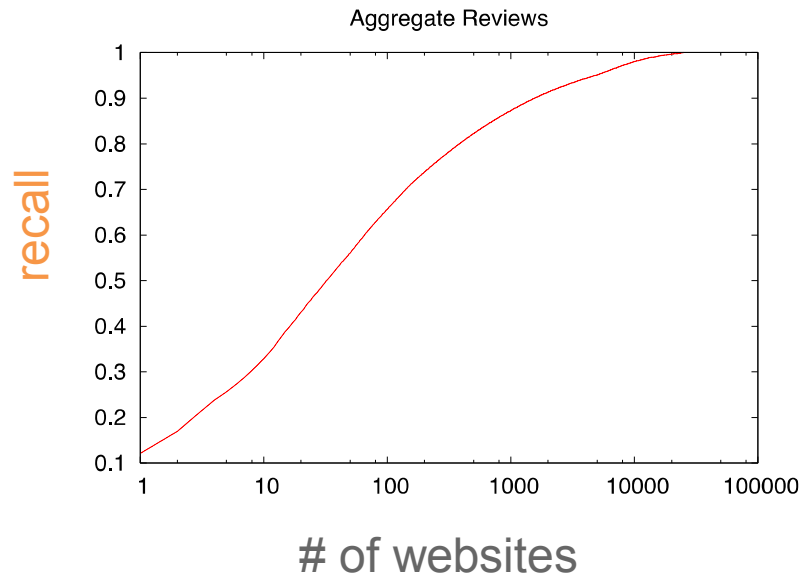
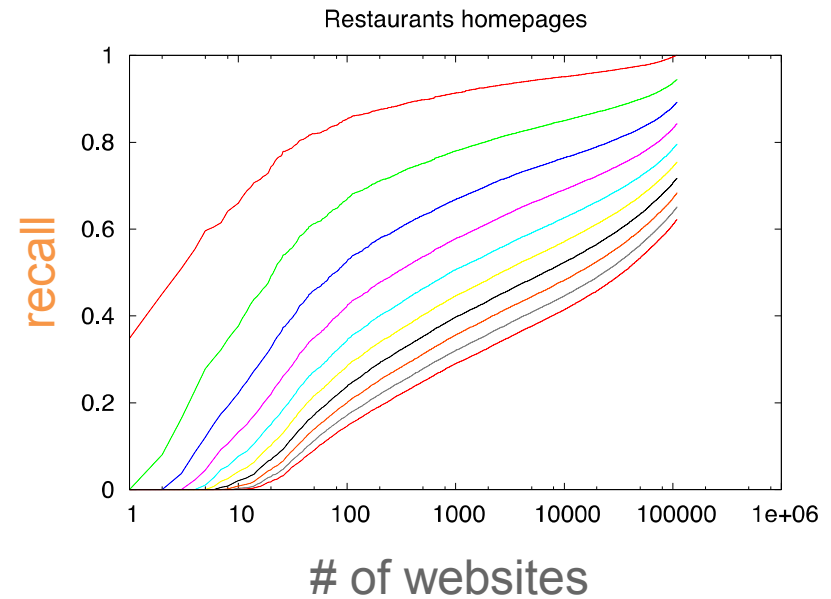
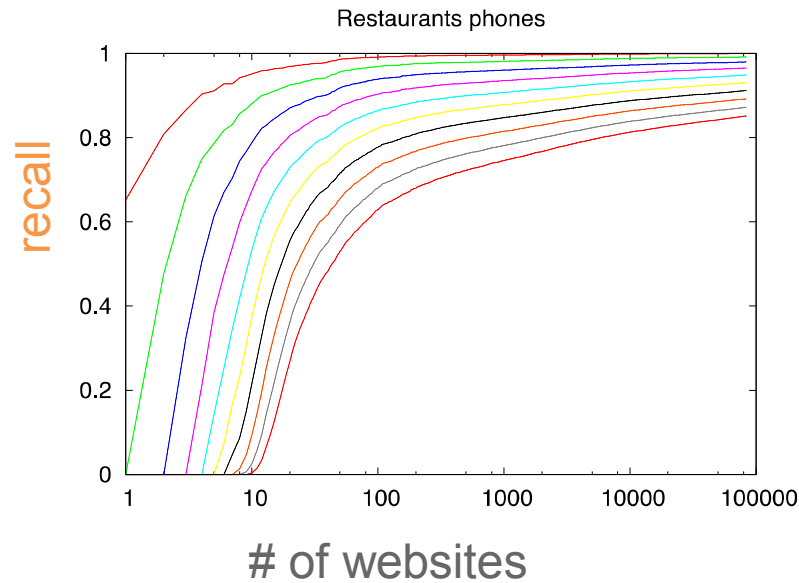
- Scalability
 - Billions of pages
 - Performance: Extraction run-time, memory
 - Limited human bandwidth
- Data Quality
 - High quality requirement (typically > 95+% precision)
- Data Coverage
 - Should cover all types of “variations” in pages representing same entity

Today's Agenda

- Introduction to Information Extraction
- **Wrapper Induction**
- List Extraction using Automatic Wrapper Generation
- Information Extraction for Unstructured Data

Why Learn Wrappers?

Problem: Given a schema, populate it with by extracting information from the web



Even for domains with well- established aggregator sites, we need to go to the long tail of websites to build a reasonably complete database.

Main idea : Use content redundancy across websites and structural coherency within websites

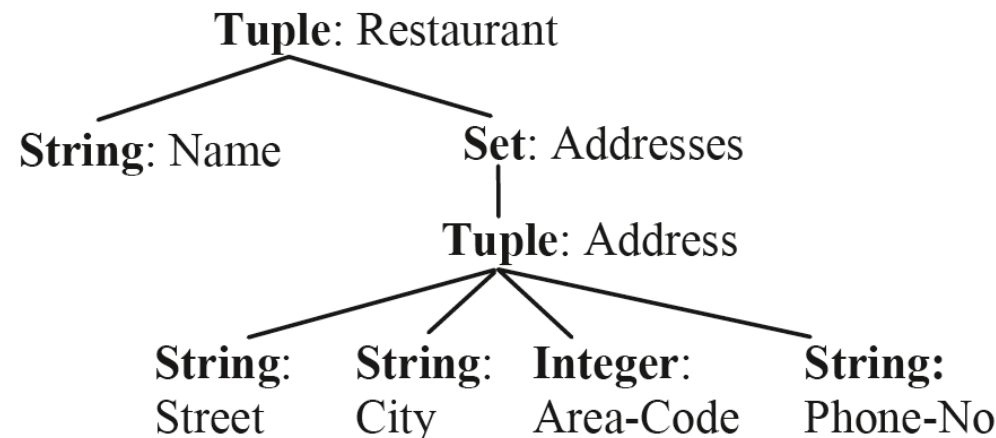
Wrapper Induction

- Using machine learning to generate extraction rules.
 - 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 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 item is extracted independent of others.
- 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 Representation: Type Tree

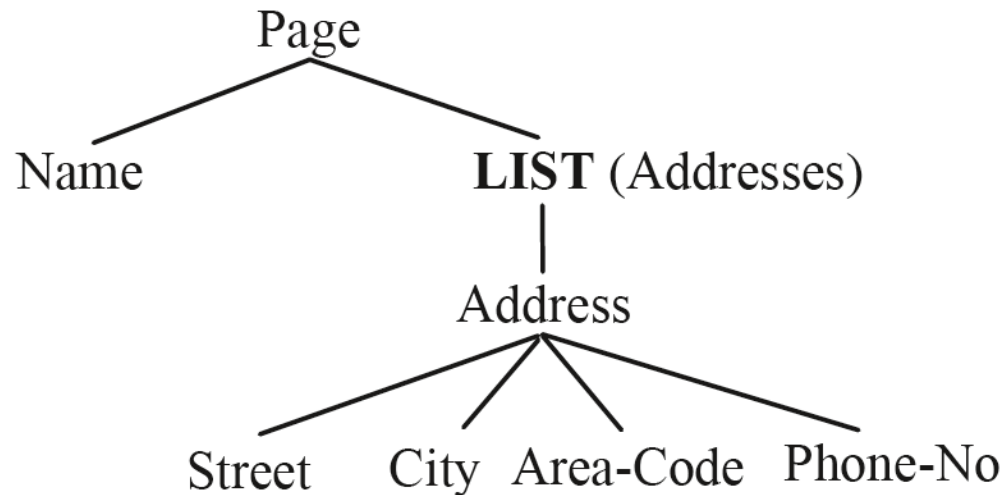
Restaurant Name: **Good Noodles**

- 205 Willow, *Glen*, Phone 1-773-366-1987
- 25 Oak, *Forest*, Phone (800) 234-7903
- 324 Halsted St., *Chicago*, Phone 1-800-996-5023
- 700 Lake St., *Oak Park*, Phone: (708) 798-0008



Data Extraction based on EC Tree

- The extraction is done using a tree structure called the *EC* tree (**embedded catalog tree**).
- The *EC* tree is based on the type tree above.



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

Extraction using Two Rules

- Each extraction is done using two rules,
 - **a start rule** and **a end rule**.
- The start rule identifies the beginning of the node and the end rule identifies the end of the node.
 - This strategy is applicable to both leaf nodes (which represent data items) and list nodes.
- For a list node, **list iteration rules** are needed to break the list into individual data records (tuple instances).

Extraction Rules

- Rules use landmarks (sequence of consecutive tokens)
 - Landmarks are used to locate the beginning and the end of a target item.
 - To extract the restaurant name “Good Noodles”, a possible start rule is **R1**: *SkipTo()*
 - is a landmark
 - End-rule could be **R2**: *SkipTo()*
- Rules are not unique; need to use disjunctions to handle variants for the same data type

Learning Extraction Rules (1)

- Stalker uses sequential covering to learn extraction rules for each target item.
 - In each iteration, it learns a perfect disjunctive 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.
- Best disjunct prefers candidates that have
 - More correct matches
 - Accepts fewer false positives
 - Fewer wildcards
 - Longer end-landmarks

Learning Extraction Rules (2)

E1: 205 Willow, <i>Glen</i>, Phone 1-<i>773</i>-366-1987
E2: 25 Oak, <i>Forest</i>, Phone (800) 234-7903
E3: 324 Halsted St., <i>Chicago</i>, Phone 1-<i>800</i>-996-5023
E4: 700 Lake St., <i>Oak Park</i>, Phone: (708) 798-0008

Function LearnDisjunct(*Examples*)

```
1  let Seed ∈ Examples be the shortest example;  
2  Candidates ← GetInitialCandidates(Seed);  
3  while Candidates ≠ ∅ do  
4    D ← BestDisjunct(Candidates);  
5    if D is a perfect disjunct then  
6      return D  
7    Candidates ← Candidates ∪ Refine(D, Seed);  
8    remove D from Candidates;  
9  return D
```

We used only the *SkipTo*() function in extraction rules. *SkipUntil*() etc. can also be used

- Disjunction refinements
 - Landmark refinement:
Increase the size of a landmark by concatenating a terminal
 - Refine *SkipTo*(<i>) to *SkipTo*(-<i>)
 - Topology refinement:
Increase the number of landmarks by adding 1-terminal landmarks, i.e., *t* and its matching wildcards
 - Refine *SkipTo*(<i>) to *SkipTo*(205)*SkipTo*(-<i>)

Active Learning for Informative Examples

- Wrapper learning needs manual labeling of training examples.
- Manual labeling is expensive
- Examples of the same formatting are of limited use
- Active learning
 1. Randomly select a small subset L of unlabeled examples from U
 2. Manually label the examples in L , and $U = U - L$
 3. Learn a wrapper W based on the labeled set L
 4. Apply W to U to find a set of informative examples L
 5. Stop if L is empty, otherwise go to step 2
- Informative examples: Co-testing
 - To extract an item, learn both backward and forward rules
 - If the two rules disagree on the beginning of a target item in the example, this example is given to the user to label.

Wrapper Maintenance

- Wrapper verification: If the site changes, does the wrapper know the change?
- Wrapper repair: If the change is correctly detected, how to automatically repair the wrapper?
- One way to deal with both problems is to learn the characteristic patterns of the target items.
- These patterns are then used to monitor the extraction to check whether the extracted items are correct.
- Re-labeling: If they are incorrect, the same patterns can be used to locate the correct items assuming that the page changes are minor formatting changes.
- Re-learning: re-learning produces a new wrapper.
- Difficult problems: These two tasks are extremely difficult because it often needs contextual and semantic information to detect changes and to find the new locations of the target items.

Today's Agenda

- Introduction to Information Extraction
- Wrapper Induction
- **List Extraction using Automatic Wrapper Generation**
- Information Extraction for Unstructured Data








Why Automate?

- Wrapper induction (supervised) has two main shortcomings:
 - It is unsuitable for a large number of sites due to the manual labeling effort.
 - Wrapper maintenance is very costly. The Web is a dynamic environment. Sites change constantly. Since rules learnt by wrapper induction systems mainly use formatting tags, if a site changes its formatting templates, existing extraction rules for the site become invalid.
- Automatic extraction is possible because data records (tuple instances) in a Web site are usually encoded using a very small number of fixed templates.
- It is possible to find these templates by mining repeated patterns.

Two Data Extraction Problems

- The general problem of data extraction is to recover the hidden schema from the HTML mark-up encoded data.
- Problem 1: Extraction given a single list page
 - **Input:** A single HTML string S , which contain k non-overlapping substrings s_1, s_2, \dots, s_k with each s_i encoding an instance of a set type. That is, each s_i contains a collection W_i of $m_i (\geq 2)$ non-overlapping sub-substrings encoding m_i instances of a tuple type.
 - **Output:** k tuple types $\sigma_1, \sigma_2, \dots, \sigma_k$, and k collections C_1, C_2, \dots, C_k of instances of the tuple types such that for each collection C_i there is a HTML encoding function enc_i such that $enc_i: C_i \rightarrow W_i$ is a bijection.
- Problem 2: Data extraction given multiple pages
 - **Input:** A collection W of k HTML strings, which encode k instances of the same type.
 - **Output:** A type σ , and a collection C of instances of type σ , such that there is a HTML encoding enc such that $enc: C \rightarrow W$ is a bijection.

A List Page with Two Data Blocks

TOP SELLERS			
			
<u>C256M/MP3 Digital Media Player, MP3 / WMA / Voice, 256MB by Centon</u>	<u>iPod Video 30GB, Black by Apple</u>	<u>iPod Video 30GB, White by Apple</u>	<u>SIRHK3 Satellite Radio Home Dock by Audiovox</u>
Was: \$49.99 \$29.99 SAVE \$20 ** **Click here for details	\$299.99 Click here for details	\$299.99 Click here for details	\$29.99 Click here for details
Compare » <input type="checkbox"/> «	Compare » <input type="checkbox"/> «	Compare » <input type="checkbox"/> «	Compare » <input type="checkbox"/> «
517 MATCHING PRODUCTS			
Page 1 of 26: 1 2 3 4 5 6 7 8 9 10 Next >> >>>			
Sort by: Popularity ▼ Top Seller Price Product Name Brand			Compare
	<u>iPod Video 30GB, Black by Apple</u> Product Number: 335469 Mfr. Part #: MA146LL/A Brand: Apple iPods & Portable Audio « Visit their Showcase	\$299.99	Add To Cart (Delivery / Pick-Up) Free Shipping Compare » <input type="checkbox"/> «
	<u>iPod Nano 2GB, Black by Apple</u> Product Number: 334248 Mfr. Part #: MA099LL/A Brand: Apple iPods & Portable Audio « Visit their Showcase Limit 1 per customer.	\$199.99	Add To Cart (Delivery / Pick-Up) Free Shipping Compare » <input type="checkbox"/> «
	<u>C256M/MP3 Digital Media Player, MP3 / WMA / Voice, 256MB by Centon</u> Product Number: 331970 Mfr. Part #: 256MP3-001	Was: \$49.99 \$29.99 SAVE \$20 after: \$10.00 instant savings	Add To Cart (Delivery / Pick-Up) Compare » <input type="checkbox"/> «

Using Regular Expressions

- The patterns can be represented using a regular expression. Data extraction can be done using the regular expression
- The key is to find the encoding template from a collection of encoded instances of the same type
- A natural way to do this is to detect repeated patterns from HTML encoding strings
- **String edit distance** and **tree edit distance** are obvious techniques for the task

String and Tree Edit Distances

- String edit distance of two strings, s_1 and s_2 , is defined as the minimum number of point mutations required to change s_1 into s_2 , where a point mutation is one of
 - Change a letter
 - Insert a letter
 - Delete a letter
- Tree edit distance between two trees A and B (labeled ordered rooted trees) is the cost associated with the minimum set of operations needed to transform A into B. The set of operations used to define tree edit distance includes three operations
 - Node removal
 - Node insertion
 - Node replacement

Building DOM Trees

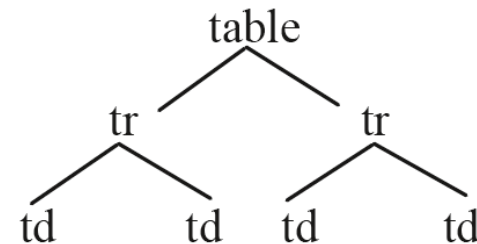
- The usual first step is to build a DOM tree (tag tree) of a HTML page.
 - Most HTML tags work in pairs. Within each corresponding tag-pair, there can be other pairs of tags, resulting in a nested structure.
 - Building a DOM tree from a page using its HTML code is thus natural.
- In the tree, each pair of tags is a **node**, and the nested tags within it are the **children** of the node.
- **HTML code cleaning:**
 - Some tags do not require closing tags (e.g., , <hr> and <p>) although they have closing tags.
 - Additional closing tags need to be inserted to ensure all tags are balanced.
 - Ill-formatted tags need to be fixed. One popular program is called **Tidy**, which can be downloaded from <http://tidy.sourceforge.net/>.

Building DOM Tree using Tags & Visual Cues

- Correcting errors in HTML can be hard.
- There are also dynamically generated pages with scripts.
- Visual information comes to the rescue.
- As long as a browser can render a page correct, a tree can be built correctly.
 - Each HTML element is rendered as a rectangle.
 - Containments of rectangles representing nesting.

```
1 <table>
2   <tr>
3     <td> data1 <td>
4     </td>data2 </td>
5   <tr>
6     <td> data3 </td>
7     <td> data4
8   </tr>
9 </table>
```

data1	data2
data3	Data4



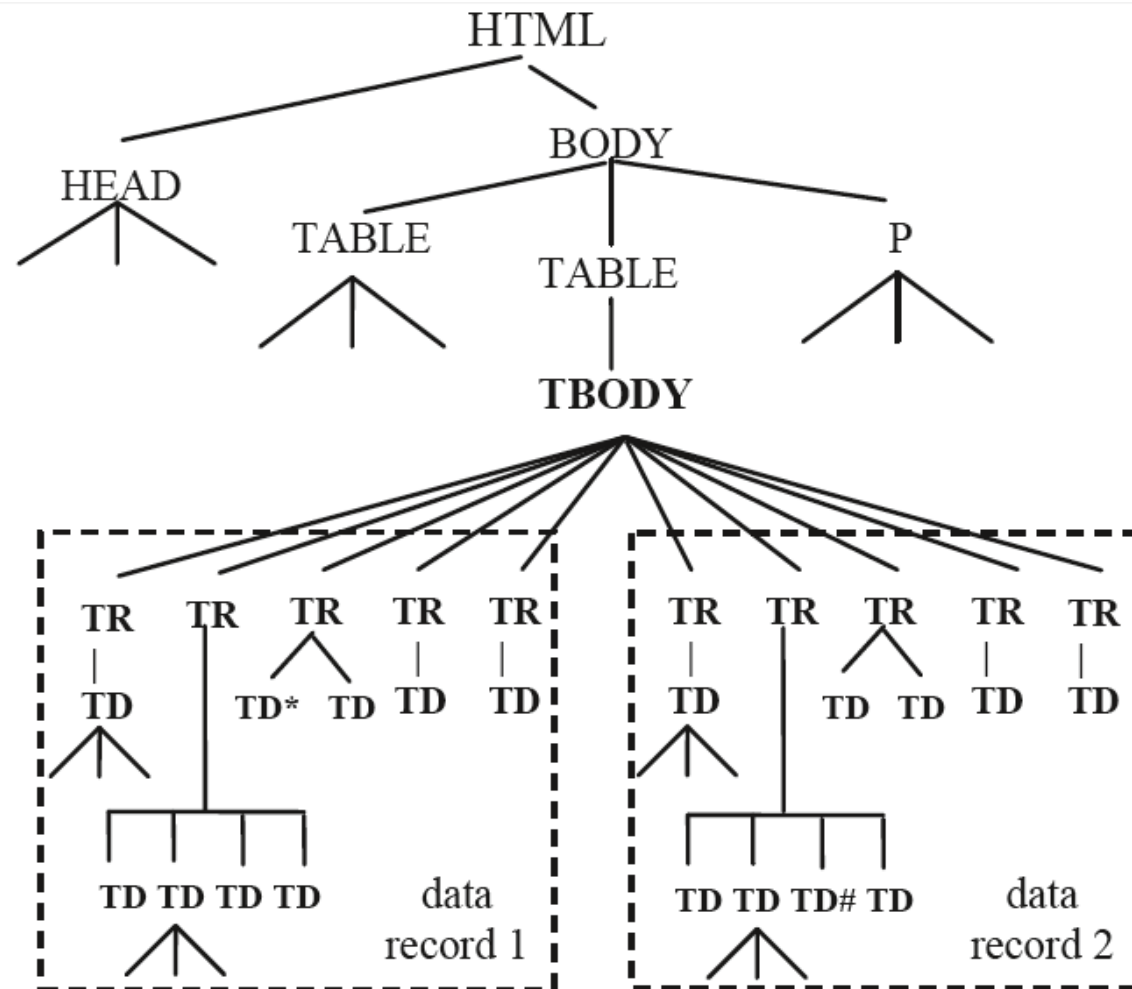
Extraction Given a List Page: Flat Data Records

- Given a single list page with multiple data records
 - Identify each list (or a data region), i.e., mine all data regions
 - Automatically segment data records in each list
 - Align data items in the data records to produce a data table for each list and also a regular expression pattern
- Since the data records are flat (no nested lists), string similarity or tree matching can be used to find similar structures.
 - Computation is a problem
 - A data record can start anywhere and end anywhere

Two Observations

- **Observation 1:** A group of data records that contains descriptions of a set of similar objects are typically presented in a contiguous region of a page and are formatted using similar HTML tags. Such a region is called a **data region**.
- **Observation 2:** A set of data records are formed by some child sub-trees of the same parent node.

The DOM Tree



The Approach

Given a page, three steps

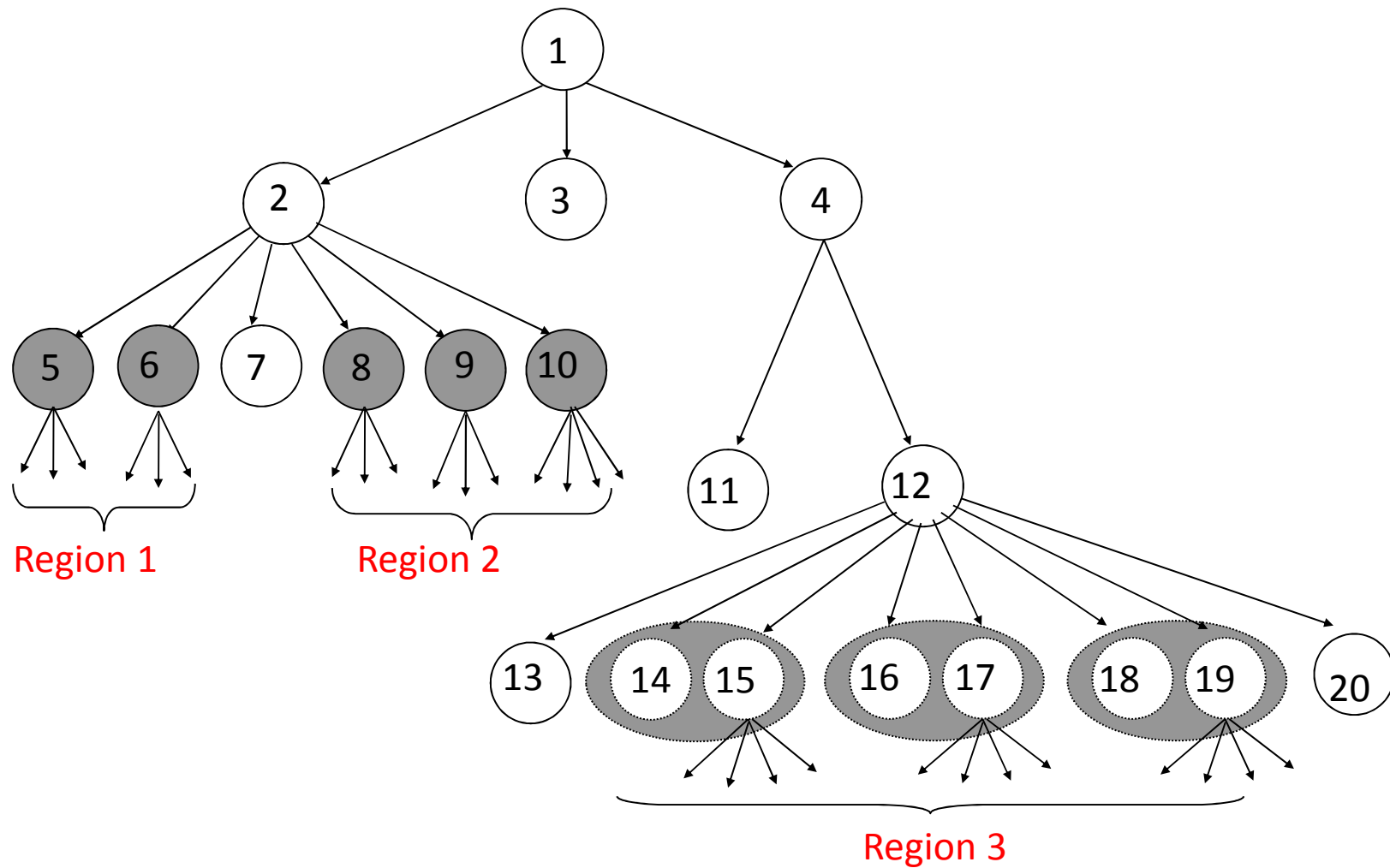
- Building the HTML Tag Tree
 - Erroneous tags, unbalanced tags, etc
- Mining Data Regions
 - String matching or tree matching
- Identifying Data Records

Rendering (or visual) information is very useful
in the whole process

Mining a Set of Similar Structures

- **Definition:** A *generalized node* (a *node combination*) of length r consists of r ($r \geq 1$) nodes in the tag tree with the following two properties:
 - the nodes all have the same parent.
 - the nodes are adjacent.
- **Definition:** A *data region* is a collection of two or more generalized nodes with the following properties:
 - the generalized nodes all have the same parent.
 - the generalized nodes all have the same length.
 - the generalized nodes are all adjacent.
 - the similarity between adjacent generalized nodes is greater than a *fixed threshold*.

Mining Data Regions

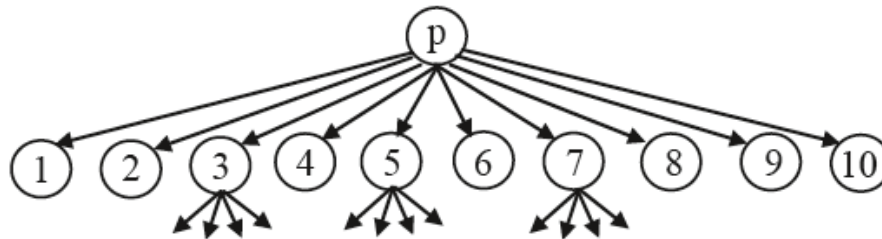


Mining Data Regions

- We need to find where each generalized node starts and where it ends.
 - perform string or tree matching
- Computation is not a problem anymore due to the two observations, we only need to perform comparisons among the children nodes of a parent node.
- Start from the root
- Given a node, perform string (or tree) comparison of all possible combinations of component nodes

Node Comparisons

- Say max components of a generalized node be 3



- Compute the following string or tree comparisons
 - Start from node 1
 - (1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8), (8, 9), (9, 10)
 - (1-2, 3-4), (3-4, 5-6), (5-6, 7-8), (7-8, 9-10)
 - (1-2-3, 4-5-6), (4-5-6, 7-8-9)
 - Start from node 2
 - (2-3, 4-5), (4-5, 6-7), (6-7, 8-9)
 - (2-3-4, 5-6-7), (5-6-7, 8-9-10)
 - Start from node 3
 - (3-4-5, 6-7-8)

Find Data Records from Generalized Nodes

- A generalized node may not represent a data record.
- In the example on the right, each row is found as a generalized node.
- This step needs to identify each of the 8 data record.
 - Not hard
 - We simply run the algorithm to mine data regions given each generalized node as input



Extract Data from Data Records

- Once a list of data records is identified, we can align and extract data items from them.
- Approaches (align multiple data records):
 - Multiple string alignment
 - Many ambiguities due to pervasive use of table related tags.
 - Multiple tree alignment (partial tree alignment)
 - Produce rooted trees for each data record
 - Together with visual information is effective
 - After alignments are done, the final seed tree can be used as the extraction pattern, or be turned into a regular expression.

Extraction Given a List Page: Nested Data Records

- Problem with the previous method
 - not suitable for nested data records, i.e., data records containing nested lists.
 - Since the number of elements in the list of each data record can be different, using a fixed threshold to determine the similarity of data records will not work.
- Solution idea
 - The problem, however, can be dealt with as follows.
 - Instead of traversing the DOM tree top down, we can traverse it post-order.
 - This ensures that nested lists at lower levels are found first based on repeated patterns before going to higher levels.
 - When a nested list is found, its records are **collapsed** to produce a single template (regex pattern).
 - This template replaces the list of nested data records.
 - When comparisons are made at a higher level, the algorithm only sees the template. Thus it is treated as a flat data record.

Extraction Given Multiple Pages

- Using previous techniques
 - **Given a set of list pages**
 - The techniques described in previous sections are for a single list page.
 - They can clearly be used for multiple list pages.
 - If multiple list pages are available, they may help improve the extraction.
 - For example, templates from all input pages may be found separately and merged to produce a single refined pattern.
 - This can deal with the situation where a single page may not contain the complete information.
 - Given a set of detail pages
 - For extraction, we can treat each detail page as a data record, and perform data extraction
 - create an artificial root node for each page
 - make the DOM tree of each page as a child sub-tree of the artificial root node
 - Perform partial tree alignments
 - Difficulty with many detail pages
 - Although a detail page focuses on a single object, the page may contain a large amount of “noise”, at the top, on the left and right and at the bottom.
 - Finding a set of detail pages automatically is non-trivial

The RoadRunner System

- Given a set of positive examples (multiple sample pages). Each contains one or more data records.
- From these pages, generate a wrapper as a union-free regular expression (i.e., no disjunction).
- Support nested data records.

The approach

- To start, a sample page is taken as the wrapper.
- The wrapper is then refined by solving mismatches between the wrapper and each sample page, which generalizes the wrapper.
 - A mismatch occurs when some token in the sample does not match the grammar of the wrapper.

Comparison Between Wrapper Induction and Automatic Extraction (1)

Wrapper induction

- Advantages
 - Only the target data are extracted as the user can label only data items that he/she is interested in.
 - Due to manual labeling, there is no integration issue for data extracted from multiple sites as the problem is solved by the user.
- Disadvantages
 - It is not scalable to a large number of sites due to significant manual efforts. Even finding the pages to label is non-trivial.
 - Wrapper maintenance (verification and repair) is very costly if the sites change frequently.

Comparison Between Wrapper Induction and Automatic Extraction (2)

Automatic extraction

- Advantages
 - It is scalable to a huge number of sites due to the automatic process.
 - There is little maintenance cost.
- Disadvantages
 - It may extract a large amount of unwanted data because the system does not know what is interesting to the user. Domain heuristics or manual filtering may be needed to remove unwanted data.
 - Extracted data from multiple sites need integration, i.e., their schemas need to be matched.

Comparison Between Wrapper Induction and Automatic Extraction (3)

- In terms of extraction accuracy, it is reasonable to assume that wrapper induction is more accurate than automatic extraction. However, there is no reported comparison.
- Applications
 - Wrapper induction should be used in applications in which the number of sites to be extracted and the number of templates in these sites are not large.
 - Automatic extraction is more suitable for large scale extraction tasks which do not require accurate labeling or integration.

Take-away Messages

- There is lots and lots of information in the structured, unstructured, semi-structured and the deep web
- It is not easy to extract that information
- Wrapper induction is one mechanism, however it involves manual efforts
- We looked at preliminary approaches for list extraction from the web

Further Reading

- Chapter 9 of “Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data” by Bing Liu
 - <http://rd.springer.com/book/10.1007/978-3-642-19460-3//page/1>
- Sunita Sarawagi. 2008. Information Extraction. *Found. Trends databases* 1, 3 (March 2008), 261-377.
DOI=10.1561/19000000003
<http://dx.doi.org/10.1561/19000000003>
- Nilesh Dalvi, Ashwin Machanavajjhala, Bo Pang. An Analysis of Structured Data on the Web. VLDB 2012.
 - http://vldb.org/pvldb/vol5/p680_nileshdalvi_vldb2012.pdf

Preview of Lecture 18: Mining Structured Information from the Web (Part 2)

- Extracting Top-K Lists from the Web
- Extracting Web Data Records Containing User-Generated Content
- Extracting Tables from the Web
 - WebTables: Exploring the Power of Tables on the Web
 - Answering Table Augmentation Queries from Unstructured Lists on the Web
 - Annotating Tables with Ontological Links
- Extracting Sets from the Web

Disclaimers

- This course represents opinions of the instructor only. It does not reflect views of Microsoft or any other entity (except of authors from whom the slides have been borrowed).
- Algorithms, techniques, features, etc mentioned here might or might not be in use by Microsoft or any other company.
- Lot of material covered in this course is borrowed from slides across many universities and conference tutorials. These are gratefully acknowledged.

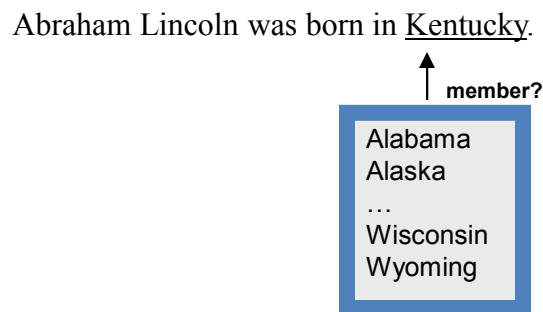
Thanks!

Today's Agenda

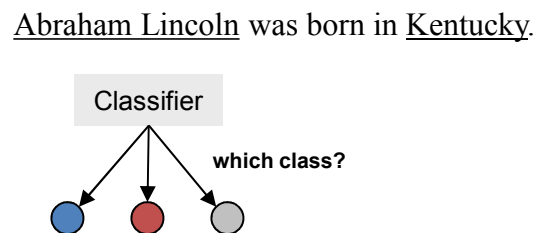
- Introduction to Information Extraction
- Wrapper Induction
- List Extraction using Automatic Wrapper Generation
- **Information Extraction for Unstructured Data**

Entity Extraction Models for Un-structured Data

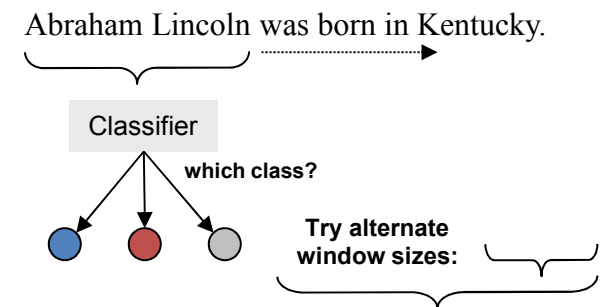
Lexicons



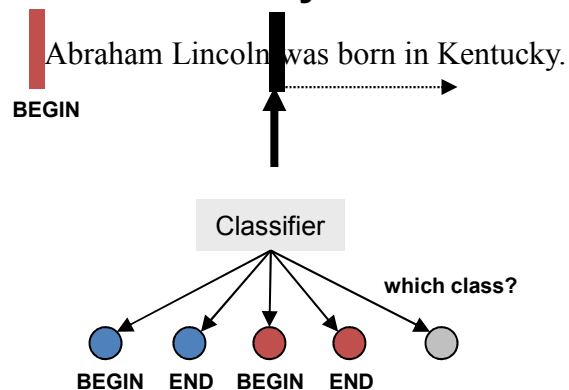
Classify Pre-segmented Candidates



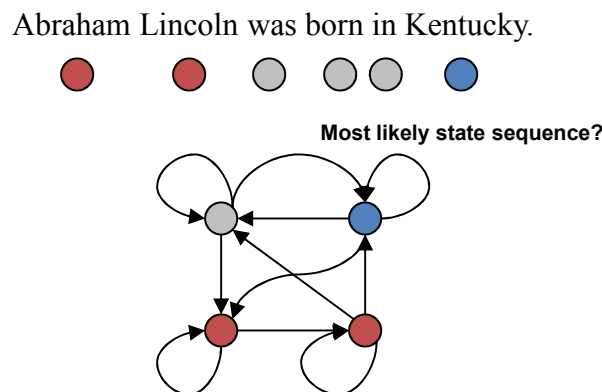
Sliding Window



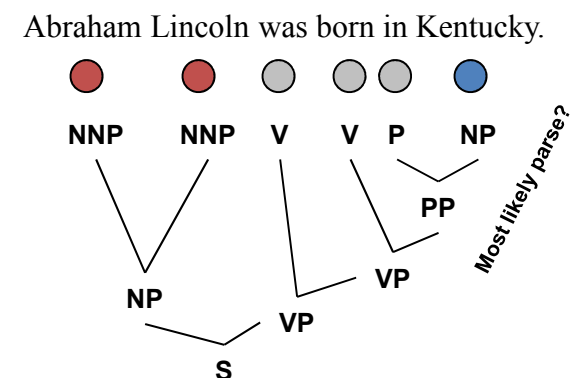
Boundary Models



Finite State Machines



Context Free Grammars



...and beyond

Any of these models can be used to capture words, formatting or both. 55

Relation Extraction

May 19 1995, Atlanta -- The Centers for Disease Control and Prevention, which is in the front line of the world's response to the deadly **Ebola** epidemic in **Zaire**, is finding itself hard pressed to cope with the crisis...

Relation Extraction

Disease Outbreaks relation

Date	Disease Name	Location
Jan. 1995	Malaria	Ethiopia
July 1995	Mad Cow Disease	U.K.
Feb. 1995	Pneumonia	U.S.

- Extract tuples of entities that are related in predefined way
 - Experts develop rules, patterns:
 - Can be defined over lexical items: “<company> located in <location>” over syntactic structures: “((Obj <company>) (Verb located) (*) (Subj <location>))”
- Machine learning
 - Supervised: Train system over manually labeled data
 - Partially-supervised: train system by bootstrapping from “seed” examples:
 - Hybrid or interactive systems:
 - Experts interact with machine learning algorithms (e.g., active learning family) to iteratively refine/extend rules and patterns
 - Interactions can involve annotating examples, modifying rules, or any combination

Event Extraction

- Similar to Relation Extraction, but:
 - Events can be **nested**
 - Significantly **more complex** (e.g., more slots) than relations/template elements
 - Often requires **coreference** resolution, **disambiguation**, **deduplication**, and **inference**
- Example: an integrated disease outbreak event [Hatunnen et al. 2002]

