POLITECNICO DI MILANO

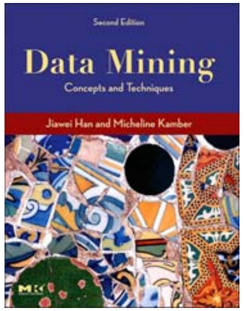
Web Mining

Data Mining and Text Mining (UIC 583 @ Politecnico di Milano)

Daniele Loiacono

References

- ❑ Jiawei Han and Micheline Kamber, "Data Mining: Concepts and Techniques", The Morgan Kaufmann Series in Data Management Systems (Second Edition)
  - Chapter 10

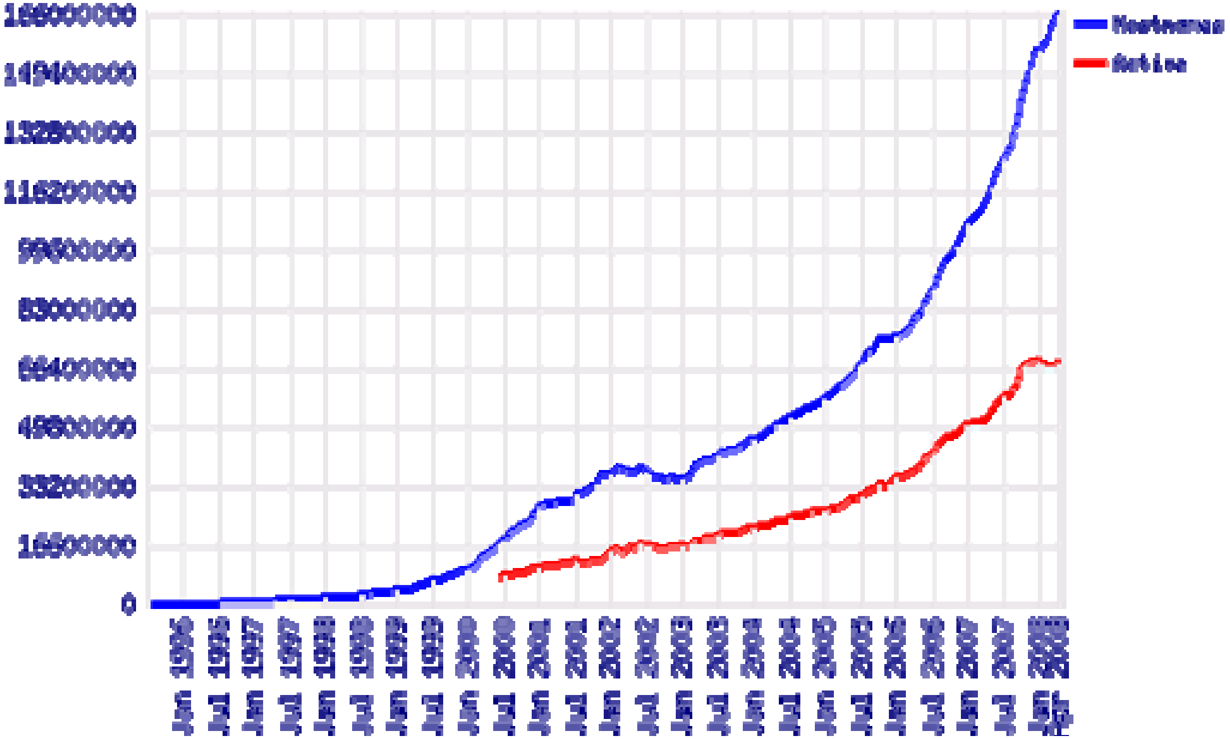


- ❑ **Web Mining Course** by *Gregory-Platesky Shapiro* available at www.kdnuggets.com

- ❑ Federico Facca and Pier Luca Lanzi. **Mining Interesting Knowledge from Weblogs: A Survey**. *Journal of Data and Knowledge Engineering*, 53(3):225–241, 2005.

How big is the Web?

165,719,150 Web Sites @Apr 2008 (Netcraft Survey)



What is Web Mining?

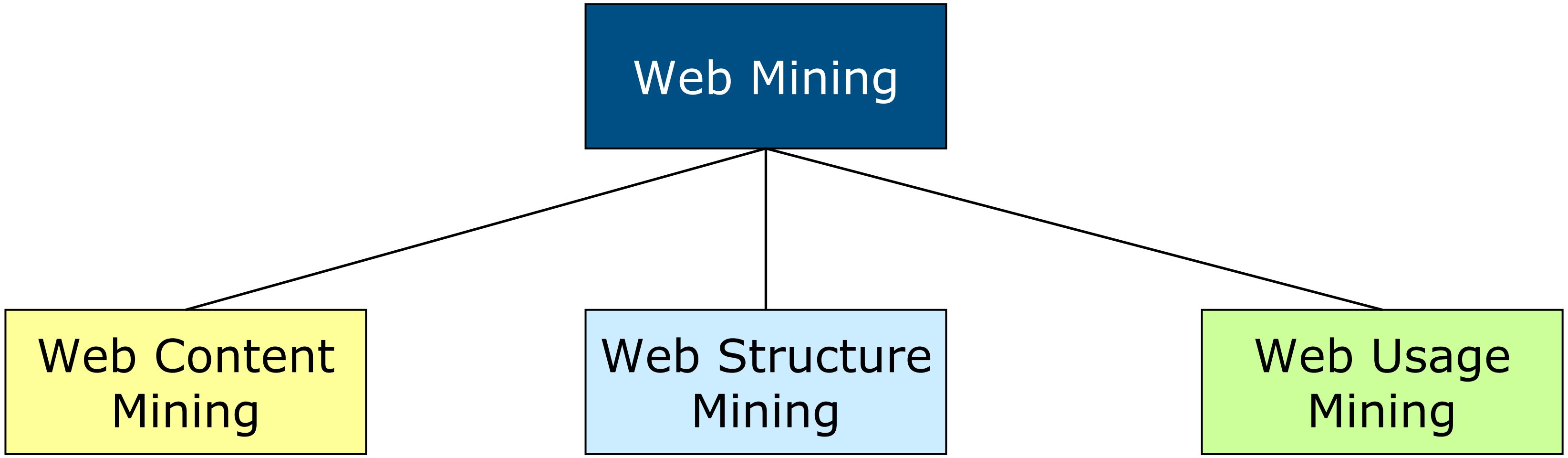
Discovering interesting and useful information from Web content and usage

- Examples
  - Web search, e.g. Google, Yahoo, MSN, Ask, …
  - Specialized search: e.g. Froogle (comparison shopping), job ads (Flipdog)
  - eCommerce
  - Recommendations (Netflix, Amazon, etc.)
  - Improving conversion rate: next best product to offer
  - Advertising, e.g. Google Adsense
  - Fraud detection: click fraud detection, …
  - Improving Web site design and performance

Web Mining Challenges

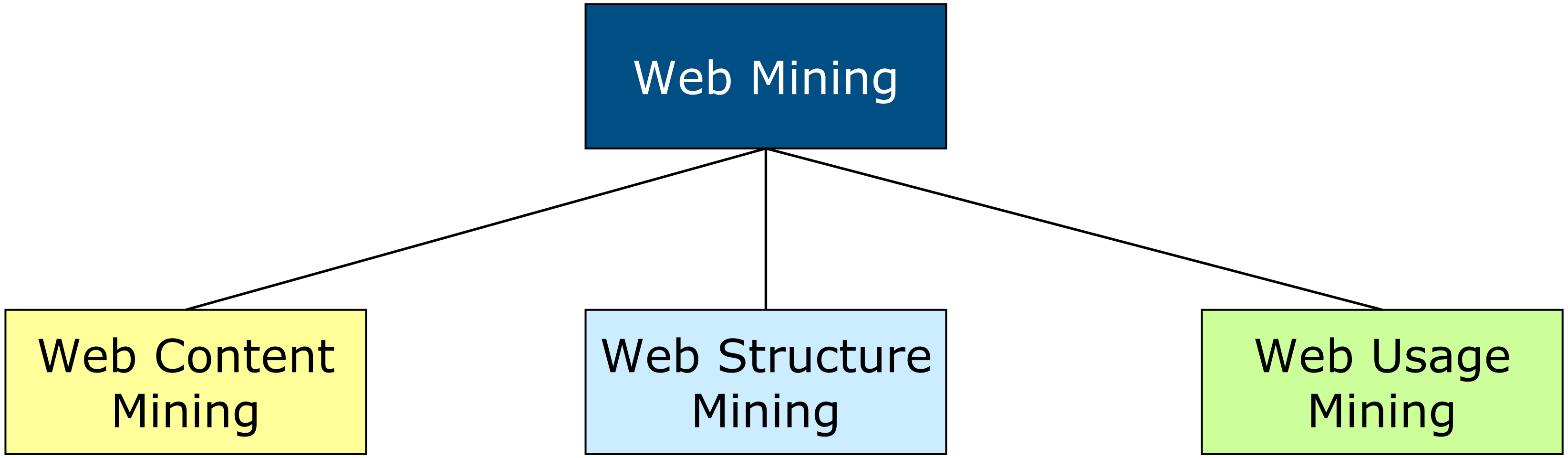
- ❑ Huge amount of data
- ❑ Complexity of Web pages
  - Different styles
  - Different contents
- ❑ Highly dynamic and rapidly growing information
  - Number of sites is rapidly growing
  - Information is constantly updated
- ❑ Web serves many user communities
  - Users with different interests, background and purposes
  - "99% of the Web information is useless to 99% of Web users"

Web Mining Taxonomy



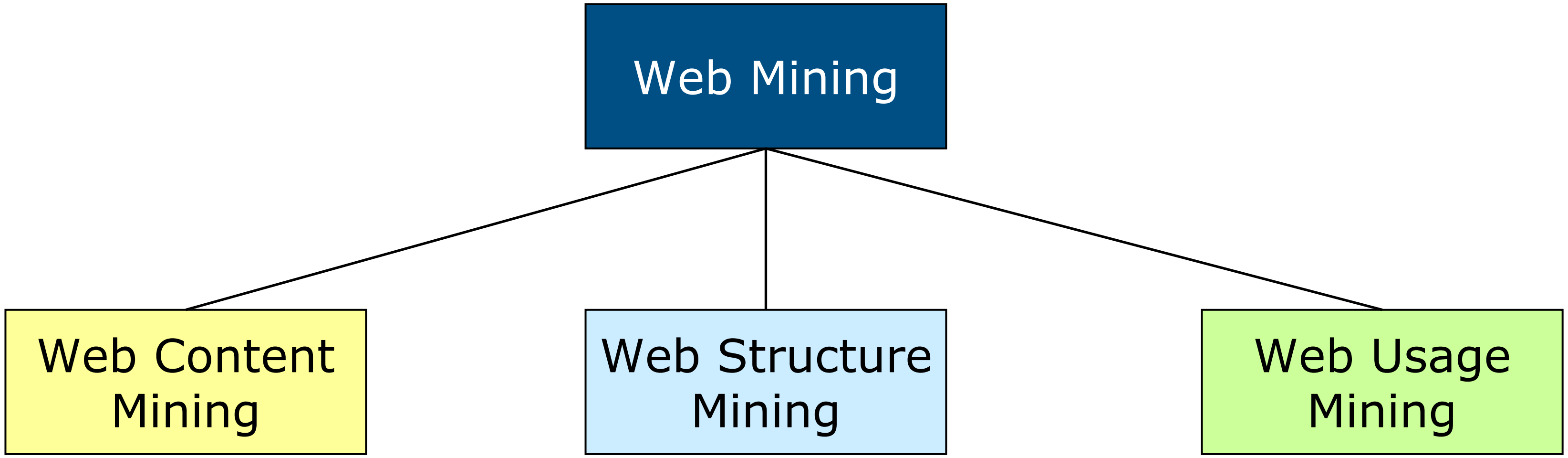
- ❑ Summarization of Web pages
- ❑ Summarization of Web searches
- ❑ Mining multimedia Web content
- ❑ Web pages classification
- ❑ …

Web Mining Taxonomy



- ❑ Mining linking structure
- ❑ Discover authoritative pages
    - PageRank
- ❑ Discover hub

Web Mining Taxonomy

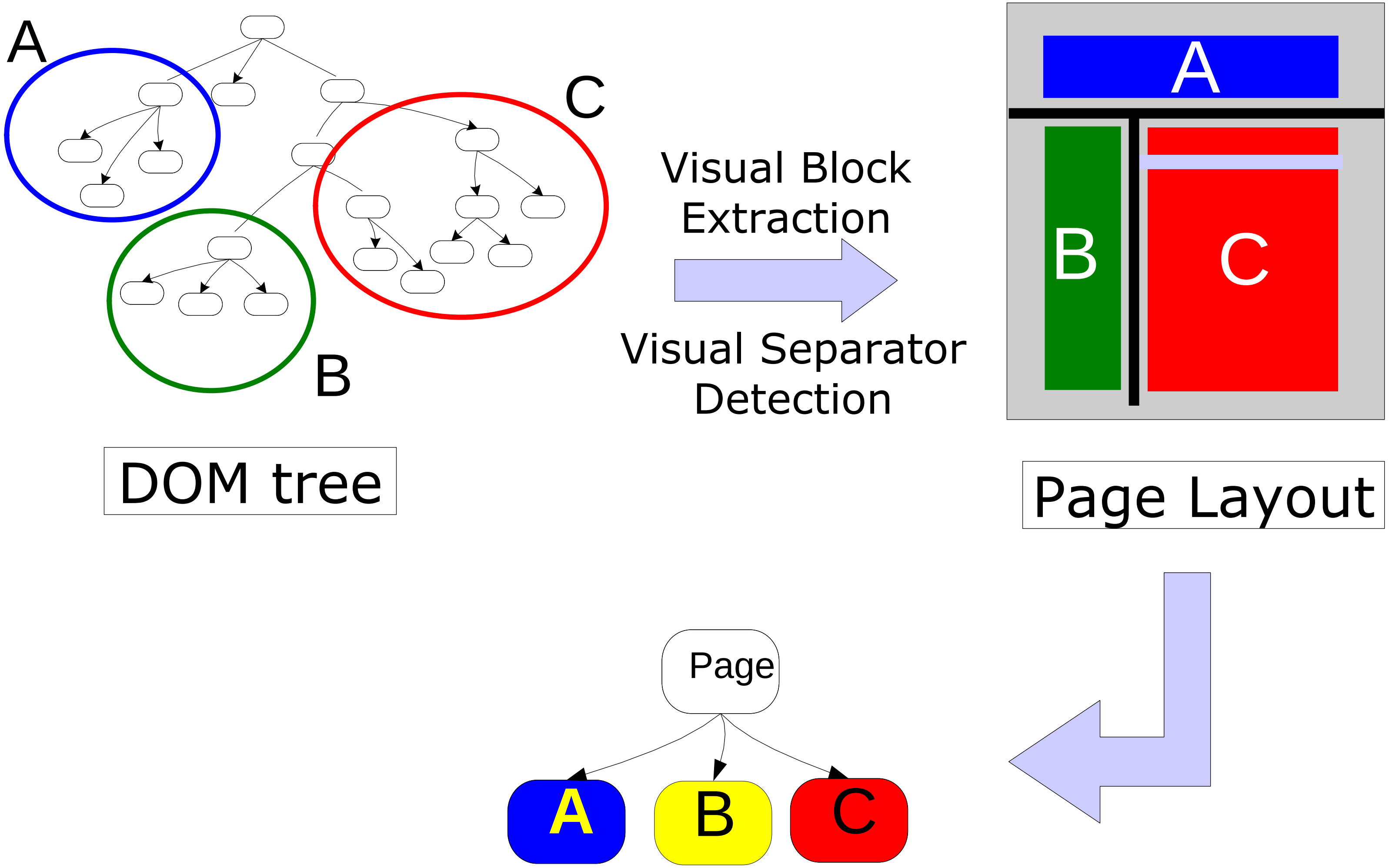


- ❑ Mining weblogs to discover usage patterns
- ❑ Applications:
  - Personalization of Web content
  - Improve Web design

Mining Web Page Layout Structure

- Web page is more than plain text
- Web page structure is defined by the **DOM** (Document Object Model) tree, where nodes are the **HTML tags**
- Issues
  - Not all the pages follows the standards
  - DOM tree does not always reflect the page semantic

Vision-based Page Segmentation



DOM tree

Visual Block Extraction

Visual Separator Detection

Page Layout

Page

Mining Web's Link Structure

- How to identify **authoritative** page?
- The answer is in the **Web linkage structure**
- Issues in Web linkage
  - Links do not always represent endorsements (e.g., adv)
  - Important competitors do not usually link each other
  - Authoritative pages are generally not self-descriptive
- To discover authorities we should also look for **hub pages**
  - Hub are pages that provide **collections of links to authorities**
  - Hub pages are not necessary highly linked
  - Hub pages implicitly confer authorities on focused topics
- **Hub and authoritative pages have a mutual reinforcement relationship**
  - A good hub page points to many good authorities, a good authority is a page pointed by many good hub pages

Hyperlink-Induce Topic Search (1)

- Startup
  - **Root set** built from results from an index-based search engine
  - **Base set** built including pages linked by and linking to the root set pages
- Authority weight, a_p, and hub weight, h_p, are iteratively computed

$$\boldsymbol{a_p = \sum_{\forall q: q \to p} h_q} \qquad h_p = \sum_{\forall q: q \leftarrow p} a_q$$

- In matrix form

$$\begin{cases} \vec{h} = \boxed{A}\vec{a} = \cdots = (AA^T)^k \vec{h} \\ \vec{a} = A^T \vec{h} = \cdots = (A^T A)^k \vec{a} \end{cases}$$

Adiacency Matrix

- The **authority weight vector** and the **hub weight vector** if normalized converge to the eigenvectors of AA^T and A^TA

Hyperlink-Induce Topic Search (2)

- Underlying assumptions:
  - Links convey endorsement
  - Pages co-linked by a certain page are likely to be related to the same topic
- VIPS-based approach
  - **Block-to-page** relationship

$$Z_{ij} = \begin{cases} 1/s_i, & \text{if block } i \text{ point to page } j \\ 0, & \text{otherwise} \end{cases}$$

    where s_i is the number of pages linked by block *i*
  - **Page-to-block** relationship

$$X_{ij} = \begin{cases} f_{p_i}(b_j), & \text{if } b_j \in p_i \\ 0, & \text{otherwise} \end{cases}$$

    where $f_p(b)$ represents how *b* is important in page *p*
  - Adjacency matrix can be defined as

$$W_P = XZ$$

Mining Multimedia Data on the Web

- Is different from general-purpose multimedia data mining
  - Multimedia data is embedded in Web pages
  - Links and surrounding text might help the data mining process
- VIPS algorithm is the basis to extract knowledge
  - A **bock-to-image** relationship can be build
  - The block-to-image relationship can be integrated with a block-level link analysis
  - The resulting **image graph** reflect the semantic relationship between the images
- The image graph can be used for classification and clustering purposes

Web Usage Mining

Web usage mining is the extraction of interesting knowledge from server log files

- Applications
  - Mining logs of a single user
    - Web content personalization
  - Mining logs of groups of users
    - Supporting Web design
- Issues
  - Where is the data?
  - How to preprocess the data?
  - Which mining techniques?

Data sources

- Logs can be collected at different levels
  - Server side
  - Proxy side
  - Client side

Data sources: server side

- Web server log
  - Standard format (e.g., LogML)
  - Large amount of information (IP, request info, etc.)
  - User session can be difficult to identify
  - Special buttons (e.g., *Back, Stop*) cannot be tracked
- TCP/IP packet sniffer
  - Data collected in real-time
  - Data from different web servers can be merged easily
  - Some special buttons can be tracked (e.g. *Stop*)
  - Does not scale very well
- Exploiting the server application layer
  - Very effective
  - Not always possible
  - Requires ad-hoc solutions for each web server

Data sources: proxy side

- ❑ Almost the same information available on server side
- ❑ Data of **groups of users** accessing to **huge groups of web servers**
- ❑ Sessions can be anyway identified

Data sources: client side

- ❑ Collecting data with JavaScript or Java applets
- ❑ Exploiting a modified Web browser
- ❑ Perfect identification of the user session
- ❑ Requires user collaboration

Preprocessing: data cleaning

- Data cleaning consists of removing from Web logs useless data for mining purposes
- Content requests (e.g. images) are usually easily removed
- Robots and Web spiders should be removed on the basis of
  - Remote hostname
  - Access to robots.txt
  - Navigation pattern

Preprocessing: session identification and reconstruction

- Goals
  - Identifying the session of different users
  - Reconstruction the navigation path in identified session
- Challenges
  - Proxy
  - Browser caching and special buttons
  - Sessionization
- Solutions
  - Cookies
  - URL rewriting
  - JavaScript (e.g. SurfAid)
  - Consistency of navigation path
  - Timeout heuristic for session termination

Applications

- Personalization of Web content
  - Behavior anticipation
  - Recommendation of interesting links
  - Content reorganizations
- Pre-fetching and caching
  - Caching and pre-fetching of content to reduce the server response time
- Support to Web design
  - Analysis of frequent patterns to improve the usability of Web sites
- E-commerce
  - Analysis of customer behaviors (attrition, fidelity, etc.)

Preprocessing: content retrieving

- Generally URLs are the only information available on pages
- A richer information about visited pages may help the discovering of interesting Web usage patterns
- Main approaches
  - Pages categorization
    - Pre-defined
    - Automatically discovered with Web mining techniques
  - Semantic Web for Web Usage Mining
    - Ontology mapping
    - Learning of ontology from data
    - Extraction of concept-based navigation paths

Mining Techniques

- The main techniques used for the analysis of collected data are
  - Association rules

    A.html, B.html => C.html

  - Sequential patterns extraction
    - General purpose algorithm (e.g., AprioriAll)
    - Ad hoc solution for Web logs (WAP-mine)
  - Clustering of sessions
    - Based on sequence alignment
    - *Association rule hypergraph partitioning*
      - build a graph representing frequent patterns
      - Edges weighting based on pattern relevance
      - Partitioning of graph to extract users’ behaviors