Parallel Computer Centre

1 Data mining

1.1 What is data mining?

The past two decades has seen a dramatic increase in the amount of information or data being stored in electronic format. This accumulation of data has taken place at an explosive rate. It has been estimated that the amount of information in the world doubles every 20 months and the size and number of databases are increasing even faster. The increase in use of electronic data gathering devices such as point-of-sale or remote sensing devices has contributed to this explosion of available data. Figure 1 from the Red Brick company illustrates the data explosion.

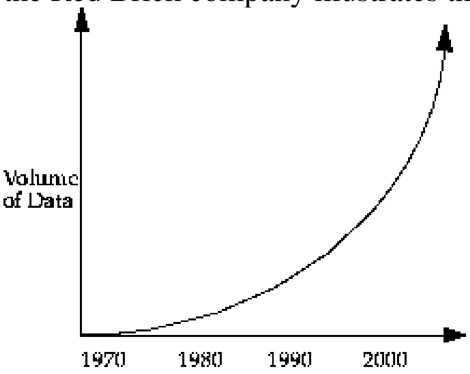


Figure 1: The Growing Base of Data

Data storage became easier as the availability of large amounts of computing power at low cost ie the cost of processing power and storage is falling, made data cheap. There was also the introduction of new machine learning methods for knowledge representation based on logic programming etc. in addition to traditional statistical analysis of data. The new methods tend to be computationally intensive hence a demand for more processing power.

Having concentrated so much attention on the accumulation of data the problem was what to do with this valuable resource? It was recognised that information is at the heart of business operations and that decision-makers could make use of the data stored to gain valuable insight into the business. Database Management systems gave access to the data stored but this was only a small part of what could be gained from the data. Traditional on-line transaction processing systems, OLTPs, are good at putting data into databases quickly, safely and efficiently but are not good at delivering meaningful analysis in return. Analysing data can provide further knowledge about a business by going beyond the data explicitly stored to derive knowledge about the business. This is where Data Mining or Knowledge Discovery in Databases (KDD) has obvious benefits for any enterprise.

The term data mining has been stretched beyond its limits to apply to any form of data analysis. Some of the numerous definitions of Data Mining, or Knowledge Discovery in Databases are:

> *Data Mining, or Knowledge Discovery in Databases (KDD) as it is also known, is the nontrivial extraction of implicit, previously unknown, and potentially useful information from data. This encompasses a number of different technical approaches, such as clustering, data summarization, learning classification rules, finding dependency net works, analysing changes, and detecting anomalies.*

William J Frawley, Gregory Piatetsky-Shapiro and Christopher J Matheus

> *Data mining is the search for relationships and global patterns that exist in large databases but are 'hidden' among the vast amount of data, such as a relationship between patient data and their medical diagnosis. These relationships represent valuable knowledge about the database and the objects in the database and, if the database is a faithful mirror, of the real world registered by the database.*

Marcel Holshemier & Arno Siebes (1994)

The analogy with the mining process is described as:

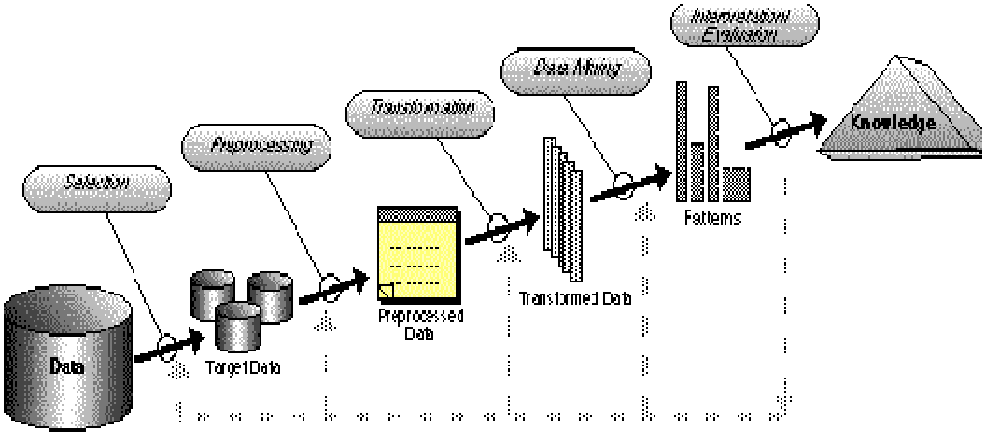
> *Data mining refers to "using a variety of techniques to identify nuggets of information or decision-making knowledge in bodies of data, and extracting these in such a way that they can be put to use in the areas such as decision support, prediction, forecasting and estimation. The data is often voluminous, but as it stands of low value as no direct use can be made of it; it is the hidden information in the data that is useful"*

Clementine User Guide, a data mining toolkit

Basically data mining is concerned with the analysis of data and the use of software techniques for finding patterns and regularities in sets of data. It is the computer which is responsible for finding the patterns by identifying the underlying rules and features in the data. The idea is that it is possible to strike gold in unexpected places as the data mining software extracts patterns not previously discernable or so obvious that no-one has noticed them before.

Data mining analysis tends to work from the data up and the best techniques are those developed with an orientation towards large volumes of data, making use of as much of the collected data as possible to arrive at reliable conclusions and decisions. The analysis process starts with a set of data, uses a methodology to develop an optimal representation of the structure of the data during which time knowledge is acquired. Once knowledge has been acquired this can be extended to larger sets of data working on the assumption that the larger data set has a structure similar to the sample data. Again this is analogous to a mining operation where large amounts of low grade materials are sifted through in order to find something of value.

The following diagram summarises the some of the stages/processes identified in data mining and knowledge discovery by Usama Fayyad & Evangelos Simoudis, two of leading exponents of this area.



The phases depicted start with the raw data and finish with the extracted knowledge which was acquired as a result of the following stages:

- Selection - selecting or segmenting the data according to some criteria e.g. all those people who own a car, in this way subsets of the data can be determined.

- Preprocessing - this is the data cleansing stage where certain information is removed which is deemed unnecessary and may slow down queries for example unnecessary to note the sex of a patient when studying pregnancy. Also the data is reconfigured to ensure a consistent format as there is a possibility of inconsistent formats because the data is drawn from several sources e.g. sex may recorded as f or m and also as 1 or 0.

- Transformation - the data is not merely transferred across but transformed in that overlays may added such as the demographic overlays commonly used in market research. The data is made useable and navigable.

- Data mining - this stage is concerned with the extraction of patterns from the data. A pattern can be defined as given a set of facts(data) *F*, a language *L*, and some measure of certainty *C* a pattern is a statement *S* in *L* that describes relationships among a subset *Fs* of *F* with a certainty *c* such that *S* is simpler in some sense than the enumeration of all the facts in *Fs*.

- Interpretation and evaluation - the patterns identified by the system are interpreted into knowledge which can then be used to support human decision-making e.g. prediction and classification tasks, summarizing the contents of a database or explaining observed phenomena.

1.2 Data mining background

Data mining research has drawn on a number of other fields such as inductive learning, machine learning and statistics etc.

1.2.1 Inductive learning

Induction is the inference of information from data and inductive learning is the model building process where the environment i.e. database is analysed with a view to finding patterns. Similar objects are grouped in classes and rules formulated whereby it is possible to predict the class of unseen objects. This process of classification identifies classes such that each class has a unique pattern of values which forms the class description. The nature of the environment is dynamic hence the model must be adaptive i.e. should be able learn.

Generally it is only possible to use a small number of properties to characterise objects so we make abstractions in that objects which satisfy the same subset of properties are mapped to the same internal representation.

Inductive learning where the system infers knowledge itself from observing its environment has two main strategies:

- supervised learning - this is learning from examples where a teacher helps the system construct a model by defining classes and supplying examples of each class. The system has to find a description of each class i.e. the common properties in the examples. Once the description has been formulated the description and the class form a classification rule which can be used to predict the class of previously unseen objects. This is similar to discriminate analysis as in statistics.

- unsupervised learning - this is learning from observation and discovery. The data mine system is supplied with objects but no classes are defined so it has to observe the examples and recognise patterns (i.e. class description) by itself. This system results in a set of class descriptions, one for each class discovered in the environment. Again this similar to cluster analysis as in statistics.

Induction is therefore the extraction of patterns. The quality of the model produced by inductive learning methods is such that the model could be used to predict the outcome of future situations in other words not only for states encountered but rather for unseen states that could occur. The problem is that most environments have different states, i.e. changes within, and it is not always possible to verify a model by checking it for all possible situations.

Given a set of examples the system can construct multiple models some of which will be simpler than others. The simpler models are more likely to be correct if we adhere to Ockhams razor, which states that if there are multiple explanations for a particular phenomena it makes sense to choose the simplest because it is more likely to capture the nature of the phenomenon.

1.2.2 Statistics

Statistics has a solid theoretical foundation but the results from statistics can be overwhelming and difficult to interpret as they require user guidance as to where and how to analyse the data. Data mining however allows the expert's knowledge of the data and the advanced analysis techniques of the computer to work together.

Statistical analysis systems such as SAS and SPSS have been used by analysts to detect unusual patterns and explain patterns using statistical models such as linear models. Statistics have a role to play and data mining will not replace such analyses but rather they can act upon more directed analyses based on the results of data mining. For example statistical induction is something like the average rate of failure of machines.

1.2.3 Machine Learning

Machine learning is the automation of a learning process and learning is tantamount to the construction of rules based on observations of environmental states and transitions. This is a broad field which includes not only learning from examples, but also reinforcement learning, learning with teacher, etc. A learning algorithm takes the data set and its accompanying information as input and returns a statement e.g. a concept representing the results of learning as output. Machine learning examines previous examples and their outcomes and learns how to reproduce these and make generalisations about new cases.

Generally a machine learning system does not use single observations of its environment but an entire finite set called the training set at once. This set contains examples i.e. observations coded in some machine readable form. The training set is finite hence not all concepts can be learned exactly.

1.2.4 Differences between Data Mining and Machine Learning

Knowledge Discovery in Databases (KDD) or Data Mining, and the part of Machine Learning (ML) dealing with learning from examples overlap in the algorithms used and the problems addressed.

The main differences are:

- KDD is concerned with finding understandable knowledge, while ML is concerned with improving performance of an agent. So training a neural network to balance a pole is part of ML, but not of KDD. However, there are efforts to extract knowledge from neural networks which are very relevant for KDD.

- KDD is concerned with very large, real-world databases, while ML typically (but not always) looks at smaller data sets. So efficiency questions are much more important for KDD.

- ML is a broader field which includes not only learning from examples, but also reinforcement learning, learning with teacher, etc.

KDD is that part of ML which is concerned with finding understandable knowledge in large sets of real-world examples. When integrating machine learning techniques into database systems to implement KDD some of the databases require:

- more efficient learning algorithms because realistic databases are normally very large and noisy. It is usual that the database is often designed for purposes different from data mining and so properties or attributes that would simplify the learning task are not present nor can they be requested from the real world. Databases are usually contaminated by errors so the data mining algorithm has to cope with noise whereas ML has laboratory type examples i.e. as near perfect as possible.

- more expressive representations for both data, e.g. tuples in relational databases, which represent instances of a problem domain, and knowledge, e.g. rules in a rule-based system, which can be used to solve users' problems in the domain, and the semantic information contained in the relational schemata.

Practical KDD systems are expected to include three interconnected phases

- Translation of standard database information into a form suitable for use by learning facilities;
- Using machine learning techniques to produce knowledge bases from databases; and
- Interpreting the knowledge produced to solve users' problems and/or reduce data spaces. Data spaces being the number of examples.

1.3 Data Mining Models

IBM have identified two types of model or modes of operation which may be used to unearth information of interest to the user.

1.3.1 Verification Model

The verification model takes an hypothesis from the user and tests the validity of it against the data. The emphasis is with the user who is responsible for formulating the hypothesis and issuing the query on the data to affirm or negate the hypothesis.

In a marketing division for example with a limited budget for a mailing campaign to launch a new product it is important to identify the section of the population most likely to buy the new product. The user formulates an hypothesis to identify potential customers and the characteristics they share. Historical data about customer purchase and demographic information can then be queried to reveal comparable purchases and the characteristics shared by those purchasers which in turn can be used to target a mailing campaign. The whole operation can be refined by 'drilling down' so that the hypothesis reduces the 'set' returned each time until the required limit is reached.

The problem with this model is the fact that no new information is created in the retrieval process but rather the queries will always return records to verify or negate the hypothesis. The search process here is iterative in that the output is reviewed, a new set of questions or hypothesis formulated to refine the search and the whole process repeated. The user is discovering the facts about the data using a variety of techniques such as queries, multidimensional analysis and visualization to guide the exploration of the data being inspected.

1.3.2 Discovery Model

The discovery model differs in its emphasis in that it is the system automatically discovering important information hidden in the data. The data is sifted in search of frequently occurring patterns, trends and generalisations about the data without intervention or guidance from the user. The discovery or data mining tools aim to reveal a large number of facts about the data in as short a time as possible.

An example of such a model is a bank database which is mined to discover the many groups of customers to target for a mailing campaign. The data is searched with no hypothesis in mind other than for the system to group the customers according to the common characteristics found.

1.4 Data Warehousing

Data mining potential can be enhanced if the appropriate data has been collected and stored in a data warehouse. A data warehouse is a relational database management system (RDMS) designed specifically to meet the needs of transaction processing systems. It can be loosely defined as any centralised data repository which can be queried for business benefit but this will be more clearly defined later. Data warehousing is a new powerful technique making it possible to extract archived operational data and overcome inconsistencies between different legacy data formats. As well as integrating data throughout an enterprise, regardless of location, format, or communication requirements it is possible to incorporate additional or expert information. It is,

> *the logical link between what the managers see in their decision support EIS applications and the company's operational activities*

John McIntyre of SAS Institute Inc

In other words the data warehouse provides data that is already transformed and summarized, therefore making it an appropriate environment for more efficient DSS and EIS applications.

1.4.1 Characteristics of a data warehouse

According to Bill Inmon, author of Building the Data Warehouse and the guru who is widely considered to be the originator of the data warehousing concept, there are generally four characteristics that describe a data warehouse:

- subject-oriented: data are organized according to subject instead of application e.g. an insurance company using a data warehouse would organize their data by customer, premium, and claim, instead of by different products (auto, life, etc.). The data organized by subject contain only the information necessary for decision support processing.

- integrated: When data resides in many separate applications in the operational environment, encoding of data is often inconsistent. For instance, in one application, gender might be coded as "m" and "f" in another by 0 and 1. When data are moved from the operational environment into the data warehouse, they assume a consistent coding convention e.g. gender data is transformed to "m" and "f".

- time-variant: The data warehouse contains a place for storing data that are five to 10 years old, or older, to be used for comparisons, trends, and forecasting. These data are not updated.

- non-volatile: Data are not updated or changed in any way once they enter the data warehouse, but are only loaded and accessed.

1.4.2 Processes in data warehousing

The first phase in data warehousing is to "insulate" your current operational information, ie to preserve the security and integrity of mission-critical OLTP applications, while giving you access to the broadest possible base of data. The resulting database or data warehouse may consume hundreds of gigabytes - or even terabytes - of disk space, what is required then are efficient techniques for storing and retrieving massive amounts of information. Increasingly, large organizations have found that only parallel processing systems offer sufficient bandwidth.

The data warehouse thus retrieves data from a variety of heterogeneous operational databases. The data is then transformed and delivered to the data warehouse/store based on a selected model (or mapping definition). The data transformation and movement processes are executed whenever an update to the warehouse data is required so there should some form of automation to manage and execute these functions. The information that describes the model and definition of the source data elements is called "metadata". The metadata is the means by which the end-user finds and understands the data in the warehouse and is an important part of the warehouse. The metadata should at the very least contain;

- the structure of the data;
- the algorithm used for summarization;
- and the mapping from the operational environment to the data warehouse.

Data cleansing is an important aspect of creating an efficient data warehouse in that it is the removal of certain aspects of operational data, such as low-level transaction information, which slow down the query times. The cleansing stage has to be as dynamic as possible to accommodate all types of queries even those which may require low-level information. Data should be extracted from production sources at regular intervals and pooled centrally but the cleansing process has to remove duplication and reconcile differences between various styles of data collection.

Once the data has been cleaned it is then transferred to the data warehouse which typically is a large database on a high performance box either SMP, Symmetric Multi-Processing or MPP, Massively Parallel Processing. Number-crunching power is another important aspect of data warehousing because of the complexity involved in processing ad hoc queries and because of the vast quantities of data that the organisation want to use in the warehouse. A data warehouse can be used in different ways for example it can be used as a central store against which the queries are run or it can be used to like a data mart. Data marts which are small warehouses can be established to provide subsets of the main store and summarised information depending on the requirements of a specific group/department. The central store approach generally uses very simple data structures with very little assumptions about the relationships between data whereas marts often use multidimensional databases which can speed up query processing as they can have data structures which are reflect the most likely questions.

Many vendors have products that provide one or more of the above described data warehouse functions. However, it can take a significant amount of work and specialized programming to provide the interoperability needed between products from multiple vendors to enable them to perform the required data warehouse processes. A typical implementation usually involves a mixture of products from a variety of suppliers.

Another approach to data warehousing is Parsaye's Sandwich Paradigm put forward by Dr. Kamran Parsaye, CEO of Information Discovery, Hermosa Beach, CA. This paradigm or philosophy encourages acceptance of the probability that the first iteration of a data-warehousing effort will require considerable revision. The Sandwich Paradigm advocates the following approach:

- pre-mine the data to determine what formats and data are needed to support a data-mining application;

- build a prototype mini-data warehouse i.e the meat of the sandwich, with most of the features envisaged for the end product;

- revise the strategies as necessary;

- build the final warehouse.

1.4.3 Data warehousing and OLTP systems

A database which is built for on line transaction processing, OLTP, is generally regarded as unsuitable for data warehousing as they have been designed with a different set of needs in mind ie maximising transaction capacity and typically having hundreds of tables in order not to lock out users etc. Data warehouses are interested in query processing as opposed to transaction processing.

OLTP systems cannot be repositories of facts and historical data for business analysis. They cannot quickly answer ad hoc queries and rapid retrieval is almost impossible. The data is inconsistent and changing, duplicate entries exist, entries can be missing and there is an absence of historical data which is necessary to analyse trends. Basically OLTP offers large amounts of raw data which is not easily understood. The data warehouse offers the potential to retrieve and analyse information quickly and easily. Data warehouses do have similarities with OLTP as shown in the table below.

Table 1: Similarities and Differences in OLAP and data warehousing

	OLTP	Data Warehouse
Purpose	Run day-to-day operations	Information retrieval and analysis
Structure	RDBMS	RDBMS
Data Model	Normalised	Multi-dimensional
Access	SQL	SQL plus data analysis extensions
Type of Data	Data that runs the business	Data that analyses the business
Condition of Data	Changing, incomplete	Historical, descriptive

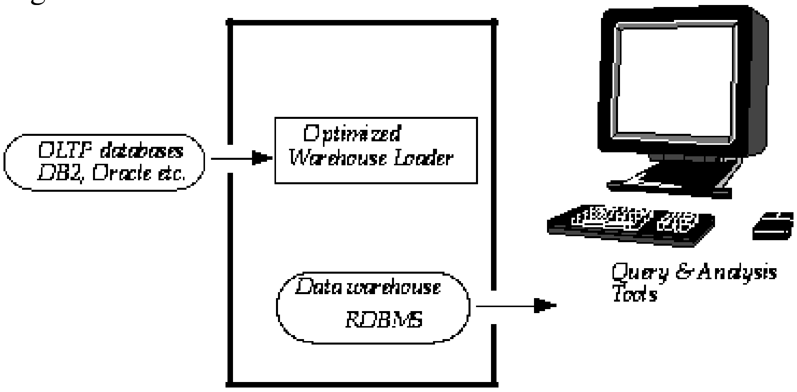
The data warehouse serves a different purpose from that of OLTP systems by allowing business analysis queries to be answered as opposed to "simple aggregations" such as ‘what is the current account balance for this customer?’ Typical data warehouse queries include such things as ‘which product line sells best in middle-America and how does this correlate to demographic data?’

1.4.4 The Data Warehouse model

Data warehousing is the process of extracting and transforming operational data into informational data and loading it into a central data store or warehouse. Once the data is loaded it is accessible via desktop query and analysis tools by the decision makers.

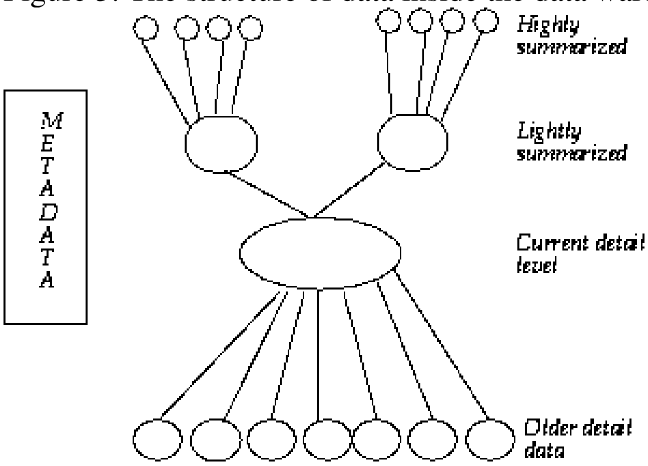
The data warehouse model is illustrated in the following diagram.

Figure 2: A data warehouse model



The data within the actual warehouse itself has a distinct structure with the emphasis on different levels of summarization as shown in the figure below.

Figure 3: The structure of data inside the data warehouse



The current detail data is central in importance as it:

- reflects the most recent happenings, which are usually the most interesting;
- it is voluminous as it is stored at the lowest level of granularity;
- it is always (almost) stored on disk storage which is fast to access but expensive and complex to manage.

Older detail data is stored on some form of mass storage, it is infrequently accessed and stored at a level detail consistent with current detailed data.

Lightly summarized data is data distilled from the low level of detail found at the current detailed level and generally is stored on disk storage. When building the data warehouse have to consider what unit of time is summarization done over and also the contents or what attributes the summarized data will contain.

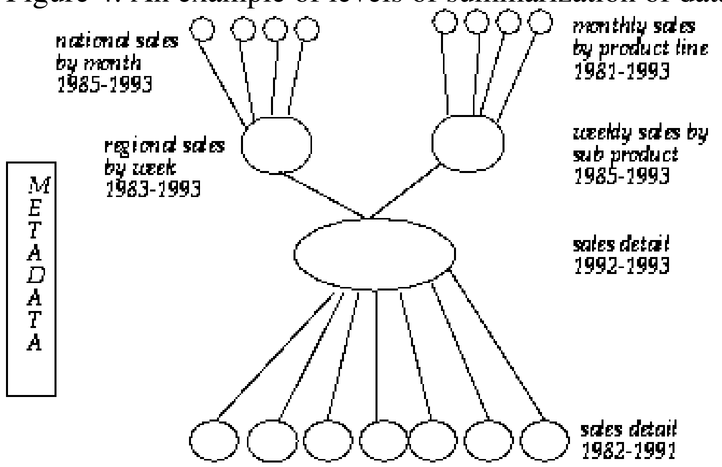
Highly summarized data is compact and easily accessible and can even be found outside the warehouse.

Metadata is the final component of the data warehouse and is really of a different dimension in that it is not the same as data drawn from the operational environment but is used as:

- a directory to help the DSS analyst locate the contents of the data warehouse,
- a guide to the mapping of data as the data is transformed from the operational environment to the data warehouse environment,
- a guide to the algorithms used for summarization between the current detailed data and the lightly summarized data and the lightly summarized data and the highly summarized data, etc.

The basic structure has been described but Bill Inmon fills in the details to make the example come alive as shown in the following diagram.

Figure 4: An example of levels of summarization of data inside the data warehouse



The diagram assumes the year is 1993 hence the current detail data is 1992-93. Generally sales data doesn’t reach the current level of detail for 24 hours as it waits until it is no longer available to the operational system i.e. it takes 24 hours for it to get to the data warehouse. Sales details are summarized weekly by subproduct and region to produce the lightly summarized detail. Weekly sales are then summarized again to produce the highly summarized data.

1.4.5 Problems with data warehousing

One of the problems with data mining software has been the rush of companies to jump on the band wagon as

> *these companies have slapped ‘data warehouse’ labels on traditional transaction-processing products, and co-opted the lexicon of the industry in order to be considered players in this fast-growing category.*

Chris Erickson, president and CEO of Red Brick (HPCwire, Oct. 13, 1995)

Red Brick Systems have established a criteria for a relational database management system (RDBMS) suitable for data warehousing, and documented 10 specialized requirements for an RDBMS to qualify as a relational data warehouse server, this criteria is listed in the next section.

According to Red Brick, the requirements for data warehouse RDBMSs begin with the loading and preparation of data for query and analysis. If a product fails to meet the criteria at this stage, the rest of the system will be inaccurate, unreliable and unavailable.

1.4.6 Criteria for a data warehouse

The criteria for data warehouse RDBMSs are as follows:

- Load Performance - Data warehouses require incremental loading of new data on a periodic basis within narrow time windows; performance of the load process should be measured in hundreds of millions of rows and gigabytes per hour and must not artificially constrain the volume of data required by the business.

- Load Processing - Many steps must be taken to load new or updated data into the data warehouse including data conversions, filtering, reformatting, integrity checks, physical storage, indexing, and metadata update. These steps must be executed as a single, seamless unit of work.

- Data Quality Management - The shift to fact-based management demands the highest data quality. The warehouse must ensure local consistency, global consistency, and referential integrity despite "dirty" sources and massive database size. While loading and preparation are necessary steps, they are not sufficient. Query throughput is the measure of success for a data warehouse application. As more questions are answered, analysts are catalysed to ask more creative and insightful questions.

- Query Performance - Fact-based management and ad-hoc analysis must not be slowed or inhibited by the performance of the data warehouse RDBMS; large, complex queries for key business operations must complete in seconds not days.

- Terabyte Scalability - Data warehouse sizes are growing at astonishing rates. Today these range from a few to hundreds of gigabytes, and terabyte-sized data warehouses are a near-term reality. The RDBMS must not have any architectural limitations. It must support modular and parallel management. It must support continued availability in the event of a point failure, and must provide a fundamentally different mechanism for recovery. It must support near-line mass storage devices such as optical disk and Hierarchical Storage Management devices. Lastly, query performance must not be dependent on the size of the database, but rather on the complexity of the query.

- Mass User Scalability - Access to warehouse data must no longer be limited to the elite few. The RDBMS server must support hundreds, even thousands, of concurrent users while maintaining acceptable query performance.

- Networked Data Warehouse - Data warehouses rarely exist in isolation. Multiple data warehouse systems cooperate in a larger network of data warehouses. The server must include tools that coordinate the movement of subsets of data between warehouses. Users must be able to look at and work with multiple warehouses from a single client workstation. Warehouse managers have to manage and administer a network of warehouses from a single physical location.

- Warehouse Administration - The very large scale and time-cyclic nature of the data warehouse demands administrative ease and flexibility. The RDBMS must provide controls for implementing resource limits, chargeback accounting to allocate costs back to users, and query prioritization to address the needs of different user classes and activities. The RDBMS must also provide for workload tracking and tuning so system resources may be optimized for maximum performance and throughput. "The most visible and measurable value of implementing a data warehouse is evidenced in the uninhibited, creative access to data it provides the end user.

- Integrated Dimensional Analysis - The power of multidimensional views is widely accepted, and dimensional support must be inherent in the warehouse RDBMS to provide the highest performance for relational OLAP tools. The RDBMS must support fast, easy creation of precomputed summaries common in large data warehouses. It also should provide the maintenance tools to automate the creation of these precomputed aggregates. Dynamic calculation of aggregates should be consistent with the interactive performance needs.

- Advanced Query Functionality - End users require advanced analytic calculations, sequential and comparative analysis, and consistent access to detailed and summarized data. Using SQL in a client/server point-and-click tool environment may sometimes be impractical or even impossible. The RDBMS must provide a complete set of analytic operations including core sequential and statistical operations.

1.5 Data mining problems/issues

Data mining systems rely on databases to supply the raw data for input and this raises problems in that databases tend be dynamic, incomplete, noisy, and large. Other problems arise as a result of the adequacy and relevance of the information stored.

1.5.1 Limited Information

A database is often designed for purposes different from data mining and sometimes the properties or attributes that would simplify the learning task are not present nor can they be requested from the real world. Inconclusive data causes problems because if some attributes essential to knowledge about the application domain are not present in the data it may be impossible to discover significant knowledge about a given domain. For example cannot diagnose malaria from a patient database if that database does not contain the patients red blood cell count.

1.5.2 Noise and missing values

Databases are usually contaminated by errors so it cannot be assumed that the data they contain is entirely correct. Attributes which rely on subjective or measurement judgements can give rise to errors such that some examples may even be mis-classified. Error in either the values of attributes or class information are known as noise. Obviously where possible it is desirable to eliminate noise from the classification information as this affects the overall accuracy of the generated rules.

Missing data can be treated by discovery systems in a number of ways such as;

- simply disregard missing values
- omit the corresponding records
- infer missing values from known values
- treat missing data as a special value to be included additionally in the attribute domain
- or average over the missing values using Bayesian techniques.

Noisy data in the sense of being imprecise is characteristic of all data collection and typically fit a regular statistical distribution such as Gaussian while wrong values are data entry errors. Statistical methods can treat problems of noisy data, and separate different types of noise.

1.5.3 Uncertainty

Uncertainty refers to the severity of the error and the degree of noise in the data. Data precision is an important consideration in a discovery system.

1.5.4 Size, updates, and irrelevant fields

Databases tend to be large and dynamic in that their contents are ever-changing as information is added, modified or removed. The problem with this from the data mining perspective is how to ensure that the rules are up-to-date and consistent with the most current information. Also the learning system has to be time-sensitive as some data values vary over time and the discovery system is affected by the 'timeliness' of the data.

Another issue is the relevance or irrelevance of the fields in the database to the current focus of discovery for example post codes are fundamental to any studies trying to establish a geographical connection to an item of interest such as the sales of a product.

1.6 Potential Applications

Data mining has many and varied fields of application some of which are listed below.

1.6.1 Retail/Marketing

- Identify buying patterns from customers
- Find associations among customer demographic characteristics
- Predict response to mailing campaigns
- Market basket analysis

1.6.2 Banking

- Detect patterns of fraudulent credit card use

- Identify ‘loyal’ customers
- Predict customers likely to change their credit card affiliation
- Determine credit card spending by customer groups
- Find hidden correlations between different financial indicators
- Identify stock trading rules from historical market data

1.6.3 Insurance and Health Care

- Claims analysis - i.e which medical procedures are claimed together
- Predict which customers will buy new policies
- Identify behaviour patterns of risky customers
- Identify fraudulent behaviour

1.6.4 Transportation

- Determine the distribution schedules among outlets
- Analyse loading patterns

1.6.5 Medicine

- Characterise patient behaviour to predict office visits
- Identify successful medical therapies for different illnesses

---

Maintained by Alan Rea, email A.Rea@qub.ac.uk
Generated with CERN WebMaker