Data Mining: Concepts and Techniques — Slides for Textbook — — Chapter 2 —

© Jiawei Han and Micheline Kamber
Intelligent Database Systems Research Lab
School of Computing Science
Simon Fraser University, Canada
http://www.cs.sfu.ca

April 3, 2003 Data Mining: Concepts and Techniques

Last Class. Introduction

- Motivation: Why data mining?
- What is data mining?
- Data Mining: On what kind of data?
- Data mining functionality
- Are all the patterns interesting?
- Major issues in data mining

Evolution of Database Technology

- 1960s:
  - Data collection, database creation, IMS and network DBMS
- 1970s:
  - Relational data model, relational DBMS implementation
- 1980s:
  - RDBMS, advanced data models (extended-relational, OO, deductive, etc.) and application-oriented DBMS (spatial, scientific, engineering, etc.)
- 1990s—2000s:
  - Data mining and data warehousing, multimedia databases, and Web databases

Steps of a KDD Process (Han)

- Learning the application domain:
  - relevant prior knowledge and goals of application
- Creating a target data set: data selection
- Data cleaning and preprocessing: (may take 60% of effort!)
- Data reduction and transformation:
  - Find useful features, dimensionality/variable reduction, invariant representation.
- Choosing functions of data mining
  - summarization, classification, regression, association, clustering.
- Choosing the mining algorithm(s)
- Data mining: search for patterns of interest
- Pattern evaluation and knowledge presentation
  - visualization, transformation, removing redundant patterns, etc.
- Use of discovered knowledge

April 3, 2003 Data Mining: Concepts and Techniques

HW#1: due April 10, 2003

- [http://varuna.ece.ucsb.edu/ece594N]: this will be active by tomorrow (Thursday) morning.
- Present an example where data mining is crucial to the success of a business. What data mining functions does this business need? Can they be performed alternatively by data query processing or simple statistical analysis?
- In answering the above (or otherwise), describe the challenges to data mining regarding data mining methodology and user interaction issues.

Last Class: “Necessity is the Mother of Invention”

- Data explosion problem
  - Automated data collection tools and mature database technology lead to tremendous amounts of data stored in databases, data warehouses and other information repositories
- Data rich, information poor!
- Solution: Data warehousing and data mining
  - Data warehousing and on-line analytical processing
  - Extraction of interesting knowledge (rules, regularities, patterns, constraints) from data in large databases

April 3, 2003 Data Mining: Concepts and Techniques
Data Mining Functionalities (1)

- Concept description: Characterization and discrimination
  - Generalize, summarize, and contrast data characteristics, e.g., dry vs. wet regions
- Association (correlation and causality)
  - Multi-dimensional vs. single-dimensional association
  - \( \text{age}(X, "20..29") \land \text{income}(X, "20..29K") \rightarrow \text{buys}(X, "PC") \) [support = 2%, confidence = 60%]
  - \( \text{contains}(T, "computer") \rightarrow \text{contains}(X, "software") \) [1%, 75%]

Data Mining Functionalities (2)

- Classification and Prediction
  - Finding models (functions) that describe and distinguish classes or concepts for future prediction
  - E.g., classify countries based on climate, or classify cars based on gas mileage
  - Presentation: decision-tree, classification rule, neural network
  - Prediction: Predict some unknown or missing numerical values
- Cluster analysis
  - Class label is unknown: Group data to form new classes

Data Mining Functionalities (3)

- Outlier analysis
  - Outlier: a data object that does not comply with the general behavior of the data
  - It can be considered as noise or exception but is quite useful in fraud detection, rare events analysis
- Trend and deviation analysis
  - Trend and deviation: regression analysis
  - Sequential pattern mining, periodicity analysis
  - Similarity-based analysis
  - Other pattern-directed or statistical analyses

Data Mining: Confluence of Multiple Disciplines

- Database Technology
- Statistics
- Machine Learning
- Data Mining
- Visualization
- Information Science
- Other Disciplines

Major Issues in Data Mining (1)

- Mining methodology and user interaction
  - Mining different kinds of knowledge in databases
  - Interactive mining of knowledge at multiple levels of abstraction
  - Incorporation of background knowledge
  - Data mining query languages and ad-hoc data mining
  - Expression and visualization of data mining results
  - Handling noise and incomplete data
  - Pattern evaluation: the interestingness problem
- Performance and scalability
  - Efficiency and scalability of data mining algorithms
  - Parallel, distributed and incremental mining methods

Major Issues in Data Mining (2)

- Issues relating to the diversity of data types
  - Handling relational and complex types of data
  - Mining information from heterogeneous databases and global information systems (WWW)
- Issues related to applications and social impacts
  - Application of discovered knowledge
    - Domain-specific data mining tools
    - Intelligent query answering
    - Process control and decision making
  - Integration of the discovered knowledge with existing knowledge: A knowledge fusion problem
  - Protection of data security, integrity, and privacy
Summary

Data mining: discovering interesting patterns from large amounts of data
A natural evolution of database technology, in great demand, with wide applications
A KDD process includes data cleaning, data integration, data selection, transformation, data mining, pattern evaluation, and knowledge presentation
Mining can be performed in a variety of information repositories
Data mining functionalities: characterization, discrimination, association, classification, clustering, outlier and trend analysis, etc.
Classification of data mining systems
Major issues in data mining

Chapter 2: Data Warehousing and OLAP Technology for Data Mining

What is a data warehouse?
A multi-dimensional data model
Data warehouse architecture
Data warehouse implementation
Further development of data cube technology
From data warehousing to data mining

What is Data Warehouse?
Defined in many different ways, but not rigorously.
A decision support database that is maintained separately from the organization’s operational database
Support information processing by providing a solid platform of consolidated, historical data for analysis.
"A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management’s decision-making process."—W. H. Inmon
Data warehousing:
The process of constructing and using data warehouses

Data Warehouse—Subject-Oriented
Organized around major subjects, such as customer, product, sales.
Focusing on the modeling and analysis of data for decision makers, not on daily operations or transaction processing.
Provide a simple and concise view around particular subject issues by excluding data that are not useful in the decision support process.

Data Warehouse—Integrated
Constructed by integrating multiple, heterogeneous data sources
Relational databases, flat files, on-line transaction records
Data cleaning and data integration techniques are applied.
Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources
E.g., Hotel price: currency, tax, breakfast covered, etc.
When data is moved to the warehouse, it is converted.

Data Warehouse—Time Variant
The time horizon for the data warehouse is significantly longer than that of operational systems.
Operational database: current value data.
Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)
Every key structure in the data warehouse
Contains an element of time, explicitly or implicitly
But the key of operational data may or may not contain “time element.”
Data Warehouse—Non-Volatile

- A physically separate store of data transformed from the operational environment.
- Operational update of data does not occur in the data warehouse environment.
- Does not require transaction processing, recovery, and concurrency control mechanisms
- Requires only two operations in data accessing:
  - initial loading of data and access of data.

Data Warehouse vs. Heterogeneous DBMS

- Traditional heterogeneous DB integration:
  - Build wrappers/mediators on top of heterogeneous databases
  - Query driven approach
    - When a query is posed to a client site, a meta-dictionary is used to translate the query into queries appropriate for individual heterogeneous sites involved, and the results are integrated into a global answer set
    - Complex information filtering, compete for resources
- Data warehouse: update-driven, high performance
  - Information from heterogeneous sources is integrated in advance and stored in warehouses for direct query and analysis

Data Warehouse vs. Operational DBMS

- OLTP (on-line transaction processing)
  - Major task of traditional relational DBMS
  - Day-to-day operations: purchasing, inventory, banking, manufacturing, payroll, registration, accounting, etc.
- OLAP (on-line analytical processing)
  - Major task of data warehouse system
  - Data analysis and decision making
  - Distinct features (OLTP vs. OLAP):
    - User and system orientation: customer vs. market
    - Data contents: current, detailed vs. historical, consolidated
    - Database design: ER + application vs. star + subject
    - View: current, local vs. evolutionary, integrated
    - Access patterns: update vs. read-only but complex queries

Why Separate Data Warehouse?

- High performance for both systems
- DBMS—tuned for OLTP: access methods, indexing, concurrency control, recovery
- Warehouse—tuned for OLAP: complex OLAP queries, multidimensional view, consolidation.
- Different functions and different data:
  - missing data: Decision support requires historical data which operational DBs do not typically maintain
  - data consolidation: DS requires consolidation (aggregation, summarization) of data from heterogeneous sources
  - data quality: different sources typically use inconsistent data representations, codes and formats which have to be reconciled

Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining
From Tables and Spreadsheets to Data Cubes

- A data warehouse is based on a multidimensional data model which views data in the form of a data cube.
- A data cube, such as sales, allows data to be modeled and viewed in multiple dimensions.
- Dimension tables, such as item (item_name, brand, type), or time(day, week, month, quarter, year).
- Fact table contains measures (such as dollars_sold) and keys to each of the related dimension tables.
- In data warehousing literature, an n-D base cube is called a base cuboid. The top most 0-D cuboid, which holds the highest-level of summarization, is called the apex cuboid. The lattice of cuboids forms a data cube.

Cube: A Lattice of Cuboids

Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures.
  - Star schema: A fact table in the middle connected to a set of dimension tables.
  - Snowflake schema: A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake.
  - Fact constellations: Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation.

Example of Star Schema

Example of Snowflake Schema

Example of Fact Constellation
A Data Mining Query Language, DMQL: Language Primitives

- Cube Definition (Fact Table)
  define cube <cube_name> [ <dimension_list> ]: <measure_list>
- Dimension Definition (Dimension Table)
  define dimension <dimension_name> as (<attribute_or_subdimension_list>)
- Special Case (Shared Dimension Tables)
  First time as “cube definition”
  define dimension <dimension_name> as <dimension_name_first_time> in cube <cube_name_first_time>

Defining a Star Schema in DMQL

define cube sales_star [time, item, branch, location]:
dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)
define dimension time as (time_key, day, day_of_week, month, quarter, year)
define dimension item as (item_key, item_name, brand, type, supplier_type)
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as (location_key, street, city, province_or_state, country)

Defining a Snowflake Schema in DMQL

define cube sales_snowflake [time, item, branch, location]:
dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)
define dimension time as (time_key, day, day_of_week, month, quarter, year)
define dimension item as (item_key, item_name, brand, type, supplier(supplier_key, supplier_type))
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as (location_key, street, city(city_key, province_or_state, country))

Defining a Fact Constellation in DMQL

define cube sales [time, item, branch, location]:
dollars_sold = sum(sales_in_dollars), avg_sales = avg(sales_in_dollars), units_sold = count(*)
define dimension time as time in cube sales
define dimension item as item in cube sales
define dimension branch as (branch_key, branch_name, branch_type)
define dimension location as location in cube sales

define cube shipping [time, item, shipper, from_location, to_location]:
dollar_cost = sum(cost_in_dollars), unit_shipped = count(*)
define dimension time as time in cube sales
define dimension item as item in cube sales
define dimension shipper as (shipper_key, shipper_name, location as location in cube sales, shipper_type)
define dimension from_location as location in cube sales
define dimension to_location as location in cube sales

Measures: Three Categories

- distributive: if the result derived by applying the function to \( n \) aggregate values is the same as that derived by applying the function on all the data without partitioning.
  - E.g., \( \text{count}() \), \( \text{sum}() \), \( \text{min}() \), \( \text{max}() \).
- algebraic: if it can be computed by an algebraic function with \( M \) arguments (where \( M \) is a bounded integer), each of which is obtained by applying a distributive aggregate function.
  - E.g., \( \text{avg}() \), \( \text{min}() \), \( \text{stddev}() \).
- holistic: if there is no constant bound on the storage size needed to describe a subaggregate.
  - E.g., \( \text{median}() \), \( \text{mode}() \), \( \text{rank}() \).

Concept hierarchy

- What is a concept hierarchy?
  - A concept hierarchy defines a sequence of mappings from a set of low level concepts to higher level, more general concepts.
  - A CH that is a total or partial order among attributes in a database schema is called a schema hierarchy.
  - Concept hierarchies may also be defined by grouping (or discretizing) values for a given dimension or attribute, resulting in a set-grouping hierarchy.
A Concept Hierarchy: Dimension (location)

- all
- region (Europe, North America, Germany, Spain, Canada, Mexico)
- country (Frankfurt, Vancouver, Toronto, L. Chan, M. Wind)

View of Warehouses and Hierarchies

**Specification of hierarchies**
- Schema hierarchy
  - day < {month < quarter, week} < year
- Set_grouping hierarchy
  - {1..10} < inexpensive

Multidimensional Data

- Sales volume as a function of product, month, and region

**Dimensions:** Product, Location, Time

Hierarchical summarization paths

- Industry, Region, Year
- Category, Country, Quarter
- Product, City, Month, Week
- Office, Day

A Sample Data Cube

**Total annual sales of TV in U.S.A.**

<table>
<thead>
<tr>
<th>Product</th>
<th>Country</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>TV</td>
<td>U.S.A.</td>
<td>1Qtr</td>
</tr>
<tr>
<td>VCR</td>
<td>Canada</td>
<td>2Qtr</td>
</tr>
<tr>
<td>PC</td>
<td>Mexico</td>
<td>3Qtr</td>
</tr>
</tbody>
</table>

Cuboids Corresponding to the Cube

- 0-D (apex) cuboid
- 1-D cuboids
- 2-D cuboids
- 3-D (base) cuboid

Browsing a Data Cube

- Visualization
- OLAP capabilities
- Interactive manipulation
Typical OLAP Operations

- **Roll up (drill-up):** summarize data
  - by climbing up hierarchy or by dimension reduction
  - from higher level summary to lower level summary or detailed data, or introducing new dimensions
- **Drill down (roll down):** reverse of roll-up from higher level summary to lower level summary or detailed data, or introducing new dimensions
- **Slice and dice:**
  - project and select
  - **Pivot (rotate):**
  - reorient the cube, visualization, 3D to series of 2D planes.
- **Other operations**
  - **drill across:** involving (across) more than one fact table
  - **drill through:** through the bottom level of the cube to its back-end relational tables (using SQL)

Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
  - A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

Design of a Data Warehouse: A Business Analysis Framework

- Four views regarding the design of a data warehouse
  - **Top-down view**
    - allows selection of the relevant information necessary for the data warehouse
  - **Data source view**
    - exposes the information being captured, stored, and managed by operational systems
  - **Data warehouse view**
    - consists of fact tables and dimension tables
  - **Business query view**
    - sees the perspectives of data in the warehouse from the view of end-user

Data Warehouse Design Process

- Top-down, bottom-up approaches or a combination of both
  - **Top-down:** Starts with overall design and planning (mature)
  - **Bottom-up:** Starts with experiments and prototypes (rapid)
- From software engineering point of view
  - **Waterfall:** structured and systematic analysis at each step before proceeding to the next
  - **Spiral:** rapid generation of increasingly functional systems, short turn around time, quick turn around
- Typical data warehouse design process
  - Choose a business process to model, e.g., orders, invoices, etc.
  - Choose the **granularity** (atomic level of data) of the business process
  - Choose the **dimensions** that will apply to each fact table record
  - Choose the **measure** that will populate each fact table record

Multi-Tiered Architecture

- **Data Warehouse**
  - **OLAP Server**
  - **Metadata Management & Integration**
  - **Data Sources**
  - **Operational DBs**
  - **Extract, Transform, Load (ETL)**
  - **Data Marts**
  - **Analysis, Query, Reports, Data mining**
- **Front-End Tools**
  - **OLAP Engine**
  - **Data Storage**
  - **Other Sources**
Three Data Warehouse Models

- Enterprise warehouse
  - collects all of the information about subjects spanning the entire organization
- Data Mart
  - a subset of corporate-wide data that is of value to a specific group of users. Its scope is confined to specific, selected groups, such as marketing data mart
- Virtual warehouse
  - A set of views over operational databases
  - Only some of the possible summary views may be materialized

Data Warehouse Development: A Recommended Approach

Define a high-level corporate data model

OLAP Server Architectures

- Relational OLAP (ROLAP)
  - Use relational or extended-relational DBMS to store and manage warehouse data and OLAP middle ware to support missing pieces
  - Include optimization of DBMS backend, implementation of aggregation navigation logic, and additional tools and services
  - greater scalability
- Multidimensional OLAP (MOLAP)
  - Array-based multidimensional storage engine (sparse matrix techniques)
  - fast indexing to pre-computed summarized data
- Hybrid OLAP (HOLAP)
  - User flexibility, e.g., low level: relational, high-level: array
- Specialized SQL servers
  - specialized support for SQL queries over star/snowflake schemas

Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

Efficient cube computations

- OLAP servers demand high performance
- E.g. queries:
  - "compute the sum of sales, grouping by item and city"
  - "Retrieve images that have this color and texture"
  - "show sequences that are similar to this given sequence"

Lattice of Cuboids

How many cuboids in an n-dim data cube?
Efficient Data Cube Computation

- Data cube can be viewed as a lattice of cuboids
  - The bottom-most cuboid is the base cuboid
  - The top-most cuboid (apex) contains only one cell
  - How many cuboids in an n-dimensional cube with L levels?

Materialization of data cube

- Materialize every (full materialization), none (no materialization), or some (partial materialization)
- Selection of which cuboids to materialize
  - Based on size, sharing, access frequency, etc.

Cube Operation

- Cube definition and computation in DMQL
  
  define sales[item, city, year]: sum(sales_in_dollars)
  compute cube sales

- Transform it into a SQL-like language (with a new operator cube by, introduced by Gray et al.'96)
  
  SELECT item, city, year, SUM(amount)
  FROM SALES
  CUBE BY item, city, year

- Need compute the following Group-Bys
  
  (date, product, customer), (date, product), (date, customer), (product, customer), (date), (product), (customer)

Cube Computation: ROLAP-Based Method

- Efficient cube computation methods
  - ROLAP-based cubing algorithms (Agarwal et al.'96)
  - Array-based cubing algorithm (Zhao et al.'97)
  - Bottom-up computation method (Bayer & Ramakrishnan'99)

- ROLAP-based cubing algorithms
  - Sorting, hashing, and grouping operations are applied to the dimension attributes in order to reorder and cluster related tuples
  - Grouping is performed on some subaggregates as a "partial grouping step"
  - Aggregates may be computed from previously computed aggregates, rather than from the base fact table

Multi-way Array Aggregation for Cube Computation

- Partition arrays into chunks (a small subcube which fits in memory).
- Compressed sparse array addressing: (chunk_id, offset)
- Compute aggregates in “multway” by visiting cube cells in the order which minimizes the # of times to visit each cell, and reduces memory access and storage cost.

What is the best traversing order to do multi-way aggregation?
Multi-Way Array Aggregation for Cube Computation (Cont.)

- Method: the planes should be sorted and computed according to their size in ascending order.
  - See the details of Example 2.12 (pp. 75-78)
  - Idea: keep the smallest plane in the main memory, fetch and compute only one chunk at a time for the largest plane
- Limitation of the method: computing well only for a small number of dimensions
  - If there are a large number of dimensions, "bottom-up computation" and iceberg cube computation methods can be explored

Indexing OLAP Data: Bitmap Index

- Index on a particular column
  - Each value in the column has a bit vector: bit-op is fast
  - The length of the bit vector: # of records in the base table
- Traditional indices map the values to a list of record ids
  - It materializes relational join in JI file and speeds up relational join — a rather costly operation
  - In data warehouses, join index relates the values of the dimensions of a star schema to rows in the fact table.
  - E.g. fact table: Sales and two dimensions city and product
    - A join index on city maintains for each distinct city a list of R-IDs of the tuples recording the Sales in the city
  - Join indices can span multiple dimensions

Indexing OLAP Data: Join Indices

- Join index: JI(R-id, S-id) where R (R-id, ...) ⊆ S (S-id, ...)
  - Traditional indices map the values to a list of record ids
    - It materializes relational join in JI file and speeds up relational join — a rather costly operation
  - In data warehouses, join index relates the values of the dimensions of a star schema to rows in the fact table.
    - E.g. fact table: Sales and two dimensions city and product
      - A join index on city maintains for each distinct city a list of R-IDs of the tuples recording the Sales in the city
  - Join indices can span multiple dimensions

Data Warehouse Back-End Tools and Utilities

- Data extraction:
  - get data from multiple, heterogeneous, and external sources
- Data cleaning:
  - detect errors in the data and rectify them when possible
- Data transformation:
  - convert data from legacy or host format to warehouse format
- Load:
  - sort, summarize, consolidate, compute views, check integrity, and build indices and partitions
- Refresh
  - propagate the updates from the data sources to the warehouse
Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

Discovery-Driven Exploration of Data Cubes

- Hypothesis-driven: exploration by user, huge search space
- Discovery-driven (Sarawagi et al. ’98)
  - pre-compute measures indicating exceptions, guide user in the data analysis, at all levels of aggregation
  - Exception: significantly different from the value anticipated, based on a statistical model
  - Visual cues such as background color are used to reflect the degree of exception of each cell
  - Computation of exception indicator (modeling fitting and computing SelfExp, InExp, and PathExp values) can be overlapped with cube construction

Examples: Discovery-Driven Data Cubes

Complex Aggregation at Multiple Granularities: Multi-Feature Cubes

- Multi-feature cubes (Ross, et al. 1998): Compute complex queries involving multiple dependent aggregates at multiple granularities
  - Ex. Grouping by all subsets of (item, region, month), find the maximum price in 1997 for each group, and the total sales among all maximum price tuples
    
    ```sql
    select item, region, month, max(price), sum(R.sales)
    from purchases
    where year = 1997
    cube by item, region, month: R
    such that R.price = max(price)
    ```
  - Continuing the last example, among the max price tuples, find the min and max shelf life, and find the fraction of the total sales due to tuple that have min shelf life within the set of all max price tuples

Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

Data Warehouse Usage

- Three kinds of data warehouse applications
  - Information processing
    - supports querying, basic statistical analysis, and reporting using crosstabs, tables, charts and graphs
  - Analytical processing
    - multidimensional analysis of data warehouse data
    - supports basic OLAP operations, slice-dice, drilling, pivoting
  - Data mining
    - knowledge discovery from hidden patterns
    - supports associations, constructing analytical models, performing classification and prediction, and presenting the mining results using visualization tools.
  - Differences among the three tasks
From On-Line Analytical Processing to On Line Analytical Mining (OLAM)

- Why online analytical mining?
  - High quality of data in data warehouses
  - DW contains integrated, consistent, cleaned data
  - Available information processing structure surrounding data warehouses
  - ODBC, OLEDB, Web accessing, service facilities, reporting and OLAP tools
  - OLAP-based exploratory data analysis
  - mining with drilling, dicing, pivoting, etc.
  - On-line selection of data mining functions
  - integration and swapping of multiple mining functions, algorithms, and tasks.
- Architecture of OLAM

Summary

- Data warehouse
  - A subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process
  - A multi-dimensional model of a data warehouse
  - Star schema, snowflake schema, fact constellations
  - A data cube consists of dimensions & measures
  - OLAP operations: drilling, rolling, slicing, dicing and pivoting
  - OLAP servers: ROLAP, MOLAP, HOLAP
  - Efficient computation of data cubes
    - Partial vs. full vs. no materialization
    - Multiway array aggregation
    - Bitmap index and join index implementations
  - Further development of data cube technology
    - Discovery-driven and multi-feature cubes
    - From OLAP to OLAM (on-line analytical mining)

References (I)


References (II)


http://www.cs.sfu.ca/~han

Thank you !!!