Twitter Sentiment Analysis Project Overview

What is Twitter Sentiment?

Twitter sentiment is a term used to define the analysis of sentiments in the tweets generated by users on social media platform like Twitter. Generally, twitter sentiments are analysed in most of the projects using parsing. Analyzing sentiments of users on twitter is fruitful to companies for their product that is mostly focused on social media trends, users sentiments and future view of the online community.

Data Pipeline:

It refers to a system for moving data from one system to another. The data may or may not be transformed, and it may be processed in real time (or streaming) instead of batches. Right from extracting or capturing data using various tools, storing raw data, cleaning, validating data, transforming data into query worthy format, visualisation of KPIs including Orchestration of the above process is data pipeline.

What is the Agenda of the project?

Agenda of the project involves Real-time streaming of Twitter Sentiments with visualization web app. We first launch an EC2 instance on AWS, and install Docker in it with tools like Apache Spark, Apache NiFi, Apache Kafka, Jupyter Lab, MongoDB, Plotly and Dash. Then, supervised classification model is created using Data exploration, Bucketizing, Stratified sampling, Dataset splitting, Extracting the features using tokenizing, removing stop words, TF-IDF etc., Creating Pipeline, Training the model, Evaluating model with binary classification evaluation and Saving classified model. It is followed by Extraction using Apache NiFi and Apache Kafka, followed by Transformation and Load using MongoDB and finally Visualizing it using python plotly and Dash with the usage of graph and table app call-back.

Usage of Dataset:

Here we are going to use Twitter sentiments data in the following ways:

- Extraction: During extraction process, NiFi process and connections are set up followed by creation of twitter app in twitter developer account. The data is streamed from the twitter API using NiFi followed by creation of topics and publishing tweets in NiFi using apache Kafka.

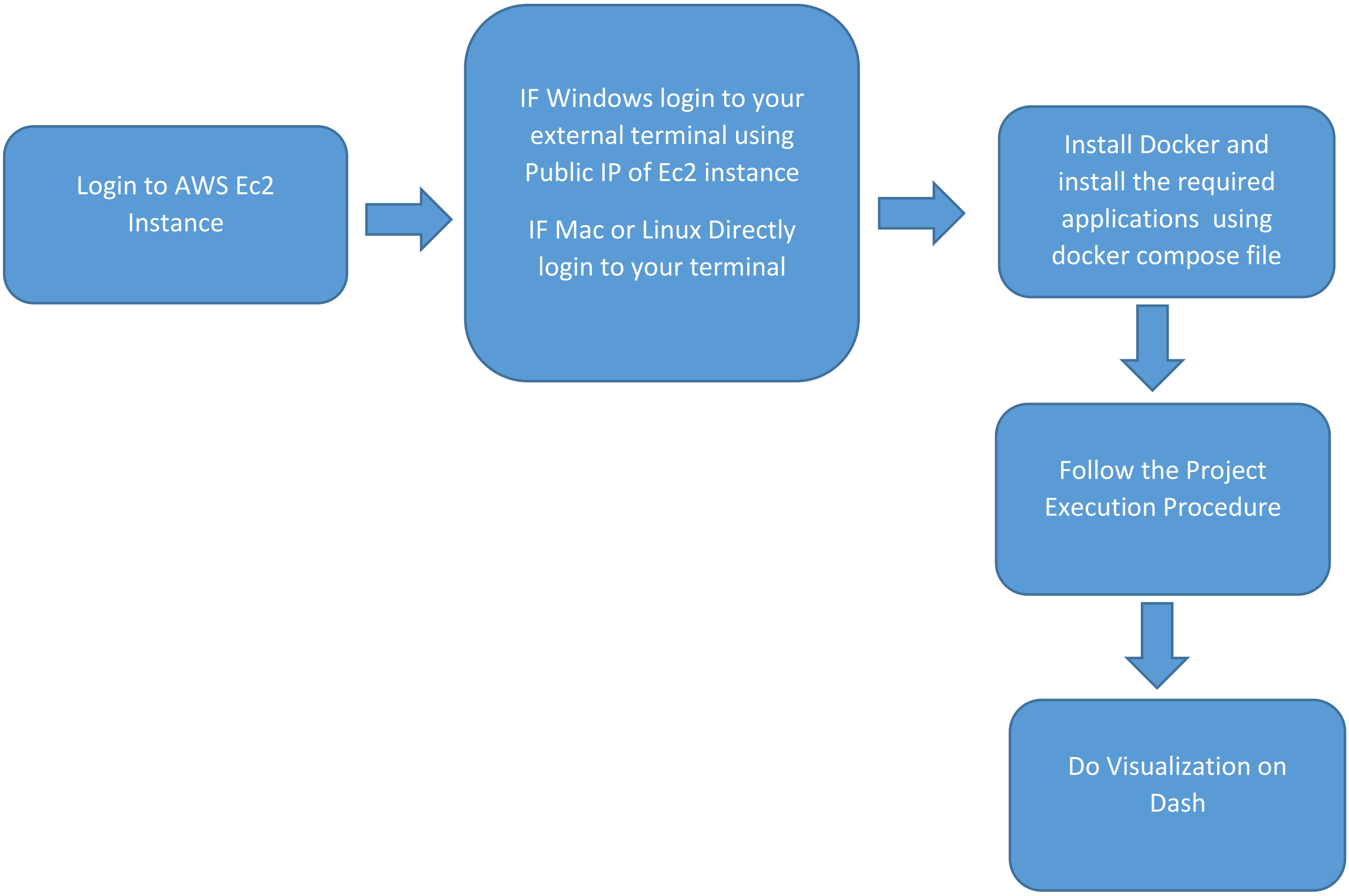
- Transformation and Load: During transformation and load process, schema is extracted from the stream of tweets followed by reading of data form apache Kafka as streaming a dataframe with extraction and cleansing of twitter data and analyzing sentiments in tweets. Then data is written in MongoDB for the visualization in Dash.

Data Analysis:

- From given website, data is downloaded containing text of review, rating of product and summary of review. Data is bucketized to label features followed by partitioning of data to homogenous sample..
- Dataset is splitted in appropriate ratios following by features extraction using tokenisation, TF-IDF and logistic regression.

- Data pipeline is created to train the model and evaluate it with binary classification evaluator followed by saving of classified model.
- The extraction process is done using NiFi and Kafka, by streaming data from twitter API using NiFi and creating topics, publishing tweets using Kafka.
- In transformation and load process, schema is extracted from twitter streams and data is read from Kafka as streaming dataframe.
- Twitter data is extracted and cleansed followed by sentiment analysis of tweets.
- Finally continuous data is loaded into MongoDB and data is visualized using scatter graph and table definitions in python plotly and Dash.

Project Workflow:


Login to AWS Ec2 Instance
IF Windows login to your external terminal using Public IP of Ec2 instance
IF Mac or Linux Directly login to your terminal
Install Docker and install the required applications using docker compose file
Follow the Project Execution Procedure
Do Visualization on Dash

Folder Structure:

Configuration & System Requirement:

Windows – External terminal is required to access the AWS from windows for e.g PUTTY

MAC OS Linux – Ubuntu

Ec2 Instance in AWS – t2.xLarge 32 GB RAM and Ubuntu 16.04 image on it

Docker Container:

docker-compose.yml

Installation:

Presentation.ipynb

Project Execution:

schemagenerator.ipynb,
sentimentanalyzer.ipynb,
streamlistener.ipynb,
sentimentvisualizer.ipynb

Tech Stack:

Docker version 3.0.0

NiFi version 1.3.2

Kafka version 2.8.0

Spark version 3.1.1

MongoDB version 4.4.5

Dash version 1.20.0