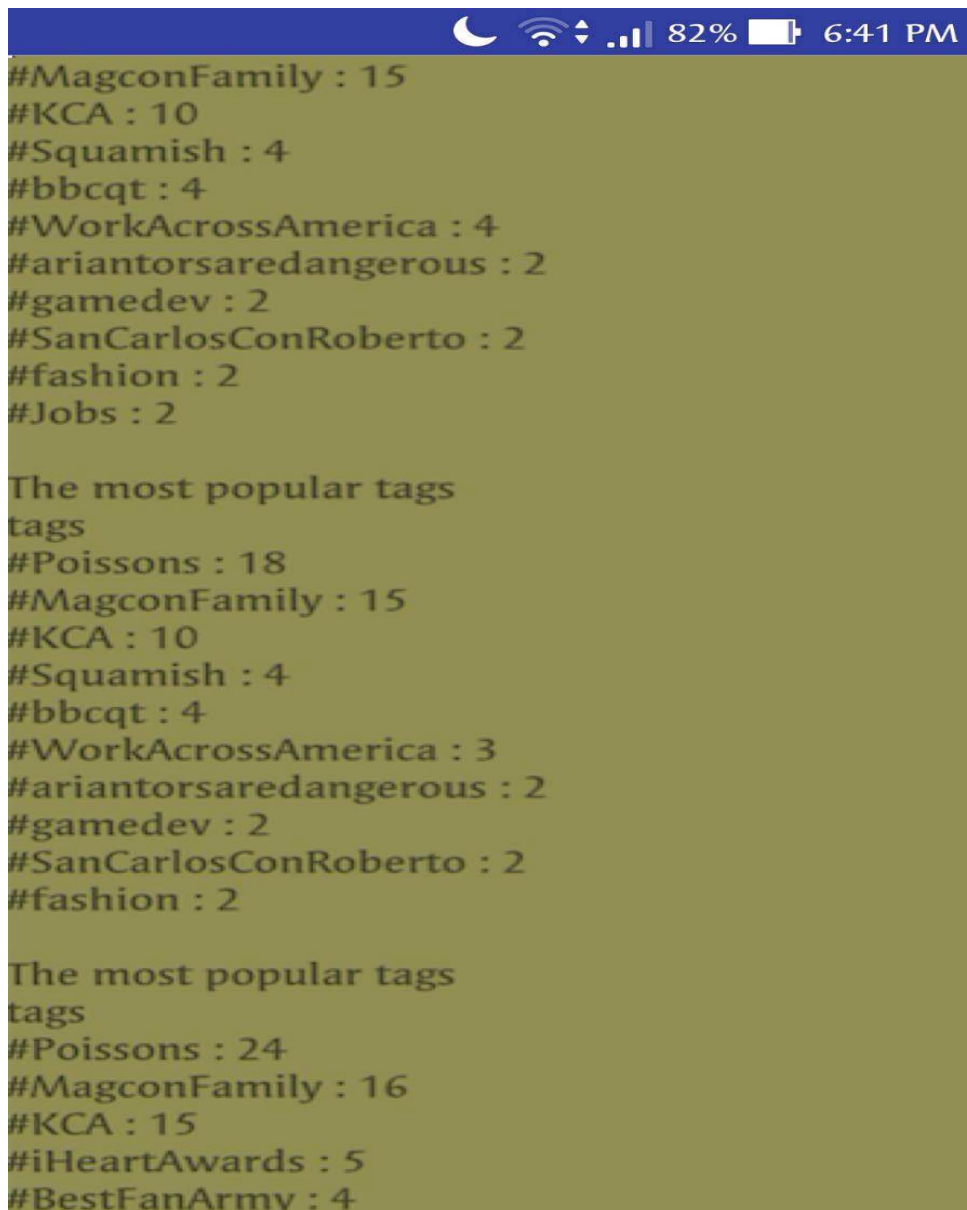


CS5542 Big Data Apps and Analytics

LAB ASSIGNMENT #5&6

1. Spark and Smartphone/Watch Application Implement a smart application with big data analytics related to your project showing the collaboration between Spark and Smart Apps. Implement Twitter Streaming and perform word count on it and publish the results and showcase it in your Smart Phone/Watch Application.



#MagconFamily : 8
#Directioners : 6
#bbcqt : 6
#GlFparty : 4
#quote : 3
#CombateRegaloDeAmorPolloyTete : 3

The most popular tags
tags

#Poissons : 24
#KCA : 15
#BestFanArmy : 10
#iHeartAwards : 10
#MagconFamily : 8
#Directioners : 6
#GlFparty : 4
#ALDUBRegaloNiLola : 4
#bbcqt : 4
#quote : 3

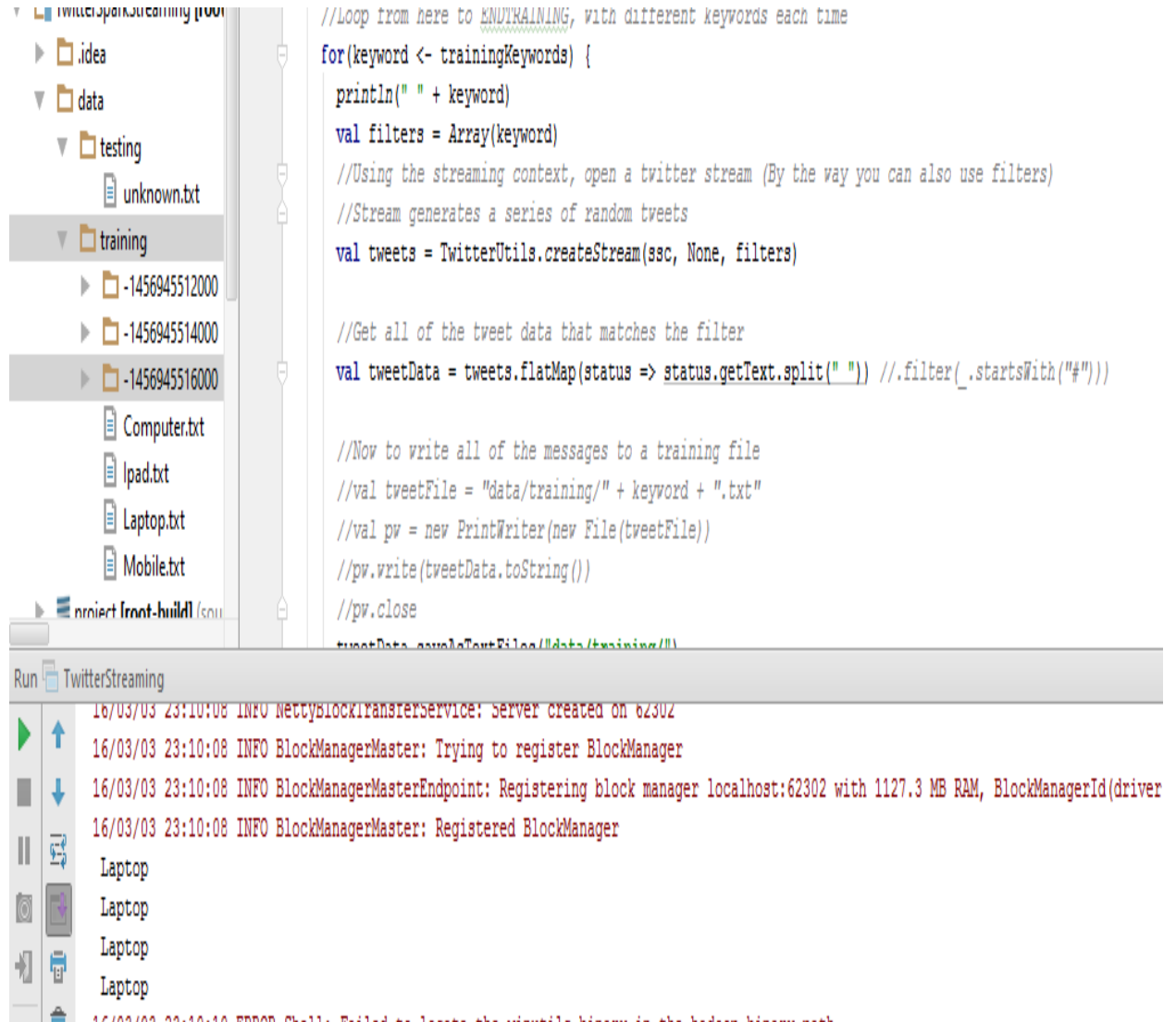
The most popular tags
tags

#Poissons : 24
#KCA : 17
#BestFanArmy : 10
#iHeartAwards : 10
#MagconFamily : 6
#Directioners : 6
#GlFparty : 4
#bbcqt : 4
#VotaLaliEsposito : 3

2.

Spark ML Lib Application Perform a machine learning algorithm with the Twitter Streaming data to categorize each Tweet

- a. Training datasets: Collect different categories of Tweets related to your project. (Categories can be based on HashTags / Subjects etc.)
- b. Test data: the upcoming twitter stream.



The screenshot shows an IDE with a project named 'TwitterStreaming'. The file explorer on the left shows a directory structure with 'data' containing 'testing' and 'training' subdirectories. The 'training' directory contains three files: '-1456945512000', '-1456945514000', and '-1456945516000'. The main editor shows a Scala script with the following code:

```
//Loop from here to ENDTRAINING, with different keywords each time
for(keyword <- trainingKeywords) {
  println(" " + keyword)
  val filters = Array(keyword)
  //Using the streaming context, open a twitter stream (By the way you can also use filters)
  //Stream generates a series of random tweets
  val tweets = TwitterUtils.createStream(ssc, None, filters)

  //Get all of the tweet data that matches the filter
  val tweetData = tweets.flatMap(status => status.getText.split(" ")) //filter(_.startsWith("#"))

  //Now to write all of the messages to a training file
  //val tweetFile = "data/training/" + keyword + ".txt"
  //val pw = new PrintWriter(new File(tweetFile))
  //pw.write(tweetData.toString())
  //pw.close
  tweetData.saveAsTextFiles("data/training/")
}
```

The console output shows the following logs:

```
Run TwitterStreaming
16/03/03 23:10:08 INFO NettyBlockTransferService: server created on 62302
16/03/03 23:10:08 INFO BlockManagerMaster: Trying to register BlockManager
16/03/03 23:10:08 INFO BlockManagerMasterEndpoint: Registering block manager localhost:62302 with 1127.3 MB RAM, BlockManagerId(driver
16/03/03 23:10:08 INFO BlockManagerMaster: Registered BlockManager
Laptop
Laptop
Laptop
Laptop
16/03/03 23:10:10 ERROR Shell: Failed to launch the driver binary for the driver binary path
```