BDA-UNIT 1&2(16 MARKS)

UNIT-1

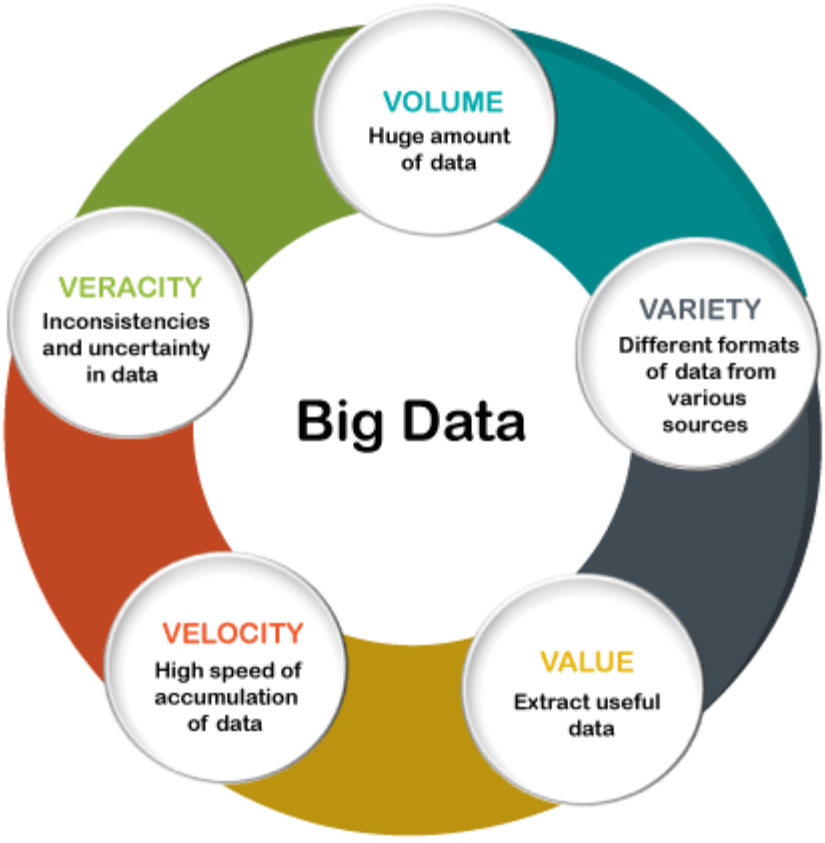
1.Explain Five “V”’s of Big Data.

Big Data contains a large amount of data that is not being processed by traditional data storage or the processing unit. It is used by many **multinational companies** to **process** the data and business of many **organizations**. The data flow would exceed **150 exabytes** per day before replication.

There are five v's of Big Data that explains the characteristics.

5 V's of Big Data

- **Volume**
- **Veracity**
- **Variety**
- **Value**
- **Velocity**

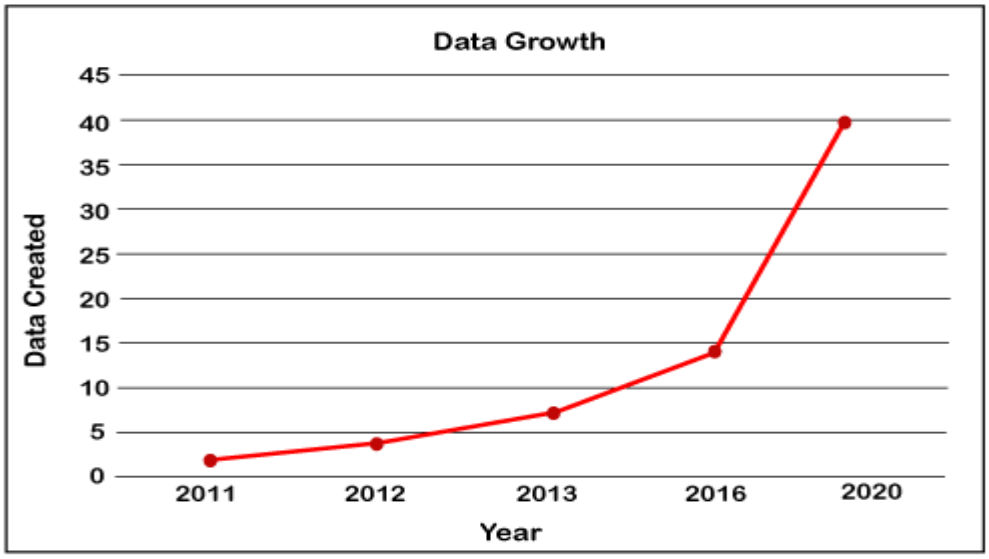


- **Volume**

The name Big Data itself is related to an enormous size. Big Data is a vast 'volumes' of data generated from many sources daily, such as **business processes, machines, social media platforms, networks, human interactions,** and many more.

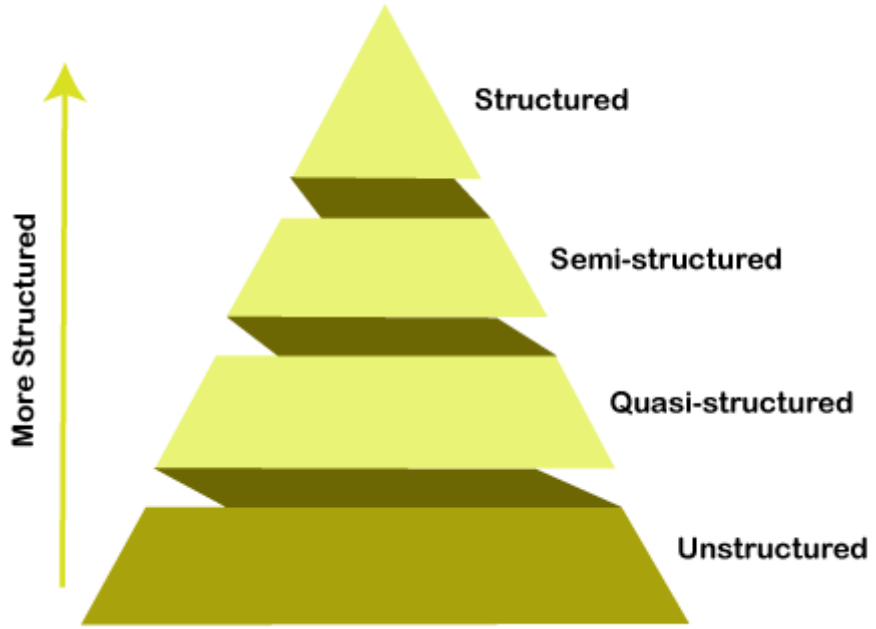
Facebook can generate approximately a **billion** messages, **4.5 billion** times that the "**Like**" button is recorded, and more than **350 million** new posts are uploaded each day. Big data technologies can handle large amounts of data.

Advertisement



- **Variety**

Big Data can be **structured, unstructured, and semi-structured** that are being collected from different sources. Data will only be collected from **databases** and **sheets** in the past, But these days the data will comes in array forms, that are **PDFs, Emails, audios, SM posts, photos, videos,** etc.



The data is categorized as below:

1. **Structured data:** In Structured schema, along with all the required columns. It is in a tabular form. Structured Data is stored in the relational database management system.
2. **Semi-structured:** In Semi-structured, the schema is not appropriately defined, e.g., **JSON, XML, CSV, TSV**, and **email**. OLTP (**Online Transaction Processing**) systems are built to work with semi-structured data. It is stored in relations, i.e., **tables**.
3. **Unstructured Data**: All the **unstructured files, log files, audio files**, and **image** files are included in the unstructured data. Some organizations have much data available, but they did not know how to **derive** the value of data since the data is raw.
4. **Quasi-structured Data:**The data format contains textual data with inconsistent data formats that are formatted with effort and time with some tools.

Example: Web server logs, i.e., the log file is created and maintained by some server that contains a list of **activities**.

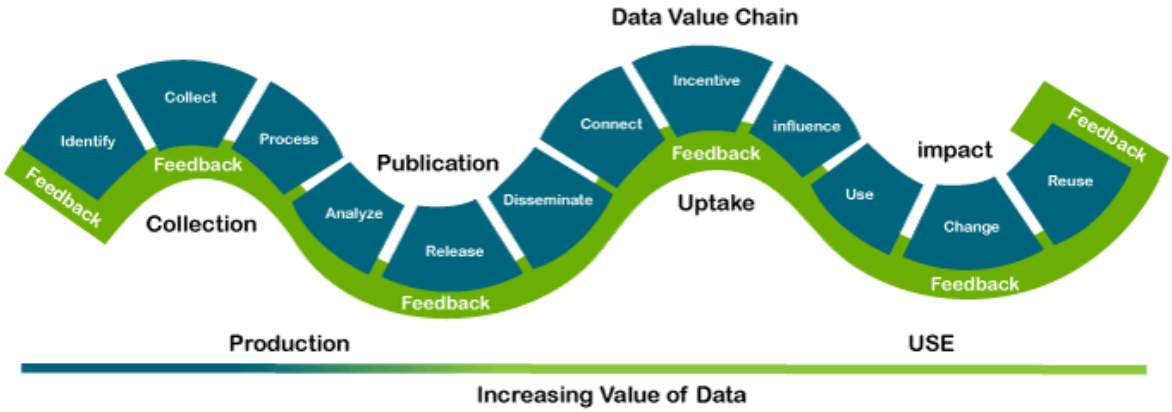
- **Veracity**

Veracity means how much the data is reliable. It has many ways to filter or translate the data. Veracity is the process of being able to handle and manage data efficiently. Big Data is also essential in business development.

For example, **Facebook posts** with hashtags.

- **Value**

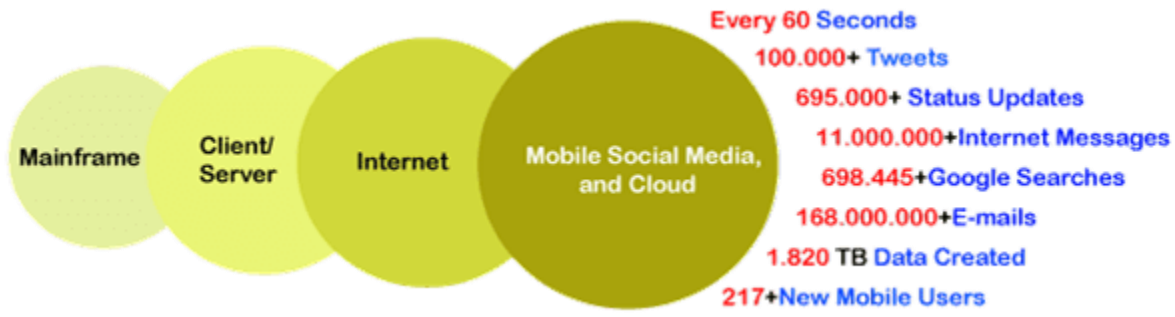
Value is an essential characteristic of big data. It is not the data that we process or store. It is **valuable** and **reliable** data that we **store, process**, and also **analyze**.



- **Velocity**

Velocity plays an important role compared to others. Velocity creates the speed by which the data is created in **real-time**. It contains the linking of incoming **data sets speeds, rate of change**, and **activity bursts**. The primary aspect of Big Data is to provide demanding data rapidly.

Big data velocity deals with the speed at the data flows from sources like **application logs, business processes, networks, and social media sites, sensors, mobile devices,** etc.



2.Explain Firewalls and its Types.

A firewall is a network security device or software that monitors and controls incoming and outgoing network traffic based on predetermined security rules. Firewalls are essential for protecting networks and systems from unauthorized access, cyber threats, and data breaches by acting as a barrier between trusted internal networks and untrusted external networks, such as the internet.

Types of Firewalls

Firewalls are categorized based on their functions, deployment methods, and the level of network security they provide. Here are some of the major types of firewalls:

- **Packet-Filtering Firewall**
- **Stateful Inspection Firewall**
- **Proxy Firewall (Application-Level Gateway)**
- **Next-Generation Firewall (NGFW)**
- **Unified Threat Management (UTM) Firewall**
- **Cloud-Based Firewall (Firewall as a Service - FWaaS)**
- **Network Address Translation (NAT) Firewall**

1. Packet-Filtering Firewall

- Function: This type inspects data packets, looking at the source and destination IP addresses, port numbers, and protocols to determine whether the packet should be allowed or denied based on predefined rules.
- Advantages: Simple and efficient, suitable for basic filtering at the network level.
- Limitations: It does not inspect the actual data content, so it can’t detect complex threats like viruses or malware.

2. Stateful Inspection Firewall

- Function: Also called dynamic packet-filtering firewalls, these track the state of active connections and make decisions based on both the packet's properties and the context of the connection.
- Advantages: More secure than packet-filtering firewalls because they consider the entire session, which reduces the risk of spoofed packets.
- Limitations: More complex and requires more processing power, which may slow down network traffic.

3. Proxy Firewall (Application-Level Gateway)

- Function: This firewall acts as an intermediary between users and the internet, inspecting the data at the application level (like HTTP for web traffic or FTP for file transfers) rather than just at the packet level.
- Advantages: Can provide deep inspection of packets and filter traffic based on application-layer data, offering more control over specific applications.
- Limitations: Slower than other types because it requires more processing to inspect each packet at the application layer.

4. Next-Generation Firewall (NGFW)

- Function: Combines traditional firewall functions with advanced features like deep packet inspection, intrusion prevention systems (IPS), application awareness, and user identity management.
- Advantages: Provides high-level security by identifying and blocking sophisticated attacks, malware, and application-layer threats.
- Limitations: More complex and expensive than traditional firewalls, and requires skilled personnel to manage.

5. Unified Threat Management (UTM) Firewall

- Function: A comprehensive security solution that combines multiple security services (firewall, antivirus, content filtering, intrusion detection, etc.) in a single device.
- Advantages: Simplifies management by integrating various security features, making it suitable for small and medium-sized businesses.
- Limitations: May not offer as much customization and flexibility as specialized, dedicated solutions.

6. Cloud-Based Firewall (Firewall as a Service - FWaaS)

- Function: A cloud-hosted firewall that provides network traffic filtering from the cloud, rather than through on-premises hardware.
- Advantages: Scalable and easy to deploy across multiple locations; ideal for businesses using cloud infrastructure.
- Limitations: Depends on internet connectivity and might have latency issues; also raises privacy concerns as traffic is processed off-site.

7. Network Address Translation (NAT) Firewall

- Function: Operates by modifying IP addresses in packets to mask internal IP addresses, creating a secure internal network structure.
- Advantages: Adds a layer of security by hiding internal IP addresses from external networks.
- Limitations: Not a standalone firewall type and usually works in conjunction with other firewalls; limited in handling more complex security threats.

By combining different types of firewalls, organizations can implement a layered defense strategy, often called "defense in depth", to create a more comprehensive and resilient security posture.

3.Explain Big Data Application and Technologies.

Big data refers to large volumes of data, both structured and unstructured, that traditional databases can't handle efficiently. Big data applications and technologies are tools and methods used to process, store, analyze, and extract insights from this massive amount of data. Here's a breakdown of key applications and technologies:

Big Data Applications

1. **Retail:** Retailers analyze customer data to predict buying behaviors, optimize pricing, and personalize marketing. For example, by analyzing purchase history and browsing patterns, stores can recommend products likely to interest customers, increasing sales.
2. **Healthcare**: Big data helps in medical research, diagnostics, and treatment. By analyzing vast amounts of patient data, doctors can predict disease outbreaks, personalize treatments, and discover insights that lead to better care.
3. **Finance:** In finance, big data is used for fraud detection, risk management, and algorithmic trading. By processing transaction data in real time, banks can identify suspicious activities, calculate risk, and make fast trading decisions.
4. **Manufacturing:** Manufacturers use big data to monitor production lines, predict machine failures, and optimize supply chains. This leads to better efficiency, lower costs, and minimized downtime.

5. **Smart Cities**: Urban planners use big data from traffic, energy, and other city services to improve infrastructure and reduce resource use. This makes cities more efficient and sustainable.

Key Big Data Technologies

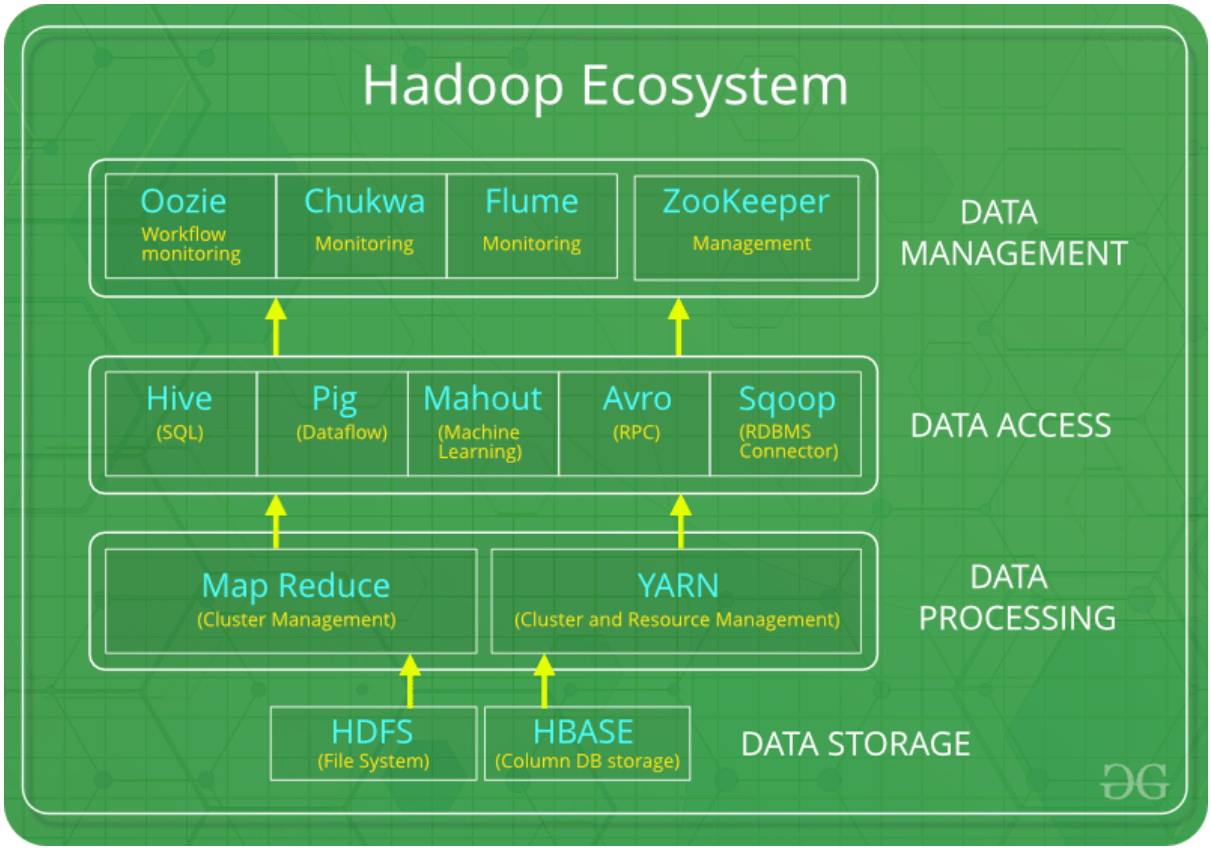
1. **Data Storage Technologies**
    - **Hadoop**: An open-source framework for storing and processing big data. It uses distributed storage, which means it can handle massive datasets by distributing them across multiple machines.
    - **NoSQL Databases (like MongoDB, Cassandra):** Unlike traditional databases, NoSQL databases can handle unstructured data and scale easily, making them suitable for big data applications.
2. **Data Processing Technologies**
    - **Apache Spark:** A powerful open-source processing engine for big data. Spark can handle batch processing and real-time streaming data, making it faster and more flexible than Hadoop's MapReduce.
    - **Apache Flink:** Designed for stream processing, Flink can handle high-throughput data flows, which is ideal for applications needing real-time data analysis.
3. **Data Integration Tools**
    - **Apache Kafka:** A distributed event streaming platform, Kafka is used to capture and manage real-time data feeds from various sources, which are then processed by big data tools.
    - **Talend:** A data integration tool that helps combine data from different sources for unified analysis.
4. **Data Analysis and Visualization Tools**
    - **Tableau:** A popular visualization tool, Tableau lets users create interactive dashboards and visuals to explore large datasets.
    - **Power BI:** Microsoft's data visualization and business intelligence tool helps in creating reports and visualizations, offering data insights for better decision-making.
    - **Machine Learning Libraries (like TensorFlow, Scikit-Learn):** These libraries allow data scientists to build predictive models and uncover patterns within big data.
5. **Data Governance and Security Technologies**
    - **Apache Ranger**: Provides centralized security and fine-grained access control for big data platforms.
    - **Data Catalogs (like Apache Atlas):** These help organize and document big data, making it easier to find and use for analysis.

In sum, big data technologies offer tools for each stage of data handling, from storage and processing to analysis and security, enabling businesses to extract value and make data-driven decisions.

4.Explain Hadoop ECO System in Detail.

Hadoop Ecosystem is a platform or a suite which provides various services to solve the big data problems. It includes Apache projects and various commercial tools and solutions. There are four major elements of Hadoop i.e. HDFS, MapReduce, YARN, and Hadoop Common Utilities. Most of the tools or solutions are used to supplement or support these major elements. All these tools work collectively to provide services such as absorption, analysis, storage and maintenance of data etc. Following are the components that collectively form a Hadoop ecosystem:

- **HDFS:** Hadoop Distributed File System
- **YARN:** Yet Another Resource Negotiator
- **MapReduce:** Programming based Data Processing
- **Spark:** In-Memory data processing
- **PIG, HIVE:** Query based processing of data services
- **HBase:** NoSQL Database
- **Mahout, Spark MLLib:** Machine Learning algorithm libraries
- **Solar, Lucene:** Searching and Indexing
- **Zookeeper:** Managing cluster
- **Oozie:** Job Scheduling



Apart from the above-mentioned components, there are many other components too that are part of the Hadoop ecosystem.
All these toolkits or components revolve around one term i.e. Data. That's the beauty of Hadoop that it revolves around data and hence making its synthesis easier.

HDFS:

- HDFS is the primary or major component of Hadoop ecosystem and is responsible for storing large data sets of structured or unstructured data across various nodes and thereby maintaining the metadata in the form of log files.
- HDFS consists of two core components i.e.
    1. Name node
    2. Data Node
- Name Node is the prime node which contains metadata (data about data) requiring comparatively fewer resources than the data nodes that stores the actual data. These data nodes are commodity hardware in the distributed environment. Undoubtedly, making Hadoop cost effective.
- HDFS maintains all the coordination between the clusters and hardware, thus working at the heart of the system.

YARN:

- Yet Another Resource Negotiator, as the name implies, YARN is the one who helps to manage the resources across the clusters. In short, it performs scheduling and resource allocation for the Hadoop System.
- Consists of three major components i.e.
    1. Resource Manager
    2. Nodes Manager
    3. Application Manager
- Resource manager has the privilege of allocating resources for the applications in a system whereas Node managers work on the allocation of resources such as CPU, memory, bandwidth per machine and later on acknowledges the resource manager. Application manager works as an interface between the resource manager and node manager and performs negotiations as per the requirement of the two.

MapReduce:

- By making the use of distributed and parallel algorithms, MapReduce makes it possible to carry over the processing's logic and helps to write applications which transform big data sets into a manageable one.
- MapReduce makes the use of two functions i.e. Map() and Reduce() whose task is:
    1. **Map()** performs sorting and filtering of data and thereby organizing them in the form of group. Map generates a key-value pair based result which is later on processed by the Reduce() method.

2. **Reduce(),**as the name suggests does the summarization by aggregating the mapped data. In simple, Reduce() takes the output generated by Map() as input and combines those tuples into smaller set of tuples.

PIG:

Pig was basically developed by Yahoo which works on a pig Latin language, which is Query based language similar to SQL.

- It is a platform for structuring the data flow, processing and analyzing huge data sets.
- Pig does the work of executing commands and in the background, all the activities of MapReduce are taken care of. After the processing, pig stores the result in HDFS.
- Pig Latin language is specially designed for this framework which runs on Pig Runtime. Just the way Java runs on the JVM.
- Pig helps to achieve ease of programming and optimization and hence is a major segment of the Hadoop Ecosystem.

HIVE:

- With the help of SQL methodology and interface, HIVE performs reading and writing of large data sets. However, its query language is called as HQL (Hive Query Language).
- It is highly scalable as it allows real-time processing and batch processing both. Also, all the SQL datatypes are supported by Hive thus, making the query processing easier.
- Similar to the Query Processing frameworks, HIVE too comes with two components: JDBC Drivers and HIVE Command Line.
- JDBC, along with ODBC drivers work on establishing the data storage permissions and connection whereas HIVE Command line helps in the processing of queries.

Mahout:

- Mahout, allows Machine Learnability to a system or application. Machine Learning, as the name suggests helps the system to develop itself based on some patterns, user/environmental interaction or on the basis of algorithms.
- It provides various libraries or functionalities such as collaborative filtering, clustering, and classification which are nothing but concepts of Machine learning. It allows invoking algorithms as per our need with the help of its own libraries.

Apache Spark:

- It’s a platform that handles all the process consumptive tasks like batch processing, interactive or iterative real-time processing, graph conversions, and visualization, etc.
- It consumes in memory resources hence, thus being faster than the prior in terms of optimization.
- Spark is best suited for real-time data whereas Hadoop is best suited for structured data or batch processing, hence both are used in most of the companies interchangeably.

Apache HBase:

- It's a NoSQL database which supports all kinds of data and thus capable of handling anything of Hadoop Database. It provides capabilities of Google's BigTable, thus able to work on Big Data sets effectively.
- At times where we need to search or retrieve the occurrences of something small in a huge database, the request must be processed within a short quick span of time. At such times, HBase comes handy as it gives us a tolerant way of storing limited data

Other Components:

Apart from all of these, there are some other components too that carry out a huge task in order to make Hadoop capable of processing large datasets. They are as follows:

- **Solr, Lucene**: These are the two services that perform the task of searching and indexing with the help of some java libraries, especially Lucene is based on Java which allows spell check mechanism, as well. However, Lucene is driven by Solr.
- **Zookeeper:** There was a huge issue of management of coordination and synchronization among the resources or the components of Hadoop which resulted in inconsistency, often. Zookeeper overcame all the problems by performing synchronization, inter-component based communication, grouping, and maintenance.
- **Oozie:** Oozie simply performs the task of a scheduler, thus scheduling jobs and binding them together as a single unit. There is two kinds of jobs .i.e Oozie workflow and Oozie coordinator jobs. Oozie workflow is the jobs that need to be executed in a sequentially ordered manner whereas Oozie Coordinator jobs are those that are triggered when some data or external stimulus is given to it.

5.Contrast between Structured and Unstructured Data.

The main difference is that structured data is defined and searchable. This includes data like dates, phone numbers, and product SKUs. Unstructured data is everything else, which is more difficult to categorize or search, like photos, videos, podcasts, social media posts, and emails. Most of the data in the world is unstructured data.

What is structured data?

Structured data is typically quantitative data that is organized and easily searchable. The programming language Structured Query Language (SQL) is used in a relational database to "query" to input and search within structured data.

Common types of structured data include names, addresses, credit card numbers, telephone numbers, star ratings from customers, bank information, and other data that can be easily searched using SQL.

Structured data examples.

In the real world, structured data could be used for things like:

- Booking a flight: Flight and reservation data, such as dates, prices, and destinations, fit neatly within the Excel spreadsheet format. When you book a flight, this information is stored in a database.
- Customer relationship management (CRM): CRM software such as Salesforce runs structured data through analytical tools to create new data sets for businesses to analyze customer behavior and preferences.

Pros and cons of structured data

There are numerous benefits – and a handful of drawbacks – to using structured data. To help you get a better idea of whether structured data is right for your own project goals, consider the following advantages and disadvantages:

Pros	Cons
It's easily searchable and used for machine learning algorithms.	It's limited in usage, meaning it can only be used for its intended purpose.
It's accessible to businesses and organizations for interpreting data.	It's limited in storage options because it's stored in systems like data warehouses with rigid schemas.
There are more tools available for analyzing structured data than unstructured.	It requires tabular formats that require rigid schema consisting of predefined fields.

Structured data tools

Structured data is typically stored and used with relational databases and data warehouses supported by SQL. Some examples of tools used to work with structured data include:

- OLAP
- MySQL
- PostgreSQL
- Oracle Database

What is unstructured data?

Unstructured data is every other type of data that is not structured. Approximately 80-90% of data is unstructured, meaning it has huge potential for competitive advantage if companies find ways to leverage it . Unstructured data includes a variety of formats such as emails, images, video files, audio files, social media posts, PDFs, and much more.

Unstructured data is typically stored in data lakes, NoSQL databases, data warehouses, and applications. Today, this information can be processed by artificial intelligence algorithms and delivers huge value for organizations.
Examples of unstructured data

In the real world, unstructured data could be used for things like:

- **Chatbots:** Chatbots are programmed to perform text analysis to answer customer questions and provide the right information.
- **Market predictions:** Data can be maneuvered to predict changes in the stock market so that analysts can adjust their calculations and investment decisions.

Pros and cons of unstructured data

Just as with structured data, there are numerous pros and cons to using unstructured data. Some of the advantages and disadvantages to using unstructured data include:

Pros	Cons
It remains undefined until it's needed, making it adaptable for data professionals to take only what they need for a specific query while storing most data in massive data lakes.	It requires data scientists to have expertise in preparing and analyzing the data, which could restrict other employees in the organization from accessing it.
Within definitions, unstructured data can be collected quickly and easily.	Special tools are needed to deal with unstructured data, further contributing to its lack of accessibility.

Unstructured data tools

Unstructured data is typically supported by flexible NoSQL-friendly data lakes and non-relational databases. As a result, some of the tools you might use to manage unstructured data include:

- MongoDB
- Hadoop
- Azure