Explanation for Spark Streaming in Lab2

Implementation details and the whole framework:

There are three parts in code.
1) Data source
2) Data processing
3) Data Storing

Data source: Kafka. Set configuration of Kafka and build integration with Spark Streaming. Then Use Receiver-less direct approach (by calling KafkaUtils.createDirectStream) to read data.

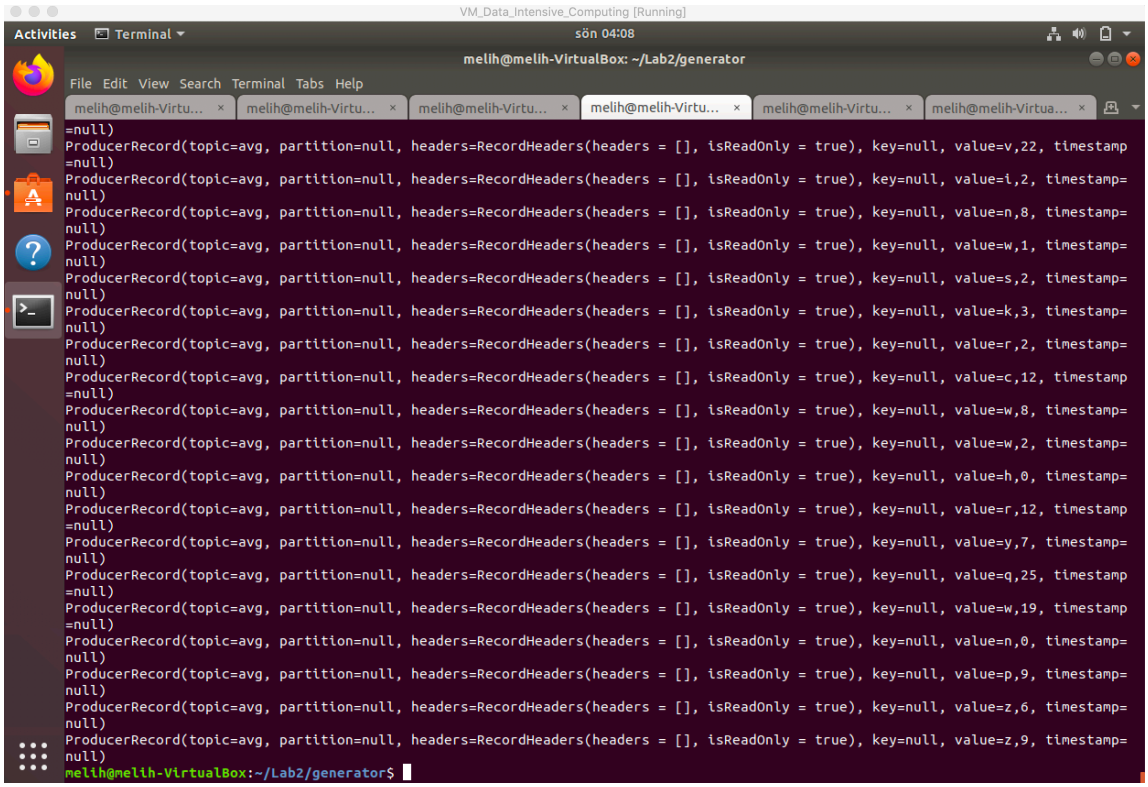
Data processing: Spark Streaming. Set configuration and create StreamingContext. Read data from Kafka as Dstream. Process it as (key, value) pairs by map function-> (String, Int). Use mapWithState to calculate the average. In update function (mappingFunc [key, value, state]), define state as (Double, Int). Int type stores old count, Double type stores old average. Then update state every time new data coming, and update the average value of each key by calculating newAvg = (oldAvg * oldCount + newVal)/newCount. (every time newCount = oldCount +1 if data exists)

Data Storing: Cassandra. Build keyspace("avg_space") and create table("avg") with columns "word: text", "count: float", set connection with Spark Streaming. Finally, use stateDstream.saveToCasandra to store the result in Cassandra.

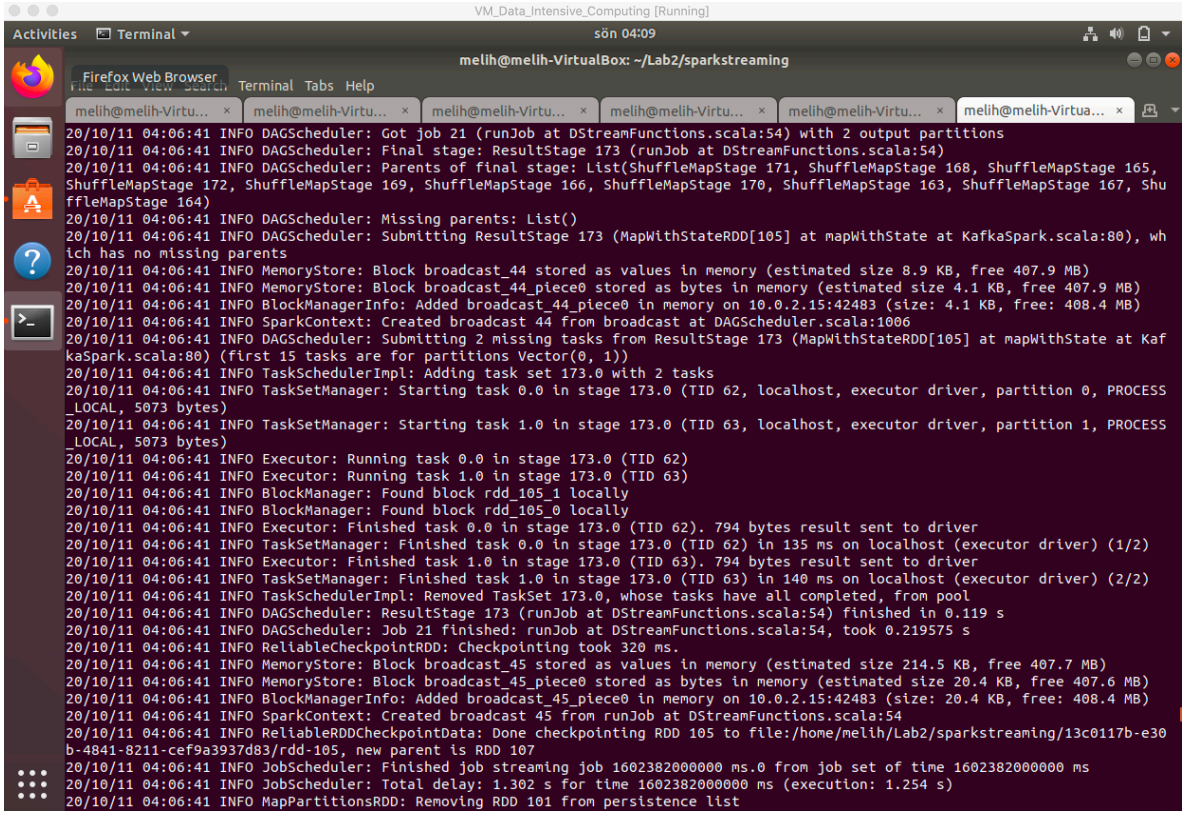
Run the code

Start Zookeeper -> start Kafka -> start Cassandra -> run spark streaming code by running sbt -> run generator code to produce data by running sbt -> check Cassandra table to show result

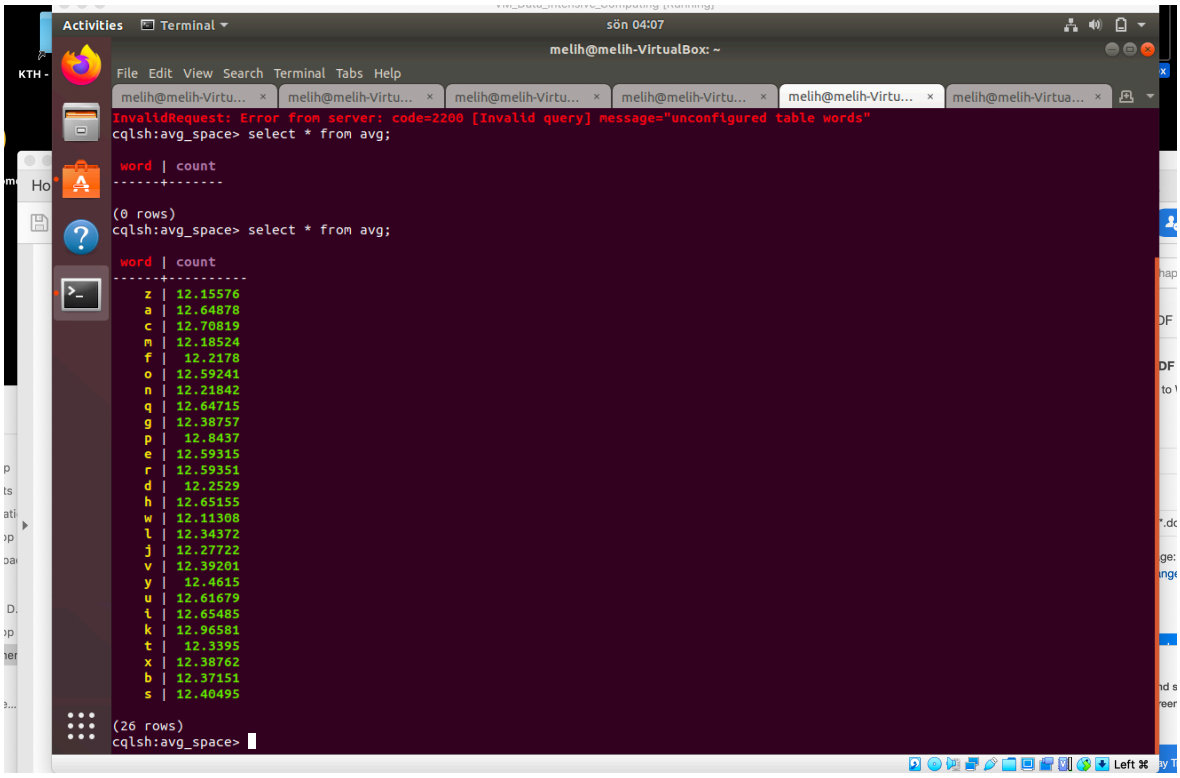
Result



Generating data



Spark Streaming running process



Final result in Cassandra