16:137:602:900C
Introduction to Cloud and Big Data Systems (Spring 2017)

Assignment 4: Project Report

Maruthi Ayyappan – Aishwarya Gunde – Beethoven Plaisir

Objectives of the project:
- **End-to-end use case using technologies covered in this course**
- **Combine streaming processing with batch processing**
- **Gain hands-on experience: Spark streaming programming**
  - **Analyze data sets using DStreams**

Step 1: Input data

Three main options
1. Default files through HDFS
– Data is being generated automatically (a few times a minute) in the following directory
/project/sensor*
– This stream of data provides data for 1000x1000 items (exhaustively)
2. Create your custom network-based data stream
3. Use Kafka and your own data stream

➔ For the input data, we created custom network-based data stream. The code is provided in the **codes.txt** file under the Step 1: Input data section.

Step 2: Online processing

• Compute wind speed variability in the last two minutes (sliding) window
– Use increases of 20 seconds for your window
– You will need to take, for each coordinate (x,y) the MAX and MIN values in that Window

• Save the result of algorithm for each window in HDFS
– Store the variability of wind speed (i.e., MAX – MIN) for each coordinate, for each window in HDFS.
– The format is open but please keep in mind that Step 3 requires processing file
➔ The online processing of computation of wind speed variability in the sliding window is provided in the **codes.txt** file under the Step 2: Online processing section.

Step 3: Batch processing

• Batch processing is under demand, i.e., you will execute this MapReduce code when needed.

• Process the data in the output file in HDFS from Step 2
– This file should contain that variability of wind speed in each point for a number of time windows

➔ The batch processing code is provided in the **codes.txt** file under the Step 3: Batch processing section.

• **Generate a heat map with the average wind speed variability for each coordinate (x,y)**
– Only a single heat map is expected
➔ The generated heat map is stored as **Heatmap.png**.

Other files:

- **input_output_average.txt:** It includes the input and output of each step.
- **Average.txt:** This is the final output file for creating heat map**.**
- **logs.txt:** It includes log of the codes.