**Data Science at Scale**

**Learning Objectives**

* Gain an understanding of Big Data and the Hadoop ecosystem
* Learn the fundamentals for Spark and PySpark and be able to apply Apache Spark to scale machine learning applications
* Explore different types of neural networks, such as Recurrent and LSTM Neural Networks, LSTM Neural Networks, and Convolutional Neural Networks

**Work to Complete**

In this unit, you'll:

* Submit ideas and project proposals for your third capstone project

'Big Data' is data whose scale, distribution, diversity, and timeliness require the use of new technical architectures and analytics to enable insights that unlock new sources of business value. The quantity of data in the world is growing daily and will continue to grow each year. Social media, for example, produces data at an unprecedented scale and is constantly doing so in the background of our lives. To drive this point home, consider this: the amount of total worldwide data in the first decade of the 21st century was measured in petabytes (PB), or 1,000,000,000,000,000 bytes. As of 2020, the amount of worldwide data is stored in zettabytes (ZB), or 1,000,000,000,000,000,000,000 bytes.

Big Data refers to datasets composed of massive amounts of both structured and unstructured data. These datasets are so large that it can be difficult to process this data using more traditional techniques. (In)famously, Big Data can be characterized by the four V's:

* **Volume**: a high volume of data
* **Velocity**: speed of data creation and growth
* **Variety**: different structures and sources
* **Veracity**: quality of the data (rarely good quality)

Big Data can seem intimidating — almost like a tsunami of information that cannot be stopped. There are two principal challenges facing us: storing the data and analyzing it. However, thanks to the development of GPUs (Graphics Processing Units, but used for storage) and advanced analytical tools, Big Data can be exploited in data science applications for banking, medicine, business, and science more than it ever could before. It's possible to surf the big wave of Big Data; you just need the right technologies and the right skills, which you’ll start to learn in this subunit and continue learning later in the course.

We will be looking at Hadoop in one subunit and Spark and Pyspark in the next. Hadoop is a collection of open-source software utilities that permit the storage and analysis of big data via a network of servers working together. Apache Spark is an open-source cluster computing framework that can be used within Python notebooks via the PySpark API. If you're working on a project that uses Python to store and analyze big data, the chances are you'll be using Hadoop, Spark, and PySpark — whether you're a data scientist, data engineer, data architect, or project manager. We want this unit to give you fluency with the language of big data, and a good degree of technical knowledge to boot.

### The Hadoop Ecosystem

Hadoop is a collection of open-source software utilities that permit the storage and analysis of big data via a network of servers working together. It is often referred to as an ecosystem due to its modularity — it comprises different components that standalone and are substitutable, but that work neatly together. This subunit starts with carefully chosen, light-touch introductions to Big Data and Hadoop. You'll then do a good portion of a LinkedIn Learning course on Hadoop to go a bit deeper into the technical specifics. Good luck!

# What is Hadoop?

When you learn about Big Data you will sooner or later come across this odd sounding word: Hadoop – but what exactly is it?

Put simply**, Hadoop can be thought of as a set of open source programs and procedures (meaning essentially they are free for anyone to use or modify, with a few exceptions) which anyone can use as the “backbone” of their big data operations**.

I’ll try to keep things simple as I know a lot of people reading this aren’t software engineers, so I hope I don’t over-simplify anything – think of this as a brief guide for someone who wants to know a bit more about the nuts and bolts that make big data analysis possible.

**The 4 Modules of Hadoop**

Hadoop is made up of “modules”, each of which carries out a particular task essential for a computer system designed for big data analytics.

**1. Distributed File-System**

The most important two are the Distributed File System, which allows data to be stored in an easily accessible format, across a large number of linked storage devices, and the MapReduce – which provides the basic tools for poking around in the data.

(A “file system” is the method used by a computer to store data, so it can be found and used. Normally this is determined by the computer’s operating system, however a Hadoop system uses its own file system which sits “above” the file system of the host computer – meaning it can be accessed using any computer running any supported OS).

**2. MapReduce**

MapReduce is named after the two basic operations this module carries out – reading data from the database, putting it into a format suitable for analysis (map), and performing mathematical operations i.e counting the number of males aged 30+ in a customer database (reduce).

**3. Hadoop Common**

The other module is Hadoop Common, which provides the tools (in Java) needed for the user’s computer systems (Windows, Unix or whatever) to read data stored under the Hadoop file system.

**4. YARN**

The final module is YARN, which manages resources of the systems storing the data and running the analysis.

Various other procedures, libraries or features have come to be considered part of the Hadoop “framework” over recent years, but Hadoop Distributed File System, Hadoop MapReduce, Hadoop Common and Hadoop YARN are the principle four.

**How Hadoop Came About**

Development of Hadoop began when forward-thinking software engineers realised that it was quickly becoming useful for anybody to be able to store and analyze datasets far larger than can practically be stored and accessed on one physical storage device (such as a hard disk).

This is partly because as physical storage devices become bigger it takes longer for the component that reads the data from the disk (which in a hard disk, would be the “head”) to move to a specified segment. Instead, many smaller devices working in parallel are more efficient than one large one.

It was released in 2005 by the Apache Software Foundation, a non-profit organization which produces open source software which powers much of the Internet behind the scenes. And if you’re wondering where the odd name came from, it was the name given to a toy elephant belonging to the son of one of the original creators!

**The Usage of Hadoop**

The flexible nature of a Hadoop system means companies can add to or modify their data system as their needs change, using cheap and readily-available parts from any IT vendor.

Today, it is the most widely used system for providing data storage and processing across “commodity” hardware – relatively inexpensive, off-the-shelf systems linked together, as opposed to expensive, bespoke systems custom-made for the job in hand. In fact it is claimed that more than half of the companies in the Fortune 500 make use of it.

Just about all of the big online names use it, and as anyone is free to alter it for their own purposes, modifications made to the software by expert engineers at, for example, Amazon and Google, are fed back to the development community, where they are often used to improve the “official” product. This form of collaborative development between volunteer and commercial users is a key feature of open source software.

In its “raw” state – using the basic modules supplied here https://hadoop.apache.org/ by Apache, it can be very complex, even for IT professionals – which is why various commercial versions have been developed such as Cloudera which simplify the task of installing and running a Hadoop system, as well as offering training and support services.

So that, in a (fairly large) nutshell, is Hadoop. Thanks to the flexible nature of the system, companies can expand and adjust their data analysis operations as their business expands. And the support and enthusiasm of the open source community behind it has led to great strides towards making big data analysis more accessible for everyone.

# What is Spark in Big Data?

Basically Spark is a framework – in the same way that Hadoop is – which provides a number of inter-connected platforms, systems and standards for Big Data projects.

Like Hadoop, **Spark is open-source and under the wing of the Apache Software Foundation. Essentially, open-source means the code can be freely used by anyone. Beyond that, it can also be altered by anyone to produce custom versions aimed at particular problems, or industries.** Volunteer developers, as well as those working at companies which produce custom versions, constantly refine and update the core software adding more features and efficiencies. In fact Spark was the most active project at Apache last year. It was also the most active of all of the open source Big Data applications, with over 500 contributors from more than 200 organizations.

Spark is seen by techies in the industry as a more advanced product than Hadoop – it is newer, and designed to work by processing data in chunks “in memory”. This means it transfers data from the physical, magnetic hard discs into far-faster electronic memory where processing can be carried out far more quickly – up to 100 times faster in some operations.

Spark has proven very popular and is used by many large companies for huge, multi-petabyte data storage and analysis. This has partly been because of its speed. Last year, Spark set a world record by completing a benchmark test involving sorting 100 terabytes of data in 23 minutes – the previous world record of 71 minutes being held by Hadoop.

Additionally, Spark has proven itself to be highly suited to Machine Learning applications. Machine Learning is one of the fastest growing and most exciting areas of computer science, where computers are being taught to spot patterns in data, and adapt their behaviour based on automated modelling and analysis of whatever task they are trying to perform.

It is designed from the ground up to be easy to install and use – if you have a background in computer science! In order to make it available to more businesses, many vendors provide their own versions (as with Hadoop) which are geared towards particular industries, or custom-configured for individual clients’ projects, as well as associated consultancy services to get it up and running.

Spark uses cluster computing for its computational (analytics) power as well as its storage. This means it can use resources from many computer processors linked together for its analytics. It’s a scalable solution meaning that if more oomph is needed, you can simply introduce more processors into the system. With distributed storage, the huge datasets gathered for Big Data analysis can be stored across many smaller individual physical hard discs. This speeds up read/write operations, because the “head” which reads information from the discs has less physical distance to travel over the disc surface. As with processing power, more storage can be added when needed, and the fact it uses commonly available commodity hardware (any standard computer hard discs) keeps down infrastructure costs.

Unlike Hadoop, Spark does not come with its own file system – instead it can be integrated with many file systems including Hadoop’s HDFS, MongoDB and Amazon’s S3 system.

Another element of the framework is Spark Streaming, which allows applications to be developed which perform analytics on streaming, real-time data – such as automatically analyzing video or social media data – on-the-fly, in real-time.

In fast changing industries such as marketing, real-time analytics has huge advantages, for example ads can be served based on a user’s behavior at a particular time, rather than on historical behavior, increasing the chance of prompting an impulse purchase.

So that’s a brief introduction to Apache Spark – what it is, how it works, and why a lot of people think that it’s the future. I hope you found it useful.

**Spark and Pyspark**

**Apache Spark is an open-source cluster computing framework that can be used within Python notebooks via the PySpark API**. Spark is said to be part of the Hadoop Ecosystem, in that it lives on top of Hadoop and is supported by Hadoop's various modules. In today's data science projects, you need to ingest big data in dedicated production environments. These often contain things like:

* Amazon S3 buckets - essentially big repositories for data)
* AWS jobs - which execute Python scripts to clean, transform, or enrich data)
* AWS crawlers - programs that make tables from parquet files, for analysis

The output of your production environment might be a set of tables that are perfectly ready for visualizations, widgets, EDA, and analysis with machine learning algorithms. Those transformation and enrichment scripts, if written in Python, will very likely make use of PySpark. In addition to understanding Python scripts that contain PySpark functions, it's essential to understand how Spark and PySpark work under the hood. Where is the data being stored? What machines are actually doing the processing? How much will it cost, and how fault-tolerant is it? This subunit will answer those questions and more!