PySpark
Combining Machine Learning & Big Data

Date : 09-09-2020 | Speaker : Ayon Roy | Event : Global AI Tour 2020

Visit - AYONROY.ML

Hello **Buddy!**

I am **Ayon Roy**

B.Tech CSE (2017-2021)

Data Science Intern @ **Lulu International Exchange**, Abu Dhabi

(**World's Leading Financial Services Company**)

Brought **Kaggle Days Meetup** Community in India for the 1st time

If you haven't heard about me yet, you might have been living under the rocks. Wake up !!

Agenda (09-09-2020)

- What is Big Data ?
- What is Machine Learning ?
- Why do we need to fuse Big Data with Machine Learning ?
- What is Spark & how it's architecture will help us in doing ML ?
- How to harness the power of Spark using Python ?
  (Here comes PySpark)
- How Spark's ML library will help us do ML seamlessly using PySpark ?

Defining Big Data

Big data is a domain that analyzes, extracts information from huge datasets which maybe beyond the ability of general tools to manage, process data.

Volume : Scale of Data

Variety : Different types of Data

Velocity : Speedy Ingestion of new Data

Veracity : Uncertainty in the Data

Defining Machine Learning

An Approach to Achieve Artificial Intelligence

Subfield of AI that aims to teach computers the ability to do tasks with data, without explicit programming.

We can get AI without using machine learning, but this would require building millions of lines of codes with complex rules and decision-trees.

So instead of hard-coding software routines with specific instructions to accomplish a particular task, machine learning is a way of “training” an algorithm so that it can learn how.

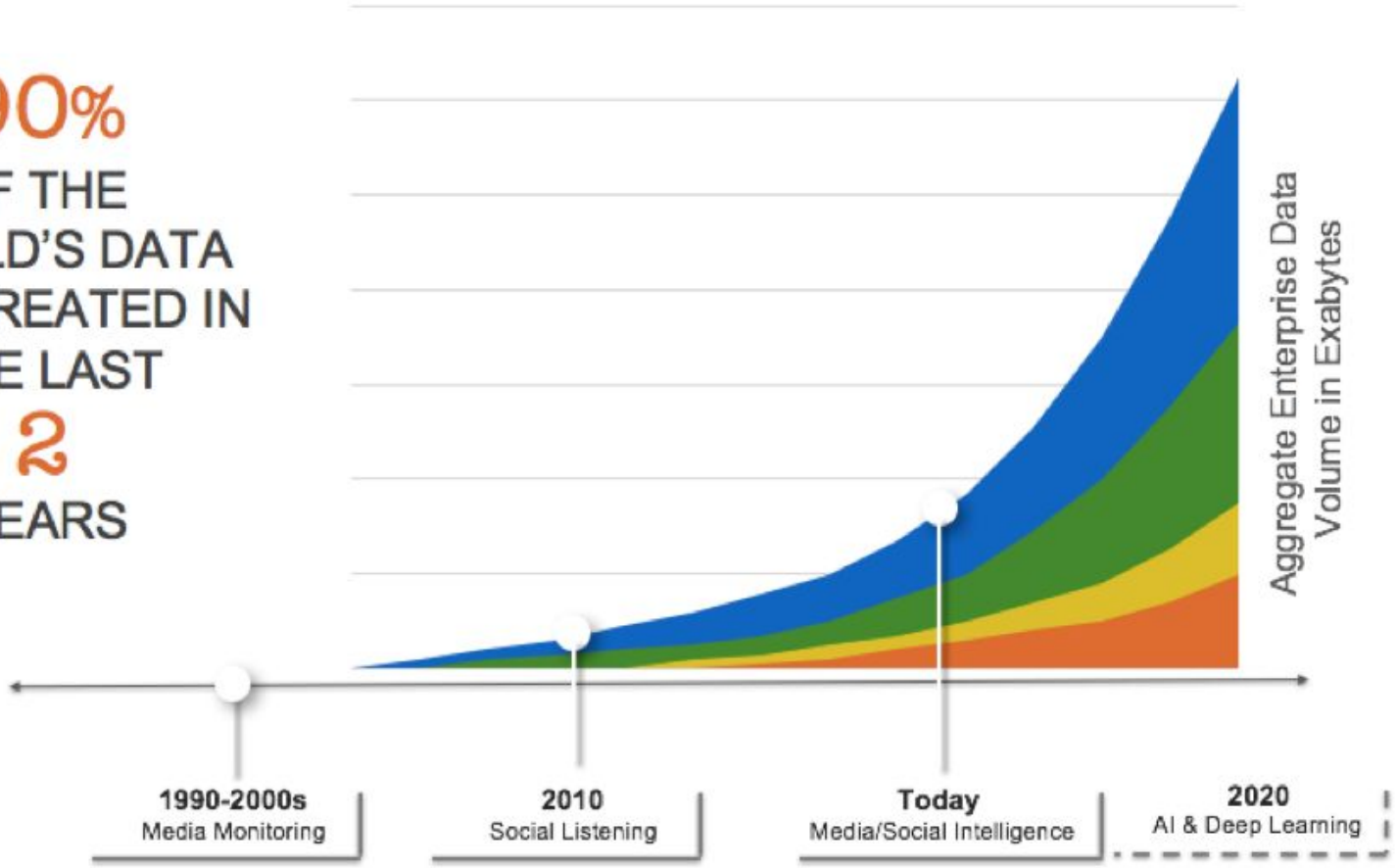
Why do we need to fuse Big Data & Machine Learning ?

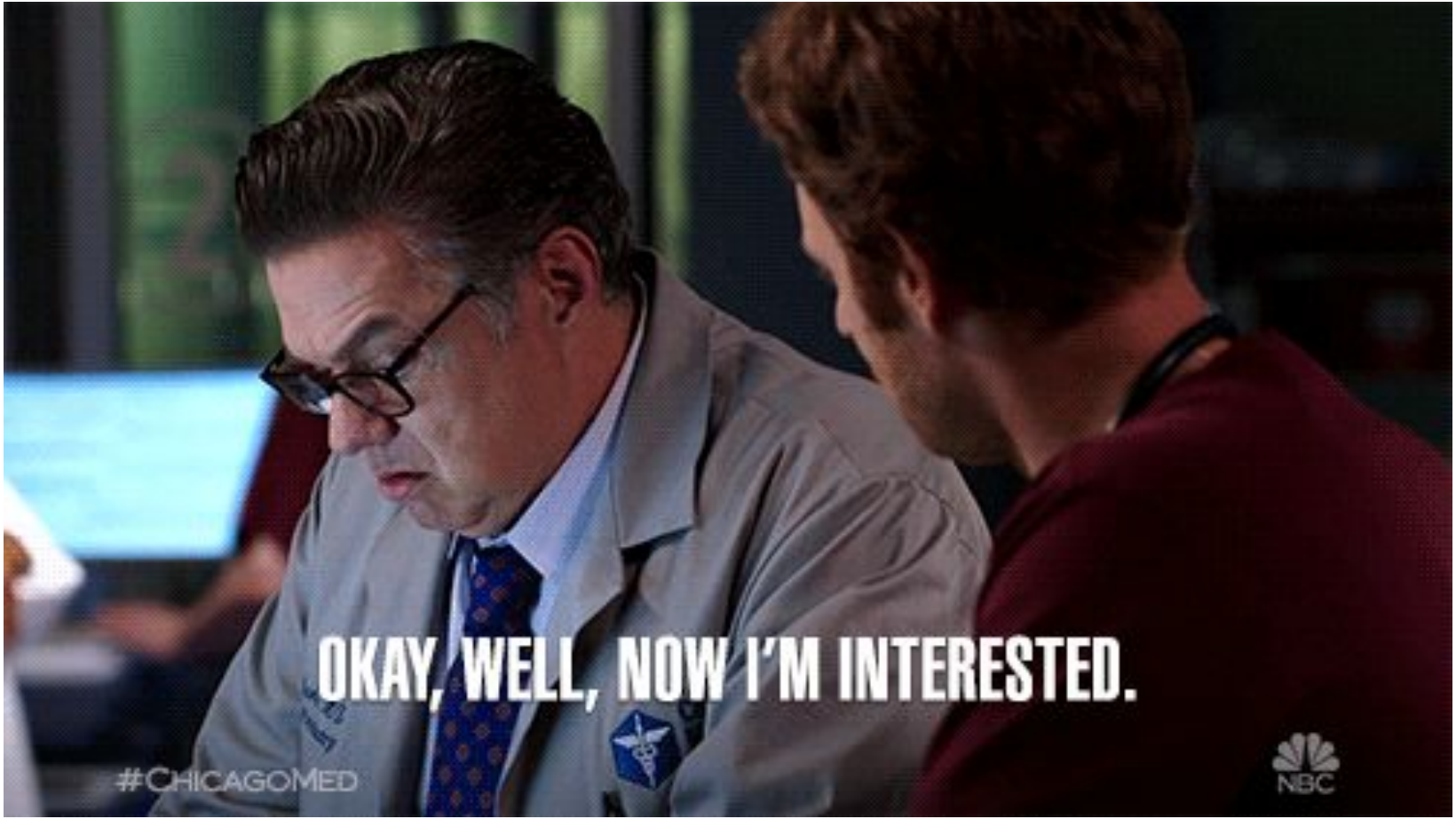
90%

OF THE WORLD'S DATA WAS CREATED IN THE LAST

2

YEARS





What is Spark ?

Apache Spark is a unified analytics engine for large-scale data processing. It provides high-level APIs in Java, Scala, Python and R, and an optimized engine that supports general execution graphs.

It also supports a rich set of higher-level tools including **Spark SQL for SQL and structured data processing, MLlib for machine learning, GraphX for graph processing, and Structured Streaming for incremental computation and stream processing.**

History of Spark APIs

Distribute collection of JVM objects

Functional Operators (map, filter, etc.)

Distribute collection of Row objects

Expression-based operations and UDFs

Logical plans and optimizer

Fast/efficient internal representations

Internally rows, externally JVM objects

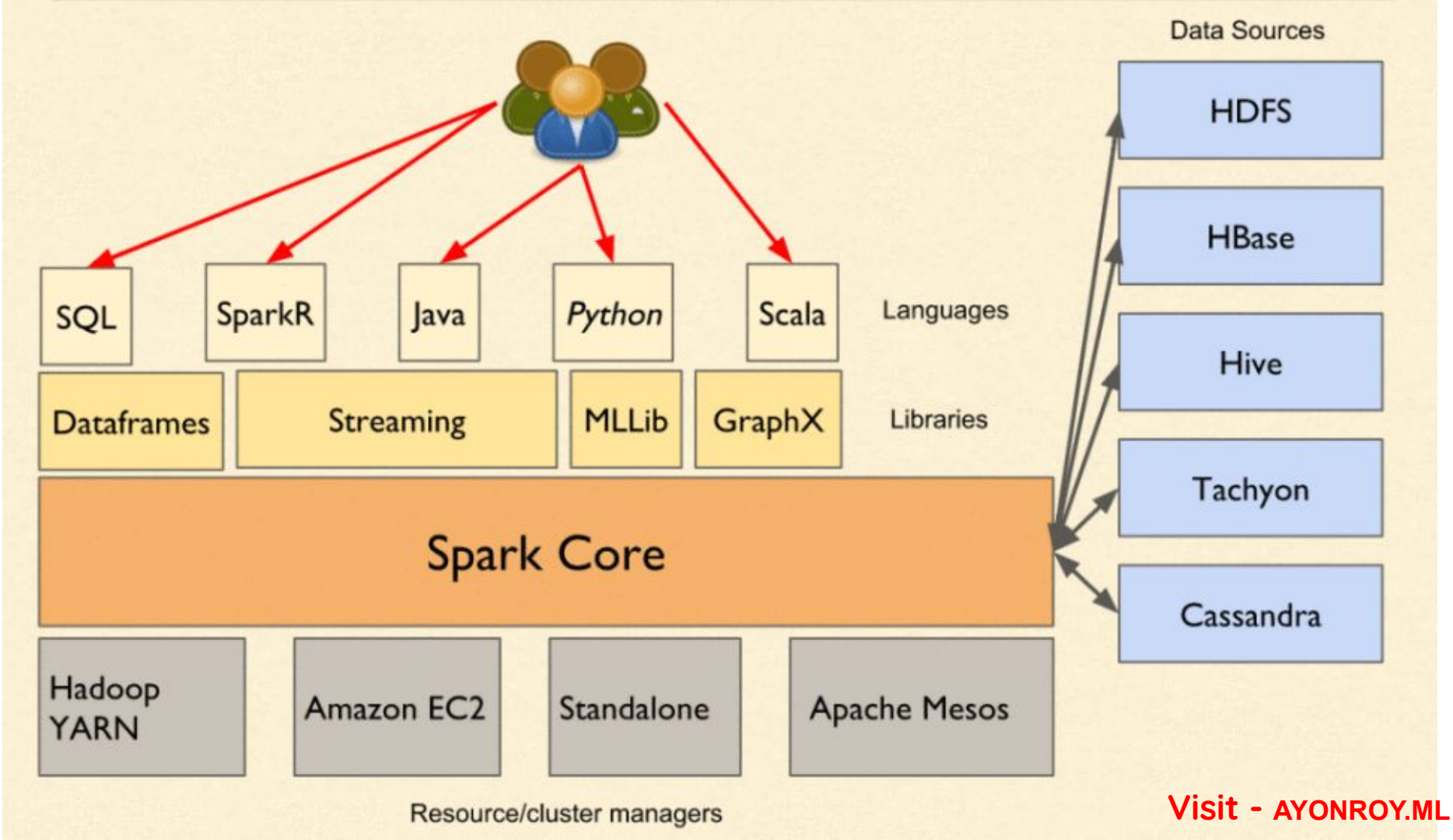
Almost the "Best of both worlds": **type safe + fast**

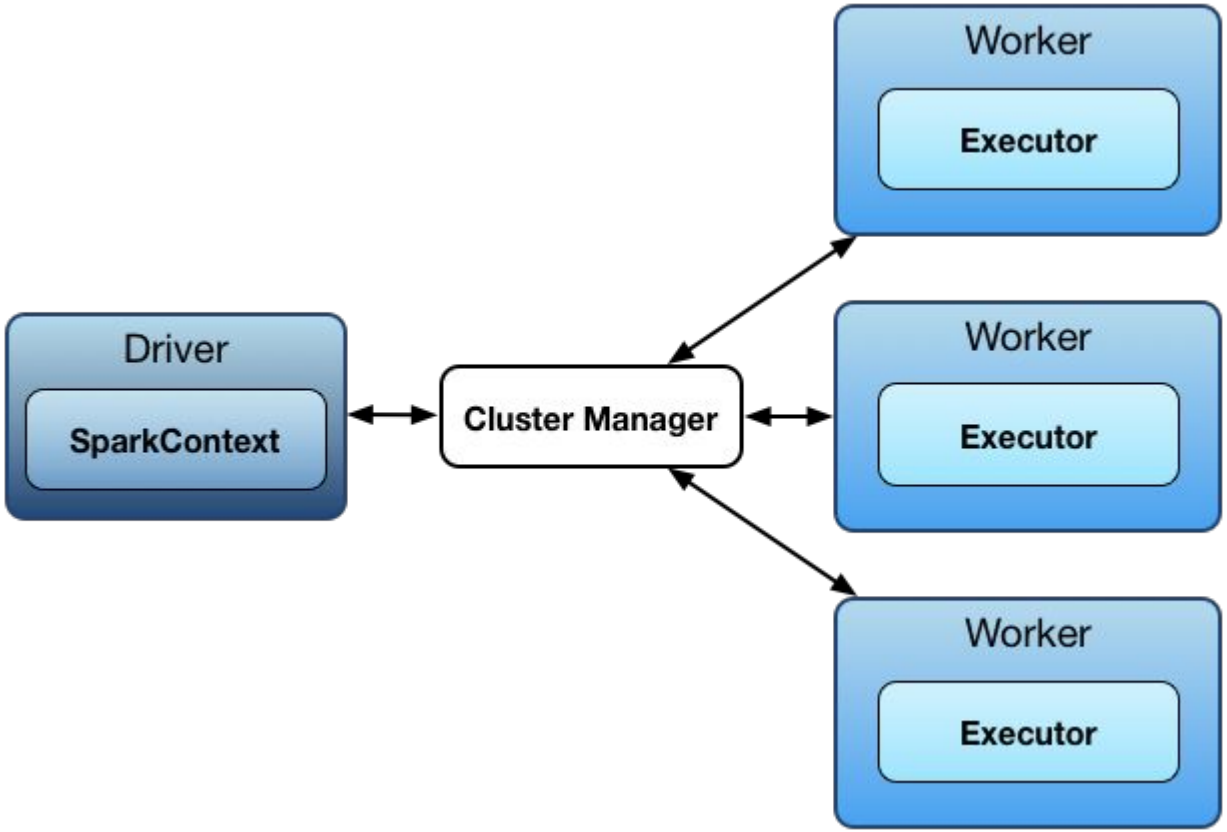
But slower than DF
Not as good for interactive analysis, especially Python

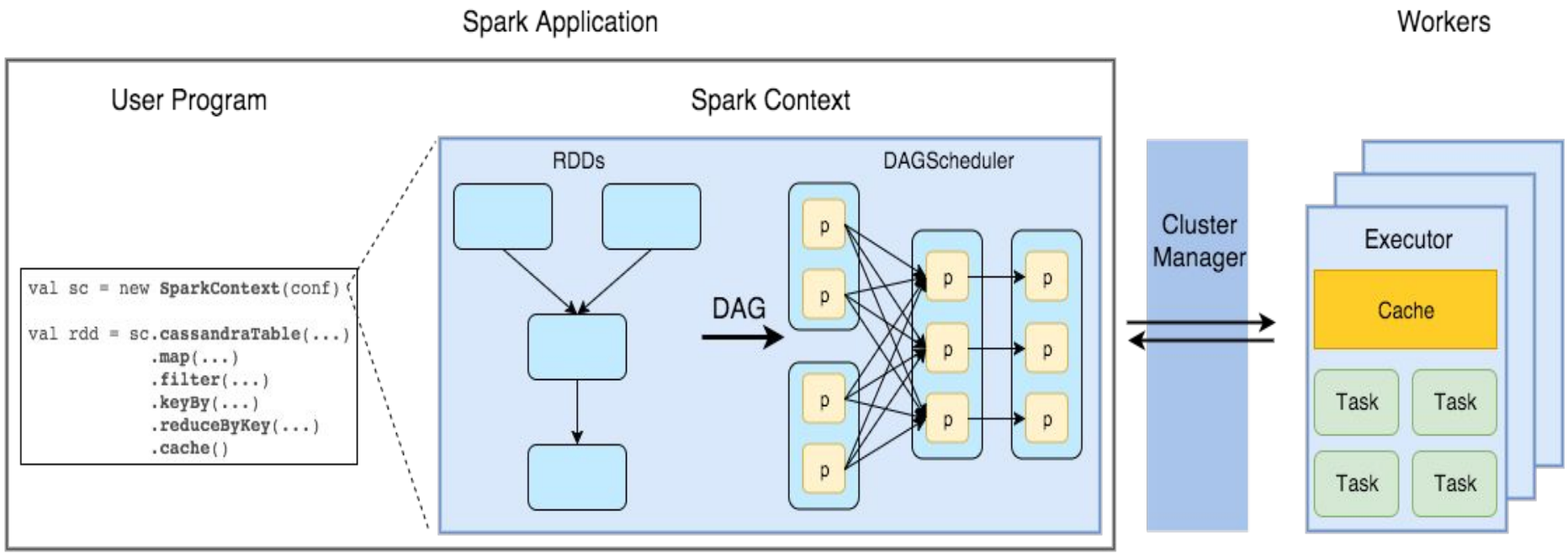
databricks

Feature	RDD	DataFrame	Dataset
Immutable	Yes	Yes	Yes
Fault Tolerant	Yes	Yes	Yes
Type-Safe	Yes	No	Yes
Schema	No	Yes	Yes
Execution Optimization	No	Yes	Yes
Optimizer Engine	N/A	Catalyst Engine	Catalyst Engine
API Level for manipulating distributed collection of data	Low	High	High
language Support	Java, Scala, Pyt	Java, Scala, Python, R	Java, Scala

How Spark's Architecture will help us in doing Machine Learning ?







- **Spark Context:** It holds a connection with Spark cluster manager. All Spark applications run as independent set of processes, coordinated by a SparkContext in a program.

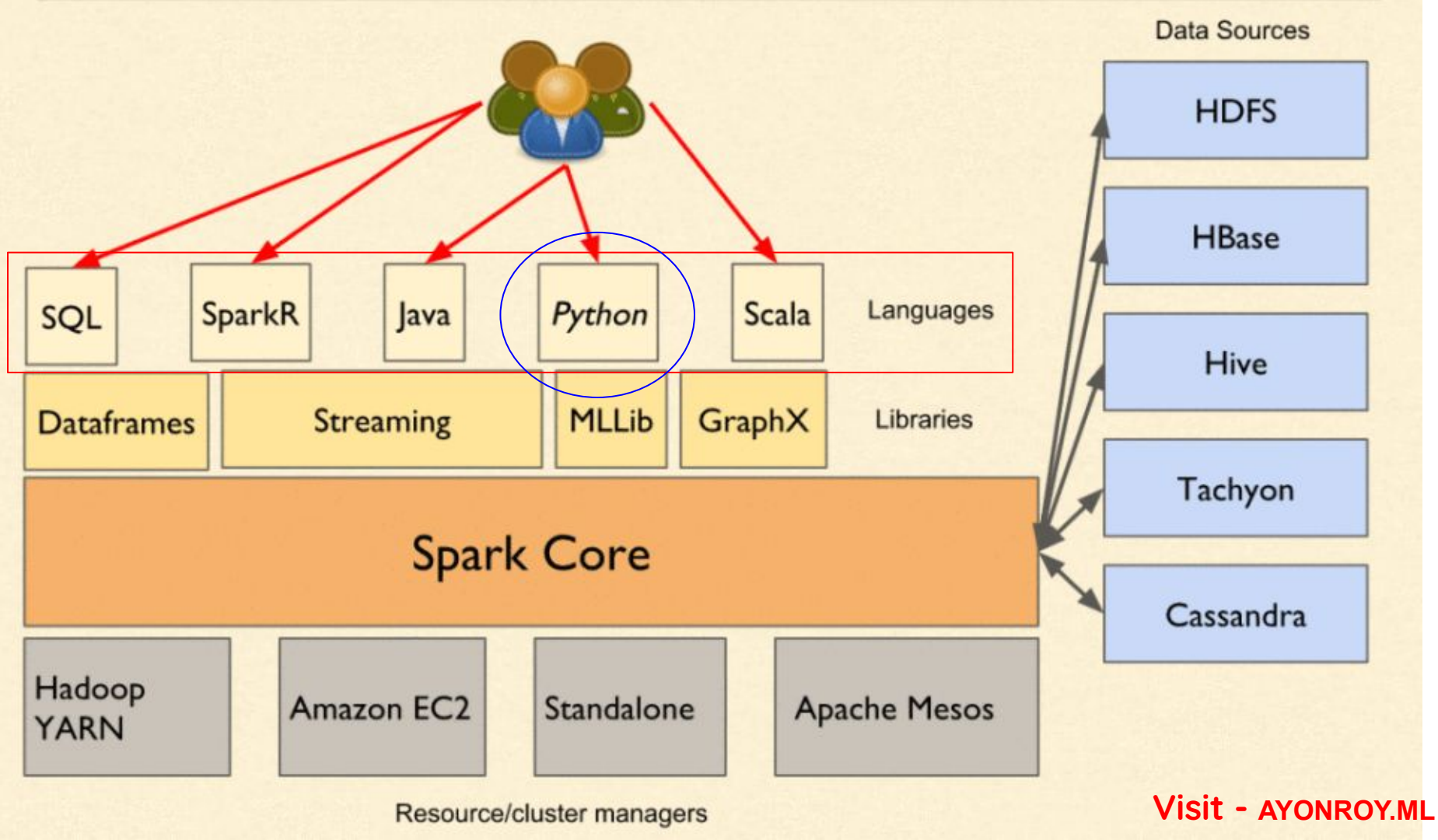
- **Driver :** A driver is incharge of the process of running the main() function of an application and creating the SparkContext.

- **Executor** : Executors are worker nodes' processes in charge of running individual tasks in a given Spark job. They are launched at the beginning of a Spark application and typically run for the entire lifetime of an application.

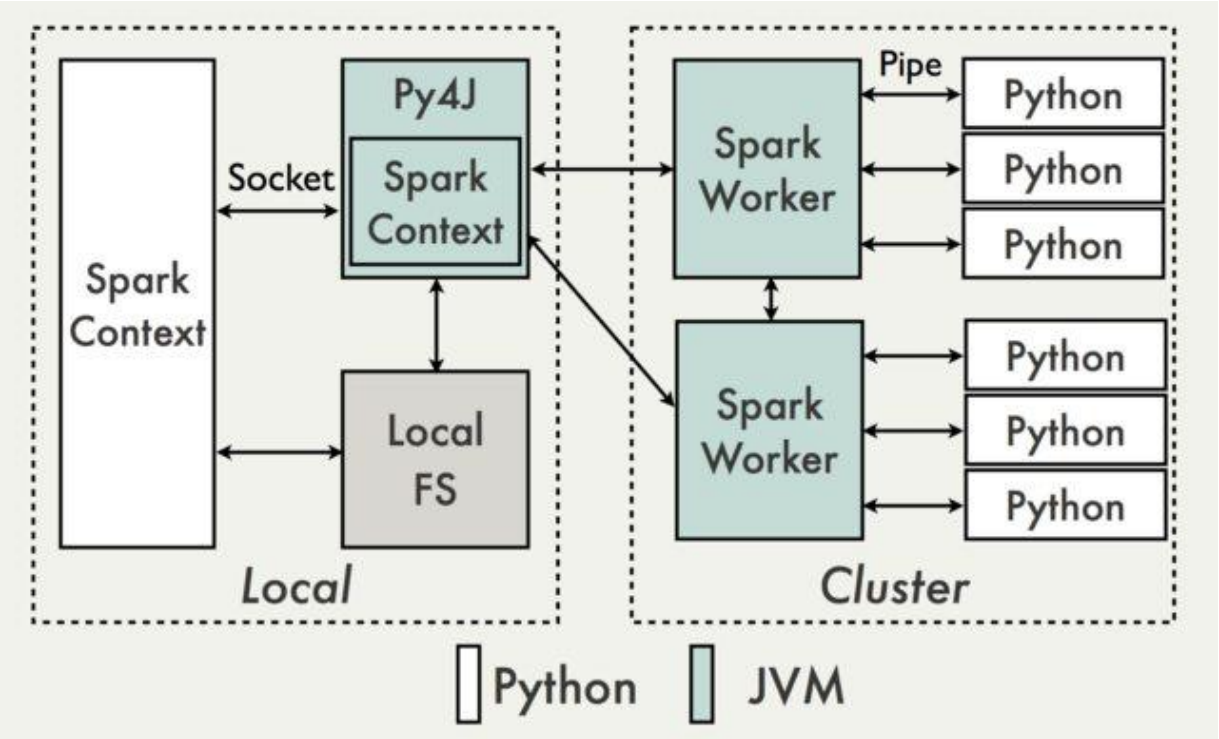
- **Worker** : A worker, on the other hand, is any node that can run program in the cluster. If a process is launched for an application, then this application acquires executors at worker node.

- **Cluster Manager:** Cluster manager allocates resources to each application in driver program. There are three types of cluster managers supported by Apache Spark – Standalone, Mesos and YARN.

How to harness the power of Spark using Python ?



Apache Spark is written in Scala programming language. To support Python with Spark, Apache Spark community released a tool, PySpark. Using PySpark, you can work with RDDs in Python programming language also. It is because of a library called Py4j that they are able to achieve this.



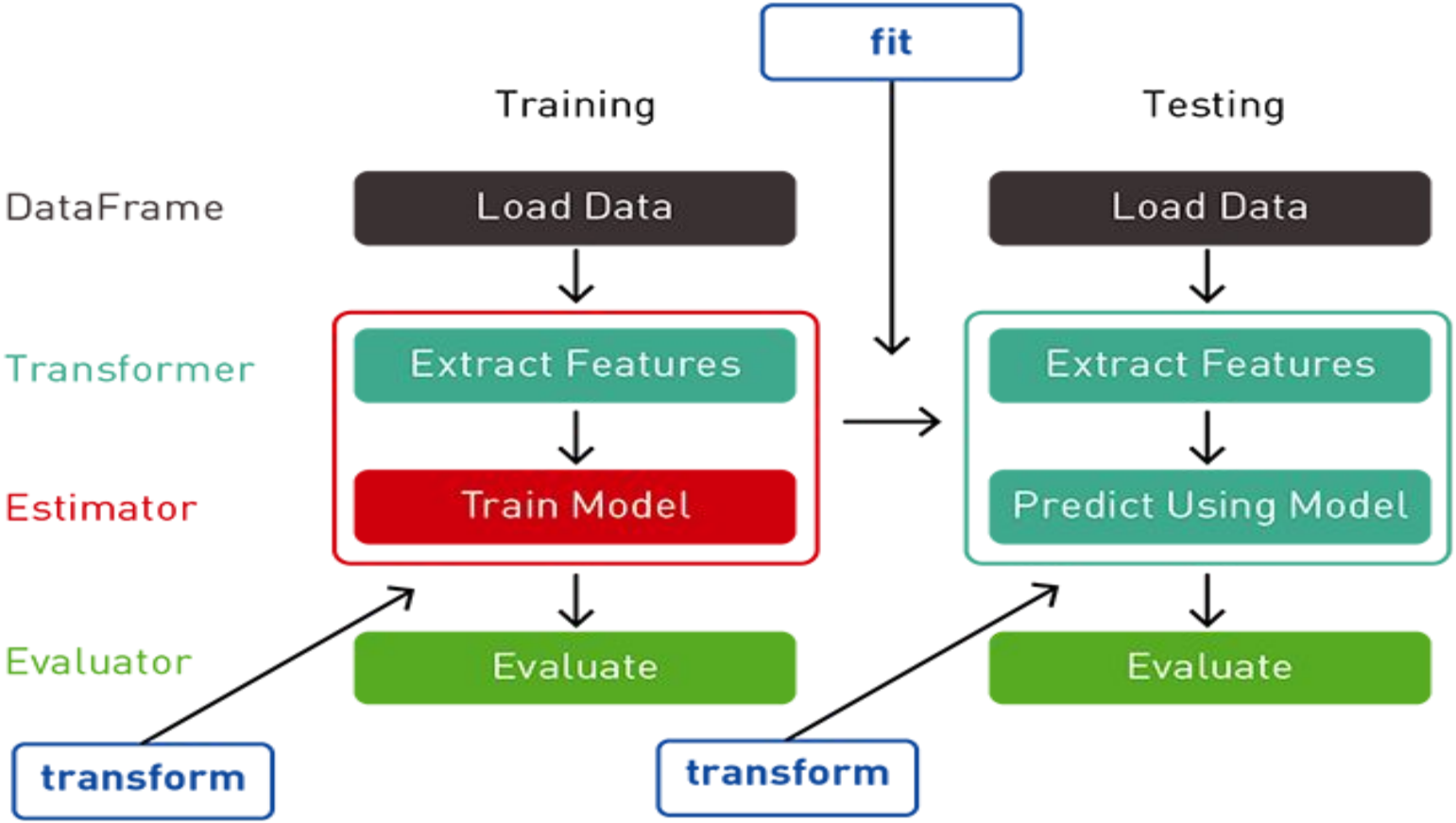
PySpark's Architecture

How Spark's ML library will help us achieve our goal using PySpark ?

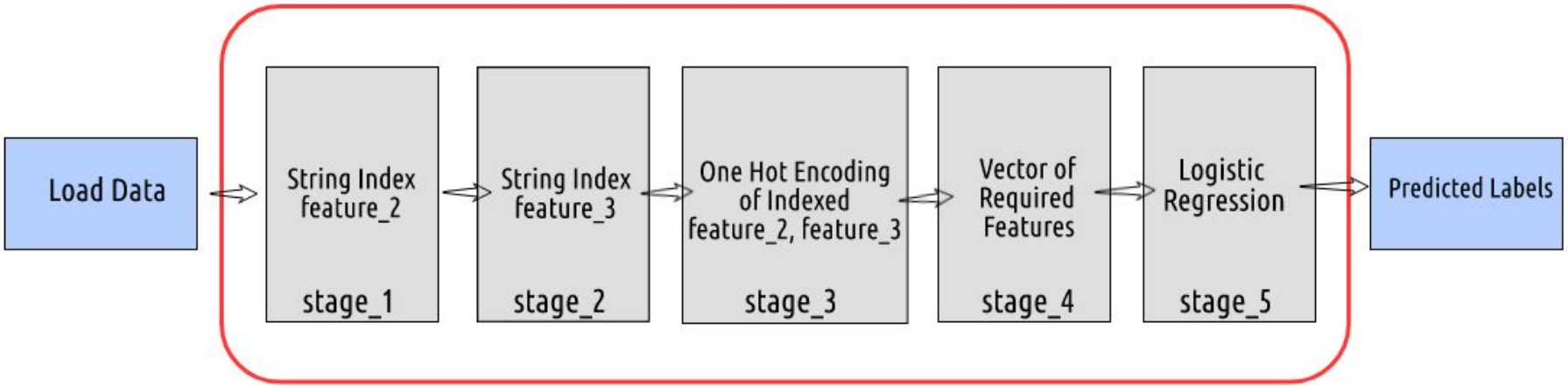
MLlib is Spark's machine learning (ML) library. Its goal is to make practical machine learning scalable and easy. At a high level, it provides tools such as:

- **ML Algorithms** : Common learning algorithms such as classification, regression, clustering, and collaborative filtering
- **Featurization** : Feature extraction, transformation, dimensionality reduction, and selection
- **Pipelines** : Tools for constructing, evaluating, and tuning ML Pipelines
- **Persistence** : Saving and load algorithms, models, and Pipelines
- **Utilities** : Linear algebra, statistics, data handling, etc.

Spark ML Workflow



- **DataFrame**: This ML API uses DataFrame from Spark SQL as an ML dataset, which can hold a variety of data types. E.g., a DataFrame could have different columns storing text, feature vectors, true labels, and predictions.
- **Transformer**: A Transformer is an algorithm which can transform one DataFrame into another DataFrame. E.g., an ML model is a Transformer which transforms a DataFrame with features into a DataFrame with predictions.
- **Estimator**: An Estimator is an algorithm which can be fit on a DataFrame to produce a Transformer. E.g., a learning algorithm is an Estimator which trains on a DataFrame and produces a model.
- **Pipeline**: A Pipeline chains multiple Transformers and Estimators together to specify an ML workflow.



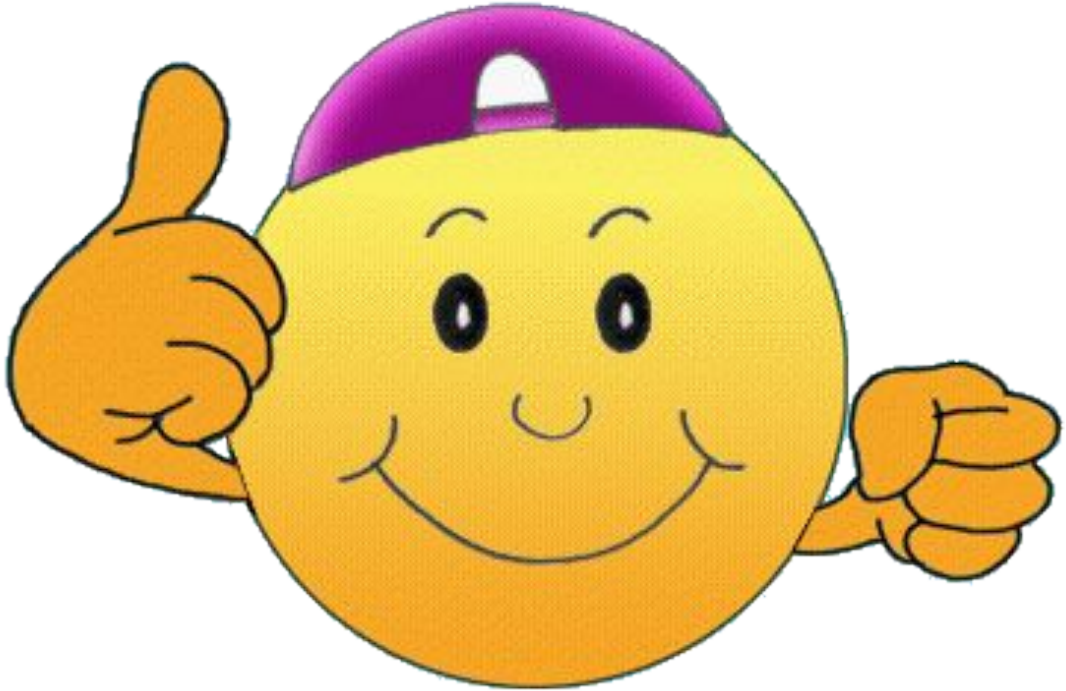
Spark Machine Learning library MLlib contains the following applications –

- Collaborative Filtering for Recommendations – Alternating Least Squares
- Logistic Regression, Lasso Regression, Ridge Regression, Linear Regression and Support Vector Machines (SVM).
- Linear Discriminant Analysis, K-Mean and Gaussian,
- Naïve Bayes, Ensemble Methods, and Decision Trees.
- PCA (Principal Component Analysis) and Singular Value Decomposition (SVD).

A few useful resources

1. https://spark.apache.org/
2. https://spark.apache.org/mllib/
3. https://www.analyticsvidhya.com/blog/2016/09/comprehensive-introduction-to-apache-spark-rdds-dataframes-using-pyspark/
4. https://docs.databricks.com/getting-started/spark/machine-learning.html
5. https://www.datacamp.com/community/tutorials/apache-spark-tutorial-machine-learning

GO FOR IT !



GOOD LUCK !

Let me answer your Questions now.

Finally, it’s your time to speak.

Danke Schoen

Questions ? Any Feedbacks ? Did you like the talk?
Tell me about it.

If you think I can help you,
connect with me via

Email : ayonroy2000@gmail.com

LinkedIn / Github / Telegram Username : ayonroy2000

Website : https://AYONROY.ML/