**Big Data**

In this digital world, everyone leaves a trace. From our travel habits to our workouts and entertainment, the increasing number of internet connected devices that we interact with on a daily basis record vast amounts of data

about us.

There’s even a name for it: Big Data.

Ernst and Young offers the following definition: “**Big Data refers to the dynamic, large and**

**disparate volumes of data being created by people, tools, and machines. It requires new, innovative, and scalable technology to collect, host, and analytically process the vast amount of data gathered in order to derive real-time business insights that relate to consumers, risk, profit, performance, productivity management, and enhanced shareholder value**.”

**Vs of Big Data**

There is no one definition of Big Data, but there are certain elements that are common

across the different definitions, such as velocity, volume, variety, veracity, and value.

* **Velocity** is the speed at which data accumulates.

Data is being generated extremely fast, in a process that never stops. Near or real-time streaming, local, and cloud-based technologies can process information very quickly.

* **Volume** is the scale of the data, or the increase in the amount of data stored.

Drivers of volume are the increase in data sources, higher resolution sensors, and scalable infrastructure.

* **Variety** is the diversity of the data.

Structured data fits neatly into rows and columns, in relational databases while unstructured data is not organized in a pre-defined way, like Tweets, blog posts, pictures, numbers, and video.

Variety also reflects that data comes from different sources, machines, people, and processes, both internal and external to organizations.

Drivers are mobile technologies, social media, wearable technologies, geo technologies, video, and many, many more.

* **Veracity** is the quality and origin of data, and its conformity to facts and accuracy.

Attributes include consistency, completeness, integrity, and ambiguity.

Drivers include cost and the need for traceability. With the large amount of data available, the debate rages on about the accuracy of data in the digital age. Is the information real, or is it false?

* **Value** is our ability and need to turn data into value. Value isn't just profit. It may have medical or social benefits, as well as customer, employee, or personal satisfaction. The main reason that people invest time to understand Big Data is to derive value from it.

Let's look at some examples of the V's in action.

* Velocity: Every 60 seconds, hours of footage are uploaded to YouTube which is generating data.
* Volume: The world population is approximately seven billion people and the vast majority are now using digital devices; mobile phones, desktop and laptop computers, wearable devices, and so on. These devices all generate, capture, and store data -- approximately 2.5 quintillion bytes every day. That's the equivalent of 10 million Blu-ray DVD's.
* Variety: Let's think about the different types of data; text, pictures, film, sound, health data from wearable devices, and many different types of data from devices connected to the Internet of Things.
* Veracity: 80% of data is considered to be unstructured and we must devise ways to produce reliable and accurate insights. The data must be categorized, analyzed, and visualized.

Data Scientists today derive insights from Big Data and cope with the challenges that

these massive data sets present. The scale of the data being collected means that it’s not feasible to use conventional data analysis tools.

However, alternative tools that leverage distributed computing power can overcome this problem. Tools such as **Apache Spark, Hadoop and its ecosystem** provide ways to extract, load, analyze, and process the data across distributed compute resources, providing new insights and knowledge.

**Big Data Processing Tools**

The Big Data processing technologies provide ways to work with large sets of structured,

semi-structured, and unstructured data so that value can be derived from big data.

In this video, we are going to talk about three open source technologies and the role

they play in big data analytics — Apache Hadoop, Apache Hive, and Apache Spark. **Hadoop** is a collection of tools that provides distributed storage and processing of big data. **Hive** is a data warehouse for data query and analysis built on top of Hadoop. **Spark** is a distributed data analytics framework designed to perform complex data analytics in real-time.

**Hadoop**, a java-based open-source framework, allows distributed storage and processing

of large datasets across clusters of computers. In Hadoop distributed system, a node is a single computer, and a collection of nodes forms a cluster. Hadoop can scale up from a single node to any number of nodes, each offering local storage and computation. Hadoop provides a reliable, scalable, and cost-effective solution for storing data with no format requirements.

Using Hadoop, you can:

* Incorporate emerging data formats, such as streaming audio, video, social media sentiment, and clickstream data, along with structured, semi-structured, and unstructured data not traditionally used in a data warehouse.
* Provide real-time, self-service access for all stakeholders.
* Optimize and streamline costs in your enterprise data warehouse by consolidating data across the organization and moving “cold” data, that is, data that is not in frequent use, to a Hadoop-based system.

One of the four main components of Hadoop is Hadoop Distributed File System, or HDFS,

which is a storage system for big data that runs on multiple commodity hardware connected

through a network. HDFS provides scalable and reliable big data storage by partitioning files over multiple nodes. It splits large files across multiple computers, allowing parallel access to them. Computations can, therefore, run in parallel on each node where data is stored. It also replicates file blocks on different nodes to prevent data loss, making it fault tolerant.

Let’s understand this through an example.

Consider a file that includes phone numbers for everyone in the United States; the numbers

for people with last name starting with A might be stored on server 1, B on server 2, and so on.

With Hadoop, pieces of this phonebook would be stored across the cluster. To reconstruct the entire phonebook, your program would need the blocks from every server in the cluster.

HDFS also replicates these smaller pieces onto two additional servers by default, ensuring

availability when a server fails, In addition to higher availability, this offers multiple benefits.

It allows the Hadoop cluster to break up work into smaller chunks and run those jobs on

all servers in the cluster for better scalability.

Finally, you gain the benefit of data locality, which is the process of moving the computation

closer to the node on which the data resides. This is critical when working with large data sets because it minimizes network congestion and increases throughput.

Some of the other benefits that come from using HDFS include:

* Fast recovery from hardware failures, because HDFS is built to detect faults and automatically recover.
* Access to streaming data, because HDFS supports high data throughput rates.
* Accommodation of large data sets, because HDFS can scale to hundreds of nodes, or computers, in a single cluster.
* Portability, because HDFS is portable across multiple hardware platforms and compatible with a variety of underlying operating systems.

**Hive** is an open-source data warehouse software for reading, writing, and managing large data

set files that are stored directly in either HDFS or other data storage systems such as Apache HBase. Hadoop is intended for long sequential scans and, because Hive is based on Hadoop, queries have very high latency—which means Hive is less appropriate for applications that need very fast response times. Also, Hive is read-based, and therefore not suitable for transaction processing that typically involves a high percentage of write operations.

Hive is better suited for data warehousing tasks such as ETL, reporting, and data analysis

and includes tools that enable easy access to data via SQL.

This brings us to **Spark**, a general-purpose data processing engine designed to extract and process large volumes of data for a wide range of applications, including Interactive Analytics, Streams Processing, Machine Learning, Data Integration, and ETL. It takes advantage of in-memory processing to significantly increase the speed of computations and spilling to disk only when memory is constrained.

Spark has interfaces for major programming languages, including Java, Scala, Python,

R, and SQL. It can run using its standalone clustering technology as well as on top of other infrastructures such as Hadoop. And it can access data in a large variety of data sources, including HDFS and Hive, making it highly versatile. The ability to process streaming data fast and perform complex analytics in real-time is the key use case for Apache Spark.

**Data Mining**

**Establishing Data Mining Goals**

The first step in data mining requires you to set up goals for the exercise. Obviously, you must identify the key questions that need to be answered. However, going beyond identifying the key questions are the concerns about the costs and benefits of the exercise. Furthermore, you must determine, in advance, the expected level of accuracy and usefulness of the results obtained from data mining. If money were no object, you could throw as many funds as necessary to get the answers required. However, the cost-benefit trade-off is always instrumental in determining the goals and scope of the data mining exercise. The level of accuracy expected from the results also influences the costs. High levels of accuracy from data mining would cost more and vice versa. Furthermore, beyond a certain level of accuracy, you do not gain much from the exercise, given the diminishing returns. Thus, the cost-benefit trade-offs for the desired level of accuracy are important considerations for data mining goals.

**Selecting Data**

The output of a data-mining exercise largely depends upon the quality of data being used. At times, data are readily available for further processing. For instance, retailers often possess large databases of customer purchases and demographics. On the other hand, data may not be readily available for data mining. In such cases, you must identify other sources of data or even plan new data collection initiatives, including surveys. The type of data, its size, and frequency of collection have a direct bearing on the cost of data mining exercise. Therefore, identifying the right kind of data needed for data mining that could answer the questions at reasonable costs is critical.

**Preprocessing Data**

Preprocessing data is an important step in data mining. Often raw data are messy, containing erroneous or irrelevant data. In addition, even with relevant data, information is sometimes missing. In the preprocessing stage, you identify the irrelevant attributes of data and expunge such attributes from further consideration. At the same time, identifying the erroneous aspects of the data set and flagging them as such is necessary. For instance, human error might lead to inadvertent merging or incorrect parsing of information between columns. Data should be subject to checks to ensure integrity. Lastly, you must develop a formal method of dealing with missing data and determine whether the data are missing randomly or systematically.

If the data were missing randomly, a simple set of solutions would suffice. However, when data are missing in a systematic way, you must determine the impact of missing data on the results. For instance, a particular subset of individuals in a large data set may have refused to disclose their income. Findings relying on an individual's income as input would exclude details of those individuals whose income was not reported. This would lead to systematic biases in the analysis. Therefore, you must consider in advance if observations or variables containing missing data be excluded from the entire analysis or parts of it.

**Transforming Data**

After the relevant attributes of data have been retained, the next step is to determine the appropriate format in which data must be stored. An important consideration in data mining is to reduce the number of attributes needed to explain the phenomena. This may require transforming data Data reduction algorithms, such as Principal Component Analysis (demonstrated and explained later in the chapter), can reduce the number of attributes without a significant loss in information. In addition, variables may need to be transformed to help explain the phenomenon being studied. For instance, an individual's income may be recorded in the data set as wage income; income from other sources, such as rental properties; support payments from the government, and the like. Aggregating income from all sources will develop a representative indicator for the individual income.

Often you need to transform variables from one type to another. It may be prudent to transform the continuous variable for income into a categorical variable where each record in the database is identified as low, medium, and high-income individual. This could help capture the non-linearities in the underlying behaviors.

**Storing Data**

The transformed data must be stored in a format that makes it conducive for data mining. The data must be stored in a format that gives unrestricted and immediate read/write privileges to the data scientist. During data mining, new variables are created, which are written back to the original database, which is why the data storage scheme should facilitate efficiently reading from and writing to the database. It is also important to store data on servers or storage media that keeps the data secure and also prevents the data mining algorithm from unnecessarily searching for pieces of data scattered on different servers or storage media. Data safety and privacy should be a prime concern for storing data.

**Mining Data**

After data is appropriately processed, transformed, and stored, it is subject to data mining. This step covers data analysis methods, including parametric and non-parametric methods, and machine-learning algorithms. A good starting point for data mining is data visualization. Multidimensional views of the data using the advanced graphing capabilities of data mining software are very helpful in developing a preliminary understanding of the trends hidden in the data set.

*Later sections in this chapter detail data mining algorithms and methods.*

**Evaluating Mining Results**

After results have been extracted from data mining, you do a formal evaluation of the results. Formal evaluation could include testing the predictive capabilities of the models on observed data to see how effective and efficient the algorithms have been in reproducing data. This is known as an "in-sample forecast". In addition, the results are shared with the key stakeholders for feedback, which is then incorporated in the later iterations of data mining to improve the process.

Data mining and evaluating the results becomes an iterative process such that the analysts use better and improved algorithms to improve the quality of results generated in light of the feedback received from the key stakeholders.