SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES (AUTONOMOUS)

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS(MCA)

LECTURE NOTES

Course: DATA MINING AND BUSINESS INTELLIGENCE

Year/Branch: I MCA/II SEMESTER

Regulation: R24

Prepared By: Dr.R.SARASWATHI

SYALLABUS :

I MCA – II SEMESTER			
COURSE CODE:	**24MCA125A**	**CREDITS:**	**3**
COURSE TITLE:	**DATA MINING AND BUSINESS INTELLIGENCE (Professional Elective-II)**	**L-T-P:**	**3-0-0**

PREREQUISITES: *A course on "Computer Networks"*

COUSE EDUCATIONAL OBJECTIVES :

CEO1*. Understanding of the importance of data mining and the principles of business intelligence*
CEO2*. To prepare data usingpre-processing techniques*
CEO3. *To Describing key business intelligence terms*
CEO4*. Determining the relevance of data to business*
CEO5*. Aligning business intelligence to organizational strategy*

UNIT- 1:

Why reporting and Analysing data, Raw data to valuable information-Lifecycle of Data - What is Business Intelligence - BI and DW in today's perspective - What is data warehousing - The building Blocks: Defining Features - Data warehouses and data 1marts - Overview of the components - Metadata in the data warehouse - Need for data warehousing - Basic elements of data warehousing - trends in data warehousing

UNIT- 2:

BI and DW architectures and its types - Relation between BI and DW - OLAP (Online analytical processing) definitions - Difference between OLAP and OLTP - Dimensional analysis - What are cubes? Drill-down and roll-up - slice and dice or rotation - OLAP models - ROLAP versus MOLAP - defining schemas: Stars, snowflakes and fact constellations

UNIT- 3:

Motivation for Data Mining - Data Mining-Definition and Functionalities – Classification of DM Systems - DM task primitives - Integration of a Data Mining system with a Database or a Data Warehouse - Issues in DM – KDD Process

UNIT- 4:

Why to pre-process data? - Data cleaning: Missing Values, Noisy Data - Data Integration and transformation - Data Reduction: Data cube aggregation, Dimensionality reduction - Data Compression - Numerosity Reduction - Data Mining Primitives - Languages and System Architectures: Task relevant data - Kind of Knowledge to be mined - Discretization and Concept Hierarchy.

UNIT- 5:

What is concept description? - Data Generalization and summarization-based characterization - Attribute relevance - class comparisons Association Rule Mining: Market basket analysis - basic concepts - Finding frequent item sets: Apriori algorithm - generating rules – Improved Apriori algorithm – IncrementalARM – Associative Classification – Rule Mining

TEXT BOOKS:

1. J. Han, M. Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann
2. M. Kantardzic, "Data mining: Concepts, models, methods and algorithms, John Wiley &Sons Inc.

REFERENCE BOOKS:

1. PaulrajPonnian, "Data Warehousing Fundamentals", John Willey.
2. M. Dunham, "Data Mining: Introductory and Advanced Topics", Pearson Education.
3. G. Shmueli, N.R. Patel, P.C. Bruce, "Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with XLMiner", Wiley India

COURSE OUTCOMES: *On successful completion of this course, students will be able to:*		POs related to COs
CO1	**Demonstrate** an understanding of the importance of data mining and the principles of business intelligence	PO1,PO2
CO2	**Understand** the BI and DW architectures and its types , Understanding the OLAP models	PO2,PO2, PO3
CO3	**Define and apply** metrics to measure the performance of various data mining algorithms	PO1,PO2,PO3
CO4	**Organize and Prepare** the data needed for data mining using pre preprocessing techniques	PO1,PO2, PO3, PO4
CO5	**Apply** BI to solve practical problems : Analyze the problem domain, use the data collected in enterprise apply the appropriate data mining technique, interpret and visualize the results and provide decision support.	PO1,PO2, PO4,PO8

CO-PO MAPPING (DETAILED; HIGH:3; MEDIUM:2; LOW:1)

Course	**POs** / **)s**	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**?O8**
116: Data Mining and Business Intelligence	**CO116.1**	**3**	**2**	**-**	**-**	**-**	**-**	**-**	**-**
	CO116.2	**3**	**3**	**3**	**-**	**-**	**-**	**-**	**-**
	CO116.3	**3**	**3**	**2**	**-**	**-**	**-**	**-**	**-**
	CO116.4	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**
	CO116.5	**3**	**2**	**-**	**2**	**-**	**-**	**-**	**2**
	CO116	**3**	**2.6**	**2.3**	**2**	**-**	**-**	**-**	**2**

UNIT–I

DATA MINING AND BUSINESS INTELLIGENCE

"Data is a precious thing and will last longer than the systems themselves."

– W. Edwards Deming

Introduction to Data Mining and Business Intelligence

Data reporting and analysis, along with data warehousing and business intelligence, form the backbone of modern data-driven business operations. This overview distils the provided content into structured sections for clarity.

Overview

Reporting organizes raw data into summaries for performance monitoring, while analysis extracts insights for improvement. The data lifecycle spans generation, collection, cleaning, storage, management, analysis, visualization, and interpretation, transforming data into actionable information. Business Intelligence (BI) integrates these via technologies like data warehousing, mining, and dashboards, with data warehouses serving as centralized repositories featuring components like source data, staging, storage, metadata, delivery systems, and management.

Objectives

- Understand reporting as data organization for monitoring business performance.
- Grasp analysis as exploring data for insights and optimization.
- Learn the data lifecycle stages to manage data effectively from creation to interpretation.
- Explore BI components, data warehousing architecture, and differences from data marts.
- Identify needs and best practices for building and maintaining data warehouses.

Learning Outcomes

- Differentiate reporting (summarization) from analysis (insight extraction).
- Map the eight-stage data lifecycle and alternative frameworks like creation-to-destruction.
- Describe data warehouse components: source/staging/storage, metadata (technical/business), data marts, information delivery (OLAP, mining tools), and management.
- Compare data warehouses (centralized, detailed, flexible) vs. data marts (departmental, summarized, easier to build).
- Recognize BI benefits like real-time analytics, cloud integration, and self-service tools.

1.1. Reporting and analyzing data

Reporting: The process of organizing data into informational summaries in order to monitor how different areas of a business are performing. Reporting includes building, configuring, consolidating, organizing formatting and summarizing.

Analysis: The process of exploring data and reports in order to extract meaningful insights, for better understanding and improvement in performance.

Key reasons to report and analyze data:

1. Informed decision-making:

By understanding patterns and trends within data, businesses can make strategic choices based on facts rather than intuition.

2. Performance tracking:

Regularly monitoring key metrics through reports helps identify areas of success and areas needing improvement, allowing for course correction.

3. Problem identification:

Analyzing data can reveal hidden issues or potential risks that might not be apparent otherwise.

4. Customer insights:

Analyzing customer data can provide valuable information about their needs, preferences, and behaviors, leading to better product and service development.

5. Operational efficiency:

Data analysis can identify areas where processes can be optimized to improve productivity and reduce costs.

6. Market understanding:

Analyzing market trends and competitor data helps businesses stay ahead of the competition.

7. Communication and transparency:

Presenting data in clear reports allows for effective communication of key findings to stakeholders.

Key Terminologies

Term	Meaning
KPI	Key Performance Indicator used to measure business success
Dashboard	Visual interface showing business metrics
Data Visualization	Graphical representation of data
Analytics	Process of discovering patterns in data

Real-World Example

An **e-commerce company like Amazon** uses reporting dashboards to track:

- Daily sales
- Customer purchase patterns
- Inventory levels

Managers analyze this data to improve marketing and supply chain decisions.

Advantages

- Helps in **data-driven decision making**
- Improves **business transparency**
- Identifies **performance gaps**
- Supports **strategic planning**

Disadvantages

- Requires **accurate and clean data**
- High **implementation cost**
- Misinterpretation may lead to **wrong decisions**

Applications

- Business performance monitoring
- Financial reporting

- Market trend analysis
- Customer behavior analysis

1.2. Raw Data to Valuable Information

Definition

Raw data refers to **unprocessed facts and figures collected from different sources**, which become useful information after processing and analysis.

Example:

Raw Data → Process → Information

Sales Transactions → Data Analysis → Sales Trend Insights

Real-World Example

Retail stores collect **customer purchase data**.

After analysis, they discover:

- Peak buying hours
- Most purchased products
- Seasonal demand trends

This helps them optimize **inventory and pricing strategies**.

Advantages

- Converts large data into **useful knowledge**
- Helps **predict future trends**
- Improves **decision making**

Limitations

- Requires **data cleaning**
- Time-consuming processing
- Needs specialized tools

1.3. The Lifecycle Of Data

Transforming **raw data into valuable information**, typically involves stages like data collection, data cleaning/processing, data storage, data analysis, interpretation, and visualization where raw data is cleaned, structured, analyzed to extract meaningful insights and patterns, ultimately producing valuable information that can be used for decision-making

Key stages in the data lifecycle:

1. Data generation occurs regardless of whether you're aware of it, especially in our increasingly online world. Some of **this data is generated by your organization, some by your customers, and some by third parties** you may or may not be aware of. Every sale, purchase, hire, communication, interaction—everything generates data. Given the proper attention, this data can often lead to powerful insights that allow you to better serve your customers and become more effective in your role.

Data is created from transactions, sensors, user inputs, etc

2. Data Collection:

Gathering raw data from various sources like **surveys, sensors, databases, or web interactions.**

You can collect data in a variety of ways, including:

- **Forms:** Web forms, client or customer intake forms, vendor forms, and human resources applications are some of the most common ways businesses generate data.
- **Surveys:** Surveys can be an effective way to gather vast amounts of information from a large number of respondents.
- **Interviews:** Interviews and focus groups conducted with customers, users, or job applicants offer opportunities to gather qualitative and subjective data that may be difficult to capture through other means.
- **Direct Observation:** Observing how a customer interacts with your website, application, or product can be an effective way to gather data that may not be offered through the methods above.

It's important to note that many organizations take a broad approach to data collection, capturing as much data as possible from each interaction and storing it for potential use. While drawing from this supply is certainly an option, it's always important to start by creating a plan to capture the data you know is critical to your project.

3. Data Cleaning/Preprocessing:

Removing errors, inconsistencies, and irrelevant data to prepare it for analysis.

Once data has been collected, it must be processed. Data processing can refer to various activities, including:

- **Data wrangling**, in which a data set is cleaned and transformed from its raw form into something more **accessible and usable**. This is also known as data cleaning, data remediation.
- **Data compression**, in which data is **transformed** into a format that can be more efficiently stored.
- **Data encryption,** in which data is translated into another form of code to protect it from privacy concerns.

4. Data Storage

After data has been collected and processed, it must be stored for future use. This is most commonly achieved through the creation of databases or datasets. These datasets may then be

stored in **the cloud, on servers, or using another form of physical storage like a hard drive, CD, cassette, or floppy disk.**

When determining how to best store data for your organization, it's important to build in a certain level of redundancy to ensure that a copy of your data will be protected and accessible, even if the original source becomes corrupted or compromised.

5. Management

Data management, also called **database management**, involves organizing, storing, and retrieving data as necessary **over the life of a data project**. While referred to here as a "step," it's an ongoing process that takes place from the beginning through the end of a project. Data management includes everything from **storage and encryption** to implementing access logs and changelogs that track who has accessed data and what changes they may have made.

6. Data Analysis:

Data analysis refers to processes that attempt to meaningful insights from raw data. **Analysts and data scientists use** different tools and strategies to conduct these analyses. Some of the more commonly used methods include statistical modeling, algorithms, artificial intelligence, data mining, and machine learning.

7. Visualization

Data visualization refers to the process of **creating graphical representations** of your information, typically through the use of one or more visualization tools. Visualizing data makes it easier to quickly communicate your analysis to a wider audience both inside and outside your organization. The form your visualization takes depends on the data you're working with, as well as the story you want to communicate.

While technically not a required step for all data projects, data visualization has become an increasingly important part of the data life cycle.

8. Interpretation

Finally, the interpretation phase of the data life cycle provides the opportunity to make sense of your analysis and visualization. Beyond simply presenting the data, this is when you investigate it through the lens of your expertise and understanding. **Your interpretation may not only include a description or explanation of what the data shows but, more importantly, what the implications may be.**

Real-World Example

In a banking system:

- Customer transactions generate data
- Data is stored in databases
- Analysts study spending patterns
- Banks offer personalized financial services

Advantages

- Improves data management
- Ensures data quality
- Supports business intelligence systems

Limitations

- Requires strong data governance
- Storage and management cost

1.4. What is Business Intelligence (BI)?

1. Definition

Business Intelligence (BI) refers to **technologies, strategies, and practices** used to collect, analyze, and present business data for decision-making.

2. Components of BI

- **Data Warehousing** – Centralized storage of structured data.
- **Data Mining** – Discovering patterns and relationships in data.
- **Reporting and Dashboards** – Visual representation of key performance indicators (KPIs).
- **Predictive Analytics** – Forecasting future trends using AI/ML.

3. Benefits of BI

- Improved **decision-making**.
- Enhanced **operational efficiency**.
- Better **customer insights**.
- Competitive **advantage**.

BI and Data Warehousing (DW) in Today's Perspective

- **Real-time BI:** Companies now demand real-time analytics instead of traditional batch processing.
- **Cloud-Based DW:** Data is now stored and analyzed on cloud platforms like AWS, Azure, and Google Cloud.
- **Big Data Integration:** BI systems now process structured and unstructured data.
- **AI and Machine Learning:** Automating analytics and predictive insights.

Self-Service BI: Non-technical users can analyze data without IT support.

1.5. Components or Building Blocks of Data Warehouse

Data warehouse Architecture and its components

Architecture is the proper arrangement of the elements. We build a data warehouse with software and hardware components. To suit the requirements of our organizations, we arrange these building we may want to boost up another part with extra tools and services. All of these depends on our circumstances

1. Source Data Component

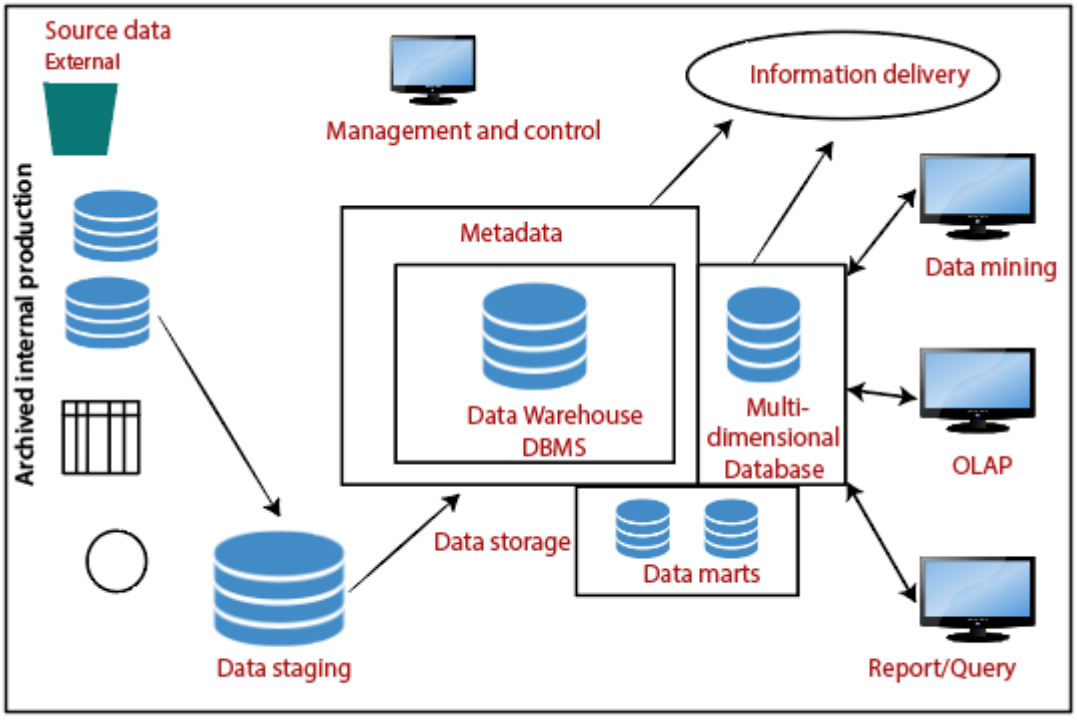
2. Data Staging Component

3. Data Storage Components

4. Metadata

5.Information delivery system

6. Data warehouse management and Control



Components or Building Blocks of Data Warehouse

The figure shows the essentiall elements of a typical warehouse. We see the Source Data component shows on the left. The Data staging element serves as the next building block. In the middle, we see the Data Storage component that handles the data warehouses data. This element not only stores and manages the data; it also keeps track of data using the metadata repository. The Information Delivery component shows on the right consists of all the different ways of making the information from the data warehouses available to the users.

1.Source Data Component

Source data coming into the data warehouses may be grouped into four broad categories:

Production Data: This type of data comes from the different operating systems of the enterprise. Based on the data requirements in the data warehouse, we choose segments of the data from the various operational modes.

Real-time data from **daily operations**, such as transactional databases (CRM, ERP), which forms the core of the warehouse data.

They perform conversions, summarization, key changes, structural changes and condensation. The data transformation is required so that the information can by used by decision support tools. The transformation produces programs, control statements, JCL code, COBOL code, UNIX scripts, and SQL DDL code etc., to move the data into data warehouse from multiple operational systems.

The functionalities of these tools are listed below:

- To remove unwanted data from operational db
- Converting to common data names and attributes
- Calculating summaries and derived data
- Establishing defaults for missing data
- Accommodating source data definition change.

Issues to be considered while data sourcing, cleanup, extract and transformation:

Data heterogeneity It refers to DBMS different nature such as it may be in different data modules, it may have different access languages, it may have data navigation methods, operations, concurrency, integrity and recovery processes etc

Internal Data: In each organization, the client keeps their "**private**" spreadsheets, reports, customer profiles, and sometimes even department databases. This is the internal data, part of which could be useful in a data warehouse.

Archived Data: Operational systems are mainly intended to run the current business. In every operational system, we periodically take the old data and store it in achieved files.

External Data: Most executives depend on information from external sources for a large percentage of the information they use. They use statistics associating to their industry produced by the external department.

2. Data Staging Component

After we have been extracted data from various operational systems and external sources, we have to prepare the files for storing in the data warehouse. The extracted data coming from several different sources need to be changed, converted, and made ready in a format that is relevant to be saved for querying and analysis.

We will now discuss **the three primary functions** that take place in the staging area.

1) Data Extraction: This method has to deal with numerous data sources. We have to employ the appropriate techniques for each data source.

2) Data Transformation

Data transformation also contains purging source data that is not useful and separating outsource records into new combinations. Sorting and merging of data take place on a large scale in the data staging area. When the data transformation function ends, we have a collection of integrated data that is cleaned, standardized, and summarized.

3) Data Loading: Two distinct categories of tasks form data loading functions. When we complete the structure and construction of the data warehouse and go live for the first time, we do the initial loading of the information into the data warehouse storage. The initial load moves high volumes of data using up a substantial amount of time.

3.Data Storage Components

A) Data warehousing

Data storage for the data warehousing is a split repository. The data repositories for the operational systems generally include only the current data. Also, these data repositories include the data structured in highly normalized for fast and efficient processing.

The data source for data warehouse is coming from operational applications. The data entered into the data warehouse transformed into an integrated structure and format. The transformation process involves conversion, summarization, filtering and condensation. The datawarehouse must be capable of holding and managing large volumes of data as well as different structure of data structures over the time.

Data warehousing is the process of storing data, and data mining is the process of analyzing that data. Both are key components of business intelligence (BI).

Data warehousing

- A centralized repository that stores data from various sources
- A relational database that can store large amounts of data
- A collection of databases that can be integrated to provide new insights
- A time-variant model that stores historical data and continuously adds new data

B) Meta data

It is data about data. It is used for maintaining, managing and using the data warehouse. It is classified into two:

a) Technical Meta data: It contains information about data warehouse data used by warehouse designer, administrator to carry out development and management tasks. It includes,

- Info about data stores Transformation descriptions. That is mapping methods from operational data base to warehouse data base
- Warehouse Object and data structure definitions for target data
- The rules used to perform clean up, and data enhancement
- Data mapping operations
- Access authorization, backup history, archive history, info delivery history, data acquisition history, data access etc.

b) Business Meta data: It contains info that gives info stored in data warehouse to users. It includes, Subject areas, and info object type including queries, reports, images, video, audio clips etc.

- Internet home pages
- Info related to info delivery system
- Data warehouse operational info such as ownerships, audit trails etc.,
- Meta data helps the users to understand content and find the data. Meta data are stored in a separate data stores which is known as informational directory or Meta data repository which helps to integrate, maintain and view the contents of the data warehouse.

C. Data marts

A data mart is a database that stores and organizes data for a specific business unit or department. It's a subset of a company's larger data storage system.

Data mart is used in the following situation:

- Extremely urgent user requirement
- The absence of a budget for a full scale data warehouse strategy
- The decentralization of business needs
- The attraction of easy to use tools and mind sized project

4. Information delivery system

An "information delivery system" in a data warehouse refers to the set of tools and processes that allow users to access, analyze, and visualize data stored within the warehouse, effectively presenting relevant information in a format that supports informed decision-making, typically through reports, dashboards, and interactive queries, delivered via user interfaces like web applications or dedicated BI tools.

Key points about information delivery systems in data warehouses:

Its purpose is to provide info to business users for decision making. There are five categories:

a) Data query and reporting tools
b) Application development tools
c) OLAP tools
d) Data mining tools

a) Query and reporting tools are used to generate query and report. There are two types of reporting tools. They are:

1.Production reporting tool used to generate regular operational reports

2. Desktop report writer are inexpensive desktop tools designed for end users.

- **Managed Query tools:** used to generate SQL query. It uses Meta layer software in between users and databases which offers a point-and-click creation of SQL statement. This tool is a

preferred choice of users to perform segment identification, demographic analysis, territory management and preparation of customer mailing lists etc.

b) Application development tools: This is a graphical data access environment which integrates

c) OLAP tools with data warehouse and can be used to access all db systems

OLAP Tools: are used to analyze the data in multi dimensional and complex views. To enable multidimensional properties it uses MDDB and MRDB where MDDB refers multi dimensional data base and MRDB refers multi relational data bases.

d) Data mining tools: are used to discover knowledge from the data warehouse data also can be used for data visualization and data correction purposes

5. Data warehouse management and Control

The management of data warehouse includes security and priority management

- Monitoring updates from multiple sources
- Data quality checks
- Managing and updating meta data
- Auditing and reporting data warehouse usage and status
- Purging data
- Replicating, sub setting and distributing data
- Backup and recovery
- Data warehouse storage management which includes capacity planning, hierarchical storage management and purging of aged data etc.,

1.6. Data Warehouse

A Data Warehouse is like a data management system or a large collection of business data used to support business intelligence and analytics. It is mostly used for data analysis and reporting purposes.

Data Mart

A data mart is an uncomplicated form of a subject-oriented database system that is concentrated on business matters, such as sales, finance, or marketing.

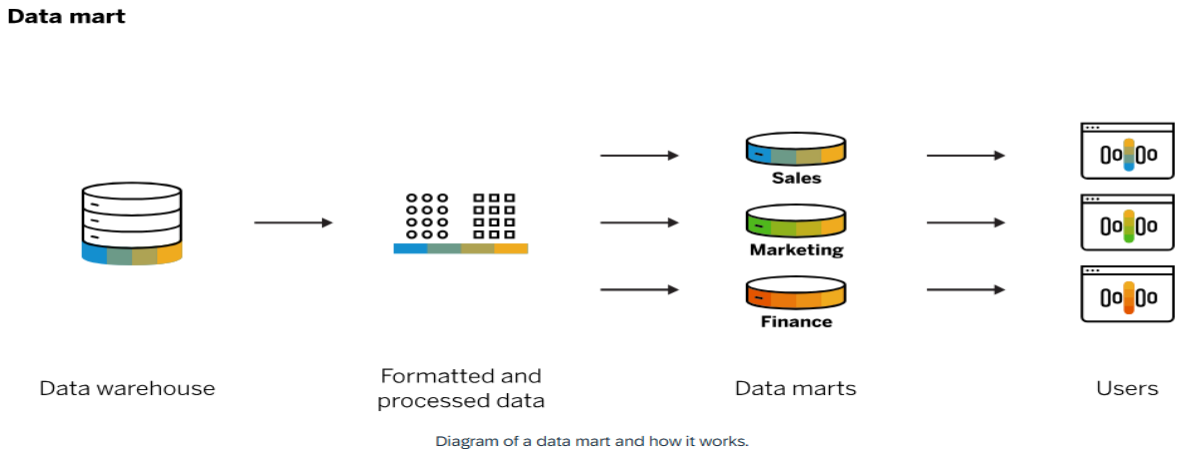


Diagram of a data mart and how it works.

Difference between Data Warehouse and Data Mart

S.N0.	Data Warehouse	Data Mart
1.	It is a centralised system.	It is not a centralised system.
2.	Data warehouse defines a top-down model.	Data mart defines a bottom-up model.
3.	Slightly denormalization is involved in data warehouses.	Highly denormalization is involved in data mart.
4.	It is tough to build a data warehouse.	It is easy to build a data mart.
5.	Fact constellation schema is preferred in data warehouses.	Star schema and snowflake schema are preferred in data mart.
6.	It is more flexible as compared to the data mart.	It is not flexible.
7.	It is data-oriented in behaviour.	Data mart is project-oriented in behaviour.
8.	They mostly have longer life spans.	It has a shorter life span.
9.	Here we get the data in a detailed format.	Here we get the summarised version of data.
10.	The data warehouse is huge in size.	It is smaller as compared to the data warehouse.

1.7. Overview of the Components

A typical **data warehouse** has four main components:

- a Central database,

- ETL (extract, transform, load) tools,
- Metadata, and
- Access tools.

All of these components are engineered for speed so that you can get results quickly and analyze data on the fly.

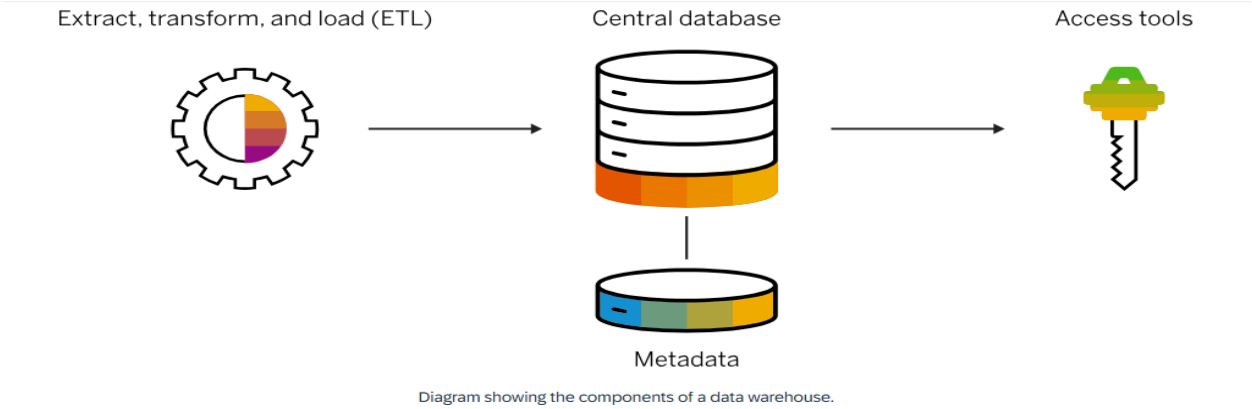


Diagram showing the components of a data warehouse.

Diagram showing the components of a data warehouse.

1. **Central database:** A database serves as the foundation of your data warehouse. Traditionally, these have been standard **relational databases** running on premise or in the cloud. But because of Big Data, the need for true, real-time performance, and a drastic reduction in the cost of RAM, in-memory databases are rapidly gaining in popularity.
2. **Data integration:** Data is pulled from source systems and modified to align the information for rapid analytical consumption using a variety of data integration approaches such **as ETL** (extract, transform, load) and ELT as well as real-time data replication, bulk-load processing, data transformation, and data quality and enrichment services.
3. **Metadata:** Metadata is data about your data. It specifies the **source, usage, values, and other features of the data sets in your data warehouse.** There is **business metadata**, which adds context to your data, and **technical metadata**, which describes how to access data – including where it resides and how it is structured.
4. **Data warehouse access tools:** Access tools allow users to interact with the data in your data warehouse. Examples of access tools include: query and reporting tools, application development tools, data mining tools, and OLAP tools.

1.8. Metadata in the Data warehouse

Metadata is simply defined as data about data. The data that is used to represent other data is known as metadata. For example, the index of a book serves as a metadata for the contents in the book. In other words, we can say that metadata is the **summarized data** that leads us to detailed data. In terms of data warehouse, we can define metadata as follows.

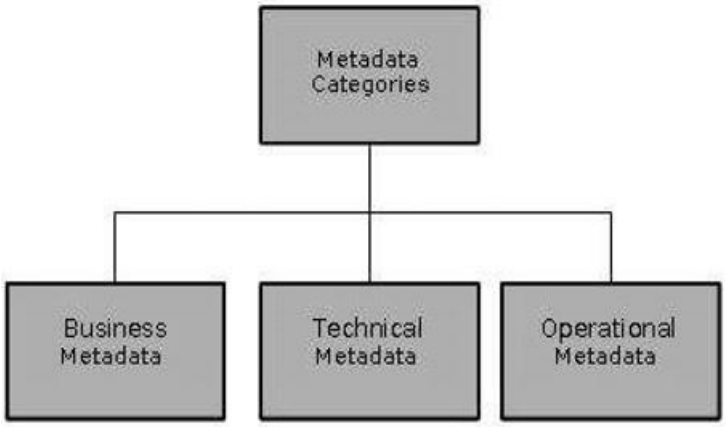
- Metadata is the road-map to a data warehouse.
- Metadata in a data warehouse defines the warehouse objects.

- Metadata acts as a directory. This directory helps the decision support system to locate the contents of a data warehouse.

Categories of Metadata

Metadata can be broadly categorized into three categories –

- **Business Metadata** – It has the data ownership information, business definition, and changing policies.
- **Technical Metadata** – It includes database system names, table and column names and sizes, data types and allowed values. Technical metadata also includes structural information such as **primary and foreign key attributes and indices**.
- **Operational Metadata** – It includes **currency of data and data lineage**. Currency of data means whether the **data is active, archived**, or purged. Lineage of data means the history of data migrated and transformation applied on it.



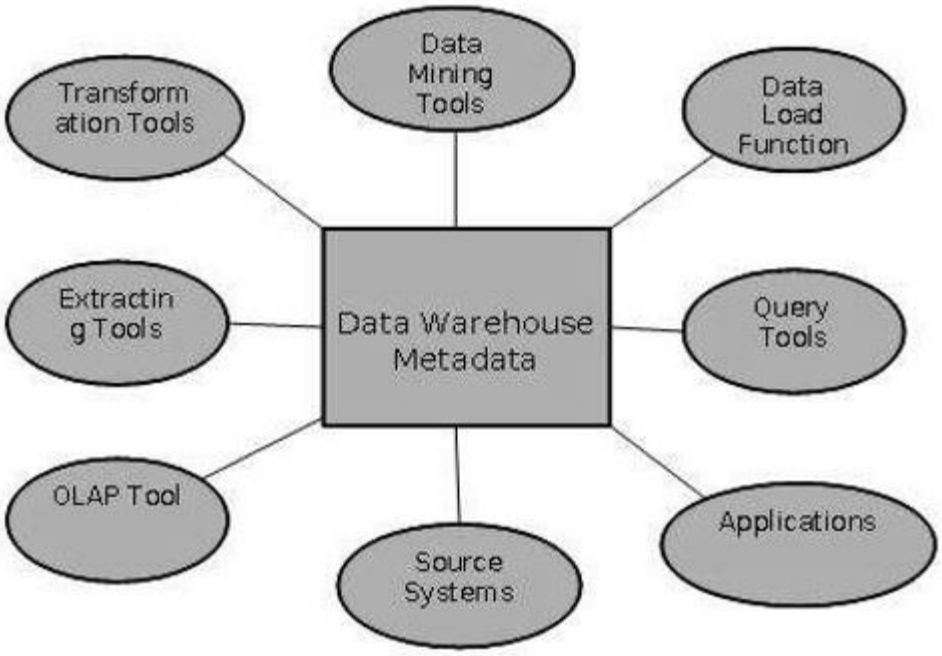
Role of Metadata

Metadata has a very important role in a data warehouse. The role of metadata in a warehouse is different from the warehouse data, yet it plays an important role. The various roles of metadata are explained below.

- Metadata acts as a **directory**.
- This directory helps the **decision support system** to locate the contents of the data warehouse.
- Metadata helps in decision support system for mapping of data when data is transformed from operational environment to data warehouse environment.
- Metadata helps in summarization between **current** detailed data and highly summarized data.
- Metadata also helps in summarization between **lightly** detailed data and highly summarized data.
- Metadata is used for **query tools**.
- Metadata is used in **extraction and cleansing tools**.

- Metadata is used **in reporting tools**.
- Metadata is used in **transformation tools.**
- Metadata plays an important role in loading functions.

The following diagram shows the roles of metadata.



Metadata Repository

Metadata repository is an integral part of a data warehouse system. It has the following metadata **Definition of data warehouse** – It includes the description of structure of data warehouse. The description is defined by schema, view, hierarchies, derived data definitions, and data mart locations and contents.

- **Business metadata** – It contains has the data ownership information, business definition, and changing policies.
- **Operational Metadata** – It includes currency of data and data lineage. Currency of data means whether the data is active, archived, or purged. Lineage of data means the history of data migrated and transformation applied on it.
- **Data for mapping from operational environment to data warehouse** – It includes the source databases and their contents, data extraction, data partition cleaning, transformation rules, data refresh and purging rules.
- **Algorithms for summarization** – It includes dimension algorithms, data on granularity, aggregation, summarizing, etc.

Challenges for Metadata Management

The importance of metadata cannot be overstated. Metadata helps in driving the accuracy of reports, validates data transformation, and ensures the accuracy of calculations. Metadata also enforces the definition of business terms to business end-users. With all these uses of metadata, it also has its challenges. Some of the challenges are discussed below.

- Metadata in **a big organization** is scattered across the organization. This metadata is spread in spreadsheets, databases, and applications.
- Metadata could be present in **text files or multimedia files**. To use this data for information management solutions, it has to be correctly defined.
- There are no industry-wide accepted standards. Data management solution vendors have narrow focus.
- There are no easy and accepted methods of passing metadata.

1.9. Need for Data Warehousing

1. Handling Large Volumes of Data: Traditional databases can only store a limited amount of data (MBs to GBs), whereas a data warehouse is designed to handle much larger datasets (TBs), allowing businesses to store and manage massive amounts of historical data.

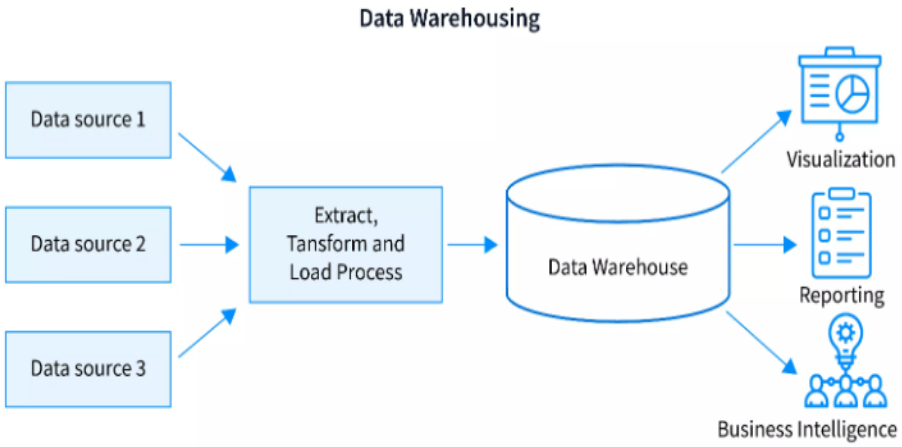
2. Enhanced Analytics: Transactional databases are not optimized for analytical purposes. A data warehouse is built specifically for data analysis, enabling businesses to **perform complex queries** and **gain insights from historical data**.

3. Centralized Data Storage: A data warehouse acts as a central repository for all organizational data, helping businesses to integrate data from multiple sources and have a unified view of their operations for better decision-making.

4. Trend Analysis: By storing historical data, a data warehouse allows businesses to analyze trends over time, enabling them to make **strategic** decisions based on **past performance** and **predict futureoutcomes**.

5. Support for Business Intelligence: Data warehouses support business intelligence tools and reporting systems, providing decision-makers with easy access to critical information, which enhances operational efficiency and supports data-driven strategies.

Basic elements of data warehousing:



Building a Data Warehouse – Some steps that are needed for building any data warehouse are as following below:

1. **To extract the data (transnational) from different data sources:** For building a data warehouse, a data is extracted from various data sources and that data is stored in central storage area. For extraction of the data Microsoft has come up with an excellent tool. When you purchase Microsoft SQL Server, then this tool will be available at free of cost.

2. **To transform the transnational data:** There are various DBMS where many of the companies stores their data. Some of them are: MS Access, MS SQL Server, Oracle, Sybase etc. Also these companies saves the data in spreadsheets, flat files, mail systems etc. Relating a data from all these sources is done while building a data warehouse.

3. **To load the data (transformed) into the dimensional database:** After building a dimensional model, the **data is loaded in the dimensional** database. This process combines the several columns together or it may split one field into the several columns. There are **two stages** at which transformation of the data can be performed and they are: while loading the data into the dimensional model or while data extraction from their origins.

4. **To purchase a front-end reporting tool:** There are top notch analytical tools are available in the market. These tools are provided by the several major vendors. A cost effective tool and Data Analyzer is released by the Microsoft on its own.

For the warehouse there is an acquisition of the data. There must be a use of multiple and heterogeneous sources for the data extraction, example databases. There is a need for the consistency for which formation of data must be done within the warehouse. Reconciliation of names, meanings and domains of data must be done from unrelated sources. There is also a need for the installation of the data from various sources in the data model of the warehouse. Conversion of the data might be done from object oriented, relational or legacy databases to a multidimensional model. One of the largest labor demanding component of data warehouse construction is data cleaning, which is one of the complex process. Before loading of the data in the warehouse, there should be cleaning of the data. All the work of loading must be done in warehouse for better performance. The only feasible and better approach for it is incremental updating. Data storage in the data warehouse:

- Refresh the data
- To provide the time variant data
- To store the data as per the data model of the warehouse
- Purging the data
- To support the updating of the warehouse data

Some of the important designs for the data warehouse are:

- Modular component design
- Consideration of the parallel architecture
- Consideration of the distributed architecture
- Usage protection
- Characteristics of available sources
- Design of the metadata component
- The fit of the data model

The major determining characteristics for the design of the warehouse are the architecture of the organizations distributed computing environment. The distributed warehouse and the federated warehouse are the two basic distributed architecture. There are some benefits from the distributed warehouse, some of them are:

- Improved load balancing
- Scalability of performance
- Higher availability

Federated warehouse is a decentralized confederation of autonomous data warehouses. Each of them has its own metadata repository. Now a days large organizations start choosing a federated data marts instead of building a huge data warehouse.

Learning Outcomes (Aligned with Bloom's Taxonomy)

After studying this unit, students will be able to:

1. **Remember** – Define BI, Data Warehouse, Data Mart.
2. **Understand** – Explain the lifecycle of data and BI architecture.
3. **Apply** – Use BI tools for business decision support.
4. **Analyze** – Compare data warehouse and data mart systems.
5. **Evaluate** – Assess advantages and limitations of BI systems.

Unit Highlights

- Introduction to **Business Intelligence and Data Warehousing**
- Importance of **reporting and analyzing data**
- Understanding **data lifecycle stages**
- Study of **data warehouse components**
- Role of **metadata and data marts**
- Trends in **mn data warehousing**

Case Study

Case Study: Retail Business Intelligence
A retail company collects data from sales transactions, customer purchases, and inventory systems. Using a **data warehouse and BI tools**, the company analyzes sales performance and customer behavior. This helps the management identify popular products, optimize stock levels, and improve marketing strategies.

Book References

1. **Jiawei Han, Micheline Kamber, Jian Pei** – *Data Mining: Concepts and Techniques*.
2. **Alex Berson & Stephen Smith** – *Data Warehousing, Data Mining and OLAP*.
3. **Pang-Ning Tan, Michael Steinbach, Vipin Kumar** – *Introduction to Data Mining*.
4. **Margaret H. Dunham** – *Data Mining: Introductory and Advanced Topics*.

S. No	QUESTIONS	Blooms Taxonomy Level
	UNIT –I INTRODUCTION	
	Part –A	
1.	What is raw data?	L1
2.	Define Business Intelligence.	L1
3.	What is the purpose of analyzing data?	L2
4.	Mention any two features of data warehouses.	L1
5.	What do you mean by data mart?	L1
6.	Define metadata in the context of data warehousing.	L1
7.	What is the need for data warehousing?	L2
8.	Name any two components of a data warehouse.	L1
9.	What is the relationship between Business Intelligence and Data Warehousing?	L2
10	List any two trends in data warehousing.	L1
	Part - B	
1	Explain the lifecycle of data and how raw data is transformed into valuable information.	L2
2	Discuss the importance of Business Intelligence and its role in modern organizations.	L4
3	What is data warehousing? Describe its basic elements and the need for it.	L2
4	Explain the architecture and components of a data warehouse with a neat diagram.	L2
5	What are the defining features of a data warehouse? How is it different from a data mart?	L4
6	Describe the relationship between Business Intelligence and Data Warehousing in today’s context.	L4
7	Explain the role and importance of metadata in data warehousing.	L2
8	Discuss current trends in data warehousing and their impact on business decision-making.	L5

UNIT – II

BI AND DATA WAREHOUSE ARCHITECTURES

" The goal is to turn data into information and information into insight." – Carly Fiorina

Overview

Business Intelligence (BI) and Data Warehouse (DW) technologies help organizations collect, store, process, and analyze large volumes of data for better decision making. A Data Warehouse acts as a centralized repository for integrated and historical data, while Business Intelligence tools analyze this data to generate meaningful insights. These technologies help organizations understand patterns, trends, and relationships in data to support strategic planning.

Objectives

- Understand Business Intelligence and Data Warehouse architecture
- Study different types of Data Warehouse architectures
- Understand OLAP concepts and multidimensional analysis
- Compare OLAP and OLTP systems

Learning Outcomes

After completing this unit, students will be able to:

- Explain BI and Data Warehouse architecture
- Differentiate OLTP and OLAP systems
- Describe OLAP operations such as roll-up, drill-down, slice and dice
- Analyze dimensional models such as star schema and snowflake schema

Importance of Studying this Unit

- Supports **data-driven decision making**
- Helps analyze **large volumes of business data**
- Improves **strategic planning in organizations**
- Enables **multidimensional data analysis**

Key Terminologies

- Business Intelligence (BI) & DW
- ETL Process (Extract, Transform, Load)
- OLAP (Online Analytical Processing)
- OLTP (Online Transaction Processing)
- OLAP Cube
- Multidimensional Data Model
- Fact Table
- OLAP Operations (Roll-up, Drill-down, Slice, Dice, Pivot)

2.1 The architecture of Business Intelligence (BI) and Data Warehouse (DW)

Overview
Business Intelligence (BI) and Data Warehouse (DW) systems are designed to collect, store, process, and analyze large volumes of organizational data to support strategic decision-making. The architecture consists of multiple layers that work together to transform raw data into meaningful information.

1. Data Sources
The data source layer is the bottom level of the architecture. It includes operational systems such as ERP systems, CRM systems, legacy databases, flat files, spreadsheets, and web applications. These systems generate large volumes of transactional data. However, this raw data is often unorganized, inconsistent, and stored in different formats.

2. ETL Process (Extract, Transform, Load)
ETL is used for data integration in data warehousing.

- Extract: Data is collected from multiple heterogeneous sources.
- Transform: Data is cleaned, validated, and formatted to remove inconsistencies and redundancy.
- Load: The processed data is stored in the data warehouse.

This process ensures high data quality, accuracy, and consistency.

3. Data Warehouse Layer
The Data Warehouse acts as a centralized repository that stores integrated and historical data. It is subject-oriented, time-variant, and non-volatile. The data warehouse supports analytical processing rather than routine transactions. It is usually designed using dimensional models such as star schema and snowflake schema to improve query performance.

4. Data Mart
A Data Mart is a subset of the data warehouse designed for a specific department such as sales, finance, or human resources. It enables department-level analysis and improves query response time by providing focused data.

5. OLAP Server
The OLAP (Online Analytical Processing) server enables multidimensional data analysis using data cubes. It supports operations such as drill-down, roll-up, slice, and dice. OLAP allows users to analyze data from different perspectives and identify patterns and trends.

6. BI Presentation Layer
The BI presentation layer is the top layer of the architecture. It includes reporting tools, dashboards, scorecards, visualization tools, and data mining applications. These tools present data in the form of charts, graphs, and performance indicators, helping managers make informed strategic decision.

Three Types of Data Warehouse Architecture

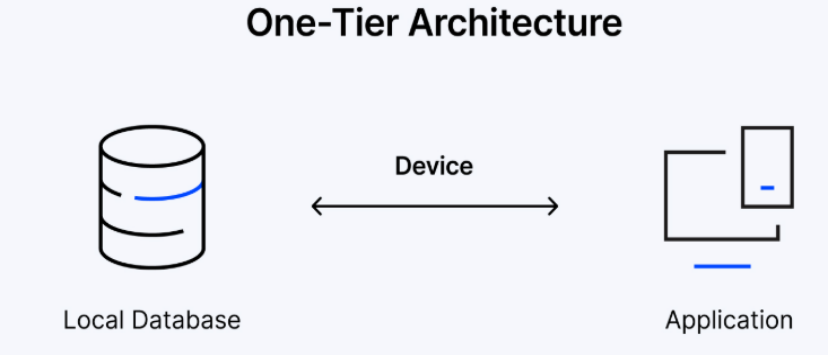
There are three common data warehouse architecture types typically used for building a data warehouse:

I. Single-Tier Architecture
II. Two-Tier Architecture
III. Three-Tier Architecture

Each type of data warehouse architecture has its own benefits and limitations. Let's explore the unique characteristics of each one of them.

I. Single-Tier Architecture

The single-tier data warehouse architecture reduces the amount of data stored in a data warehouse by building a more compact data set. Its advantage is that it helps remove data redundancies and improves the quality of your data.



However, it isn't the ideal solution for agencies that own large volumes of data and operate with multiple data streams because it's inefficient.

The single-tier architecture has three layers:

- A source layer
- A data warehouse layer
- An analysis layer

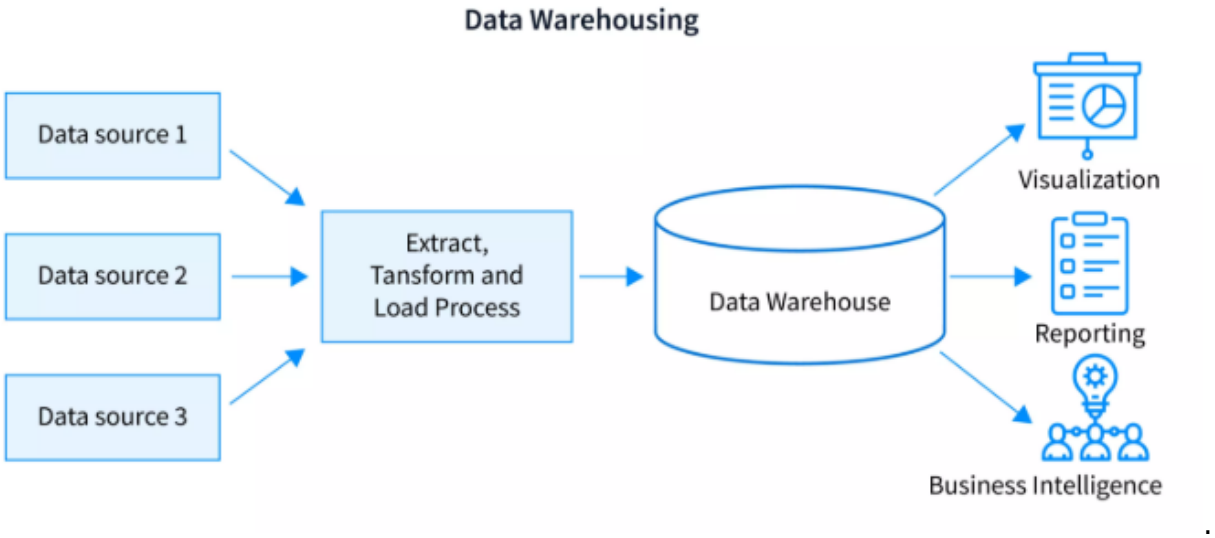
In the single-tier architecture, only the source layer is physical. The data warehouse layer is virtual and provides data in a multidimensional view, created by an intermediate processing layer.

Example: A small retail store might use a single-tier architecture to store and analyze sales data.

Drawbacks: Performance can be affected when transactional and analytical processes are not separate, and it struggles with enterprise-wide data access and scalability.

II. Two-Tier Architecture

Unlike the single-tier architecture, the two-tier architecture contains a data staging area that ensures any data you load into the warehouse is cleansed and in the right format. It's found between the source layer and the data warehouse layer, as depicted in the image below.



Most businesses that use data marts as a server make use of the two-tier data warehouse architecture, which is also made up of two tiers:

1. The Data Tier

This is the layer where actual data is stored after various ETL processes have been used to load data into the data warehouse.

It's also made up of three layers:

- A source layer
- A data staging layer
- A data warehouse layer

2. The Client Tier

This layer is where clients can use data stored in the data warehouse to generate insights for making informed, data-driven decisions. You can modify or transform this layer based on the data trends that you discover from your analysis reports.

And it's made up of a single layer:

An analysis layer

Some disadvantages of the two-tier architecture are that it's not scalable, has network limitations, and only supports a small number of users.

III. Three-Tier Architecture

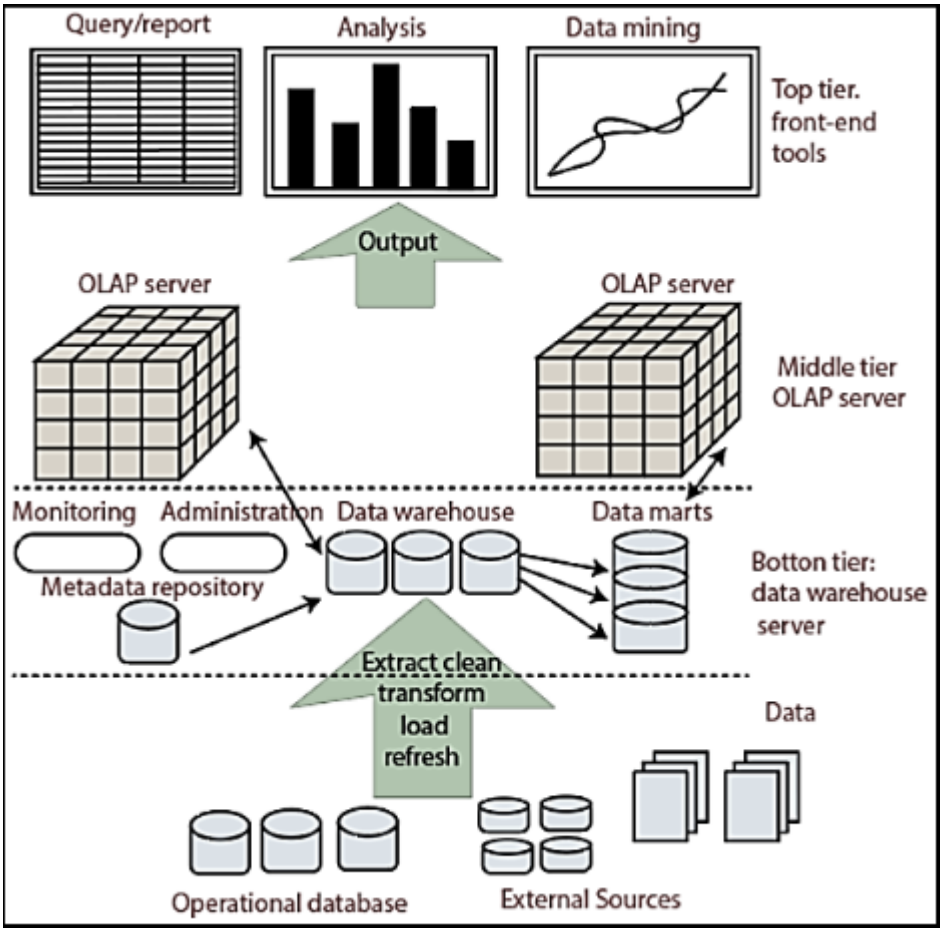
The three-tier architecture is what most organisations go for when building a data warehouse system. It solves the connectivity problems that the two-tier architecture commonly faces.

The three-tier architecture is useful for extensive, enterprise-wide systems. But its disadvantage is the additional storage space it uses through the redundant, reconciled layer.

The three-tier architecture also has three tiers:

- A bottom tier
- A top tier
- A middle tier

These three tiers are commonly called the layers of a data warehouse architecture. Let's take an in-depth look at these layers.



Layers of a Data Warehouse Architecture

1. Bottom Tier

The bottom-tier, also called the data warehouse layer, is where data is extracted, transformed and loaded into the data repository using backend tools.

2. Middle Tier

The middle tier is responsible for arranging data into a more suitable structure for complex querying and analysis. This process is done with an Online Analytical Processing (OLAP) server and it's implemented using two models:

The Relational OLAP model (also called ROLAP), which assigns multidimensional data processes to standard relational operations.

The Multidimensional OLAP (also called MOLAP) model, which implements multidimensional information and operations.

3. Top Tier

The top-tier is basically the front-end layer that houses various tools and APIs (Application Programming Interfaces) you can use for high-level data analysis, querying, reporting and data mining. It's where end-users can access, interact and extract data from the warehouse.

2.2 Relation between Business Intelligence (BI) and Data Warehousing:

Business Intelligence: Large business organizations usually receive large amounts of data from various sources. This data is always exploitable to obtain diverse sets of information that help in making better business decisions. These actionable insights may be descriptive, predictive, or prescriptive. BI represents the various methods and tools used for the collection, integration, analysis and visualization of business information. It could be considered synonymous with data analytics in particular to the business world.

Data Warehouse: Data Warehouse is a system and set of technologies at the back-end, that helps in collecting large amounts of dissimilar data from various sources and storing them for later use. Good data warehouses have business meaning backed into them facilitating future extraction and analysis. Business Intelligence is one of the applications that make use of data warehouses. Data Warehouses generally follow a multidimensional paradigm (related to OLAP) where data is held in Fact Tables (tables covering numbers such as revenue or costs) and Dimensions (things we want to view the facts by, such as region, office, or week)

Business Intelligence	Data Warehouse
It is a set of tools and methods to analyze data and discover, extract and formulate actionable information that would be useful for business decisions.	It is a system for storage of data from various sources in an orderly manner as to facilitate business-minded reads and writes.
It is a Decision Support System (DSS).	It is a data storage system.
Serves at the front end.	Serves at the back end.
The aim of business intelligence is to enable users to make	A data warehouse's main aim is to provide the users of business

Business Intelligence	Data Warehouse
informed, data-driven decisions.	intelligence; a structured and comprehensive view of available data of an organization.
Collects data from the data warehouse for analysis.	Collects data from various disparate sources and organizes it for efficient BI analysis.
Comprises business reports, charts, graphs, etc.	Comprises of data held in "fact tables" and "dimensions" with business meaning incorporated into them.
BI as such doesn't have much use without a data warehouse as large amounts of various and useful data is required for analysis.	BI is one of many use-cases for data warehouses, there are more applications for this system.
Handled by **executives and analysts** relatively higher up in the hierarchy.	Handled and maintained by **data engineers and system administrators** who report to/work for the executives and analysts.
The role of Business Intelligence lies in improving the performance of business by utilizing tools and approaches that focus on counts, statistics, and visualization.	The reflection of actual database development and integration process is given by Data Warehouse and in addition, Data Profiling and Company validation standards.
It deals with- OLAP (Online Analytical Processing) Data Visualization Data Mining Query/Reporting Tools	It deals with- Acquiring/gathering of data Metadata management Cleaning of data Transforming data Data dissemination Data recovery/backup planning
Examples of BI software: SAP, Sisense, Datapine, Looker, etc.	Examples of Data warehouse software: BigQuery, Snowflake, Amazon, Redshift, Panoply, etc.

2.3 Online Analytical Processing

Online Analytical Processing can be defined as a set of tools and approaches to represent data from multiple dimensions. In a broader sense, it includes a bunch of practices aimed at modeling data/databases and creating specific analytical solutions. OLAP systems are capable of combining classic tables in a sort of table of tables, which can be visualized as a 3D OLAP Cube for simplicity.

A typical OLAP system will include the following components that perform dedicated functions to handle analytical queries.

Data source. This could be a transactional database or any other storage we take data from. The data in its standard format isn't optimized for OLAP queries, so it requires transformation and remodeling before it can be used.

OLAP database is where we store data for analysis. Usually, transformation takes place before the data is uploaded to a database, but the approach may vary.

OLAP cube is basically a tool for representing multidimensional data for analysis. As we're talking about online analytical processing, cubes are deployed on a dedicated server.

An OLAP cube allows analytics to group or slice items by different categories. They are primarily designed to run complex queries, which can't be handled by the usual OLTP databases.

2.4 OLTP vs OLAP: technology comparison

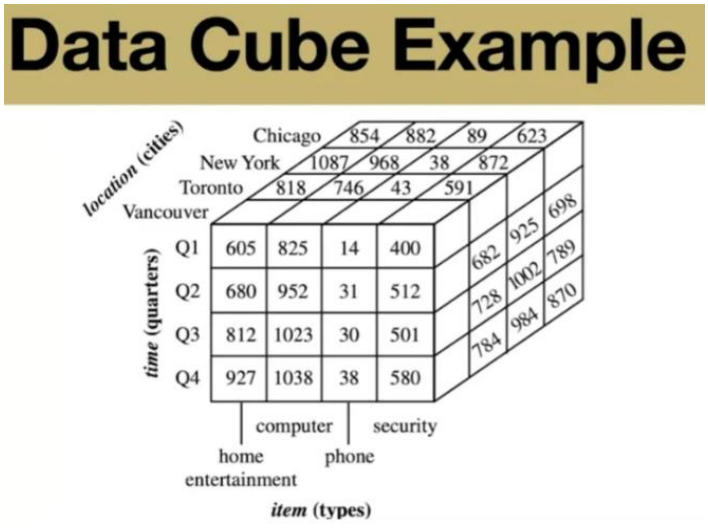
There are numerous differences between OLTP and OLAP databases in terms of purpose, information structure, and data access capabilities. The table below compares the main aspects of these two systems.

Criteria	OLAP	OLTP
Purpose	OLAP helps you analyze large volumes of data to support decision-making.	OLTP helps you manage and process real-time transactions.
Data source	OLAP uses historical and aggregated data from multiple sources.	OLTP uses real-time and transactional data from a single source.
Data structure	OLAP uses multidimensional (cubes) or relational databases.	OLTP uses relational databases.
Data model	OLAP uses star schema, snowflake schema, or other analytical models.	OLTP uses normalized or denormalized models.

Volume of data	OLAP has large storage requirements. Think terabytes (TB) and petabytes (PB).	OLTP has comparatively smaller storage requirements. Think gigabytes (GB).
Response time	OLAP has longer response times, typically in seconds or minutes.	OLTP has shorter response times, typically in milliseconds
Example applications	OLAP is good for analyzing trends, predicting customer behavior, and identifying profitability.	OLTP is good for processing payments, customer data management, and order processing.

2.5 What are Cubes?

A data cube in data mining is a multidimensional, multidimensional data structure (or array) used to represent, store, and aggregate large, complex datasets across multiple dimensions (e.g., time, product, location). It enables rapid, interactive analysis and OLAP operations—such as drilling down, rolling up, slicing, and dicing—to uncover insights, patterns, and trends, which are critical for business intelligence



Basic analytical operations of OLAP

Four types of analytical OLAP operations are:

1. Roll-up
2. Drill-down

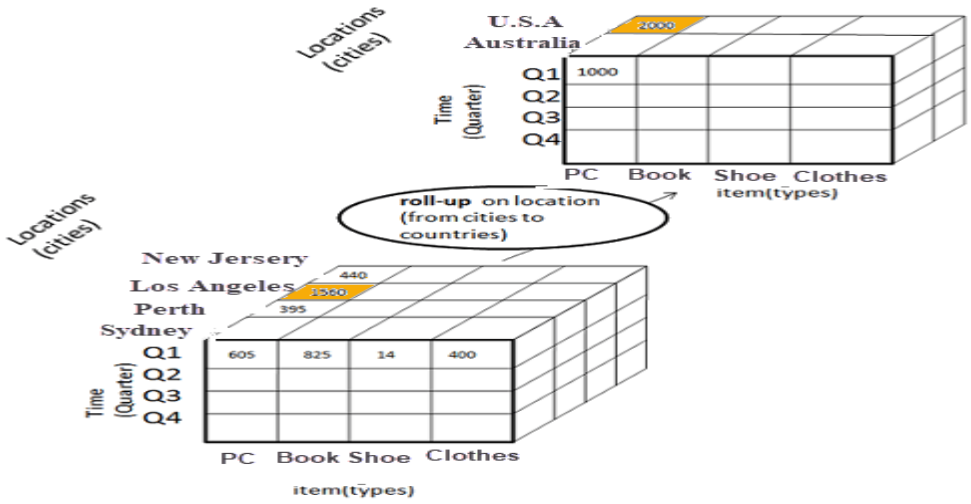
3. Slice and dice
4. Pivot (rotate)

1) Roll-up:

Roll-up is also known as "consolidation" or "aggregation." The Roll-up operation can be performed in 2 ways

- Reducing dimensions
- Climbing up concept hierarchy. Concept hierarchy is a system of grouping things based on their order or level.

Consider the following diagram



Roll up the location dimension

- In this example, cities New jersey and Lost Angles and rolled up into country USA
- The sales figure of New Jersey and Los Angeles are 440 and 1560 respectively. They become 2000 after roll-up
- In this aggregation process, data is location hierarchy moves up from city to the country.
- In the roll-up process at least one or more dimensions need to be removed. In this example, Cities dimension is removed.

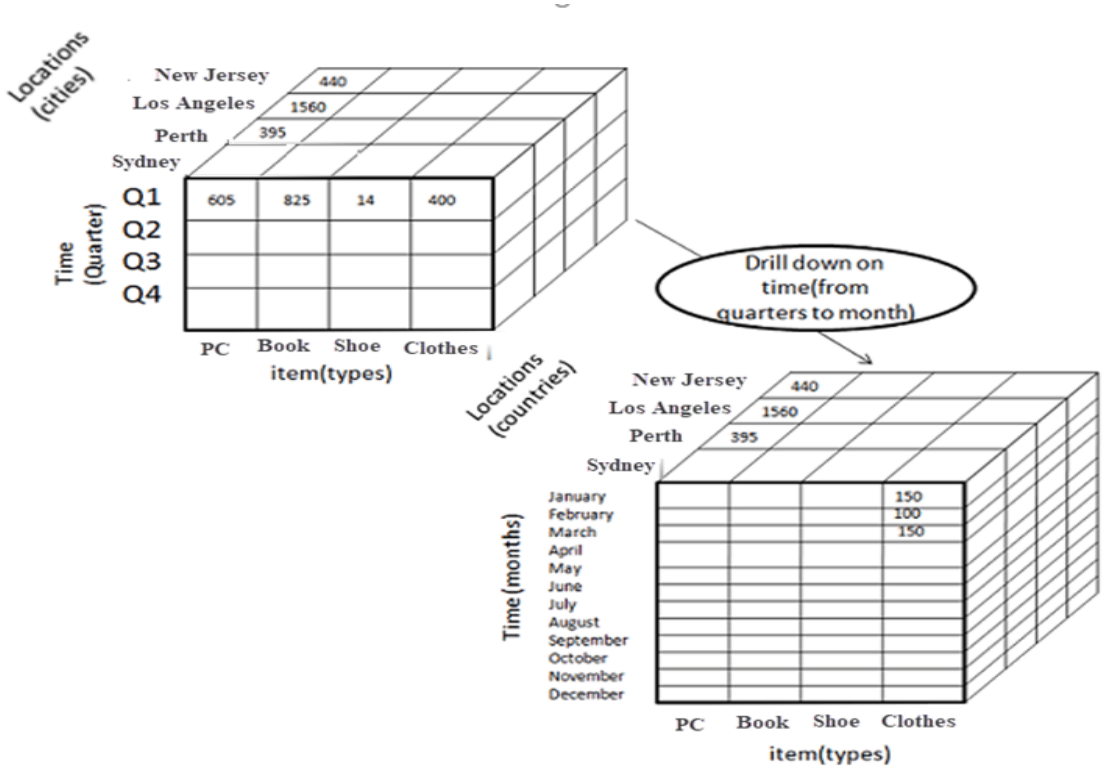
2) Drill-down

In drill-down data is fragmented into smaller parts. It is the opposite of the rollup process. It can be done via

- Moving down the concept hierarchy
- Increasing a dimension

Drill down allows a user to move from high-level data (e.g., annual sales) to a lower level (e.g., monthly sales). Here we use the concept of hierarchy that applies to every single

dimension. So, in the "time" dimension, we can move down from yearly figures to weekly or even daily records. This depends on how you store your data and model the actual cube.



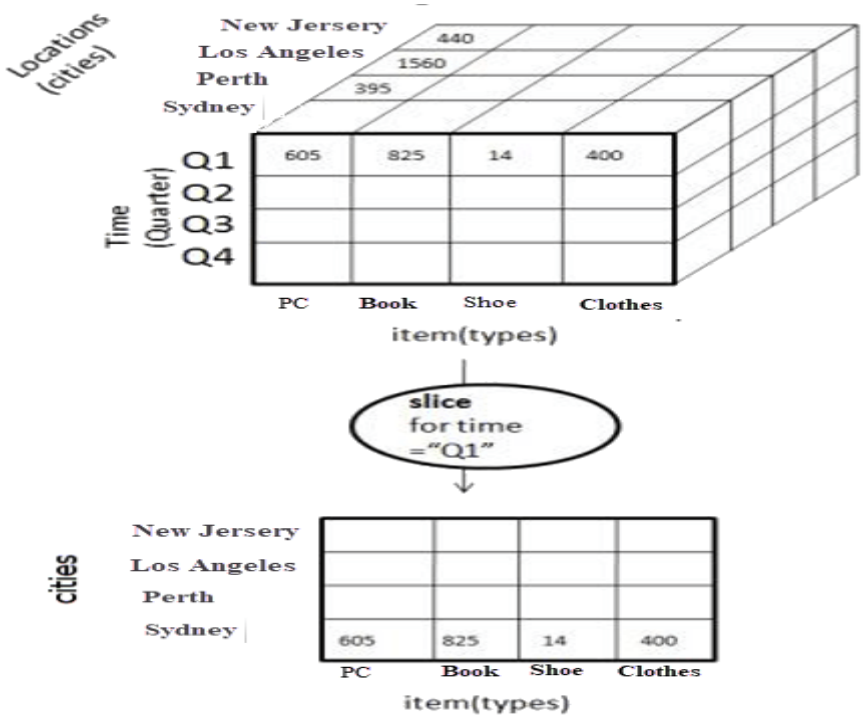
Drill-down operation in OLAP

Consider the diagram above

- Quater Q1 is drilled down to months January, February, and March. Corresponding sales are also registers.
- In this example, dimension months are added.

3) Slice:

- Here, one dimension is selected, and a new sub-cube is created.
- Following diagram explain how slice operation performed:

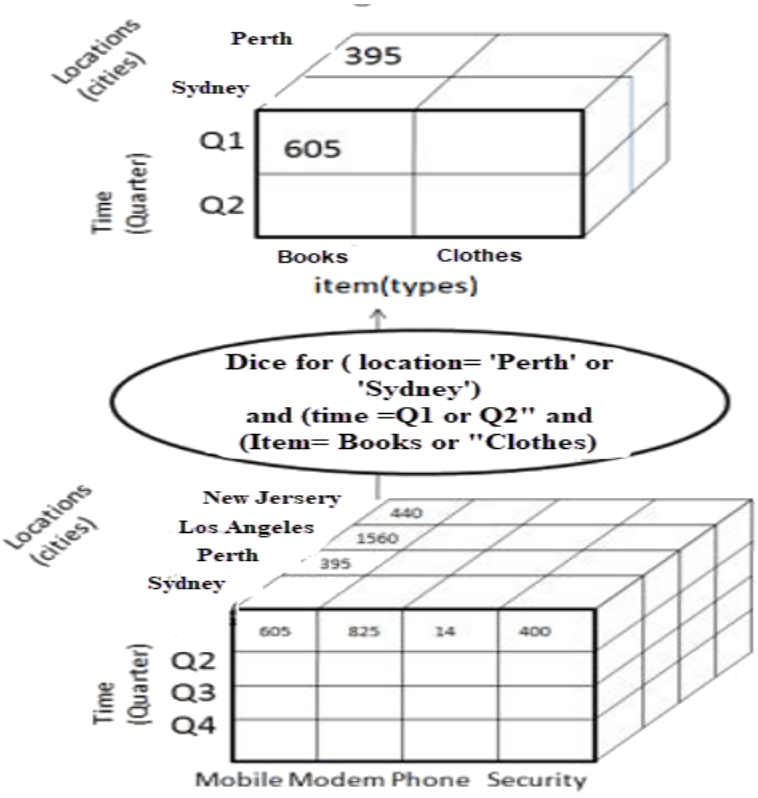


Slice operation in OLAP

- Dimension Time is Sliced with Q1 as the filter.
- A new cube is created altogether

Dice:

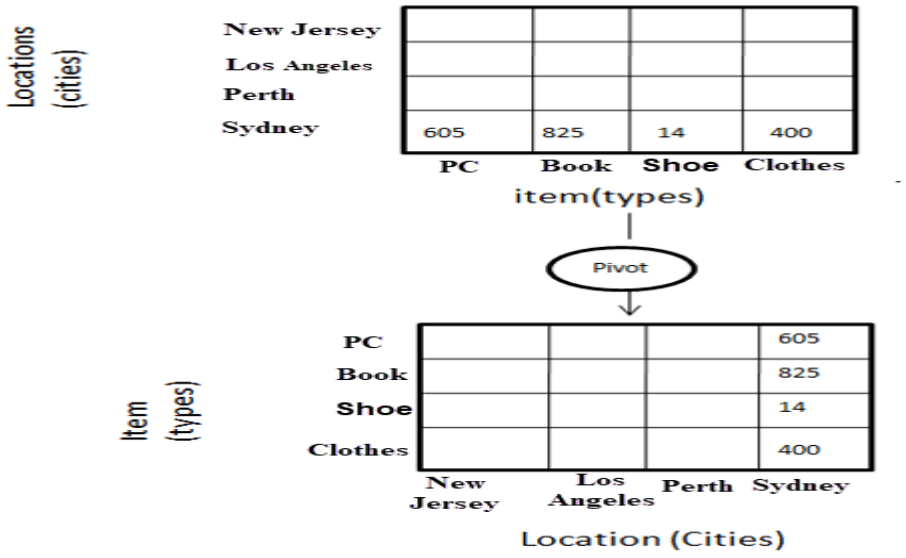
- This operation is similar to a slice. The difference in dice is you select 2 or more dimensions that result in the creation of a sub-cube.



Dice operation in OLAP

4) Pivot

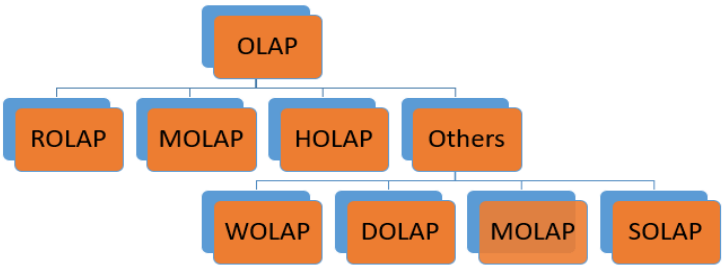
- In Pivot, you rotate the data axes to provide a substitute presentation of data.
- In the following example, the pivot is based on item types.
- Analysts can gain a new view of data by rotating the data axes of the cube.



Pivot operation in OLAP

2.5 Types of OLAP Models

These are the different types of OLAP Models:



1. Relational OLAP Models (ROLAP)
2. Multidimensional OLAP Models (MOLAP)
3. Hybrid OLAP Models (HOLAP)

Other Types

1. Web-Enabled OLAP Models (WOLAP)
2. Desktop OLAP Models (DOLAP)
3. Mobile OLAP Models (MOLAP)
4. Spatial OLAP Models (SOLAP)

1. Relational OLAP Models (ROLAP)

ROLAP stands for Relational OLAP Model, a relational OLAP application.

These are servers that sit between a relational back-end server and the user front-end tools.

They save and manage warehouse data via a relational or extended-relational database management system, and they employ OLAP middleware to fill in the gaps.

ROLAP models execute optimization for each DBMS back end, aggregate navigation logic implementation, and other tools and services.

ROLAP systems mainly use data from relational databases, where the base data and dimension tables are stored as relational tables. This model also allows for multidimensional data analysis.

This method works by modifying the data in a relational database to simulate the slicing and dicing functionality of standard OLAP. Each method of slicing and dicing is essentially the same as adding a "WHERE" clause to a SQL statement.

Advantages

Can Handle Large Volumes of Information: The data size limitation of ROLAP technology depends on the data size of the underlying RDBMS. So, ROLAP itself does not restrict the data amount.

Disadvantages

Performance may be Slow: Each ROLAP report is a SQL query (or multiple SQL queries) in a relational database, and the query time can be extended if the underlying data is substantial.

SQL Functions Limit ROLAP Technology: SQL statements are used to query a relational database, and SQL statements do not meet all needs.

2. Multidimensional OLAP Models (MOLAP)

MOLAP stands for Multidimensional OLAP Model, an application based on multidimensional DBMS.

The foundation of a MOLAP model is a native logical model that allows multidimensional data and operations directly. Data is physically stored in multidimensional arrays and accessed using positional algorithms.

The scalability of ROLAP technology is generally higher than that of MOLAP technology.

One of the key differences between **MOLAP and ROLAP** is that data is summarised and stored efficiently in a multidimensional cube rather than a relational database. Data is formatted into proprietary forms per the client's reporting requirements in the MOLAP model, with computations pre-generated on the cubes.

Advantages

Excellent Performance: MOLAP cubes are designed for quick data retrieval and are ideal for slicing and dicing activities.

Can Conduct Sophisticated Calculations: When the cube is constructed, all evaluations are pre-generated. As a result, not only are complex calculations possible, but they also return rapidly.

Disadvantages

It Can Only Handle a Certain Amount of Data: Because all calculations are done when the cube is produced, a vast quantity of data cannot be stored in the cube itself.

Additional Investment is Required: Cube technology is usually proprietary and not already in use within the company. As a result, additional human and capital resources will likely be required to implement MOLAP technology.

3. Hybrid OLAP Models (HOLAP)

HOLAP or Hybrid OLAP Model is an application that combines relational and multidimensional approaches.

HOLAP combines MOLAP and ROLAP's greatest characteristics into a single architecture. HOLAP systems store a larger amount of detailed data in relational tables, while aggregations are saved in pre-calculated cubes.

For defined data, HOLAP can drill down from the cube to the relational tables. A hybrid OLAP model is provided by Microsoft SQL Server 2000.

Advantages

HOLAP combines the advantages of MOLAP and ROLAP.

It allows quick access at all aggregate levels.

HOLAP reduces disc space requirements by storing only the aggregate data on the OLAP models while keeping the detail records in the relational database. As a result, no duplicate copy of the detail record is kept.

Disadvantages

- HOLAP architecture is somewhat complex because it supports both MOLAP and ROLAP models.

Other Types

Here is a compiled list of some of the OLAP industry's lesser-known brands.

1. Web-Enabled OLAP Models (WOLAP)
2. Desktop OLAP Models (DOLAP)
3. Mobile OLAP Models (MOLAP)
4. Spatial OLAP Models (SOLAP)

1. Web-Enabled OLAP Models (WOLAP)

WOLAP refers to an OLAP program that may be accessed through a web browser. WOLAP is a three-tiered architecture that consists of three components: a client, middleware, and database server, as opposed to standard client/server OLAP systems.

One example of this type of model in HTML solution is an OLAP tool that allows the user to execute some specific OLAP queries or reports from a browser and no other functionality would be available.

2. Desktop OLAP Models (DOLAP)

DOLAP (Desktop OLAP) Model allows a user to download a piece of data from a database or source and work with it locally or on their desktop.

3. Mobile OLAP Models (MOLAP)

MOLAP or Mobile OLAP (MOLAP) allows users to utilize their mobile devices to access and work on OLAP data and applications.

4. Spatial OLAP Models (SOLAP)

SOLAP (Spatial OLAP) combines the capabilities of both GIS and OLAP into a single user interface. It helps with both spatial and non-spatial data management.

Spatial OLAP, for example, can be used to analyze regional weather trends. Assume there are approximately 3,000 weather probes strewn across British Columbia (BC), each recording daily temperature and precipitation for a small area and transferring data to a provincial weather station.

Challenges of OLAP Models

Some disadvantages of OLAP Models are:

- Pre-modeling is required.
- High reliance on IT.
- Inadequate calculation capabilities.
- Sluggish reaction time.
- Lack of interactive analysis ability.

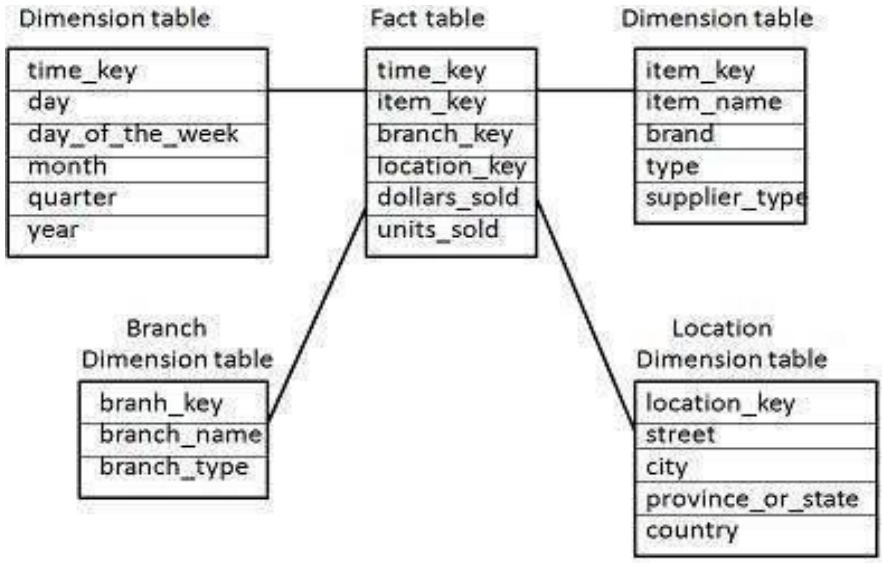
The model's abstraction prevents business workers from freely analyzing.

2.6 Warehousing - Schemas

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 maintaining 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.

1.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.



A **Star Schema** is the simplest and most widely used schema in a data warehouse. In this model, a single central **fact table** is connected directly to multiple **dimension tables**, forming a structure that resembles a star. The fact table contains quantitative data such as sales amount, quantity, profit, etc., along with foreign keys that link to the dimension tables. The dimension tables store descriptive information like product details, customer information, time,

and location. Because dimension tables are denormalized (not split into multiple smaller tables), the structure is simple and easy to understand. Star schema provides fast query performance since fewer joins are required. It is mainly used for simple business analysis such as calculating total sales by product, region, or year. However, it may lead to data redundancy because descriptive information is stored repeatedly in dimension tables.

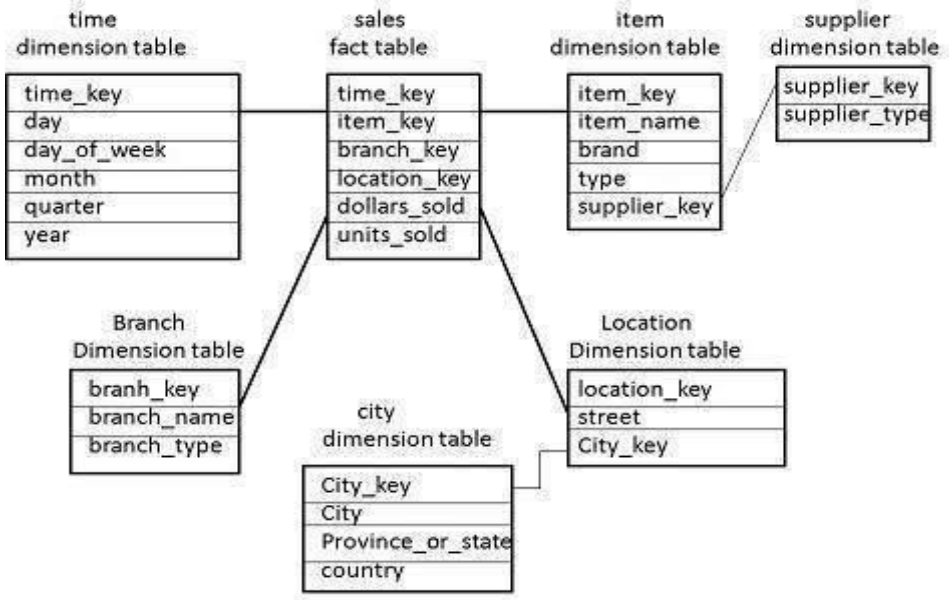
- There is a fact table at the center. It contains the keys to each of four dimensions.
- The fact table also contains the attributes, namely dollars sold and units sold.

  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.

2. Snowflake Schema

A Snowflake Schema is an advanced version of the star schema in which dimension tables are normalized into multiple related tables. Instead of keeping all descriptive attributes in one dimension table, the data is divided into sub-dimension tables to remove redundancy. For example, in a product dimension, category and brand details may be stored in separate tables. This structure creates a shape similar to a snowflake due to branching of tables. The snowflake schema reduces data redundancy and improves storage efficiency. It also maintains better data integrity through normalization. However, it requires more joins during query processing, which may reduce performance compared to star schema. Snowflake schema is suitable when dimension data is large and complex.

- 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 twodimension tables, namely item and supplier table.



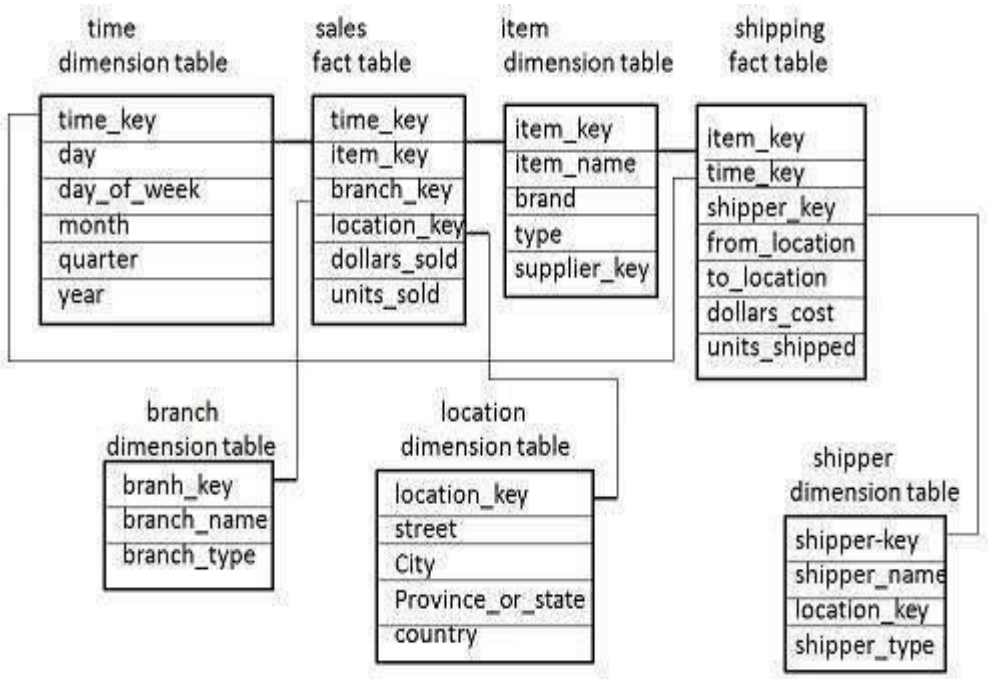
- Now the item dimension table contains the attributes item_key, item_name, type, brand,and supplier-key.
- The supplier key is linked to the supplier dimension table. The supplier dimension tablecontains the attributes supplier_key and supplier_type.

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

3. Fact Constellation Schema

A Fact Constellation Schema, also known as a Galaxy Schema, is a complex data warehouse schema that contains multiple fact tables sharing common dimension tables. Unlike star and snowflake schemas, which usually have only one fact table, fact constellation supports multiple business processes within the same database. For example, a retail company may have separate fact tables for sales, returns, and inventory, but they all share common dimensions such as product, time, and customer. This design allows comprehensive analysis across different processes. Fact constellation schema is mainly used in large enterprises where multiple departments require integrated analysis. Although it provides powerful analytical capability, it is complex to design and maintain.

- 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, and location dimension tables are shared between the sales and shipping fact table.

UNIT HIGHLIGHTS

- Business Intelligence (BI) systems support **strategic decision-making** using data analysis.
- Data Warehouse acts as a **centralized repository for integrated and historical data**.
- Data is collected from **multiple heterogeneous sources** such as ERP, CRM, databases, and web applications.
- The **ETL process (Extract, Transform, Load)** is used to integrate and prepare data for analysis.
- Data warehouses store **subject-oriented, time-variant, and non-volatile data**.
- **Data marts** provide department-specific data for faster analysis.
- **OLAP servers** enable multidimensional analysis using data cubes.
- OLAP operations include **drill-down, roll-up, slice, and dice**.
- The **BI presentation layer** provides reports, dashboards, and visualization tools.
- BI tools help managers identify **patterns, trends, and insights for better decisions**.

CASE STUDY

Retail Business Intelligence System

A large **retail supermarket chain** collects sales data from multiple stores daily. The data includes information about **products, customers, sales transactions, and store locations**.

1. The data from different store databases is collected through the **ETL process**.
2. The ETL system **extracts, cleans, and integrates the data** before loading it into the **data warehouse**.
3. The organization creates **data marts for departments** such as sales, marketing, and inventory.
4. Using **OLAP tools**, managers analyze product sales across different regions and time periods.
5. BI dashboards display **sales trends, customer preferences, and product performance**.

Book References

1. **Jiawei Han, Micheline Kamber, and Jian Pei** – *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers.
2. **Alex Berson and Stephen J. Smith** – *Data Warehousing, Data Mining, and OLAP*, McGraw-Hill.
3. **Ralph Kimball and Margy Ross** – *The Data Warehouse Toolkit*, Wiley Publications.
4. **Larissa T. Moss and Shaku Atre** – *Business Intelligence Roadmap*, Addison-Wesley.
5. **Paulraj Ponniah** – *Data Warehousing Fundamentals*, Wiley India.

UNIT- II: BI AND DW ARCHITECTURES		
Part - A		
1	What is the full form of OLAP?	L1
2	Define OLTP.	L1
3	What is a data cube in OLAP?	L2
4	Write one difference between OLAP and OLTP.	L2
5	What is dimensional analysis?	L1
6	Define drill-down and roll-up operations in OLAP.	L1
7	What is slicing and dicing in OLAP?	L2
8	Define ROLAP and MOLAP.	L2
9	What is a star schema?	L1
10	What is a fact constellation schema?	L1
11	What is the full form of OLAP?	L1
Part – B		
1	Explain the BI and DW architectures and describe various types.	L2,L4
2	Discuss the relationship between Business Intelligence and Data Warehousing with examples.	L2,L3
3	Define OLAP. Differentiate between OLAP and OLTP with suitable examples.	L2,L4
4	Explain dimensional analysis in detail with the help of a data cube example.	L3,L4
5	Describe OLAP operations: drill-down, roll-up, slice, dice, and rotation with suitable illustrations.	L3,L4
6	Compare ROLAP and MOLAP. Mention advantages and disadvantages of each.	L4
7	Describe various OLAP models in detail.	L2,L4
8	Explain different types of schemas in data warehousing: star, snowflake, and fact constellation with diagrams.	L2,L4

UNIT- 3

ISSUES IN DATAMINING – KDD PROCESS

"The goal is to turn data into information and information into insight." – Carly Fiorina

Overview

Data Mining is the process of discovering useful patterns and knowledge from large volumes of data. It helps organizations analyze data and support effective decision-making. This unit explains the concepts of data mining, its functionalities, and classification of data mining systems. It also discusses the issues in data mining and the steps involved in the Knowledge Discovery in Databases (KDD) process.

Objectives

- To understand the **concept and motivation of data mining**.
- To study **data mining functionalities and different mining tasks**.
- To analyze the **classification of data mining systems** and their techniques.
- To understand **data mining task primitives and integration with databases/data warehouses**.
- To identify **major issues in data mining and the steps involved in the KDD process**.

Learning Outcomes

After completing this unit, students will be able to:

- **Define data mining** and explain its major functionalities.
- **Describe important data mining techniques** such as classification, clustering, and association analysis.
- **Explain the classification of data mining systems** and data mining task primitives.
- **Analyze the integration of data mining with database and data warehouse systems**.
- **Explain issues in data mining and the steps of the KDD process**.

Importance of Studying this Unit

- Helps students understand how **large datasets can be analyzed to discover useful knowledge**.
- Provides insights into **data-driven decision-making techniques** used in modern organizations.
- Enhances understanding of **data mining algorithms and knowledge discovery processes**.
- Enables students to learn about **real-world applications of data mining in business and industry**.
- Prepares students to handle **complex data analysis tasks in research and industry**.

Key Terminologies

- Data Mining
- Classification
- Clustering
- Association Rule Mining
- Outlier Analysis
- Evolution Analysis

- KDD (Knowledge Discovery **in Databases)**

DATA MINING

Data Mining is the process **of discovering interesting patterns** (or Knowledge) from large amounts of data.

Data sources can include databases, data warehouses, the Web, other information repositories, or data that are streamed into the system dynamically.

Data Mining refers to extracting or "Mining" Knowledge from large amount of data.

3.1 Data Mining Functionalities- What kinds of patterns can be mined?

Data mining functionalities are used to represent the type of patterns that have to be discovered in **data mining tasks**. In general,

Data mining tasks can be classified into **two types** including

1.**Descriptive** - Descriptive task Characterize the general properties of the data in the database.

2. **Predictive** - Predictive mining tasks act inference on the current information to develop predictions.

Data mining functionalities, and the kinds of patterns they can discover,are described below:

1. Class/Concept Descriptions : Characterization and Discrimination
2. Mining Frequent Patterns, Associations, and Correlations
3. Classification and Predictions
4. Cluster analysis
5. Outlier Analysis
6. Evolution Analysis

1. Class/Concept Descriptions : Characterization and Discrimination:

Data can be associated with **classes or concepts**, concept refers to a collection of data items such as Computers,printers etc.

Concepts of Customers-bigSpenders, budgetSpenders,...

How to describe these items or concepts?

Descriptions can be derived as

- **Data Characterization**
- **Data Discrimination**
- **Or both of the Data Characterization & Data Discrimination**

Data Characterization: This refers to the summarizing the general characteristics of a target class of data. The output of the data characterization can be presented in various forms include pie charts, bar charts, curves, multidimensional data cubes.

Example: To study the characteristics of software products with sales increased by 10% in the previous years. To summarize the characteristics of the customer who spend more than $5000 a year at AllElectronics, the result is general profile of those customers such as that they are 40-50 years old, employee and have excellent credit rating.

Data Discrimination: It comparing the target class with one or set of classes. It is a comparison of the general features of the target class data objects against the general features of objects from one or multiple contrasting classes.

Example: Compare the general features of software products, whose sales increase by 10% in the last year with those whose sales decrease by 30% during the same period

2. Mining Frequent Patterns, Associations, and Correlations: Frequent patterns are nothing but things that are found to be most common in the data. There are different kinds of frequencies that can be observed in the dataset.

- **Frequent item set:** This applies to a number of items that can be seen together regularly for eg: milk and sugar.
- **Frequent Subsequence:** This refers to the pattern series that often occurs regularly such as purchasing a phone followed by a back cover.
- **Frequent Substructure:** It refers to the different kinds of data structures such as trees and graphs that may be combined with the itemset or subsequence.

Association Analysis: The process involves uncovering the relationship between data and deciding the rules of the association. It is a way of discovering the relationship between various items.

Example: Suppose we want to know which items are frequently purchased together. An example for such a rule mined from a transactional database is,
buys (X, "computer") ⇒ buys (X, "software") [support = 1%, confidence = 50%],
where X is a variable representing a customer. A confidence, or certainty, of 50% means that if a customer buys a computer, there is a 50% chance that she will buy software as well. A 1% support means that 1% of all the transactions under analysis show that computer and software are purchased together.

Correlation Analysis: Correlation is a mathematical technique that can show whether and how **strongly the pairs of attributes are related to each other**. For example, Highted people tend to have more weight.

3. Classification and Predictions

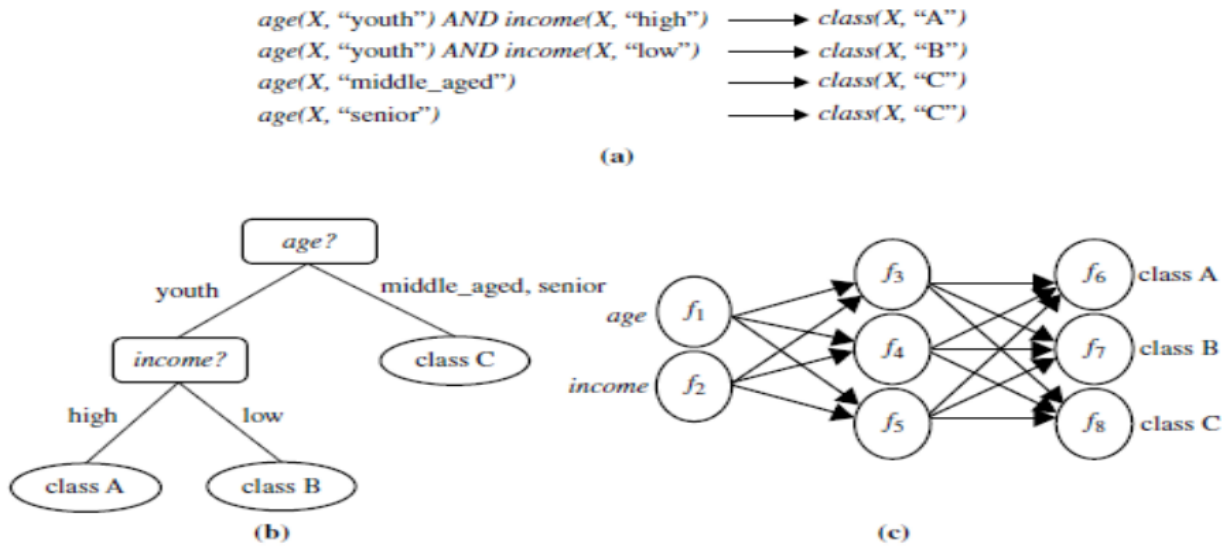
Classification

- The process of finding a model that describes and distinguishes the data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown.

- The derived model is based on the analysis of a set of training data(data objects whose class label is known)
- The model can be represented in Classification (IF-THEN) rules, Decision tree, Neural networks,etc.,

A decision tree is a flowchart-like tree structure, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions. Decision trees can easily be converted to classification rules.

A neural network, when used for classification, is typically a collection of neuron-like processing units with weighted connections between the units. There are many other methods for constructing classification models, such as naive Bayesian classification, Support vector machines, and k-nearest-neighbor classification. Whereas classification predicts categorical (discrete, unordered) labels, regression models continuous-valued functions. Regression analysis is a statistical methodology that is most often used for numeric prediction, although other methods exist as well.

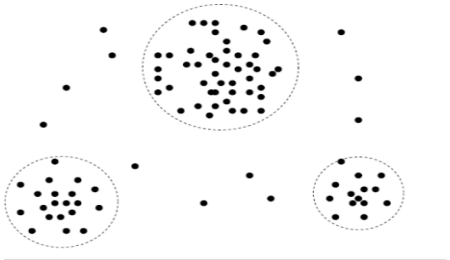


A classification model can be represented in various forms: (a) IF-THEN rules, (b) a decision tree, or (c) a neural network.

- **Prediction** – It defines predict some unavailable data values or pending trends. An object can be anticipated based on the attribute values of the object and attribute values of the classes. It can be a prediction of missing numerical values or increase/decrease trends in time-related information.

4. Cluster analysis

Class label is unknown group data to form new classes. Clusters of objects are formed based on the principle of **maximizing intra-class** similarity and **minimizing interclass** similarity Unlike classification and



A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters

regression, which analyze class-labeled (training) data sets, clustering analyzes data objects without consulting class labels. In many cases, class labeled data may simply not exist at the beginning. Clustering can be used to generate class labels for a group of data.

5. Outlier Analysis

A data set may contain objects that do not comply with the general behavior or model of the data. These data objects are outliers. Many data mining methods discard outliers as **noise** or exceptions.

However, in some applications (e.g., fraud detection) the rare events can be more interesting than the more regularly occurring ones. The analysis of outlier data is referred to as outlier analysis or anomaly mining. For example, Outlier analysis. Outlier analysis may uncover fraudulent usage of credit cards by detecting purchases of unusually large amounts for a given account number in comparison to regular charges incurred by the same account. Outlier values may also be detected with respect to the locations and types of purchase, or the purchase frequency .

6. Evolution analysis

Evolution analysis is the study of data sets that may have undergone a stage of transformation or change.

Evolution analysis describes and models regularities or trends for objects whose behavior changes over time

It provides time-related data clustering and assists in finding trends or changes with features like periodicity, time-series data, and trend similarity.

In addition to aiding in data classification, characterisation, discrimination, and grouping for multivariate time series, the evolution analysis model represents evolving trends in data.

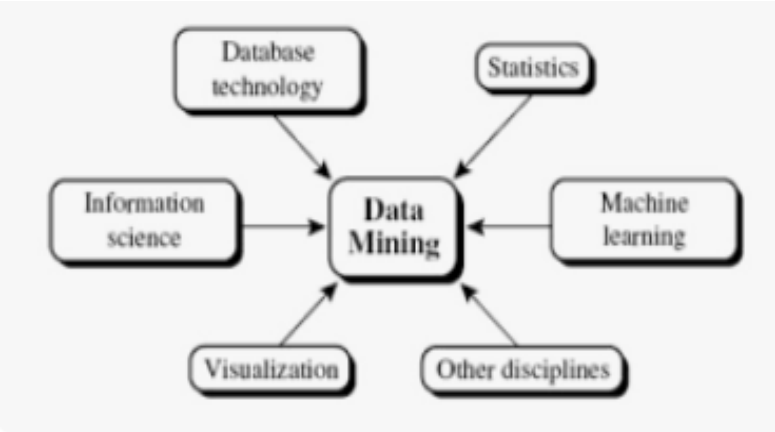
Example: Identify Stock evolution regularities for overall stocks and for the stocks of particular companies.

3.2 Classification of Data mining Systems

Data Mining is considered as an interdisciplinary field. It includes a set of various disciplines such as statistics, database systems, machine learning, visualization and information sciences.

Classification of the data mining system helps users to understand the system and match their requirements with such systems.

The data mining system can be classified according to the following criteria:



Some Other Classification Criteria:

- Classification according to kind of databases mined
- Classification according to kind of knowledge mined
- Classification according to kinds of techniques utilized
- Classification according to applications adapted

Classification according to kind of databases mined

We can classify the data mining system according to kind of databases mined. Database system can be classified according to different criteria such as data models, types of data etc. And the data mining system can be classified accordingly. For example if we classify the database according to data model then we may have a relational, transactional, object- relational, or data warehouse mining system.

- Relational
- Transactional
- Object-relational
- Data warehouse mining system

Classification according to kind of knowledge mined

We can classify the data mining system according to kind of knowledge mined. It is means data mining system are classified on the basis of functionalities such as:

- Data Characterization & Data Discrimination
- Association and Correlation Analysis
- Classification
- Prediction
- Clustering
- Outlier Analysis
- Evolution Analysis

Classification according to kinds of techniques utilized

We can classify the data mining system according to kind of techniques used. We can describes these techniques according to degree of user interaction involved or the methods of analysis employed.

Classification according to applications adapted

We can classify the data mining system according to application adapted. These applications are as follows:

- Finance
- Telecommunications
- DNA
- Stock Markets
- E-mail

3.3 Data mining Task primitives

A data mining task can be specified in the form of a data mining query, which is input to the data mining system. A data mining query is defined in terms of data mining task primitives. These primitives allow the user to interactively communicate with the data mining system during the mining process to discover interesting patterns.

Here is the list of Data Mining Task Primitives

- Set of task relevant data to be mined.
- Kind of knowledge to be mined.
- Background knowledge to be used in discovery process.
- Interestingness measures and thresholds for pattern evaluation.
- Representation for visualizing the discovered patterns.

Primitives for specifying a data mining task

- Task-relevant data: This primitive specifies the data upon which mining is to be performed.
- It involves specifying– the database and tables or data warehouse

containing the relevant data, conditions for selecting the relevant data, the relevant attributes or dimensions for exploration, and instructions regarding the ordering or grouping of the data retrieved.

Knowledge type to be mined: This primitive specifies the specific data mining function to be performed, such as characterization, discrimination, association, classification, clustering, or evolution analysis.

- As well, the user can be more specific and provide pattern templates that all discovered patterns must match. These templates or meta patterns (also called meta rules or meta queries), can be used to guide the discovery.

Background knowledge: This primitive allows users to specify knowledge they have about the domain to be mined.

- Such knowledge can be used to guide the knowledge discovery process and evaluate the patterns that are found.

- The several kinds of background knowledge, this chapter focuses on concept hierarchies.

Pattern interestingness measure: This primitive allows users to specify functions that are used to separate uninteresting patterns from knowledge and may be used to guide the mining process, as well as to evaluate the discovered patterns.

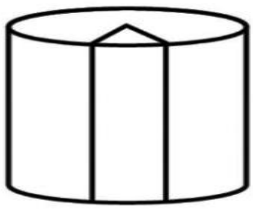
- This allows the user to confine the number of uninteresting patterns returned by the process, as a data mining process may generate a large number of patterns

Interestingness measures can be specified for such pattern characteristics as simplicity, certainty, utility and novelty.

- **Visualization of discovered patterns:** This primitive refers to the form in which discovered patterns are to be displayed.

- In order for data mining to be effective in conveying knowledge to users, data mining systems should be able to display the discovered patterns in multiple forms such as rules, tables, cross tabs (cross-tabulations), pie or bar charts, decision trees, cubes or other visual representations.

A data mining query language can be designed to incorporate these primitives, allowing users to flexibly interact with data mining systems. Having a data mining query language provides a foundation on which user-friendly graphical interfaces can be built.



Task-relevant data

Database or data warehouse name

Database tables or data warehouse cubes

Conditions for data selection

Relevant attributes or dimensions

Knowledge type to be mined

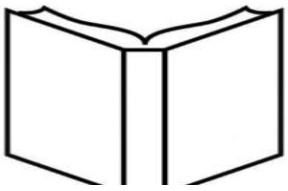
Characterization & Discrimination

Association

Classification

prediction

Clustering



Background knowledge

Concept hierarchies

User beliefs about relationships in the data

Pattern interestingness measures

Simplicity

Certainty (e.g., confidence)

Utility (e.g., support)

Novelty

Visualization of discovered patterns

Rules, tables, reports,

charts, graphs,

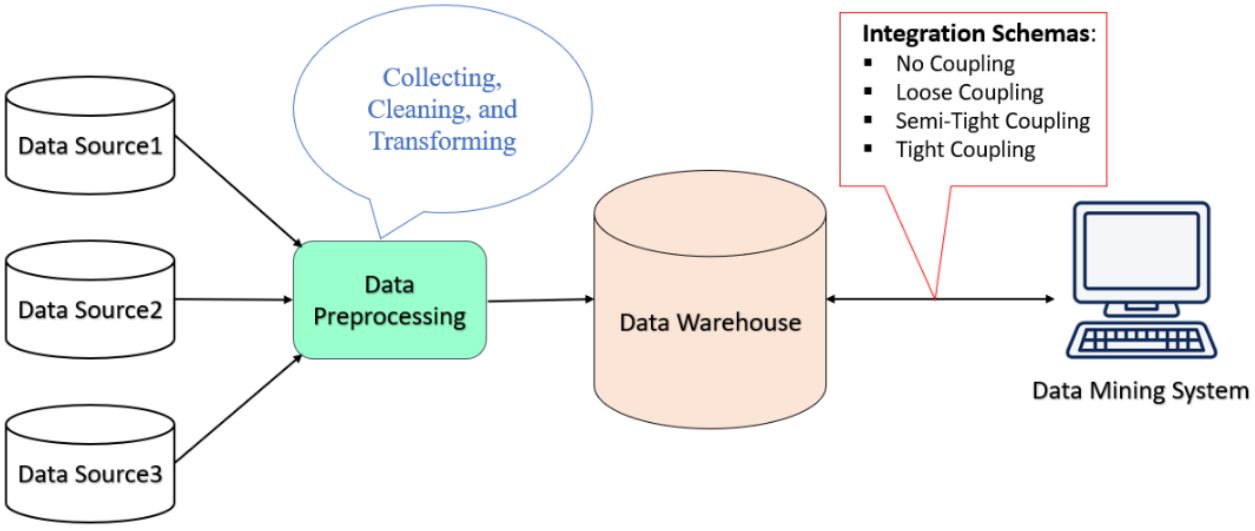
decision trees, and cubes

3.4 Integration of Data mining system with a Data warehouse

The data mining system is **integrated** with a database or data warehouse system so that it can do its **tasks in an effective mode**. A data mining system operates in an environment that needs to **communicate** with other data systems like a Database or Datawarehouse system.

There are different possible integration (coupling) schemes as follows:

1. No Coupling
2. Loose Coupling
3. Semi-Tight Coupling
4. Tight Coupling



1. No Coupling

No coupling means that a Data Mining system will not utilize any function of a Data Base or Data Warehouse system.

It may fetch data from a particular source (such as a file system), process data using some data mining algorithms, and then store the mining results in another file.

Drawbacks of No Coupling

First, without using a Database/Data Warehouse system, a Data Mining system may spend a substantial amount of time finding, collecting, cleaning, and transforming data.

Second, there are many tested, Scalable algorithms and data structures implemented in Database and Data Warehouse systems.

2. Loose Coupling

In this Loose coupling, the data mining system uses some facilities / services of a database or data warehouse system. The data is fetched from a data repository managed by these (DB/DW) systems.

Data mining approaches are used to process the data and then the processed data is saved either in a file or in a designated area in a database or data warehouse.

Loose coupling is better than no coupling because it can fetch any portion of data stored in Databases or Data Warehouses by using query processing, indexing, and other system facilities.

Drawbacks of Loose Coupling

It is difficult for loose coupling to achieve high scalability and good performance with large data sets.

3. Semi-Tight Coupling

Semi-tight coupling means that besides linking a Data Mining system to a Data Base/Data Warehouse system, efficient implementations of a few essential data mining primitives can be provided in the DB/DW system. These primitives can include (**sorting, indexing, aggregation, histogram analysis, multi way join, and pre computation of some essential statistical measures, such as sum, count, max, min, standard deviation).**

Advantage of Semi-Tight Coupling

This Coupling will enhance the performance of Data Mining systems

4. Tight Coupling

Tight coupling means that a Data Mining system is smoothly integrated into the Data Base/Data Warehouse system. The data mining subsystem is treated as one functional component of information system. Data mining **queries and functions** are optimized based on **mining query analysis, data structures, indexing schemes, and query processing methods of a DB or DW system.**

- Issues in DM

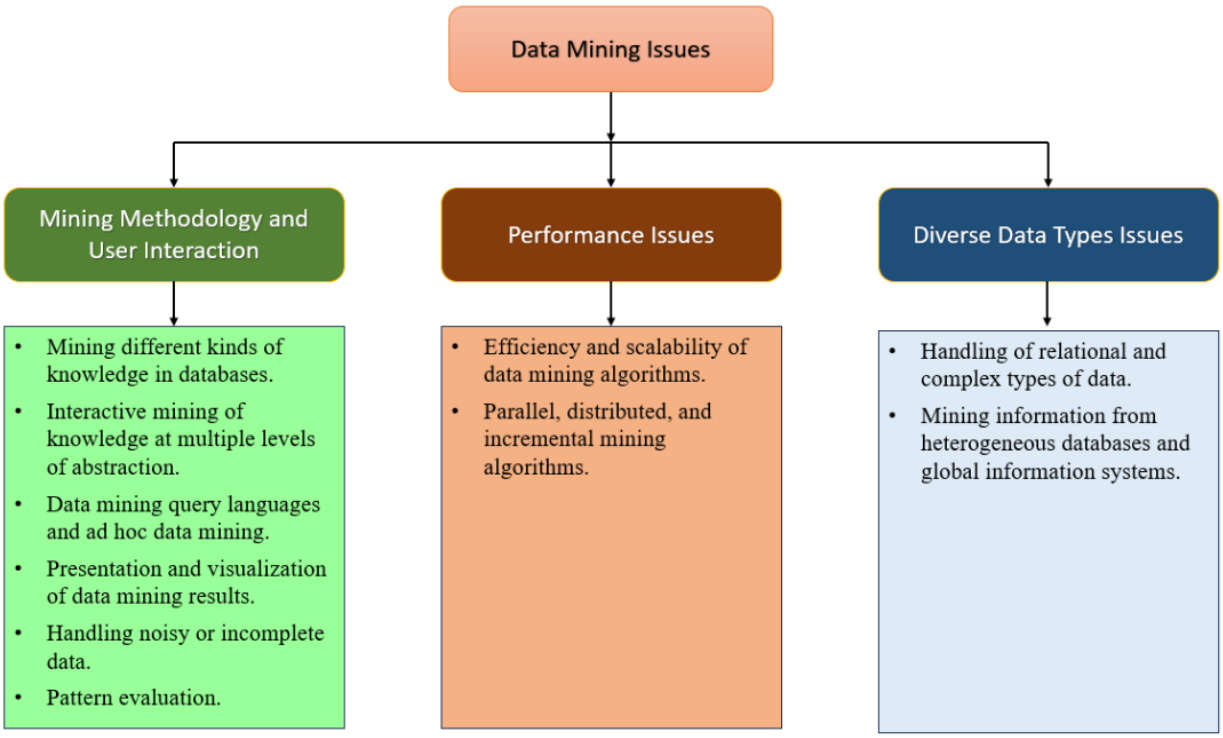
3.5 Major issues in Data Mining

Data mining, the process of extracting knowledge from data, has become increasingly important as the amount of data generated by individuals, organizations, and machines has grown

exponentially. Data mining is not an easy task, as the algorithms used can get very complex and data is not always available at one place. It needs to be integrated from various heterogeneous data sources.

The above factors may lead to some issues in data mining. These issues are mainly divided into three categories, which are given below:

1. **Mining Methodology and User Interaction**
2. **Performance Issues**
3. **Diverse Data Types Issues**



Mining Methodology and User Interaction

It refers to the following kinds of issues

- **Mining different kinds of knowledge in databases** – Different users may be interested in different kinds of knowledge. Therefore, it is necessary for data mining to cover a broad range of knowledge discovery task.
- **Interactive mining of knowledge at multiple levels of abstraction** – The data mining process needs to be interactive because it allows users to focus the search for patterns, providing and refining data mining requests based on the returned results.
- **Data mining query languages and ad hoc data mining** – Data Mining Query language that allows the user to describe ad hoc mining tasks, should be integrated with a data warehouse query language and optimized for efficient and flexible data mining.

- **Presentation and visualization of data mining results** – Once the patterns are discovered it needs to be expressed in high level languages, and visual representations. These representations should be easily understandable.
- **Handling noisy or incomplete data** – The data cleaning methods are required to handle the noise and incomplete objects while mining the data regularities. If the data cleaning methods are not there then the accuracy of the discovered patterns will be poor.
- **Pattern evaluation** – The patterns discovered should be interesting because either they represent common knowledge or lack novelty.

Performance Issues

There can be performance-related issues such as follows

- **Efficiency and scalability of data mining algorithms** – In order to effectively extract the information from huge amount of data in databases, data mining algorithm must be efficient and scalable.
- **Parallel, distributed, and incremental mining algorithms** – The factors such as huge size of databases, wide distribution of data, and complexity of data mining methods motivate the development of parallel and distributed data mining algorithms. These algorithms divide the data into partitions which is further processed in a parallel fashion. The incremental algorithms, update databases without mining the data again from scratch.

Diverse Data Types Issues

- **Handling of relational and complex types of data** – The database may contain complex data objects, multimedia data objects, spatial data, temporal data etc. It is not possible for one system to mine all these kinds of data.
- **Mining information from heterogeneous databases and global information systems** – The data is available at different data sources on LAN or WAN. These data source may be structured, semi structured or unstructured. Therefore, mining the knowledge from them adds challenges to data mining.

3.6 Knowledge Discovery from Data (KDD)

The need of data mining is to extract useful information from large datasets and use it to make predictions or better decision-making. Nowadays, data mining is used in almost all places where a large amount of data is stored and processed.

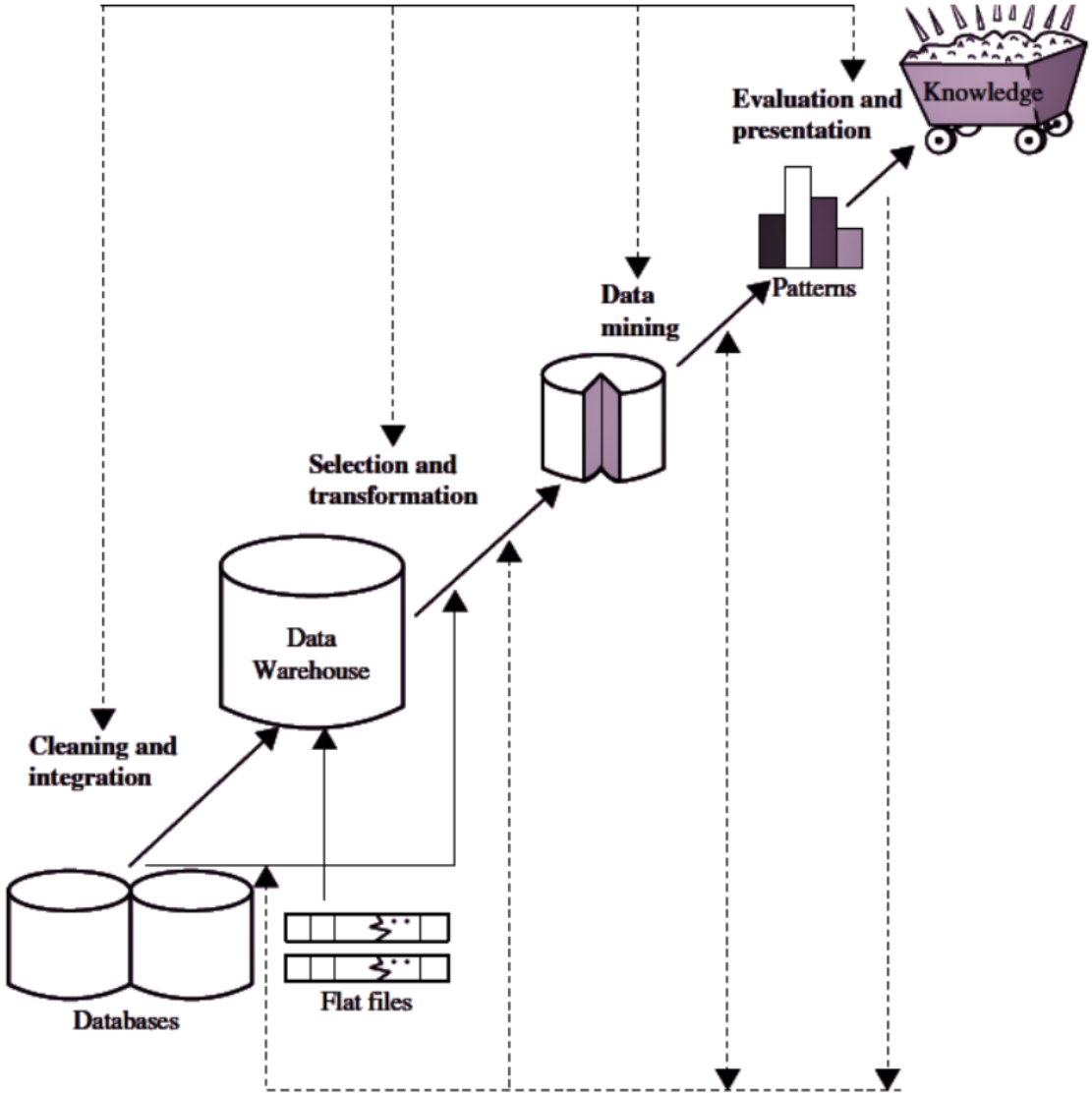
For examples: Banking sector, Market Basket Analysis, Network Intrusion Detection.

Data Mining also known as **Knowledge Discovery from Data or KDD**.

Knowledge Discovery from Data (KDD) Process

KDD is a process that involves the extraction of useful, previously unknown, and potentially valuable information from large datasets.

The KDD process is an iterative process and it requires multiple iterations of the above steps to extract accurate knowledge from the data.



The following steps are included in KDD process:

1. **Data Cleaning**
2. **Data Integration**
3. **Data Selection**
4. **Data Transformation**
5. **Data Mining**
6. **Pattern Evaluation**
7. **Knowledge Representation**

1. Data Cleaning

Data cleaning is defined as removal of **noisy** and **irrelevant/ inconsistent** data from data collection.

Cleaning in case of **Missing values**.

- Cleaning **noisy data**, where noise is a **random** or **variance error**.
- Cleaning with Data Discrepancy Detection and Data Transformation Tools.

2. Data Integration

Data integration is defined as heterogeneous **data from multiple data sources** combined in a common source (Data Warehouse).

i.e., In this step, multiple data sources may be combined as single data source.

Data integration using

- Data Migration tools
- Data Synchronization tools
- ETL(Extract-Load-Transformation) process.

A popular trend in the information industry is to perform **data cleaning** and **data integration** as a **data pre processing** step, where the resulting data are stored in a **data warehouse**.

3. Data Selection

Data selection is defined as the process where **data relevant to the analysis** is decided and retrieved from the data collection.

This step in the KDD process is **identifying** and **selecting** the relevant data for analysis.

- Decision Trees
- Naïve Bayes
- Regression
- Clustering
- Neural Network

4. Data Transformation

Data Transformation is defined as the process of **transforming data into appropriate form** required by mining procedure. This step involves reducing the data dimensionality, aggregating the data, normalizing it, and discretizing it to prepare it for further analysis.

Data Transformation is a Two-step process:

- **Data Mapping:** Assigning elements from source base to destination to capture transformation.
- **Code Generation:** Creation of the actual data transformation program.

5. Data Mining

This is the heart of the KDD process and involves applying **various data mining techniques** to the transformed data to discover **hidden patterns, trends, relationships**, and **insights**.

A few of the most common data mining techniques include clustering, classification, association rule mining, and anomaly detection.

6. Pattern Evaluation

After the data mining, the next step is to evaluate the **discovered patterns** to determine their usefulness and relevance. This involves assessing the quality of the patterns, evaluating their significance, and selecting the most promising patterns for further analysis.

Pattern Evaluation: Pattern Evaluation is defined as identifying patterns representing knowledge based on given measures.

- Find interestingness score of each pattern.
- Uses Summarization and Visualization to make data understandable by user.

7. Knowledge Representation

This step involves **representing the knowledge** extracted from the data in a way humans can easily understand and use.

Knowledge Representation: Knowledge representation is defined as technique which utilizes visualization tools to represent data mining results.

- Reports
- Tables
- Discriminant rules
- Classification rules
- Characterization rules,etc.,

Unit Highlights

- Introduction to **Data Mining concepts and motivation**.
- Study of **data mining functionalities** such as classification, clustering, and association analysis.
- Understanding the **classification of data mining systems**.
- Explanation of **data mining task primitives** and their role in knowledge discovery.
- Integration of **data mining systems with databases and data warehouses**.
- Discussion of **major issues and challenges in data mining**.
- Understanding the **steps involved in the KDD (Knowledge Discovery in Databases) process**.

Case Study (Retail Market Basket Analysis)

A retail supermarket collects transaction data from thousands of customers every day. By applying **data mining techniques**, the store analyzes which products are frequently purchased together. For example, the analysis may show that customers who buy **bread often purchase butter and jam**. Using this information, the supermarket places these items close to each other and offers promotional discounts. This improves **sales, customer satisfaction, and business decision-making**.

Book References

1. **Jiawei Han, Micheline Kamber, and Jian Pei** – *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers.
2. **Margaret H. Dunham** – *Data Mining: Introductory and Advanced Topics*, Pearson Education.
3. **Arun K. Pujari** – *Data Mining Techniques*, Universities Press.

UNIT- III		
Part - A		
1	What is data mining?	L1
2	Mention any two functionalities of data mining.	L1
3	List any two issues in data mining.	L1
4	Define KDD (Knowledge Discovery in Databases).	L1
5	What is the motivation for data mining?	L2
6	Mention two types of data mining systems.	L1
7	Define data mining task primitives.	L1
8	Write any two steps in the KDD process.	L2
9	What is the role of a data warehouse in data mining?	L2
10	State any one challenge in integrating data mining with a database	L2
Part – B		
1	Explain the motivation for data mining and discuss why it is important in modern organizations.	L2,L4
2	Define data mining. Explain its major functionalities with examples.	L1,L2
3	Classify data mining systems based on various criteria and explain each type.	L2,L4
4	What are data mining task primitives? Describe their role in specifying a data mining query.	L2,L3
5	Explain the integration of data mining systems with databases or data warehouses.	L2,L3
6	Discuss various issues and challenges in data mining.	L4,L5
7	What is the KDD process? Explain each step in the Knowledge Discovery in Databases process in detail.	L2,L4
8	Compare and contrast data mining and KDD. Explain how data mining fits into the KDD process.	L4,L5

UNIT- 4

DATA PRE-PROCESSING

Data is a precious thing and will last longer than the systems themselves." – Tim Berners-Lee

Overview

Data preprocessing is an important step in the data mining process that prepares raw data for analysis. Real-world data is often incomplete, noisy, and inconsistent, which may affect the accuracy of mining results. Data preprocessing techniques such as data cleaning, data integration, data reduction, and data transformation help improve data quality. This unit explains various preprocessing methods and their role in improving the efficiency and reliability of data mining systems.

Objectives

- To understand the **importance of data preprocessing in data mining**.
- To study different **data preprocessing techniques such as cleaning, integration, reduction, and transformation**.
- To analyze methods for handling **missing values and noisy data**.
- To understand **data discretization and concept hierarchies**.
- To study **data mining primitives and system architecture**.

Learning Outcomes

After completing this unit, students will be able to:

- Define **data preprocessing and its importance in data mining**.
- Explain techniques for **data cleaning, integration, reduction, and transformation**.
- Analyze methods for **handling missing values and noisy data**.
- Describe **data mining primitives and system architecture**.
- Apply **discretization and concept hierarchy techniques** in data analysis.

Importance of Studying this Unit

- Improves the **quality and accuracy of data used for mining**.
- Helps remove **noise, inconsistencies, and missing values** from datasets.
- Enhances the **performance of data mining algorithms**.
- Reduces the **size and complexity of large datasets**.
- Enables effective **knowledge discovery from large datasets**.

Key Concepts

- Data Preprocessing
- Data Cleaning
- Missing Values
- Noisy Data
- Data Integration
- Data Reduction
- Data Transformation
- Data Compression
- Data Mining Primitives

- Discretization
- Concept Hierarchy

4.1 What is Data Preprocessing?

Data preprocessing is the process of preparing raw data for analysis by cleaning and transforming it into a usable format. In data mining it refers to preparing raw data for mining by performing tasks like cleaning, transforming, and organizing it into a format suitable for mining algorithms.

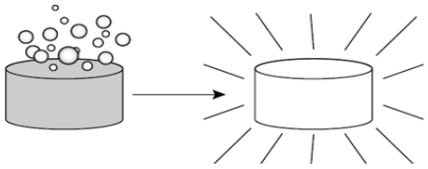
- Goal is to improve the quality of the data.
- Helps in handling missing values, removing duplicates, and normalizing data.
- Ensures the accuracy and consistency of the dataset.

Steps in Data Preprocessing

Some key steps in data preprocessing are:

- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation

4.1. Data Cleaning: It is the process of identifying and correcting errors or inconsistencies in the dataset. It involves handling missing values, removing duplicates, and correcting incorrect or outlier data to ensure the dataset is accurate and reliable. Clean data is essential for effective analysis, as it improves the quality of results and enhances the performance of data models.



Missing Values: This occur when data is absent from a dataset. You can either ignore the rows with missing data or fill the gaps manually, with the attribute mean, or by using the most probable value. This ensures the dataset remains accurate and complete for analysis.

Noisy Data: It refers to irrelevant or incorrect data that is difficult for machines to interpret, often caused by errors in data collection or entry. It can be handled in several ways:

a) Binning: Binning methods smooth a sorted data value by consulting its "neighbourhood," that is, the values around it. The sorted values are distributed into several "buckets," or bins. Because binning methods consult the neighbourhood of values, they perform local smoothing. **There are three kinds of binning.** They are:

- **Smoothing by Bin Means**: In this method, each value in a bin is replaced by the mean value of the bin. For example, the mean of the values 4, 8, and 15 in Bin 1 is 9. Therefore, each original value in this bin is replaced by the value 9.
- **Smoothing by Bin Medians:** In this method, each value in a bin is replaced by the median value of the bin. For example, the median of the values 4, 8, and 15 in Bin 1 is 8. Therefore, each original value in this bin is replaced by the value 8.
- **Smoothing by Bin Boundaries:** In this method, the minimum and maximum values in each bin are identified as the bin boundaries. Each bin value is then replaced by the closest boundary value.

  For example, the middle value of the values 4, 8, and 15 in Bin 1is replaced with nearest boundary i.e., 4.

Example:

Sorted data for price (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin medians:

Bin 1: 8, 8, 8
Bin 2: 21, 21, 21
Bin 3: 28, 28, 28

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

b) Regression: Data smoothing can also be done by regression, a technique that used to predict the numeric values in a given data set. It analyses the relationship between a target variable (dependent) and its predicate variable (independent).

- Regression is a form of a supervised machine learning technique that tries to predict any continuous valued attribute.
- Regression done in two ways;

  a. **Linear regression** involves finding the "best" line to fit two attributes (or variables) so that one attribute can be used to predict the other.

  b. **Multiple linear regression** is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

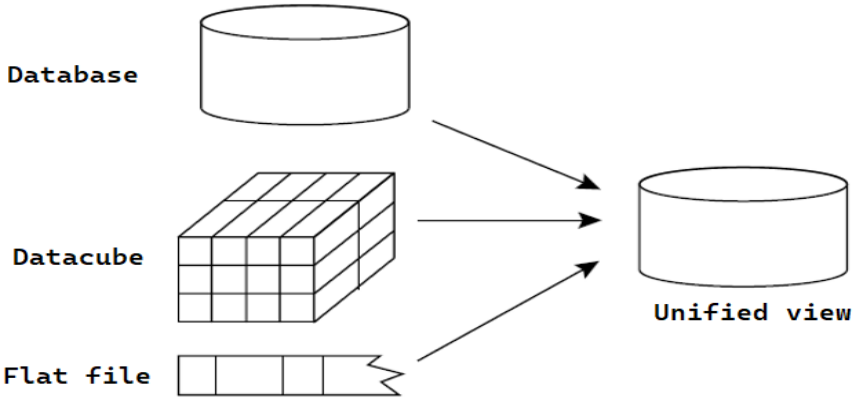
c) Clustering: It supports in identifying the outliers. The similar values are organized into clusters and those values which fall outside the cluster are known as outliers.

4.2. Data Integration

Data integration is the process of combining data from multiple sources into a single, unified view. This process involves identifying and accessing the different data sources, mapping the data to a common format. Different data sources may include multiple data cubes, databases, or flat files.

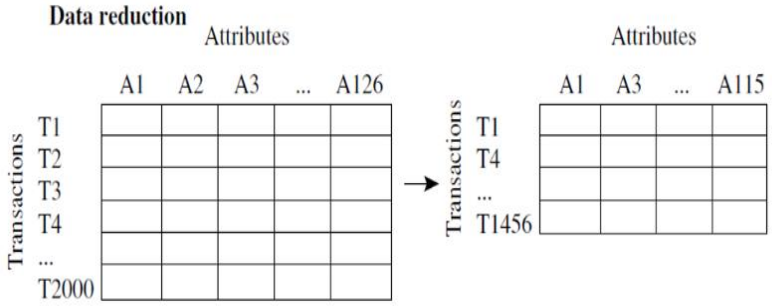
The goal of data integration is to make it easier to access and analyze data that is spread across multiple systems or platforms, in order to gain a more complete and accurate understanding of the data.

Data integration strategy is typically described using a triple (G, S, M) approach, where G denotes the global schema, S denotes the schema of the heterogeneous data sources, and M represents the mapping between the queries of the source and global schema.



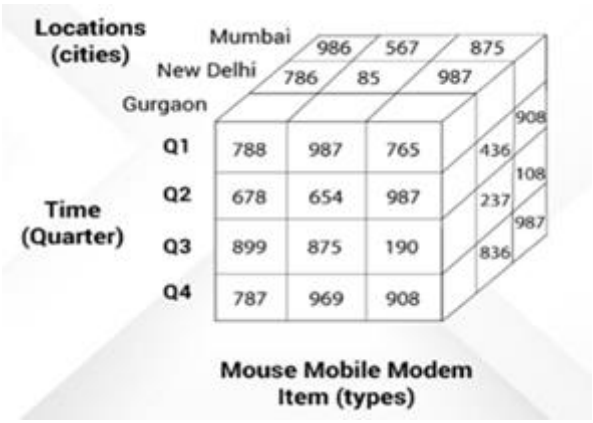
3. Data Reduction: It reduces the dataset's size while maintaining key information. This can be done through feature selection, which chooses the most relevant features, and feature extraction, which transforms the data into a lower-dimensional space while preserving important details.

It uses various reduction techniques such as,



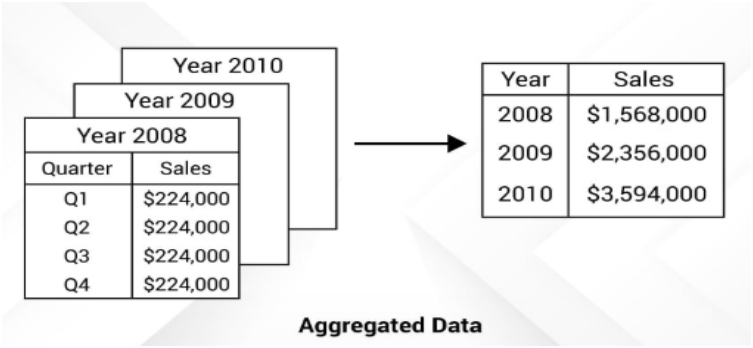
A) Data cube aggregation, where aggregation operations are applied to the data in the construction of a data cube.3-dimensional data in a tabular structure, representing this data in a cube format increases the readability of the data:

Each side of the cube represents one dimension- Time, Location, and Item Type (Mouse mobile



modem).

Another usability of the data cube aggregation in data mining is when we want to aggregate the data values. In the example below, we have quarterly sales data for different years from 2008 to 2010



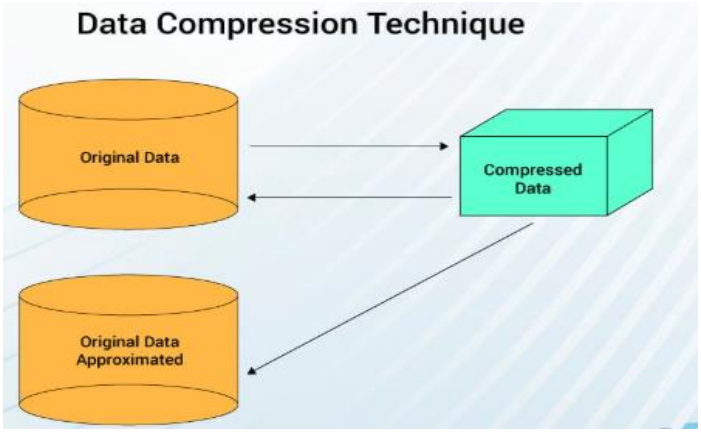
B) Data Compression

Data compression in data mining as the name suggests simply compresses the data. This technique encapsulates the data or information into a condensed form by eliminating duplicate, not needed information. It changes the structure of the data without taking much space and is represented in a binary form.

There are two types of data compression:

1. Lossless Compression: When the compressed data can be restored or reconstructed back to its original form without the loss of any information then it is referred to as lossless compression.

2. Lossy Compression: When the compressed data cannot be restored or reconstructed back into its original form then referred to as Lossy compression.



Data compression technique varies based on the type of data.

String data: In string compression, the data is modified in a limited manner without complete expansion; hence the string is mostly lossless as the data can be retrieved back to its original form. Therefore it is lossless data compression. There are extensive theories and well-tuned algorithms that are used for data compression.

Audio or video data: Unlike string data, audio or video data cannot be recreated to its original shape, hence is lossy data compression. At times, it may be possible to reconstruct small bits or pieces of the signal data but you cannot restore it to its whole form.

Time Sequential data: The time-sequential data is not audio data. It is by large, usually short data fragments and it varies slowly with time as is used for data compression.

4.3 Data Compression Methodologies

There are two methodologies:

1. Dimensionality
2. Numerosity reduction

1. Dimensionality Reduction

Dimensions are also known as features or attributes . These dimensions are nothing else but properties of the data, i.e., describing what the data is about.

For instance, we have the data of employees of a company. We have their name, age, gender, location, education, and income. All these variables do nothing but help us to *know, understand and describe the data point*.

As the features increase, the sparsity of the dataset also increases. The sparsity indicates that there is a relatively higher percentage of the variables that do not contain actual data. These "empty" cells or NA values take up unnecessary storage.

Dimensionality Reduction in data mining is the process of reducing the data by removing these features from the data. There are three techniques for this:

- **Wavelet Transformation**

Wavelet Transform in Data Mining is a form of lossy data compression.

Let's say we have a data vector Y, by applying the wavelet transform on this vector Y, we would receive a different numerical data vector Y', where the length of both the vectors Y and Y' are the same. Now, you may be wondering how transforming Y into Y' helps us to reduce the data. This Y' data can be trimmed or truncated whereas the actual vector Y cannot be compressed.

Let's say we have a data vector Y. When we apply wavelet transformation on this vector Y, we get a different numerical data vector Y' where the length of both the vectors Y and Y' are the same now.

The reason it is called 'wavelet transform' is that the information here is present in the form of waves, like how a frequency is depicted graphically as signals. The wavelet transform also has efficiency for data cubes, sparse or skewed data. It is mostly applicable for image compression and for signal processing.

- **Principal Component Analysis**

(e.g., Principal Component Analysis): A technique that reduces the number of variables in a dataset while retaining its essential information.

Principal component analysis – PCA in data mining, a technique for data reduction in data mining, groups the important variables into a component taking the maximum information present within the data and discards the other, not important variables.

Now, let's say out of total n variables, k are such variables that are identified and are part of this new component. This component is now what is representative of the data and used for further analysis.

In short, PCA in data mining is applied to reducing multi-dimensional data into lower-dimensional data. This is done by eliminating variables containing the same information as provided by other variables and combining the relevant variables into components. The principal component analysis is also useful for sparse, and skewed data.

- **Feature Selection or Attribute Subset Selection**

The attribute subset selection or feature selection method decreases the data volume by removing unnecessary variables. Hence, the name feature selection. This is done in such a way that the probability distribution of the reduced data is similar to that of the actual data, given the original variables.

2. Numerosity Reduction: Reducing the number of data points by methods like sampling to simplify the dataset without losing critical patterns.

Another methodology in data reduction in data mining is numerosity reduction in which the volume of the data is reduced by representing it in a lower format. There are two types of this technique: parametric and non-parametric numerosity reduction.

1. Parametric Reduction

The parametric numerosity reduction technique holds an assumption that the data fits into the model. Hence, it estimates the model parameters, and stores only these estimated parameters, and not the original or the actual data. The other data is discarded, leaving out the potential outliers.

The ways to perform parametric numerosity reduction are: Regression and Log-Linear. Both the parametric methods of regression and log-linear methods are applicable for sparse and skewed data.

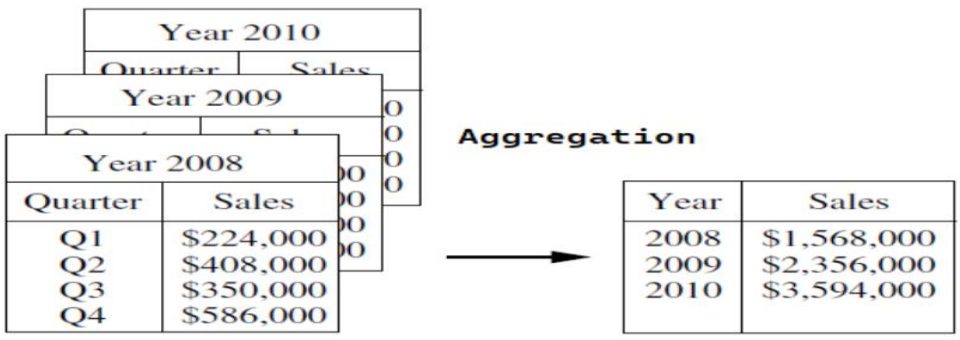
Regression: Linear Regression analysis is used for studying or summarizing the relationship between variables that are linearly related. The regression is also of two kinds:

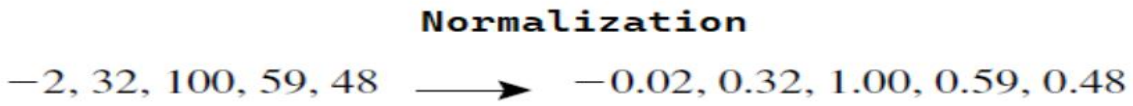
Simple Linear regression and Multiple Linear regression.

Type	What it Means
Simple Linear Regression	When we want to explore the relationship between only two variables: the independent variable (x) and the dependent variable (y). The best fit line is represented as $Y = b_0 + b_1x$, where b_0 is the intercept and b_1 is the regression coefficient.
Multiple Linear Regression	When we want to evaluate the impact of more than one independent variable on the dependent variable. The regression equation becomes $Y = b_0 + b_1x_1 + b_2x_2$, where b_0 is the intercept and b_1, b_2 are regression coefficients.

4. Data Transformation:

Data transformation in data mining refers to the process of converting raw data into a format that is suitable for **analysis and modelling**. The goal of data transformation is to prepare the data for data mining so that it can be used to extract useful insights and knowledge.





Data transformation typically involves several steps, including:

1. **Smoothing:** It is a process that is used to remove noise from the dataset using techniques include binning, regression, and clustering.

2. **Attribute construction (or feature construction):** In this, new attributes are constructed and added from the given set of attributes to help the mining process.

3. **Aggregation:** In this, summary or aggregation operations are applied to the data. **For example**, the daily sales data may be aggregated to compute **monthly and annual total amounts.**

4. **Data normalization**: This process involves converting all data variables into a small range. such as -1.0 to 1.0, or 0.0 to 1.0.

5. **Generalization:** It converts low-level data attributes to high-level data attributes using concept hierarchy. For Example, Age initially in Numerical form (22,) is converted into categorical value (young, old).

Method Name	**Irregularity**	**Output**
Data Cleaning	Missing, Nosie, and Inconsistent data	Quality Data before Integration
Data Integration	Different data sources (data cubes, databases, or flat files)	Unified view
Data Reduction	Huge amounts of data can take a long time, making such analysis impractical or infeasible.	Reduce the size of a dataset and maintains the integrity.
Data Transformation	Raw data	Prepare the data for data mining

4.4 Data Mining Primitives

A data mining query is defined in terms of the following primitives

Task-relevant data: "This is the database portion to be investigated. For example, suppose that" you are a manager of All Electronics in charge of sales in the United States and Canada. In particular, you would like to study the buying trends of customers in Canada. Rather than mining on the entire database. These are referred to as relevant attributes

The kinds of knowledge to be mined: "This specifies the data mining functions to be performed," such as characterization, discrimination, association, classification, clustering, or evolution analysis. For instance, if studying the buying habits of customers in Canada, you may choose to mine associations between customer profiles and the items that these customers like to buy.

Background knowledge: "Users can specify background knowledge, or knowledge about the" domain to be mined. This knowledge is useful for guiding the knowledge discovery process, and for evaluating the patterns found. There are several kinds of background knowledge.

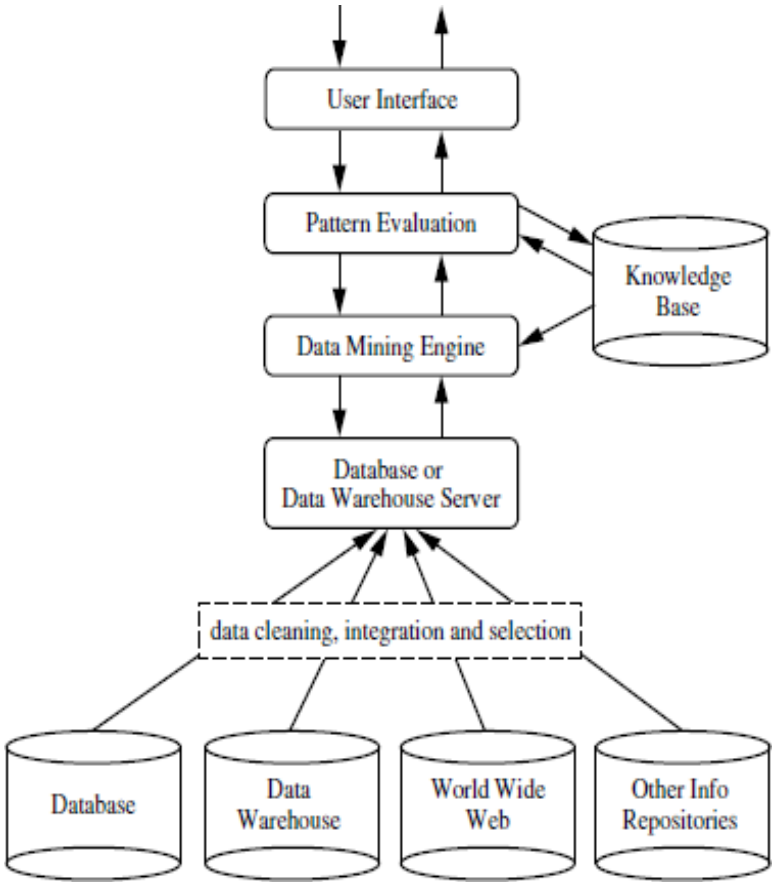
Interestingness measures: "These functions are used to separate uninteresting patterns from" knowledge. They may be used to guide the mining process, or after discovery, to evaluate the discovered patterns. Different kinds of knowledge may have different interestingness measures.

Presentation and visualization of discovered patterns: "This refers to the form in which" discovered patterns are to be displayed. Users can choose from different forms for knowledge presentation, such as rules, tables, charts, graphs, decision trees, and cubes.

Task-Relevant Data

4.5 Architecture of DataMining

A typical datamining system may have the following major components.



Knowledge Base:

This is the domain knowledge that is used to guide the search or evaluate the interestingness of resulting patterns. Such knowledge can include concept hierarchies, used to organize attributes or attribute values into different levels of abstraction. Knowledge such as user beliefs, which can be used to assess a pattern's interestingness based on its unexpectedness, may also be included. Other examples of domain knowledge are additional interestingness constraints or thresholds, and metadata (e.g., describing data from multiple heterogeneous sources).

1. ***DataMining Engine:***
   This is essential to the data mining systemand ideally consists ofa set of functional modules for tasks such as characterization, association and correlationanalysis, classification, prediction, cluster analysis, outlier analysis, and evolutionanalysis.
2. ***Pattern Evaluation Module:***
   This component typically employs interestingness measures interacts with the data mining modules so as to focus the search toward interesting patterns. It may use interestingness thresholds to filter out discovered patterns. Alternatively, the pattern evaluation module may be integrated with the mining module, depending on the implementation of the datamining method used. For efficient data mining, it is highly recommended to push the evaluation of pattern

interestingness as deep as possible into the mining process so as to confine the search to only the interesting patterns.

3. ***Userinterface:***
   This module communicates between users and the data mining system, allowing the user to interact with the system by specifying a data mining query or task, providing information to help focus the search, and performing exploratory datamining based on the intermediate data mining results. In addition, this component allows the user to browse database and data warehouse schemas or data structures, evaluate mined patterns, and visualize the patterns in different forms.

4.6 Data Discretization

- Dividing the range of a continuous attribute into intervals.
- Interval labels can then be used to replace actual data values.
- Reduce the number of values for a given continuous attribute.
- Some classification algorithms only accept categorically attributes.
- This leads to a concise, easy-to-use, knowledge-level representation of mining results.
- Discretization techniques can be categorized based on whether it uses class information or not such as follows:
    - ***Supervised Discretization -*** This discretization process uses **class** information.
    - ***Unsupervised Discretization -*** This discretization process does **not use class** information.
- Discretization techniques can be categorized based on which direction it proceeds as follows:

Top-down Discretization -

- If the process starts by first finding one or a few points called split points or cut points to split the entire attribute range and then repeat this recursively on the resulting intervals.

Bottom-up Discretization -

- Starts by considering all of the continuous values as potential split-points.
- Removes some by merging neighborhood values to form intervals, and then recursively applies this process to the resulting intervals.

4.6 Concept Hierarchies

- Discretization can be performed rapidly on an attribute to provide a hierarchical partitioning of the attribute values, known as a ***Concept Hierarchy.***
- Concept hierarchies can be used to reduce the data by collecting and replacing low-level concepts with higher-level concepts.

- In the multidimensional model, data are organized into multiple dimensions, and each dimension contains multiple levels of abstraction defined by concept hierarchies.
- This organization provides users with the flexibility to view data from different perspectives.
- Data mining on a reduced data set means fewer input and output operations and is more efficient than mining on a larger data set.
- Because of these benefits, discretization techniques and concept hierarchies are typically applied before data mining, rather than during mining.

Typical Methods of Discretization and Concept Hierarchy Generation for Numerical Data

1] Binning

- Binning is a top-down splitting technique based on a specified number of bins.
- Binning is an unsupervised discretization technique because it does not use class information.
- In this, The sorted values are distributed into several buckets or bins and then replaced with each bin value by the bin mean or median.
- It is further classified into
    - ***Equal-width (distance) partitioning***
    - ***Equal-depth (frequency) partitioning***

2] Histogram Analysis

- It is an unsupervised discretization technique because histogram analysis does not use class information.
- Histograms partition the values for an attribute into disjoint ranges called buckets.
- It is also further classified into
    - ***Equal-width histogram***
    - ***Equal frequency histogram***
- The histogram analysis algorithm can be applied recursively to each partition to automatically generate a multilevel concept hierarchy, with the procedure terminating once a pre-specified number of concept levels has been reached.

3] Cluster Analysis

- Cluster analysis is a popular data discretization method.
- A clustering algorithm can be applied to discretize a numerical attribute of A by partitioning the values of A into clusters or groups.

- Clustering considers the distribution of A, as well as the closeness of data points, and therefore can produce high-quality discretization results.
- Each initial cluster or partition may be further decomposed into several subcultures, forming a lower level of the hierarchy.

4] Entropy-Based Discretization

- Entropy-based discretization is a supervised, top-down splitting technique.
- It explores class distribution information in its calculation and determination of split points.
- Let D consist of data instances defined by a set of attributes and a class-label attribute.
- The class-label attribute provides the class information per instance.
- In this, the interval boundaries or split-points defined may help to improve classification accuracy.
- The entropy and information gain measures are used for decision tree induction.

5] Interval Merge by χ2 Analysis

- It is a bottom-up method.
- Find the best neighboring intervals and merge them to form larger intervals recursively.
- The method is supervised in that it uses class information.
- Chi Merge treats intervals as discrete categories.
- The basic notion is that for accurate discretization, the relative class frequencies should be fairly consistent within an interval.
- Therefore, if two adjacent intervals have a very similar distribution of classes, then the intervals can be merged.
- Otherwise, they should remain separate.

Unit Highlights

- Introduction to **Data Preprocessing**.
- Study of **Data Cleaning** (missing values, noisy data).
- Understanding **Data Integration** from multiple sources.
- Learning **Data Reduction techniques**.
- Study of **Data Transformation methods**.
- Understanding **Discretization and Concept Hierarchy**.

Case Study

A retail company collects sales data from different branches. The data may contain missing values and duplicate records. Using **data preprocessing techniques**, the company cleans and integrates the data. After preprocessing, the company analyzes the data to understand customer buying patterns and improve sales strategies.

Book References

1. **Jiawei Han and Micheline Kamber** – *Data Mining: Concepts and Techniques*.
2. **Arun K. Pujari** – *Data Mining Techniques*.
3. **Margaret H. Dunham** – *Data Mining: Introductory and Advanced Topics*.

UNIT- V : DATA GENERALIZATION AND SUMMARIZATION		
Part – A		
1	What is concept description?	L1
2	Define data generalization.	L1
3	What is summarization-based characterization?	L2
4	Define attribute relevance.	L1
5	What is class comparison in data mining?	L2
6	Define association rule mining.	L1
7	What is market basket analysis?	L2
8	What is the Apriori algorithm used for?	L1
9	Mention one improvement in the improved Apriori algorithm.	L2
Part – B		
1	What is concept description? Explain data generalization and summarization-based characterization in detail.	L2,L4
2	Explain the relevance of attributes and the method of class comparisons in data summarization.	L2,L5
3	Define association rule mining. Explain market basket analysis with an example.	L2,L3
4	Describe the working of the Apriori algorithm with a suitable example and candidate generation steps.	L3,L4
5	Explain the rule generation process from frequent itemsets using the Apriori approach.	L2,L3
6	Compare Apriori and Improved Apriori algorithms. Explain how efficiency is improved.	L3,L4
7	What is Incremental ARM? How does it differ from traditional ARM approaches?	L2,L4
8	Explain the concept of associative classification. How is it different from traditional classification?	L2,L4

UNIT- 5

DATA GENERALIZATION AND SUMMARIZATION

"The goal of data mining is to discover useful patterns and knowledge from large datasets."

-W. Edwards Deming

1. Overview

Data generalization and summarization are techniques used in data mining to convert large volumes of detailed data into meaningful and simplified information. These techniques help decision makers understand patterns and trends easily. Association rule mining discovers relationships between items in large datasets. Algorithms like Apriori are used to find frequent itemsets and generate useful rules for analysis.

2. Objectives

After studying this unit, students will be able to:

- Understand the concept of data generalization and summarization.
- Learn the concept description and class comparison techniques.
- Understand association rule mining and market basket analysis.
- Learn Apriori algorithm and improved methods for mining frequent itemsets.
- Analyze real-world applications of association rules.

3. Learning Outcomes

At the end of this unit, students will be able to:

- Explain concept description in data mining.
- Identify methods for data generalization and summarization.
- Apply Apriori algorithm to discover frequent patterns.
- Interpret association rules using support, confidence, and lift.
- Analyze real-world problems using association rule mining techniques.

4. Importance of Studying this Unit

This unit helps students understand how large data sets can be summarized and analyzed to extract useful knowledge. It plays an important role in business decision-making, marketing strategies, and recommendation systems. Association rule mining is widely used in retail, healthcare, and e-commerce industries. Understanding these techniques helps students apply data mining concepts in real-world

5. Key Concepts

Important concepts covered in this unit include:

- Concept Description
- Data Generalization
- Data Summarization
- Market Basket Analysis
- Association Rule Mining
- Frequent Itemsets
- Apriori Algorithm

• • Incremental ARM
• Associative Classification

5.1 WHAT IS CONCEPT DESCRIPTION?

Descriptive vs. predictive data mining

❖ **Descriptive mining:**

describes concepts or task-relevant data sets in concise, summarative, informative, discriminative forms

❖ **Predictive mining:** Based on data and analysis, constructs models for the database, and predicts the trend and properties of unknown data

❖ Concept description:

- Characterization: provides a concise and succinct **summarization** of the given collection of data
- Comparison: provides descriptions **comparing** two or more collections of data

WHAT IS CONCEPT DESCRIPTION?

- A concept usually refers to a collection of data such as frequent_buyers, graduate_students etc.
- As a data mining task, concept description is not a simple enumeration (number of things done one by one) of the data.
- Concept description generates descriptions for characterization and Comparison of the data it is also called class description.

Characterization provides a concise and brief summarization

While concept or class comparison (also known as discrimination) provides (inequity) comparing two or more collections of data.

Example

- Discriminations Given the ABC Company database, for example, examining individual customer transactions.
- Sales managers may prefer to view the data generalized to higher levels, such as summarized by customer groups according to geographic regions, frequency of purchases per group and customer income.

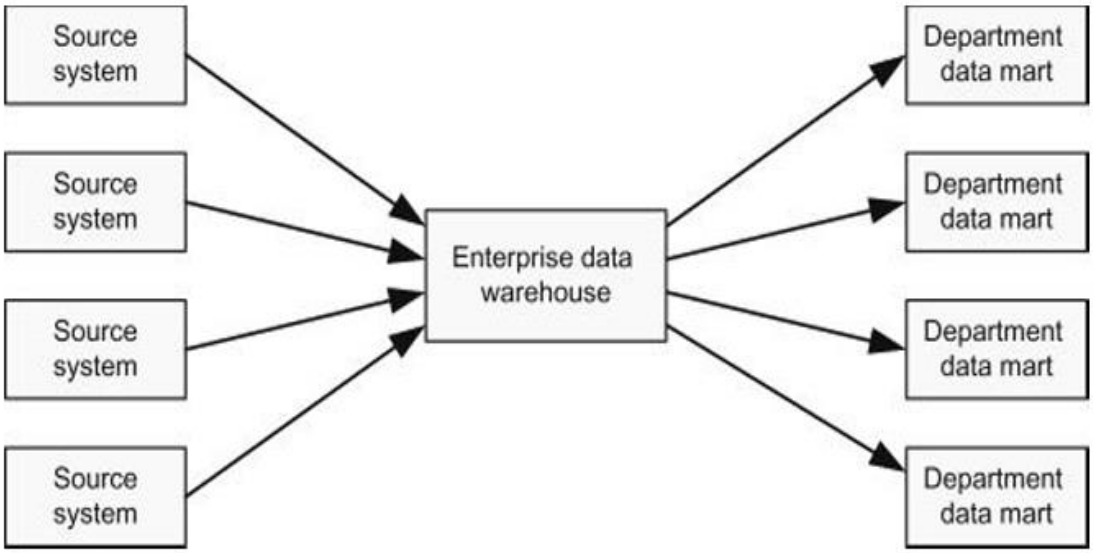
5.2 DATA GENERALIZATION AND SUMMARIZATION

❖ Data generalization

- A process which abstracts a large set of task-relevant data in a database from a low conceptual levels to higher ones.

❖ Approaches:

- Data cube approach(OLAP approach)
- Attribute-oriented induction approach



DATA GENERALIZATION

PRESENTATION OF GENERALIZED RESULTS

- **Generalized relation:**
  - Relations where some or all attributes are generalized, with counts or other aggregation values accumulated.

- **Cross *tabulation:***

  - *Mapping* results into cross tabulation form (similar to contingency tables).

- **Visualization techniques:**

  Pie charts, bar charts, curves, cubes, and other visual forms.

- **Quantitative characteristic rules:**

  - Mapping generalized result into characteristic rules with quantitative information associated with it

Presentation—Generalized Relation

location	item	sales (in million dollars)	count (in thousands)
Asia	TV	15	300
Europe	TV	12	250
North_America	TV	28	450
Asia	computer	120	1000
Europe	computer	150	1200
North_America	computer	200	1800

A generalized relation for the sales in 1997.

Class Characterization: An Example

Initial Relation

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
...	...	...	...	...	...	...	...
Removed	Retained	Sci,Eng, Bus	Country	Age range	City	Removed	Excl, VG,..

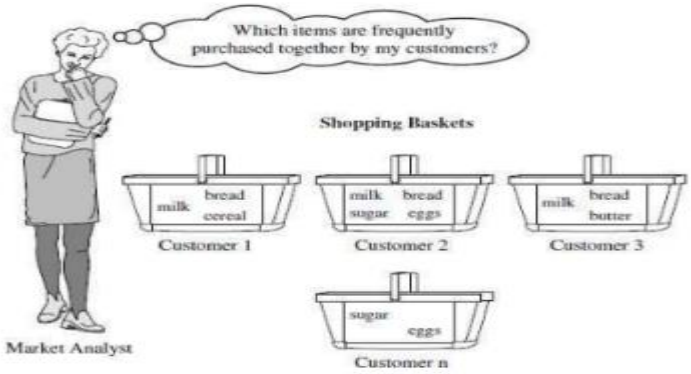
Prime Generalized Relation

Gender	Major	Birth_region	Age_range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very-good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
...	...	...	...	...	...	...

Gender \ Birth_Region	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

5.3 MARKET BASKET ANALYSIS

Market Basket Analysis is a modeling technique based upon the theory that if you **buy a certain group of items**, you are more (or less) likely **to buy another group of items**. For example, if you are in an English pub and you buy a pint of beer and don't buy a bar meal, you are more likely to buy crisps (US. chips) at the same time than somebody who didn't buy beer.



❖ The set of items a customer buys is referred to as an itemset, and market basket analysis seeks to find relationships between purchases.

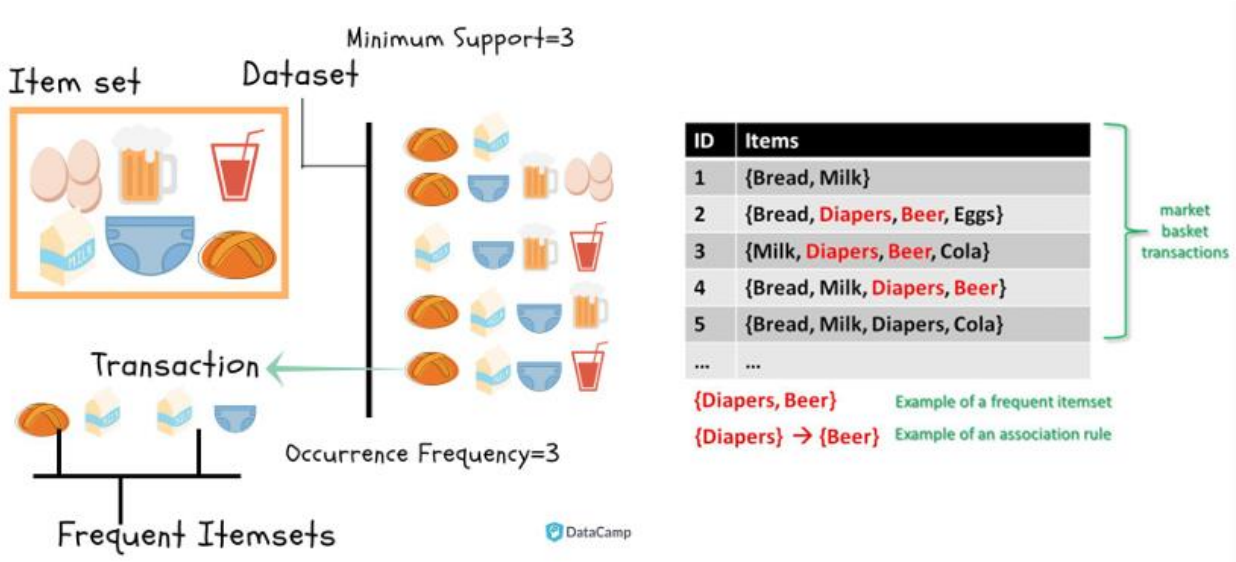
❖ Typically the relationship will be in the form of a rule:

- IF {beer, no bar meal} THEN {crisps}.

The probability that a customer will buy beer without a bar meal (i.e. that the antecedent is true) is referred to as the **support** for the rule. The conditional probability that a customer will purchase crisps is referred to as the **confidence**.

❖ The algorithms for performing market basket analysis are fairly straightforward (Berry and Linhoff is a reasonable introductory resource for this). The complexities mainly arise in exploiting taxonomies, avoiding combinatorial explosions (a supermarket may stock 10,000 or more line items), and dealing with the large amounts of transaction data that may be available.

❖ A major difficulty is that a large number of the rules found may be trivial for anyone familiar with the business. Although the volume of data has been reduced, we are still asking the user to find a needle in a haystack. Requiring rules to have a high minimum support level and a high confidence level risks missing any exploitable result we might have found.



MARKET BASKET ANALYSIS

How Association Rules are Evaluated

The strength and reliability of an association rule are measured using three key metrics.

Support and Confidence for Itemset A and B are represented by formulas:

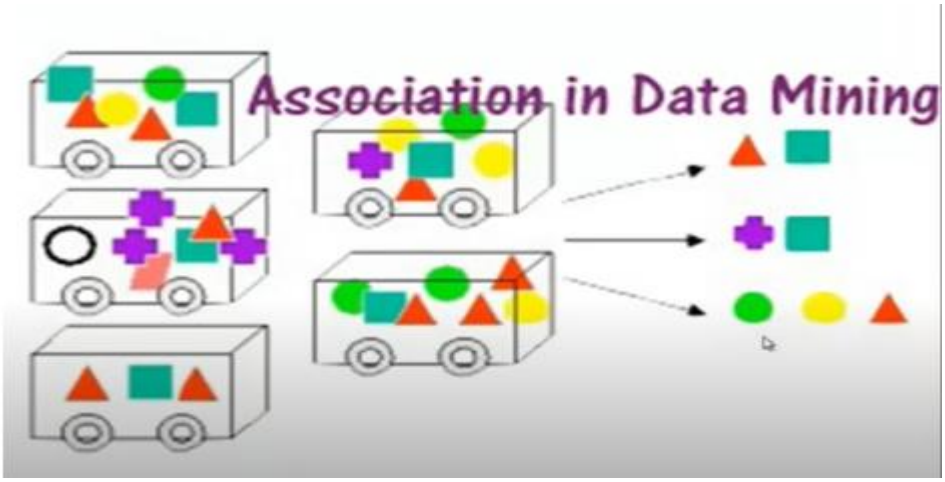
$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence (A}\rightarrow\text{B)} = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

Lift (A,B) = C(A→B)/S(B)

5.4 ASSOCIATION RULE MINING

Association rule learning is a rule-based machine learning method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using some measures of interestingness



❖ Learning of Association rules is used to find relationships between attributes in large databases. An association rule, A=> B, will be of the form" for a set of transactions, some value of itemset A determines the values of itemset B under the condition in which minimum support and confidence are met".

❖ Support and Confidence can be represented by the following example:

```
Bread=> butter [support=2%, confidence-60%]
```

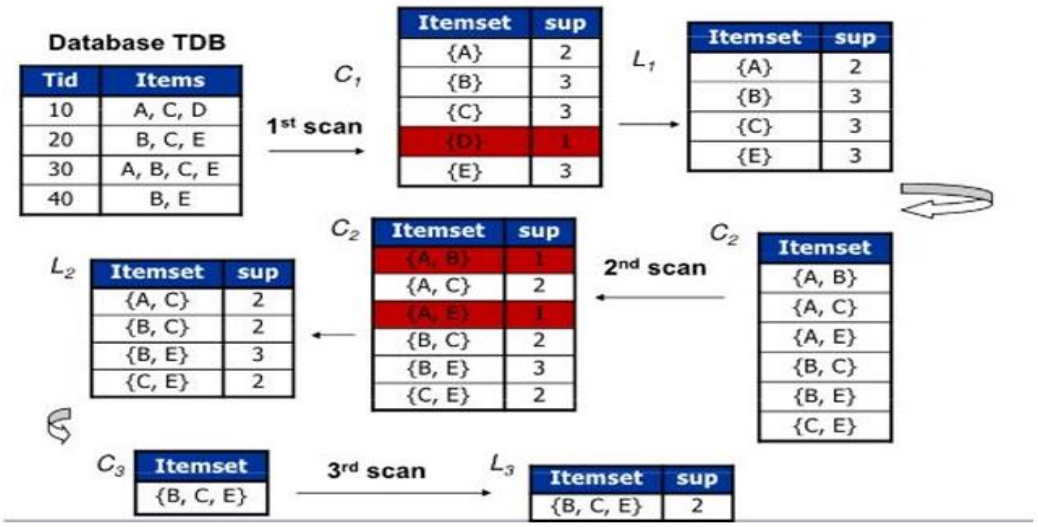
❖ The above statement is an example of an association rule. This means that there is a 2% transaction that bought bread and butter together and there are 60% of customers who bought bread as well as butter.

ASSOCIATION RULE MINING

Support and Confidence for Itemset A and B are represented by formulas:

$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence (A}\rightarrow\text{B)} = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$



$$Lift(A \Rightarrow B) = \frac{Confidence\,(A \Rightarrow B)}{Expected\ Confidence\,(A \Rightarrow B)} = \frac{Confidence\,(A \Rightarrow B)}{P(B)} = \frac{P(A \cap B)}{P(A) \cdot P(B)} = \frac{Confidence\,(A \Rightarrow B)}{Support\,(B)}$$

Association rule mining consists of 2 steps:
1. Find all the frequent itemsets.
2. Generate association rules from the above frequent itemsets.

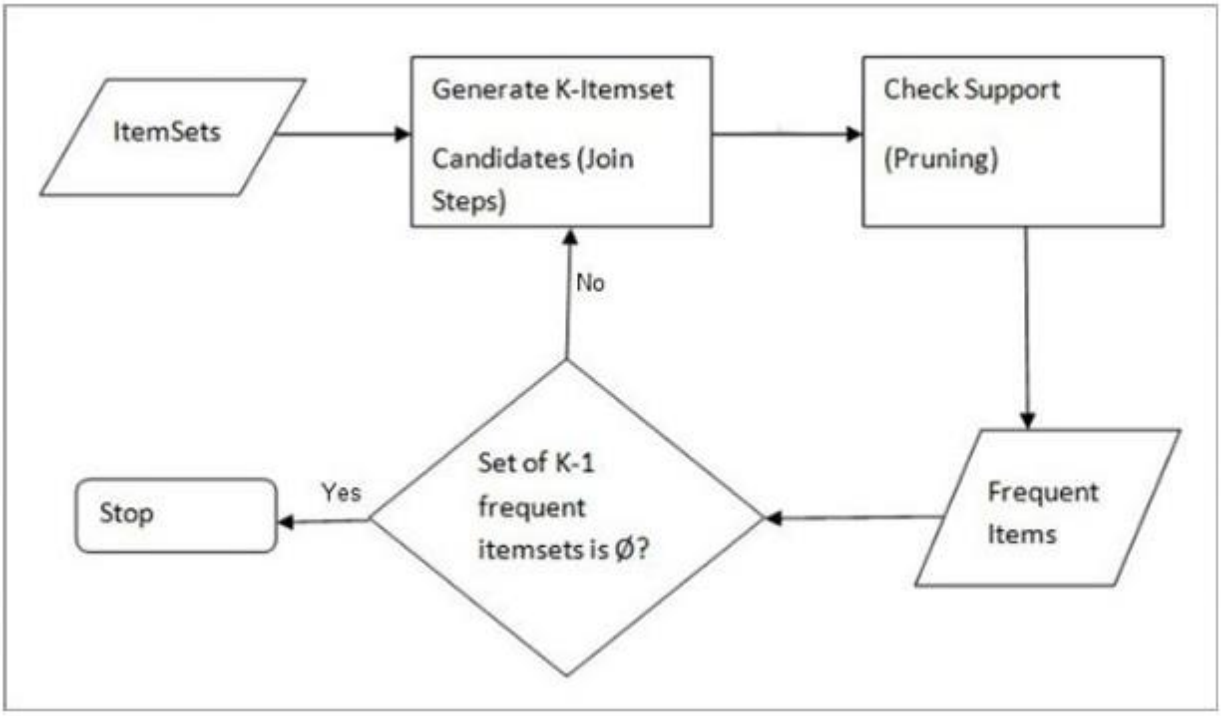
EXAMPLE OF ASSOCIATION RULE

FINDING FREQUENT ITEM SETS:
APRIORI ALGORITHM:

TID	Items
1	Bread, Peanuts, Milk, Fruit, Jam
2	Bread, Jam, Soda, Chips, Milk, Fruit
3	Steak, Jam, Soda, Chips, Bread
4	Jam, Soda, Peanuts, Milk, Fruit
5	Jam, Soda, Chips, Milk, Bread
6	Fruit, Soda, Chips, Milk
7	Fruit, Soda, Peanuts, Milk
8	Fruit, Peanuts, Cheese, Yogurt

Examples
{bread} ⇒ {milk}
{soda} ⇒ {chips}
{bread} ⇒ {jam}

- Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

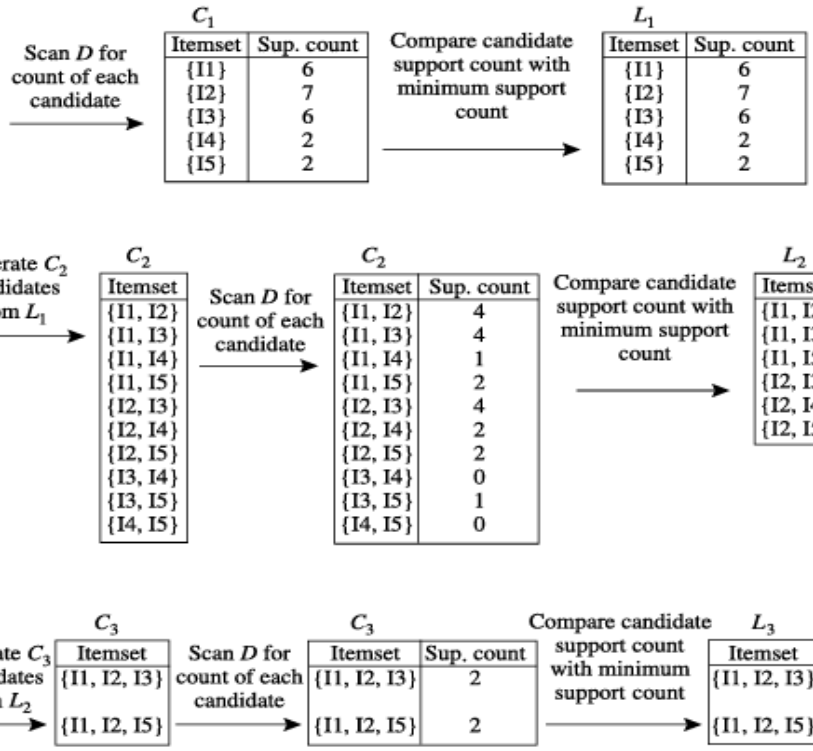


Example of Apriori: Support threshold=50%, Confidence= 60%

EXAMPLE:1

Transactional Data for an *AllElectronics* Branch

TID	*List of item_IDs*
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3



EXAMPLE:2

Transaction	List of items
T1	I1,I2,I3
T2	I2,I3,I4
T3	I4,I5
T4	I1,I2,I4
T5	I1,I2,I3,I5
T6	I1,I2,I3,I4

Solution:

Support threshold=50% => 0.5*6= 3 => min_sup=3

1. Count Of Each Item

TABLE-2

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

2. Prune Step: TABLE -2 shows that I5 item does not meet min_sup=3, thus it is deleted, only I1, I2, I3, I4 meet min_sup count.

TABLE-3

Item	Count
I1	4
I2	5
I3	4
I4	4

3. Join Step: Form 2-itemset. From **TABLE-1** find out the occurrences of 2-itemset.

TABLE-4

Item	Count
I1,I2	4
I1,I3	3
I1,I4	2
I2,I3	4
I2,I4	3
I3,I4	2

4. Prune Step: TABLE -4 shows that item set {I1, I4} and {I3, I4} does not meet min_sup, thus it is deleted.

TABLE-5

Item	Count
I1,I2	4
I1,I3	3
I2,I3	4
I2,I4	3

5. Join and Prune Step: Form 3-itemset. From the **TABLE- 1** find out occurrences of 3-itemset. From **TABLE-5**, find out the 2-itemset subsets which support min_sup.

We can see for itemset {I1, I2, I3} subsets, {I1, I2}, {I1, I3}, {I2, I3} are occurring in **TABLE-5** thus {I1, I2, I3} is frequent.

We can see for itemset {I1, I2, I4} subsets, {I1, I2}, {I1, I4}, {I2, I4}, {I1, I4} is not frequent, as it is not occurring in **TABLE-5** thus {I1, I2, I4} is not frequent, hence it is deleted.

TABLE-6

Item
I1,I2,I3
I1,I2,I4
I1,I3,I4
I2,I3,I4

Only {I1, I2, I3} is frequent.

6. Generate Association Rules: From the frequent itemset discovered above the association could be:

{I1, I2} => {I3}

Confidence = support {I1, I2, I3} / support {I1, I2} = (3/ 4)* 100 = 75%

{I1, I3} => {I2}

Confidence = support {I1, I2, I3} / support {I1, I3} = (3/ 3)* 100 = 100%

{I2, I3} => {I1}

Confidence = support {I1, I2, I3} / support {I2, I3} = (3/ 4)* 100 = 75%

{I1} => {I2, I3}

Confidence = support {I1, I2, I3} / support {I1} = (3/ 4)* 100 = 75%

{I2} => {I1, I3}

Confidence = support {I1, I2, I3} / support {I2 = (3/ 5)* 100 = 60%

{I3} => {I1, I2}

Confidence = support {I1, I2, I3} / support {I3} = (3/ 4)* 100 = 75%

This shows that all the above association rules are strong if minimum confidence threshold is 60%.

- Join Step: C_k is generated by joining L_{k-1} with itself
- Prune Step: Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset
- Pseudo-code :C_k: Candidate itemset of size k
  L_k: frequent itemset of size k

```
L1 = {frequent items};
for (k = 1; Lk !=∅; k++) do begin
   Ck+1 = candidates generated from Lk;
   for each transaction t in database do
         increment the count of all candidates in Ck+1
         that are contained in t
   Lk+1 = candidates in Ck+1 with min_support
   end
return ∪k Lk;
```

5.5 IMPROVED APRIORI ALGORITHM

Many methods are available for improving the efficiency of the algorithm.

1. **Hash-Based Technique**: This method uses a hash-based structure called a hash table for generating the k-itemsets and its corresponding count. It uses a hash function for generating the table.
2. **Transaction Reduction**: This method reduces the number of transactions scanning in iterations. The transactions which do not contain frequent items are marked or removed.
3. **Partitioning**: This method requires only two database scans to mine the frequent itemsets. It says that for any itemset to be potentially frequent in the database, it should be frequent in at least one of the partitions of the database.
4. **Sampling:** This method picks a random sample S from Database D and then searches for frequent itemset in S. It may be possible to lose a global frequent itemset. This can be reduced by lowering the min_sup.
5. **Dynamic Itemset Counting:** This technique can add new candidate itemsets at any marked start point of the database during the scanning of the database.

APPLICATION OF ALGORITHM

Some fields where Apriori is used:

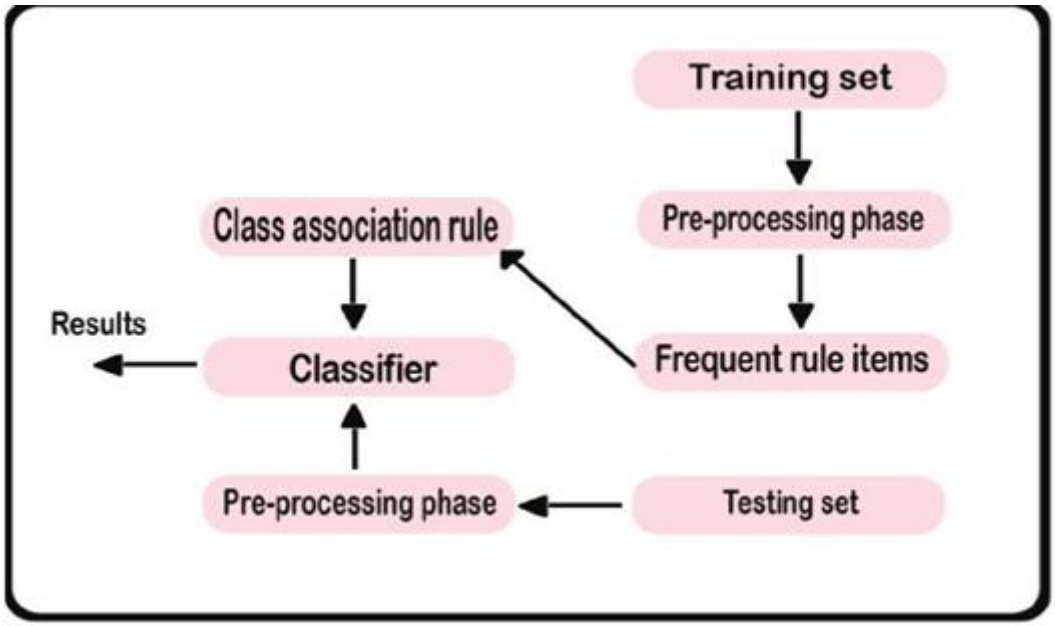
1. **In Education Field:** Extracting association rules in data mining of admitted students through characteristics and specialties.
2. **In the Medical field:** For example Analysis of the patient's database.
3. **In Forestry:** Analysis of probability and intensity of forest fire with the forest fire data.
4. Apriori is used by many companies like Amazon in the by Google for the auto-complete feature.

5.6 INCREMENTAL ARM

- It is noted that analysis of past transaction data can provide very valuable information on customer buying behavior, and thus **improve the quality of business decisions.**
- With the increasing use of the record-based databases whose data is being continuously added, updated, deleted etc.
- Examples of such applications include Web log records, stock market data, grocery sales data, transactions in e-commerce, and daily weather/traffic records etc.
- In many applications, we would like to mine the transaction database for a fixed amount of most recent data (say, data in the last 12 months).
- Mining is not a one-time operation, a naive approach to solve the incremental mining problem is to re-run the mining algorithm on the updated database.

5.7 ASSOCIATIVE CLASSIFICATION

- Associative classification (AC) is a branch of a wide area of scientific study known as data mining. Associative classification makes use of association rule mining for extracting efficient rules, which can precisely generalize the training data set, in the rule discovery process.
- An associative classifier (AC) is a kind of supervised learning model that uses association rules to assign a target value. The term associative classification was coined by Bing Liu et al., in which the authors defined a model made of rules "whose right-hand side are restricted to the classification class attribute".



Association Mining

• **Association rule mining**: – Finding frequent patterns, associations, correlations, or causal structures among sets of items or objects in transaction databases, relational databases, and

other information repositories.

• **Applications:** – Basket data analysis, cross-marketing, catalog design, loss-leader analysis, clustering, classification, etc.

• Examples.

- Rule form: "Body ® Head [support, confidence]".
- buys(x, "diapers") ® buys(x, "beers") [0.5%, 60%]
- major(x, "CS") ^ takes(x, "DB") ® grade(x, "A") [1%, 75%]

Association Rule: Basic Concepts

Given: (1) database of transactions, (2) each transaction is a list of items (purchased by a customer in a visit)

● **Find:** all rules that correlate the presence of one set of items with that of another set of items

– E.g., 98% of people who purchase tires and auto accessories also get automotive services done

• **Applications**

– *==> Maintenance Agreement (What the store should do to boost Maintenance Agreement sales)

– Home Electronics *==> (What other products should the store stocks up?)

– Attached mailing in direct marketing – Detecting "ping-pong"ing of patients, faulty "collisions"

Rule Measures: Support and Confidence

• **Find all the rules** X & Y => Z with minimum confidence and support

- support, s, probability that a transaction contains {X 4 Y 4 Z}
- confidence, c, conditional probability that a transaction having {X 4 Y} also contains Z

Let minimum support 50%, and minimum confidence 50%, we have

- A => C (50%, 66.6%)
- C=> A (50%, 100%)

Transaction ID	Items Bought
2000	A,B,C
1000	A,C
4000	A,D
5000	B,E,F

Association Rule Mining: A Road Map B,E,F

• Boolean vs. quantitative associations (Based on the types of values handled)

- buys(x, "SQLServer") ^ buys(x, "DMBook") ® buys(x, "DBMiner") [0.2%, 60%]
- age(x, "30..39") ^ income(x, "42..48K") ® buys(x, "PC") [1%, 75%]

• Single dimension vs. multiple dimensional associations (see ex. Above)

• Single level vs. multiple-level analysis

- What brands of beers are associated with what brands of diapers?

• Various extensions

- Correlation, causality analysis
  - Association does not necessarily imply correlation or causality
- Maxpatterns and closed itemsets
- Constraints enforced

• E.g., small sales (sum < 100) trigger big buys (sum > 1,000)?

Apriori Algorithm and Its Role in Associative Classification

In associative classification, the Apriori algorithm is a key method that is essential for identifying popular item sets. The method finds itemsets that meet a minimal support criterion via an iterative technique, creating strong correlations between qualities. Its main function in

associative categorization is to produce a set of frequent item sets from which association rules may be derived.
Utilizing the "apriori property," which stipulates that any non? frequent itemset must have non? frequent subsets, the method effectively prunes the search space.

Association Rule Mining is a powerful technique used to uncover meaningful relationships between variables within large datasets. They are designed to discover "if-then" patterns, providing insights into how data items are related and frequently occur together. These rules are particularly useful in identifying correlations and dependencies, enabling data-driven decision-making.

For instance, in a retail dataset, an association rule might identify that "if a customer buys bread, they are likely to buy butter". Such insights help businesses improve cross-selling strategies, inventory management, and customer satisfaction.

Key Components of Association Rules
Antecedent: The "if" part of the rule, representing the condition.
Example: A customer buys bread.
Consequent: The "then" part of the rule, representing the outcome.
Example: The customer also buys butter.
Association rules are derived through algorithms that evaluate the frequency and strength of these relationships. They use metrics like support, confidence, and lift to measure the relevance and reliability of discovered patterns. These rules have applications in various fields, such as retail, healthcare, and marketing, where analyzing customer behavior or trends is critical for success.

Rule Evaluation Metrics
Association rules are evaluated using key metrics that determine their relevance, strength, and reliability. These metrics include support, confidence, and lift, which quantify the frequency and strength of relationships between data items.

1. Support
Support measures how frequently an itemset (both antecedent and consequent) appears in the dataset. It provides an indication of how common a particular association is.

$$\text{Support} = \frac{\text{Transactions containing both antecedent and consequent}}{\text{Total transactions}}$$

Formula:
Example: If bread and butter appear together in 100 out of 1,000 transactions, the support is:

$$\text{Support} = \frac{100}{1000} = 0.10\,(10\%)$$

A higher support value indicates a more frequently occurring pattern in the dataset.
2. Confidence
Confidence measures the likelihood of the consequent occurring given that the antecedent has

already occurred. It evaluates the reliability of the rule.

Formula:

$$\text{Confidence} = \frac{\text{Support of antecedent and consequent}}{\text{Support of antecedent}}$$

Example: If 70% of customers who buy bread also buy butter, the confidence is:

$$\text{Confidence} = 70\% = 0.70$$

Higher confidence suggests a stronger relationship between the antecedent and consequent.

3. Lift

Lift measures the strength of an association compared to its random occurrence in the dataset. It identifies how much more likely the antecedent and consequent are to appear together than independently.

Formula:

$$\text{Lift} = \frac{\text{Confidence}}{\text{Support of consequent}}$$

Example: A lift value greater than 1 indicates a strong positive association, while a value equal to 1 suggests no association. For instance, if the lift is 1.5, it means the antecedent makes the consequent 1.5 times more likely.

How Does Association Rule Learning Work?

Association rule learning is a multi-step process designed to identify meaningful patterns and relationships in large datasets. It involves two main stages:

Identifying Frequent Itemsets: The process begins by identifying frequent itemsets—combinations of items that appear together in transactions with a frequency above a predefined threshold. Metrics like support are used to measure how often these itemsets occur in the dataset. For example, a frequent itemset might reveal that bread and butter are purchased together in 10% of transactions.

Generating Association Rules: Once frequent itemsets are identified, association rules are generated. These rules take the form of if-then statements that describe relationships between items (e.g., "If a customer buys bread, they are likely to buy butter"). Metrics such as confidence and lift are applied to evaluate the strength and reliability of these rules.

Iterative Refinement

The process is iterative, with thresholds for support and confidence adjusted to refine the rules. This ensures that only the most significant and actionable rules are selected. For instance, a rule with low confidence may be excluded from further analysis.

Through this systematic approach, association rule learning uncovers valuable insights from raw data, enabling organizations to make data-driven decisions.

Types of Association Rule Learning Algorithms

Several algorithms are used for association rule learning, each with unique strengths and applications. The three most commonly used algorithms are:

1. Apriori Algorithm

The Apriori algorithm employs a breadth-first search approach to identify frequent itemsets. It relies on the principle that all subsets of a frequent itemset must also be frequent, reducing the search space.

Advantage: Simple to implement and effective for small datasets with low dimensionality.

Limitation: Performance degrades significantly with large or dense datasets due to repeated scanning of the database.

2. Eclat Algorithm

The Eclat algorithm uses a depth-first search strategy to discover frequent itemsets. Instead of scanning the database multiple times, it represents transactions as vertical itemsets and directly computes intersections.

Advantage: Efficient for datasets with sparse data or where there are fewer frequent itemsets.

3. FP-Growth Algorithm

The FP-Growth (Frequent Pattern Growth) algorithm leverages a prefix-tree structure called the FP-tree to represent transactional data compactly. Unlike Apriori, it avoids generating candidate itemsets explicitly, making it faster and more efficient.

Advantage: Significantly faster and more memory-efficient than Apriori, especially for large datasets.

Applications of Association Rules

Association rules are widely applied across various industries to uncover patterns and relationships in data, enabling better decision-making and operational efficiency.

1. Retail and Market Basket Analysis: Retailers use association rules to identify frequently purchased product combinations, helping them optimize store layouts or create product bundles to increase sales.

Example: A supermarket discovers that customers who buy bread often purchase butter and jam, leading to strategic placement of these items together.

2. Healthcare: In healthcare, association rules help discover co-occurrence patterns in symptoms, aiding in diagnostic processes and treatment plans.

Example: Identifying that patients with high blood pressure often have a higher risk of developing diabetes can guide preventative care strategies.

3. E-Commerce and Recommendation Systems: E-commerce platforms leverage association rules to build recommendation systems that enhance user experiences and drive sales.

Example: Amazon's "Customers who bought this also bought" feature suggests complementary products, boosting cross-selling opportunities.

4. Fraud Detection: Association rules are used in financial services to identify unusual patterns in transaction data, which can help detect fraudulent activities.

Example: Flagging transactions that deviate significantly from established spending patterns for further investigation.

Example of Association Rules

Consider a small transaction dataset where customers purchase items like bread, butter, and milk.

Dataset Example:

Transaction ID	Items Purchased
1	Bread, Butter
2	Bread, Milk
3	Bread, Butter, Milk
4	Milk
5	Bread, Butter

Rule Discovery Process:

Rule Example: "If bread is purchased, then butter is likely to be purchased."

1. **Support Calculation:**
Support = Transactions containing both bread and butter ÷ Total transactions

$$\text{Support} = \frac{3}{5} = 0.6\,(60\%)$$

2. **Confidence C**
Confidence = Suppor ead

$$\text{Confidence} = \frac{3}{4} = 0.75\,(75\%)$$

3. **Lift Calculation:**
Lift = Confidence ÷ Support of butter

$$\text{Lift} = \frac{0.75}{0.6} = 1.25$$

A lift value greater than 1 indicates a positive association between bread and butter.

This example demonstrates how association rules are derived and evaluated, providing actionable insights from transactional data.

Unit Highlights

- Understanding **Concept Description** and its role in data mining.
- Learning **Data Generalization and Data Summarization** techniques.
- Studying **Market Basket Analysis** for analyzing customer purchase patterns.
- Understanding **Association Rule Mining** and rule generation.
- Learning **Apriori Algorithm** to find frequent itemsets.
- Exploring **Improved Apriori Techniques** like Hashing, Sampling, and Partitioning.
- Understanding **Incremental ARM** for dynamic databases.
- Studying **Associative Classification** in machine learning.

Case Study

Supermarket Market Basket Analysis

A supermarket collected transaction data from thousands of customers. Using **Association Rule Mining**, they discovered that customers who buy **bread** often buy **butter** and **jam** together.

Using this knowledge, the supermarket placed these products close to each other on the shelves. As a result, customers easily purchased related products, which increased overall sales.

This case study shows how **data mining techniques help businesses understand customer behavior and improve marketing strategies**.

Book References

1. Jiawei Han, Micheline Kamber, Jian Pei – **Data Mining: Concepts and Techniques**
2. Alex Berson & Stephen Smith – **Data Warehousing, Data Mining and OLAP**
3. Pang-Ning Tan, Michael Steinbach – **Introduction to Data Mining**

UNIT- V DATA GENERALIZATION AND SUMMARIZATION		
Part – A		
1	What is concept description?	L1
2	Define data generalization.	L1
3	What is summarization-based characterization?	L2
5	Define attribute relevance.	L1
6	What is class comparison in data mining?	L2
7	Define association rule mining.	L1
8	What is market basket analysis?	L2
9	What is the Apriori algorithm used for?	L1
10	Mention one improvement in the improved Apriori algorithm.	L2
Part – B		
1	What is concept description? Explain data generalization and summarization-based characterization in detail.	L2,L4
2	Explain the relevance of attributes and the method of class comparisons in data summarization.	L2,L5
3	Define association rule mining. Explain market basket analysis with an example.	L2,L3
4	Describe the working of the Apriori algorithm with a suitable example and candidate generation steps.	L3,L4
5	Explain the rule generation process from frequent itemsets using the Apriori approach.	L2,L3
6	Compare Apriori and Improved Apriori algorithms. Explain how efficiency is improved.	L3,L4
7	What is Incremental ARM? How does it differ from traditional ARM approaches?	L2,L4
8	Explain the concept of associative classification. How is it different from traditional classification?	L2,L4