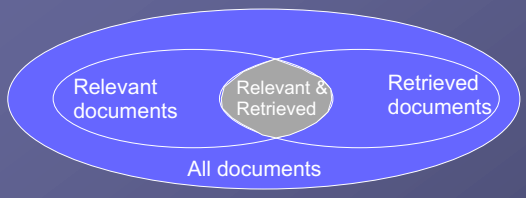
Text Databases and IR

- Text databases (document databases)
  - Large collections of documents from various sources: news articles, research papers, books, digital libraries, e-mail messages, and Web pages, library database, etc.
  - Data stored is usually *semi-structured*
  - Traditional information retrieval techniques become inadequate for the increasingly vast amounts of text data
- Information retrieval vs. database systems
  - Some DB problems are not present in IR, e.g., update, transaction management, complex objects, concurrency control, recovery
  - Some IR problems are not addressed well in DBMS, e.g., unstructured documents, approximate search using keywords and the notion of relevance

Information Retrieval

- Information retrieval
  - A field developed in parallel with database systems
  - Information is organised into (a large number of) documents
  - Information retrieval problem: locating relevant documents based on user input, such as keywords or example documents
- Typical IR systems
  - Online library catalogue systems
  - Online document management systems

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

$$recall = \frac{|\{relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

Keyword-Based Retrieval

- Keywords
  - 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 Databases

- The Problem
  - 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, with,* etc.
  - Stop lists may vary when document set varies

Similarity-Based Retrieval in Text Databases (2)

- 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)* = number of occurrences of the word t_i in document d_j
  - 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: $sim(v_1, v_2) = \frac{v_1 v_2}{|v_1||v_2|}$

Latent Semantic Indexing

- Basic idea
  - Similar documents have similar word frequencies
  - Difficulty: the size of the term frequency matrix is very large
  - Use a singular value decomposition (SVD) techniques to reduce the size of frequency table
  - Retain the *K* most significant rows of the frequency table
- Method
  - Create a term frequency matrix, *freq_matrix*
  - SVD construction: Compute the singular value decomposition of *freq_matrix* by splitting it into 3 matrices, *U*, *S*, *V*
  - Vector identification: For each document *d*, replace its original document vector by a new term excluding the eliminated terms
  - Index creation: Store the set of all vectors, indexed by one of a number of techniques

Other Text Retrieval Indexing Techniques

- Inverted index
  - Maintains two hash- or B+-tree indexed tables:
    - Document table: a set of document records <doc_id, post_list>
    - Term table: a set of term records, <term, posting_list>
  - Answer query: Find all docs associated with one or a set of terms
  - Advantage: easy to implement
  - Disadvantage: do not handle well synonymy and polysemy, and posting lists could be too long (storage could be very large)
- Signature file
  - 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
  - Associate a signature with each document

Types of Text Data Mining

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

Keyword-based association analysis

- Definition
  - Collect sets of keywords or terms that occur frequently together and then find the association or correlation relationships among them
- Pre-process
  - Process the text data by parsing, stemming, removing stop words, etc.
- 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

Automatic document classification

- Motivation
  - Automatic classification for the tremendous number of on-line text documents (Web pages, e-mails, etc.)
- A classification problem
  - Training set: Human experts generate a training data set
  - Classification: The computer system discovers the classification rules
  - Application: The discovered rules can be applied to classify new/unknown documents
- Remark
  - Text document classification differs from the classification of relational data
  - Document databases are not structured according to attribute-value pairs

Association-Based Document Classification

- Extract keywords and terms by information retrieval and simple association analysis techniques
- Obtain concept hierarchies of keywords and terms using
  - Available term classes, such as WordNet
  - Expert knowledge
  - Some keyword classification systems
- Classify documents in the training set into class hierarchies
- Apply term association mining method to discover sets of associated terms
- Use terms to maximally distinguish 1 class of documents from others
- Derive a set of association rules associated with each document class
- Order the classification rules based on their occurrence frequency and discriminative power
- Use the rules to classify new documents

Document Clustering

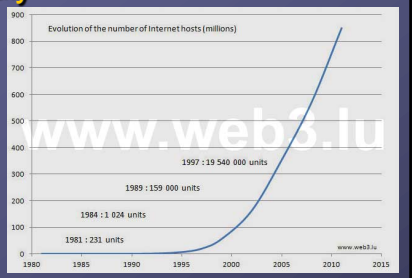
- Goal
  - Automatically group related documents based on their contents
- Remark
  - Require no training sets or predetermined taxonomies, generate a taxonomy at runtime
- Major steps
  - Pre-processing
    - Remove stop words, stem, feature extraction, lexical analysis,
  - Hierarchical clustering
    - Compute similarities applying clustering algorithms,

Mining the World-Wide Web

- The WWW is huge, widely distributed, global information service centre for
  - Information services: news, advertisements, consumer information, financial management, education, government, e-commerce, etc.
  - Rich & dynamic hyper-link information
  - Web page access and usage information
  - WWW provides rich sources for data mining
- Challenges
  - Too large for effective data warehousing and data mining
  - Too complex and heterogeneous: no standards and structure

Mining the World-Wide Web

- Growing and changing very rapidly
  - January 2017 → 1.06 billion Internet hosts were available
  - More than double in 10 years



- Serves broad diversity of user communities
- Only a small portion of the information on the Web is truly relevant or useful
  - 99% of the Web information is useless to 99% of Web users
  - How can we find high-quality Web pages on a specified topic?

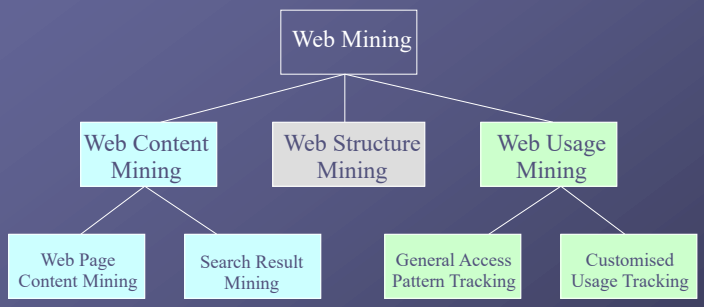
Web search engines

- Index-based
  - Search the Web, index Web pages, and build and store huge keyword-based indices
  - Help to locate sets of Web pages containing certain keywords
- Deficiencies
  - A topic of any breadth may easily contain hundreds of thousands of documents
  - Many documents that are highly relevant to a topic may not contain keywords defining them (polysemy)

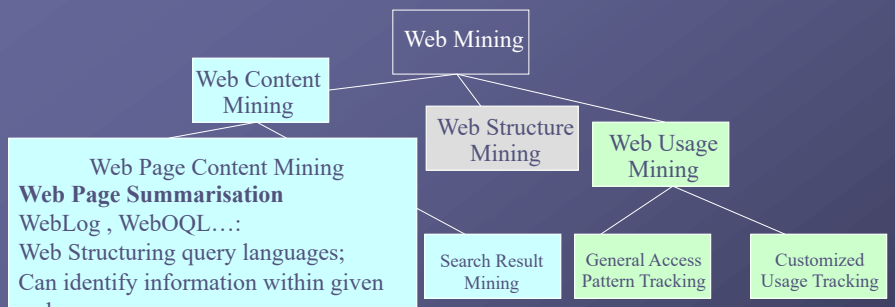
Web Mining: A more challenging task

- Search for
  - Web access patterns
  - Web structures
  - Regularity and dynamics of Web contents
- Problems
  - The "abundance" problem
  - Limited coverage of the Web: hidden Web sources, majority of data in DBMS
  - Limited query interface based on keyword-oriented search
  - Limited customisation to individual users

Web Mining Taxonomy



Mining the World-Wide Web



Web Page Content Mining

Web Page Summarisation

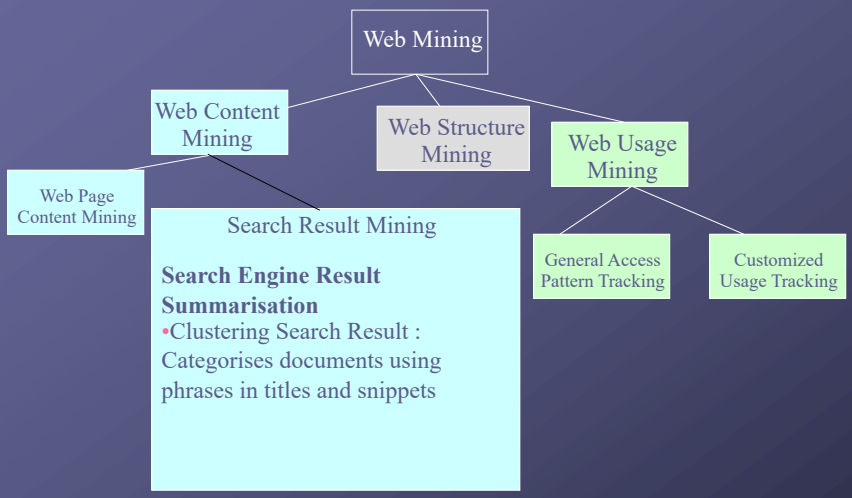
WebLog , WebOQL…:

Web Structuring query languages;

Can identify information within given web pages

- Ahoy:Uses heuristics to distinguish personal home pages from other web pages
- ShopBot: Looks for product prices within web pages

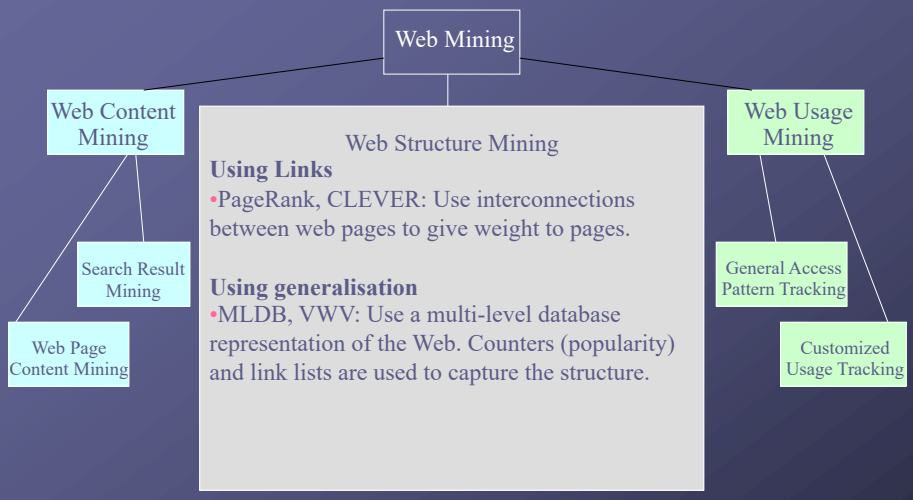
Mining the World-Wide Web



Search Engine Result Summarisation

- Clustering Search Result : Categorises documents using phrases in titles and snippets

Mining the World-Wide Web



Web Structure Mining

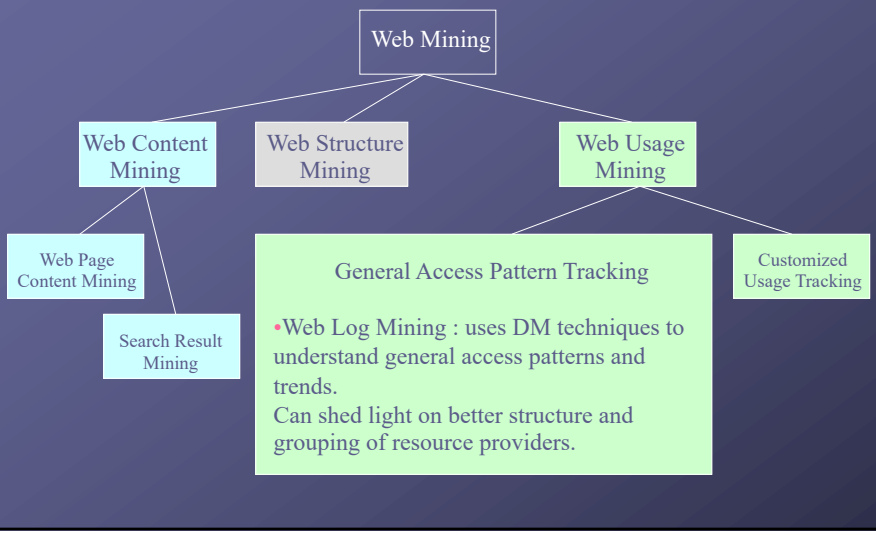
Using Links

- PageRank, CLEVER: Use interconnections between web pages to give weight to pages.

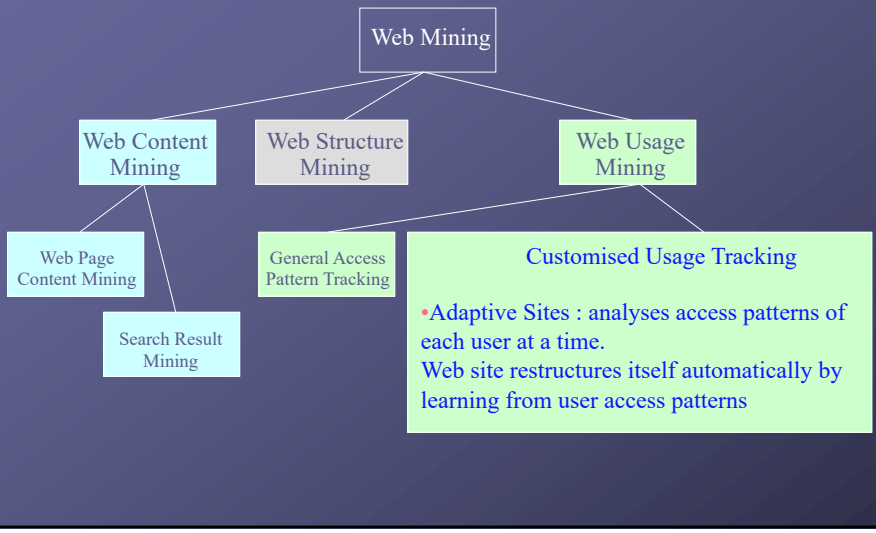
Using generalisation

- MLDB, VWV: Use a multi-level database representation of the Web. Counters (popularity) and link lists are used to capture the structure.

Mining the World-Wide Web



Mining the World-Wide Web



Mining the Web's Link Structures

- Finding authoritative Web pages
  - Retrieving pages that are not only relevant, but also of high quality, or authoritative on the topic
- Hyperlinks can infer the notion of authority
  - The Web consists not only of pages, but also of hyperlinks pointing from one page to another
  - These hyperlinks contain an enormous amount of latent human annotation
  - A hyperlink pointing to another Web page, this can be considered as the author's endorsement of the other page

Mining the Web's Link Structures

- Problems with the Web linkage structure
  - Not every hyperlink represents an endorsement
    - Other purposes are for navigation or for paid advertisements
    - If the majority of hyperlinks are for endorsement, the collective opinion will still dominate
  - One authority will rarely have its Web page point to its rival authorities in the same field
  - Authoritative pages are not often particularly descriptive
- Hub (another important category of Web pages)
  - Set of Web pages that provides collections of links to authorities

HITS (Hyperlink-Induced Topic Search)

- Explore interactions between hubs and authoritative pages
- Use an index-based search engine to form the root set
  - Many of these pages are presumably relevant to the search topic
  - Some of them should contain links to the majority of the prominent authorities
- Expand the root set into a base set
  - Include all of the pages that the root-set pages link to, and all of the pages that link to a page in the root set, up to a designated size cut-off
- Apply weight-propagation
  - An iterative process that determines numerical estimates of hub and authority weights

Systems Based on HITS

- The Problem
  - Output a short list of the pages with large hub weights, and the pages with large authority weights for the given search topic
- Systems based on the HITS algorithm
  - Clever, Google: achieve better quality search results than those generated by term-index engines such as AltaVista and those created by human ontologists such as Yahoo!
- Difficulties from ignoring textual contexts
  - Drifting: when hubs contain multiple topics
  - Topic hijacking: when many pages from a single Web site point to the same single popular site

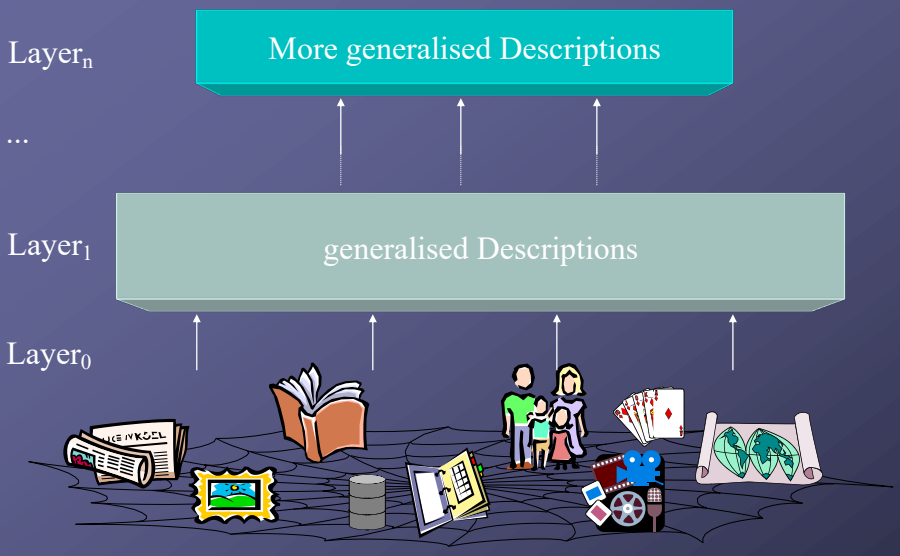
Automatic Classification of Web Documents

- Class label
  - Assign a class label to each document from a set of predefined topic categories
  - Based on a set of examples of pre-classified documents
- Example
  - Use Yahoo!'s taxonomy and its associated documents as training and test sets
  - Derive a Web document classification scheme
  - Use the scheme to classify new Web documents by assigning categories from the same taxonomy
- Keyword-based document classification methods
- Statistical models

Multilayered Web Information Base

- $Layer_0$: the Web itself
- $Layer_1$: the Web page descriptor layer
  - Contains descriptive information for pages on the Web
  - An abstraction of $Layer_0$: substantially smaller but still rich enough to preserve most of the interesting, general information
  - Organised into dozens of semi-structured classes
    - *document, person, organisation, ads, directory, sales, software, game, stocks, library_catalog, geographic_data, scientific_data*, etc.
- $Layer_2$ and up: various Web directory services constructed on top of $Layer_1$
  - provide multidimensional, application-specific services

Multiple Layered Web Architecture



Mining the WWW

Layer-0: Primitive data

Layer-1: dozen of database relations representing types of objects (metadata)

document, organisation, person, software, game, map, image, ...

• **document**(file_addr, authors, title, publication, publication_date, abstract, language, table_of_contents, category_description, keywords, index, multimedia_attached, num_pages, format, first_paragraphs, size_doc, timestamp, access_frequency, links_out,...)

• **person**(last_name, first_name, home_page_addr, position, picture_attached, phone, e-mail, office_address, education, research_interests, publications, size_of_home_page, timestamp, access_frequency, ...)

• **image**(image_addr, author, title, publication_date, category_description, keywords, size, width, height, duration, format, parent_pages, colour_histogram, Colour_layout, Texture_layout, Movement_vector, localisation_vector, timestamp, access_frequency, ...)

Mining the World-Wide Web

Layer-2: simplification of layer-1

- **doc_brief**(file_addr, authors, title, publication, publication_date, abstract, language, category_description, key_words, major_index, num_pages, format, size_doc, access_frequency, links_out)
- **person_brief** (last_name, first_name, publications, affiliation, e-mail, research_interests, size_home_page, access_frequency)

Layer-3: generalisation of layer-2

- **cs_doc**(file_addr, authors, title, publication, publication_date, abstract, language, category_description, keywords, num_pages, form, size_doc, links_out)
- **doc_summary**(affiliation, field, publication_year, count, first_author_list, file_addr_list)
- **doc_author_brief**(file_addr, authors, affiliation, title, publication, pub_date, category_description, keywords, num_pages, format, size_doc, links_out)
- **person_summary**(affiliation, research_interest, year, num_publications, count)

XML and Web Mining

- XML can help to extract the correct descriptors
  - Standardisation would greatly facilitate information extraction

```
<NAME> eXtensible Markup Language</NAME>
<RECOM>World-Wide Web Consortium</RECOM>
<SINCE>1998</SINCE>
<VERSION>1.0</VERSION>
<DESC>Meta language that facilitates more meaningful and
precise declarations of document content</DESC>
<HOW>Definition of new tags and DTDs</HOW>
```

  - Potential problem
    - XML can help solve heterogeneity for vertical applications, but the freedom to define tags can make horizontal applications on the Web more heterogeneous

Benefits of Multi-Layer Meta-Web

- Benefits:
  - Multi-dimensional Web info summary analysis
  - Approximate and intelligent query answering
  - Web high-level query answering (WebSQL, WebML)
  - Web content and structure mining
  - Observing the dynamics/evolution of the Web
- Is it realistic to construct such a meta-Web?
  - Benefits even if it is partially constructed
  - Benefits may justify the cost of tool development, standardisation and partial restructuring

Web Usage Mining

- Mining Web log records to discover user access patterns of Web pages
- Applications
  - Target potential customers for electronic commerce
  - Enhance the quality and delivery of Internet information services to the end user
  - Improve Web server system performance
  - Identify potential prime advertisement locations
- Web logs provide rich information about Web dynamics
  - Typical Web log entry includes the URL requested, the IP address from which the request originated, and a timestamp

Techniques for Web usage mining

- Construct multidimensional view on the Weblog database
  - Perform multidimensional OLAP analysis to find the top *N* users, top *N* accessed Web pages, most frequently accessed time periods, etc.
- Perform data mining on Weblog records
  - Find association patterns, sequential patterns, and trends of Web accessing
  - May need additional information,e.g., user browsing sequences of the Web pages in the Web server buffer
- Conduct studies to
  - analyse system performance, improve system design by Web caching, Web page pre-fetching, and Web page swapping

Mining the WWW

- Design of a Web Log Miner
  - Web log is filtered to generate a relational database
  - A data cube is generated from database
  - OLAP is used to drill-down and roll-up in the cube
  - OLAM is used for mining interesting knowledge