Data Warehouse

A data warehouse is a repository of information collected from multiple sources, stored under a unified schema, and that usually resides at a single site. Data warehouses are constructed via a process of data cleaning, data integration data transformation, data loading, and periodic data refreshing. The data are stored to provided information from a historical perspective (such as from the past 5-10 years) and are typically summarized. For example:→ Storing the details of each sales transaction, the data warehouse may store a summary of the transactions per item type for each store or summarized to a higher level, for each sales regions.

A data warehouse is usually modeled by a multidimensional database structure, where each dimension corresponds to an attribute or a set of attributes in the schema, and each cell stores the values of some aggregate measure such as count or sales-amount.

* Usage

Health care, Banking, Retail, Data Mining.

* Need of Warehouse

— Large data store
— Smart store
— efficient retrieve
→ Heterogenetic

* Willion .H. Inmon

→ Subject oriented
→ Integrated
→ Non-volatile
→ time-variant

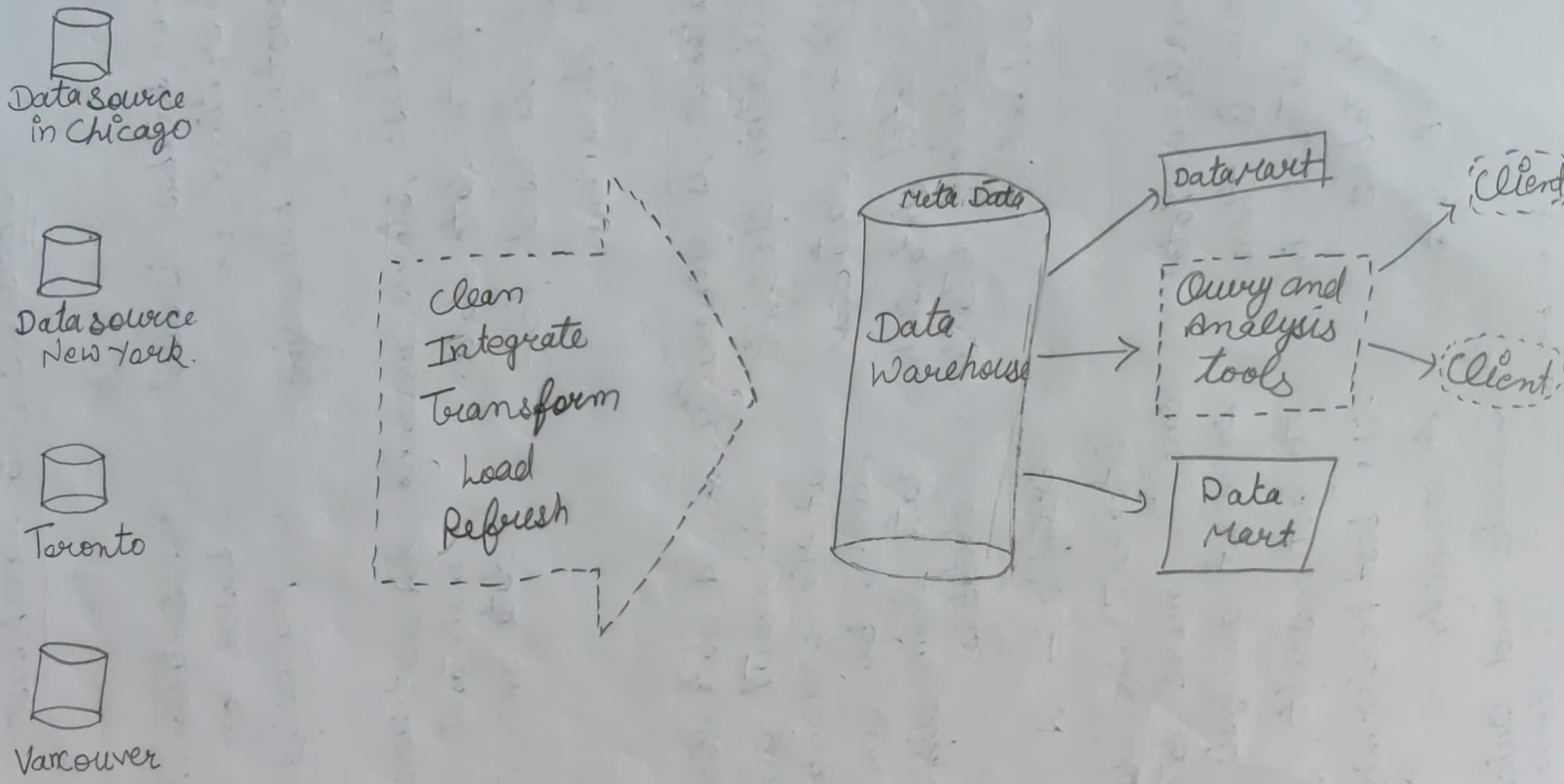
* Challenges face by design Warehouse

→ Data quality
→ Data Analytics
→ User exception
→ Cost
→ Performance
→ quality Accurance.

* Framework of a data Warehouse
* Advantage of data Warehouse

→ To clean data
→ Query processing multiple options
→ High query performance
→ local processing at source warehouse.
→ Easy way of reporting access multiple system.

Framework


Data source in Chicago
Data source New York
Toronto
Vancouver
Clean
Integrate
Transform
Load
Refresh
Meta Data
Data Warehouse
Data Mart
Query and analysis tools
Client
Client

DBMS V/S Data Warehouse

DBMS	Data Warehouse
① A common Database is based on operational or transactional processing.	* A Data warehous is based on analytical processing.
* A database stores currently and up-to-date data. which is used for daily operations.	* A data warehouse maintains historical data over time.
* A database is generally application specific Example - A database stores related data, such as the student details in a school.	* A data warehouse is integrated generally at the organization level, by combining data from different database. Example :- A data warehouse integrates the data from one or more databases, so that analysis be done to get result, such as the best performing school in a city
* Normalized data structure is there in a database in separate tables	* Dynamic and quick analysis of data is done
* 100 MB to GB data store.	* 100 GB to KB

1. Introduction :-

Data Mining is defined as the procedure of extracting ... other words we

Data Marts :- A data mart contains a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to specific selected subjects. For example :- A marketing data mart may confine its subjects to customer, item and sales. The data contained in data marts tend to be summarized.

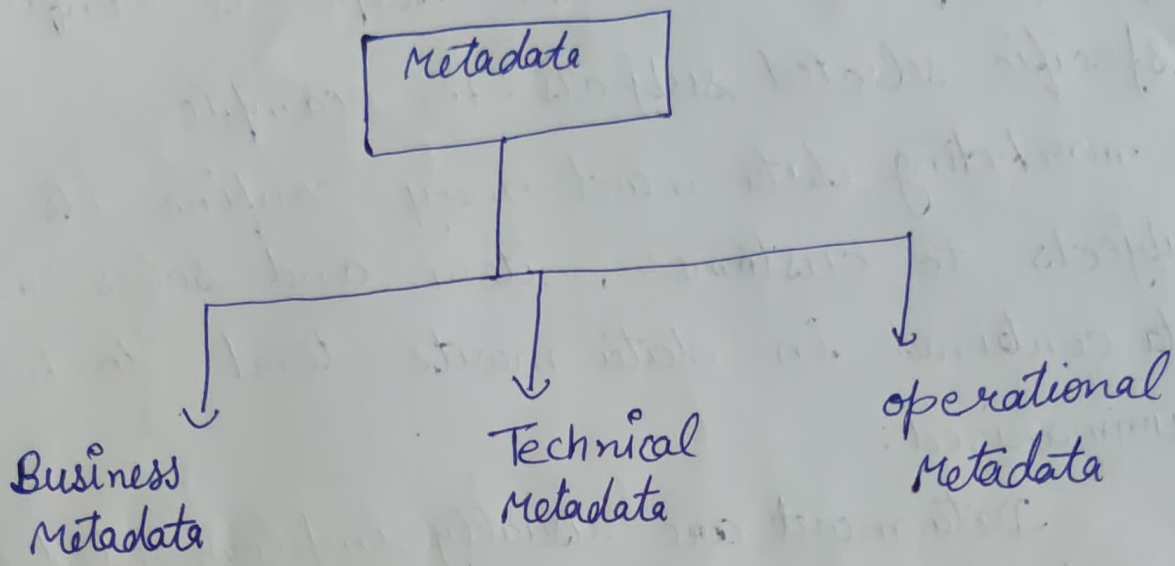
Data mart are usually implemented on low-cost departmental server that are Unix/Linux or Window based.

Depending on the source of data, data marts can be categorized as independent or dependent. Independent data marts are sourced from data captured from one or more operational system or external information providers, or from data generated locally within a particular department or geographic area.

Metadata :- 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.

Categories of Meladata :→



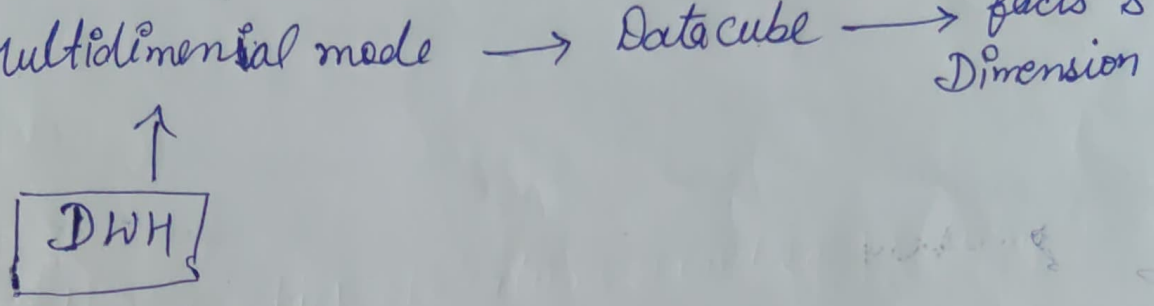
① Business :→ It has the data ownership information, business definition, and changing policies.

② Technical Metadata :- It includes database system names, table and column names and size, data types and allowed values.. Technical Metadata also includes structural information such as primary and foreign key attributes and indices.

③ operational Metadata :— It includes currency of data and data lineage. Currency of data means whether the data is active, archived, or purged.

Introduction :-

Data warehouse and OLAP tool are based on multidimensional data model. In this model data is presented in the form of the data cube where as the data cube is define as represent of the data in multiple dimension It is define in terms of fact & Dimensions



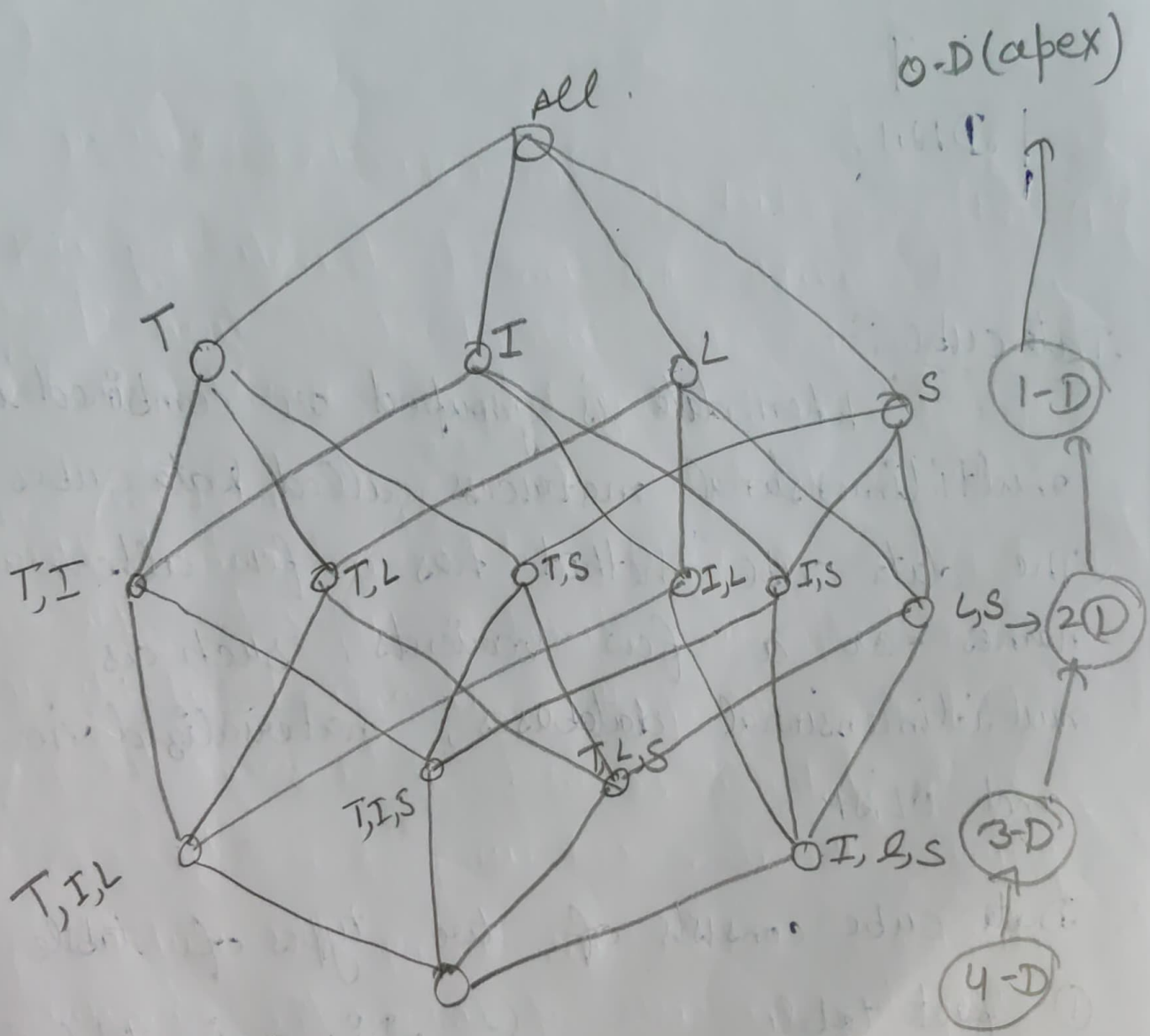
Data cube :-

When data is grouped or combined in multidimensional matricess called Data cubes. The data cube method has a few alternatives names or a few variants, such as multidimensional databases, materialized views, and OLAP

Data cube consists of two types of table

① fact table ② Dimension table

Based on different number of Dimensional or different subsets of data can be created from Data cube. Such subset are known as data cubides. The arrangment of all possible cubides in the worker is known as data lattice. Lattice consists of different level of summarization starting from based cuboids to apex cubid



Time, item, location, supplier

Introduction :-

Data Mining is defined as the procedure of Extractin information from huge sets of data. In other words we can say that data mining is mining knowledge from data.

→ There is a huge amount of data available in the information industry. This data is of no use until it is converted into useful information. It is necessary to analyze this huge amount of data and extract useful information from it.

→ Extraction of information is not only the process we need to perform: Data mining also involves other processes such as data cleaning, Data integration, Data transformation Pattern Evaluation and Data presentation. Once all these processes are over, we would be able to use this information in many applications such as

Market Analysis (M)
Fraud Detection (F)
Customer Retention (C)
Production Control (P)
Science Exploration. (S)

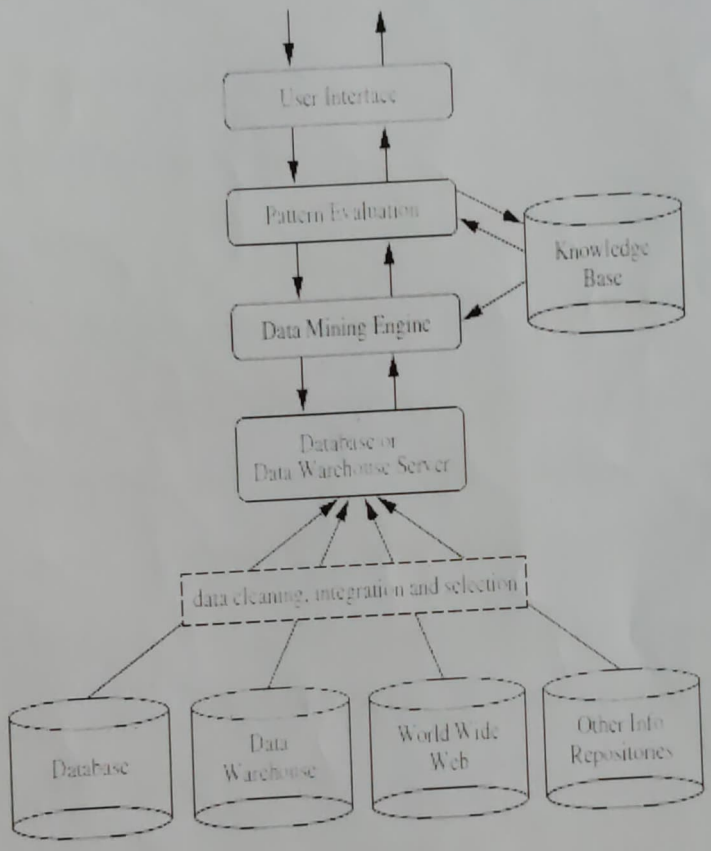
* WHAT IS DATA MINING :-

Data Mining is defined as Extracting information from huge sets of data. The information or knowledge extracted can be used for many applications.

- **Summarization** – providing a more compact representation of the data set, including visualization and report generation.

1.4 Architecture of Data Mining

A typical data mining system may have the following major components.



1. **Knowledge Base:**

This is the domain knowledge that is used to guide the search orevaluate the interestingness of resulting patterns. Such knowledge can include concepthierarchies,

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).

2. **Data Mining 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.

3. **Pattern Evaluation Module:**

This component typically employs interestingness measures interacts with the data mining modules so as to focus thesearch toward interesting patterns. It may use interestingness thresholds to filterout discovered patterns. Alternatively, the pattern evaluation module may be integratedwith the mining module, depending on the implementation of the datamining method used. For efficient data mining, it is highly recommended to pushthe evaluation of pattern interestingness as deep as possible into the mining processso as to confine the search to only the interesting patterns.

4. **User interface:**

Thismodule communicates between users and the data mining system,allowing the user to interact with the system by specifying a data mining query ortask, providing information to help focus the search, and performing exploratory datamining based on the intermediate data mining results. In addition, this componentallows the user to browse database and data warehouse schemas or data structures,evaluate mined patterns, and visualize the patterns in different forms.

and identifying anomalous data that could represent data entry keying errors.

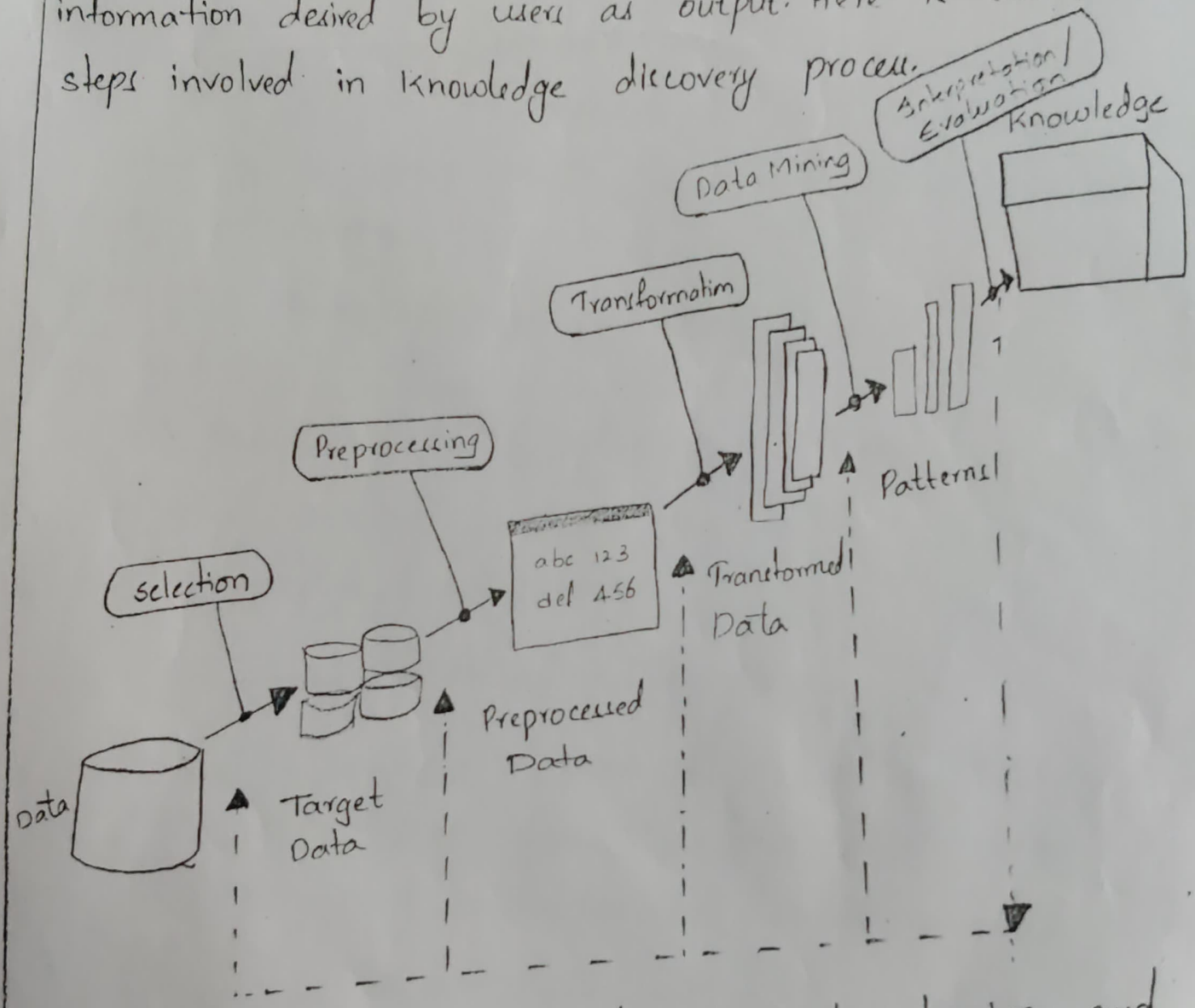
1.3 Tasks of Data Mining

Data mining involves six common classes of tasks:

- **Anomaly detection (Outlier/change/deviation detection)** – The identification of unusual data records, that might be interesting or data errors that require further investigation.
- **Association rule learning (Dependency modelling)** – Searches for relationships between variables. For example a supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes. This is sometimes referred to as market basket analysis.
- **Clustering** – is the task of discovering groups and structures in the data that are in some way or another "similar", without using known structures in the data.
- **Classification** – is the task of generalizing known structure to apply to new data. For example, an e-mail program might attempt to classify an e-mail as "legitimate" or as "spam".
- **Regression** – attempts to find a function which models the data with the least error.

* KDD:-

KDD stands for Knowledge Discovery in Database. Data mining is an essential step in the process of knowledge discovery. There are seven different stages in KDD process. This process takes raw data as input and provides useful information desired by users as output. Here is the list of steps involved in knowledge discovery process.



- Preprocessing of database consists of Data cleaning and Data integration.
- KDD is an iterative process.

Data Warehouse Implementation :→

It is defined by number of cube or cuboid occipy the memory space as the warehouse contant huge volumes of data is become necessary to implement data cuboid for fast data analysis and efficient query processing.

1. Efficient Computation of data cubes
2. The compute cube operation
3. Indexing OLAP Data (Modeling OLAP data)
4) Efficient processing OLAP queries.
5) OLAP Server architecture.

1.) Efficient computation of data cubes.

- i) Data cube can be viewed as a lattice of cuboids.
  The bottom-most cuboids is the base cuboids.
  The top-most cuboids (apex) contains only one cells.

ii) Materialization of data cube :→

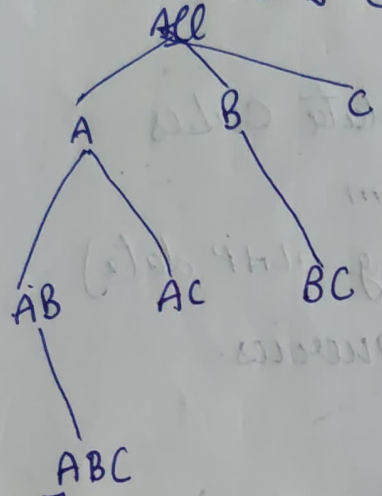
→ Full materialization refers to the computation of all cuboids in a data cube lattice.

→ Partial materialization refers to the selective computation of a subset of the cuboid cells in the lattice. Iceberg cubes and shell fragments are examples of partial materialization.

Association Rule Mining :→

i) iceberg cube is a data cube that stores only those cube cells that have an aggregate value some minimum support threshold

ii) shell fragments of a data cube, only some cuboids involving a small number of dimensions are computed; and queries on additional combinations of the dimensions can be computed on the fly.



iceberg cube.

→ No materialization :- Do not precompute any of the nonbase cuboids.

③ Indexing OLAP Data :→

Indexing is another approach which can help efficient data searching.

Two types of indexing :-

i) Bitmap ii) Join

i) Bitmap :→

Bitmap indexing ~~is~~ allow you to reduce the response time of the bigger SQL queries.

Association rule [illegible]

Join Index table

Location	Sales_key
main street	T57
"	T238
"	T884

Join index table

item	Sales_key
----	----
Sony-TV	T57
Sony-TV	T459
-- -	- -

Two dimension

--	---	-- -
main Street	Sony-TV	T57
- -	----	---

④ OLAP queries:→

i.) Determine which operations should be perform on the available cuboids:- This involves transforming any selection, projection, roll-up, drill-down operations specified in the query into correspoind SQL and/ OLAP operations.

ii.) Determine to which materialized cuboid the relevent operations should be applied:- This involved identifying all of the materialized cuboids that may potentially be used to answer the query, pruning the above set using knowledge of dominance relationships among the cuboids, estimating the costs of using the remaining materialized cuboids, and selecting the cuboid with the least cost.

The bitmap indexing method is popular in OLAP products because it allow quick searching in data cubes. The bitmap index is an alternative representation of the RID (Record-ID)

→

RID	item	city
R_1	H	D
R_2	C	D
R_3	P	D
R_4	S	D
R_5	H	M
R_6	C	M
R_7	P	M
R_8	S	M

Item bitmap

RID	H	C	P	S
R_1	1	0	0	0
R_2	0	1	0	0
R_3	0	0	1	0
R_4	0	0	0	1
R_5	1	0	0	0
R_6	0	1	0	0
R_7	0	0	1	0
R_8	0	0	0	1

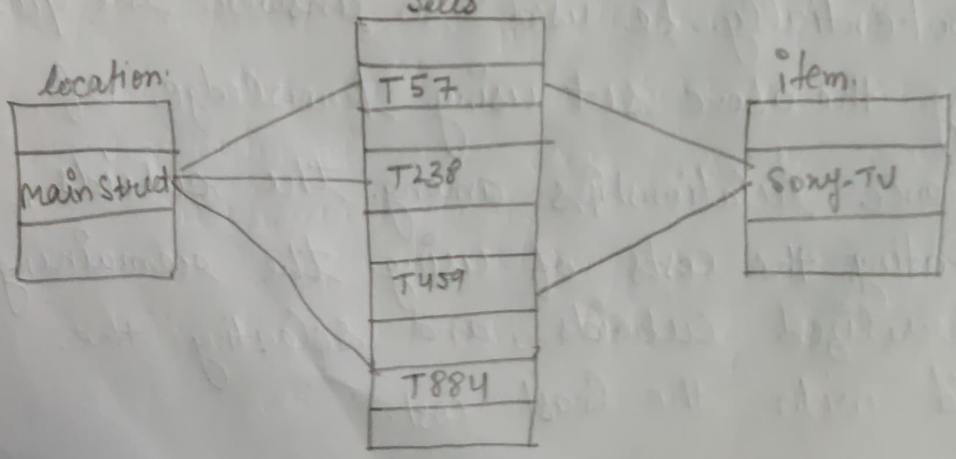
City

RID	D	M
R_1	1	0
R_2	1	0
R_3	1	0
R_4	1	0
R_5	0	1
R_6	0	1
R_7	0	1
R_8	0	1

H → Home entertainment
C → computer
P → Phone
S → Security

D → Delhi
M → Mumbai

Join Indexing:- This indexing is popular used in relational query process. It registered the join row to different table for faster data ~~acces~~ access or join indexing registers the joinable rows of two relations from a relational database.



Unit-3

OLAP Server Architectures.

*Relational OLAP (ROLAP) :→ These are the intermediate servers that stand in between a relational back-end server and front-end tools. They use a relational or extended-relational DBMS to store and manage warehouse data and OLAP middleware to support missing pieces. ROLAP servers include optimization for each DBMS back-end, implementation of aggregation navigation logic, and additional tools and services. ROLAP technology tends to have greater scalability than MOLAP technology.

*Multidimensional OLAP :→ These servers support multi-dimensional views of data through array-based multidimensional storage engines. They map multidimensional views direct to data cube array structure. Many OLAP servers adopt a two-level storage representation to handle sparse and dense data sets: The dense subcubes are identified and stored as array structure, while the ~~sparse~~ sparse subcubes employ compression technology for efficient storage utilization.

* Hybrid OLAP:- The hybrid OLAP approach combines ROLAP and MOLAP technology, benefitting from the greater scalability of ROLAP and the faster computation of MOLAP. for example, a HOLAP server may allow large volumes of detail data to be stored in a relational database, while aggregations are kept in a separate MOLAP store.

* Specialized SQL Servers:- To meet the growing demand of OLAP processing in relational database, some relational and data warehousing ~~firms~~ firms implement specialized SQL servers which provide advanced query language and query processing support for SQL queries over star and snowflake schemas in a read-only environment.

Data-Warehouse Back-End tools :-

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

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

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

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

→ Refresh :- which propagates the updates from the data sources to the warehouse.

Unit -3

Association Rule Mining :→

Association rule learning is a type of unsupervised learning technique that checks for the dependency of one data item on another data item and maps accordingly so that it can be more profitable. It tries to find some intersting relations or associations among the variables of datasets. The association rule learning is one of the very important concept of machine learning, and it is employed in market based Analysis, Web usage mining, continuous production etc.

For example, if a customer buys bread he most likely can also buy butter, eggs, or milk. So these products are stored within a shelf or mostly nearby.

Association rule learning can be divided into three type of algorithms :—

1) Apriori
2. Eclat
3 F-P Growth Algorithm

Apriori Algorithm

It uses frequent itemset to generate association rules & it is designed to work on datasets or databases that contain transactions.

This algo. uses a BFS & Hash tree to calculate the itemset associations efficiently.

Frequent itemset :— FI are those items whose support is greater than threshold value.

Eg :— A = {1,2, 3,4,5} B = {2,3,7}

2,3 are FI values.

Step-I Determine support of itemsets & Select min support & confidence.

Step-II Take all supports in transaction with higher support value than min or selected support value.

Step-III Find all rules of these sets that have higher confidence value than threshold or min confidence

Step-IV Sort the rules as decreasing order of lift.

Apriori Algorithm:-

It uses frequent itemsets to generate association rules & it is designed to work on datasets or databases that contain transactions.

This algo. uses a BFS & Hash tree to calculate the itemset associations efficiently.

Frequent itemset:- FI are those items whose support is greater than threshold value.

Eg:- A = {1,2, 3,4,5} B = {2,3,7}

2,3 are FI values.

Step-I: Determine support of itemsets & Select min support & confidence.

Step-II Take all supports in transaction with higher support value than min or selected support value.

step-III Find all rules of these sets that have higher confidence value than threshold or min confidence

Step-IV Sort the rules as decreasing order of lift.

Apriori Algo.

Eg:-

Tid.	Itemsets
T_1	A, B
T_2	B, D
T_3	B, C
T_4	A, B, D
T_5	A, C
T_6	B, C
T_7	A, C
T_8	A, B, C, E
T_9	A, B, C

Given :

min support = 2.

min confidence = 50%.

C → Customer

FI → Frequent itemset.

Step ① Calculating C_1 and F_1

C_1

Itemset	Support count
A	6
B	7
C	6
D	2
E	1

→ F_1

Itemset	SC
A	6
B	7
C	6
D	2

Step ② Calculating C_2 and F_2.

C_2

Itemset	SC
(A, B)	4
(A, C)	4
(A, D)	1
(B, C)	4
(B, D)	2
(C, D)	0

→ F_2

Itemset	SC
(A, B)	4
(B, C)	4
(A, C)	4
(B, D)	2

Step ③ Candidate Generation C_3 & F_3

Is	SC
(A, B, C)	2
(B, C, D)	0
(A, C, D)	0
(A, B, D)	1

→ F_3

Itemset	SC
(A, B, C)	2

Step(4) Finding Association rules for subsets:-

→ We will calculate the confidence using

$$\frac{Sup(A \cap B)}{A} \rightarrow$$ ~~Sup (A∩B)∩C~~ $Sup = \frac{A \rightarrow B}{A}$

→ After calculating we will exclude rules that have less ~~than~~ confidence than min threshold (50%)

Rules	Support	Confidence.
$A \cap B \rightarrow C$	2	$Sup((A \cap B) \cap C) / Sup(A \cap B) = 2/4 = 50\%$
$B \cap C \rightarrow A$	2	$Sup((B \cap C) \cap A) / Sup(B \cap C) = 2/4 = 50\%$
$A \cap C \rightarrow B$	2	$Sup((A \cap C) \cap B) / Sup(A \cap C) = 2/4 = 50\%$
$C \rightarrow A \cap B$	2	$Sup((C) \cap (A \cap B)) / Sup(C) = 2/5 = 40\%$
$A \rightarrow B \cap C$	2	$Sup(A) (B \cap C) / Sup(A) = 2/6 = 33.33\%$
$B \rightarrow A \cap C$	2	$Sup((B)(A \cap C) / Sub(B) = 2/7 = 28.\%$

Frequent Pattern Growth Algo: → It is improved version of Apriori Algo. It represents the database in the form of tree structure. that is known as frequent pattern or tree.

Tid	Item.
T_1	E, K, M, N, O, Y
T_2	D, E, K, N, O, Y
T_3	A, E, K, M
T_4	C, K, M, U, Y
T_5	C, E, I, K, O, O

min Support = 3.

(2)

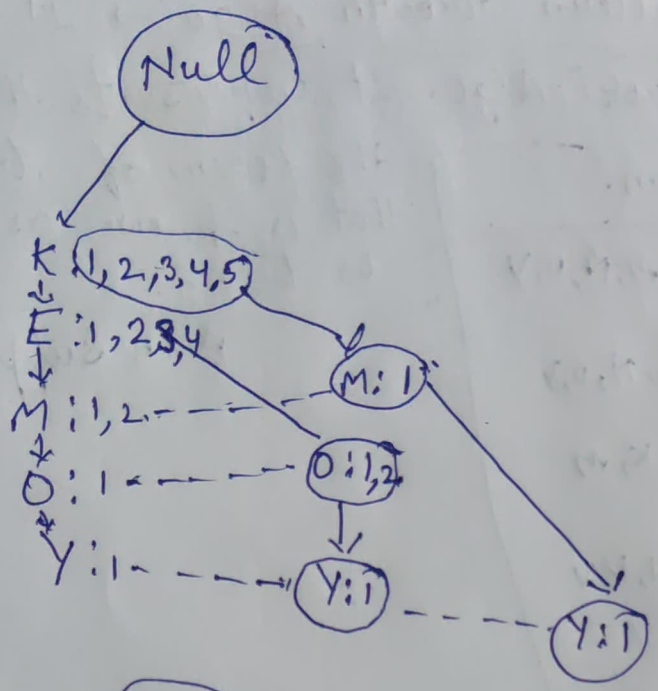
Item	Frequency
A	1
C	2
D	1
E	4
I	1
K	5
M	3
N	2
U	1
O	3
Y	3

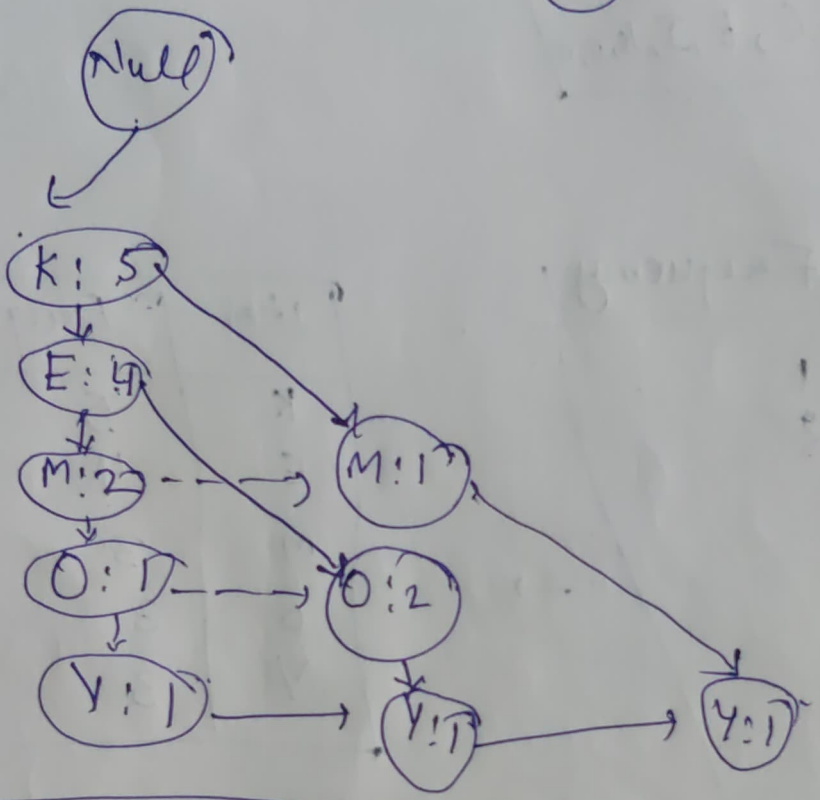
→

Item	Frequency
K	5
E	4
M	3
O	3
Y	3

(3)

Tid	Items	ordered - Itemset
T_1	E, K, M, N, O, Y	K, E, M, O, Y
T_2	D, E, K, N, O, Y	K, E, O, Y
T_3	A, E, K, M	K, E, M
T_4	C, K, M, U, Y	K, M, Y
T_5	C, E, I, K, O, O	K, E, O





Item.	Conditional Pattern Base	Conditional Freq.
Y.	{K,E,M,O : 1}, {K,E,O : 1}, {K,M,Y : 1}	{K : 3}
O	{K,E,M : 1}, {K,E : 2}	{K,E : 3}
M.	{K,E : 2}, {K : 1}	{K : 3}
E	{K : 4}	{K : 4}

Items.	Frequent Pattern Generated
Y	{<K, Y : 3>}
O	{<K, O : 3>}, {<E, O : 3>}, {<K, E, O : 3>}
M	{<K, M : 3>}
E	{<K, E : 3>}
K	___

$K \rightarrow Y$, $Y \rightarrow K$

$K \rightarrow O$, $E \rightarrow O$, $O \rightarrow K$, $O \rightarrow E$, $E \rightarrow K$, $K \rightarrow E$

$K \rightarrow M$, $M \rightarrow K$.

$K \rightarrow E$, $E \rightarrow K$.

* Eclat Algorithm

Eclat Algorithm stands for Equivalence class. Clustering and bottom-up Lattice Traversal It is one of popular methods of Association rule mining Algorithm.

Transaction id	Bread	Butter	milk	Coke	Jam
T_1	1	1	0	0	1
T_2	0	1	0	1	0
T_3	0	1	1	0	0
T_4	1	1	0	1	0
T_5	1	0	1	0	0
T_6	0	1	1	0	0
T_7	1	0	1	0	0
T_8	1	1	1	0	1
T_9	1	1	1	0	1

K=1, Minimum Support = 2

Item	Tidset
Bread	$T_1, T_4, T_5, T_7, T_8, T_9$
Butter	$T_1, T_2, T_3, T_4, T_6, T_8, T_9$
Milk	$T_3, T_5, T_6, T_7, T_8, T_9$
Cake	T_2, T_4
Jam	T_1, T_8

K=2, MS=2.

item	Tidset
(B, Bu)	T_1, T_4, T_8, T_9
(B, M)	T_5, T_7, T_8, T_9
(B, C)	T_4
(B, J)	T_1, T_8
(Bu, M)	T_3, T_6, T_8, T_9
(Bu, C)	T_2, T_4
(Bu, J)	T_1, T_8
(M, C)	∅
(M, J)	T_8
(C, J)	∅

K=3, MS=2.

item	Tidset
B, BU, M	T_8, T_9
B, BU, J	T_8, T_1
B, BU, C	T_4
B, M, J	T_8

K=4.

item	Tidset
B, BU, M, J	T_8

Advantages:

1. **Memory Requirements** :→ ECLAT Algo. uses a Depth-first Search Approach, it uses less memory than Apriori.

- **Speed** :→ The ELCAT Algo. is typically faster than Apriori Algo.

- **Number of Computations** :— The ECLAT Algo. does not involves the repeated scanning of the data to compute the individual support values.

Decision Tree :-

Decision tree algorithm falls under the category of supervised learning. The can be used to solve both regression and classification problems. It is a tree that helps us in decision making. purpose. the decision tree creates classification or regression models as a tree structure. Decision tree uses the tree representation to solve the problem decision tree contains ~~tree~~ 3 types of nodes.

1. **root node**:- It is the top most node in the tree data which is inside the node is known as attribute.

2. **Internal node**:→ Each internal node denotes a test on attribute. nodes which are in between root node ~~and~~ are called as internal nodes.

3. **Leaf node**:- We call last nodes as leaf nodes. They represents output is class label.

Advantages: 1.) It does not require any domain knowledge.

2. classification steps of decision tree are simple and fast.
3. Missing values in data does not effect output.
4. A decision tree model is automatic and does not require a standardization of data.

key factors :- Building a decision tree is all about discovery attributes that return the highest data gain

Entropy :→

Entropy refers to a common way to measure impurity. In the decision tree. measure the impurity in dataset.

Information Gain :-

Information Gain refers to the decline in entropy after the dataset is split. It is also called as entropy reduction.

Example :-

Day	Weather	Temperature	Humidity	Wind	Play
1	Sunny	Hot	High	weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Cloudy	mild	High	Strong	Yes
4	Rainy	mild	High	Strong	No
5	Sunny	mild	Normal	Strong	Yes
6	Rainy	Cool	Normal	Strong	No
7	Rainy	mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	mild	High	Strong	No