Principles of Big Data Management

Phase-2 Report

Extracted tweets: https://drive.google.com/drive/u/0/my-drive

GitHub link: https://github.com/rahuldhar123/Twitter-Data-analysis/tree/master/Phase-2

Team Members:

1. Rahul Dhamerla - 16282037
2. Teja Devarapalli - 16282634
3. Nikita Goyal - 16285353

Phase-2: Analyzing and Visualizing of twitter data

Applications:

- Apache Spark SQL
- Python
- IntelliJ Idea IDE
- Twitter Developer Account

Collecting tweets from Twitter:

- Firstly, we have created a developer account in Twitter using below link. https://apps.twitter.com/
- Below are the variables that contains the user credentials to access Twitter API.
- access_token = "1089997729219178496-BvhcwdDw6eGPbPJM5wJYqOvsq7rE7G"
- access_token_secret = "xZu9ILsesdPiRi5quCxx94mQTfrj00sMEedGrddvYOmsQ"
- consumer_key = "f4qQNYSFaGxO4iB6SamtIZarf"
- consumer_secret = "PTuSFEtZAeRP9nov16hJTWQzFNedKTSDKR9ZSS1Iu65JiSbXvO"
- We have written python program that is used to fetch tweets in JSON format. (Tweetscollection.py)
- The tweet data is collected on the concept based on to analyze and visualize the data regarding various characters in Game of Thrones a TV Show.
- We have written 10 analytic queries and performed visualization on them.

Queries and Output:

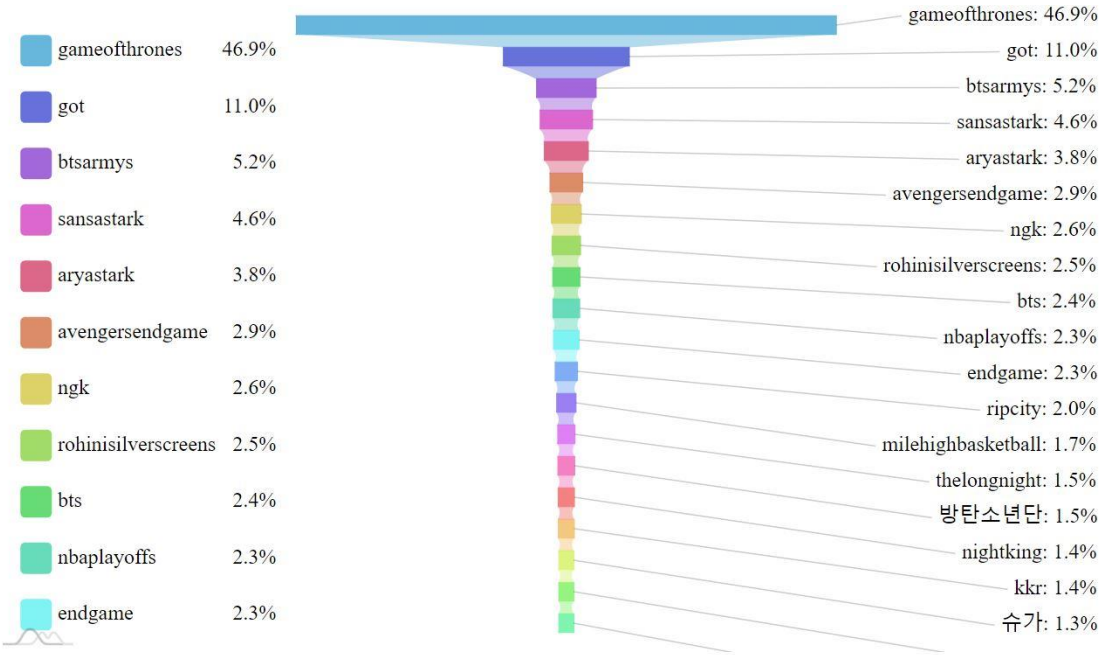
Query-1 : This query fetches the tweets on hashtags

```
val hashtag = sqlContext.sql("SELECT LOWER(hashtags.text) As Hashtags, COUNT(*) AS
total_count FROM tweets LATERAL VIEW EXPLODE(entities.hashtags) t1 AS hashtags GROUP
BY LOWER(hashtags.text) ORDER BY total_count DESC LIMIT 20")

hashtag.show()
```

Output:

```
hh C:\Users\rahul\IdeaProjects\hh        queries › main(args: Array[String])
Run:  queries
19/05/05 22:14:36 INFO TaskSetManager: Finished task 0.0 in stage 3.0 (TID 216) in 98 ms on localhost (executor driver) (1/1)
19/05/05 22:14:36 INFO TaskSchedulerImpl: Removed TaskSet 3.0, whose tasks have all completed, from pool
19/05/05 22:14:36 INFO DAGScheduler: ResultStage 3 (show at queries.scala:71) finished in 0.105 s
19/05/05 22:14:36 INFO DAGScheduler: Job 1 finished: show at queries.scala:71, took 10.376072 s
+-------------------+-----------+
|           Hashtags|total_count|
+-------------------+-----------+
|      gameofthrones|       3700|
|               got|        864|
|          btsarmys|        410|
|        sansastark|        361|
|         aryastark|        303|
|    avengersendgame|        226|
|               ngk|        207|
|rohinisilverscreens|        197|
|               bts|        188|
|       nbaplayoffs|        185|
|           endgame|        178|
|           ripcity|        155|
| milehighbasketball|        134|
|       thelongnight|        120|
|         방탄소년단|        117|
|         nightking|        113|
|               kkr|        112|
|              슈가|        105|
|              suga|        105|
|           rcbvsrh|        105|
+-------------------+-----------+
```

Visualization:



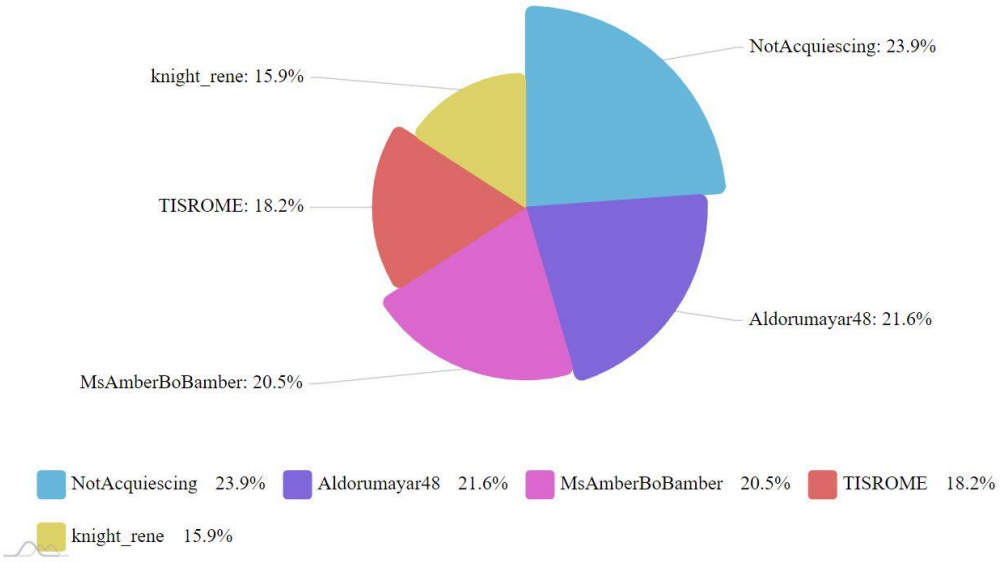
Query-2 : Which user tweeted most about which GOTCharacters-

```
val r1 = sqlContext.sql("SELECT UserName,GOTCharacters,count(*) as count FROM disCat2
WHERE GOTCharacters in ('AryaStark') group by UserName,GOTCharacters order by count
desc")
```

Output:

```
19/05/05 22:42:59 INFO TaskSetManager: Finished task 199.0 in stage 2.0 (TID 215) in 14 ms on localhost (executor driver) (200/200)
19/05/05 22:42:59 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
19/05/05 22:42:59 INFO DAGScheduler: ResultStage 2 (show at queries.scala:92) finished in 1.151 s
19/05/05 22:42:59 INFO DAGScheduler: Job 1 finished: show at queries.scala:92, took 6.428776 s
19/05/05 22:42:59 INFO CodeGenerator: Code generated in 10.7949 ms
+---------------+-------------+-----+
|       UserName|GOTCharacters|count|
+---------------+-------------+-----+
| GellingSabrina|    AryaStark|   12|
|EasterGenevieve|    AryaStark|    3|
|  AvatarJohnson|    AryaStark|    3|
|    EnLaRayaWeb|    AryaStark|    3|
|       kiwiokay|    AryaStark|    2|
|     nickerpops|    AryaStark|    2|
|      gogomi468|    AryaStark|    2|
| _jonacontreras|    AryaStark|    2|
|     OderoAlulu|    AryaStark|    2|
|   DionerTrejos|    AryaStark|    2|
|      evepachek|    AryaStark|    2|
|     Xoxo1992Kp|    AryaStark|    2|
|    ghkd_dlswns|    AryaStark|    1|
|     TMKJO_0110|    AryaStark|    1|
|   imgsynthesis|    AryaStark|    1|
|       nctylove|    AryaStark|    1|
|         Jnofsh|    AryaStark|    1|
|     jspark1127|    AryaStark|    1|
|       LeeN_C_T|    AryaStark|    1|
|      loveu_128|    AryaStark|    1|
+---------------+-------------+-----+
```

Visualization:



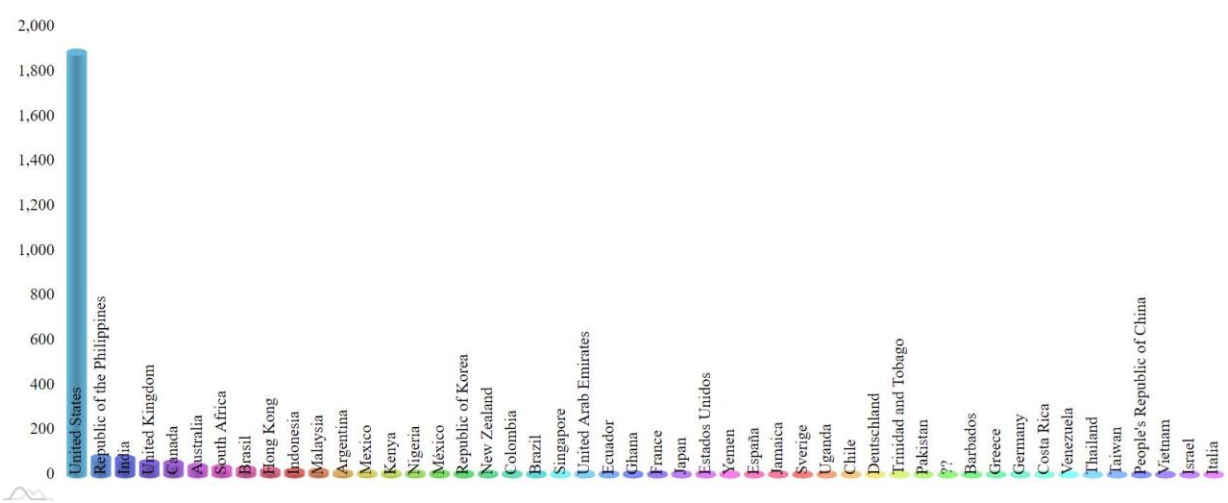
Query-3 : Tweets from different countries about GOT:

```
val countrytweetscount=sqlContext.sql("SELECT distinct place.country, count(*) as
count FROM tweets where place.country is not null " + "GROUP BY place.country ORDER BY
count DESC")
countrytweetscount.createOrReplaceTempView("countrytweetscount")
```

Output:

```
hh C:\Users\rahul\IdeaProjects\hh        queries > main(args: Array[String])
Run: queries
19/05/05 22:46:10 INFO DAGScheduler: Job 1 finished: show at queries.scala:102, took 6.776431 s
19/05/05 22:46:10 INFO CodeGenerator: Code generated in 11.4304 ms
+--------------------+-----+
|             country|count|
+--------------------+-----+
|       United States| 1887|
|Republic of the P...|   77|
|               India|   73|
|      United Kingdom|   54|
|              Canada|   52|
|           Australia|   37|
|        South Africa|   31|
|              Brasil|   25|
|           Hong Kong|   18|
|           Indonesia|   15|
|            Malaysia|   15|
|           Argentina|    9|
|              Mexico|    8|
|               Kenya|    8|
|             México|    7|
|             Nigeria|    7|
|         New Zealand|    6|
|   Republic of Korea|    6|
|United Arab Emirates|    5|
|              Brazil|    5|
+--------------------+-----+
only showing top 20 rows
```

Visualization:



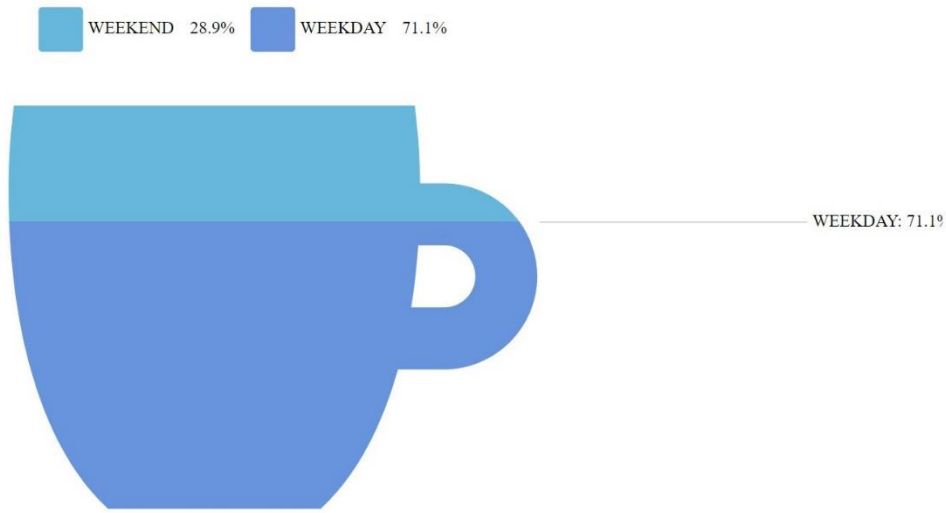
Query-4: Tweets count on different days.

```
val day_data = sqlContext.sql("SELECT substring(user.created_at,1,3) as day from
tweets where text is not null")

day_data.createOrReplaceTempView("day_data")

val days_final = sqlContext.sql(
  """ SELECT Case
    |when day LIKE '%Mon%' then 'WEEKDAY'
    |when day LIKE '%Tue%' then 'WEEKDAY'
    |when day LIKE '%Wed%' then 'WEEKDAY'
    |when day LIKE '%Thu%' then 'WEEKDAY'
    |when day LIKE '%Fri%' then 'WEEKDAY'
    |when day LIKE '%Sat%' then 'WEEKEND'
    |when day LIKE '%Sun%' then 'WEEKEND'
    | else
    | null
    | end as day1 from day_data where day is not null""".stripMargin)

days_final.createOrReplaceTempView("days_final")

val res = sqlContext.sql("SELECT day1 as Day,Count(*) as Day_Count from days_final
where day1 is not null group by day1 order by count(*) desc")
```

Output:

```
19/05/05 22:48:22 INFO Executor: Finished task 178.0 in stage 2.0 (TID 192). 4839 bytes result sent to driver
19/05/05 22:48:22 INFO TaskSetManager: Finished task 178.0 in stage 2.0 (TID 192) in 135 ms on localhost (executor driver) (198/200)
19/05/05 22:48:22 INFO Executor: Finished task 120.0 in stage 2.0 (TID 215). 5052 bytes result sent to driver
19/05/05 22:48:22 INFO Executor: Finished task 45.0 in stage 2.0 (TID 214). 5138 bytes result sent to driver
19/05/05 22:48:22 INFO TaskSetManager: Finished task 120.0 in stage 2.0 (TID 215) in 47 ms on localhost (executor driver) (199/200)
19/05/05 22:48:22 INFO TaskSetManager: Finished task 45.0 in stage 2.0 (TID 214) in 63 ms on localhost (executor driver) (200/200)
19/05/05 22:48:22 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
19/05/05 22:48:22 INFO DAGScheduler: ResultStage 2 (show at queries.scala:131) finished in 0.950 s
19/05/05 22:48:22 INFO DAGScheduler: Job 1 finished: show at queries.scala:131, took 4.446624 s
19/05/05 22:48:22 INFO CodeGenerator: Code generated in 11.7977 ms
+-------+---------+
|    Day|Day_Count|
+-------+---------+
|WEEKDAY|   104394|
|WEEKEND|    42423|
+-------+---------+

19/05/05 22:48:22 INFO FileSourceStrategy: Pruning directories with:
19/05/05 22:48:22 INFO FileSourceStrategy: Post-Scan Filters: isnotnull(user#41),isnotnull(text#38),isnotnull(substring(user#41.created_at, 1,
19/05/05 22:48:22 INFO FileSourceStrategy: Output Data Schema: struct<text: string, user: struct<contributors_enabled: boolean, created_at: st
19/05/05 22:48:22 INFO FileSourceScanExec: Pushed Filters: IsNotNull(user),IsNotNull(text)
```

Visualization:



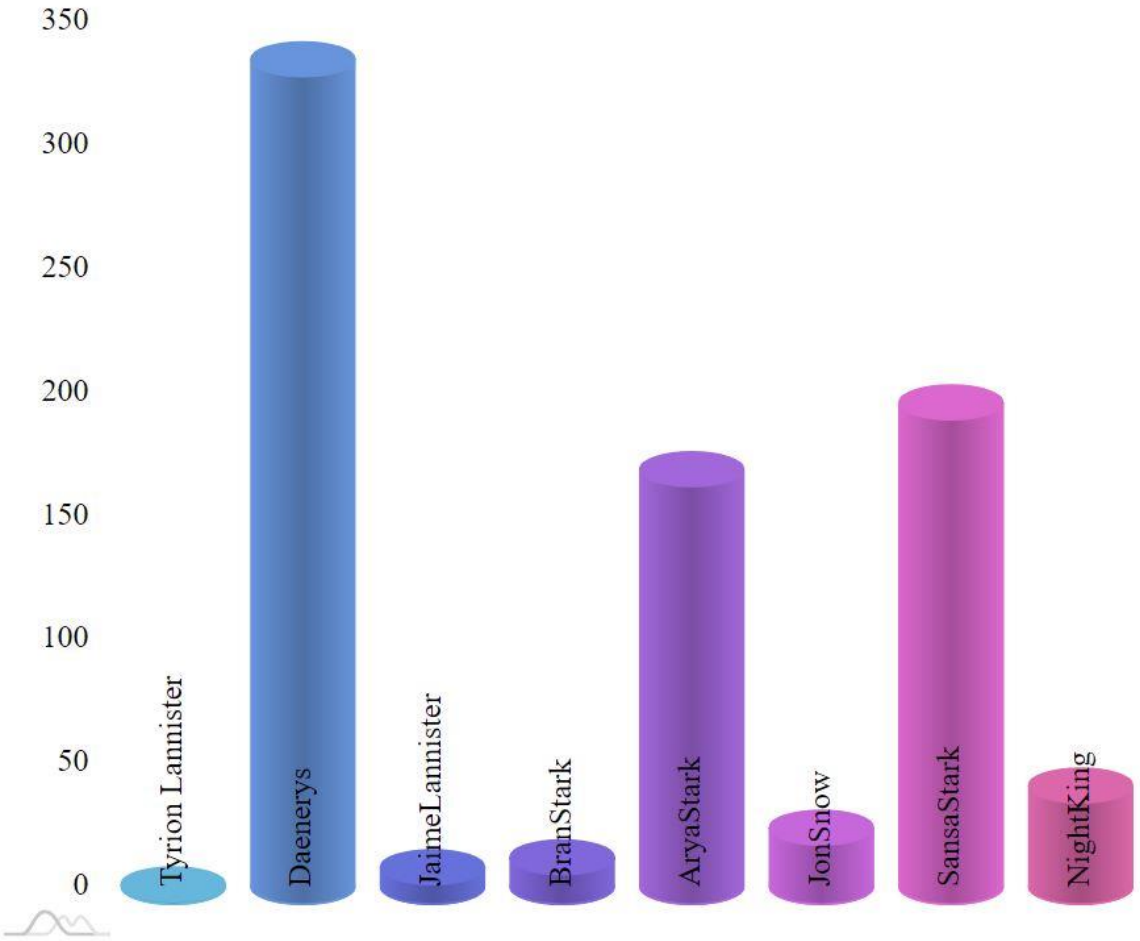
Query-5: Tweets count for different types of Characters

```
val r1 = sqlContext.sql("select loc,GotCharacters,count(*) as count from disCat2 " +
  "group by loc,GotCharacters order by count desc")
```

Output:

```
19/05/05 22:50:41 INFO Executor: Finished task 74.0 in stage 22.0 (TID 620). 4182 bytes result sent to driver
19/05/05 22:50:41 INFO Executor: Finished task 71.0 in stage 22.0 (TID 617). 4182 bytes result sent to driver
19/05/05 22:50:41 INFO Executor: Finished task 29.0 in stage 22.0 (TID 622). 4229 bytes result sent to driver
19/05/05 22:50:41 INFO TaskSetManager: Finished task 66.0 in stage 22.0 (TID 613) in 22 ms on localhost (execut
19/05/05 22:50:41 INFO TaskSetManager: Finished task 74.0 in stage 22.0 (TID 620) in 11 ms on localhost (execut
19/05/05 22:50:41 INFO TaskSetManager: Finished task 71.0 in stage 22.0 (TID 617) in 25 ms on localhost (execut
19/05/05 22:50:41 INFO TaskSetManager: Finished task 29.0 in stage 22.0 (TID 622) in 14 ms on localhost (execut
19/05/05 22:50:41 INFO TaskSchedulerImpl: Removed TaskSet 22.0, whose tasks have all completed, from pool
19/05/05 22:50:41 INFO DAGScheduler: ResultStage 22 (show at queries.scala:158) finished in 0.150 s
19/05/05 22:50:41 INFO DAGScheduler: Job 6 finished: show at queries.scala:158, took 0.154621 s
19/05/05 22:50:41 INFO BlockManagerInfo: Removed broadcast_5_piece0 on DESKTOP-6VFTPQA:49172 in memory (size: 1
+----------------+-----+
|   GOTCharacters|Count|
+----------------+-----+
|Tyrion Lannister|    1|
|        Daenerys|  335|
|  JaimeLannister|    8|
|       BranStark|   12|
|       AryaStark|  169|
|         JonSnow|   24|
|      SansaStark|  196|
|       NightKing|   41|
+----------------+-----+

19/05/05 22:50:41 INFO ContextCleaner: Cleaned accumulator 125
```

Visualization:



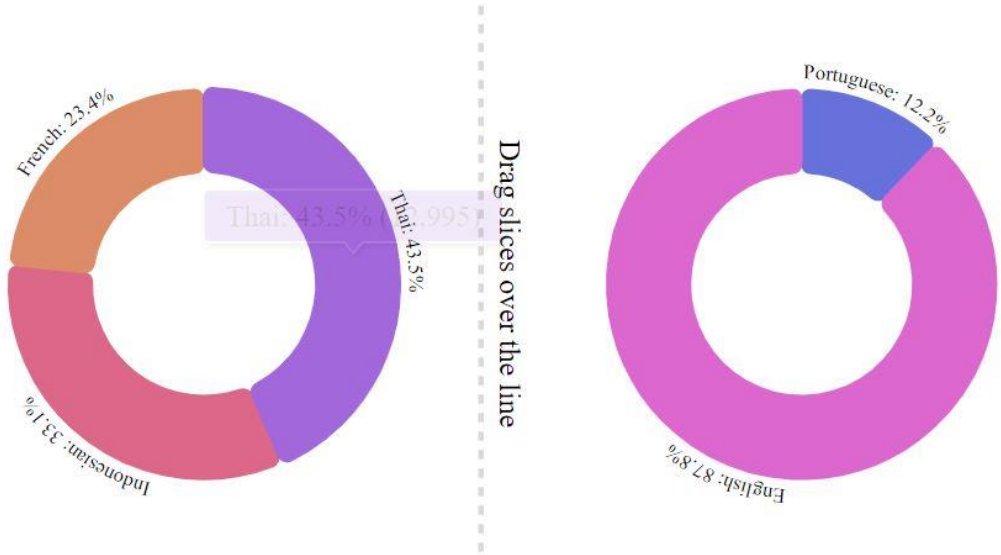
Query-6: Popular languages used for tweets about GOT

```
val langWstCount = sqlContext.sql("SELECT distinct id," +
  "CASE when user.lang LIKE '%en%' then 'English'"+
  "when user.lang LIKE '%ja%' then 'Japanese'"+
  "when user.lang LIKE '%es%' then 'Spanish'"+
  "when user.lang LIKE '%fr%' then 'French'"+
    "when user.lang LIKE '%vi%' then 'Vietnamese'"+
  "when user.lang LIKE '%zh-cn%' then 'Chinese (Simplified)'"+
  "when user.lang LIKE '%zh-tw%' then 'Chinese (Traditional)'"+
  "END AS language from tweets where text is not null")
langWstCount.createOrReplaceTempView("langWstCount")
var langWstDataCount=sqlContext.sql("SELECT language, Count(language) as Count from
langWstCount where id is NOT NULL and language is not null group by language order by
Count DESC")
```

Output:

```
19/05/05 22:57:28 INFO DAGScheduler: Job 1 finished: show at queries.scala:205, took 18.091917 s
19/05/05 22:57:28 INFO CodeGenerator: Code generated in 14.7028 ms
+--------------------+------+
|            language| Count|
+--------------------+------+
|             English|134455|
|             Spanish|  4381|
|          Portuguese|  1857|
|                Thai|  1295|
|          Indonesian|   999|
|              French|   703|
|            Japanese|   539|
|              Korean|   535|
|              German|   291|
|             Russian|   219|
|             Italian|   191|
|              Arabic|   178|
|          Vietnamese|   173|
|             Turkish|   170|
|             Swedish|   126|
|              Polish|    70|
|               Dutch|    67|
|Chinese (Traditio...|    38|
|             Finnish|    36|
|            Romanian|    35|
+--------------------+------+
only showing top 20 rows
```

Visualization:



Query-7: Retweet Count

```
val retweetcount=sqlContext.sql("SELECT user.screen_name, COUNT(*) as total FROM
tweets WHERE retweeted_status.user is not null GROUP BY user.screen_name ORDER BY
total desc LIMIT 5")
retweetcount.createOrReplaceTempView("retweetcount")
```

Output:

```
19/05/05 22:59:14 INFO Executor: Finished task 0.0 in stage 3.0 (TID 216). 2980 bytes result sent to driv
19/05/05 22:59:14 INFO TaskSetManager: Finished task 0.0 in stage 3.0 (TID 216) in 91 ms on localhost (ex
19/05/05 22:59:14 INFO TaskSchedulerImpl: Removed TaskSet 3.0, whose tasks have all completed, from pool
19/05/05 22:59:14 INFO DAGScheduler: ResultStage 3 (show at queries.scala:214) finished in 0.099 s
19/05/05 22:59:14 INFO DAGScheduler: Job 1 finished: show at queries.scala:214, took 15.975411 s
+---------------+-----+
|    screen_name|total|
+---------------+-----+
| NotAcquiescing|   21|
|  Aldorumayar48|   19|
|MsAmberBoBamber|   18|
|        TISROME|   16|
|    knight_rene|   14|
+---------------+-----+

19/05/05 22:59:14 INFO FileSourceStrategy: Pruning directories with:
19/05/05 22:59:14 INFO FileSourceStrategy: Post-Scan Filters: isnotnull(retweeted_status#36),isnotnull(re
19/05/05 22:59:14 INFO FileSourceStrategy: Output Data Schema: struct<retweeted_status: struct<contributo
19/05/05 22:59:14 INFO FileSourceScanExec: Pushed Filters: IsNotNull(retweeted_status)
```

Visualization:

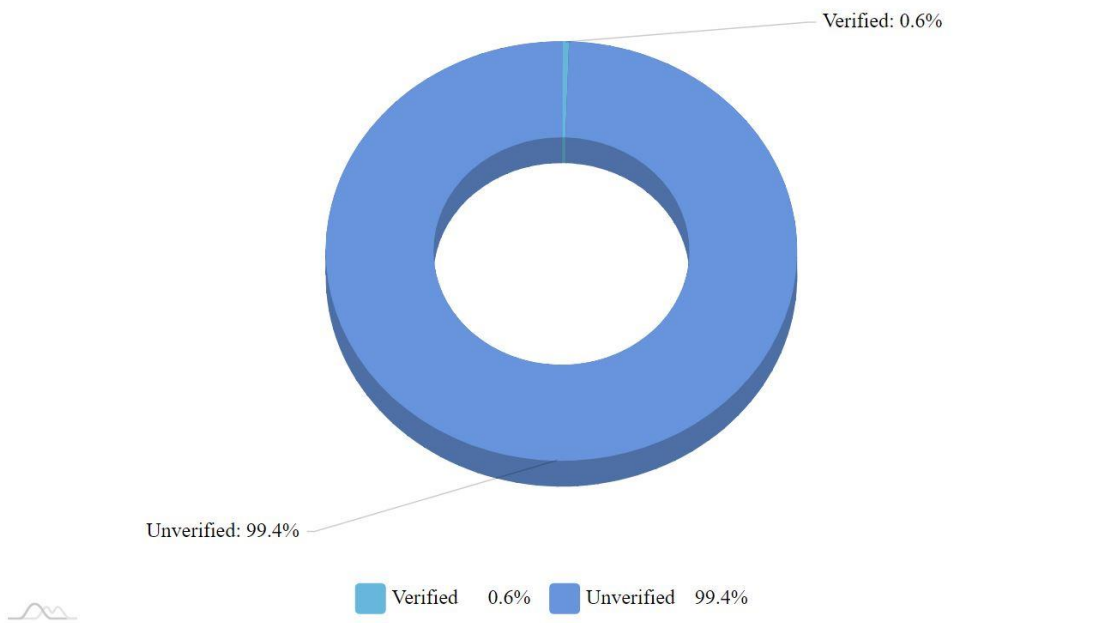
Query-8: Account Verification

```
var acctVerifydata=sqlContext.sql("select sum(case when user.verified = true then 1
else 0 end)Verified,sum(case when user.verified = false then 1 else 0 end)Unverified
from tweets")
```

Output:

```
19/05/05 23:03:45 INFO Executor: Running task 0.0 in stage 2.0 (TID 16)
19/05/05 23:03:45 INFO ShuffleBlockFetcherIterator: Getting 8 non-empty blocks including 8 local blocks and 0 remote block
19/05/05 23:03:45 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 8 ms
19/05/05 23:03:45 INFO Executor: Finished task 0.0 in stage 2.0 (TID 16). 1590 bytes result sent to driver
19/05/05 23:03:45 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 16) in 65 ms on localhost (executor driver) (1/
19/05/05 23:03:45 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
19/05/05 23:03:45 INFO DAGScheduler: ResultStage 2 (show at queries.scala:228) finished in 0.074 s
19/05/05 23:03:45 INFO DAGScheduler: Job 1 finished: show at queries.scala:228, took 3.123923 s
+--------+----------+
|Verified|Unverified|
+--------+----------+
|     814|    146003|
+--------+----------+

19/05/05 23:03:45 INFO FileSourceStrategy: Pruning directories with:
19/05/05 23:03:45 INFO FileSourceStrategy: Post-Scan Filters:
19/05/05 23:03:45 INFO FileSourceStrategy: Output Data Schema: struct<user: struct<contributors_enabled: boolean, created_
19/05/05 23:03:45 INFO FileSourceScanExec: Pushed Filters:
Exception in thread "main" org.apache.spark.sql.AnalysisException: path file:/C:/Users/rahul/Desktop/PB_Spring2019/Project
```

Visualization:



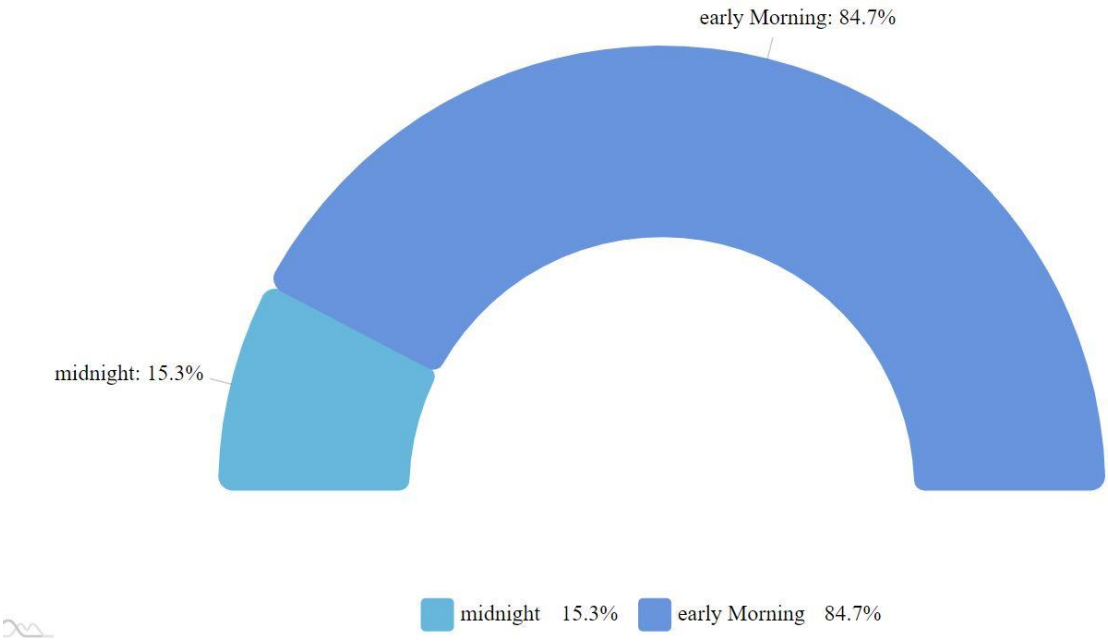
Query-9: On which hours tweets flow is high

```
val timehour = sqlContext.sql("SELECT SUBSTRING(created_at,12,2) as hour from tweets
where text is not null")

timehour.createOrReplaceTempView("timehour")
```

Output:

```
19/05/05 23:05:19 INFO TaskSetManager: Finished task 198.0 in stage 2.0 (TID 213) in 16 ms on localhost (executor driver) (196/200)
19/05/05 23:05:19 INFO TaskSetManager: Finished task 187.0 in stage 2.0 (TID 202) in 61 ms on localhost (executor driver) (197/200)
19/05/05 23:05:19 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 53 ms
19/05/05 23:05:19 INFO Executor: Finished task 195.0 in stage 2.0 (TID 210). 4753 bytes result sent to driver
19/05/05 23:05:19 INFO TaskSetManager: Finished task 195.0 in stage 2.0 (TID 210) in 59 ms on localhost (executor driver) (198/200)
19/05/05 23:05:19 INFO Executor: Finished task 199.0 in stage 2.0 (TID 215). 4966 bytes result sent to driver
19/05/05 23:05:19 INFO TaskSetManager: Finished task 199.0 in stage 2.0 (TID 215) in 36 ms on localhost (executor driver) (199/200)
19/05/05 23:05:19 INFO Executor: Finished task 140.0 in stage 2.0 (TID 214). 4974 bytes result sent to driver
19/05/05 23:05:19 INFO TaskSetManager: Finished task 140.0 in stage 2.0 (TID 214) in 48 ms on localhost (executor driver) (200/200)
19/05/05 23:05:19 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
19/05/05 23:05:19 INFO DAGScheduler: ResultStage 2 (show at queries.scala:253) finished in 1.053 s
19/05/05 23:05:19 INFO DAGScheduler: Job 1 finished: show at queries.scala:253, took 4.626268 s
19/05/05 23:05:19 INFO CodeGenerator: Code generated in 14.4368 ms
+------------+------------+
|        hour|tweets_count|
+------------+------------+
|earlymorning|      144205|
|    midnight|        2612|
+------------+------------+

19/05/05 23:05:19 INFO FileSourceStrategy: Pruning directories with:
19/05/05 23:05:19 INFO FileSourceStrategy: Post-Scan Filters: isnotnull(text#38),isnotnull(CASE WHEN ((cast(substring(created_at#8, 12, 2) as int) >= (
19/05/05 23:05:19 INFO FileSourceStrategy: Output Data Schema: struct<created_at: string, text: string>
19/05/05 23:05:19 INFO FileSourceScanExec: Pushed Filters: IsNotNull(text)
Exception in thread "main" org.apache.spark.sql.AnalysisException: path file:/C:/Users/rahul/Desktop/PB Spring2019/Project/Phase-2/query9 already exis
    at org.apache.spark.sql.execution.datasources.InsertIntoHadoopFsRelationCommand.run(InsertIntoHadoopFsRelationCommand.scala:114)
```

Visualization:



Query-10: Tweets from different states about a particular GOT character

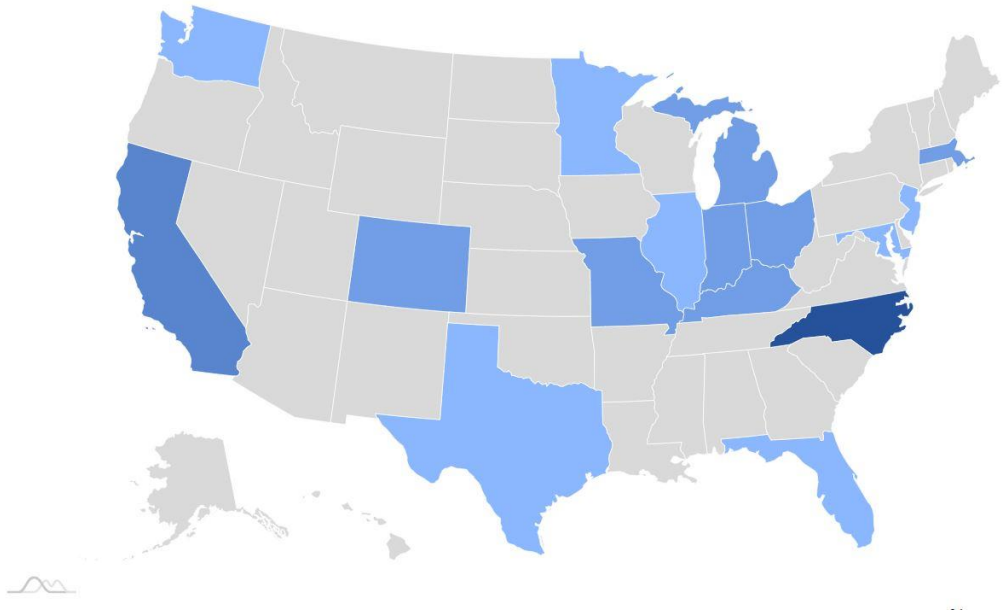
```
val AryaStarkRDD =  sqlContext.sql("""  SELECT 'AryaStark' as GOTCharacters,
user.location as loc from tweets where text LIKE '%#AryaStark%' """)
```

Output:

```
19/05/05 23:08:08 INFO TaskSchedulerImpl: Removed TaskSet 13.0, whose tasks have all completed, from pool
19/05/05 23:08:08 INFO DAGScheduler: ResultStage 13 (show at queries.scala:481) finished in 0.186 s
19/05/05 23:08:08 INFO DAGScheduler: Job 6 finished: show at queries.scala:481, took 0.192047 s
+-------------+-----+----------+
|GOTCharacters|state|type_count|
+-------------+-----+----------+
|    AryaStark|   IN|         2|
|    AryaStark|   CA|         3|
|    AryaStark|   WA|         1|
|    AryaStark|   CO|         2|
|    AryaStark|   TX|         1|
|    AryaStark|   MN|         1|
|    AryaStark|   MA|         2|
|    AryaStark|   MD|         1|
|    AryaStark|   KY|         2|
|    AryaStark|   OH|         2|
|    AryaStark|   IL|         1|
|    AryaStark|   MO|         2|
|    AryaStark|   NC|         5|
|    AryaStark|   MI|         2|
|    AryaStark|   NJ|         1|
|    AryaStark|   FL|         1|
+-------------+-----+----------+
```

4: Run 6: TODO Build SBT Console sbt shell Terminal

Visualization:



Testing:

- On taking the output of a query, we have checked the table data in online to the count we received in the group by query are equal and if there are any discrepancies and resolved missing data with naming conventions of the characters.
- Unit Testing.