B.TECH. (CSE)
V SEMESTER

UE19CS322 – BIG DATA
PROJECT REPORT

ON

MACHINE LEARNING WITH SPARK MLLIB

SUBMITTED BY

NAME	SRN
1) A Spoorthi Alva	PES2UG19CS001
2) Amulya S Dinesh	PES2UG19CS035
3) Noorain Raza	PES2UG19CS269

AUGUST– DECEMBER 2021

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ELECTRONIC CITY CAMPUS,

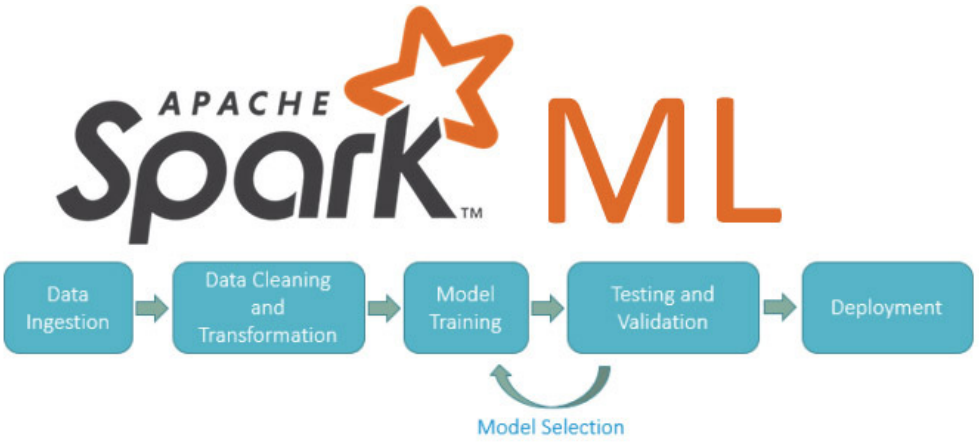
BENGALURU – 560100, KARNATAKA, INDIA

Spark MLlib is used to perform machine learning in Apache Spark. MLlib consists of popular algorithms and utilities. MLlib in Spark is a scalable Machine learning library that discusses both high-quality algorithms and high speed. The machine learning algorithms like regression, classification, clustering, pattern mining, and collaborative filtering. Lower level machine learning primitives like generic gradient descent optimization algorithms are also present in MLlib.

DATASET CHOSEN:

SPAM is the dataset we chose to work on. It has two csv files, namely, train.csv and test.csv. As the name indicates, they are used to train the model and test the model respectively. The origin of this dataset is https://www.kaggle.com/wanderfj/enron-spam.

DESIGN DETAILS:



Spark MLlib tools are

1. ML Algorithms-MLlib standardizes APIs to make it easier to combine multiple algorithms into a single pipeline, or workflow.
2. Featurization-includes feature extraction, transformation, dimensionality reduction, and selection.
3. Pipelines-A Pipeline chains multiple Transformers and Estimators together to specify an ML workflow. It also provides tools for constructing, evaluating and tuning ML Pipelines. Pipeline workflow will proceed in the following model
    - One hot encoder estimator
    - String indexer
    - Vector Assembler

4. Persistence-Persistence helps in saving and loading algorithms, models, and Pipelines.
5. Utilities-Utilities for linear algebra, statistics, and data handling. Example: mllib.linalg is MLlib utilities for linear algebra.

SURFACE LEVEL IMPLEMENTATION DETAILS:

- ➔ Preprocessing to convert csv to json, using stringtokenizer and TF-IDF
- ➔ Using RDD's for implementation
- ➔ Using the concept of ***NLP***
- ➔ Training the model for spam mail detection
- ➔ Testing the model after training

RDD is among the abstractions of Spark. Spark RDD handles partitioning data across all the nodes in a cluster. It holds them in the memory pool of the cluster as a single unit. There are two operations performed on RDDs:

- ➔ *Transformation*: It is a function that produces new RDD from the existing RDDs.
- ➔ *Action*: We trigger the transformation to proceed.

REASON BEHIND DESIGN DECISIONS:

ML Algorithms form the core of MLlib. These include common learning algorithms such as classification, regression, clustering, and collaborative filtering. MLlib standardizes APIs to make it easier to combine multiple algorithms into a single pipeline, or workflow. The key concepts are the Pipelines API, where the pipeline concept is inspired by the scikit-learn project.
So, on assessing the requirements to achieve the objective:

1. *Process huge amount of data*
2. *Input from multiple sources*
3. *Easy to use*
4. *Fast processing*

TAKE AWAY FROM THE PROJECT:

- ➔ Ability to analyze large data streams and process them for machine learning tasks using Spark Streaming and Spark MLLib to draw insights and deploy models for predictive tasks.
- ➔ The project helped us obtain an in-depth understanding of how applications in the real world work with large data streams