CBCS SCHEME

USN

21CS71

Seventh Semester B.E./B.Tech Degree Examination, Dec.2024/Jan.2025

Big Data Analytics

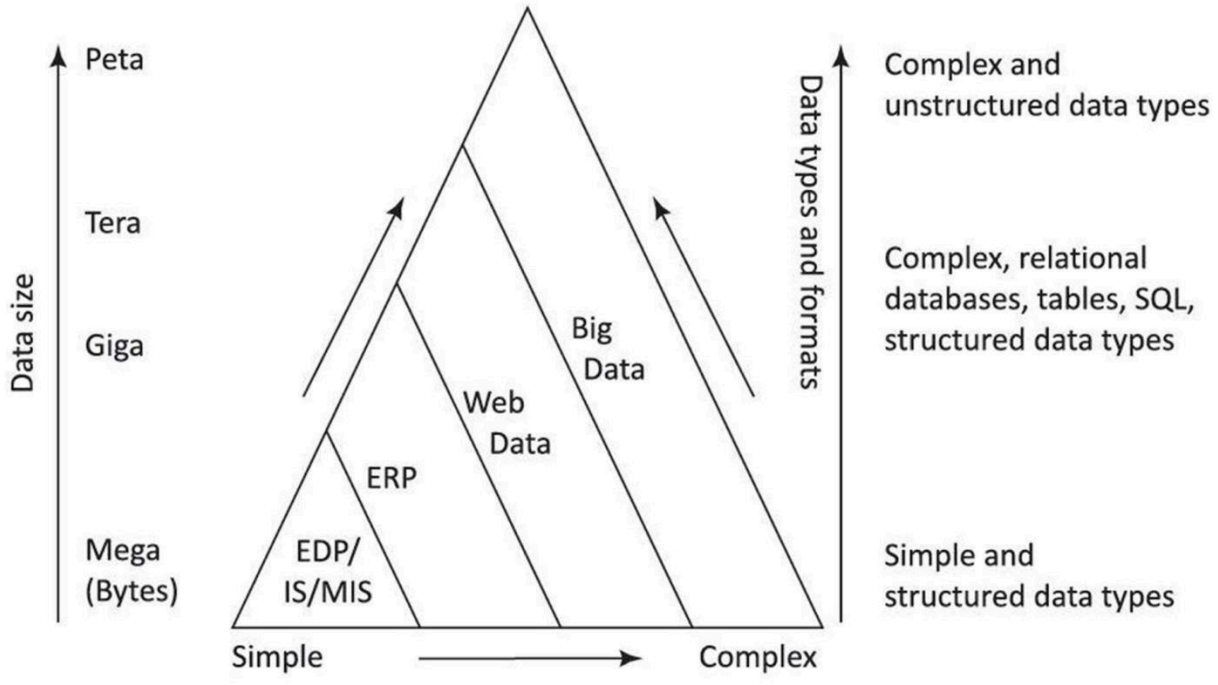
Big Data Analytics VTU Solution

Module-1

1 a. Discuss the evolution of Big Data. (06 Marks)

b. Explain the characteristics of Big Data. (04 Marks)

c. Explain Data Architecture Design, with a neat diagram. (10 Marks)



The rise in technology has led to the production and storage of voluminous amounts of data. Earlier megabytes (10^6 B) were used but nowadays petabytes (10^{15} B) are used for processing, analysis, discovering new facts and generating new knowledge.

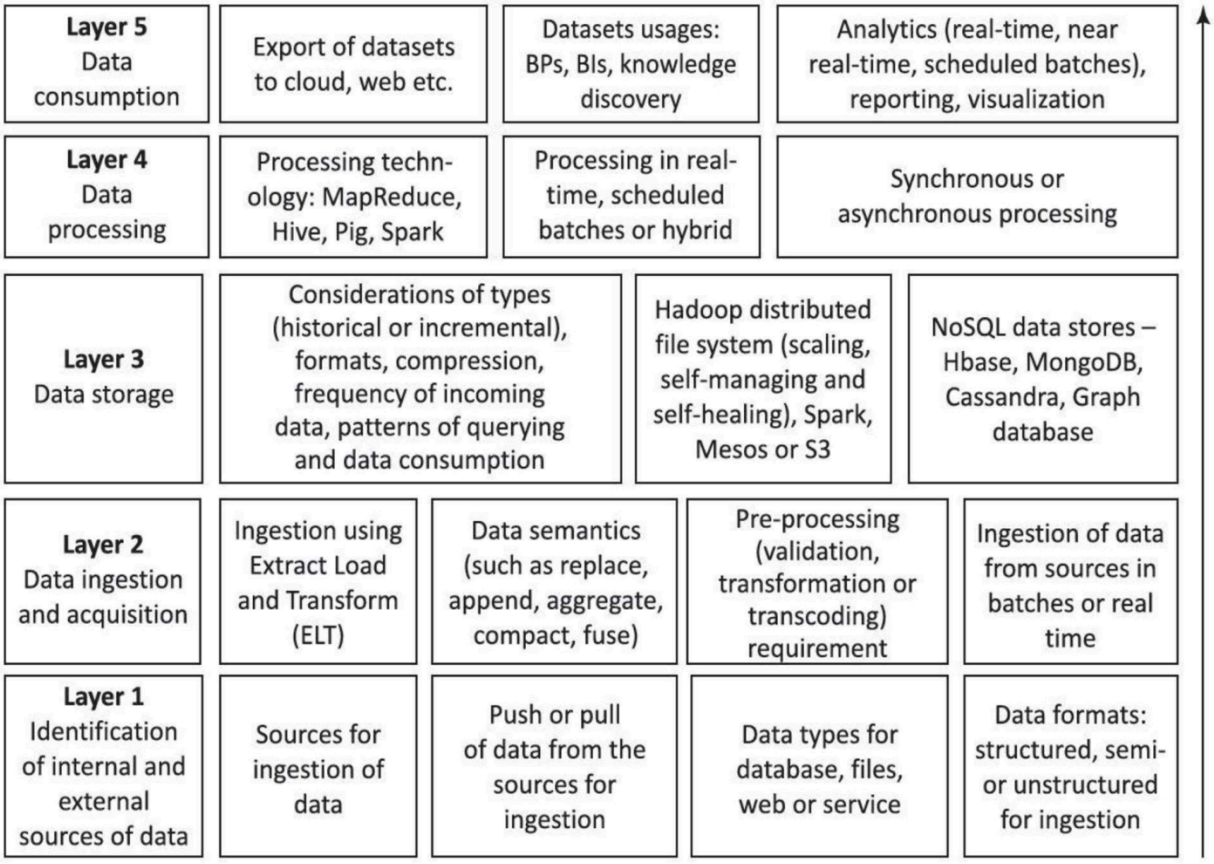
Conventional systems for storage, processing and analysis pose challenges in large growth in volume of data, variety of data, various forms and formats, increasing complexity, faster generation of data and need of quickly processing, analyzing and usage.Figure 1.1 shows data usage and growth. As size and complexity increase, the proportion of unstructured data types also increase.An example of a traditional tool for structured data storage and querying is RDBMS. Volume, velocity and variety (3Vs) of data need the usage of number of programs and tools for analyzing and processing at a very high speed.

1.b

Big Data Characteristics

- Volume: is related to size of the data
- Velocity: refers to the speed of generation of data.
- Variety: comprises of a variety of data
- Veracity: quality of data captured, which can vary greatly, affecting its accurate analysis

1.C



- identification of data sources,
- acquisition, ingestion, extraction, pre-processing, transformation of data,
- Data storage at files, servers, cluster or cloud,
- data-processing,
- data consumption in the number of programs and tools.

Logical layer 1 (L1) is for identifying data sources, which are external, internal or both. The layer 2 (L2) is for data-ingestion.Data ingestion means a process of absorbing information, just like the process of absorbing nutrients and medications into the body by eating or drinking them .Ingestion is the process of obtaining and importing data for immediate use or transfer. Ingestion may be in batches or in real time using pre-processing or semantics.

Layer 1

- L1 considers the following aspects in a design:
    - Amount of data needed at ingestion layer 2 (L2)

- Push from L1 or pull by L2 as per the mechanism for the usages
- Source data-types: Database, files, web or service

- Source formats, i.e., semi-structured, unstructured or structured.

Layer 2

- Ingestion and ETL processes either in real time, which means store and use the data as generated, or in batches.
- Batch processing is using discrete datasets at scheduled or periodic intervals of time.

Layer 3

- Data storage type (historical or incremental), format, compression, incoming data
- frequency, querying patterns and consumption requirements for L4 or L5
- Data storage using Hadoop distributed file system or NoSQL data stores—HBase, Cassandra, MongoDB.

Layer 4

- Data processing software such as MapReduce, Hive, Pig, Spark, Spark Mahout, Spark Streaming
- Processing in scheduled batches or real time or hybrid
- Processing as per synchronous or asynchronous processing requirements at L5.

Layer 5

- Data integration
- Datasets usages for reporting and visualization
- Analytics (real time, near real time, scheduled batches), BPs, BIs, knowledge discovery
- Export of datasets to cloud, web or other systems

OR

2 a. Explain Analytics Scalability to Big Data and Massive parallel processing platforms. (12 Marks)

b. Explain Big Data Analytics applications with one case study. (08 Marks)

2.a Analytical Scalability

Vertical scalability means scaling up the given system's resources and increasing the system's analytics, reporting and visualization capabilities. This is an additional way to solve problems of greater complexities. Scaling up means designing the algorithm according to the architecture that uses resources efficiently.

x terabyte of data take time t for processing, code size with increasing complexity increase by factor n, then scaling up means that processing takes equal, less or much less than (n * t).

Horizontal scalability means increasing the number of systems working in coherence and scaling out the workload. Processing different datasets of a large dataset deploys horizontal scalability. Scaling out means using more resources and distributing the processing and storage tasks in parallel. The easiest way to scale up and scale out execution of analytics software is to implement it on a bigger machine with more CPUs for greater volume, velocity, variety and complexity of data. The software will definitely perform better on a bigger machine.

Massive Parallel Processing Platforms

Parallelization of tasks can be done at several levels:

- distributing separate tasks onto separate threads on the same CPU
- distributing separate tasks onto separate CPUs on the same computer
- distributing separate tasks onto separate computers.

2.b

Data are important for most aspect of marketing, sales and advertising. Customer Value (CV) depends on three factors - quality, service and price. Big data analytics deploy large volume of data to identify and derive intelligence using predictive models about the individuals. The facts enable marketing companies to decide what products to sell.

A definition of marketing is the creation, communication and delivery of *value* to customers. Customer (desired) value means what a customer desires from a product. Customer (perceived) value means what the customer believes to have received from a product after purchase of the product. Customer value analytics (CVA) means analyzing what a customer really needs. CVA makes it possible for leading marketers, such as Amazon to deliver the consistent customer experiences.

Big Data Analytics in Detection of Marketing Frauds

Big Data analytics enable fraud detection. Big Data usages has the following features-for enabling detection and prevention of frauds:

- Fusing of existing data at an enterprise data warehouse with data from sources such as social media, websites, biogs, e-mails, and thus enriching existing data
- Using multiple sources of data and connecting with many applications
- Providing greater insights using querying of the multiple source data
- Analyzing data which enable structured reports and visualization
- Providing high volume data mining, new innovative applications and thus leading to new business intelligence and knowledge discovery

---

Module-2

3 a. List and explain the core components of Hadoop. (10 Marks)
b. Explain Hadoop Distributed File System. (10 Marks)

3.a

The Hadoop core components of the framework

are:

Hadoop Common - The common module contains the libraries and utilities that are required by the other modules of Hadoop. For example, Hadoop commonly provides various components and interfaces for distributed file systems and general input/output. This includes serialization, Java RPC (Remote Procedure Call) and file-based data structures.

Hadoop Distributed File System (HDFS) - A Java-based distributed file system which can store all kinds of data on the disks at the clusters.

MapReduce vl - Software programming model in Hadoop 1 using Mapper and Reducer. The vl processes large sets of data in parallel and in batches.

YARN - Software for managing resources for computing. The user application tasks or sub- tasks run in parallel at the Hadoop, uses scheduling and handles the requests for the resources in distributed running of the tasks.

MapReduce v2 - Hadoop 2 YARN-based system for parallel processing of large datasets and distributed processing of the application tasks.

3.b.

HDFS is a core component of Hadoop. HDFS is designed to run on a cluster of computers and servers at cloud-based utility services.

HDFS stores Big Data which may range from GBs (1 GB= 230 B) to PBs (1 PB=

1015 B, nearly the 250 B). HDFS stores the data in a distributed manner in order to compute fast. The distributed data store in HDFS stores data in any format regardless of schema.

HDFS Storage

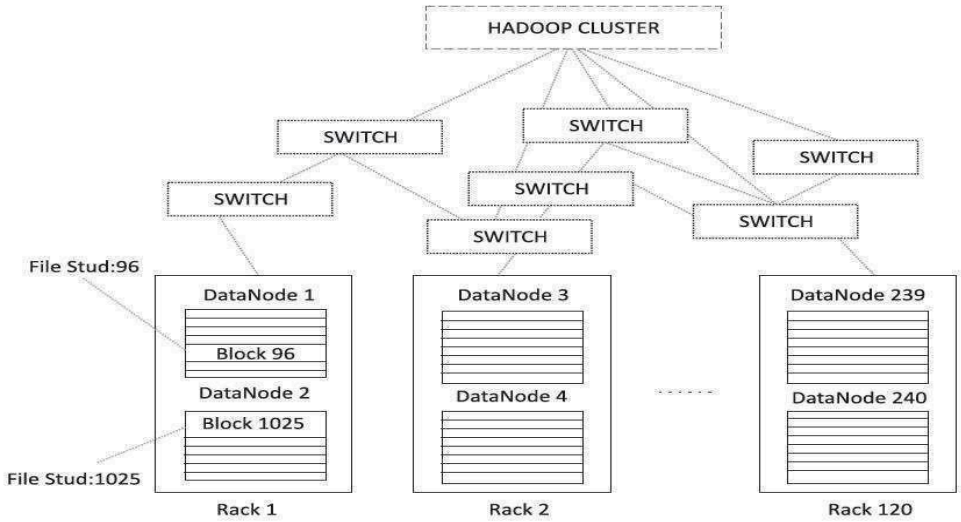
Hadoop data store concept implies storing the data at a number of dusters. Each cluster has a number of data stores, called racks. Each rack stores a number of DataNodes. Each DataNode has a large number of data blocks. The racks distribute across a cluster. The nodes have processing and storage capabilities. The nodes have the data in data blocks to run the application tasks. The data blocks replicate by default at least on three DataNodes in the same or remote nodes.

Data at the stores enable running the distributed applications including analytics, data mining, OLAP using the clusters. A file, containing the data, divides into data blocks. A data block default size is 64 MBs

Hadoop HDFS features are as follows

i. Create, append, delete, rename and attribute modification functions

ii. Content of individual file cannot be modified or replaced but appended with new data at



OR

4 a. Define MapReduce Frame work and its functions. **(06 Marks)**

b. Explain steps on the request to MapReduce and the types of process in MapReduce. **(10 Marks)**

c. Explain in brief on Flume Hadoop Tool. **(04 Marks)**

4.a

Mapper means software for doing the assigned task after organizing the data blocks imported using the keys. A key specified in a command line of Mapper. The command maps the key to the data, which an application uses.

Reducer means software for reducing the mapped data by using the aggregation, query or user-specified function. The reducer provides a concise cohesive response for the application.

Aggregation function means the function that groups the values of multiple rows together to

result in a single value of more significant meaning or measurement. For example, functions such as count, sum, maximum, minimum, deviation and standard deviation.

Querying function means a function that finds the desired values. For example, a function for finding the best student of a class who has shown the best performance in an examination.

MapReduce allows writing applications to process huge amounts of data, in parallel, on large clusters of servers. The cluster size does not limit as such to process in parallel. The parallel programs of MapReduce are useful for performing large scale data analysis using multiple machines in the cluster.

Features o fMapReduce framework are as follows:

- Provides automatic parallelization and distribution of computation based on several processors
- Processes data stored on distributed clusters of DataNodes and racks
- Allows processing large amount of data in parallel
- Provides scalability for usages of large number of servers
- Provides Map Reduce batch-oriented programming model in Hadoop version 1
- Provides additional processing modes in Hadoop 2 YARN-based system and enables required parallel processing. For example, for queries, graph databases, streaming data, messages, real-time OLAP and ad hoc analytics with Big Data 3V characteristics.

4.b

MapReduce Workflow Steps

1. Input Splitting

The input data is split into fixed-size blocks (e.g., 128MB or 64MB).

Each block is processed by a separate Map task.

2. Mapping (Map Phase)

Each split is passed to a Mapper function.

The Mapper processes the data and outputs it in the form of key-value pairs:

map(input) → list(key, value)

3. Shuffling & Sorting

The intermediate key-value pairs are grouped by key.

Values with the same key are shuffled together and sorted.

This prepares them for the Reduce phase.

4. Reducing (Reduce Phase)
The Reducer function takes each key and its list of values:
reduce(key, list_of_values) → output

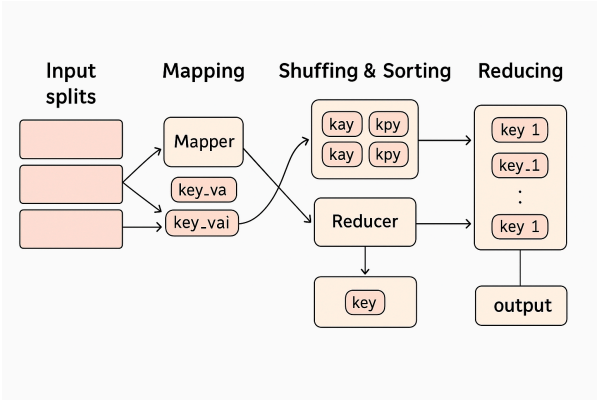
It processes them to produce the final output.

5. Final Output
The output from each Reducer is written to HDFS (or another file system).

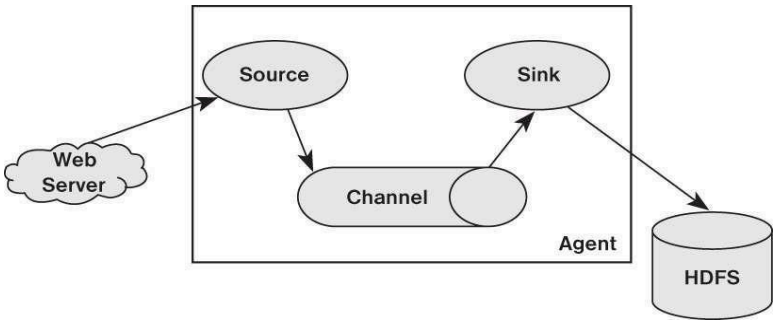
Typically stored as files, one per reducer.

🗂️ Example: Word Count

Input Text	**"Hello world, hello MapReduce"**
Map Output	(hello, 1), (world, 1), (hello, 1), (MapReduce, 1)
Shuffle/Sort	hello → [1, 1], world → [1], MapReduce → [1]
Reduce Output	(hello, 2), (world, 1), (MapReduce, 1)



4.c



Apache Flume is an independent agent designed to collect, transport, and store data into HDFS. Often data transport involves a number of Flume agents that may traverse a series of machines and locations. Flume is often used for log files, social media-generated data, email messages, and just about any continuous data source. As shown in Figure 7.3, a Flume agent is composed of three components.

- *Source.* The source component receives data and sends it to a channel. It can send the data to more than one channel. The input data can be from a real-time source (e.g., weblog) or another Flume agent.
- *Channel.* A channel is a data queue that forwards the source data to the sink destination. It can be thought of as a buffer that manages input (source) and output (sink) flow rates.
- *Sink.* The sink delivers data to destination such as HDFS, a local file, or another Flume agent.

A Flume agent must have all three of these components defined. A Flume agent can have several sources, channels, and sinks. Sources can write to multiple channels, but a sink can take data from only a single channel. Data written to a channel remain in the channel until a sink removes the data. By default, the data in a channel are kept in memory but may be optionally stored on disk to prevent data loss in the event of a network failure.

Module-5

5 a. Explain about No SQL datastore and its characteristics. (10 Marks)

b. Describe the principle of working of the CAP theorem. (10 Marks)

5.a NOSQL DATA STORE

SQL is a programming language based on relational algebra. It is a declarative language and it defines the data schema . SQL creates databases and RDBMS s. RDBMS uses tabular data stores with relational algebra, precisely defined operators with relations as the operands. Relations are a set of tuples. Tuples are named attributes. A tuple identifies uniquely by keys called candidate keys.

ACID Properties in SQL Transactions

Atomicity of transaction means all operations in the transaction must complete, and if interrupted, then must be undone (rolled back). For example, if a customer withdraws an amount then the bank in first operation enters the withdrawn amount in the table and in the next operation modifies the balance with new amount available. Atomicity means both should be completed, else undone if interrupted in between.Consistency in transactions means that a transaction must maintain the integrity constraint, and follow the consistency principle. For example, the difference of sum of deposited amounts and withdrawn amounts in a bank account must equal the last balance. All three data need to be consistent. Isolation of transactions means two transactions of the database must be isolated from each other and done separately. Durability means a transaction must persist once completed

NOSQL

A new category of data stores is NoSQL (means Not Only SQL) data stores. NoSQL is an altogether new approach of thinking about databases, such as schema flexibility, simple relationships, dynamic schemas, auto sharding, replication, integrated caching, horizontal scalability of shards, distributable tuples, semi-structures data and flexibility in approach. Issues with NoSQL data stores are lack of standardization in approaches, processing difficulties for complex queries, dependence on eventually consistent results in place of consistency in all states.

Big Data NoSQL

NoSQL records are in non-relational data store systems. They use flexible data models. The records use multiple schemas. NoSQL data stores are considered as semi-structured data. Big Data Store uses NoSQL.

5.b

CAP Theorem Among C, A and P, two are at least present for the application/service/process. Consistency means all copies have the same value like in traditional DBs. Availability means at

least one copy is available in case a partition becomes inactive or fails. For example, in web applications, the other copy in the other partition is available. Partition means parts which are active but may not

cooperate (share) as in distributed DBs.

1. Consistency in distributed databases means that all nodes observe the same data at the same time. Therefore, the operations in one partition of the database should reflect in other related partitions in case of distributed database. Operations, which change the sales data from a specific showroom in a table should also reflect in changes in related tables which are using that sales data.

2. Availability means that during the transactions, the field values must be available in other partitions of the database so that each request receivesa response on success as well as failure. (Failure causes the response to request from the replicate of data). Distributed databases require transparency between one another. Network failure may lead to data unavailability in a certain partition in case of no replication.

Replication ensures availability.

3. Partition means division of a large database into different databases without affecting the operations on them by adopting specified procedures.

4. Partition tolerance: Refers to continuation of operations as a whole even in case of message loss, node failure or node not reachable.

Brewer's CAP (consistency, Availability and fartition Tolerance) theorem demonstrates that any distributed system cannot guarantee C, A and P together.Consistency- All nodes observe the same data at the same time.

2. Availability- Each request receives a response on success/failure.

3. Partition Tolerance-The system continues to operate as a whole even in case of message

loss, node failure or node not reachable.

Partition tolerance cannot be overlooked for achieving reliability in a distributed database

system. Thus, in case of any network failure, a choice canbe:

• Database must answer, and that answer would be old or wrong data (AP).

• Database should not answer, unless it receives the latest copy of the data(CP).

The CAP theorem implies that for a network partition system, the choice of consistency and

availability are mutually exclusive. CA means consistency andavailability, AP means availability

and partition tolerance and CP means consistency and partition tolerance. Figure 3.1 shows the CAP theorem usage in Big Data Solutions.

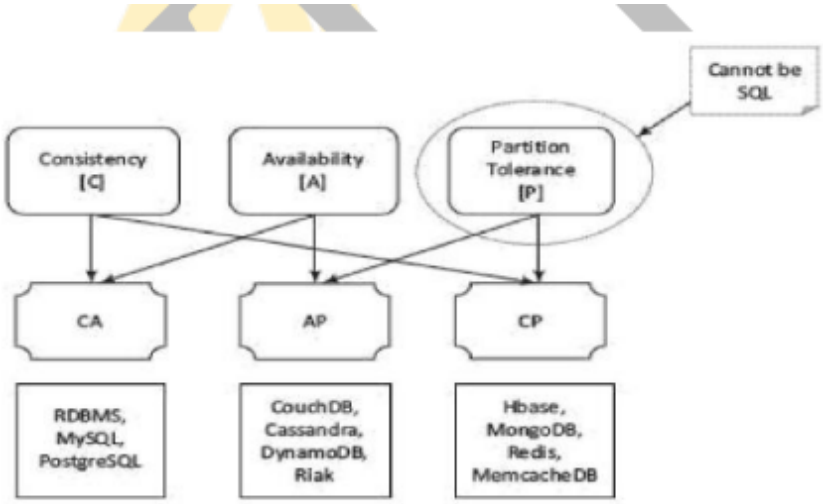


Figure 3.1 CAP theorem in Big Data solutions

6.

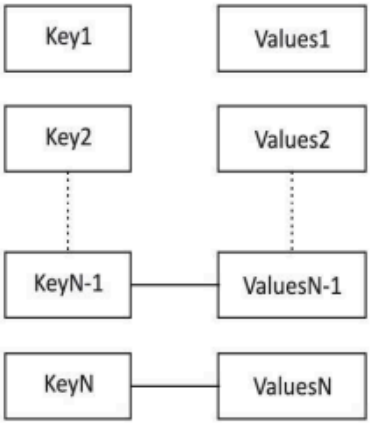
OR

6 a. Demonstrate the working of key- value store with an example. **(10 Marks)**
b. Describe the features of MongoDB, and its industrial application. **(10 Marks)**

6.a

Key-Value Store

The simplest way to implement a schema-less data store is to use key-value pairs.The data store characteristics are high performance, scalability and flexibility. Data retrieval is fast in key-value pairs data store. A simple string called, key maps to a large data string or BLOB (Basic Large Object). Key-value store accesses use a primary key for accessing the values. Therefore, the store can be easily scaled up for very large data. The concept is similar to a hash table where a unique key points to a particular item(s) of data. Figure 3.4 shows key-value pairs architectural pattern and example of students' database as key-value pairs



Key	Value
"Ashish"	"Category: Student; Class: B.Tech.; Semester: VII; Branch: Engineering; Mobile:9999912345"
"Mayuri"	"Category: student; class: M.Tech.; Mobile:8888823456"

Advantages of a key-value store are as follows:

1. Data Store can store any data type in a value field. The key-value system stores the information as a BLOB of data (such as text, hypertext, images,video and audio) and return the same BLOB when the data is retrieved. Storage is like an English dictionary. Query for a word retrieves the meanings, usages, different forms as a single item in the dictionary. Similarly, querying for key retrieves the values.
2. A query just requests the values and returns the values as a single item. Values can be of any data type.
3. Key-value store is eventually consistent.
4. Key-value data store may be hierarchical or may be ordered key-value store.
5. Returned values on queries can be used to convert into lists, table- columns, data-frame fields and columns.
6. Have (i) scalability, (ii) reliability, (iii) portability and (iv) low operationalcost.
7. The key can be synthetic or auto-generated. The key is flexible and can be represented in many formats: (i) Artificially generated strings created from a hash of a value, (ii) Logical path names to images or files, (iii) RESTweb-service calls (request response cycles), and (iv) SQL queries.

Limitations of key-value store architectural pattern are:

1. No indexes are maintained on values, thus a subset of values is not searchable.
2. Key-value stores does not provide traditional database capabilities, such as atomicity of

transactions,or consistency when multiple transactions are executed simultaneously. The application needs to
implement such capabilities.
3. Maintaining unique values as keys may become more difficult when the volume of data increases.One cannot retrieve a single result when a key- value pair is not uniquely identified.
4. Queries cannot be performed on individual values. No clause like 'where' in a relational database that filters a result set.

6.b

Features of Mango D B
MongoDB data store is a physical container for collections. Each DB gets its own set of files on the file system. A number of DBs can run on a single MongoDB server. DB is default DB in MongoDB that stores within a data folder. The database server ofMongoDB is mongod and the client is mongo.
2. Collection stores a number of MongoDB documents. It is analogous to a table of RDBMS. A collection exists within a single DB to achieve a singlepurpose. Collections may store

documents that do not have the same fields. Thus, documents of the collection are schema-less. Thus, it is possible to store documents of varying structuresin a collection. Practically,

in an RDBMS, it is required to define a column and its data type, but does not need them while working with the MongoDB.
3. Document model is well defined. Structure of document is clear, Document is the unit of storing data in a MongoDB database. Documents are analogousto the records of RDBMS table. Insert, update and delete operations can beperformed on a collection. Document use

]SON OavaScript Object Notation) approach for storing data.]SON is a lightweight, self-describing format used to interchange data between various applications. JSON data basically

has key-value pairs. Documents have dynamic schema.

4. MongoDB is a document data store in which one collection holds differentdocuments. Data store in the form of]SON-style documents. Number of fields, content and size of the document can differ from one document to another.

5. Storing of data is flexible, and data store consists of JSON-like documents. This implies that the fields can vary from document to document and data structure can be changed over time;]SON has a standard structure, and scalable way of describing hierarchical data (Example 3.3(ii)).

6. Storing of documents on disk is in BSON serialization format. BSON is a binary representation of JSON documents. The mongo JavaScript shell and MongoDB language drivers perform translation between BSON and language-specific document representation.

7. Querying, indexing, and real time aggregation allows accessing and analyzing the data efficiently**.**

Module-4

7 a. Explain the process in MapReduce when client submitting a job, with a neat diagram. **(10 Marks)**

b. Explain Hive Integration and workflow steps involved with a diagram. **(10 Marks)**

7.a

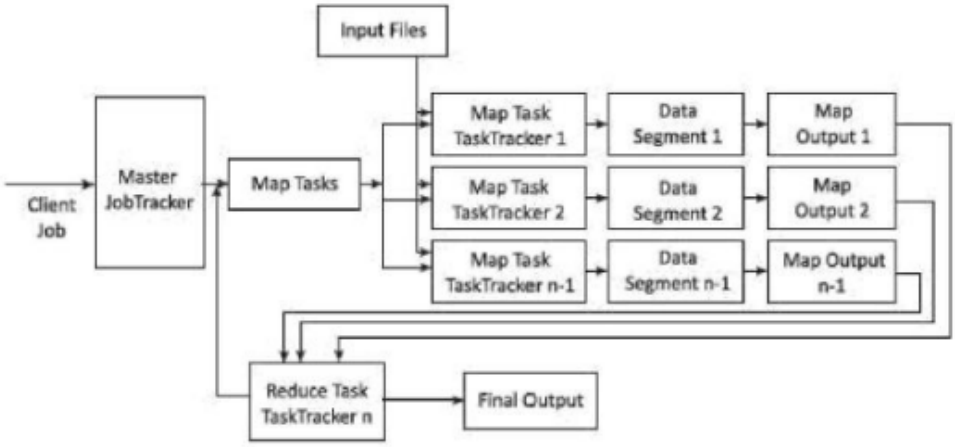


Figure 4.3 MapReduce process on client submitting a job

Figure 4.3 shows MapReduce process when a client submits a job, and the succeeding actions by the JobTracker andTaskTracker. JobTracker and Task Tracker MapReduce consists of a single master JobTracker and one slave TaskTracker per cluster node. The master is responsible for scheduling the component tasks in a job onto the slaves, monitoring them and re-executing the failed tasks. The slaves execute the tasks as directed by the master.

The data for a MapReduce task is initially at input files. The input files typically

reside in the HDFS. The files may be line-based log files, binary format file, multi-line input records, or something else entirely different.

The MapReduce framework operates entirely on key, value-pairs. The framework views the input to the task as a set of (key, value)pairs and produces a set of (key, value) pairs as the output of the task, possiblyof different types.

Map-Tasks

Map task means a task that implements a map(), which runs user application codes for each key-value pair (kl, vl). Key kl is a set of keys. Key kl maps to group of data values (Section 3.3.1). Values vl are a large string which is read from the input file(s).

The output of map() would be zero (when no values are found) or intermediate key-value pairs (k2, v2). The value v2 is the information for the transformation

operation at the reduce task using aggregation or other reducing functions.

Reduce task refers to a task which takes the output v2 from the map as an input and combines those data pieces into a smaller set of data using a combiner. The reduce task is always performed after the map task.

The Mapper performs a function on individual values in a dataset irrespective of the data size of the input. That means that the Mapper works on a single data set.

Figure 4.4 shows logical view of functioning of map().

7.b

Hive Integration and Workflow Steps

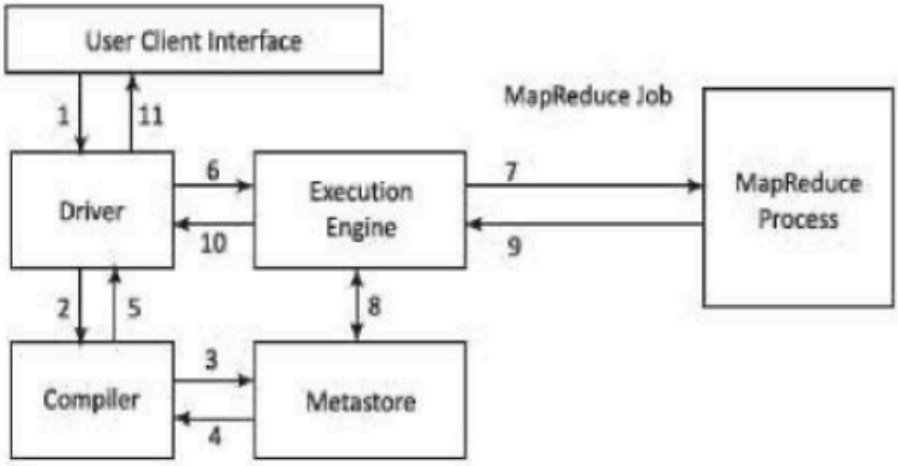


Figure 4.11 Dataflow sequences and workflow steps

The workflow steps are as follows :

Execute Query: Hive interface (CLI or Web Interface) sends a query to DatabaseDriver to execute the query.

Get Plan: Driver sends the query to query compiler that parses the query to

Get Metadata: Compiler sends metadata request to Metastore (of any database, such as MySQL).

Send Metadata: Metastore sends metadata as a response to compiler.

Send Plan: Compiler checks the requirement and resends the plan to driver. The parsing and compiling of the query is complete at this place.

Execute Plan: Driver sends the execute plan to execution engine.

Execute Job: Internally, the process of execution job is a MapReduce job. The execution engine sends the job to JobTracker, which is in Name node and it assigns this job to TaskTracker, which is in Data node. Then , the query executes the job.

Metadata Operations: Meanwhile the execution engine can execute the metadata operations with Metastore.

Fetch Result: Execution engine receives the results from Data nodes.

Send Results: Execution engine sends the result to Driver.

Send Results: Driver sends the results to Hive Interfaces.

OR

8 a. Using HiveQL for the following : **(10 Marks)**
   i) Create a table with partition
   ii) Add, rename and drop a partition to a table.
   b. What is PIG in BigData? Explain the feature of PIG. **(10 Marks)**

8.a CREATE TABLE employees (

```
  emp_id     INT,
  name       STRING,
  salary     FLOAT
```

)

PARTITIONED BY (department STRING)

STORED AS PARQUET; -- or TEXTFILE, ORC, etc.

8a.add

ALTER TABLE employees ADD PARTITION (department='Sales');

Drop,rename

-- Step 1: Add new partition

ALTER TABLE employees ADD PARTITION (department='Marketing') LOCATION '/user/hive/warehouse/employees/hr/';

-- Step 2: Drop old partition
ALTER TABLE employees DROP PARTITION (department='HR');

ALTER TABLE employees DROP PARTITION (department='HR');

8.b

1. It is an abstract over MapReduce
2. It is an execution framework for parallel processing
3. Reduces the complexities of writing a MapReduce program
4. Is a high-level dataflow language. Dataflow language means that a Pig
5. operation node takes the inputs and generates the output for the next node
6. Is mostly used in HDFS environment
7. Performs data manipulation operations at files at data nodes in Hadoop.

Apache Pig is a high-level platform for processing large data sets using Hadoop. It uses a language called **Pig Latin**. Here are some of its key features:

🐷 Top Features of Apache Pig

1. High-Level Language (Pig Latin)

- Easier to write than raw MapReduce.

- Similar to SQL, but designed for **data flow** programming.

2. Abstraction Over MapReduce

- Pig scripts are internally converted into **MapReduce jobs**, so you don't need to write low-level

Java code.

3. Handles Structured, Semi-Structured, and Unstructured Data

- Works well with **logs**, **JSON**, **text files**, **CSV**, etc.

4. Extensible

- You can write custom **User Defined Functions (UDFs)** in Java, Python, or other languages.

5. Optimization Opportunities

- Pig automatically optimizes the execution of the script, just like a DB query planner.

6. Multi-language UDF Support

- You can write UDFs in **Java**, **Python**, **JavaScript**, **Ruby**, and **Groovy**.

7. Interactive and Batch Modes

- Supports both **script execution (batch)** and **interactive Grunt shell**.

8. Schema-less

- You don't have to define schema upfront, which adds flexibility.

9. Easy to Learn and Use

- Less verbose and complex than native MapReduce.

10. Integration with Hadoop Ecosystem

- Works seamlessly with **HDFS**, **HBase**, **Hive**, and other Hadoop components.

—

-

Module-5

9 a. Explain linear and non-linear relationship with essential graphs in machine learning. **(10 Marks)**

b. Write the block diagram of text mining process and explain its phases. **(10 Marks)**

9.a

A linear relationship exists between two variables, say x and y, when a straight line (y = a0 + a1.x) can fit on a graph, with at least some reasonable degree of accuracy. The a1 is the linearity coefficient. For example, a scatter chart can suggest a linear relationship, which means a straight line. Figure 6.1 shows a scatter plot, which fits a linear relationship between the number of students opting for computer courses in years between 2000 and 2017.

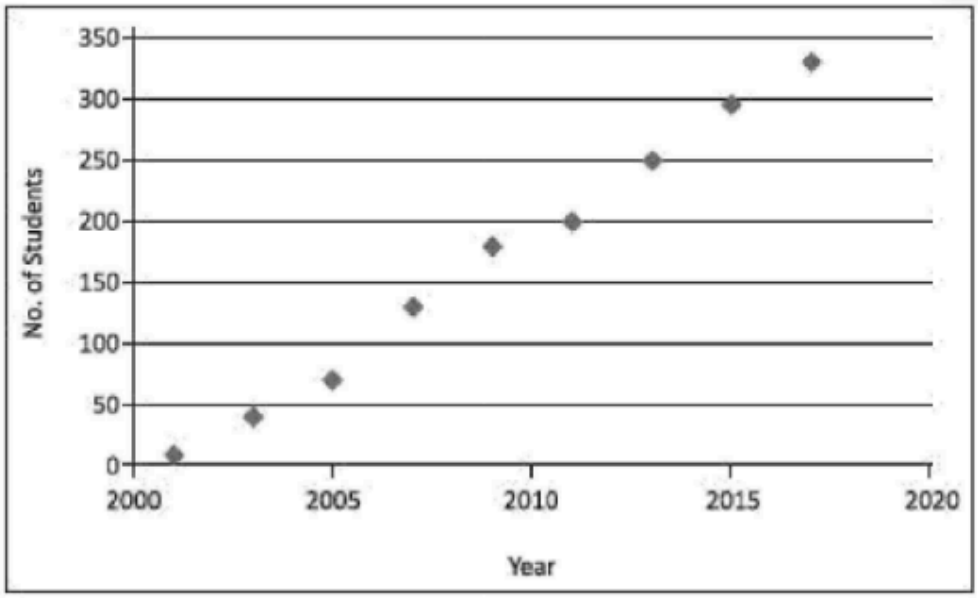


Figure 5.1 Scatter plot for linear relationship between students opting for computer courses in years between 2000 and 2017

A linear relationship can be positive or negative. A positive relationship implies if one variable increases in value, the other also increases in value. A negative relationship, on the

other hand, implies when one increases in value, the other decreases in value. Perfect, strong

or weak linearship categories depend upon the bonding between the two variables.

A non-linear relationship is said to exist between two quantitative variables when a curve (y

= a0 + a1.x + a2.x2 + •••) can be used to fit the data points. The fit should be with at least

some reasonable degree of accuracy for the fitted parameters, a0, a1, a2 ••• Expression for y

then generally predicts the values of one quantitative variable from the values of the other

quantitative variable with considerably more accuracy than a straight line.

Consider an example of non-linear relationship: The side of a square and its area are not linear. In fact, they have quadratic relationship. If the side of a square doubles, then its area increases four times. The relationship predicts the area from the side.

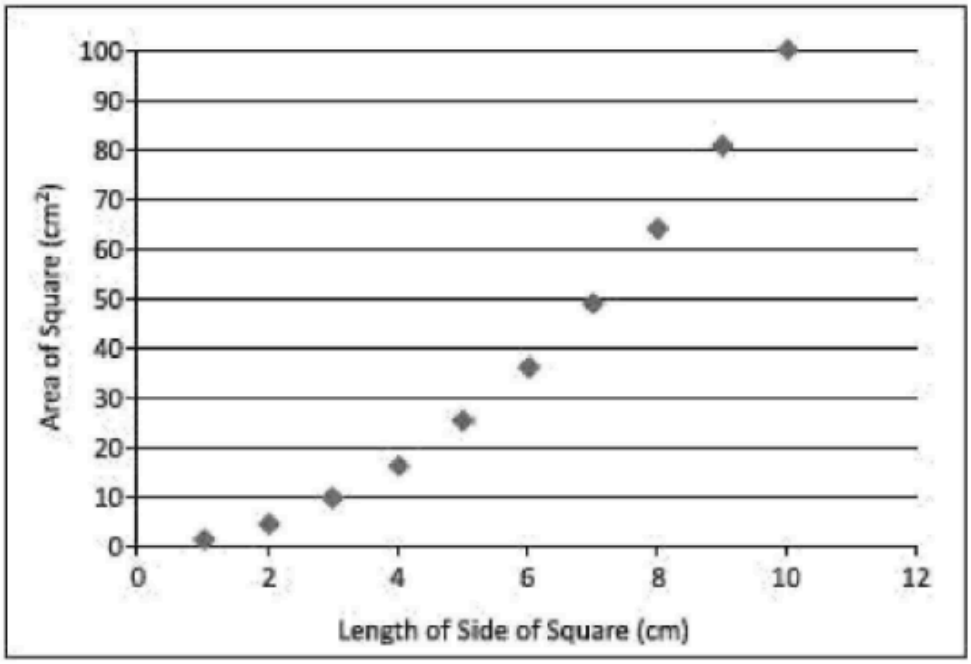
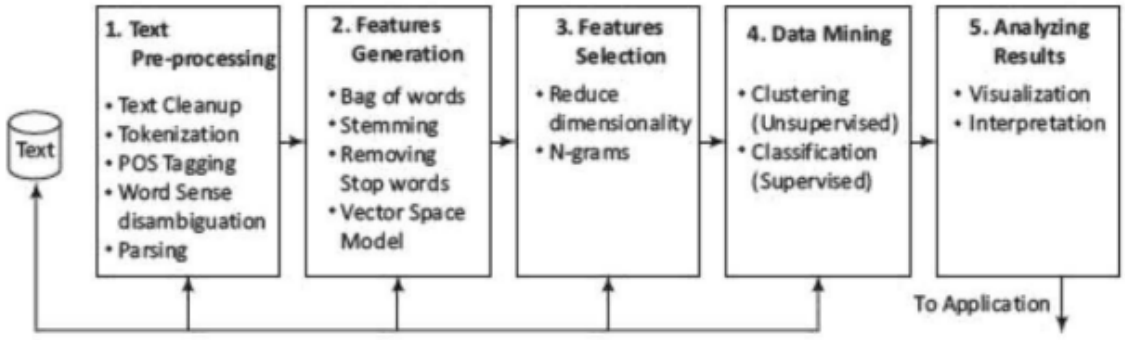


Figure 5.2 scatter plot in case of a non-linear relationship between side of square and its area

9.b



Text Mining is a rapidly evolving area of research. As the amount of social media and other text data grows, there is need for efficient abstraction and categorization of meaningful information from the text.

The five phases for processing text are asfollows:

Phase 1: Text pre-processing enables Syntactic/Semantic text-analysis and does the

followings:

1. Text cleanup is a process of removing unnecessary or unwanted information. Text cleanup converts the raw data by filling up the missing values, identifies and removes outliers, and resolves the inconsistencies. For example, removing comments, removing or escaping "%20" from URL for the web pages or cleanup the typing error, such as teh (the), do n't (do not) [%20 specifies space in a URL].

2. Tokenization is a process of splitting the cleanup text into tokens (words) using white spaces and punctuation marks as delimiters.

3. Part of Speech (POS) tagging is a method that attempts labeling of each token (word) with an appropriate POS. Tagging helps in recognizing names of people, places, organizations and titles. The English language set includes the noun, verb, adverb, adjective, prepositions and conjunctions. Part of Speech encoded in the annotation system of the Penn

Treebank Project has 36 POS tags.4

4. Word sense disambiguation is a method, which identifies the sense of a word used in

a sentence; that gives meaning in case the word has multiple meanings. The

methods, which resolve the ambiguity of words can be context or proximity based.

Some examples of such words are bear, bank, cell and bass.

5. Parsing is a method, which generates a parse-tree for each sentence. Parsing

attempts and infers the precise grammatical relationships between different words in

a given sentence.

Phase 2: Features Generation is a process which first defines features (variables,

predictors). Some of the ways of feature generations are:

1. Bag of words-Order of words is not that important for certain applications.

Text document is represented by the words it contains (and their occurrences).

Document classification methods commonly use the bag-of-words model. The pre-processing of a document first provides a document with a bag of words. Document

classification methods then use the occurrence (frequency) of each word as a feature for training a classifier. Algorithms do not directly apply on the bag of words, but use the frequencies.

2. Stemming-identifies a word by its root.

(i) Normalizes or unifies variations of the same concept, such as speak for three variations, i.e., speaking, speaks, speakers denoted by [speaking, speaks, speaker- + speak]

(ii)Removes plurals, normalizes verb tenses and remove affixes.

Stemming reduces the word to its most basic element. For example, impurification -+ pure.

3. Removing stop words from the feature space-they are the common words, unlikely to help text mining. The search program tries to ignore stop words. For example, ignores a, at, for, it, in and are.

4. Vector Space Model (VSM)-is an algebraic model for representing text documents as vectors of identifiers, word frequencies or terms in the document index. VSM uses the method of term frequency-inverse document frequency (TF-IDF) and evaluates how important a word is in a document. When used in document classification, VSM also refers to the bag-of-words model. This bag of words is required to be converted into a term-vector in VSM. The term vector provides the numeric values corresponding to each term appearing in a document. The term vector is very helpful in feature generation and selection.Term frequency and inverse document frequency (IDF) are important metrics in text analysis.TF-IDF weighting is most common- Instead of the simple TF, IDF is used to weight the importance of word in the document.Phase 3: Features Selection is the process that selects a subset of features by rejecting irrelevant and/or redundant features (variables, predictors or dimension) according to defined

criteria. Feature selection process does the following:

1. Dimensionality reduction-Feature selection is one of the methods of division and therefore, dimension reduction. The basic objective is to eliminate irrelevant and redundant data.

Redundant features are those, which provide no extra information. Irrelevant features provide no useful or relevant information in any context.

Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are dimension reduction methods. Discrimination ability of a feature measures relevancy of features. Correlation helps in finding the redundancy of the feature. Two features are redundant to each other if their values correlate with each other.

2. N-gram evaluation-finding the number of consecutive words of interest and extract them. For example, 2-gram is a two words sequence, ["tasty food", "Good one"]. 3-gram is a three words sequence, ["Crime Investigation Department"].

3. Noise detection and evaluation of outliers methods do the identification of unusual or suspicious items, events or observations from the data set. This step helps in cleaning the data.

The feature selection algorithm reduces dimensionality that not only improves the performance of learning algorithm but also reduces the storage requirement for a dataset. The process enhances data understanding and its visualization.

Phase 4: Data mining techniques enable insights about the structured database that resulted from the previous phases. Examples of techniques are:

1. Unsupervised learning (for example, clustering)

(i) The class labels (categories) of training data are unknown

(ii)Establish the existence of groups or clusters in the data

Good clustering methods use high intra-cluster similarity and low inter-cluster similarity. Examples of uses - biogs, pattern

and trends.

2. Supervised learning (for example, classification)

(i) The training data is labeled indicating the class

(ii)New data is classified based on the training set

Classification is correct when the known label of test sample is identical with the resulting

class computed from the classification model.Examples of uses are news filtering application, where it is required to automatically assign incoming documents to pre-defined categories; email spam filtering, where it is identified whether incoming email messages are spam or not.Example of text classification methods are Naive Bayes Classifier and SVMs.

3. Identifying evolutionary patterns in temporal text streams-the method is useful in a wide range of applications, such as summarizing of events in news articles and extracting the research trends in the scientific literature.

Phase 5: Analysing results

(i) Evaluate the outcome of the complete process.

(ii) Interpretation of Result- If acceptable then results obtained can be used as an input for next

set of sequences. Else, the result can be discarded, and try to understand what and why the

The process failed.

(iii) Visualization - Prepare visuals from data, and build a prototype.

(iv)Use the results for further improvement in activities at the enterprise, industry or institution.

OR

10 a. With a neat diagram, write the steps in K-means clustering. (10 Marks)

b. Explain the purpose of web usage analytics and the significance of web graphs. (10 Marks)

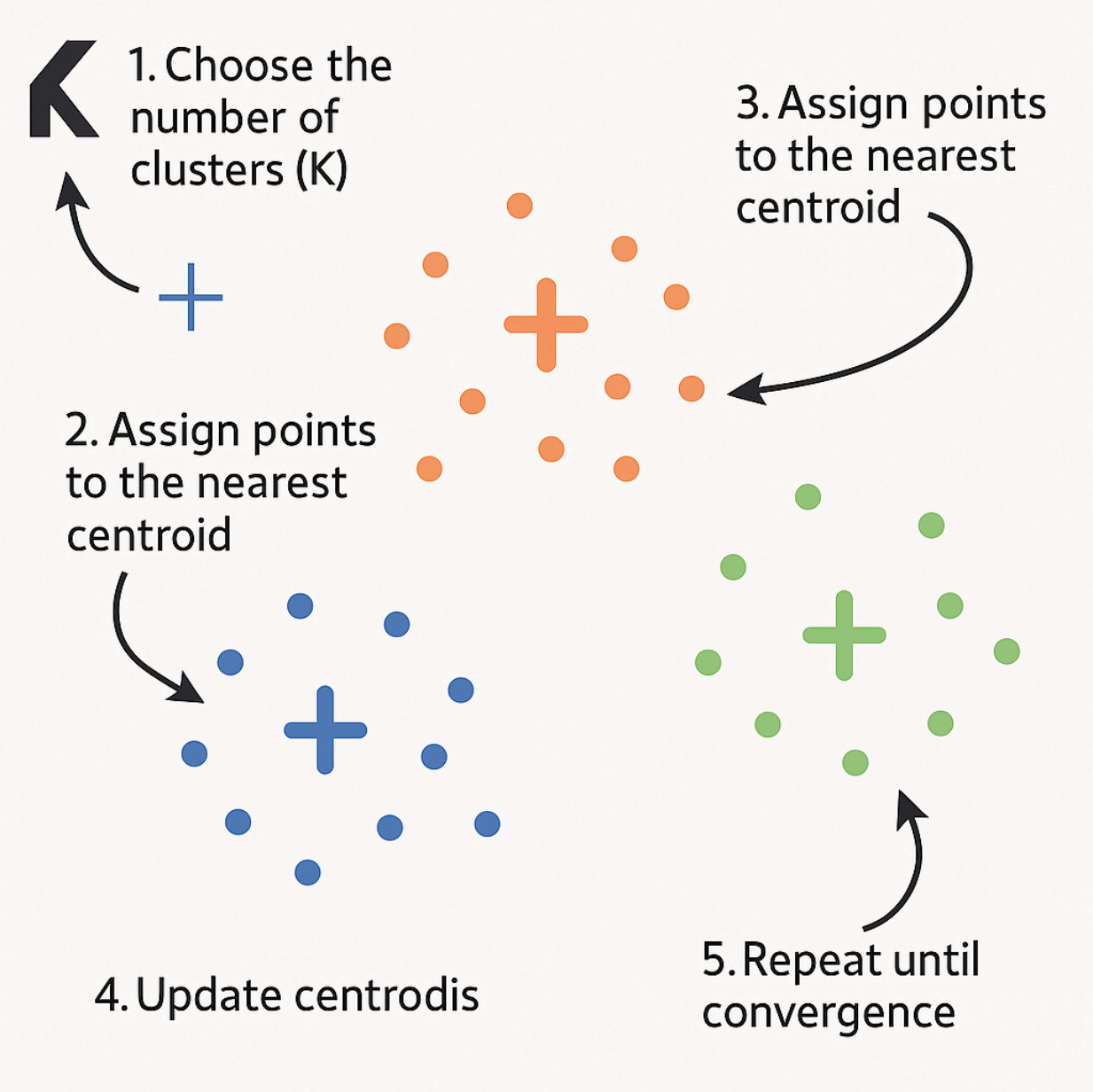
* * * * *

K-Means is an **unsupervised machine learning algorithm** used to group data into **K clusters** based on feature similarity.

Steps in K-Means Clustering

1. **Choose the number of clusters (K)**

2. **Initialize** K random centroids (cluster centers)
3. **Assign** each data point to the **nearest centroid**
4. **Update** centroids by calculating the **mean** of all points in a cluster
5. **Repeat** steps 3 & 4 until centroids **don't change (converge)** or max iterations reached



10.b

Purpose of Web Usage Analytics

Web Usage Analytics is all about analyzing **user behavior** on websites. Its main purposes include:

1. Understanding User Behavior

- Track which pages users visit, how long they stay, and what they click on.

- Helps in identifying popular content and user interests.

2. Improving User Experience

- By analyzing navigation paths, you can optimize site structure, menus, and layouts.
- Reduce bounce rates and improve site usability.

3. Personalization

- Recommend content or products based on past user interactions.
- E.g., Amazon shows “recommended for you” based on usage patterns.

4. Business Decision Making

- Identify conversion bottlenecks, checkout drop-offs, or top-performing landing pages.
- Guide marketing and sales strategies.

5. Anomaly and Fraud Detection

- Detect suspicious behavior, such as bots or fraud attempts, through unusual access patterns.

Significance of Web Graphs

A **Web Graph** represents the **structure of the web**, where:

- **Nodes (vertices)** = Web pages
- **Edges (links)** = Hyperlinks between pages

Why Web Graphs Matter:

1. Link Analysis & PageRank

- Google's **PageRank** algorithm uses the web graph to rank search results based on link structure.

2. Crawling & Indexing

- Search engines use web graphs to prioritize which pages to crawl next based on how they're

linked.

3. Community Detection

- Helps identify clusters of related pages (e.g., news sites, tech blogs).

4. Detecting Dead Links

- Easily spot pages that are no longer reachable or isolated.

5. Enhancing Recommendations

- Use graph-based algorithms to recommend related articles or products based on link proximity.
