Session 22: DEPLOYING A SPARK APPLICATION

Assignment 22.2

Student Name:	Abarajithan SA
Course:	Big Data Hadoop & Spark Training
Start Date:	2017-09-09
End Date:	2017-11-26

Assignment 22.2– DEPLOYING A SPARK APPLICATION

Contents

Introduction

In this assignment, we are going to analyze Sentiment analysis on demonetization

Problem Statement

Implement the below blog at your end and send the complete documentation.
https://docs.google.com/document/d/14YUd_wi-KJTBEqqtyoNtPMFeORjFU-_yId4hJ11q90o/edit

Sentiment analysis on demonetization

Let us find out the views of different people on the demonetization by analyzing the tweets from twitter. Here is the dataset where twitter tweets are gathered in CSV format. You can download the dataset from the below link

https://drive.google.com/open?id=0ByJLBTmJojjzNkRsZWJiY1VGc28

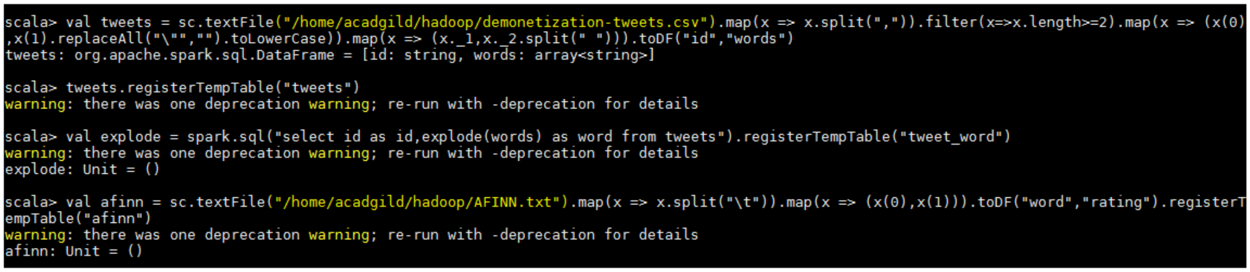
Source codes

val tweets = sc.textFile("/home/acadgild/hadoop/demonetization-tweets.csv").map(x => x.split(",")).filter(x=>x.length>=2).map(x => (x(0).replaceAll("\"",""),x(1).replaceAll("\"","").toLowerCase)).map(x => (x._1,x._2.split(" "))).toDF("id","words")

tweets.registerTempTable("tweets")

val explode = spark.sql("select id as id,explode(words) as word from tweets").registerTempTable("tweet_word")

val afinn = sc.textFile("/home/acadgild/hadoop/AFINN.txt").map(x => x.split("\t")).map(x => (x(0),x(1))).toDF("word","rating").registerTempTable("afinn")

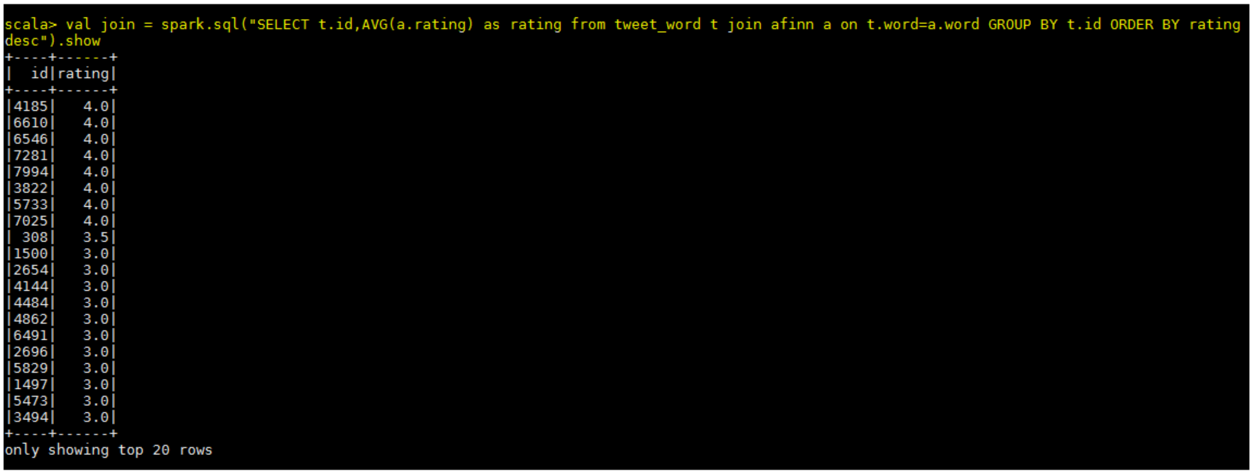


```
scala> val tweets = sc.textFile("/home/acadgild/hadoop/demonetization-tweets.csv").map(x => x.split(",")).filter(x=>x.length>=2).map(x => (x(0)
,x(1).replaceAll("\"","").toLowerCase)).map(x => (x._1,x._2.split(" "))).toDF("id","words")
tweets: org.apache.spark.sql.DataFrame = [id: string, words: array<string>]

scala> tweets.registerTempTable("tweets")
warning: there was one deprecation warning; re-run with -deprecation for details

scala> val explode = spark.sql("select id as id,explode(words) as word from tweets").registerTempTable("tweet_word")
warning: there was one deprecation warning; re-run with -deprecation for details
explode: Unit = ()

scala> val afinn = sc.textFile("/home/acadgild/hadoop/AFINN.txt").map(x => x.split("\t")).map(x => (x(0),x(1))).toDF("word","rating").registerT
empTable("afinn")
warning: there was one deprecation warning; re-run with -deprecation for details
afinn: Unit = ()
```

Desired result

val join = spark.sql("SELECT t.id,AVG(a.rating) as rating from tweet_word t join afinn a on t.word=a.word GROUP BY t.id ORDER BY rating desc").show

Expected Output



```
scala> val join = spark.sql("SELECT t.id,AVG(a.rating) as rating from tweet_word t join afinn a on t.word=a.word GROUP BY t.id ORDER BY rating
desc").show
+----+------+
|  id|rating|
+----+------+
|4185|   4.0|
|6610|   4.0|
|6546|   4.0|
|7281|   4.0|
|7994|   4.0|
|3822|   4.0|
|5733|   4.0|
|7025|   4.0|
| 308|   3.5|
|1500|   3.0|
|2654|   3.0|
|4144|   3.0|
|4484|   3.0|
|4862|   3.0|
|6491|   3.0|
|2696|   3.0|
|5829|   3.0|
|1497|   3.0|
|5473|   3.0|
|3494|   3.0|
+----+------+
only showing top 20 rows
```