Spark MLlib

Milad Alshomary
milad.alshomary@uni-weimar.de

Agenda:

- Motivation
  - Machine Learning definition.
  - Challenges.
- Apache Spark review.
- Apache Spark MLlib
  - Basic data types
  - Algorithms
  - Pipeline
- Spark on Yarn

Motivation

- Machine Learning Definition:
  - A field of study that gives computers the ability to learn without being explicitly programmed.
  - It’s the study and construction of algorithms that can learn from **DATA** to make predictions.

Motivation

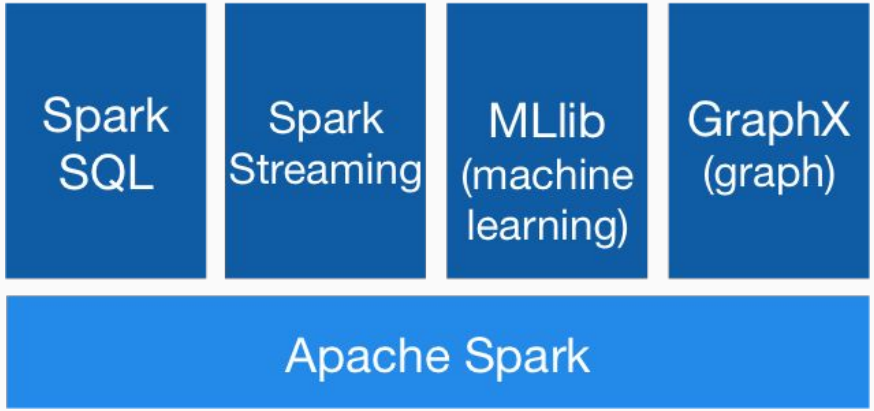
- Machine Learning Challenges:
  - **DATA** in Machine Learning is said to be **EXPERIENCE**.
  - With more **DATA** in hand we could build more accurate ML models.
  - As the size of the data grows, handling it on one machine becomes impossible due the limitation of memory and cpu.
  - The solution for this limitation is to distribute this data across multiple machines and process it in parallel way.

Apache Spark MLlib

- Apache Spark:
  - Open source processing engine that runs parallel tasks to to process data distributed on a cluster of nodes.
  - **Map/reduce** methodology for processing the data.
  - 10x faster than **hadoop** because of the memory utilizations.
  - Tasks/applications can be programmed using: Python, Scala or java.

Apache Spark MLlib

- Apache Spark:



Apache Spark MLlib

- **MLlib**:
  - A Spark component runs on top of Apache Spark core and provide a framework to execute machine learning routines on massive data sets distributed on a cluster.

Apache Spark MLlib

- **MLlib**:

spark.mllib package

- Contains various machine learning routines and run over RDDs
- RDD: is a collection of elements partitioned across the nodes of a cluster and can be operated in parallel.

spark.ml package

- Provide a unified interface to construct machine learning pipelines and runs on top of Spark dataFrams.
- DataFrams: Is distributed collection of data organized into named column. Its like a table in relational databse and could be constructed from wide range of sources like: hdfs files, RDDs or structured files.

Apache Spark MLlib

- **Basic data types in MLlib**:
  - Local vector: 0-based indices and double typed values. Stored on single machine and can be sparse or dense. Example: vector of weights.
  - Labeled point: A class of vector (sparse or dense) and doubled value label. Example: classification on labeled data.
  - Local Matrix: Integer typed column and row indices and double typed values. Stored in single machine and can be dense or sparse (csc format).
  - Distributed Matrix: A long typed row and column indices. It is distributed on RDDs.
    - RawMatrix, RowIndexedMatrix, CoordinateMatrix (COO format).

Apache Spark MLlib

- **Machine Learning Routines in MLlib**:
  - **Classification**: The task of building ML models that learn from labeled data and predict the label of new data. SparkMLlib provide various number of ml algorithms like: Linear models, Naive Bayes, Decision trees.
  - **Clustering:** The task of finding patterns in data and grouping items into distinguished clusters. One of the clustering algorithms provided by Spark is K-means algorithm.
  - **Collaborative filtering:** A task of finding recommendations for users based on their previous behaviour and the relations between items. This relation is represented by a matrix of user-item relations.
  - Many other ML routines like: dimensionality reduction, feature extraction… etc.

Apache Spark MLlib

- **Machine Learning Routines in MLlib**: ***Classification example***
  - A file contains lines of daily forecast reads.
  - Each line represent a value 0/1 if it rains that day or not, and 3 values/features (wind, humidity, temperature). This file could be hdfs file distribute over a cluster.
  - The goal is to build ML model to predict rain using Naive Bayes algorithm.

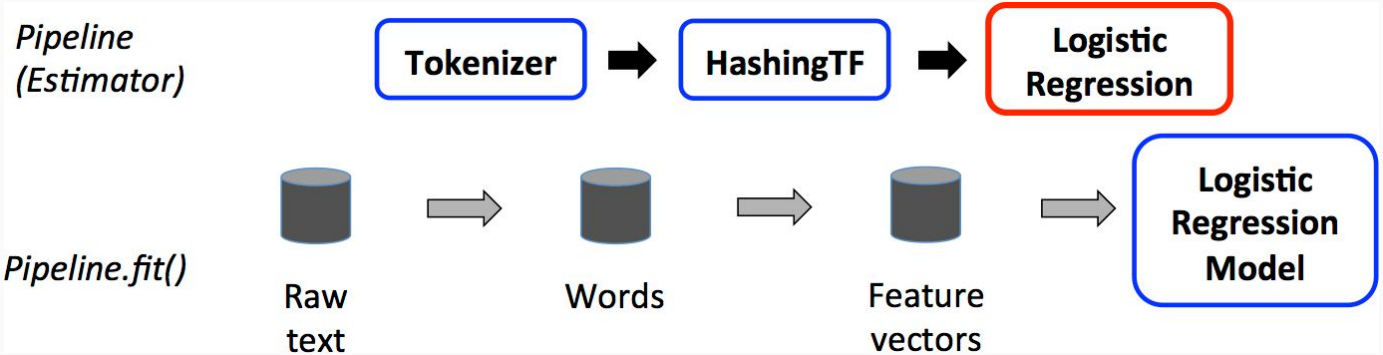
Demo Time!!

Apache Spark MLlib

- **Machine Learning PipeLines with spark.ml**:
  - Usually performing machine learning routine is not one simple task but it is done by executing sequence of machine learning tasks over the data to build the ML model.
  - Using Spark.ml you could build a **pipeline** of tasks(**tranformation**, **estimation**) that is being performed on a **dataset.**

Apache Spark MLlib

- **Machine Learning PipeLines Example**:



Apache Spark MLlib

- **Machine Learning PipeLines Example**: *text classification example*
  - Json file where each element represent a document (id, text, spam/not spam)
  - The task is to build a machine learning with the following steps (tokenization, weighting using hashingTF, learning a regression model).
  - The final goal is to build a machine learning classification model to prefect spam documents based on the their text column.

Demo Time!!

Apache Spark MLlib

- **Running Spark on YARN**:
  - Spark gives the ability to run Spark applications on YARN.
  - When Running Spark on YARN cluster. Resource management, scheduling and security are controlled by YARN.
  - In YARN, for each application running in the cluster there is an application manager associated with it. This application manager responsible for communicating with the resource manager to request resources and then allocate this resources. Then the application manager runs the tasks on the node managers.

Apache Spark MLlib

- **Running Spark on YARN**:

	Cluster Deployment mode	Client Deployment mode
Driver	**Runs in the application manager on the YARN cluster**	**Runs on the client where the job was submitted**
Request Resources	**Application Manager**	**Application Manager**
Start Executor process	**Application Manager**	**Application Manager**
Support Spark-shell	**No**	**Yes**

THE END