1 (a) Explain the Multitier structure of Data Warehouse with neat diagram

Diagram -4 Marks

Explanation- 2 Marks

1 (b) What are the features of data warehouse?

Generally a data warehouses adopts a three-tier architecture. Following are the three tiers of the data warehouse architecture.

·**Bottom Tier** − The bottom tier of the architecture is the data warehouse database server. It is the relational database system. We use the back end tools and utilities to feed data into the bottom tier. These back end tools and utilities perform the Extract, Clean, Load, and refresh functions.

·**Middle Tier** − In the middle tier, we have the OLAP Server that can be implemented in either of the following ways.

By Relational OLAP (ROLAP), which is an extended relational database management system. The ROLAP maps the operations on multidimensional data to standard relational operations.

By Multidimensional OLAP (MOLAP) model, which directly implements the multidimensional data and operations.

Top-Tier − This tier is the front-end client layer. This layer holds the query tools and reporting tools, analysis tools and data mining tools

1 b) What are the features of data warehouse?

1*4 Features

A data warehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision-making process.

Subject-Oriented: A data warehouse can be used to analyze a particular subject area. For example, "sales" can be a particular subject.

Integrated: A data warehouse integrates data from multiple data sources. For example, source A and source B may have different ways of identifying a product, but in a data warehouse, there will be only a single way of identifying a product.

Time-Variant: Historical data is kept in a data warehouse. For example, one can retrieve data from 3 months, 6 months, 12 months, or even older data from a data warehouse. This contrasts

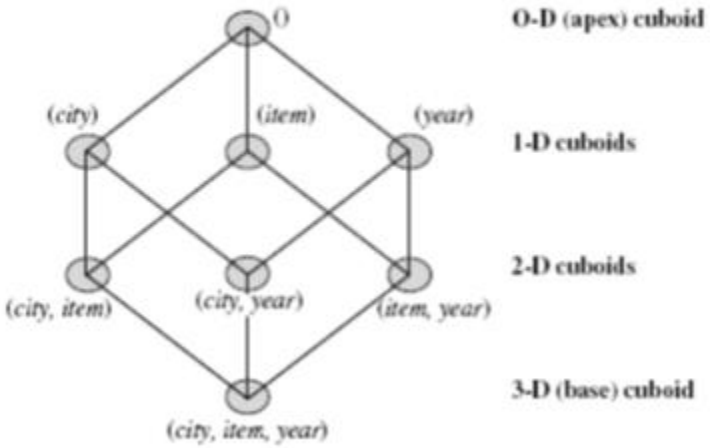
with a transactions system, where often only the most recent data is kept. For example, a transaction system may hold the most recent address of a customer, where a data warehouse can hold all addresses associated with a customer.

Non-volatile: Once data is in the data warehouse, it will not change. So, historical data in a data warehouse should never be altered.

2 a) Why multidimensional views of data and data cubes are used? With neat diagram explain data cube implementation

Diagram -2 Marks

Explanation- 4 Marks



At the core of multidimensional data analysis is the efficient computation of aggregations across many sets of dimensions. In SQL terms, these aggregations are referred to as group-by's. Each group-by can be represented by a *cuboid*, where the set of group-by's forms a lattice of cuboids defining a data cube. In this subsection, we explore issues relating to the efficient computation of data cubes

The compute cube Operator and the Curse of Dimensionality

One approach to cube computation extends SQL so as to include a compute cube operator. The compute cube operator computes aggregates over all subsets of the dimensions specified in the operation. This can require excessive storage space, especially for large numbers of dimensions. We start with an intuitive look at what is involved in the efficient computation of data cubes.

The base cuboid contains all three dimensions, *city*, *item*, and *year*. It can return the total sales for any combination of the three dimensions. The apex cuboid, or 0-D cuboid, refers to the case where the group-by is empty. It contains the total sum of all sales. The base cuboid is the least generalized (most specific) of the cuboids. The apex cuboid is the most generalized (least specific) of the cuboids, and is often denoted as all. If we start at the apex cuboid and explore downward in the lattice, this is equivalent to

drilling down within the data cube. If we start at the base cuboid and explore upward, this is akin to rolling up.

Similar to the SQL syntax, the data cube above could be defined as

define cube sales cube [city, item, year]: sum(sales in dollars) .

For a cube with *n* dimensions, there are a total of 2*n* cuboids, including the base cuboid. A statement such as

compute cube sales cube

Online analytical processing may need to access different cuboids for different queries. Therefore, it may seem like a good idea to compute in advance all or at least some of the cuboids in a data cube Precomputation leads to fast response time and avoids some redundant computation. Most, if not all, OLAP products resort to some degree of precomputation of multidimensional aggregates.

Partial Materialization: Selected Computation of Cuboids

There are three choices for data cube materialization given a base cuboid:

1. No materialization: Do not precompute any of the “nonbase” cuboids. This leads to computing expensive multidimensional aggregates on-the-fly, which can be extremely slow.

2. Full materialization: Precompute all of the cuboids. The resulting lattice of computed cuboids is referred to as the *full cube*. This choice typically requires huge amounts of memory space in order to store all of the precomputed cuboids.

3. Partial materialization: Selectively compute a proper subset of the whole set of possible cuboids. Alternatively, we may compute a subset of the cube, which contains only those cells that satisfy some user-specified criterion, such as where the tuple count of each cell is above some threshold. We will use the term *subcube* to refer to the latter case, where only some of the cells may be precomputed for various cuboids. Partial materialization represents an interesting trade-off between storage space and response time.

2 b)What is ETL? List the steps of ETL process

Definition-1 mark

Explanation-3 marks

Extraction, Transformation, and Loading

Data warehouse systems use back-end tools and utilities to populate and refresh their data These tools and utilities include the following functions:

Data extraction, which typically gathers data from multiple, heterogeneous, and external sources.

Data cleaning, which detects errors in the data and rectifies them when possible .

Data transformation, which converts data from legacy or host format to warehouse format.

Load, which sorts, summarizes, consolidates, computes views, checks integrity, and builds indices and partitions.

Refresh, which propagates the updates from the data sources to the warehouse. Besides cleaning, loading, refreshing, and metadata definition tools, data warehouse systems usually provide a good set of data warehouse management tools. Data cleaning and data transformation are important steps in improving the data quality and, subsequently, the data mining results Because we are mostly interested in the aspects of data warehousing technology related to data mining.

3 a) Explain different types of schemas with example for each. Generate a star schema for university with necessary dimensions

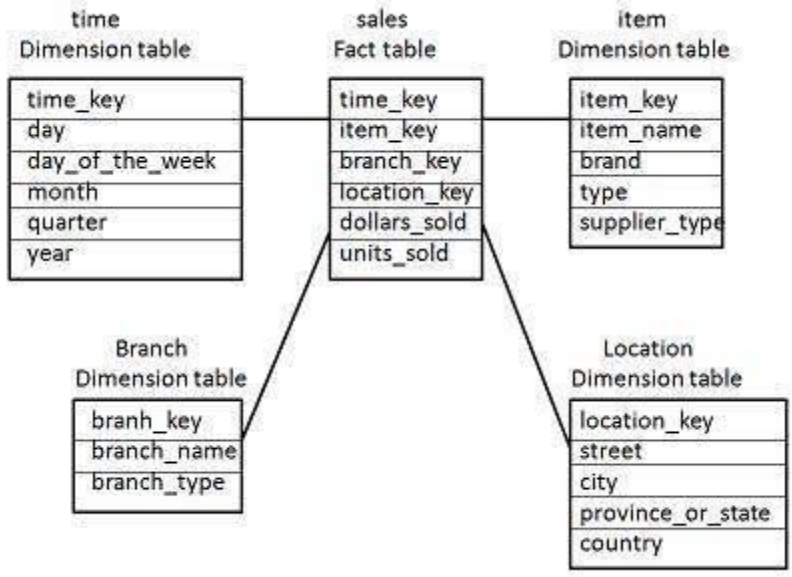
Each schema with example:2*3=6 marks

University schema:4 marks

Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema. In this chapter, we will discuss the schemas used in a data warehouse.

Star Schema

- Each dimension in a star schema is represented with only one-dimension table.
- This dimension table contains the set of attributes.
- The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.



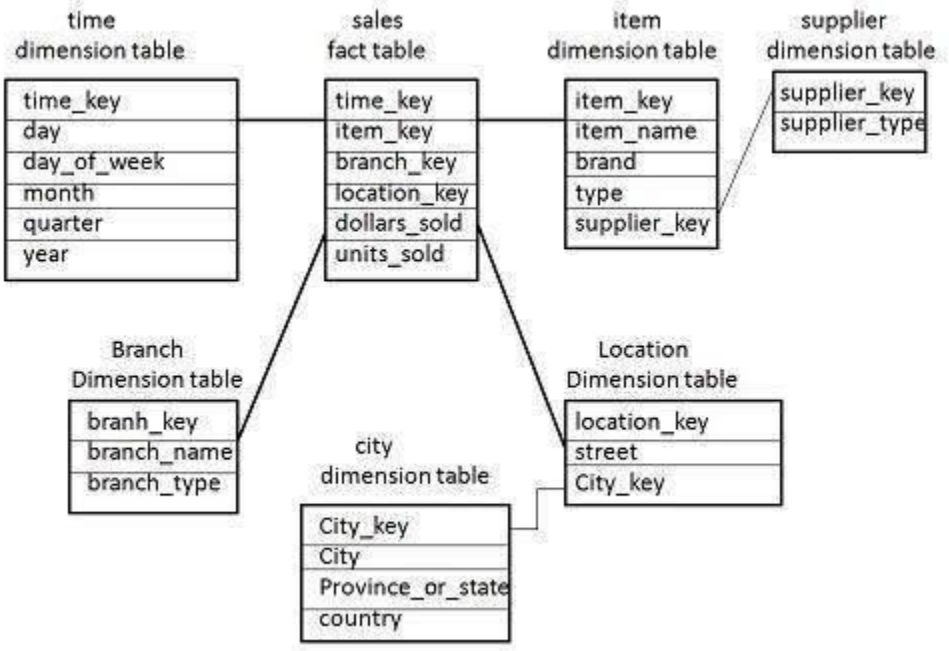
- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.

Note − Each dimension has only one dimension table and each table holds a set of attributes. For example, the location dimension table contains the attribute set {location_key, street, city, province_or_state,country}. This constraint may cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province_or_state and country.

Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.

- Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.

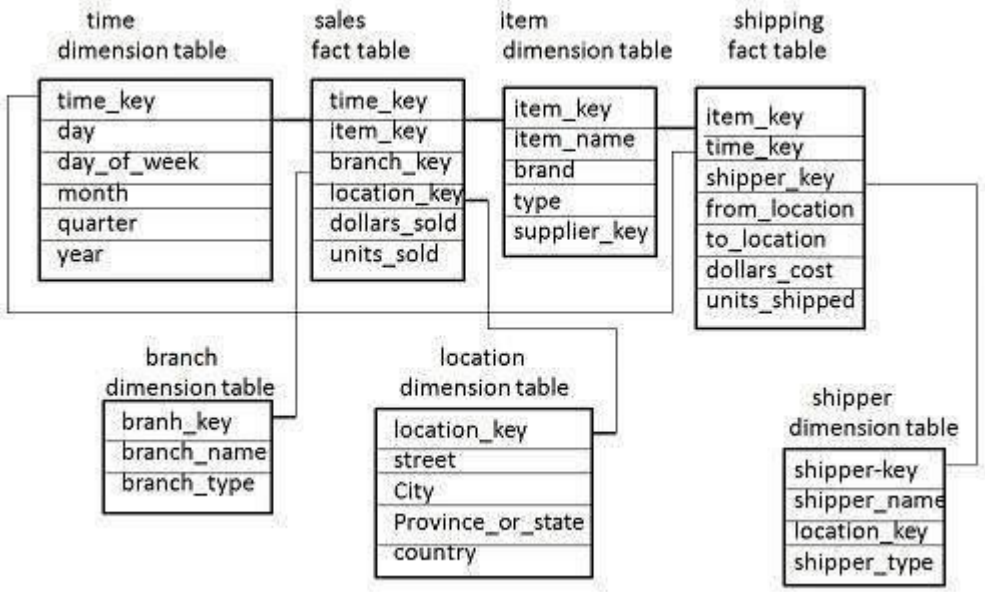


- Now the item dimension table contains the attributes item_key, item_name, type, brand, and supplier-key.
- The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier_key and supplier_type.

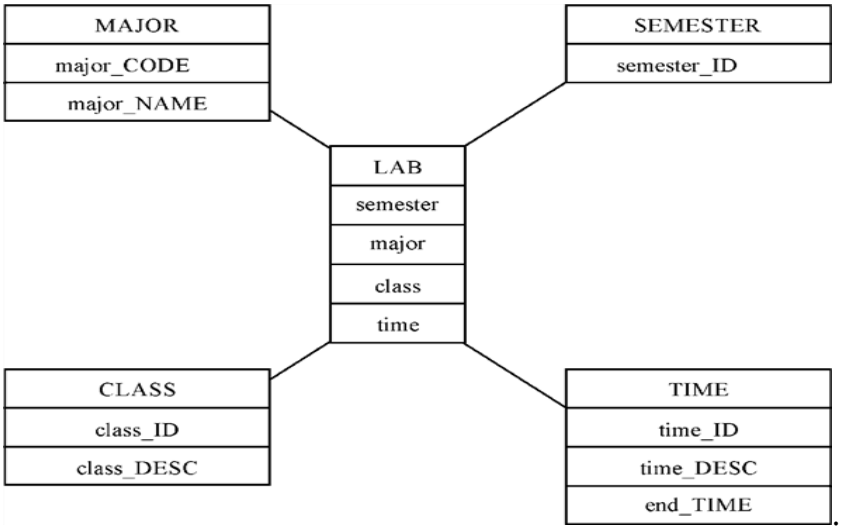
Note − Due to normalization in the Snowflake schema, the redundancy is reduced and therefore, it becomes easy to maintain and the save storage space.

Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.
- The following diagram shows two fact tables, namely sales and shipping.



- The sales fact table is same as that in the star schema.
- The shipping fact table has the five dimensions, namely item_key, time_key, shipper_key, from_location, to_location.
- The shipping fact table also contains two measures, namely dollars sold and units sold.
- It is also possible to share dimension tables between fact tables. For example, time, item



- ,

4 a) What are data cube/OLAP operations? Explain in detail

- **Roll-up (Drill-up):** The roll-up operation performs aggregation on a data cube, either by climbing up a concept hierarchy for a dimension or by dimension reduction.Performing roll-up using climbing up a concept hierarchy:Consider a hierarchy defined as the total order “street<city <province="" or="" state="" <country.”="" <="" p="">Rather than grouping the data by city, the resulting cube groups the data by country.Performing roll-up using dimension reduction:One or more dimensions are removed from the given cube.Consider a sales data cube containing only the two dimensions location and time.Roll-up may be performed by removing the time dimension, resulting in an aggregation of the total sales by location, rather than by both location and by time.

- **Drill-down:** Drill-down is the reverse of roll-up. It navigates from less detailed data to more detailed data. Drill-down can be realized by either stepping down a concept hierarchy for a dimension or introducing additional dimensionsPerforming a drill-down operation using stepping down a concept hierarchyConsider time defined as “day <month <quarter="" <year.”="" <="" p=""> Drill-down occurs by descending the time hierarchy from the level of quarter to the more detailed level of month.Performing a drill-down operation by adding new dimensions to a cubeConsider the central cube of the figure drill-down on can occur by introducing an additional dimension, such as customer group.

- **Slice:** The slice operation performs a selection on one dimension of the given cube, resulting in a sub cube. The figure shows a slice operation where the sales data are selected from the central cube for the dimension ‘time’ using the criterion ‘time= “Q1” ’. Dice: The dice operation defines a sub cube by performing a selection on two or more dimensions. The figure shows a dice operation on the central cube based on the following selection criteria that involve three dimensions: (location = “Toronto” or “Vancouver”) and (time = “Q1” or “Q2”) and (item = “home entertainment” or “computer”).

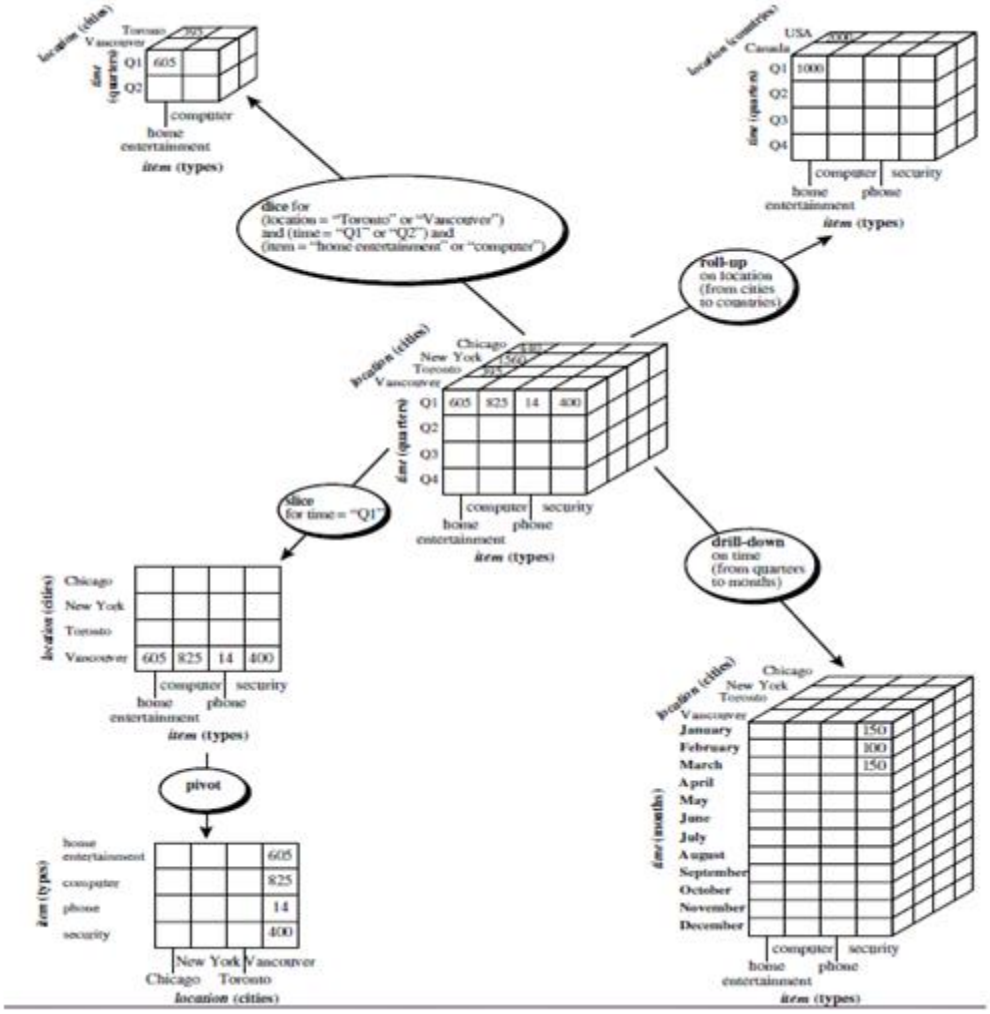
- Pivot (rotate): Pivot is a visualization operation that rotates the data axes in view in order to provide an alternative presentation of the data.The figure shows a pivot operation where the item and location axes in a 2-D slice are rotated.Other examples include rotating the axes in a 3-D cube, or transforming a 3-D cube into a series of 2-D planes.

- **Other OLAP operations(extra points for reference)**

• **Drill-across** operation executes queries involving more than one fact table.

• **Drill-through** operation uses relational SQL facilities to drill through the bottom level of a data cube down to its back-end relational tables. Ranking the top N or bottom N items in lists, as well as computing moving averages, growth rates, interests, internal rates of return, depreciation,

currency conversions, and statistical functions. LAP offers analytical modeling capabilities, including a calculation engine for deriving ratios, variance, and so on, and for computing measures across multiple dimensions It can generate summarizations, aggregations, and hierarchies at each granularity level and at every dimension intersection. LAP also supports functional models for forecasting, trend analysis, and statistical analysis. In this context, an OLAP engine is a powerful data analysis tool.



5 (a) Differentiate between ROLAP, MOLAP and HOLAP.

(b) Explain the differences between OLTP and OLAP.

Any 5 diffrence 5*1=5 Marks

	MOLAP	ROLAP

Data structure	Multidimensional database using sparse arrays	Relational tables (each cell is a row)
Disk space	Separate database for data cube; large for large data cubes	May not require any space other than that available in the data warehouse
Retrieval	Fast(pre-computed)	Slow (computes on-the-fly)
Scalability	Limited (cubes can be very large)	Excellent
Best suited for	Inexperienced users, limited set of queries	Experienced users, queries change frequently
DBMS facilities	Usually weak	Usually very strong

	OLAP	**OLTP**
Abbreviation	It stands for 'Online Analytical Processing'.	It stands for 'Online Transaction Processin
Use	It is used for Query Processing.	It is used for Transaction Processing.
Data	• It holds historical data. • It stores only relevant data. • It is de-normalized in	• It holds current data. • It stores all data. • It is normalized for efficient transaction pr • It has a small database. • It contains volatile data.

	the analytical process. • It has a large database. • It contains non-volatile data.	
Type	It is analysis driven.	It is application driven.
Source	The data comes from various OLTP sources.	It is the original source of data.
Purpose	To help with planning, problem solving, and decision support.	To control and run fundamental business tas
Business	It reveals the multi-dimensional view of all types of business activities.	It reveals the ongoing business process.
Updates	There are periodic long-running batch jobs which refresh the data.	Short and fast inserts and updates initiated by en
Queries	They are often complex queries involving aggregations.	They are standardized and simple queries

Speed	It is slow depending on the data.	It is very fast.
Market	It is customer orientated.	It is market orientated.
Database design	It is de-normalized with fewer tables and makes use of star or snowflake schemas.	It is highly normalized with many tables.
View	It represents managerial view.	It represents clerical or operator view.
Users	It has few concurrent users.	It has many concurrent users.

6 (a) What is Data Mining? Explain the process of knowledge discovery in databases (KDD) with neat diagram

Diagram:3 marks

Explanation:4 marks

Many people treat data mining as a synonym for another popularly used term, knowledge discovery from data, or KDD, while others view data mining as merely an essential step in the process of knowledge discovery. The knowledge discovery process is shown in Figure 1.4 as an iterative sequence of the following steps:

1. Data cleaning (to remove noise and inconsistent data)

2. Data integration (where multiple data sources may be combined)

3. Data selection (where data relevant to the analysis task are retrieved from the database)

4. Data transformation (where data are transformed and consolidated into forms appropriate for mining by performing summary or aggregation operations)4 5. Data mining (an essential process where intelligent methods are applied to extract data patterns)

6. Pattern evaluation : Identify the truly interesting patterns representing knowledge

7. Knowledge presentation (where visualization and knowledge representation techniques

are used to present mined knowledge to users)

Steps 1 through 4 are different forms of data preprocessing, where data are prepared for mining. The data mining step may interact with the user or a knowledge base. The interesting patterns are presented to the user and may be stored as new knowledge in the knowledge base.

The preceding view shows data mining as one step in the knowledge discovery process, albeit an essential one because it uncovers hidden patterns for evaluation. However, in industry, in media, and in the research milieu, the term *data mining* is often used to refer to the entire knowledge discovery process (perhaps because the term is shorter than *knowledge discovery from data*). Therefore, we adopt a broad view of data mining functionality: Data mining is the *process* of discovering interesting patterns and knowledge from *large* amounts of data. The data sources can include databases, data warehouses, the Web, other information repositories, or data that are streamed into the system dynamically.

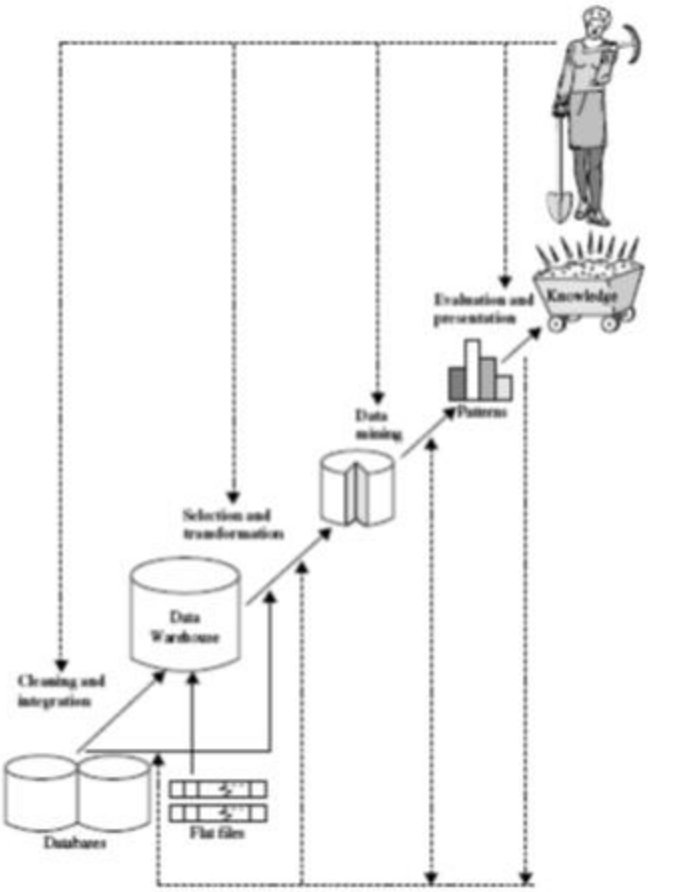


Figure 1.4 Data mining as a step in the process of knowledge discovery.

(b)Write a short note on data mining application

Any 3 application 1*3=3marks

1) Market basket analysis

2) Education

3) Climate prediction

(8) Explain the different challenges motivated the development of data mining technologies

Scalability

Because of advances in data generation and collection, data sets with sizes of gigabytes, terabytes, or even petabytes are becoming common. If data mining algorithms are to handle these massive data sets, then they must be scalable. Many data mining algorithms employ special

search strategies to handle exponential search problems. Scalability may also require the implementation of novel data structures to access individual records in an efficient manner. For instance, out-of-core algorithms may be necessary when processing data sets that cannot fit into main memory. Scalability can also be improved by using sampling or developing parallel and distributed algorithms.

High dimensionality

It is now common to encounter data sets with hundreds or thousands of attributes instead of the handful common a few decades ago. In bioinformatics, progress in microarray technology has produced gene

expression data involving thousands of features. Data sets with temporal or spatial components also tend to have high dimensionality. For example, consider a data set that contains measurements of temperature at various locations. If the temperature measurements are taken repeatedly for an extended

period, the number of dimensions (features) increases in proportion to the number of measurements taken.

Heterogeneous and Complex Data Traditional data analysis methods often deal with data sets containing attributes of the same type, either continuous or categorical. As the role of data mining in business, science, medicine, and other flelds has grown, so has the need for techniques that can handle heterogeneous attributes. Recent years have also seen the emergence of more complex data objects. Examples of such non-traditional types of data include collections of Web pages containing semi-structured text and hyperlinks; DNA data with sequential and three-dimensional structure; and climate data that consists of time series measurements (temperature, pressure, etc.) at various locations on the Earth's surface.

Data ownership and Distribution

Sometimes, the data needed for an analysis is not stored in one location or owned by one organization. Instead, the data is geographically distributed among resources belonging to multiple entities. This requires the development of distributed data mining techniques. Among the key challenges faced by distributed data mining algorithms include (1) how to reduce the amount of communication needed to perform the distributed computatior, (2) how to effectively consolidate the data mining results obtained from multiple sources, and (3) how to address data security issues.

Non-traditional Analysis

The traditional statistical approach is based on a hypothesize-and-test paradigm. In other words, a hypothesis is proposed, an experiment is designed to gather the data, and then the data is analyzed

with respect to the hypothesis. Unfortunately, this process is extremely laborintensive.